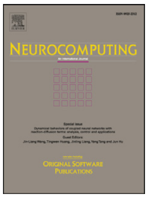
Neurocomputing 000 (2017) 1–8

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom



On the distributed optimization over directed networks[☆]

Chenguang Xi[a], Qiong Wu[b], Usman A. Khan[a,*]

[a] *Department of Electrical and Computer Engineering, Tufts University, 161 College Ave., Medford, MA 02155, USA*
[b] *Department of Mathematics, Tufts University, 503 Boston Ave., Medford, MA 02155, USA*

ARTICLE INFO

Article history:
Received 2 October 2016
Revised 3 March 2017
Accepted 13 June 2017
Available online xxx

Communicated by Choon Ki Ahn

Keywords:
Distributed optimization
Multi-agent networks
Directed graphs
Distributed subgradient descent

ABSTRACT

In this paper, we propose a distributed algorithm, called Directed-Distributed Subgradient Descent (D-DSD), to solve multi-agent optimization problems over *directed* graphs. Existing algorithms mostly deal with similar problems under the assumption of undirected networks, i.e., requiring the weight matrices to be doubly-stochastic. The row-stochasticity of the weight matrix guarantees that all agents reach consensus, while the column-stochasticity ensures that each agent's local (sub)gradient contributes equally to the global objective. In a directed graph, however, it may not be possible to construct a doubly-stochastic weight matrix in a distributed manner. We overcome this difficulty by augmenting an additional variable for each agent to record the change in the state evolution. In each iteration, the algorithm simultaneously constructs a row-stochastic matrix and a column-stochastic matrix instead of only a doubly-stochastic matrix. The convergence of the new weight matrix, depending on the row-stochastic and column-stochastic matrices, ensures agents to reach both consensus and optimality. The analysis shows that the proposed algorithm converges at a rate of $O(\frac{\ln k}{\sqrt{k}})$, where k is the number of iterations.

1. Introduction

Distributed computation and optimization has received significant recent interest in many areas, e.g., distributed machine learning, [2], distributed estimation, [33], cognitive networks, [13], source localization, [23], distributed coordination, [30], and message routing, [19]. The related problems can be posed as the minimization of a sum of objectives, $\sum_{i=1}^{n} f_i(\mathbf{x})$, where $f_i : \mathbb{R}^p \to \mathbb{R}$ is a private objective function at the ith agent. There are two general types of distributed algorithms to solve this problem. The first type is a (sub)gradient based method [4,6,9,12,17,18,24], where at each iteration a (sub)gradient related step is calculated, followed by averaging with neighbors in the network. The main advantage of these methods is computational simplicity. The (sub)gradient based methods are generalized to mirror descent methods [10,11,34] by using the Bregman divergence as distance-measuring function rather than the Euclidean distance. The second type of distributed algorithms are based on augmented Lagrangians, where at each iteration the primal variables are solved to minimize a Lagrangian related function, followed by updating the dual variables accordingly, e.g., the Distributed Alternating Direction Method of Multipliers (D-ADMM), [14,26,31]. The latter type is preferred when agents can solve the local optimization problem efficiently. Most proposed distributed algorithms, [4,6,9,11,12,14,17,18,24,26,31,34], assume undirected graphs. The primary reason behind assuming the undirected graphs is to obtain a doubly-stochastic weight matrix. The row-stochasticity of the weight matrix guarantees that all agents reach consensus, while the column-stochasticity ensures optimality.

In this paper, we propose a (sub)gradient based method solving distributed optimization problem over the *directed* graph, which we refer to as the Directed-Distributed Subgradient Descent (D-DSD). Clearly, a directed topology has broader applications in contrast to undirected graphs and may further result in reduced communication cost and simplified topology design. We start by explaining the necessity of weight matrices being doubly-stochastic in existing (sub)gradient based method, e.g., DSD. In the iteration of DSD, agents will not reach consensus if the row sum of the weight matrix is not equal to one. On the other hand, if the column of the weight matrix does not sum to one, each agent will contribute differently to the network. Since doubly-stochastic matrices may not be achievable in a directed graph, the original methods, e.g., DSD, no longer work. We overcome this difficulty in a directed graph by augmenting an additional variable for each agent to record the state updates. In each iteration of the D-DSD algorithm, we simultaneously construct a row-stochastic matrix and a column-stochastic matrix instead of only a doubly-stochastic

[☆] This work has been partially supported by an NSF Award # CCF-1350264.

[*] Corresponding author.
E-mail addresses: chenguang.xi@tufts.edu (C. Xi), qiong.wu@tufts.edu (Q. Wu), khan@ece.tufts.edu (U.A. Khan).

http://dx.doi.org/10.1016/j.neucom.2017.06.038
0925-2312/© 2017 Elsevier B.V. All rights reserved.

matrix. We give an intuitive explanation of our proposed algorithm and further provide convergence and convergence rate analysis as well.

In the context of directed graphs, related work has considered (sub)gradient based algorithms, [15,16,27–29], by combining subgradient descent and push-sum consensus. The push-sum algorithm, [1,7], is first proposed in consensus problems[1] to achieve average-consensus given a column-stochastic matrix. The idea is based on computing the stationary distribution (the left eigenvector of the weight matrix corresponding to eigenvalue 1) for the Markov chain characterized by the multi-agent network and canceling the imbalance by dividing with the left-eigenvector. The algorithms in [15,16,27–29] follow a similar spirit of push-sum consensus and propose nonlinear (because of division) methods. In contrast, our algorithm follows linear iterations and does not involve any division while providing the same convergence rate as the nonlinear one in e.g., [16]. Finally, the analysis and proofs in our work are completely different than the nonlinear counterparts described here.

The remainder of the paper is organized as follows. In Section 2, we provide the problem formulation and show the reason why DSD fails to converge to the optimal solution over directed graphs. We subsequently present the D-DSD algorithm and the necessary assumptions. The convergence analysis of the D-DSD algorithm is studied in Section 3, consisting of agents' consensus analysis and optimality analysis. The convergence rate analysis and numerical experiments are presented in Sections 4 and 5. Section 6 contains concluding remarks.

Notation: Lowercase bold letters denote vectors and uppercase italic letters denote matrices. We denote by $[\mathbf{x}]_i$ the ith component of a vector $\mathbf{x}$, and by $[A]_{ij}$ the (i, j)th element of a matrix, A. An n-dimensional vector of all ones or zeros is represented by $\mathbf{1}_n$ or $\mathbf{0}_n$. The notation $0_{n\times n}$ represents an $n \times n$ matrix with all elements equal to zero. The inner product of two vectors $\mathbf{x}$ and $\mathbf{y}$ is $\langle \mathbf{x}, \mathbf{y}\rangle$. We use $\|\mathbf{x}\|$ to denote the standard Euclidean norm.

2. Problem formulation

Consider a strongly-connected network of n agents communicating over a directed graph, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of agents, and $\mathcal{E}$ is the collection of ordered pairs, $(i, j), i, j \in \mathcal{V}$, such that agent j can send information to agent i. Define $\mathcal{N}_i^{\text{in}}$ to be the collection of in-neighbors, i.e., the set of agents that can send information to agent i. Similarly, $\mathcal{N}_i^{\text{out}}$ is defined as the out-neighborhood of agent i, i.e., the set of agents that can receive information from agent i. We allow both $\mathcal{N}_i^{\text{in}}$ and $\mathcal{N}_i^{\text{out}}$ to include the node i itself. Note that in a directed graph $\mathcal{N}_i^{\text{in}} \neq \mathcal{N}_i^{\text{out}}$, in general. We focus on solving a convex optimization problem that is distributed over the above network. In particular, the network of agents cooperatively solve the following optimization problem:

$$\text{P1}: \quad \min f(\mathbf{x}) = \sum_{i=1}^n f_i(\mathbf{x}),$$

where each $f_i : \mathbb{R}^p \to \mathbb{R}$ is convex, not necessarily differentiable, representing the local objective function at agent i.

