DQM: Decentralized Quadratically Approximated Alternating Direction Method of Multipliers

Aryan Mokhtari, Wei Shi, Qing Ling, and Alejandro Ribeiro

***Abstract*—This paper considers decentralized consensus optimization problems where nodes of a network have access to different summands of a global objective function. Nodes cooperate to minimize the global objective by exchanging information with neighbors only. A decentralized version of the alternating directions method of multipliers (DADMM) is a common method for solving this category of problems. DADMM exhibits linear convergence rate to the optimal objective for strongly convex functions but its implementation requires solving a convex optimization problem at each iteration. This can be computationally costly and may result in large overall convergence times. The decentralized quadratically approximated ADMM algorithm (DQM), which minimizes a quadratic approximation of the objective function that DADMM minimizes at each iteration, is proposed here. The consequent reduction in computational time is shown to have minimal effect on convergence properties. Convergence still proceeds at a linear rate with a guaranteed factor that is asymptotically equivalent to the DADMM linear convergence rate factor. Numerical results demonstrate advantages of DQM relative to DADMM and other alternatives in a logistic regression problem.**

***Index Terms*—Alternating direction method of multipliers, decentralized optimization, multi-agent network.**

I. Introduction

DECENTRALIZED algorithms are used to solve optimization problems where components of the objective are available at different nodes of a network. Nodes access their local cost functions only but try to minimize the aggregate cost by exchanging information with their neighbors. Specifically, consider a variable $\tilde{\mathbf{x}} \in \mathbb{R}^p$ and a connected network containing n nodes each of which has access to a local cost function $f_i : \mathbb{R}^p \to \mathbb{R}$. The nodes' goal is to find the optimal argument of the global cost function $\sum_{i=1}^n f_i(\tilde{\mathbf{x}})$,

$$\tilde{\mathbf{x}}^* = \operatorname*{argmin}_{\tilde{\mathbf{x}}} \sum_{i=1}^{n} f_i(\tilde{\mathbf{x}}). \tag{1}$$

Problems of this form arise in, e.g., decentralized control [2]–[4], wireless systems [5]–[7], sensor networks [8]–[11], and large scale machine learning [12]–[14]. In this paper we assume that the local costs f_i are twice differentiable and strongly convex.

There are different algorithms to solve (1) in a decentralized manner which can be divided into two major categories. The ones that operate in the primal domain and the ones that operate in the dual domain. Among primal domain algorithms, decentralized (sub)gradient descent (DGD) methods are well studied [15]–[17]. They can be interpreted as either a mix of local gradient descent steps with successive averaging or as a penalized version of (1) with a penalty term that encourages agreement between adjacent nodes. This latter interpretation has been exploited to develop the network Newton (NN) methods that attempt to approximate the Newton step of this penalized objective in a distributed manner [18], [19]. The methods that operate in the dual domain consider a constraint that enforces equality between nodes' variables. They then ascend on the dual function to find optimal Lagrange multipliers with the solution of (1) obtained as a byproduct [8], [20]–[27]. Among dual descent methods, decentralized implementation of the alternating directions method of multipliers (ADMM), known as DADMM, is proven to be very efficient with respect to convergence time [8], [20]–[23], [28].

A fundamental distinction between primal methods such as DGD and NN and dual domain methods such as DADMM is that the former compute local gradients and Hessians at each iteration while the latter minimize local pieces of the Lagrangian at each step—this is necessary since the gradient of the dual function is determined by Lagrangian minimizers. Thus, iterations in dual domain methods are, in general, more costly because they require solution of a convex optimization problem. However, dual methods also converge in a smaller number of iterations because they compute approximations to $\tilde{\mathbf{x}}^*$ instead of descending towards $\tilde{\mathbf{x}}^*$. Having complementary advantages, the choice between primal and dual methods depends on the relative cost of computation and communication for specific problems and platforms. Alternatively, one can think of developing methods that combine the advantages of ascending in the dual domain without requiring solution of an optimization problem at each iteration. This can be accomplished by the decentralized linearized ADMM (DLM) algorithm [29], [30], which replaces the minimization of a convex objective required by ADMM with the minimization of a first order linear approximation of the objective function. This yields per-iteration problems that can be solved with a computational cost akin to the computation of a gradient and a method with convergence properties closer to DADMM than DGD.

If a first order approximation of the objective is useful, a second order approximation should decrease convergence times

Manuscript received July 08, 2015; revised January 14, 2016; accepted March 18, 2016. Date of publication March 31, 2016; date of current version August 12, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yue Rong. Work supported by NSF CAREER CCF-0952867, Office of Naval Research N00014-12-1-0997, NSF China 61573331, and NSF Anhui 1608085QF130. This paper expands results and presents proofs that were preliminarily reported in Proceedings of the Global Conference on Signal and Information Processing (GlobalSIP) 2015 [1].

A. Mokhtari and A. Ribeiro are with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: aryanm@seas.upenn.ed; aribeiro@seas.upenn.edu).

W. Shi and Q. Ling are with the Department of Automation, University of Science and Technology of China, Anhui, 230026, China (e-mail: hiwei00@mail.ustc.edu.cn; qingling@mail.ustc.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2016.2548989

further. The decentralized quadratically approximated ADMM (DQM) algorithm that we propose here minimizes a quadratic approximation of the Lagrangian minimization of each ADMM step. This quadratic approximation requires computation of local Hessians but results in an algorithm with convergence properties that are: (i) better than the convergence properties of DLM; (ii) asymptotically identical to the convergence behavior of DADMM. The technical contribution of this paper is to prove that (i) and (ii) are true from both analytical and practical perspectives.

We begin the paper by discussing solution of (1) with DADMM and its linearized version DLM (Section II). Both of these algorithms perform updates on dual and primal auxiliary variables that are identical and computationally simple. They differ in the manner in which principal primary variables are updated. DADMM solves a convex optimization problem and DLM solves a regularized linear approximation. We follow with an explanation of DQM that differs from DADMM and DLM in that it minimizes a quadratic approximation of the convex problem that DADMM solves exactly and DLM approximates linearly (Section III). We also explain how DQM can be implemented in a distributed manner (Proposition 1 and Algorithm 1). Convergence properties of DQM are then analyzed (Section IV) where linear convergence is established (Theorem 1 and Corollary 1). Key in the analysis is the error incurred when approximating the exact minimization of DADMM with the quadratic approximation of DQM. This error is shown to decrease as iterations progress (Proposition 2) faster than the rate that the error of DLM approaches zero (Proposition 3). This results in DQM having a guaranteed convergence factor strictly smaller than the DLM factor that approaches the guaranteed factor of DADMM for large iteration index (Section IV.A). We corroborate analytical results with numerical evaluations in a logistic regression problem (Section V). We show that DQM does outperform DLM and show that convergence paths of DQM and DADMM are almost identical (Section V.A). Overall computational cost of DQM is shown to be smaller, as expected.

Notation: Vectors are written as $\mathbf{x}\in\mathbb{R}^n$ and matrices as $\mathbf{A}\in\mathbb{R}^{n\times n}$. Given n vectors $\mathbf{x}_i$, the vector $\mathbf{x}=[\mathbf{x}_1;\ldots;\mathbf{x}_n]$ represents a stacking of the elements of each individual $\mathbf{x}_i$. We use $\|\mathbf{x}\|$ to denote the Euclidean norm of vector $\mathbf{x}$ and $\|\mathbf{A}\|$ to denote the Euclidean norm of matrix $\mathbf{A}$. The notation $\mathbf{A}\preceq\mathbf{B}$ implies that the matrix $\mathbf{B}-\mathbf{A}$ is positive semidefinite. The gradient of a function f at point $\mathbf{x}$ is denoted as $\nabla f(\mathbf{x})$ and the Hessian is denoted as $\nabla^2 f(\mathbf{x})$. We use $\sigma(\mathbf{B})$ to denote the singular values of matrix $\mathbf{B}$ and $\boldsymbol{\lambda}(\mathbf{A})$ to denote the eigenvalues of matrix $\mathbf{A}$.

II. Distributed Alternating Directions Method of Multipliers

Consider a connected network with n nodes and m edges where the set of nodes is $\mathcal{V}=\{1,\ldots,n\}$ and the set of ordered edges $\mathcal{E}$ contains pairs (i,j) indicating that i can communicate to j. We restrict attention to symmetric networks in which $(i,j)\in\mathcal{E}$ if and only if $(j,i)\in\mathcal{E}$ and define node i's neighborhood as the set $\mathcal{N}_i=\{j\mid(i,j)\in\mathcal{E}\}$. We further assume that there is no self loop in the network, i.e., $\nexists\,(i,j)\in\mathcal{E}$ such that $i=j$. In problem (1) agent i has access to the local objective function $f_i(\tilde{\mathbf{x}})$ and agents cooperate to minimize the global cost $\sum_{i=1}^n f_i(\tilde{\mathbf{x}})$. This specification is more naturally formulated by defining variables $\mathbf{x}_i$ representing the local copies of the variable $\tilde{\mathbf{x}}$. We also define the auxiliary variables $\mathbf{z}_{ij}$ associated with edge $(i,j)\in\mathcal{E}$ and rewrite (1) as

$$\{\mathbf{x}_i^*\}_{i=1}^n:=\operatorname*{argmin}_{\mathbf{x}}\sum_{i=1}^n f_i(\mathbf{x}_i),$$
$$\text{s.t. }\mathbf{x}_i=\mathbf{z}_{ij},\ \mathbf{x}_j=\mathbf{z}_{ij},\ \text{for all }(i,j)\in\mathcal{E}. \tag{2}$$

The constraints $\mathbf{x}_i=\mathbf{z}_{ij}$ and $\mathbf{x}_j=\mathbf{z}_{ij}$ enforce that the variable $\mathbf{x}_i$ of each node i is equal to the variables $\mathbf{x}_j$ of its neighbors $j\in\mathcal{N}_i$. This condition in association with network connectivity implies that a set of variables $\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ is feasible for problem (2) if and only if all the variables $\mathbf{x}_i$ are equal to each other, i.e., if $\mathbf{x}_1=\cdots=\mathbf{x}_n$. Therefore, problems (1) and (2) are equivalent in the sense that for all i and j the optimal arguments of (2) satisfy $\mathbf{x}_i^*=\tilde{\mathbf{x}}^*$ and $\mathbf{z}_{ij}=\tilde{\mathbf{x}}^*$, where $\tilde{\mathbf{x}}^*$ is the optimal argument of (1).

Assign an arbitrary order to the edges e_{ij} in the network from 1 to m. This way, the index of the edge e_{ij} which starts from node i and ends at node j is an integer e from the set $\{1,\ldots,m\}$. To write problem (2) in a matrix form, define $\mathbf{A}_s\in\mathbb{R}^{mp\times np}$ as the block source matrix which contains $m\times n$ square blocks $(\mathbf{A}_s)_{e,i}\in\mathbb{R}^{p\times p}$. The block $(\mathbf{A}_s)_{e,i}$ is not identically null if and only if the ordering index of the edge $e_{ij}\in\mathcal{E}$ is e in which case $(\mathbf{A}_s)_{e,i}=\mathbf{I}_p$. Likewise, the block destination matrix $\mathbf{A}_d\in\mathbb{R}^{mp\times np}$ contains $m\times n$ square blocks $(\mathbf{A}_d)_{e,i}\in\mathbb{R}^{p\times p}$. The square block $(\mathbf{A}_d)_{e,i}=\mathbf{I}_p$ when e is the index associate to the edge $e_{ji}\in\mathcal{E}$ and is null otherwise. Further define $\mathbf{x}:=[\mathbf{x}_1;\ldots;\mathbf{x}_n]\in\mathbb{R}^{np}$ as a vector concatenating all local variables $\mathbf{x}_i$, the vector $\mathbf{z}:=[\mathbf{z}_1;\ldots;\mathbf{z}_m]\in\mathbb{R}^{mp}$ concatenating all auxiliary variables $\mathbf{z}_e=\mathbf{z}_{ij}$ based on the defined ordering, and the aggregate function $f:\mathbb{R}^{np}\to\mathbb{R}$ as $f(\mathbf{x}):=\sum_{i=1}^n f_i(\mathbf{x}_i)$. We can then rewrite (2) as

$$\mathbf{x}^*:=\operatorname*{argmin}_{\mathbf{x}} f(\mathbf{x}),\text{s.t. }\mathbf{A}_s\mathbf{x}-\mathbf{z}=\mathbf{0},\ \mathbf{A}_d\mathbf{x}-\mathbf{z}=\mathbf{0}. \tag{3}$$

Define now the matrix $\mathbf{A}=[\mathbf{A}_s;\mathbf{A}_d]\in\mathbb{R}^{2mp\times np}$ which stacks the source and destination matrices, and the matrix $\mathbf{B}=[-\mathbf{I}_{mp};-\mathbf{I}_{mp}]\in\mathbb{R}^{2mp\times mp}$ which stacks two negative identity matrices of size mp to rewrite (3) as

$$\mathbf{x}^*:=\operatorname*{argmin}_{\mathbf{x}} f(\mathbf{x}),\text{s.t. }\mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{z}=\mathbf{0}. \tag{4}$$

DADMM is the application of ADMM to solve (4). To develop this algorithm introduce Lagrange multipliers $\boldsymbol{\alpha}_e=\boldsymbol{\alpha}_{ij}$ and $\boldsymbol{\beta}_e=\boldsymbol{\beta}_{ij}$ associated with the constraints $\mathbf{x}_i=\mathbf{z}_{ij}$ and $\mathbf{x}_j=\mathbf{z}_{ij}$ in (2), respectively. Define $\boldsymbol{\alpha}:=[\boldsymbol{\alpha}_1;\ldots;\boldsymbol{\alpha}_m]$ as the concatenation of the multipliers $\boldsymbol{\alpha}_e$ which yields the multiplier of the constraint $\mathbf{A}_s\mathbf{x}-\mathbf{z}=\mathbf{0}$ in (3). Likewise, the corresponding Lagrange multiplier of the constraint $\mathbf{A}_d\mathbf{x}-\mathbf{z}=\mathbf{0}$ in (3) can be obtained by stacking the multipliers $\boldsymbol{\beta}_e$ to define $\boldsymbol{\beta}:=[\boldsymbol{\beta}_1;\ldots;\boldsymbol{\beta}_m]$. Grouping $\boldsymbol{\alpha}$ and β into $\boldsymbol{\lambda}:=[\boldsymbol{\alpha};\boldsymbol{\beta}]\in\mathbb{R}^{2mp}$ leads to the Lagrange multiplier $\boldsymbol{\lambda}$ associated with the constraint $\mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{z}=\mathbf{0}$ in (4). Using these definitions and introducing a positive constant $c>0$ we write the augmented Lagrangian of (4) as

$$\mathcal{L}(\mathbf{x},\mathbf{z},\boldsymbol{\lambda}):=f(\mathbf{x})+\boldsymbol{\lambda}^T(\mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{z})+\frac{c}{2}\|\mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{z}\|^2. \tag{5}$$

The idea of ADMM is to minimize the Lagrangian $\mathcal{L}(\mathbf{x},\mathbf{z},\boldsymbol{\lambda})$ with respect to $\mathbf{x}$, follow by minimizing the updated Lagrangian with respect to $\mathbf{z}$, and finish each iteration with an update of the multiplier $\boldsymbol{\lambda}$ using dual ascent. To be more precise, consider the time index $k \in \mathbb{N}$ and define $\mathbf{x}_k, \mathbf{z}_k$, and $\boldsymbol{\lambda}_k$ as the iterates at step k. At this step, the augmented Lagrangian is minimized with respect to $\mathbf{x}$ to obtain the iterate

$$\mathbf{x}_{k+1} = \underset{\mathbf{x}}{\operatorname{argmin}}\, f(\mathbf{x}) + \boldsymbol{\lambda}_k^T(\mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{z}_k) + \frac{c}{2}\| \mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{z}_k \|^2. \tag{6}$$

Then, the augmented Lagrangian is minimized with respect to the auxiliary variable $\mathbf{z}$ using the updated variable $\mathbf{x}_{k+1}$ to obtain

$$\mathbf{z}_{k+1} = \underset{\mathbf{z}}{\operatorname{argmin}}\, f(\mathbf{x}_{k+1}) + \boldsymbol{\lambda}_k^T(\mathbf{A}\mathbf{x}_{k+1}+\mathbf{B}\mathbf{z}) + \frac{c}{2}\| \mathbf{A}\mathbf{x}_{k+1}+\mathbf{B}\mathbf{z} \|^2. \tag{7}$$

After updating the variables $\mathbf{x}$ and $\mathbf{z}$, the Lagrange multiplier $\boldsymbol{\lambda}_k$ is updated through the dual ascent iteration

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + c(\mathbf{A}\mathbf{x}_{k+1}+\mathbf{B}\mathbf{z}_{k+1}). \tag{8}$$

The DADMM algorithm is obtained by observing that the structure of the matrices $\mathbf{A}$ and $\mathbf{B}$ is such that (6)–(8) can be implemented in a distributed manner [8], [20], [21].