Assumption 1. In order to solve the above problem, we make the following assumptions:

(a) The agent graph, $\mathcal{G}$, is strongly-connected.
(b) Each local function, $f_i : \mathbb{R}^p \to \mathbb{R}$, is convex, $\forall i \in \mathcal{V}$.
(c) The solution set of Problem P1 and the corresponding optimal value exist. Formally, we have

$$\mathbf{x}^* \in \mathcal{X}^* = \left\{\mathbf{x} \middle| f(\mathbf{x}) = \min_{\mathbf{y}\in\mathbb{R}^p} f(\mathbf{y})\right\}, f^* = \min f(\mathbf{x}).$$

(d) The (sub)gradient, $\nabla f_i(\mathbf{x})$, is bounded:

$$\|\nabla f_i(\mathbf{x})\| \leq D,$$

for all $\mathbf{x} \in \mathbb{R}^p, i \in \mathcal{V}$.

The Assumptions 1 are standard in distributed optimization, see related literature, [18], and references therein. Before describing our algorithm, we first recap the DSD algorithm, [17], to solve P1 in an undirected graph. This algorithm requires doubly-stochastic weight matrices. We analyze the influence to the result of the DSD when the weight matrices are *not* doubly-stochastic.

2.1. Distributed subgradient descent

Consider Distributed Subgradient Descent (DSD), [17], to solve P1. Agent i updates its estimate as follows:

$$\mathbf{x}_i^{k+1} = \sum_{j=1}^n w_{ij}\mathbf{x}_j^k - \alpha_k \nabla f_i^k, \tag{1}$$

where w_{ij} is a non-negative weight such that $W = \{w_{ij}\}$ is doubly-stochastic. The scalar, α_k, is a diminishing but non-negative step-size, satisfying the persistence conditions, [8,12]: $\sum_{k=0}^\infty \alpha_k = \infty$, $\sum_{k=0}^\infty \alpha_k^2 < \infty$, and the vector ∇f_i^k is a (sub)gradient of f_i at $\mathbf{x}_i^k$. For the sake of argument, consider W to be row-stochastic but not column-stochastic. Clearly, $\mathbf{1}$ is a right eigenvector of W, and let $\boldsymbol{\pi} = \{\pi_i\}$ be its left eigenvector corresponding to eigenvalue 1. Summing over i in Eq. (1), we get

$$\begin{aligned}\widehat{\mathbf{x}}^{k+1} &\triangleq \sum_{i=1}^n \pi_i \mathbf{x}_i^{k+1},\\ &= \sum_{j=1}^n\left(\sum_{i=1}^n \pi_i w_{ij}\right)\mathbf{x}_j^k - \alpha_k\sum_{i=1}^n \pi_i \nabla f_i(\mathbf{x}_i^k),\\ &= \widehat{\mathbf{x}}^k - \alpha_k \sum_{i=1}^n \pi_i \nabla f_i^k, \end{aligned} \tag{2}$$

where $\pi_j = \sum_{i=1}^n \pi_i w_{ij}, \forall i, j$. If we assume that the agents reach an agreement, then Eq. (2) can be viewed as an inexact (central) subgradient descent (with $\sum_{i=1}^n \pi_i \nabla f_i(\mathbf{x}_i^k)$ instead of $\sum_{i=1}^n \pi_i \nabla f_i(\widehat{\mathbf{x}}^k)$) minimizing a new objective, $\widehat{f}(\mathbf{x}) \triangleq \sum_{i=1}^n \pi_i f_i(\mathbf{x})$. As a result, the agents reach consensus and converge to the minimizer of $\widehat{f}(\mathbf{x})$.

Now consider the weight matrix, W, to be column-stochastic but not row-stochastic. Let $\overline{\mathbf{x}}^k$ be the average of agents estimates at time k, then Eq. (1) leads to

$$\begin{aligned}\overline{\mathbf{x}}^{k+1} &\triangleq \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i^{k+1},\\ &= \frac{1}{n}\sum_{j=1}^n\left(\sum_{i=1}^n w_{ij}\right)\mathbf{x}_j^k - \frac{\alpha_k}{n}\sum_{i=1}^n \nabla f_i(\mathbf{x}_i^k),\\ &= \overline{\mathbf{x}}^k - \left(\frac{\alpha_k}{n}\right)\sum_{i=1}^n \nabla f_i^k. \end{aligned} \tag{3}$$

Eq. (3) reveals that the average, $\overline{\mathbf{x}}^k$, of agents estimates follows an inexact (central) subgradient descent ($\sum_{i=1}^n \nabla f_i(\mathbf{x}_i^k)$ instead of $\sum_{i=1}^n \nabla f_i(\overline{\mathbf{x}}^k)$) with stepsize α^k/n, thus reaching the minimizer of $f(\mathbf{x})$. Despite the fact that the average, $\overline{\mathbf{x}}^k$, reaches the optima, $\mathbf{x}^*$, of $f(\mathbf{x})$, the optima is not achievable for each agent because consensus can not be reached with a matrix that is not necessary row-stochastic.

[1] See, [5,20–22,25,32], for additional information on average-consensus problems.

Eqs. (2) and (3) explain the importance of doubly-stochastic matrices in consensus-based optimization. The row-stochasticity guarantees all of the agents to reach a consensus, while column-stochasticity ensures each local (sub)gradient to contribute equally to the global objective.

2.2. Directed-distributed subgradient descent (D-DSD)

From the above discussion, we note that reaching a consensus requires the right eigenvector (corresponding to eigenvalue 1) to lie in span$\{\mathbf{1}_n\}$, and minimizing the global objective requires the corresponding left eigenvector to lie in span$\{\mathbf{1}_n\}$. Both the left and right eigenvectors of a doubly-stochastic matrix are $\mathbf{1}_n$, which, in general, is not possible in directed graphs. In this paper, we introduce *Directed-Distributed Subgradient Descent* (D-DSD) that overcomes the above issues by augmenting an additional variable at each agent and thus constructing a new weight matrix, $M\in\mathbb{R}^{2n\times 2n}$, whose left and right eigenvectors (corresponding to eigenvalue 1) are in the form: $[\mathbf{1}_n^\top,\mathbf{v}^\top]$ and $[\mathbf{1}_n^\top,\mathbf{u}^\top]^\top$. Formally, we describe D-DSD as follows.

At kth iteration, each agent, $j\in\mathcal{V}$, maintains two vectors: $\mathbf{x}_j^k$ and $\mathbf{y}_j^k$, both in $\mathbb{R}^p$. Agent j sends its state estimate, $\mathbf{x}_j^k$, as well as a weighted auxiliary variable, $b_{ij}\mathbf{y}_j^k$, to each out-neighbor, $i\in\mathcal{N}_j^{\text{out}}$, where b_{ij}'s are such that:

$$b_{ij}=\begin{cases} >0, & i\in\mathcal{N}_j^{\text{out}}, \\ 0, & \text{otw.}, \end{cases} \qquad \sum_{i=1}^{n} b_{ij}=1, \forall j.$$

Agent i updates the variables, $\mathbf{x}_i^{k+1}$ and $\mathbf{y}_i^{k+1}$, with the information received from its in-neighbors, $j\in\mathcal{N}_i^{\text{in}}$, as follows:

$$\mathbf{x}_i^{k+1}=\sum_{j=1}^{n} a_{ij}\mathbf{x}_j^k+\epsilon\mathbf{y}_i^k-\alpha_k\nabla f_i(\mathbf{x}_i^k), \tag{4a}$$

$$\mathbf{y}_i^{k+1}=\mathbf{x}_i^k-\sum_{j=1}^{n} a_{ij}\mathbf{x}_j^k+\sum_{j=1}^{n} b_{ij}\mathbf{y}_j^k-\epsilon\mathbf{y}_i^k, \tag{4b}$$

where:

$$a_{ij}=\begin{cases} >0, & j\in\mathcal{N}_i^{\text{in}}, \\ 0, & \text{otw.}, \end{cases} \qquad \sum_{j=1}^{n} a_{ij}=1, \forall i.$$

The diminishing step-size, $\alpha_k\geq 0$, satisfies the persistence conditions, [8,12]: $\sum_{k=0}^{\infty}\alpha_k=\infty$, $\sum_{k=0}^{\infty}\alpha_k^2<\infty$. The scalar, ϵ, is a small positive number, which plays a key role in the convergence of the algorithm[2]. For an illustration of the message passing between agents in the implementation of Eq. (4), see Fig. 1 on how agent i sends information to its out-neighbors and agent l receives information from its in-neighbors. In Fig. 1, the weights $b_{j_1 i}$ and $b_{j_2 i}$ are designed by agent i, and satisfy $b_{ii}+b_{j_1 i}+b_{j_2 i}=1$. To analyze the algorithm, we denote $\mathbf{z}_i^k\in\mathbb{R}^p$, $\mathbf{g}_i^k\in\mathbb{R}^p$, and $M\in\mathbb{R}^{2n\times 2n}$ as follows:

$$\mathbf{z}_i^k=\begin{cases} \mathbf{x}_i^k, & i\in\{1,\ldots,n\}, \\ \mathbf{y}_{i-n}^k, & i\in\{n+1,\ldots,2n\}, \end{cases}$$

$$\mathbf{g}_i^k=\begin{cases} \nabla f_i(\mathbf{x}_i^k), & i\in\{1,\ldots,n\}, \\ \mathbf{0}_p, & i\in\{n+1,\ldots,2n\}, \end{cases}$$

$$M=\begin{bmatrix} A & \epsilon I \\ I-A & B-\epsilon I \end{bmatrix}, \tag{5}$$

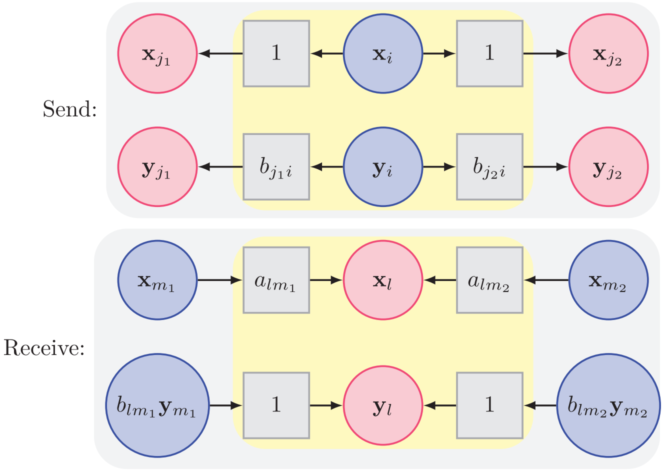


Fig. 1. Illustration of the message passing between agents by Eq. (4).

where $A=\{a_{ij}\}$ is row-stochastic, $B=\{b_{ij}\}$ is column-stochastic. Consequently, Eq. (4) can be represented compactly as follows: for any $i\in\{1,\ldots,2n\}$, at $k+1$th iteration,

$$\mathbf{z}_i^{k+1}=\sum_{j=1}^{2n}[M]_{ij}\mathbf{z}_j^k-\alpha_k\mathbf{g}_i^k. \tag{6}$$

We refer to the iterative relation in Eq. (6) as the Directed-Distributed Subgradient Descent (D-DSD) method, since it has the same form as DSD except the dimension doubles due to a new weight matrix $M\in\mathbb{R}^{2n\times 2n}$ as defined in Eq. (5). It is worth mentioning that even though Eq. (6) looks similar to DSD, [17], the convergence analysis of D-DSD does not exactly follow that of DSD. This is because the weight matrix, M, has negative entries. Besides, M is not a doubly-stochastic matrix, i.e., the row sum is not 1. Hence, the tools in the analysis of DSD are not applicable, e.g., $\|\sum_j[M]_{ij}\mathbf{z}_j-\mathbf{x}^*\|\leq\sum_j[M]_{ij}\|\mathbf{z}_j-\mathbf{x}^*\|$ does not necessarily hold because $[M]_{ij}$ are not non-negative. In next section, we prove the convergence of D-DSD.

3. Convergence analysis

The convergence analysis of D-DSD can be divided into two parts. In the first part, we discuss the *consensus property* of D-DSD, i.e., we capture the decrease in $\|\mathbf{z}_i^k-\overline{\mathbf{z}}^k\|$ for $i\in\{1,\ldots,n\}$, as the D-DSD progresses, where we define $\overline{\mathbf{z}}^k$ as the accumulation point:

$$\overline{\mathbf{z}}^k\triangleq\frac{1}{n}\sum_{j=1}^{2n}\mathbf{z}_i^k=\frac{1}{n}\sum_{j=1}^{n}\mathbf{x}_i^k+\frac{1}{n}\sum_{j=1}^{n}\mathbf{y}_i^k. \tag{7}$$

The decrease in $\|\mathbf{z}_i^k-\overline{\mathbf{z}}^k\|$ reveals that all agents approach a common accumulation point. We then show the *optimality property* in the second part, i.e., the decrease in the difference between the function evaluated at the accumulation point and the optimal solution, $f(\overline{\mathbf{z}}^k)-f(\mathbf{x}^*)$. We combine the two parts to establish the convergence.

3.1. Consensus property

To show the consensus property, we study the convergence behavior of the weight matrices, M^k, in Eq. (5) as k goes to infinity. We use an existing results on such matrices M, based on which we show the convergence behavior as well as the convergence rate. We borrow the following from [3].

Lemma 1. *(Cai et al.[3]) Assume the graph is strongly-connected. M is the weighting matrix defined in Eq. (5), and the constant ϵ in M*

[2] Note that in the implementation of Eq. (4), each agent needs the knowledge of its out-neighbors. In a more restricted setting, e.g., a broadcast application where it may not be possible to know the out-neighbors, we may use $b_{ij}=|\mathcal{N}_j^{\text{out}}|^{-1}$; thus, the implementation only requires knowing the out-degrees, see, e.g., [15,16] for similar assumptions.

satisfies $\epsilon \in (0, \Upsilon)$, *where* $\Upsilon := \frac{1}{(20+8n)^n}(1-|\lambda_3|)^n$, *where* λ_3 *is the third largest eigenvalue of* M *in Eq. (5) by setting* $\epsilon = 0$. *Then the weighting matrix, M, defined in Eq. (5), has a simple eigenvalue 1 and all other eigenvalues have magnitude smaller than one.*

Based on Lemma 1, we now provide the convergence behavior as well as the convergence rate of the weight matrix, M.