The updates for the auxiliary variable $\mathbf{z}$ and the Lagrange multiplier $\boldsymbol{\lambda}$ are not costly in terms of computation time. However, updating the primal variable $\mathbf{x}$ can be expensive as it entails the solution of an optimization problem (cf. (6)). The DLM algorithm avoids this cost with an inexact update of the primal variable iterate $\mathbf{x}_{k+1}$. This inexact update relies on approximating the aggregate function value $f(\mathbf{x}_{k+1})$ in (6) through a regularized linearization of the aggregate function f in a neighborhood of the current variable $\mathbf{x}_k$. This regularized approximation takes the form $f(\mathbf{x}) \approx f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T(\mathbf{x}-\mathbf{x}_k) + (\rho/2)\|\mathbf{x}-\mathbf{x}_k\|^2$ for a given positive constant $\rho > 0$. Consequently, the update formula for the primal variable $\mathbf{x}$ in DLM replaces the DADMM exact minimization in (6) by the minimization of the quadratic form

$$\mathbf{x}_{k+1} = \underset{\mathbf{x}}{\operatorname{argmin}}\, f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T(\mathbf{x}-\mathbf{x}_k) + \frac{\rho}{2}\|\mathbf{x}-\mathbf{x}_k\|^2 + \boldsymbol{\lambda}_k^T(\mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{z}_k) + \frac{c}{2}\|\mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{z}_k\|^2. \tag{9}$$

The first order optimality condition for (9) implies that the updated variable $\mathbf{x}_{k+1}$ satisfies

$$\nabla f(\mathbf{x}_k) + \rho(\mathbf{x}_{k+1}-\mathbf{x}_k) + \mathbf{A}^T\boldsymbol{\lambda}_k + c\mathbf{A}^T(\mathbf{A}\mathbf{x}_{k+1}+\mathbf{B}\mathbf{z}_k) = \mathbf{0}. \tag{10}$$

According to (10), the updated variable $\mathbf{x}_{k+1}$ can be computed by inverting the positive definite matrix $\rho\mathbf{I} + c\mathbf{A}^T\mathbf{A}$. This update can also be implemented in a distributed manner.

The sequence of variables $\mathbf{x}_k$ generated by DLM converges linearly to the optimal argument $\mathbf{x}^*$ [29]. Although this is the same rate of DADMM, linear convergence factor of DLM is larger than the one for DADMM (see Section IV.A), and can be much smaller depending on the condition number of the local functions f_i (see Section V.A). To close the gap between these factors we can use a second order approximation of (6). This is the idea of DQM that we introduce in the following section.

III. DQM: Decentralized Quadratically Approximated ADMM

DQM uses a local quadratic approximation of the primal function $f(\mathbf{x})$ around the current iterate $\mathbf{x}_k$. If we let $\mathbf{H}_k := \nabla^2 f(\mathbf{x}_k)$ denote the primal function Hessian evaluated at $\mathbf{x}_k$ the quadratic approximation of f at $\mathbf{x}_k$ is $f(\mathbf{x}) \approx f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T(\mathbf{x}-\mathbf{x}_k) + (1/2)(\mathbf{x}-\mathbf{x}_k)^T\mathbf{H}_k(\mathbf{x}-\mathbf{x}_k)$. Using this approximation in (6) yields the DQM update that we therefore define as

$$\mathbf{x}_{k+1} := \underset{\mathbf{x}}{\operatorname{argmin}}\, f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T(\mathbf{x}-\mathbf{x}_k) + \frac{1}{2}(\mathbf{x}-\mathbf{x}_k)^T\mathbf{H}_k(\mathbf{x}-\mathbf{x}_k) + \boldsymbol{\lambda}_k^T(\mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{z}_k) + \frac{c}{2}\|\mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{z}_k\|^2. \tag{11}$$

Comparison of (9) and (11) shows that in DLM the quadratic term $(\rho/2)\|\mathbf{x}_{k+1}-\mathbf{x}_k\|^2$ is added to the first-order approximation of the primal objective function, while in DQM the second order approximation of the primal objective function is used to reach a more accurate approximation for $f(\mathbf{x})$. Since (11) is a quadratic program, the first order optimality condition yields a system of linear equations that can be solved to find $\mathbf{x}_{k+1}$,

$$\nabla f(\mathbf{x}_k) + \mathbf{H}_k(\mathbf{x}_{k+1}-\mathbf{x}_k) + \mathbf{A}^T\boldsymbol{\lambda}_k + c\mathbf{A}^T(\mathbf{A}\mathbf{x}_{k+1}+\mathbf{B}\mathbf{z}_k) = \mathbf{0}. \tag{12}$$

This update can be solved by inverting the matrix $\mathbf{H}_k + c\mathbf{A}^T\mathbf{A}$ which is invertible if, as we are assuming, $f(\mathbf{x})$ is strongly convex.

The DADMM updates in (7) and (8) are used verbatim in DQM, which is therefore defined by recursive application of (12), (7), and (8). It is customary to consider the first order optimality conditions of (7) and to reorder terms in (8) to rewrite the respective updates as

$$\begin{aligned} \mathbf{B}^T\boldsymbol{\lambda}_k + c\mathbf{B}^T(\mathbf{A}\mathbf{x}_{k+1}+\mathbf{B}\mathbf{z}_{k+1}) &= \mathbf{0}, \\ \boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k - c(\mathbf{A}\mathbf{x}_{k+1}+\mathbf{B}\mathbf{z}_{k+1}) &= \mathbf{0}. \end{aligned} \tag{13}$$

DQM is then equivalently defined by recursive solution of the system of linear equations in (12) and (13). This system, as is the case of DADMM and DLM, can be reworked into a simpler form that reduces communication cost. To derive this simpler form we assume a specific structure for the initial vectors $\boldsymbol{\lambda}_0 = [\boldsymbol{\alpha}_0; \boldsymbol{\beta}_0]$, $\mathbf{x}_0$, and $\mathbf{z}_0$ as introduced in the following assumption.

Assumption 1: Define the oriented incidence matrix as $\mathbf{E}_o := \mathbf{A}_s - \mathbf{A}_d$ and the unoriented incidence matrix as $\mathbf{E}_u := \mathbf{A}_s + \mathbf{A}_d$. The initial Lagrange multipliers $\boldsymbol{\alpha}_0$ and $\boldsymbol{\beta}_0$, and the initial variables $\mathbf{x}_0$ and $\mathbf{z}_0$ are chosen such that:

a) The multipliers are opposites of each other, $\boldsymbol{\alpha}_0 = -\boldsymbol{\beta}_0$.
b) The initial primal variables satisfy $\mathbf{E}_u\mathbf{x}_0 = 2\mathbf{z}_0$.
c) The initial multiplier $\boldsymbol{\alpha}_0$ lies in the column space of $\mathbf{E}_o$.

First notice that we call $\mathbf{E}_o := \mathbf{A}_s - \mathbf{A}_d$ as the oriented incidence matrix, since it provides information about the direction of the edges in the network, while the unoriented incidence matrix $\mathbf{E}_u := \mathbf{A}_s + \mathbf{A}_d$ only indicates the two end points of each edge without any information about the direction of the edge.

Assumption 1 is minimally restrictive. The only non-elementary condition is (c) but that can be satisfied by $\boldsymbol{\alpha}_0 = \mathbf{0}$.

Nulling all other variables, i.e., making $\boldsymbol{\beta}_0 = \mathbf{0}, \mathbf{x}_0 = \mathbf{0}$, and $\mathbf{z}_0 = \mathbf{0}$ is a trivial choice to comply with conditions (a) and (b) as well. An important consequence of the initialization choice in (1) is that if the conditions in Assumption 1 are true at time $k = 0$ they stay true for all subsequent iterations $k > 0$ as we state next.

Lemma 1: Consider the DQM algorithm as defined by (12)–(13). If Assumption 1 holds, then for all $k \geq 0$ the Lagrange multipliers $\boldsymbol{\alpha}_k$ and $\boldsymbol{\beta}_k$, and the variables $\mathbf{x}_k$ and $\mathbf{z}_k$ satisfy:

a) The multipliers are opposites of each other, $\boldsymbol{\alpha}_k = -\boldsymbol{\beta}_k$.
b) The primal variables satisfy $\mathbf{E}_u \mathbf{x}_k = 2\mathbf{z}_k$.
c) The multiplier $\boldsymbol{\alpha}_k$ lies in the column space of $\mathbf{E}_o$.

Proof: See Appendix A. ■

The validity of (c) in Lemma 1 is important for the convergence analysis of Section IV. The validity of (a) and (b) means that maintaining multipliers $\boldsymbol{\alpha}_k$ and $\boldsymbol{\beta}_k$ is redundant because they are opposites and that maintaining variables $\mathbf{z}_k$ is also redundant because they can be computed as $\mathbf{z}_k = \mathbf{E}_u \mathbf{x}_k/2$. It is then possible to replace (12)–(13) by a simpler system of linear equations as we explain in the following proposition.

Proposition 1: Consider the DQM algorithm as defined by (12)–(13) and define the sequence $\phi_k := \mathbf{E}_o^T \boldsymbol{\alpha}_k$. Further define the unoriented Laplacian as $\mathbf{L}_u := (1/2)\, \mathbf{E}_u^T \mathbf{E}_u$, the oriented Laplacian as $\mathbf{L}_o = (1/2)\, \mathbf{E}_o^T \mathbf{E}_o$, and the degree matrix as $\mathbf{D} := (\mathbf{L}_u + \mathbf{L}_o)/2$. If Assumption 1 holds true, the DQM iterates $\mathbf{x}_k$ can be generated as

$$\begin{aligned} \mathbf{x}_{k+1} &= (2c\mathbf{D} + \mathbf{H}_k)^{-1} \left[(c\mathbf{L}_u + \mathbf{H}_k)\, \mathbf{x}_k - \nabla f(\mathbf{x}_k) - \phi_k \right], \\ \phi_{k+1} &= \phi_k + c\mathbf{L}_o \mathbf{x}_{k+1}. \end{aligned} \tag{14}$$

Proof: See Appendix B. ■

Proposition 1 states that by introducing the sequence of variables ϕ_k, the DQM primal iterates $\mathbf{x}_k$ can be computed through the recursive expressions in (14). These recursions are simpler than (12)–(13) because they eliminate the auxiliary variables $\mathbf{z}_k$ and reduce the dimensionality of $\boldsymbol{\lambda}_k$—twice the number of edges—to that of ϕ_k—the number of nodes. Further observe that if (14) is used for implementation we do not have to make sure that the conditions of Assumption 1 are satisfied. We just need to pick $\phi_0 := \mathbf{E}_o^T \boldsymbol{\alpha}_0$ for some $\boldsymbol{\alpha}_0$ in the column space of $\mathbf{E}_0$—which is not difficult, we can use, e.g., $\phi_0 = \mathbf{0}$. The role of Assumption 1 is to state conditions for which the expressions in (12)–(13) are an equivalent representation of (14) that we use for convergence analyses.

The structure of the primal objective function Hessian $\mathbf{H}_k$, the degree matrix $\mathbf{D}$, and the oriented and unoriented Laplacians $\mathbf{L}_o$ and $\mathbf{L}_u$ make distributed implementation of (14) possible. Indeed, the matrix $2c\mathbf{D} + \mathbf{H}_k$ is block diagonal and its i-th diagonal block is given by $2cd_i\mathbf{I} + \nabla^2 f_i(\mathbf{x}_i)$ which is locally available for node i. Likewise, the inverse matrix $(2c\mathbf{D} + \mathbf{H}_k)^{-1}$ is block diagonal and locally computable since the i-th diagonal block is $(2cd_i\mathbf{I} + \nabla^2 f_i(\mathbf{x}_i))^{-1}$. Computations of the products $\mathbf{L}_u \mathbf{x}_k$ and $\mathbf{L}_o \mathbf{x}_{k+1}$ can be implemented in a decentralized manner as well, since the Laplacian matrices $\mathbf{L}_u$ and $\mathbf{L}_o$ are block neighbor sparse in the sense that the (i, j)-th block is not null if and only if nodes i and j are neighbors or $j = i$. Therefore, nodes can compute their local parts for the products $\mathbf{L}_u \mathbf{x}_k$ and $\mathbf{L}_o \mathbf{x}_{k+1}$ by exchanging information with their neighbors. By defining components of the vector ϕ_k as $\phi_k := [\phi_{1,k}, \ldots, \phi_{n,k}]$, the update formula in (14) for the individual agents can then be written block-wise as

$$\begin{aligned} \mathbf{x}_{i,k+1} = \left(2cd_i\mathbf{I} + \nabla^2 f_i(\mathbf{x}_{i,k})\right)^{-1} \Bigg[& cd_i\mathbf{x}_{i,k} + c\sum_{j\in\mathcal{N}_i} \mathbf{x}_{j,k} \\ & + \nabla^2 f_i(\mathbf{x}_{i,k})\, \mathbf{x}_{i,k} - \nabla f_i(\mathbf{x}_{i,k}) - \phi_{i,k} \Bigg], \end{aligned} \tag{15}$$

where $\mathbf{x}_{i,k}$ corresponds to the iterate of node i at step k. Notice that the definition

$$\mathbf{L}_u := (1/2)\mathbf{E}_u^T \mathbf{E}_u = (1/2)(\mathbf{A}_s + \mathbf{A}_d)^T (\mathbf{A}_s + \mathbf{A}_d) \tag{16}$$

is used to simplify the i-th component of $c\mathbf{L}_u \mathbf{x}_k$ as $c\sum_{j\in\mathcal{N}_i}(\mathbf{x}_{i,k} + \mathbf{x}_{j,k})$ which is equivalent to $cd_i\mathbf{x}_{i,k} + c\sum_{j\in\mathcal{N}_i}\mathbf{x}_{j,k}$. Further, using the definition

$$\mathbf{L}_o = (1/2)\mathbf{E}_o^T \mathbf{E}_o = (1/2)(\mathbf{A}_s - \mathbf{A}_d)^T (\mathbf{A}_s - \mathbf{A}_d), \tag{17}$$

the i-th component of the product $c\mathbf{L}_o \mathbf{x}_{k+1}$ in (16) can be simplified as $c\sum_{j\in\mathcal{N}_i}(\mathbf{x}_{i,k} - \mathbf{x}_{j,k})$. Therefore, the second update formula in (14) can be locally implemented at each node i as

$$\phi_{i,k+1} = \phi_{i,k} + c\sum_{j\in\mathcal{N}_i} (\mathbf{x}_{i,k+1} - \mathbf{x}_{j,k+1}). \tag{18}$$

Algorithm 1: DQM method at node i

Require: Initial local iterates $\mathbf{x}_{i,0}$ and ϕ_0.

1: **for** $k = 0, 1, 2, \ldots$ **do**
2: Update the local iterate $\mathbf{x}_{i,k}$ as

$$\begin{aligned} \mathbf{x}_{i,k+1} = \left(2cd_i\mathbf{I} + \nabla^2 f_i(\mathbf{x}_{i,k})\right)^{-1} \Bigg[& cd_i\mathbf{x}_{i,k} + c\sum_{j\in\mathcal{N}_i} \mathbf{x}_{j,k} \\ & + \nabla^2 f_i(\mathbf{x}_{i,k})\, \mathbf{x}_{i,k} - \nabla f_i(\mathbf{x}_{i,k}) - \phi_{i,k} \Bigg]. \end{aligned}$$

3: Exchange iterates $\mathbf{x}_{i,k+1}$ with neighbors $j \in \mathcal{N}_i$.
4: Update local dual variable $\phi_{i,k}$ as

$$\phi_{i,k+1} = \phi_{i,k} + c\sum_{j\in\mathcal{N}_i} (\mathbf{x}_{i,k+1} - \mathbf{x}_{j,k+1}).$$

5: **end for**

The proposed DQM method is summarized in Algorithm 1. The initial value for the local iterate $\mathbf{x}_{i,0}$ can be any arbitrary vector in $\mathbb{R}^p$. The initial vector $\phi_{i,0}$ should be in column space of $\mathbf{E}_o^T$. To guarantee satisfaction of this condition, the initial vector is set as $\phi_{i,0} = \mathbf{0}$. At each iteration k, updates of the primal and dual variables in (15) and (16) are computed in Steps 2 and 4, respectively. Nodes exchange their local variables $\mathbf{x}_{i,k}$ with their neighbors $j \in \mathcal{N}_i$ in Step 3, since this information is required for the updates in Steps 2 and 4.

DADMM, DQM, and DLM occupy different points in a trade-off curve of computational cost per iteration and number of iterations needed to achieve convergence. The computational cost of each DADMM iteration is large in general because it requires solution of the optimization problem in (6). The cost of DLM iterations is minimal because the solution of (10) can be reduced

to the inversion of a block diagonal matrix; see [30]. The cost of DQM iterations is larger than the cost of DLM iterations because they require evaluation of local Hessians as well as inversion of the matrices $2cd_i\mathbf{I}+\nabla^2 f_i(\mathbf{x}_{i,k})$ to implement (15). But the cost is smaller than the cost of DADMM iterations except in cases in which solving (6) is easy. In terms of the number of iterations required until convergence, DADMM requires the least and DLM the most. The foremost technical conclusions of the convergence analysis presented in the following section are: (i) convergence of DQM is strictly faster than convergence of DLM; (ii) asymptotically in the number of iterations, the per iteration improvements of DADMM and DQM are identical. It follows from these observations that DQM achieves target optimality in a number of iterations similar to DADMM but with iterations that are computationally cheaper.

IV. Convergence Analysis

In this section we show that the sequence of iterates $\mathbf{x}_k$ generated by DQM converges linearly to the optimal argument $\mathbf{x}^*=[\tilde{\mathbf{x}}^*;\ldots;\tilde{\mathbf{x}}^*]$. As a byproduct of this analysis we also obtain a comparison between the linear convergence factors of DLM, DQM, and DADMM. To derive these results we make the following assumptions.

Assumption 2: The network is such that any singular value of the unoriented incidence matrix $\mathbf{E}_u$, defined as $\sigma(\mathbf{E}_u)$, satisfies $0<\gamma_u\leq\sigma(\mathbf{E}_u)\leq\Gamma_u$ where γ_u and Γ_u are constants; the smallest non-zero singular value of the oriented incidence matrix $\mathbf{E}_o$ is $\gamma_o>0$.

Assumption 3: The local objective functions $f_i(\mathbf{x})$ are twice differentiable and the eigenvalues of their local Hessians $\nabla^2 f_i(\mathbf{x})$ are bounded within positive constants m and M where $0<m\leq M<\infty$ so that for all $\mathbf{x}\in\mathbb{R}^p$ it holds

$$m\mathbf{I}\preceq\nabla^2 f_i(\mathbf{x})\preceq M\mathbf{I}. \tag{19}$$

Assumption 4: The local Hessians $\nabla^2 f_i(\mathbf{x})$ are Lipschitz continuous with constant L so that for all $\mathbf{x},\hat{\mathbf{x}}\in\mathbb{R}^p$ it holds

$$\|\nabla^2 f_i(\mathbf{x})-\nabla^2 f_i(\hat{\mathbf{x}})\|\leq L\ \|\mathbf{x}-\hat{\mathbf{x}}\|. \tag{20}$$

The eigenvalue bounds in Assumption 2 are measures of network connectivity. Note that the assumption that all the singular values of the unoriented incidence matrix $\mathbf{E}_u$ are positive implies that the graph is non-bipartite. Thus, for every connected non-bipartite graphs the conditions in Assumption 2 are satisfied. The conditions imposed by assumptions 3 and 4 are typical in the analysis of second order methods; see, e.g., ([31], Chapter 9). The lower bound for the eigenvalues of the local Hessians $\nabla^2 f_i(\mathbf{x})$ implies strong convexity of the local objective functions $f_i(\mathbf{x})$ with constant m, while the upper bound M for the eigenvalues of the local Hessians $\nabla^2 f_i(\mathbf{x})$ is tantamount to Lipschitz continuity of local gradients $\nabla f_i(\mathbf{x})$ with Lipschitz constant M. Further note that as per the definition of the aggregate objective $f(\mathbf{x}):=\sum_{i=1}^n f_i(\mathbf{x}_i)$, the Hessian $\mathbf{H}(\mathbf{x}):=\nabla^2 f(\mathbf{x})\in\mathbb{R}^{np\times np}$ is block diagonal with i-th diagonal block given by the i-th local objective function Hessian $\nabla^2 f_i(\mathbf{x}_i)$. Therefore, the bounds for the local Hessians' eigenvalues in (17) also hold for the aggregate function Hessian. Thus, we have that for any $\mathbf{x}\in\mathbb{R}^{np}$ the eigenvalues of the Hessian $\mathbf{H}(\mathbf{x})$ are uniformly bounded as

$$m\mathbf{I}\preceq\mathbf{H}(\mathbf{x})\preceq M\mathbf{I}. \tag{21}$$

Assumption 4 also implies an analogous condition for the aggregate function Hessian $\mathbf{H}(\mathbf{x})$ as we show in the following lemma.

Lemma 2: Consider the definition of the aggregate function $f(\mathbf{x}):=\sum_{i=1}^n f_i(\mathbf{x}_i)$. If Assumption 4 holds true, the aggregate function Hessian $\mathbf{H}(\mathbf{x})=:\nabla^2 f(\mathbf{x})$ is Lipschitz continuous with constant L. I.e., for all $\mathbf{x},\hat{\mathbf{x}}\in\mathbb{R}^{np}$ we can write

$$\|\mathbf{H}(\mathbf{x})-\mathbf{H}(\hat{\mathbf{x}})\|\leq L\ \|\mathbf{x}-\hat{\mathbf{x}}\|. \tag{22}$$

Proof: See Appendix C. ■

DQM can be interpreted as an attempt to approximate the primal update of DADMM. Therefore, we evaluate the performance of DQM by studying a measure of the error of the approximation in the DQM update relative to the DADMM update. In the primal update of DQM, the gradient $\nabla f(\mathbf{x}_{k+1})$ is estimated by the approximation $\nabla f(\mathbf{x}_k)+\mathbf{H}_k(\mathbf{x}_{k+1}-\mathbf{x}_k)$. Therefore, we can define the DQM error vector $\mathbf{e}_k^{\mathrm{DQM}}$ as

$$\mathbf{e}_k^{\mathrm{DQM}}:=\nabla f(\mathbf{x}_k)+\mathbf{H}_k(\mathbf{x}_{k+1}-\mathbf{x}_k)-\nabla f(\mathbf{x}_{k+1}). \tag{23}$$

Based on the definition in (21), the approximation error of DQM vanishes when the difference of two consecutive iterates $\mathbf{x}_{k+1}-\mathbf{x}_k$ approaches zero. This observation is formalized in the following proposition by introducing an upper bound for the error vector norm $\|\mathbf{e}_k^{\mathrm{DQM}}\|$ in terms of the difference norm $\|\mathbf{x}_{k+1}-\mathbf{x}_k\|$.

Proposition 2: Consider the DQM method as introduced in (12)–(13) and the error $\mathbf{e}_k^{\mathrm{DQM}}$ defined in (21). If Assumptions 1–4 hold true, the DQM error norm $\|\mathbf{e}_k^{\mathrm{DQM}}\|$ is bounded above by

$$\left\|\mathbf{e}_k^{\mathrm{DQM}}\right\|\leq\min\left\{2M\ \|\mathbf{x}_{k+1}-\mathbf{x}_k\|,\frac{L}{2}\|\mathbf{x}_{k+1}-\mathbf{x}_k\|^2\right\}. \tag{24}$$

Proof: See Appendix D. ■

Proposition 2 asserts that the error norm $\|\mathbf{e}_k^{\mathrm{DQM}}\|$ is bounded above by the minimum of a linear and a quadratic term of the iterate difference norm $\|\mathbf{x}_{k+1}-\mathbf{x}_k\|$. Hence, the approximation error vanishes as the sequence of iterates $\mathbf{x}_k$ converges. We will show in Theorem 1 that the sequence $\|\mathbf{x}_{k+1}-\mathbf{x}_k\|$ converges to zero which implies that the error vector $\mathbf{e}_k^{\mathrm{DQM}}$ converges to the null vector $\mathbf{0}$. Notice that after a number of iterations the term $(L/2)\|\mathbf{x}_{k+1}-\mathbf{x}_k\|$ becomes smaller than $2M$, which implies that the upper bound in (22) can be simplified as $(L/2)\|\mathbf{x}_{k+1}-\mathbf{x}_k\|^2$ for sufficiently large k. This is important because it implies that the error vector norm $\|\mathbf{e}_k^{\mathrm{DQM}}\|$ eventually becomes proportional to the quadratic term $\|\mathbf{x}_{k+1}-\mathbf{x}_k\|^2$ and, as a consequence, it vanishes faster than the term $\|\mathbf{x}_{k+1}-\mathbf{x}_k\|$.

Utilize now the definition in (21) to rewrite the primal variable DQM update in (12) as

$$\nabla f(\mathbf{x}_{k+1})+\mathbf{e}_k^{\mathrm{DQM}}+\mathbf{A}^T\boldsymbol{\lambda}_k+c\mathbf{A}^T(\mathbf{A}\mathbf{x}_{k+1}+\mathbf{B}\mathbf{z}_k)=\mathbf{0}. \tag{25}$$

Comparison of (23) with the optimality condition for the DADMM update in (6) shows that they coincide except for the gradient approximation error term $\mathbf{e}_k^{\mathrm{DQM}}$. The DQM and DADMM updates for the auxiliary variables $\mathbf{z}_k$ and the dual variables $\boldsymbol{\lambda}_k$ are identical (cf. (7), (8), and (13)), as already observed.

Further let the pair $(\mathbf{x}^*, \mathbf{z}^*)$ stand for the unique solution of (2) with uniqueness implied by the strong convexity assumption and define $\boldsymbol{\alpha}^*$ as the unique optimal multiplier that lies in the column space of $\mathbf{E}_o$—see Lemma 1 of [29] for a proof that such optimal dual variable exists and is unique. To study convergence properties of DQM we modify the system of DQM equations defined by (13) and (23), which is equivalent to the system (12)–(13), to include terms that involve differences between current iterates and the optimal arguments $\mathbf{x}^*, \mathbf{z}^*$, and $\boldsymbol{\alpha}^*$. We state this reformulation in the following lemma.

Lemma 3: Consider the DQM method as defined by (12)–(13) and its equivalent formulation in (13) and (23). If Assumption 1 holds true, then the optimal arguments $\mathbf{x}^*, \mathbf{z}^*$, and $\boldsymbol{\alpha}^*$ satisfy

$$\nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}^*) + \mathbf{e}_k^{\mathrm{DQM}} + \mathbf{E}_o^T(\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^*) - c\mathbf{E}_u^T(\mathbf{z}_k - \mathbf{z}_{k+1}) = \mathbf{0}, \tag{26}$$

$$2(\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}_k) - c\mathbf{E}_o(\mathbf{x}_{k+1} - \mathbf{x}^*) = \mathbf{0}, \tag{27}$$

$$\mathbf{E}_u(\mathbf{x}_k - \mathbf{x}^*) - 2(\mathbf{z}_k - \mathbf{z}^*) = \mathbf{0}. \tag{28}$$

Proof: See Appendix E. ∎

With the preliminary results in Lemmata 2 and 3 and Proposition 2 we can state our convergence results. To do so, define the energy function $V: \mathbb{R}^{mp\times mp} \to \mathbb{R}$ as

$$V(\mathbf{z}, \boldsymbol{\alpha}) := c \parallel \mathbf{z} - \mathbf{z}^* \parallel^2 + \frac{1}{c} \parallel \boldsymbol{\alpha} - \boldsymbol{\alpha}^* \parallel^2. \tag{29}$$

The energy function $V(\mathbf{z}, \boldsymbol{\alpha})$ captures the distances of the variables $\mathbf{z}_k$ and $\boldsymbol{\alpha}_k$ to the respective optimal arguments $\mathbf{z}^*$ and $\boldsymbol{\alpha}^*$. To simplify notation we further define the variable $\mathbf{u} \in \mathbb{R}^{2mp}$ and matrix $\mathbf{C} \in \mathbb{R}^{2mp\times 2mp}$ as

$$\mathbf{u} := \begin{bmatrix} \mathbf{z} \\ \boldsymbol{\alpha} \end{bmatrix}, \ \mathbf{C} := \begin{bmatrix} c\mathbf{I}_{mp} & \mathbf{0} \\ \mathbf{0} & (1/c)\mathbf{I}_{mp} \end{bmatrix}. \tag{30}$$

Based on the definitions in (28), the energy function in (27) can be alternatively written $V(\mathbf{z}, \boldsymbol{\alpha}) = V(\mathbf{u}) = \parallel \mathbf{u} - \mathbf{u}^* \parallel_{\mathbf{C}}^2$, where $\mathbf{u}^* = [\mathbf{z}^*; \boldsymbol{\alpha}^*]$. The energy sequence $V(\mathbf{u}_k) = \parallel \mathbf{u}_k - \mathbf{u}^* \parallel_{\mathbf{C}}^2$ converges to zero at a linear rate as we state in the following theorem.

Theorem 1: Consider the DQM method as defined by (12)–(13), let the constant c be such that $c > 4M^2/(m\gamma_u^2)$, and define the sequence of non-negative variables ζ_k as

$$\zeta_k := \min\left\{\frac{L}{2} \parallel \mathbf{x}_{k+1} - \mathbf{x}_k \parallel, 2M\right\}. \tag{31}$$

Further, consider arbitrary constants μ, μ', and η with $\mu, \mu' > 1$ and $\eta_k \in (\zeta_k/m, c\gamma_u^2/\zeta_k)$. If Assumptions 1–4 hold true, then the sequence $\parallel \mathbf{u}_k - \mathbf{u}^* \parallel_{\mathbf{C}}^2$ generated by DQM satisfies

$$\parallel \mathbf{u}_{k+1} - \mathbf{u}^* \parallel_{\mathbf{C}}^2 \leq \frac{1}{1+\delta_k} \parallel \mathbf{u}_k - \mathbf{u}^* \parallel_{\mathbf{C}}^2 \tag{32}$$

where the sequence of positive scalars δ_k is given by

$$\delta_k = \min\left\{\frac{(\mu-1)(c\gamma_u^2 - \eta_k\zeta_k)\gamma_o^2}{\mu\mu'(c\Gamma_u^2\gamma_u^2 + 4\zeta_k^2/c(\mu'-1))}, \frac{m - \zeta_k/\eta_k}{c\Gamma_u^2/4 + \mu M^2/c\gamma_o^2}\right\}. \tag{33}$$

Proof: See Appendix F. ∎

Notice that δ_k is a decreasing function of ζ_k and that ζ_k is bounded above by $2M$. Therefore, if we substitute ζ_k by $2M$ in (31), the inequality in (30) is still valid. This substitution implies that the sequence $\parallel \mathbf{u}_k - \mathbf{u}^* \parallel_{\mathbf{C}}^2$ converges linearly to zero with a coefficient not larger than $1-\delta$ with $\delta = \delta_k$ following from (30) with $\zeta_k = 2M$. The more generic definition of ζ_k in (29) is important for the rate comparisons in Section IV.A. Observe that in order to guarantee that $\delta_k > 0$ for all $k \geq 0$, η_k is chosen from the interval $(\zeta_k/m, c\gamma_u^2/\zeta_k)$. This interval is non-empty since the constant c is chosen as $c > 4M^2/(m\gamma_u^2) \geq \zeta_k^2/(m\gamma_u^2)$. This condition on the parameter c is required for the convergence proof of DQM, while c can be any positive constant in DADMM [21]. However, the optimal choices of c for DQM and DADMM are equal in practice—see Section V—and this condition does not affect the asymptotic result for the linear convergence factor of DQM.

The linear convergence in Theorem 1 is for the vector $\mathbf{u}_k$ which includes the auxiliary variable $\mathbf{z}_k$ and the multipliers $\boldsymbol{\alpha}_k$. Linear convergence of the primal variables $\mathbf{x}_k$ to the optimal argument $\mathbf{x}^*$ follows as a corollary that we establish next.

Corollary 1: Under the assumptions in Theorem 1, the sequence of squared norms $\parallel \mathbf{x}_k - \mathbf{x}^* \parallel^2$ generated by the DQM algorithm converges R-linearly to zero, i.e.,

$$\parallel \mathbf{x}_k - \mathbf{x}^* \parallel^2 \leq \frac{4}{c\gamma_u^2} \parallel \mathbf{u}_k - \mathbf{u}^* \parallel_{\mathbf{C}}^2. \tag{34}$$

Proof: Notice that according to (26) we can write $\parallel \mathbf{E}_u(\mathbf{x}_k - \mathbf{x}^*) \parallel^2 = 4 \parallel \mathbf{z}_k - \mathbf{z}^* \parallel^2$. Since γ_u is the smallest singular value of $\mathbf{E}_u$, we obtain that

$$\parallel \mathbf{x}_k - \mathbf{x}^* \parallel^2 \leq (4/\gamma_u^2) \parallel \mathbf{z}_k - \mathbf{z}^* \parallel^2. \tag{35}$$

Moreover, according to the relation

$$\parallel \mathbf{u}_k - \mathbf{u}^* \parallel_{\mathbf{C}}^2 = c \parallel \mathbf{z}_k - \mathbf{z}^* \parallel^2 + (1/c) \parallel \boldsymbol{\alpha}_k - \boldsymbol{\alpha}^* \parallel^2 \tag{36}$$

we can write $c \parallel \mathbf{z}_k - \mathbf{z}^* \parallel^2 \leq \parallel \mathbf{u}_k - \mathbf{u}^* \parallel_{\mathbf{C}}^2$. Combining these two inequalities yields the claim in (32). ∎

As per Corollary 1, convergence of the sequence $\mathbf{x}_k$ to $\mathbf{x}^*$ is dominated by a linearly decreasing sequence. Notice that the sequence of squared norms $\parallel \mathbf{x}_k - \mathbf{x}^* \parallel^2$ need not be monotonically decreasing as the energy sequence $\parallel \mathbf{u}_{k+1} - \mathbf{u}^* \parallel_{\mathbf{C}}^2$ is.

A. Convergence Rates Comparison

Based on the result in Corollary 1, the sequence of iterates $\mathbf{x}_k$ generated by DQM converges. This observation implies that the sequence $\parallel \mathbf{x}_{k+1} - \mathbf{x}_k \parallel$ approaches zero. Hence, the sequence of scalars ζ_k defined in (29) converges to 0 as time passes, since ζ_k is bounded above by $(L/2) \parallel \mathbf{x}_{k+1} - \mathbf{x}_k \parallel$. Using this fact that $\lim_{k\to\infty}\zeta_k = 0$ to compute the limit of δ_k in (31) and further making $\mu' \to 1$ in the resulting limit we have that

$$\lim_{k\to\infty} \delta_k = \min\left\{\frac{(\mu-1)\gamma_o^2}{\mu\Gamma_u^2}, \frac{m}{c\Gamma_u^2/4 + \mu M^2/c\gamma_o^2}\right\}. \tag{37}$$

Notice that the limit of δ_k in (33) is identical to the factor of linear convergence for DADMM [21]. Therefore, we conclude that as time passes the factor of linear convergence for DQM approaches the one for DADMM.

To compare the convergence rates of DLM, DQM and DADMM we define the error of the gradient approximation

for DLM as

$$\mathbf{e}_k^{\mathrm{DLM}} = \nabla f(\mathbf{x}_k) + \rho(\mathbf{x}_{k+1} - \mathbf{x}_k) - \nabla f(\mathbf{x}_{k+1}), \quad (38)$$

which is the difference of exact gradient $\nabla f(\mathbf{x}_{k+1})$ and the DLM gradient approximation $\nabla f(\mathbf{x}_k) + \rho(\mathbf{x}_{k+1} - \mathbf{x}_k)$. Similar to the result in Proposition 2 for DQM we can show that the DLM error vector norm $\|\mathbf{e}_k^{\mathrm{DLM}}\|$ is bounded by a factor of $\| \mathbf{x}_{k+1} - \mathbf{x}_k \|$.