Lemma 2. *Assume that the network is strongly-connected, and M is the weight matrix that defined in Eq. (5).Then,*

(a) *The sequence of* $\{M^k\}$, *as k goes to infinity, converges to the following limit:*

$$\lim_{k\to\infty} M^k = \begin{bmatrix} \frac{\mathbf{1}_n\mathbf{1}_n^\top}{n} & \frac{\mathbf{1}_n\mathbf{1}_n^\top}{n} \\ 0 & 0 \end{bmatrix};$$

(b) *For all* $i, j \in \mathcal{V}$, *the entries* $[M^k]_{ij}$ *converge to their limits as* $k \to \infty$ *at a geometric rate, i.e., there exist bounded constants,* $\Gamma \in \mathbb{R}$, *and* $0 < \gamma < 1$, *such that*

$$\left\| M^k - \begin{bmatrix} \frac{\mathbf{1}_n\mathbf{1}_n^\top}{n} & \frac{\mathbf{1}_n\mathbf{1}_n^\top}{n} \\ 0 & 0 \end{bmatrix} \right\|_\infty \le \Gamma\gamma^k.$$

Proof. Note that the sum of each column of M equals one, so 1 is an eigenvalue of M with a corresponding left (row) eigenvector $[\mathbf{1}_n^\top \ \mathbf{1}_n^\top]$. We further have $M[\mathbf{1}_n^\top \ \mathbf{0}_n^\top]^\top = [\mathbf{1}_n^\top \ \mathbf{0}_n^\top]^\top$, so $[\mathbf{1}_n^\top \ \mathbf{0}_n^\top]^\top$ is a right (column) eigenvector corresponding to the eigenvalue 1. According to Lemma 1, 1 is a simple eigenvalue of M and all other eigenvalues have magnitude smaller than one. We represent M^k in the Jordan canonical form for some P_i and Q_i

$$M^k = \frac{1}{n}[\mathbf{1}_n^\top \ \mathbf{0}_n^\top]^\top[\mathbf{1}_n^\top \ \mathbf{1}_n^\top] + \sum_{i=2}^{n} P_i J_i^k Q_i, \tag{8}$$

where the diagonal entries in J_i are smaller than one in magnitude for all i. The statement (a) follows by noting that $\lim_{k\to\infty} J_i^k = 0$, for all i.

From Eq. (8), and with the fact that all eigenvalues of M except 1 have magnitude smaller than one, there exist some bounded constants, Γ and $\gamma \in (0, 1)$, such that

$$\begin{aligned} \left\| M^k - \begin{bmatrix} \frac{\mathbf{1}_n\mathbf{1}_n^\top}{n} & \frac{\mathbf{1}_n\mathbf{1}_n^\top}{n} \\ 0 & 0 \end{bmatrix} \right\| &= \left\| \sum_{i=2}^{n} P_i J_i^k Q_i \right\|, \\ &\le \sum_{i=2}^{n} \|P_i\| \|Q_i\| \left\|J_i^k\right\| \le \Gamma\gamma^k, \end{aligned}$$

from which we get the desired result. □

Using the result from Lemma 1, Lemma 2 shows the convergence behavior of the power of the weight matrix, and further show that its convergence is bounded by a geometric rate. Lemma 2 plays a key role in proving the consensus properties of D-DSD. Based on Lemma 2, we bound the difference between agent estimates in the following lemma. More specifically, we show that the agent estimates, $\mathbf{x}_i^k$, approaches the accumulation point, $\bar{\mathbf{z}}^k$, and the auxiliary variable, $\mathbf{y}_i^k$, goes to $\mathbf{0}_n$, where $\bar{\mathbf{z}}^k$ is defined in Eq. (7).

Lemma 3. *Let the Assumptions A1 hold. Let* $\{\mathbf{z}_i^k\}$ *be the sequence over k generated by the D-DSD algorithm, Eq. (6). Then, there exist some bounded constants,* Γ *and* $0 < \gamma < 1$, *such that:*

(a) *for* $1 \le i \le n$, *and* $k \ge 1$,

$$\begin{aligned} \left\| \mathbf{z}_i^k - \bar{\mathbf{z}}^k \right\| \le & \Gamma\gamma^k \sum_{j=1}^{2n} \left\| \mathbf{z}_j^0 \right\| + n\Gamma D \sum_{r=1}^{k-1} \gamma^{k-r}\alpha_{r-1} \\ & + 2D\alpha_{k-1}; \end{aligned}$$

(b) *for* $n+1 \le i \le 2n$, *and* $k \ge 1$,

$$\left\| \mathbf{z}_i^k \right\| \le \Gamma\gamma^k \sum_{j=1}^{2n} \left\| \mathbf{z}_j^0 \right\| + n\Gamma D \sum_{r=1}^{k-1} \gamma^{k-r}\alpha_{r-1}.$$

Proof. For any $k \ge 1$, we write Eq. (6) recursively

$$\begin{aligned} \mathbf{z}_i^k =& \sum_{j=1}^{2n} [M^k]_{ij}\mathbf{z}_j^0 - \sum_{r=1}^{k-1}\sum_{j=1}^{2n} [M^{k-r}]_{ij}\alpha_{r-1}\mathbf{g}_j^{r-1} \\ & - \alpha_{k-1}\mathbf{g}_i^{k-1}. \end{aligned} \tag{9}$$

Since every column of M sums up to one, we have for any r $\sum_{i=1}^{2n}[M^r]_{ij} = 1$. Considering the recursive relation of $\mathbf{z}_i^k$ in Eq. (9), we obtain that $\bar{\mathbf{z}}^k$ can be represented as

$$\bar{\mathbf{z}}^k = \sum_{j=1}^{2n}\frac{1}{n}\mathbf{z}_j^0 - \sum_{r=1}^{k-1}\sum_{j=1}^{2n}\frac{1}{n}\alpha_{r-1}\mathbf{g}_j^{r-1} - \frac{1}{n}\sum_{j=1}^{2n}\alpha_{k-1}\mathbf{g}_j^{k-1}. \tag{10}$$

Subtracting Eq. (10) from (9) and taking the norm, we obtain that for $1 \le i \le n$,

$$\begin{aligned} \left\| \mathbf{z}_i^k - \bar{\mathbf{z}}^k \right\| \le & \sum_{j=1}^{2n} \left\| [M^k]_{ij} - \frac{1}{n} \right\| \left\| \mathbf{z}_j^0 \right\| \\ & + \sum_{r=1}^{k-1}\sum_{j=1}^{n} \left\| [M^{k-r}]_{ij} - \frac{1}{n} \right\| \alpha_{r-1} \left\| \nabla f_j(\mathbf{x}_j^{r-1}) \right\| \\ & + \alpha_{k-1} \left\| \nabla f_i(\mathbf{x}_i^{k-1}) \right\| + \frac{1}{n}\sum_{j=1}^{n}\alpha_{k-1}\left\| \nabla f_j(\mathbf{x}_j^{k-1}) \right\|. \end{aligned} \tag{11}$$

The proof of part (a) follows by applying the result of Lemma 2 to Eq. (11) and noticing that the (sub)gradient is bounded by a constant D. Similarly, by taking the norm of Eq. (9), we obtain that for $n+1 \le i \le 2n$,

$$\begin{aligned} \left\| \mathbf{z}_i^k \right\| \le & \sum_{j=1}^{2n} \left\| [M^k]_{ij} \right\| \left\| \mathbf{z}_j^0 \right\| \\ & + \sum_{r=1}^{k-1}\sum_{j=1}^{n} \left\| [M^{k-r}]_{ij} \right\| \alpha_{r-1} \left\| \nabla f_j(\mathbf{x}_j^{r-1}) \right\|. \end{aligned}$$

The proof of part (b) follows by applying the result of Lemma 2 to the preceding relation and considering the boundedness of (sub)gradient in Assumption 1(e). □

Using Lemma 3, we now draw our first conclusion on the consensus property at the agents. Proposition 1 reveals that all agents asymptotically reach consensus.