Proposition 3: Consider the DLM algorithm with updates in (7)–(9) and the error vector $\mathbf{e}_k^{\mathrm{DLM}}$ defined in (34). If Assumptions 1–4 hold true, the DLM error vector norm $\|\mathbf{e}_k^{\mathrm{DLM}}\|$ satisfies

$$\|\mathbf{e}_k^{\mathrm{DLM}}\| \leq (\rho + M) \| \mathbf{x}_{k+1} - \mathbf{x}_k \|. \quad (39)$$

Proof: See Appendix D. ■

The result in Proposition 3 differs from Proposition 2 in that the DLM error $\|\mathbf{e}_k^{\mathrm{DLM}}\|$ vanishes at a rate of $\| \mathbf{x}_{k+1} - \mathbf{x}_k \|$ whereas the DQM error $\|\mathbf{e}_k^{\mathrm{DQM}}\|$ eventually becomes proportional to $\| \mathbf{x}_{k+1} - \mathbf{x}_k\|^2$. This results in DLM failing to approach the convergence behavior of DADMM as we show in the following theorem.

Theorem 2: Consider the DLM method as introduced in (7)–(9). Assume that the constant c is chosen such that $c > (\rho + M)^2/(m\gamma_u^2)$. Moreover, consider $\mu, \mu' > 1$ as arbitrary constants and η as a positive constant chosen from the interval $((\rho + M)/m, c\gamma_u^2/(\rho + M))$. If Assumptions 1–4 hold true, then the sequence $\| \mathbf{u}_k - \mathbf{u}^*\|_{\mathbf{C}}^2$ generated by DLM satisfies

$$\| \mathbf{u}_{k+1} - \mathbf{u}^*\|_{\mathbf{C}}^2 \leq \frac{1}{1+\delta} \| \mathbf{u}_k - \mathbf{u}^* \|_{\mathbf{C}}^2 \quad (40)$$

where the scalar δ is given by

$$\delta = \min\left\{ \frac{(\mu - 1)\left(c\gamma_u^2 - \eta_k(\rho + M)\right)\gamma_o^2}{\mu\mu'\left(c\Gamma_u^2\gamma_u^2 + 4(\rho+M)^2/c(\mu' - 1)\right)}, \right.$$
$$\left. \frac{m - (\rho + M)/\eta_k}{c\Gamma_u^2/4 + \mu M^2/c\gamma_o^2} \right\}. \quad (41)$$

Proof: See Appendix F. ■

Based on the result in Theorem 2, the sequence $\| \mathbf{u}_{k+1} - \mathbf{u}^*\|_{\mathbf{C}}^2$ generated by DLM converges linearly to 0. This result is similar to the convergence properties of DQM as shown in Theorem 1; however, the factor of linear convergence $1/(1 + \delta)$ in (36) is smaller than the factor $1/(1 + \delta_k)$ in (33).

V. Numerical Analysis

In this section we compare the performances of DLM, DQM and DADMM in solving a logistic regression problem. Consider a training set with points whose classes are known and the goal is finding the classifier that minimizes the loss function. Let q be the number of training points available at each node of the network. Therefore, the total number of training points is nq. The training set $\{\mathbf{s}_{il}, y_{il}\}_{l=1}^q$ at node i contains q pairs of $(\mathbf{s}_{il}, y_{il})$, where $\mathbf{s}_{il}$ is a feature vector and $y_{il} \in \{-1, 1\}$ is the corresponding class. The goal is to estimate the probability $\mathrm{P}(y = 1 \mid \mathbf{s})$ of having label $y = 1$ for a given feature vector $\mathbf{s}$ whose class is not known. Logistic regression models this probability as $\mathrm{P}(y = 1 \mid \mathbf{s}) = 1/(1 + \exp(-\mathbf{s}^T\tilde{\mathbf{x}}))$ for a linear classifier $\tilde{\mathbf{x}}$ that is computed based on the training samples. It follows from this model that the maximum log-likelihood estimate of the classifier $\tilde{\mathbf{x}}$ given the training samples $\{\{\mathbf{s}_{il}, y_{il}\}_{l=1}^q\}_{i=1}^n$ is

$$\tilde{\mathbf{x}}^* := \underset{\tilde{\mathbf{x}} \in \mathbb{R}^p}{\operatorname{argmin}} \sum_{i=1}^n \sum_{l=1}^q \log\left[1 + \exp\left(-y_{il}\mathbf{s}_{il}^T\tilde{\mathbf{x}}\right)\right]. \quad (42)$$

The optimization problem in (38) can be written in the form (1). To do so, simply define the local objective functions f_i as

$$f_i(\tilde{\mathbf{x}}) = \sum_{l=1}^q \log\left[1 + \exp\left(-y_{il}\mathbf{s}_{il}^T\tilde{\mathbf{x}}\right)\right]. \quad (43)$$

We define the optimal argument for decentralized optimization as $\mathbf{x}^* = [\tilde{\mathbf{x}}^*; \ldots; \tilde{\mathbf{x}}^*]$. Note that the reference (ground truth) logistic classifiers $\tilde{\mathbf{x}}^*$ for all the experiments in this section are pre-computed with a centralized method.

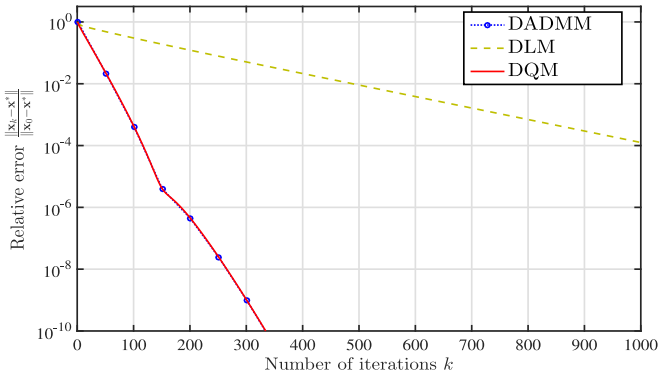


Fig. 1. Relative error $\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \|$ of DADMM, DQM, and DLM versus number of iterations for a random network of size $n = 10$. The convergence path of DQM is similar to the one for DADMM and they outperform DLM by orders of magnitude.

A. Comparison of DLM, DQM, and DADMM

We compare the convergence paths of the DLM, DQM, and DADMM algorithms for solving the logistic regression problem in (38). Edges between the nodes are randomly generated with the connectivity ratio r_c. Observe that the connectivity ratio r_c is the probability of two nodes being connected.

In the first experiment we set the number of nodes as $n = 10$ and the connectivity ratio as $r_c = 0.4$. Each agent holds $q = 5$ samples and the dimension of feature vectors is $p = 3$. Fig. 1 illustrates the relative errors $\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \|$ for DLM, DQM, and DADMM versus the number of iterations. In the experiments, we have hand-optimized the parameter c for these three algorithms separately and reported the convergence result for the optimal choice of c. The optimal choices for the three methods are $c_{\mathrm{ADMM}} = 0.7, c_{\mathrm{DLM}} = 5.5$, and $c_{\mathrm{DQM}} = 0.7$. The convergence path of DQM is almost identical to the convergence path of DADMM. Moreover, DQM outperforms DLM by orders of magnitude. To be more precise, the relative errors $\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \|$ for DQM and DADMM after $k = 300$ iterations are below 10^{-9}, while for DLM the relative error after the same number of iterations is 5×10^{-2}. Conversely, achieving accuracy

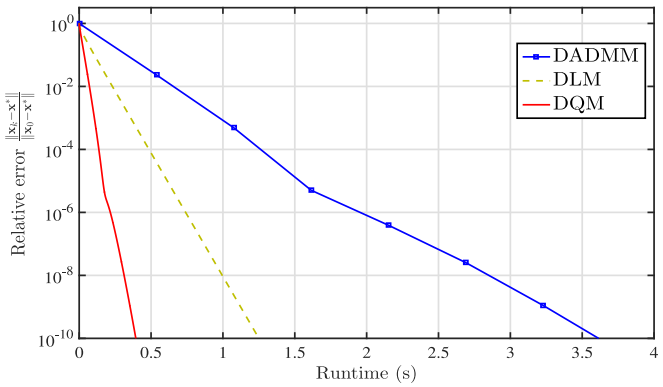


Fig. 2. Relative error $\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \|$ of DADMM, DQM, and DLM versus runtime for the setting in Fig. 1. The computational cost of DQM is lower than DADMM and DLM.

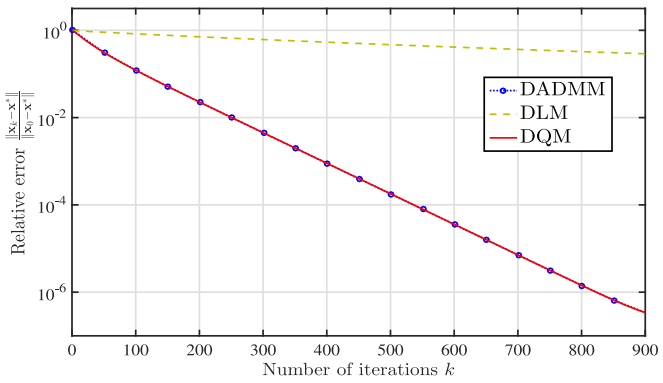


Fig. 3. Relative error $\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \|$ of DADMM, DQM, and DLM versus number of iterations for a random network of size $n = 100$. The performances of DQM and DADMM are almost identical. DLM is impractical in this setting.

$\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \| = 10^{-3}$ for DQM and DADMM requires 91 iterations, while DLM requires 758 iterations to reach the same accuracy. Hence, the number of iterations that DLM requires to achieve a specific accuracy is 8 times more than the one for DQM.

Observe that the computational complexity of DQM is lower than DADMM. Therefore, DQM outperforms DADMM in terms of convergence time or number of required operations until convergence. This phenomenon is shown in Fig. 2 by comparing the relative of errors of DLM, DQM, and DADMM versus CPU runtime. According to Fig. 2, DADMM achieves the relative error $\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \| = 10^{-10}$ after running for 3.6 seconds, while DLM and DQM require 1.3 and 0.4 seconds, respectively, to achieve the same accuracy.

We also compare the performances of DLM, DQM, and DADMM in a larger scale logistic regression problem by setting size of network $n = 100$, number of sample points at each node $q = 20$, and dimension of feature vectors $p = 10$. We keep the rest of the parameters as in Fig. 1. Convergence paths of the relative errors $\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \|$ for DLM, DQM, and DADMM versus the number of iterations are illustrated in Fig. 3. Different choices of parameter c are considered for these algorithms and the best for each is chosen for the final comparison.

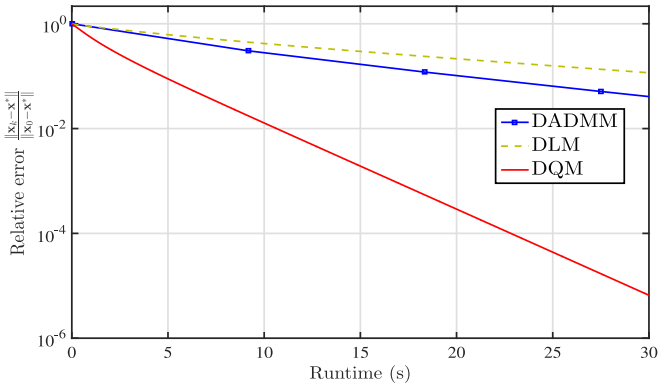


Fig. 4. Relative error $\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \|$ of DADMM, DQM, and DLM versus runtime for the setting in Fig. 3. The convergence time of DADMM is slightly faster relative to DLM, while DQM is the most efficient method among these three algorithms.

The optimal choices of parameter c for DADMM, DLM, and DQM are $c_{\mathrm{ADMM}} = 0.68$, $c_{\mathrm{DLM}} = 12.3$, and $c_{\mathrm{DQM}} = 0.68$, respectively. The results for the large scale problem in Fig. 3 are similar to the results in Fig. 1. We observe that DQM performs as well as DADMM, while both outperform DLM. To be more precise, DQM and DADMM after $k = 900$ iterations reach the relative error $\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \| = 3.4 \times 10^{-7}$, while the relative error of DLM after the same number of iterations is 2.9×10^{-1}. Conversely, achieving the accuracy $\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \| = 0.3$ for DQM and DADMM requires 52 iterations, while DLM requires 870 iterations to reach the same accuracy. Hence, in this setting the number of iterations that DLM requires to achieve a specific accuracy is 16 times more than the one for DQM. These numbers show that the advantages of DQM relative to DLM are more significant in large scale problems.

Notice that in large scale logistic regression problems we expect larger condition number for the objective function f. In these scenarios we expect to observe a poor performance by the DLM algorithm that only operates on first-order information. This expectation is satisfied by comparing the relative errors of DLM, DQM, and DADMM versus runtime for the large scale problem in Fig. 4. In this case, DLM is even worse than DADMM that has a very high computational complexity. Similar to the result in Fig. 3, DQM has the best performance among these three methods.

Notice that the inequalities in (30) for DQM and in Theorem 1 of [21] for DADMM give upper bounds for the convergence rate of these algorithms. One may say in practice the linear convergence factor of DQM does not approach the factor of DADMM. However, the numerical results show that in practice the factor of linear convergence of DQM is almost identical to the one for DADMM which verifies the theoretical result in Section IV.

B. Effect of the Regularization Parameter c

The parameter c has a significant role in the convergence of DADMM. Likewise, choosing the optimal choice of c is critical in the convergence of DQM. We study the effect of c by tuning this parameter for a fixed network and training set. We use all the parameters in Fig. 1 and we compare performance of the DQM algorithm for the values $c = 0.2, c = 0.4, c = 0.8$, and $c = 1$.

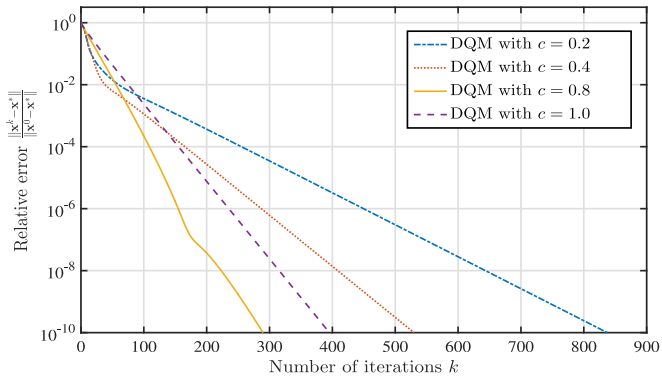


Fig. 5. Relative error $\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \|$ of DQM for parameters $c = 0.2$, $c = 0.4$, $c = 0.8$, and $c = 1$ when the network is formed by $n = 10$ nodes and the connectivity ratio is $r_c = 0.4$. The best performance belongs to $c = 0.8$.

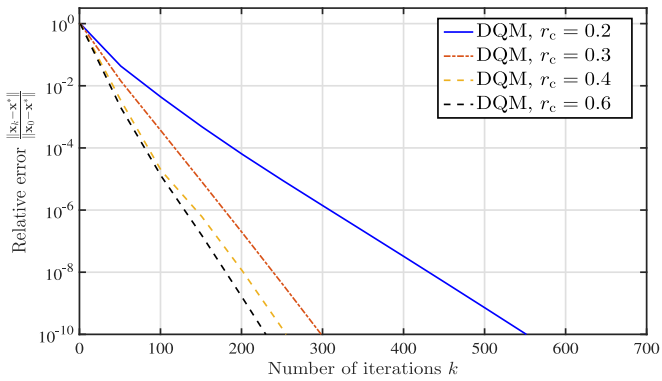


Fig. 6. Relative error $\| \mathbf{x}_k - \mathbf{x}^* \| / \| \mathbf{x}_0 - \mathbf{x}^* \|$ of DQM for random graphs with different connectivity ratios r_c. The linear convergence of DQM accelerates by increasing the connectivity ratio.

Fig. 5 illustrates the convergence paths of the DQM algorithm for different choices of the parameter c. The best performance among these choices is achieved for $c = 0.8$. The comparison of the plots in Fig. 5 shows that increasing or decreasing the parameter c is not necessarily leads to a faster convergence. We can interpret c as the stepsize of DQM for which the optimal choice may vary for problems with different network sizes, network topologies, condition numbers of objective functions, etc.

C. Effect of Network Topology

According to (31), the factor of linear convergence for DQM depends on the bounds for the singular values of the oriented and unoriented incidence matrices $\mathbf{E}_o$ and $\mathbf{E}_u$. These bounds are related to connectivity ratio of the network. We study how network topology affects the convergence speed of DQM. We use different values for the connectivity ratio to generate random graphs with different number of edges. In this experiment we use the connectivity ratios $r_c = \{0.2, 0.3, 0.4, 0.6\}$ to generate the networks. The rest of the parameters are the same as the parameters in Fig. 1. Notice that since the connectivity parameters of these graphs are different, the optimal choices of c for these graphs are different. The convergence paths of DQM with the connectivity ratios $r_c = \{0.2, 0.3, 0.4, 0.6\}$ are shown in Fig. 6. The optimal choices of the parameter c for these graphs are $c_{0.2} = 0.28, c_{0.3} = 0.25, c_{0.4} = 0.31$, and $c_{0.6} = 0.28$, respectively. Fig. 6 shows that the linear convergence of DQM accelerates by increasing the connectivity ratio of the graph.

VI. Conclusion

A decentralized quadratically approximated version of the alternating direction method of multipliers (DQM) is proposed for solving decentralized optimization problems where components of the objective function are available at different nodes of a network. DQM minimizes a quadratic approximation of the convex problem that DADMM solves exactly at each step, and hence reduces the computational complexity of DADMM. Under some mild assumptions, linear convergence of the sequence generated by DQM is proven. Moreover, the factor of linear convergence for DQM approaches that of DADMM asymptotically. Numerical results for a logistic regression problem verify the analytical results that convergence paths of DQM and DADMM are similar for large iteration index, while the computational complexity of DQM is significantly smaller than DADMM.

Appendix A
Proof of Lemma 1

According to the update for the Lagrange multiplier $\boldsymbol{\lambda}$ in (13), we can substitute $\boldsymbol{\lambda}_k$ by $\boldsymbol{\lambda}_{k+1} - c(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{z}_{k+1})$. Applying this substitution into the first equation of (13) leads to

$$\mathbf{B}^T \boldsymbol{\lambda}_{k+1} = \mathbf{0}. \tag{44}$$

Observing the definitions $\mathbf{B} = [-\mathbf{I}_{mp}; -\mathbf{I}_{mp}]$ and $\boldsymbol{\lambda} = [\boldsymbol{\alpha}; \boldsymbol{\beta}]$, and the result in (40), we obtain $\boldsymbol{\alpha}_{k+1} = -\boldsymbol{\beta}_{k+1}$ for $k \geq 0$. Considering the initial condition $\boldsymbol{\alpha}_0 = -\boldsymbol{\beta}_0$, we obtain that $\boldsymbol{\alpha}_k = -\boldsymbol{\beta}_k$ for $k \geq 0$ which follows the first claim in Lemma 1.

Based on the definitions $\mathbf{A} = [\mathbf{A}_s; \mathbf{A}_d]$, $\mathbf{B} = [-\mathbf{I}_{mp}; -\mathbf{I}_{mp}]$, and $\boldsymbol{\lambda} = [\boldsymbol{\alpha}; \boldsymbol{\beta}]$, we can split the update for the Lagrange multiplier $\boldsymbol{\lambda}$ in (8) as

$$\boldsymbol{\alpha}_{k+1} = \boldsymbol{\alpha}_k + c\left[\mathbf{A}_s \mathbf{x}_{k+1} - \mathbf{z}_{k+1}\right], \tag{45}$$

$$\boldsymbol{\beta}_{k+1} = \boldsymbol{\beta}_k + c\left[\mathbf{A}_d \mathbf{x}_{k+1} - \mathbf{z}_{k+1}\right]. \tag{46}$$

Observing the result that $\boldsymbol{\alpha}_k = -\boldsymbol{\beta}_k$ for $k \geq 0$, summing up the equations in (41) and (42) yields

$$\left(\mathbf{A}_s + \mathbf{A}_d\right) \mathbf{x}_{k+1} = 2\mathbf{z}_{k+1}. \tag{47}$$

Considering the definition of the oriented incidence matrix $\mathbf{E}_u = \mathbf{A}_s + \mathbf{A}_d$, we obtain that $\mathbf{E}_u \mathbf{x}_k = 2\mathbf{z}_k$ holds for $k > 0$. According to the initial condition $\mathbf{E}_u \mathbf{x}_0 = 2\mathbf{z}_0$, we can conclude that the relation $\mathbf{E}_u \mathbf{x}_k = 2\mathbf{z}_k$ holds for $k \geq 0$.

Subtract the update for $\boldsymbol{\beta}_k$ in (42) from the update for $\boldsymbol{\alpha}_k$ in (41) and consider the relation $\boldsymbol{\beta}_k = -\boldsymbol{\alpha}_k$ to obtain

$$\boldsymbol{\alpha}_{k+1} = \boldsymbol{\alpha}_k + \frac{c}{2}\left(\mathbf{A}_s - \mathbf{A}_d\right) \mathbf{x}_{k+1}. \tag{48}$$

Substituting $\mathbf{A}_s - \mathbf{A}_d$ in (44) by $\mathbf{E}_o$ implies that

$$\boldsymbol{\alpha}_{k+1} = \boldsymbol{\alpha}_k + \frac{c}{2}\mathbf{E}_o \mathbf{x}_{k+1}. \tag{49}$$

Hence, if $\boldsymbol{\alpha}_k$ lies in the column space of matrix $\mathbf{E}_o$, then $\boldsymbol{\alpha}_{k+1}$ also lies in the column space of $\mathbf{E}_o$. According to the third

condition of Assumption 1, $\boldsymbol{\alpha}_0$ satisfies this condition, therefore $\boldsymbol{\alpha}_k$ lies in the column space of matrix $\mathbf{E}_o$ for all $k \geq 0$.

APPENDIX B
PROOF OF PROPOSITION 1

Based on the update for the multiplier $\boldsymbol{\lambda}$ in (8), we can substitute $\boldsymbol{\lambda}_k$ by $\boldsymbol{\lambda}_{k+1} - c\left(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{z}_{k+1}\right)$ to simplify (12) as

$$\begin{aligned}\nabla f\left(\mathbf{x}_k\right) \;& + \mathbf{H}_k\left(\mathbf{x}_{k+1} - \mathbf{x}_k\right) + \mathbf{A}^T\boldsymbol{\lambda}_{k+1} \\ & + c\mathbf{A}^T\mathbf{B}\left(\mathbf{z}_k - \mathbf{z}_{k+1}\right) = \mathbf{0}. \end{aligned} \tag{50}$$

Considering the first result of Lemma 1 that $\boldsymbol{\alpha}_k = -\boldsymbol{\beta}_k$ for $k \geq 0$ in association with the definition $\mathbf{A} = [\mathbf{A}_s; \mathbf{A}_d]$ implies that the product $\mathbf{A}^T\boldsymbol{\lambda}_{k+1}$ is equivalent to

$$\mathbf{A}^T\boldsymbol{\lambda}_{k+1} = \mathbf{A}_s^T\boldsymbol{\alpha}_{k+1} + \mathbf{A}_d^T\boldsymbol{\beta}_{k+1} = \left(\mathbf{A}_s - \mathbf{A}_d\right)^T\boldsymbol{\alpha}_{k+1}. \tag{51}$$

According to the definition $\mathbf{E}_o := \mathbf{A}_s - \mathbf{A}_d$, the right hand side of (47) can be simplified as

$$\mathbf{A}^T\boldsymbol{\lambda}_{k+1} = \mathbf{E}_o^T\boldsymbol{\alpha}_{k+1}. \tag{52}$$

Based on the structures of the matrices $\mathbf{A}$ and $\mathbf{B}$, and the definition $\mathbf{E}_u := \mathbf{A}_s + \mathbf{A}_d$, we can simplify $\mathbf{A}^T\mathbf{B}$ as

$$\mathbf{A}^T\mathbf{B} = -\mathbf{A}_s^T - \mathbf{A}_d^T = -\mathbf{E}_u^T. \tag{53}$$

Substituting the results in (48) and (49) into (46) leads to

$$\begin{aligned}\nabla f\left(\mathbf{x}_k\right) \; & + \mathbf{H}_k\left(\mathbf{x}_{k+1} - \mathbf{x}_k\right) + \mathbf{E}_o^T\boldsymbol{\alpha}_{k+1} \\ & + c\mathbf{E}_u^T\left(\mathbf{z}_{k+1} - \mathbf{z}_k\right) = \mathbf{0}. \end{aligned} \tag{54}$$

The second result in Lemma 1 states that $\mathbf{z}_k = \mathbf{E}_u\mathbf{x}_k/2$. Multiplying both sides of this equality by $\mathbf{E}_u^T$ from left we obtain that $\mathbf{E}_u^T\mathbf{z}_k = \mathbf{E}_u^T\mathbf{E}_u\mathbf{x}_k/2$ for $k \geq 0$. Observing the definition of the unoriented Laplacian $\mathbf{L}_u := \mathbf{E}_u^T\mathbf{E}_u/2$, we obtain that the product $\mathbf{E}_u^T\mathbf{z}_k$ is equal to $\mathbf{L}_u\mathbf{x}_k$ for $k \geq 0$. Thus, we can substitute $\mathbf{E}_u^T\left(\mathbf{z}_{k+1} - \mathbf{z}_k\right)$ by $\mathbf{L}_u\left(\mathbf{x}_{k+1} - \mathbf{x}_k\right)$ in (50) to write

$$\nabla f\left(\mathbf{x}_k\right) + \left(\mathbf{H}_k + c\mathbf{L}_u\right)\left(\mathbf{x}_{k+1} - \mathbf{x}_k\right) + \mathbf{E}_o^T\boldsymbol{\alpha}_{k+1} = \mathbf{0}. \tag{55}$$

Observe that the new variables ϕ_k are defined as $\phi_k := \mathbf{E}_o^T\boldsymbol{\alpha}_k$. Multiplying both sides of (45) by $\mathbf{E}_o^T$ from the left hand side and considering the definition of oriented Laplacian $\mathbf{L}_o = \mathbf{E}_o^T\mathbf{E}_o/2$ follows the update rule of ϕ_k in (14), i.e.,

$$\phi_{k+1} = \phi_k + c\mathbf{L}_o\mathbf{x}_{k+1}. \tag{56}$$

According to the definition $\phi_k = \mathbf{E}_o^T\boldsymbol{\alpha}_k$ and the update formula in (52), we can conclude that $\mathbf{E}_o^T\boldsymbol{\alpha}_{k+1} = \phi_{k+1} = \phi_k + c\mathbf{L}_o\mathbf{x}_{k+1}$. Substituting $\mathbf{E}_o^T\boldsymbol{\alpha}_{k+1}$ by $\phi_k + c\mathbf{L}_o\mathbf{x}_{k+1}$ in (51) yields

$$\nabla f\left(\mathbf{x}_k\right) + \left(\mathbf{H}_k + c\mathbf{L}_u\right)\left(\mathbf{x}_{k+1} - \mathbf{x}_k\right) + \phi_k + c\mathbf{L}_o\mathbf{x}_{k+1} = \mathbf{0}. \tag{57}$$

Observing the definition $\mathbf{D} = \left(\mathbf{L}_u + \mathbf{L}_o\right)/2$ we rewrite (53) as

$$\left(\mathbf{H}_k + 2c\mathbf{D}\right)\mathbf{x}_{k+1} = \left(\mathbf{H}_k + c\mathbf{L}_u\right)\mathbf{x}_k - \nabla f\left(\mathbf{x}_k\right) - \phi_k. \tag{58}$$

Multiplying both sides of (54) by $\left(\mathbf{H}_k + 2c\mathbf{D}\right)^{-1}$ from the left hand side yields the first update in (14).

APPENDIX C
PROOF OF LEMMA 2

Consider two arbitrary vectors $\mathbf{x} := [\mathbf{x}_1; \ldots; \mathbf{x}_n] \in \mathbb{R}^{np}$ and $\hat{\mathbf{x}} := [\hat{\mathbf{x}}_1; \ldots; \hat{\mathbf{x}}_n] \in \mathbb{R}^{np}$. Since the aggregate function Hessian is block diagonal where the i-th diagonal block is given by $\nabla^2 f_i\left(\mathbf{x}_i\right)$, we obtain that the difference of Hessians $\mathbf{H}\left(\mathbf{x}\right) - \mathbf{H}\left(\hat{\mathbf{x}}\right)$ is also block diagonal where the i-th diagonal block $\mathbf{H}(\mathbf{x})_{ii} - \mathbf{H}\left(\hat{\mathbf{x}}\right)_{ii}$ is

$$\mathbf{H}(\mathbf{x})_{ii} - \mathbf{H}\left(\hat{\mathbf{x}}\right)_{ii} = \nabla^2 f_i\left(\mathbf{x}_i\right) - \nabla^2 f_i\left(\hat{\mathbf{x}}_i\right). \tag{59}$$

Consider any vector $\mathbf{v} \in \mathbb{R}^{np}$ and separate each p components of vector $\mathbf{v}$ and consider it as a new vector called $\mathbf{v}_i \in \mathbb{R}^p$, i.e., $\mathbf{v} := [\mathbf{v}_1; \ldots; \mathbf{v}_n]$. Observing the relation for the difference $\mathbf{H}\left(\mathbf{x}\right) - \mathbf{H}\left(\hat{\mathbf{x}}\right)$ in (55), the symmetry of matrices $\mathbf{H}\left(\mathbf{x}\right)$ and $\mathbf{H}\left(\hat{\mathbf{x}}\right)$, and the definition of Euclidean norm of a matrix that $\| \mathbf{A} \| = \sqrt{\boldsymbol{\lambda}_{\max}\left(\mathbf{A}^T\mathbf{A}\right)}$, we obtain that the squared difference norm $\| \mathbf{H}\left(\mathbf{x}\right) - \mathbf{H}\left(\hat{\mathbf{x}}\right) \|^2$ can be written as

$$\begin{aligned}\|\mathbf{H}\left(\mathbf{x}\right) - \mathbf{H}\left(\hat{\mathbf{x}}\right)\|^2 &= \max_{\mathbf{v}} \frac{\mathbf{v}^T\left[\mathbf{H}\left(\mathbf{x}\right) - \mathbf{H}\left(\hat{\mathbf{x}}\right)\right]^2\mathbf{v}}{\| \mathbf{v} \|^2} \\ &= \max_{\mathbf{v}} \frac{\sum_{i=1}^n \mathbf{v}_i^T\left[\nabla^2 f_i\left(\mathbf{x}_i\right) - \nabla^2 f_i\left(\hat{\mathbf{x}}_i\right)\right]^2\mathbf{v}_i}{\| \mathbf{v} \|^2} \end{aligned} \tag{60}$$

Using the Cauchy-Schwarz inequality we can write

$$\begin{aligned}\mathbf{v}_i^T\left[\nabla^2 f_i\left(\mathbf{x}_i\right) - \nabla^2 f_i\left(\hat{\mathbf{x}}_i\right)\right]^2\mathbf{v}_i \leq \|\nabla^2 f_i\left(\mathbf{x}_i\right) \\ - \nabla^2 f_i\left(\hat{\mathbf{x}}_i\right)\|^2 \| \mathbf{v}_i \|^2 \end{aligned} \tag{61}$$

Substituting the upper bound in (57) into (56) implies that the squared norm $\|\mathbf{H}\left(\mathbf{x}\right) - \mathbf{H}\left(\hat{\mathbf{x}}\right)\|^2$ is bounded above as

$$\begin{aligned}&\|\mathbf{H}\left(\mathbf{x}\right) - \mathbf{H}\left(\hat{\mathbf{x}}\right)\|^2 \\ &\leq \max_{\mathbf{v}} \frac{\sum_{i=1}^n \|\nabla^2 f_i\left(\mathbf{x}_i\right) - \nabla^2 f_i\left(\hat{\mathbf{x}}_i\right)\|^2 \| \mathbf{v}_i \|^2}{\| \mathbf{v} \|^2}. \end{aligned} \tag{62}$$

Observe that Assumption 3 states that local objective functions Hessian $\nabla^2 f_i\left(\mathbf{x}_i\right)$ are Lipschitz continuous with constant L, i.e., $\| \nabla^2 f_i\left(\mathbf{x}_i\right) - \nabla^2 f_i\left(\hat{\mathbf{x}}_i\right) \| \leq L \| \mathbf{x}_i - \hat{\mathbf{x}}_i \|$. Considering this inequality the upper bound in (58) can be changed by replacing $\|\nabla^2 f_i\left(\mathbf{x}_i\right) - \nabla^2 f_i\left(\hat{\mathbf{x}}_i\right) \|$ by $L \| \mathbf{x}_i - \hat{\mathbf{x}}_i \|$ which yields

$$\|\mathbf{H}\left(\mathbf{x}\right) - \mathbf{H}\left(\hat{\mathbf{x}}\right)\|^2 \leq \max_{\mathbf{v}} \frac{L^2\sum_{i=1}^n \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2 \| \mathbf{v}_i \|^2}{\sum_{i=1}^n \| \mathbf{v}_i \|^2}. \tag{63}$$

Note that for any sequences of scalars such as a_i and b_i, the inequality $\sum_{i=1}^n a_i^2 b_i^2 \leq \left(\sum_{i=1}^n a_i^2\right)\left(\sum_{i=1}^n b_i^2\right)$ holds. If we divide both sides of this relation by $\sum_{i=1}^n b_i^2$ and set $a_i = \| \mathbf{x}_i - \hat{\mathbf{x}}_i \|$ and $b_i = \| \mathbf{v}_i \|$, we obtain

$$\frac{\sum_{i=1}^n \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2 \| \mathbf{v}_i \|^2}{\sum_{i=1}^n \| \mathbf{v}_i \|^2} \leq \sum_{i=1}^n \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2. \tag{64}$$

Combining the two inequalities in (59) and (60) leads to

$$\|\mathbf{H}\left(\mathbf{x}\right) - \mathbf{H}\left(\hat{\mathbf{x}}\right)\|^2 \leq \max_{\mathbf{v}} L^2 \sum_{i=1}^n \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2. \tag{65}$$

Since the right hand side of (61) does not depend on $\mathbf{v}$ we can eliminate the maximization with respect to $\mathbf{v}$. Further, note that according to the structure of vectors $\mathbf{x}$ and $\hat{\mathbf{x}}$, we can write $\|\mathbf{x}-\hat{\mathbf{x}}\|^2=\sum_{i=1}^{n}\|\mathbf{x}_i-\hat{\mathbf{x}}_i\|^2$. These two observations in association with (61) imply that

$$\|\mathbf{H}(\mathbf{x})-\mathbf{H}(\hat{\mathbf{x}})\|^2\leq L^2\|\mathbf{x}-\hat{\mathbf{x}}\|^2, \tag{66}$$

Computing the square roots of terms in (62) yields (20).