Proposition 1. *Let the Assumptions A1 hold. Let* $\{\mathbf{z}_i^k\}$ *be the sequence over k generated by the D-DSD algorithm, Eq. (6). Then,* $\mathbf{z}_i^k$ *satisfies*

(a) *for* $1 \le i \le n$,

$$\sum_{k=1}^{\infty} \alpha_k \left\| \mathbf{z}_i^k - \bar{\mathbf{z}}^k \right\| < \infty;$$

(b) *for* $n+1 \le i \le 2n$,

$$\sum_{k=1}^{\infty} \alpha_k \left\| \mathbf{z}_i^k \right\| < \infty.$$

Proof. Based on the result of Lemma 3(a), we obtain, for $1 \le i \le n$,

$$\sum_{k=1}^{K} \alpha_k \left\| \mathbf{z}_i^k - \bar{\mathbf{z}}^k \right\| \le \Gamma\left(\sum_{j=1}^{2n}\left\|\mathbf{z}_j^0\right\|\right)\sum_{k=1}^{K}\alpha_k\gamma^k$$

$$+ n\Gamma D\sum_{k=1}^{K}\sum_{r=1}^{k-1}\gamma^{(k-r)}\alpha_k\alpha_{r-1} + 2D\sum_{k=0}^{K-1}\alpha_k^2. \tag{12}$$

With the basic inequality $ab \le \frac{1}{2}(a^2+b^2)$, $a, b \in \mathbb{R}$, we have:

$$2\sum_{k=1}^{K}\alpha_k\gamma^k \le \sum_{k=1}^{K}\left[\alpha_k^2+\gamma^{2k}\right] \le \sum_{k=1}^{K}\alpha_k^2 + \frac{1}{1-\gamma^2};$$

and

$$\begin{aligned}\sum_{k=1}^{K}\sum_{r=1}^{k-1}\gamma^{(k-r)}\alpha_k\alpha_{r-1} &\le \frac{1}{2}\sum_{k=1}^{K}\alpha_k^2\sum_{r=1}^{k-1}\gamma^{(k-r)}\\ &+ \frac{1}{2}\sum_{r=1}^{K-1}(\alpha_{r-1})^2\sum_{k=r+1}^{K}\gamma^{(k-r)} \le \frac{1}{1-\gamma}\sum_{k=1}^{K}\alpha_k^2.\end{aligned}$$

The proof of part (a) follows by applying the preceding relations to Eq. (12) along with $\sum_{k=0}^{K}\alpha_k^2 < \infty$ as $K \to \infty$. Following the same spirit in the proof of part (b), we can reach the conclusion of part (b). □

Since $\sum_{k=1}^{\infty}\alpha_k = \infty$, Proposition 1 shows that all agents reach consensus at the accumulation point, $\overline{\mathbf{z}}^k$, asymptotically, i.e., for all $1 \le i \le n$, $1 \le j \le n$,

$$\lim_{k\to\infty}\mathbf{z}_i^k = \lim_{k\to\infty}\overline{\mathbf{z}}^k = \lim_{k\to\infty}\mathbf{z}_j^k, \tag{13}$$

and for $n+1 \le i \le 2n$, the states, $\mathbf{z}_i^k$, asymptotically, converge to zero, i.e., for $n+1 \le i \le 2n$,

$$\lim_{k\to\infty}\mathbf{z}_i^k = 0. \tag{14}$$

We next show how the accumulation point, $\overline{\mathbf{z}}^k$, approaches the optima, $\mathbf{x}^*$, as D-DSD progresses.

3.2. Optimality property

The following lemma gives an upper bound on the difference between the objective evaluated at the accumulation point, $f(\overline{\mathbf{z}}^k)$, and the optimal objective value, f^*.

Lemma 4. *Let the Assumptions A1 hold. Let* $\left\{\mathbf{z}_i^k\right\}$ *be the sequence over k generated by the D-DSD algorithm, Eq. (6). Then,*

$$\begin{aligned}2\sum_{k=0}^{\infty}\alpha_k\left(f(\overline{\mathbf{z}}^k) - f^*\right) &\le n\left\|\overline{\mathbf{z}}^0 - \mathbf{x}^*\right\|^2 + nD^2\sum_{k=0}^{\infty}\alpha_k^2\\ &+ \frac{4D}{n}\sum_{i=1}^{n}\sum_{k=0}^{\infty}\alpha_k\left\|\mathbf{z}_i^k - \overline{\mathbf{z}}^k\right\|.\end{aligned} \tag{15}$$

Proof. Consider Eq. (6) and the fact that each column of M sums to one, we have

$$\begin{aligned}\overline{\mathbf{z}}^{k+1} &= \frac{1}{n}\sum_{j=1}^{2n}\left[\sum_{i=1}^{2n}[M]_{ij}\right]\mathbf{z}_j^k - \alpha_k\frac{1}{n}\sum_{i=1}^{2n}\mathbf{g}_i^k,\\ &= \overline{\mathbf{z}}^k - \frac{\alpha_k}{n}\sum_{i=1}^{n}\nabla f_i(\mathbf{z}_i^k).\end{aligned}$$

Therefore, we obtain that

$$\begin{aligned}\left\|\overline{\mathbf{z}}^{k+1} - \mathbf{x}^*\right\|^2 &= \left\|\overline{\mathbf{z}}^k - \mathbf{x}^*\right\|^2 + \left\|\frac{\alpha_k}{n}\sum_{i=1}^{n}\nabla f_i(\mathbf{z}_i^k)\right\|^2\\ &- 2\frac{\alpha_k}{n}\sum_{i=1}^{n}\left\langle\overline{\mathbf{z}}^k - \mathbf{x}^*, \nabla f_i(\mathbf{z}_i^k)\right\rangle.\end{aligned} \tag{16}$$

Denote $\nabla f_i^k = \nabla f_i(\mathbf{z}_i^k)$. Since $\|\nabla f_i^k\| \le D$, we have

$$\begin{aligned}\left\langle\overline{\mathbf{z}}^k - \mathbf{x}^*, \nabla f_i^k\right\rangle &= \left\langle\overline{\mathbf{z}}^k - \mathbf{z}_i^k, \nabla f_i^k\right\rangle + \left\langle\mathbf{z}_i^k - \mathbf{x}^*, \nabla f_i^k\right\rangle\\ &\ge \left\langle\overline{\mathbf{z}}^k - \mathbf{z}_i^k, \nabla f_i^k\right\rangle + f_i(\mathbf{z}_i^k) - f_i(\mathbf{x}^*)\\ &\ge -D\left\|\mathbf{z}_i^k - \overline{\mathbf{z}}^k\right\| + f_i(\mathbf{z}_i^k) - f_i(\overline{\mathbf{z}}^k) + f_i(\overline{\mathbf{z}}^k) - f_i(\mathbf{x}^*)\\ &\ge -2D\left\|\mathbf{z}_i^k - \overline{\mathbf{z}}^k\right\| + f_i(\overline{\mathbf{z}}^k) - f_i(\mathbf{x}^*).\end{aligned} \tag{17}$$

By substituting Eq. (17) in Eq. (16), and rearranging the terms, we obtain that

$$\begin{aligned}2\alpha_k\left(f(\overline{\mathbf{z}}^k) - f^*\right) &\le n\left\|\overline{\mathbf{z}}^k - \mathbf{x}^*\right\|^2 - n\left\|\overline{\mathbf{z}}^{k+1} - \mathbf{x}^*\right\|^2\\ &+ nD^2\alpha_k^2 + \frac{4D}{n}\sum_{i=1}^{n}\alpha_k\left\|\mathbf{z}_i^k - \overline{\mathbf{z}}^k\right\|.\end{aligned} \tag{18}$$

The desired result is achieved by summing Eq. (18) over time from $k = 0$ to ∞. □

We are ready to present the main result of this paper, by combining all the preceding results.