Appendix D
Proofs of Propositions 2 and 3

The fundamental theorem of calculus implies that the difference of gradients $\nabla f(\mathbf{x}_{k+1})-\nabla f(\mathbf{x}_k)$ can be written as

$$\nabla f(\mathbf{x}_{k+1})-\nabla f(\mathbf{x}_k) = \int_0^1 \mathbf{H}(s\mathbf{x}_{k+1}+(1-s)\mathbf{x}_k)(\mathbf{x}_{k+1}-\mathbf{x}_k)\,ds. \tag{67}$$

By computing norms of both sides of (63) and considering that norm of integral is smaller than integral of norm we obtain that

$$\|\nabla f(\mathbf{x}_{k+1})-\nabla f(\mathbf{x}_k)\| \leq \int_0^1 \|\mathbf{H}(s\mathbf{x}_{k+1}+(1-s)\mathbf{x}_k)(\mathbf{x}_{k+1}-\mathbf{x}_k)\|ds. \tag{68}$$

The upper bound M for the eigenvalues of the Hessians as in (19), implies that $\|\mathbf{H}(s\mathbf{x}+(1-s)\hat{\mathbf{x}})(\mathbf{x}-\hat{\mathbf{x}})\|\leq M\|\mathbf{x}-\hat{\mathbf{x}}\|$. Substituting this upper bound into (64) leads to

$$\|\nabla f(\mathbf{x}_{k+1})-\nabla f(\mathbf{x}_k)\|\leq M\|\mathbf{x}_{k+1}-\mathbf{x}_k\|. \tag{69}$$

The error vector norm $\|\mathbf{e}_k^{\mathrm{DLM}}\|$ in (34) is bounded above as

$$\|\mathbf{e}_k^{\mathrm{DLM}}\|\leq\|\nabla f(\mathbf{x}_{k+1})-\nabla f(\mathbf{x}_k)\|+\rho\|\mathbf{x}_{k+1}-\mathbf{x}_k\|. \tag{70}$$

By substituting the upper bound for $\|\nabla f(\mathbf{x}_{k+1})-\nabla f(\mathbf{x}_k)\|$ in (65) into (66), the claim in (35) follows.

To prove (22), first we show that $\|\mathbf{e}_k^{\mathrm{DQM}}\|\leq 2M\|\mathbf{x}_{k+1}-\mathbf{x}_k\|$ holds. Observe that the norm of the error vector $\mathbf{e}_k^{\mathrm{DQM}}$ defined in (21) can be upper bounded using the triangle inequality as

$$\|\mathbf{e}_k^{\mathrm{DQM}}\|\leq\|\nabla f(\mathbf{x}_{k+1})-\nabla f(\mathbf{x}_k)\|+\|\mathbf{H}_k(\mathbf{x}_{k+1}-\mathbf{x}_k)\|. \tag{71}$$

Based on the Cauchy-Schwarz inequality and the upper bound M for the eigenvalues of Hessians as in (19), we obtain $\|\mathbf{H}_k(\mathbf{x}_{k+1}-\mathbf{x}_k)\|\leq M\|\mathbf{x}_{k+1}-\mathbf{x}_k\|$. Further, as mentioned in (65) the difference of gradients $\|\nabla f(\mathbf{x}_{k+1})-\nabla f(\mathbf{x}_k)\|$ is upper bounded by $M\|\mathbf{x}_{k+1}-\mathbf{x}_k\|$. Substituting these upper bounds for the terms in the right hand side of (67) yields

$$\|\mathbf{e}_k^{\mathrm{DQM}}\|\leq 2M\|\mathbf{x}_{k+1}-\mathbf{x}_k\|. \tag{72}$$

The next step is to show that $\|\mathbf{e}_k^{\mathrm{DQM}}\|\leq(L/2)\|\mathbf{x}_{k+1}-\mathbf{x}_k\|^2$. Adding and subtracting the integral $\int_0^1\mathbf{H}(\mathbf{x}_k)(\mathbf{x}_{k+1}-\mathbf{x}_k)\,ds$ to the right hand side of (63) results in

$$\nabla f(\mathbf{x}_{k+1})-\nabla f(\mathbf{x}_k)=\int_0^1\mathbf{H}(\mathbf{x}_k)(\mathbf{x}_{k+1}-\mathbf{x}_k)\,ds + \int_0^1[\mathbf{H}(s\mathbf{x}_{k+1}+(1-s)\mathbf{x}_k)-\mathbf{H}(\mathbf{x}_k)](\mathbf{x}_{k+1}-\mathbf{x}_k)\,ds. \tag{73}$$

First observe that the integral $\int_0^1\mathbf{H}(\mathbf{x}_k)(\mathbf{x}_{k+1}-\mathbf{x}_k)\,ds$ can be simplified as $\mathbf{H}(\mathbf{x}_k)(\mathbf{x}_{k+1}-\mathbf{x}_k)$. Observing this simplification and regrouping the terms yield

$$\nabla f(\mathbf{x}_{k+1})-\nabla f(\mathbf{x}_k)-\mathbf{H}(\mathbf{x}_k)(\mathbf{x}_{k+1}-\mathbf{x}_k) = \int_0^1[\mathbf{H}(s\mathbf{x}_{k+1}+(1-s)\mathbf{x}_k)-\mathbf{H}(\mathbf{x}_k)](\mathbf{x}_{k+1}-\mathbf{x}_k)\,ds. \tag{74}$$

Computing norms of both sides of (70), considering the fact that norm of integral is smaller than integral of norm, and using Cauchy-Schwarz inequality lead to

$$\|\nabla f(\mathbf{x}_{k+1})-\nabla f(\mathbf{x}_k)-\mathbf{H}(\mathbf{x}_k)(\mathbf{x}_{k+1}-\mathbf{x}_k)\| \leq\int_0^1\|\mathbf{H}(s\mathbf{x}_{k+1}+(1-s)\mathbf{x}_k)-\mathbf{H}(\mathbf{x}_k)\|\,\|\mathbf{x}_{k+1}-\mathbf{x}_k\|\,ds. \tag{75}$$

Lipschitz continuity of the Hessian as in (20) implies that $\|\mathbf{H}(s\mathbf{x}_{k+1}+(1-s)\mathbf{x}_k)-\mathbf{H}(\mathbf{x}_k)\|\leq sL\|\mathbf{x}_{k+1}-\mathbf{x}_k\|$. By substituting this upper bound into the integral in (71) and substituting the left hand side of (71) by $\|\mathbf{e}_k^{\mathrm{DQM}}\|$ we obtain

$$\|\mathbf{e}_k^{\mathrm{DQM}}\|\leq\int_0^1 sL\|\mathbf{x}_{k+1}-\mathbf{x}_k\|^2ds. \tag{76}$$

Simplification of the integral in (72) follows

$$\|\mathbf{e}_k^{\mathrm{DQM}}\|\leq\frac{L}{2}\|\mathbf{x}_{k+1}-\mathbf{x}_k\|^2. \tag{77}$$

The results in (68) and (73) follow the claim in (22).

Appendix E
Proof of Lemma 3

In this section we first introduce an equivalent version of Lemma 3 for the DLM algorithm. Then, we show the validity of both lemmata in a general proof.

Lemma 4: Consider DLM as defined by (7)–(9). If Assumption 1 holds true, then the optimal arguments $\mathbf{x}^*,\mathbf{z}^*$, and $\boldsymbol{\alpha}^*$ satisfy

$$\nabla f(\mathbf{x}_{k+1})-\nabla f(\mathbf{x}^*)+\mathbf{e}_k^{\mathrm{DLM}}+\mathbf{E}_o^T(\boldsymbol{\alpha}_{k+1}-\boldsymbol{\alpha}^*) - c\mathbf{E}_u^T(\mathbf{z}_k-\mathbf{z}_{k+1})=\mathbf{0}, \tag{78}$$

$$2(\boldsymbol{\alpha}_{k+1}-\boldsymbol{\alpha}_k)-c\mathbf{E}_o(\mathbf{x}_{k+1}-\mathbf{x}^*)=\mathbf{0}, \tag{79}$$

$$\mathbf{E}_u(\mathbf{x}_k-\mathbf{x}^*)-2(\mathbf{z}_k-\mathbf{z}^*)=\mathbf{0}. \tag{80}$$

Notice that the claims in Lemmata 3 and 4 are identical except in the error term of the first equalities. To provide a general framework to prove the claim in these lemmata we introduce $\mathbf{e}_k$

as the general error vector. By replacing $\mathbf{e}_k$ with $\mathbf{e}_k^{\mathrm{DQM}}$ we obtain the result of DQM in Lemma 3 and by setting $\mathbf{e}_k = \mathbf{e}_k^{\mathrm{DLM}}$ the result in Lemma 4 follows. We start with the following Lemma that captures the KKT conditions of optimization problem (4).

Lemma 5: Consider the optimization problem (4). The optimal Lagrange multiplier $\boldsymbol{\alpha}^*$, primal variable $\mathbf{x}^*$ and auxiliary variable $\mathbf{z}^*$ satisfy the following system of equations

$$\nabla f\left(\mathbf{x}^*\right) + \mathbf{E}_o^T \boldsymbol{\alpha}^* = \mathbf{0}, \mathbf{E}_o \mathbf{x}^* = \mathbf{0}, \mathbf{E}_u \mathbf{x}^* = 2\mathbf{z}^*. \tag{81}$$

Proof: First observe that the KKT conditions of the decentralized optimization problem in (4) are given by

$$\nabla f\left(\mathbf{x}^*\right) + \mathbf{A}^T \boldsymbol{\lambda}^* = \mathbf{0}, \mathbf{B}^T \boldsymbol{\lambda}^* = \mathbf{0}, \mathbf{A}\mathbf{x}^* + \mathbf{B}\mathbf{z}^* = \mathbf{0}. \tag{82}$$

Based on the definitions of the matrix $\mathbf{B} = [-\mathbf{I}_{mp}; -\mathbf{I}_{mp}]$ and the optimal Lagrange multiplier $\boldsymbol{\lambda}^* := [\boldsymbol{\alpha}^*; \boldsymbol{\beta}^*]$, we obtain that $\mathbf{B}^T \boldsymbol{\lambda}^* = \mathbf{0}$ in (78) is equivalent to $\boldsymbol{\alpha}^* = -\boldsymbol{\beta}^*$. Considering this result and the definition $\mathbf{A} = [\mathbf{A}_s; \mathbf{A}_d]$, we obtain

$$\mathbf{A}^T \boldsymbol{\lambda}^* = \mathbf{A}_s^T \boldsymbol{\alpha}^* + \mathbf{A}_d^T \boldsymbol{\beta}^* = \left(\mathbf{A}_s - \mathbf{A}_d\right)^T \boldsymbol{\alpha}^*. \tag{83}$$

The definition $\mathbf{E}_o := \mathbf{A}_s - \mathbf{A}_d$ implies that the right hand side of (79) can be simplified as $\mathbf{E}_o^T \boldsymbol{\alpha}^*$ which shows $\mathbf{A}^T \boldsymbol{\lambda}^* = \mathbf{E}_o^T \boldsymbol{\alpha}^*$. Substituting $\mathbf{A}^T \boldsymbol{\lambda}^*$ by $\mathbf{E}_o^T \boldsymbol{\alpha}^*$ into the first equality in (78) follows the first claim in (77).

Decompose the KKT condition $\mathbf{A}\mathbf{x}^* + \mathbf{B}\mathbf{z}^* = \mathbf{0}$ in (78) based on the definitions of $\mathbf{A}$ and $\mathbf{B}$ as

$$\mathbf{A}_s \mathbf{x}^* - \mathbf{z} = \mathbf{0}, \mathbf{A}_d \mathbf{x}^* - \mathbf{z} = \mathbf{0}. \tag{84}$$

Subtracting the equalities in (80) implies that $\left(\mathbf{A}_s - \mathbf{A}_d\right)\mathbf{x}^* = \mathbf{0}$ which by considering the definition $\mathbf{E}_o = \mathbf{A}_s - \mathbf{A}_d$, the second equation in (77) follows. Summing up the equalities in (80) yields $\left(\mathbf{A}_s + \mathbf{A}_d\right)\mathbf{x}^* = 2\mathbf{z}$. This observation in association with the definition $\mathbf{E}_u = \mathbf{A}_s - \mathbf{A}_d$ follows the third equation in (77). ■

Proofs of Lemmata 3 and 4: First note that the results in Lemma 1 are also valid for DLM [30]. Now, consider the first order optimality condition for primal updates of DQM and DLM in (12) and (10), respectively. Further, recall the definitions of error vectors $\mathbf{e}_k^{\mathrm{DQM}}$ and $\mathbf{e}_k^{\mathrm{DLM}}$ in (21) and (34), respectively. Combining these observations we obtain that

$$\nabla f\left(\mathbf{x}_{k+1}\right) + \mathbf{e}_k + \mathbf{A}^T \boldsymbol{\lambda}_k + c\mathbf{A}^T\left(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{z}_k\right) = \mathbf{0}. \tag{85}$$

Notice that by setting $\mathbf{e}_k = \mathbf{e}_k^{\mathrm{DQM}}$ we obtain the update for primal variable of DQM; likewise, setting $\mathbf{e}_k = \mathbf{e}_k^{\mathrm{DLM}}$ yields to the update of DLM.

Observe that the relation $\boldsymbol{\lambda}_k = \boldsymbol{\lambda}_{k+1} - c\left(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{z}_{k+1}\right)$ holds for both DLM and DQM according to the update formula for Lagrange multiplier in (8) and (13). Substituting $\boldsymbol{\lambda}_k$ by $\boldsymbol{\lambda}_{k+1} - c\left(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{z}_{k+1}\right)$ in (81) follows

$$\nabla f\left(\mathbf{x}_{k+1}\right) + \mathbf{e}_k + \mathbf{A}^T \boldsymbol{\lambda}_{k+1} + c\mathbf{A}^T\mathbf{B}\left(\mathbf{z}_k - \mathbf{z}_{k+1}\right) = \mathbf{0}. \tag{86}$$

Based on the result in Lemma 1, the components of the Lagrange multiplier $\boldsymbol{\lambda} = [\boldsymbol{\alpha}; \boldsymbol{\beta}]$ satisfy $\boldsymbol{\alpha}_{k+1} = -\boldsymbol{\beta}_{k+1}$. Hence, the product $\mathbf{A}^T \boldsymbol{\lambda}_{k+1}$ can be simplified as $\mathbf{A}_s^T \boldsymbol{\alpha}_{k+1} - \mathbf{A}_d^T \boldsymbol{\alpha}_{k+1} = \mathbf{E}_o^T \boldsymbol{\alpha}_{k+1}$ considering the definition that $\mathbf{E}_o = \mathbf{A}_s - \mathbf{A}_d$. Furthermore, note that according to the definitions we have that $\mathbf{A} = [\mathbf{A}_s; \mathbf{A}_d]$ and $\mathbf{B} = [-\mathbf{I}; -\mathbf{I}]$ which implies that $\mathbf{A}^T\mathbf{B} = -\left(\mathbf{A}_s + \mathbf{A}_d\right)^T = -\mathbf{E}_u^T$. By making these substitutions into (82) we can write

$$\nabla f\left(\mathbf{x}_{k+1}\right) + \mathbf{e}_k + \mathbf{E}_o^T \boldsymbol{\alpha}_{k+1} - c\mathbf{E}_u^T\left(\mathbf{z}_k - \mathbf{z}_{k+1}\right) = \mathbf{0}. \tag{87}$$

The first result in Lemma 5 is equivalent to $\nabla f\left(\mathbf{x}^*\right) + \mathbf{E}_o^T \boldsymbol{\alpha}^* = \mathbf{0}$. Subtracting both sides of this equation from the relation in (83) follows the first claim of Lemmata 3 and 4.

We proceed to prove the second and third claims in Lemmata 3 and 4. The update formula for $\boldsymbol{\alpha}_k$ in (45) and the second result in Lemma 5 that $\mathbf{E}_o \mathbf{x}^* = 0$ imply that the second claim of Lemmata 3 and 4 are valid. Further, the result in Lemma 1 guarantees that $\mathbf{E}_u \mathbf{x}_k = 2\mathbf{z}_k$. This result in conjunction with the result in Lemma 5 that $\mathbf{E}_u \mathbf{x}^* = 2\mathbf{z}^*$ leads to the third claim of Lemmata 3 and 4.

Appendix F
Proofs of Theorems 1 and 2

To prove Theorems 1 and 2 we show a sufficient condition for the claims in these theorems. Then, we prove these theorems by showing validity of the sufficient condition. To do so, we use the general coefficient $\boldsymbol{\beta}_k$ which is equivalent to ζ_k in the DQM algorithm and equivalent to $\rho + M$ in the DLM method. These definitions and the results in Propositions 2 and 3 imply that

$$\| \mathbf{e}_k \| \leq \boldsymbol{\beta}_k \| \mathbf{x}_{k+1} - \mathbf{x}_k \|, \tag{88}$$

where $\mathbf{e}_k$ is $\mathbf{e}_k^{\mathrm{DQM}}$ in DQM and $\mathbf{e}_k^{\mathrm{DLM}}$ in DLM. The sufficient condition of Theorems 1 and 2 is studied in the following lemma.

Lemma 6: Consider the DLM and DQM algorithms as defined in (7)–(9) and (12)–(13), respectively. Further, conducer δ_k as a sequence of positive scalars. If Assumptions 1–4 hold true then the sequence $\| \mathbf{u}_k - \mathbf{u}^* \|_{\mathbf{C}}^2$ converges linearly as

$$\| \mathbf{u}_{k+1} - \mathbf{u}^* \|_{\mathbf{C}}^2 \leq \frac{1}{1+\delta_k} \| \mathbf{u}_k - \mathbf{u}^* \|_{\mathbf{C}}^2, \tag{89}$$

if the following inequality holds true,

$$\begin{aligned} &\boldsymbol{\beta}_k \| \mathbf{x}_{k+1} - \mathbf{x}^* \| \| \mathbf{x}_{k+1} - \mathbf{x}_k \| + \delta_k c \| \mathbf{z}_{k+1} - \mathbf{z}^* \|^2 \\ &+ \frac{\delta_k}{c} \| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^* \|^2 \\ &\leq m \| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2 + c \| \mathbf{z}_{k+1} - \mathbf{z}_k \|^2 + \frac{1}{c} \| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}_k \|^2. \end{aligned} \tag{90}$$

Proof: Proving linear convergence of the sequence $\| \mathbf{u}_k - \mathbf{u}^* \|_{\mathbf{C}}^2$ as mentioned in (85) is equivalent to showing that

$$\delta_k \| \mathbf{u}_{k+1} - \mathbf{u}^* \|_{\mathbf{C}}^2 \leq \| \mathbf{u}_k - \mathbf{u}^* \|_{\mathbf{C}}^2 - \| \mathbf{u}_{k+1} - \mathbf{u}^* \|_{\mathbf{C}}^2. \tag{91}$$

According to the definition $\| \mathbf{a} \|_{\mathbf{C}}^2 := \mathbf{a}^T \mathbf{C} \mathbf{a}$ we can show that

$$\begin{aligned} &2(\mathbf{u}_k - \mathbf{u}_{k+1})^T \mathbf{C} (\mathbf{u}_{k+1} - \mathbf{u}^*) \\ &= \| \mathbf{u}_k - \mathbf{u}^* \|_{\mathbf{C}}^2 - \| \mathbf{u}_{k+1} - \mathbf{u}^* \|_{\mathbf{C}}^2 - \| \mathbf{u}_k - \mathbf{u}_{k+1} \|_{\mathbf{C}}^2. \end{aligned} \tag{92}$$

The relation in (88) shows that the right hand side of (87) can be substituted by $2(\mathbf{u}_k - \mathbf{u}_{k+1})^T \mathbf{C} (\mathbf{u}_{k+1} - \mathbf{u}^*) +$

$\| \mathbf{u}_k - \mathbf{u}_{k+1} \|_{\mathbf{C}}^2$. Applying this substitution into (87) leads to

$$\delta_k \| \mathbf{u}_{k+1} - \mathbf{u}^* \|_{\mathbf{C}}^2 \leq 2(\mathbf{u}_k - \mathbf{u}_{k+1})^T \mathbf{C} (\mathbf{u}_{k+1} - \mathbf{u}^*) + \| \mathbf{u}_k - \mathbf{u}_{k+1} \|_{\mathbf{C}}^2. \quad (93)$$

This observation implies that to prove the linear convergence as claimed in (85), the inequality in (89) should be satisfied.

We proceed by finding a lower bound for the term $2(\mathbf{u}_k - \mathbf{u}_{k+1})^T \mathbf{C}(\mathbf{u}_{k+1} - \mathbf{u}^*)$ in (89). By regrouping the terms in (83) and multiplying both sides of equality by $(\mathbf{x}_{k+1} - \mathbf{x}^*)^T$ from the left hand side we obtain that the inner product $(\mathbf{x}_{k+1} - \mathbf{x}^*)^T (\nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}^*))$ is equivalent to

$$\begin{aligned}(\mathbf{x}_{k+1} - \mathbf{x}^*)^T (\nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}^*)) &= -(\mathbf{x}_{k+1} - \mathbf{x}^*)^T \mathbf{e}_k - (\mathbf{x}_{k+1} - \mathbf{x}^*)^T \mathbf{E}_o^T (\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^*) \\ &+ c(\mathbf{x}_{k+1} - \mathbf{x}^*)^T \mathbf{E}_u^T (\mathbf{z}_k - \mathbf{z}_{k+1}). \end{aligned} \quad (94)$$

Based on (25), we can substitute $(\mathbf{x}_{k+1} - \mathbf{x}^*)^T \mathbf{E}_o^T (\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^*)$ in (90) by $(2/c)(\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}_k)^T (\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^*)$. Further, the result in (26) implies that the term $c(\mathbf{x}_{k+1} - \mathbf{x}^*)^T \mathbf{E}_u^T (\mathbf{z}_k - \mathbf{z}_{k+1})$ in (90) is equivalent to $2c(\mathbf{z}_k - \mathbf{z}_{k+1})^T (\mathbf{z}_{k+1} - \mathbf{z}^*)$. Applying these substitutions into (90) leads to

$$\begin{aligned}(\mathbf{x}_{k+1} - \mathbf{x}^*)^T (\nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}^*)) &= -(\mathbf{x}_{k+1} - \mathbf{x}^*)^T \mathbf{e}_k + \frac{2}{c}(\boldsymbol{\alpha}_k - \boldsymbol{\alpha}_{k+1})^T (\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^*) \\ &+ 2c(\mathbf{z}_k - \mathbf{z}_{k+1})^T (\mathbf{z}_{k+1} - \mathbf{z}^*). \end{aligned} \quad (95)$$

Based on the definitions of matrix $\mathbf{C}$ and vector $\mathbf{u}$ in (28), the last two summands in the right hand side of (91) can be simplified as

$$\begin{aligned}&\frac{2}{c}(\boldsymbol{\alpha}_k - \boldsymbol{\alpha}_{k+1})^T (\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^*) + 2c(\mathbf{z}_k - \mathbf{z}_{k+1})^T (\mathbf{z}_{k+1} - \mathbf{z}^*) \\ &= 2(\mathbf{u}_k - \mathbf{u}_{k+1})^T \mathbf{C} (\mathbf{u}_{k+1} - \mathbf{u}^*). \end{aligned} \quad (96)$$

Considering the simplification in (92) we can rewrite (91) as

$$\begin{aligned}&(\mathbf{x}_{k+1} - \mathbf{x}^*)^T (\nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}^*)) \\ &= -(\mathbf{x}_{k+1} - \mathbf{x}^*)^T \mathbf{e}_k + 2(\mathbf{u}_k - \mathbf{u}_{k+1})^T \mathbf{C} (\mathbf{u}_{k+1} - \mathbf{u}^*). \end{aligned} \quad (97)$$

Observe that the objective function f is strongly convex with constant m which implies the inequality $m \| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2 \leq (\mathbf{x}_{k+1} - \mathbf{x}^*)^T (\nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}^*))$ holds true. Considering this inequality from the strong convexity of objective function f and the simplification for the inner product $(\mathbf{x}_{k+1} - \mathbf{x}^*)^T (\nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}^*))$ in (93), the following inequality holds

$$m \| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2 + (\mathbf{x}_{k+1} - \mathbf{x}^*)^T \mathbf{e}_k \leq 2(\mathbf{u}_k - \mathbf{u}_{k+1})^T \mathbf{C} (\mathbf{u}_{k+1} - \mathbf{u}^*). \quad (98)$$

Substituting the lower bound for the term $2(\mathbf{u}_k - \mathbf{u}_{k+1})^T \mathbf{C} (\mathbf{u}_{k+1} - \mathbf{u}^*)$ in (94) into (89), it follows that the following condition is sufficient to have (85),

$$\delta_k \| \mathbf{u}_{k+1} - \mathbf{u}^* \|_{\mathbf{C}}^2 \leq m \| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2 + (\mathbf{x}_{k+1} - \mathbf{x}^*)^T \mathbf{e}_k + \| \mathbf{u}_k - \mathbf{u}_{k+1} \|_{\mathbf{C}}^2. \quad (99)$$

We emphasize that inequality (95) implies the linear convergence result in (85). Therefore, our goal is to show that if (86) holds, the relation in (95) is also valid and consequently the result in (85) holds. According to the definitions of matrix $\mathbf{C}$ and vector $\mathbf{u}$ in (28), we can substitute $\| \mathbf{u}_{k+1} - \mathbf{u}^* \|_{\mathbf{C}}^2$ by $c \| \mathbf{z}_{k+1} - \mathbf{z}^* \|^2 + (1/c) \| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^* \|^2$ and $\| \mathbf{u}_k - \mathbf{u}_{k+1} \|_{\mathbf{C}}^2$ by $c \| \mathbf{z}_{k+1} - \mathbf{z}_k \|^2 + (1/c) \| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}_k \|^2$. Making these substitutions into (95) yields

$$\begin{aligned}\delta_k c \| \mathbf{z}_{k+1} - \mathbf{z}^* \|^2 + \frac{\delta_k}{c} \| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^* \|^2 &\leq m \| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2 + (\mathbf{x}_{k+1} - \mathbf{x}^*)^T \mathbf{e}_k \\ &+ c \| \mathbf{z}_{k+1} - \mathbf{z}_k \|^2 + \frac{1}{c} \| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}_k \|^2. \end{aligned} \quad (100)$$

The inequality in (84) implies that $- \| \mathbf{e}_k \|$ is lower bounded by $-\boldsymbol{\beta}_k \| \mathbf{x}_{k+1} - \mathbf{x}_k \|$. This lower bound in conjunction with the fact that inner product of two vectors is not smaller than the negative of their norms product leads to

$$(\mathbf{x}_{k+1} - \mathbf{x}^*)^T \mathbf{e}_k \geq -\boldsymbol{\beta}_k \| \mathbf{x}_{k+1} - \mathbf{x}^* \| \| \mathbf{x}_{k+1} - \mathbf{x}_k \|. \quad (101)$$

Substituting $(\mathbf{x}_{k+1} - \mathbf{x}^*)^T \mathbf{e}_k$ in (96) by its lower bound in (97) leads to a sufficient condition for (96) as in (86), i.e.,

$$\begin{aligned}&\boldsymbol{\beta}_k \| \mathbf{x}_{k+1} - \mathbf{x}^* \| \| \mathbf{x}_{k+1} - \mathbf{x}_k \| + \delta_k c \| \mathbf{z}_{k+1} - \mathbf{z}^* \|^2 + \frac{\delta_k}{c} \| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^* \|^2 \\ &\leq m \| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2 + c \| \mathbf{z}_{k+1} - \mathbf{z}_k \|^2 + \frac{1}{c} \| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}_k \|^2. \end{aligned} \quad (102)$$

Observe that if (98) holds true, then (96) and its equivalence (95) are valid and as a result the inequality in (85) is also satisfied. ■

According to the result in Lemma 6, the sequence $\| \mathbf{u}_k - \mathbf{u}^* \|^2$ converges linearly to 0 as mentioned in (85) if the inequality in (86) holds true. Therefore, in the following proof we show that for

$$\delta_k = \min \left\{ \frac{(\mu - 1)(c\gamma_u^2 - \eta_k \boldsymbol{\beta}_k)\gamma_o^2}{\mu\mu' \left(c\Gamma_u^2 \gamma_u^2 + 4\boldsymbol{\beta}_k^2 / c(\mu' - 1) \right)}, \frac{m - \boldsymbol{\beta}_k / \eta_k}{c\Gamma_u^2 / 4 + \mu M^2 / c\gamma_o^2} \right\}, \quad (103)$$

the inequality in (86) holds and consequently (85) is valid.

Proofs of Theorems 1 and 2: We show that if the constant δ_k is chosen as in (99), then the inequality in (86) holds true. To do this first we should find an upper bound for $\boldsymbol{\beta}_k \| \mathbf{x}_{k+1} - \mathbf{x}^* \| \| \mathbf{x}_{k+1} - \mathbf{x}_k \|$ regarding the terms in the right hand side of (86). Observing the result of Lemma 1 that $\mathbf{E}_u \mathbf{x}_k = 2\mathbf{z}_k$ for times k and $k+1$, we can write

$$\mathbf{E}_u (\mathbf{x}_{k+1} - \mathbf{x}_k) = 2(\mathbf{z}_{k+1} - \mathbf{z}_k). \quad (104)$$

The singular values of $\mathbf{E}_u$ are bounded below by γ_u. Hence, (100) implies that $\| \mathbf{x}_{k+1} - \mathbf{x}_k \|$ is upper bounded by

$$\| \mathbf{x}_{k+1} - \mathbf{x}_k \| \leq \frac{2}{\gamma_u} \| \mathbf{z}_{k+1} - \mathbf{z}_k \| . \tag{105}$$

Multiplying both sides of (101) by $\boldsymbol{\beta}_k \| \mathbf{x}_{k+1} - \mathbf{x}^* \|$ yields

$$\boldsymbol{\beta}_k \| \mathbf{x}_{k+1} - \mathbf{x}^* \| \| \mathbf{x}_{k+1} - \mathbf{x}_k \| \leq \frac{2\boldsymbol{\beta}_k}{\gamma_u} \| \mathbf{x}_{k+1} - \mathbf{x}^* \| \| \mathbf{z}_{k+1} - \mathbf{z}_k \| . \tag{106}$$

Notice that for any vectors $\mathbf{a}$ and $\mathbf{b}$ and positive constant $\eta_k > 0$ the inequality $2 \| \mathbf{a} \| \| \mathbf{b} \| \leq (1/\eta_k) \| \mathbf{a} \|^2 + \eta_k \| \mathbf{b} \|^2$ holds true. By setting $\mathbf{a} = \mathbf{x}_{k+1} - \mathbf{x}^*$ and $\mathbf{b} = \left(1/\gamma_u^2\right)(\mathbf{z}_{k+1} - \mathbf{z}_k)$ the inequality $2 \| \mathbf{a} \| \| \mathbf{b} \| \leq (1/\eta_k) \| \mathbf{a} \|^2 + \eta_k \| \mathbf{b} \|^2$ is equivalent to

$$\frac{2}{\gamma_u} \| \mathbf{x}_{k+1} - \mathbf{x}^* \| \| \mathbf{z}_{k+1} - \mathbf{z}_k \| \leq \frac{1}{\eta_k} \| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2 + \frac{\eta_k}{\gamma_u^2} \| \mathbf{z}_{k+1} - \mathbf{z}_k \|^2. \tag{107}$$

Substituting the upper bound for $(2/\gamma_u) \| \mathbf{x}_{k+1} - \mathbf{x}^* \| \| \mathbf{z}_{k+1} - \mathbf{z}_k \|$ in (103) into (102) yields

$$\boldsymbol{\beta}_k \| \mathbf{x}_{k+1} - \mathbf{x}^* \| \| \mathbf{x}_{k+1} - \mathbf{x}_k \| \leq \frac{\boldsymbol{\beta}_k}{\eta_k} \| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2 + \frac{\eta_k \boldsymbol{\beta}_k}{\gamma_u^2} \| \mathbf{z}_{k+1} - \mathbf{z}_k \|^2. \tag{108}$$

Notice that inequality (104) provides an upper bound for $\boldsymbol{\beta}_k \| \mathbf{x}_{k+1} - \mathbf{x}^* \| \| \mathbf{x}_{k+1} - \mathbf{x}_k \|$ in (86) regarding the terms in the right hand side of inequality which are $\| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2$ and $\| \mathbf{z}_{k+1} - \mathbf{z}_k \|^2$. The next step is to find upper bounds for the other two terms in the left hand side of (86) regarding the terms in the right hand side of (86) which are $\| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2$, $\| \mathbf{z}_{k+1} - \mathbf{z}_k \|^2$, and $\| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}_k \|^2$. First we start with $\| \mathbf{z}_{k+1} - \mathbf{z}^* \|^2$. The relation in (26) and the upper bound Γ_u for the singular values of matrix $\mathbf{E}_u$ yield

$$\delta_k c \| \mathbf{z}_{k+1} - \mathbf{z}^* \|^2 \leq \frac{\delta_k c \Gamma_u^2}{4} \| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2. \tag{109}$$

The next step is to bound $(\delta_k / c) \| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^* \|$ in terms of the term in the right hand side of (47). We use the result of following lemma to derive an upper bound for $\| \mathbf{E}_o^T (\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^*) \|^2$.