Theorem 1. *Let the Assumptions A1 hold. Let* $\left\{\mathbf{z}_i^k\right\}$ *be the sequence over k generated by the D-DSD algorithm, Eq. (6). Then, for any agent i, we have*

$$\lim_{k\to\infty} f(\mathbf{z}_i^k) = f^*.$$

Proof. Since that the step-size follows that $\sum_{k=0}^{\infty}\alpha_k^2 < \infty$, and $\sum_{k=0}^{\infty}\alpha_k\|\mathbf{z}_i^k - \overline{\mathbf{z}}^k\| < \infty$ from Lemma 1, we obtain from Eq. (15) that

$$2\sum_{k=0}^{\infty}\alpha_k\left(f(\overline{\mathbf{z}}^k) - f^*\right) < \infty, \tag{19}$$

which reveals that $\lim_{k\to\infty} f(\overline{\mathbf{z}}^k) = f^*$, by considering that $\sum_{k=0}^{\infty}\alpha_k = \infty$. In Eq. (13), we have already shown that $\lim_{k\to\infty}\mathbf{z}_i^k = \lim_{k\to\infty}\overline{\mathbf{z}}^k$. Therefore, we obtain the desired result. □

4. Convergence rate

In this section, we show the convergence rate of D-DSD. Let $f_m := \min_k f(\overline{\mathbf{z}}^k)$, we have

$$(f_m - f^*)\sum_{k=0}^{K}\alpha_k \le \sum_{k=0}^{K}\alpha_k(f(\overline{\mathbf{z}}^k) - f^*) \tag{20}$$

By combining Eqs. (12), (15) and (20), it can be verified that Eq. (15) can be represented in the following form:

$$(f_m - f^*)\sum_{k=0}^{K}\alpha_k \le C_1 + C_2\sum_{k=0}^{K}\alpha_k^2,$$

or equivalently,

$$(f_m - f^*) \le \frac{C_1}{\sum_{k=0}^{K}\alpha_k} + \frac{C_2\sum_{k=0}^{K}\alpha_k^2}{\sum_{k=0}^{K}\alpha_k}, \tag{21}$$

where the constants, C_1 and C_2, are given by

$$\begin{aligned}C_1 =& \frac{n}{2}\left\|\overline{\mathbf{z}}^0 - \mathbf{x}^*\right\|^2 - \frac{n}{2}\left\|\overline{\mathbf{z}}^{K+1} - \mathbf{x}^*\right\|^2\\ &+ D\Gamma\sum_{j=1}^{2n}\left\|\mathbf{z}_j^0\right\|\frac{1}{1-\gamma^2},\\ C_2 =& \frac{nD^2}{2} + 4D^2 + D\Gamma\sum_{j=1}^{2n}\left\|\mathbf{z}_j^0\right\| + \frac{2D^2\Gamma}{1-\gamma}.\end{aligned}$$

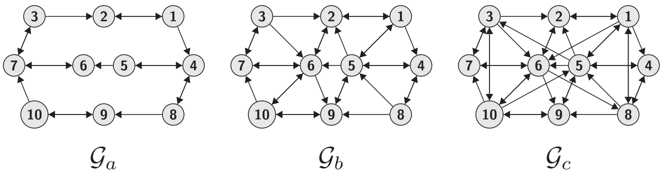


Fig. 2. Strongly-connected but non-balanced digraphs.

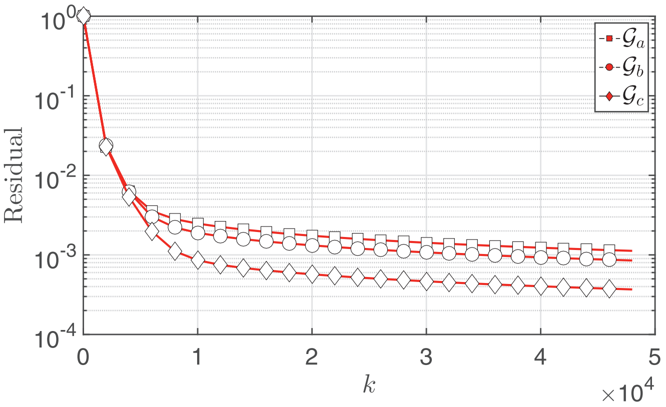


Fig. 3. Plot of residuals $\frac{\|\mathbf{x}_k-\mathbf{x}^*\|_F}{\|\mathbf{x}_0-\mathbf{x}^*\|_F}$ for digraph $\mathcal{G}_a, \mathcal{G}_b, \mathcal{G}_c$ as D-DSD progresses.

Eq. (21) actually has the same form as the equations in analyzing the convergence rate of DSD (recall, e.g., [17]). In particular, when $\alpha_k = k^{-1/2}$, the first term in Eq. (21) leads to

$$\frac{C_1}{\sum_{k=0}^{K}\alpha_k} = C_1\frac{1/2}{K^{1/2}-1} = O\left(\frac{1}{\sqrt{K}}\right),$$

while the second term in Eq. (21) leads to

$$\frac{C_2\sum_{k=0}^{K}\alpha_k^2}{\sum_{k=0}^{K}\alpha_k} = C_2\frac{\ln K}{2(\sqrt{K}-1)} = O\left(\frac{\ln K}{\sqrt{K}}\right).$$

It can be observed that the second term dominates, and the overall convergence rate is $O\left(\frac{\ln k}{\sqrt{k}}\right)$. As a result, D-DSD has the same convergence rate as DSD. The restriction of directed graph does not effect the speed.

5. Numerical experiment

5.1. Distributed least square problem

We first consider a least squares problem on a directed graph: each agent owns a private objective function, $\mathbf{s}_i = R_i\mathbf{x} + \mathbf{n}_i$, where $\mathbf{s}_i \in \mathbb{R}^{m_i}$ and $R_i \in \mathbb{R}^{m_i \times p}$ are measured data, $\mathbf{x} \in \mathbb{R}^p$ is unknown states, and $\mathbf{n}_i \in \mathbb{R}^{m_i}$ is random unknown noise. The goal is to estimate $\mathbf{x}$. This problem can be formulated as a distributed optimization problem solving

$$\min\ f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\|R_i\mathbf{x} - \mathbf{s}_i\|.$$

We consider the network topology as the digraphs shown in Fig. 2. We employ identical setting and graphs as [3]. In [3], the value of $\epsilon = 0.7$ is chosen for each $\mathcal{G}_a, \mathcal{G}_b, \mathcal{G}_c$. Fig. 3shows the convergence of the D-DSD algorithm for three digraphs displayed in Fig. 2. Once the weight matrix, M, defined in Eq. (5), converges, the D-DSD ensures the convergence. Moreover, it can be observed that the residuals decrease faster as the number of edges increases, from $\mathcal{G}_a$ to $\mathcal{G}_c$. This indicates faster convergence when there are more communication channels available for information exchange. In Fig. 4, we display the trajectories of both states, $\mathbf{x}$ and $\mathbf{y}$, when the D-DSD, Eq. (6), is applied on digraph $\mathcal{G}_a$ with parameter $\epsilon = 0.7$. Recall that in Eqs. (13) and (14), we have shown that as times, k, goes to infinity, the state, $\mathbf{x}_i^k$ of all agents will converges to a same accumulation point, $\overline{\mathbf{z}}^k$, which is the optimal solution of the problem, and $\mathbf{y}_i^k$ of all agents converges to zero, which are shown in Fig. 4. In the next experiment, we compare the performance between the D-DSD and others distributed optimization algorithms over directed graphs. The red curve in Fig. 5 is the plot of residuals of D-DSD on $\mathcal{G}_a$. In Fig. 5, we also shown the convergence behavior of two other algorithms on the same digraph. The blue line is the plot of residuals with a DSD algorithm using a row-stochastic matrix. As we have discussed is Section 2, when the weight matrix is restricted to be row-stochastic, DSD actually minimizes a new objective function $\widehat{f}(\mathbf{x}) = \sum_{i=1}^{n}\pi_i f_i(\mathbf{x})$ where $\boldsymbol{\pi} = \{\pi_i\}$ is the left eigenvector of the weight matrix corresponding to eigenvalue 1. So it does not converge to the true $\mathbf{x}^*$. The black curve shows the convergence behavior of the subgradient-push algorithm, proposed in [15,16]. Our algorithm has the same convergence rate as the subgradient-push algorithm, which is $O\left(\frac{\ln k}{\sqrt{k}}\right)$.