Lemma 7: For any vectors $\mathbf{a}, \mathbf{b}$, and $\mathbf{c}$, and constants μ and μ' which are larger than 1, i.e., $\mu, \mu' > 1$, we can write

$$\left(1 - \frac{1}{\mu'}\right)\left(1 - \frac{1}{\mu}\right) \| \mathbf{c} \|^2 \leq \| \mathbf{a} + \mathbf{b} + \mathbf{c} \|^2 + (\mu' - 1) \| \mathbf{a} \|^2 + (\mu - 1)\left(1 - \frac{1}{\mu'}\right) \| \mathbf{b} \|^2. \tag{110}$$

Consider the result in (106) and set

$$\mathbf{a} = c\mathbf{E}_u^T (\mathbf{z}_k - \mathbf{z}_{k+1}), \mathbf{b} = \nabla f(\mathbf{x}^*) - \nabla f(\mathbf{x}_{k+1}),$$

and $\mathbf{c} = \mathbf{E}_o^T (\boldsymbol{\alpha}^* - \boldsymbol{\alpha}_{k+1})$. By choosing these values and observing equality (24) we obtain $\mathbf{a} + \mathbf{b} + \mathbf{c} = \mathbf{e}_k$. Hence, by making these substitutions for $\mathbf{a}, \mathbf{b}, \mathbf{c}$, and $\mathbf{a} + \mathbf{b} + \mathbf{c}$ into (106) we can write

$$\left(1 - \frac{1}{\mu'}\right)\left(1 - \frac{1}{\mu}\right) \| \mathbf{E}_o^T (\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^*) \|^2 \leq \| \mathbf{e}_k \|^2 + (\mu' - 1) \| c\mathbf{E}_u^T (\mathbf{z}_k - \mathbf{z}_{k+1}) \|^2 + (\mu - 1)\left(1 - \frac{1}{\mu'}\right) \| \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}^*) \|^2. \tag{111}$$

Notice that according to the result in Lemma 1, the Lagrange multiplier $\boldsymbol{\alpha}_k$ lies in the column space of $\mathbf{E}_o$ for all $k \geq 0$. Further, recall that the optimal multiplier $\boldsymbol{\alpha}^*$ also lies in the column space of $\mathbf{E}_o$. These observations show that $\boldsymbol{\alpha}^* - \boldsymbol{\alpha}_k$ is in the column space of $\mathbf{E}_o$. Hence, there exits a vector $\mathbf{r} \in \mathbb{R}^{np}$ such that $\boldsymbol{\alpha}^* - \boldsymbol{\alpha}_k = \mathbf{E}_o \mathbf{r}$. This relation implies that $\| \mathbf{E}_o^T (\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^*) \|^2$ can be written as $\| \mathbf{E}_o^T \mathbf{E}_o \mathbf{r} \|^2 = \mathbf{r}^T \left(\mathbf{E}_o^T \mathbf{E}_o\right)^2 \mathbf{r}$. Observe that since the eigenvalues of matrix $\left(\mathbf{E}_o^T \mathbf{E}_o\right)^2$ are the squared of eigenvalues of the matrix $\mathbf{E}_o^T \mathbf{E}_o$, we can write $\mathbf{r}^T \left(\mathbf{E}_o^T \mathbf{E}_o\right)^2 \mathbf{r} \geq \gamma_o^2 \mathbf{r}^T \mathbf{E}_o^T \mathbf{E}_o \mathbf{r}$, where γ_o is the smallest non-zero singular value of the oriented incidence matrix $\mathbf{E}_o$. Observing this inequality and the definition $\boldsymbol{\alpha}^* - \boldsymbol{\alpha}_k = \mathbf{E}_o \mathbf{r}$ we can write

$$\| \mathbf{E}_o^T (\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^*) \|^2 \geq \gamma_o^2 \| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^* \|^2. \tag{112}$$

Observe that the error norm $\| \mathbf{e}_k \|$ is bounded above by $\boldsymbol{\beta}_k \| \mathbf{x}_{k+1} - \mathbf{x}_k \|$ as in (84) and the norm $\| c\mathbf{E}_u^T (\mathbf{z}_k - \mathbf{z}_{k+1}) \|^2$ is upper bounded by $c^2 \Gamma_u^2 \| \mathbf{z}_k - \mathbf{z}_{k+1} \|^2$ since all the singular values of the unoriented matrix $\mathbf{E}_u$ are smaller than Γ_u. Substituting these upper bounds and the lower bound in (108) into (107) implies

$$\left(1 - \frac{1}{\mu'}\right)\left(1 - \frac{1}{\mu}\right) \gamma_o^2 \| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^* \|^2 \leq \boldsymbol{\beta}_k^2 \| \mathbf{x}_{k+1} - \mathbf{x}_k \|^2 + (\mu' - 1) c^2 \Gamma_u^2 \| \mathbf{z}_k - \mathbf{z}_{k+1} \|^2 + (\mu - 1)\left(1 - \frac{1}{\mu'}\right) M^2 \| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2. \tag{113}$$

Considering the result in (101), $\| \mathbf{x}_{k+1} - \mathbf{x}_k \|$ is upper by $(2/\gamma_u) \| \mathbf{z}_{k+1} - \mathbf{z}_k \|$. Therefore, we can substitute $\| \mathbf{x}_{k+1} - \mathbf{x}_k \|$ in the right hand side of (109) by its upper bound $(2/\gamma_u) \| \mathbf{z}_{k+1} - \mathbf{z}_k \|$. Making this substitution, dividing both sides by $(1 - 1/\mu')(1 - 1/\mu)\gamma_o^2$, and regrouping the terms lead to

$$\| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}^* \|^2 \leq \frac{\mu M^2}{\gamma_o^2} \| \mathbf{x}_{k+1} - \mathbf{x}^* \|^2 + \left[\frac{4\mu\mu' \boldsymbol{\beta}_k^2}{\gamma_u^2 \gamma_o^2 (\mu - 1)(\mu' - 1)} + \frac{\mu\mu' c^2 \Gamma_u^2}{(\mu - 1)\gamma_o^2}\right] \| \mathbf{z}_k - \mathbf{z}_{k+1} \|^2. \tag{114}$$

Considering the upper bounds for $\boldsymbol{\beta}_k \| \mathbf{x}_{k+1} - \mathbf{x}^* \| \| \mathbf{x}_{k+1} - \mathbf{x}_k \|$, $\| \mathbf{z}_{k+1} - \mathbf{z}^* \|^2$, and $\| \boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}_k \|^2$, in (104), (105), and

(110), respectively, we obtain that if the inequality

$$\left[\frac{\boldsymbol{\beta}_k}{\eta_k}+\frac{\delta_k c\Gamma_u^2}{4}+\frac{\delta_k \mu M^2}{c\gamma_o^2}\right]\| \mathbf{x}_{k+1}-\mathbf{x}^* \|^2 + \left[\frac{4\delta_k\mu\mu'\boldsymbol{\beta}_k^2}{c\gamma_u^2\gamma_o^2(\mu-1)(\mu'-1)}+\frac{\delta_k\mu\mu' c\Gamma_u^2}{(\mu-1)\gamma_o^2}+\frac{\eta_k\boldsymbol{\beta}_k}{\gamma_u^2}\right] \times \| \mathbf{z}_{k+1}-\mathbf{z}_k\|^2 \leq m\| \mathbf{x}_{k+1}-\mathbf{x}^*\|^2 + c\| \mathbf{z}_{k+1}-\mathbf{z}_k\|^2+\frac{1}{c}\| \boldsymbol{\alpha}_{k+1}-\boldsymbol{\alpha}_k \|^2. \tag{115}$$

holds true, (86) is satisfied. Hence, the last step is to show that for the specific choice of δ_k in (99) the result in (111) is satisfied. In order to make sure that (111) holds, it is sufficient to show that the coefficients of $\| \mathbf{x}_{k+1}-\mathbf{x}^*\|^2$ and $\| \mathbf{z}_{k+1}-\mathbf{z}_k\|^2$ in the left hand side of (111) are smaller than the ones in the right hand side. Hence, we should verify the validity of inequalities

$$\frac{\boldsymbol{\beta}_k}{\eta_k}+\frac{\delta_k c\Gamma_u^2}{4}+\frac{\delta_k\mu M^2}{c\gamma_o^2}\leq m, \tag{116}$$

$$\frac{4\delta_k\mu\mu'\boldsymbol{\beta}_k^2}{c\gamma_u^2\gamma_o^2(\mu-1)(\mu'-1)}+\frac{\delta_k\mu\mu' c\Gamma_u^2}{(\mu-1)\gamma_o^2}+\frac{\eta_k\boldsymbol{\beta}_k}{\gamma_u^2}\leq c. \tag{117}$$

Considering the inequality for δ_k in (99) we obtain that (112) and (113) are satisfied. Hence, if δ_k satisfies condition in (99), (111) and consequently (86) are satisfied. Now recalling the result of Lemma 6 that inequality (86) is a sufficient condition for the linear convergence in (85), we obtain that the linear convergence holds. By setting $\boldsymbol{\beta}_k=\zeta_k$ we obtain the linear convergence of DQM in Theorem 1 is valid and the linear coefficient in (99) can be simplified as (31). Moreover, setting $\boldsymbol{\beta}_k=\rho+M$ follows the linear convergence of DLM as in Theorem 2 with the linear constant in (37).

Proof of Lemma 7: First notice that for any two vectors $\mathbf{x}$ and $\mathbf{y}$ we can write

$$-2\mathbf{x}^T\mathbf{y}\leq\frac{1}{\rho}\| \mathbf{x}\|^2+\rho\| \mathbf{y}\|^2, \tag{118}$$

where $\rho>0$ is a positive scalar. This result holds since $\| \left(1/\sqrt{\rho}\right)\mathbf{x}-\sqrt{\rho}\mathbf{y}\|^2\geq 0$. By adding $\| \mathbf{x}\|^2+\| \mathbf{y}\|^2$ to both sides of (114) and regrouping the terms we obtain

$$\| \mathbf{x}-\mathbf{y}\|^2\leq\left(1+\frac{1}{\rho}\right)\| \mathbf{x}\|^2+(1+\rho)\| \mathbf{y}\|^2. \tag{119}$$

The result in (115) holds for any $\rho>0$. If we define $\zeta>1$ as $\zeta=\rho+1$ we can rewrite (115) as

$$\| \mathbf{x}-\mathbf{y}\|^2\leq\frac{\zeta}{\zeta-1}\| \mathbf{x}\|^2+\zeta\| \mathbf{y}\|^2, \tag{120}$$

for any $\zeta>1$. According to the result in (116), for the vectors $\mathbf{u}=\mathbf{x}-\mathbf{y}$ and $\mathbf{z}$ it is true that

$$\| \mathbf{u}-\mathbf{z}\|^2\leq\frac{\eta}{\eta-1}\| \mathbf{u}\|^2+\eta\| \mathbf{z}\|^2, \tag{121}$$

where $\eta>1$. Substituting $\mathbf{u}$ by $\mathbf{x}-\mathbf{y}$ in (117) implies

$$\| \mathbf{x}-\mathbf{y}-\mathbf{z}\|^2\leq\frac{\eta}{\eta-1}\| \mathbf{x}-\mathbf{y}\|^2+\eta\| \mathbf{z}\|^2. \tag{122}$$

By substituting the upper bound in (116) for $\| \mathbf{x}-\mathbf{y}\|^2$ into (118) we obtain that

$$\| \mathbf{x}-\mathbf{y}-\mathbf{z}\|^2\leq\frac{\zeta}{\zeta-1}\frac{\eta}{\eta-1}\| \mathbf{x}\|^2+\frac{\zeta\eta}{\eta-1}\| \mathbf{y}\|^2+\eta\| \mathbf{z}\|^2. \tag{123}$$

Multiplying both sides of (119) by $(\zeta-1)(\eta-1)/(\zeta\eta)$ yields

$$\left(1-\frac{1}{\zeta}\right)\left(1-\frac{1}{\eta}\right)\| \mathbf{x}-\mathbf{y}-\mathbf{z}\|^2 \leq\| \mathbf{x}\|^2+(\zeta-1)\| \mathbf{y}\|^2+(\eta-1)\left(1-\frac{1}{\zeta}\right)\| \mathbf{z}\|^2. \tag{124}$$

By setting $\eta=\mu,\zeta=\mu',\mathbf{x}=\mathbf{a}+\mathbf{b}+\mathbf{c},\mathbf{y}=\mathbf{a}$, and $\mathbf{z}=\mathbf{b}$ the result in (106) follows.

REFERENCES

[1] A. Mokhtari, W. Shi, Q. Ling, and A. Ribeiro, "Decentralized quadratically approximated alternating direction method of multipliers," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Dec. 2015, pp. 795–799.

[2] F. Bullo, J. Cortés, and S. Martinez, *Distributed Control of Robotic Networks: A Mathematical Approach to Motion Coordination Algorithms*. Princeton, NJ, USA: Princeton Univ. Press, 2009.

[3] 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. Inf.*, vol. 9, pp. 427–438, 2013.

[4] C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3122–3136, 2008.

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

[6] G. Scutari, F. Facchinei, P. Song, D. P. Palomar, and J.-S. Pang, "Decomposition by partial linearization: Parallel optimization of multi-agent systems," *IEEE Trans. Signal Process.*, vol. 62, no. 3, pp. 641–656, 2014.

[7] A. Ribeiro, "Optimal resource allocation in wireless communication and networking," *EURASIP J. Wireless Commun. Netw.*, vol. 2012, no. 1, pp. 1–19, 2012.

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

[9] U. A. Khan, S. Kar, and J. M. Moura, "Diland: An algorithm for distributed sensor localization with noisy distance measurements," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1940–1947, 2010.

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

[11] S. Barbarossa and G. Scutari, "Decentralized maximum-likelihood estimation for sensor networks composed of nonlinearly coupled dynamical systems," *IEEE Trans. Signal Process.*, vol. 55, no. 7, pp. 3456–3470, 2007.

[12] R. Bekkerman, M. Bilenko, and J. Langford, *Scaling Up Machine Learning: Parallel and Distributed Approaches*. Cambridge, U.K.: Cambridge Univ. Press, 2011.

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

[14] V. Cevher, S. Becker, and M. Schmidt, "Convex optimization for big data: Scalable, randomized, and parallel algorithms for big data analytics," *IEEE Signal Process. Mag.*, vol. 31, no. 5, pp. 32–43, 2014.

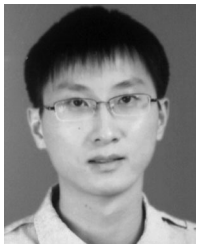
[15] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, 2009.

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

[17] K. Yuan, Q. Ling, and W. Yin, "On the convergence of decentralized gradient descent," 2013, arXiv preprint arXiv:1310.7063.

[18] A. Mokhtari, Q. Ling, and A. Ribeiro, "Network newton—Part I: Algorithm and convergence," 2015, arXiv preprint arXiv:1504.06017.
[19] A. Mokhtari, Q. Ling, and A. Ribeiro, "Network newton—Part II: Convergence rate and implementation," 2015, arXiv preprint arXiv:1504.06020.
[20] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.
[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, 2014.
[22] F. Iutzeler, P. Bianchi, P. Ciblat, and W. Hachem, "Explicit convergence rate of a distributed alternating direction method of multipliers," *IEEE Trans. Autom. Control*, vol. 61, no. 4, pp. 892–904, 2015.
[23] E. Ghadimi, A. Teixeira, I. Shames, and M. Johansson, "Optimal parameter selection for the alternating direction method of multipliers (ADMM): Quadratic problems," *IEEE Trans. Autom. Control*, vol. 60, no. 3, pp. 644–658, 2015.
[24] M. G. Rabbat *et al.*, "Generalized consensus computation in networked systems with erasure links," in *Proc. IEEE 6th Workshop Signal Process. Adv. Wireless Commun.*, 2005, pp. 1088–1092.
[25] D. Jakovetic, J. M. Moura, and J. Xavier, "Linear convergence rate of a class of distributed augmented Lagrangian algorithms," *IEEE Trans. Autom. Control*, vol. 60, no. 4, pp. 922–936, 2015.
[26] N. Chatzipanagiotis and M. M. Zavlanos, "On the convergence rate of a distributed augmented Lagrangian optimization algorithm," in *Proc. IEEE Amer. Control Conf. (ACC)*, 2015, pp. 541–546.
[27] T. Erseghe, D. Zennaro, E. Dall'Anese, and L. Vangelista, "Fast consensus by the alternating direction multipliers method," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5523–5537, 2011.
[28] A. Makhdoumi and A. Ozdaglar, "Convergence rate of distributed ADMM over networks," 2016, arXiv preprint arXiv:1601.00194.
[29] Q. Ling, and A. Ribeiro, "Decentralized linearized alternating direction method of multipliers," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2014, pp. 5447–5451.
[30] 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, 2015.
[31] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

Aryan Mokhtari received the B.Sc. degree in electrical engineering from Sharif University of Technology, Tehran, Iran, in 2011, and the M.Sc. degree in electrical engineering from the University of Pennsylvania (UPenn), Philadelphia, PA, in 2014. Since 2012, he has been working towards the Ph.D. degree in the Department of Electrical and Systems Engineering, UPenn. His research interests include optimization, signal processing, and machine learning. His current research focuses on stochastic optimization, online learning, and decentralized/distributed optimization.



Wei Shi received the B.E. degree in automation and Ph.D. degree in control science and engineering from the University of Science and Technology of China, Hefei, in 2010 and 2015, respectively. After his M.Sc. and Ph.D. studies, in 2015 he joined Coordinated Science Laboratory, the University of Illinois at Urbana-Champaign, Urbana, as a Postdoctoral Research Associate. His research interests are optimization and its applications in signal processing and control. Now he is specifically focused on decentralized/distributed/parallel optimization.

Qing Ling received the B.S. degree in automation and the Ph.D. degree in control theory and control engineering from University of Science and Technology of China, Hefei, Anhui, China, in 2001 and 2006, respectively. From 2006 to 2009, he was a Postdoctoral Research Fellow with Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, Michigan, USA. Since 2009, he has been an Associate Professor with Department of Automation, University of Science and Technology of China. His current research focuses on decentralized network optimization and its applications.

Alejandro Ribeiro received the B.Sc. degree in electrical engineering from the Universidad de la Republica Oriental del Uruguay, Montevideo, in 1998 and the M.Sc. and Ph.D. degree in electrical engineering from the Department of Electrical and Computer Engineering, the University of Minnesota, Minneapolis in 2005 and 2007. From 1998 to 2003, he was a member of the technical staff at Bellsouth Montevideo. After his M.Sc. and Ph.D. studies, in 2008 he joined the University of Pennsylvania (Penn), Philadelphia, where he is currently the Rosenbluth Associate Professor at the Department of Electrical and Systems Engineering. His research interests are in the applications of statistical signal processing to the study of networks and networked phenomena. His current research focuses on wireless networks, network optimization, learning in networks, networked control, robot teams, structured representations of networked data structures, and graph signal processing. Dr. Ribeiro received the 2014 O. Hugo Schuck best paper award, the 2012 S. Reid Warren, Jr. Award presented by Penn's undergraduate student body for outstanding teaching, the NSF CAREER Award in 2010, and student paper awards at the 2013 American Control Conference (as adviser), as well as the 2005 and 2006 International Conferences on Acoustics, Speech and Signal Processing. Dr. Ribeiro is a Fulbright scholar and a Penn Fellow.