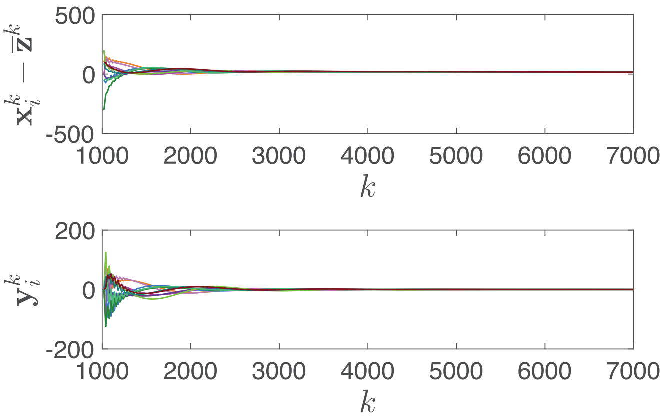


Fig. 4. Sample paths of states, $\mathbf{x}_i^k$, and $\mathbf{y}_i^k$, for all agents on digraphs $\mathcal{G}_a$ with $\epsilon = 0.7$ as D-DSD progresses.

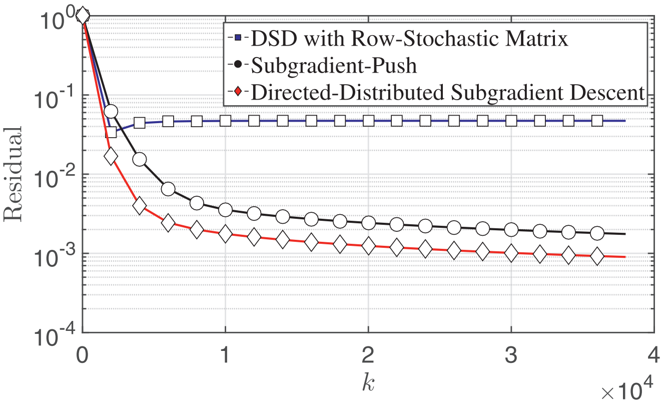


Fig. 5. Plot of residuals $\frac{\|\mathbf{x}_k-\mathbf{x}^*\|_F}{\|\mathbf{x}_0-\mathbf{x}^*\|_F}$. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article).

5.2. Regularized support vector machine

We now study D-DSD over a larger scale network. We consider the regularized support vector machine using the hinge loss function and the 2-norm penalty. We assume that the n examples-label pairs $\{(\mathbf{a}_i, b_i)\}_{i=1}^{n}$, where $\mathbf{a}_i \in \mathbb{R}^m$ and $b_i \in \{+1, -1\}$, are distributed over a directed network. Therefore, the regularized support vector

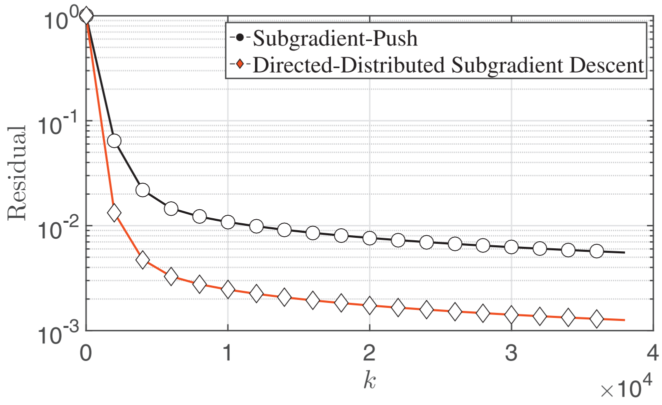


Fig. 6. Plot of residuals $\frac{\|\mathbf{x}_k-\mathbf{x}^*\|_F}{\|\mathbf{x}_0-\mathbf{x}^*\|_F}$ as (D-)DSD progresses.

machine can be formulated as follows:

$$\min f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\max\{0, 1 - b_i(\mathbf{a}_i^\top \mathbf{x})\} + \frac{\lambda}{2}\|\mathbf{x}\|^2,$$

where $\lambda > 0$ is the regularization parameter. Note that f is convex but non-differentiable, and the subgradient is bounded. We let $n = 4$, $N = 50$, and $\lambda = 0.02$. The convergence result is illustrated in Fig. 6. We see that D-DSD is in the same order of convergence rate as the subgradient-push algorithm, which is $O\left(\frac{\ln k}{\sqrt{k}}\right)$.

6. Conclusions

In this paper, we describe Directed-Distributed Subgradient Descent (D-DSD), to solve the problem of minimizing a sum of convex objective functions over a *directed* graph. Existing distributed algorithms, e.g., Distributed Subgradient Descent (DSD), deal with the same problem under the assumption of undirected networks. The primary reason behind assuming the undirected graphs is to obtain a doubly-stochastic weight matrix. The row-stochasticity of the weight matrix guarantees that all agents reach consensus, while the column-stochasticity ensures optimality, i.e., each agents local (sub)gradient contributes equally to the global objective. In a directed graph, however, it may not be possible to construct a doubly-stochastic weight matrix in a distributed manner. In each iteration of D-DSD, we simultaneously constructs a row-stochastic matrix and a column-stochastic matrix instead of only a doubly-stochastic matrix. The convergence of the new weight matrix, depending on the row-stochastic and column-stochastic matrices, ensures agents to reach both consensus and optimality. The analysis shows that the D-DSD converges at a rate of $O(\frac{\ln k}{\sqrt{k}})$, where k is the number of iterations.

References

[1] F. Benezit, V. Blondel, P. Thiran, J. Tsitsiklis, M. Vetterli, Weighted gossip: distributed averaging using non-doubly stochastic matrices, in: Proceedings of the IEEE International Symposium on Information Theory, 2010, pp. 1753–1757, doi:10.1109/ISIT.2010.5513273.

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, Found. Trends Mach. Learn. 3 (1) (2011) 1–122, doi:10.1561/2200000016.

[3] K. Cai, H. Ishii, Average consensus on general strongly connected digraphs, Automatica 48 (11) (2012) 2750–2761. http://dx.doi.org/10.1016/j.automatica.2012.08.003.

[4] J.C. Duchi, A. Agarwal, M.J. Wainwright, Dual averaging for distributed optimization: Convergence analysis and network scaling, IEEE Trans. Automatic Control 57 (3) (2012) 592–606, doi:10.1109/TAC.2011.2161027.

[5] A. Jadbabaie, J. Lin, A.S. Morse, Coordination of groups of mobile autonomous agents using nearest neighbor rules, IEEE Trans. Automatic Control 48 (6) (2003) 988–1001, doi:10.1109/TAC.2003.812781.

[6] B. Johansson, T. Keviczky, M. Johansson, K.H. Johansson, Subgradient methods and consensus algorithms for solving convex optimization problems, in: Proceedings of the 47th IEEE Conference on Decision and Control, 2008, pp. 4185–4190, doi:10.1109/CDC.2008.4739339.

[7] D. Kempe, A. Dobra, J. Gehrke, Gossip-based computation of aggregate information, in: Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science, 2003, pp. 482–491, doi:10.1109/SFCS.2003.1238221.

[8] H.J. Kushner, G. Yin, Stochastic approximation and recursive algorithms and applications, 35, Springer Science & Business Media, 2003.

[9] J. Li, G. Chen, Z. Dong, Z. Wu, Distributed mirror descent method for multi-agent optimization with delay, Neurocomputing 177 (2016) 643–650.

[10] J. Li, G. Chen, Z. Dong, Z. Wu, M. Yao, Distributed mirror descent method for saddle point problems over directed graphs, Complexity 21 (S2) (2016) 178–190.

[11] J. Li, G. Li, Z. Wu, C. Wu, Stochastic mirror descent method for distributed multi-agent optimization, Optim. Lett. (2016) 1–19.

[12] I. Lobel, A. Ozdaglar, D. Feijer, Distributed multi-agent optimization with state-dependent communication, Math. Program. 129 (2) (2011) 255–284, doi:10.1007/s10107-011-0467-x.

[13] G. Mateos, J.A. Bazerque, G.B. Giannakis, Distributed sparse linear regression, IEEE Trans. Signal Process. 58 (10) (2010) 5262–5276, doi:10.1109/TSP.2010.2055862.

[14] J.F.C. Mota, J.M.F. Xavier, P.M.Q. Aguiar, M. Puschel, D-ADMM: a communication-efficient distributed algorithm for separable optimization, IEEE Trans. Signal Process. 61 (10) (2013) 2718–2723, doi:10.1109/TSP.2013.2254478.

[15] A. Nedic, A. Olshevsky, Distributed optimization over time-varying directed graphs, in: Proceedings of the 52nd IEEE Annual Conference on Decision and Control, 2013, pp. 6855–6860, doi:10.1109/CDC.2013.6760975.

[16] A. Nedic, A. Olshevsky, Distributed optimization over time-varying directed graphs, IEEE Trans. Automatic Control PP (99) (2014), doi:10.1109/TAC.2014.2364096. 1–1

[17] A. Nedic, A. Ozdaglar, Distributed subgradient methods for multi-agent optimization, IEEE Trans. Automatic Control 54 (1) (2009) 48–61, doi:10.1109/TAC.2008.2009515.

[18] A. Nedic, A. Ozdaglar, P.A. Parrilo, Constrained consensus and optimization in multi-agent networks, IEEE Trans. Automatic Control 55 (4) (2010) 922–938, doi:10.1109/TAC.2010.2041686.

[19] G. Neglia, G. Reina, S. Alouf, Distributed gradient optimization for epidemic routing: A preliminary evaluation, in: Proceedings of the, 2009, pp. 1–6, doi:10.1109/WD.2009.5449659.

[20] R. Olfati-Saber, J.A. Fax, R.M. Murray, Consensus and cooperation in networked multi-agent systems, Proc. IEEE 95 (1) (2007) 215–233, doi:10.1109/JPROC.2006.887293.

[21] R. Olfati-Saber, R.M. Murray, Consensus protocols for networks of dynamic agents, in: Proceedings of the IEEE American Control Conference, 2, 2003, pp. 951–956, doi:10.1109/ACC.2003.1239709.

[22] R. Olfati-Saber, R.M. Murray, Consensus problems in networks of agents with switching topology and time-delays, IEEE Trans. Automatic Control 49 (9) (2004) 1520–1533, doi:10.1109/TAC.2004.834113.

[23] M. Rabbat, R. Nowak, Distributed optimization in sensor networks, in: Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks, 2004, pp. 20–27, doi:10.1109/IPSN.2004.1307319.

[24] S.S. Ram, A. Nedic, V.V. Veeravalli, Distributed stochastic subgradient projection algorithms for convex optimization, J. Optim Theory Appl. 147 (3) (2010) 516–545, doi:10.1007/s10957-010-9737-7.

[25] C.W. Reynolds, Flocks, herds and schools: a distributed behavioral model, in: Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, ACM, New York, NY, USA, 1987, pp. 25–34, doi:10.1145/37401.37406.

[26] W. Shi, Q. Ling, K. Yuan, G. Wu, W. Yin, On the linear convergence of the ADMM in decentralized consensus optimization, IEEE Trans. Signal Process. 62 (7) (2014) 1750–1761, doi:10.1109/TSP.2014.2304432.

[27] K.I. Tsianos, The role of the network in distributed optimization algorithms: convergence rates, scalability, communication/computation tradeoffs and communication delays, Department of Electrical and Computer Engineering. McGill University, 2013 Ph.D. thesis.

[28] K.I. Tsianos, S. Lawlor, M.G. Rabbat, Consensus-based distributed optimization: practical issues and applications in large-scale machine learning, in: Proceedings of the 50th Annual Allerton Conference on Communication, Control, and Computing, 2012, pp. 1543–1550, doi:10.1109/Allerton.2012.6483403.

[29] K.I. Tsianos, S. Lawlor, M.G. Rabbat, Push-sum distributed dual averaging for convex optimization, in: Proceedings of the 51st IEEE Annual Conference on Decision and Control, 2012, pp. 5453–5458, doi:10.1109/CDC.2012.6426375.

[30] H. Wang, X. Liao, T. Huang, Average consensus in sensor networks via broadcast multi-gossip algorithms, Neurocomputing 117 (2013) 150–160.

[31] E. Wei, A. Ozdaglar, Distributed alternating direction method of multipliers, in: Proceedings of the 51st IEEE Annual Conference on Decision and Control, 2012, pp. 5445–5450, doi:10.1109/CDC.2012.6425904.

[32] L. Xiao, S. Boyd, S.J. Kim, Distributed average consensus with least-mean-square deviation, J. Parallel Distrib. Comput. 67 (1) (2007) 33–46. http://dx.doi.org/10.1016/j.jpdc.2006.08.010.

[33] W. Yang, H. Shi, Sensor selection schemes for consensus based distributed estimation over energy constrained wireless sensor networks, Neurocomputing 87 (2012) 132–137.

[34] D. Yuan, Y. Hong, D.W. Ho, G. Jiang, Optimal distributed stochastic mirror descent for strongly convex optimization, arXiv:1610.04702(2016).

Chenguang Xi received his B.S. degree in Microelectronics from Shanghai Jiao Tong University, China, in 2010, M.S. and Ph.D. degrees in Electrical and Computer Engineering from Tufts University, in 2012 and 2016, respectively. His research interests include distributed optimization, tensor analysis, and source localization.

Qiong Wu received his Bachelor of Applied Science and Master of Science in Applied Mathematics from Harbin Institute of Technology in 2008 and 2010, respectively. He received his Ph.D. degree in Mathematics from Tufts University in 2016. His research interests lie in the areas of statistics and applied probability.

Usman A. Khan received his B.S. degree (with honors) in Electrical Engineering from University of Engineering and Technology, Lahore-Pakistan, in 2002, M.S. degree in Electrical and Computer Engineering (ECE) from University of Wisconsin-Madison in 2004, and Ph.D. degree in ECE from Carnegie Mellon University in 2009. Currently, he is an Assistant Professor with the ECE Department at Tufts University. He received the NSF Career award in Jan. 2014 and is an IEEE Senior Member since Feb. 2014. His research interests lie in efficient operation and planning of complex infrastructures and include statistical signal processing, networked control and estimation, and distributed algorithms. Dr. Khan is on the editorial board of IEEE Transactions on Smart Grid and an associate member of Sensor Array and Multichannel Technical Committee with the IEEE Signal Processing Society. He has served on the Technical Program Committees of several IEEE conferences and has organized and chaired several IEEE workshops and sessions. His graduate students have won multiple Best Student Paper awards. His work was presented as Keynote speech at BiOS SPIE Photonics West–Nanoscale Imaging, Sensing, and Actuation for Biomedical Applications IX.

Please cite this article as: C. Xi et al., On the distributed optimization over directed networks, Neurocomputing (2017), http://dx.doi.org/10.1016/j.neucom.2017.06.038