KYBERNETIKA — VOLUME 57 (2021), NUMBER 2, PAGES 256–271

A STOCHASTIC MIRROR-DESCENT ALGORITHM FOR SOLVING $AXB = C$ OVER AN MULTI-AGENT SYSTEM

Yinghui Wang and Songsong Cheng

In this paper, we consider a distributed stochastic computation of $AXB = C$ with local set constraints over an multi-agent system, where each agent over the network only knows a few rows or columns of matrixes. Through formulating an equivalent distributed optimization problem for seeking least-squares solutions of $AXB = C$, we propose a distributed stochastic mirror-descent algorithm for solving the equivalent distributed problem. Then, we provide the sublinear convergence of the proposed algorithm. Moreover, a numerical example is also given to illustrate the effectiveness of the proposed algorithm.

Keywords: distributed computation of matrix equation, multi-agent system, sublinear convergence, stochastic mirror descent algorithm

Classification: 68M15, 93A14

1. INTRODUCTION

The increasing scale and big data in engineering have posed new challenges for traditional centralized optimization and control recently. As a result, distributed algorithms have attracted much research attention. Particularly, distributed optimization, which agents over the network cooperately seeks a global optimal solution, has become more and more popular [3, 13, 14, 16, 17].

The aforementioned algorithms were based on Euclidean projection, whose local projections could be easily computed. In these cases, the local constraints sets could only be described by simple sets, such as balls, hyperplanes, and bounded constraints. Distributed stochastic mirror descent algorithms [6, 15], based on Bregman divergence [1], were developed to solve distributed problems with complex local constraints sets. [6, 15] provided sublinear convergence when local objective functions were strongly convex and convex, respectively.

In fact, distributed computation of the linear algebraic equation $Ax = c$, which is also recognized as linear regression problem in machine learning or linear system identification in engineering, has attracted much research attention. Mainly distributed algorithms [5, 8, 9, 12] have been proposed to solve the linear algebraic equation $Ax = c$. [8, 9] presented linear convergence of distributed deterministic algorithms, while [5, 12]

DOI: 10.14736/kyb-2021-2-0256

presented sublinear convergence of distributed stochastic algroithm when agents over the network could only get incomplete row or column information of matrixes.

Moreover, distributed computation of different matrix equations have been another hot topic, since matrix equations have played important roleds in engineering areas. [17] studied four kinds of different distirbuted computations of unconstrainted linear matrix equation $AXB = C$ (a generation of matrix equation $AX = C$ and algerbraic equation $Ax = c$) and provided primal-dual algorithms for solving these computations. Further, [4] and [2] provided distributed algorithms for solving Sylvester equation $AX + XB = C$ and constrainted Stein equation $X + AXB = C$. It was noteworthy that the algorithms proposed in [2, 4, 17], whose linear rates were not given explicitly, were all continuous-time algorithms.

The objective of this paper is to study a distributed stochastic computation of constrainted linear matrix equation $AXB = C$. Moreover, mirror descent techniques are used in our algorithm so as to reduce the computation cost of complex local constraints. The contributions of this paper are summarized as follows:

(a) We studied a distributed stochastic computation of $AXB = C$ discussed in [17] constaints and proposed a distributed stochastic mirror descent algorithm for solving $AXB = C$ with complex local constaints. [4, 17] studieds unconstraint matrix equations and [2] studied matrix equation with global constraints, while we studied $AXB = C$ with local constraints. Moreover, we do not need assumptions that each agents get explicit information of some row and column of matrixs A, B and C.

(b) We give the $O(\frac{1}{T})$ (sublinear) convergence in expectation of the proposed algorithm when the second moments of gradient noises were summable, which recovers the best convergence rate for distributed mirror-descent algorithms [6, 15]. Still, we show that agents can find an ϵ-solution in $O(\frac{1}{\epsilon})$ communication rounds with $O(\frac{1}{\epsilon})$ rounds of local stochastic gradient evaluations of the proposed algorithm.

The rest of the paper is organized as follows. Preliminaries are given in Section 2. In Section 3, the distributed description of solving $AXB = F$ with set constraints and a distributed stochastic mirror-descent algorithm are presented. The proposed algorithm is further analyzed in Section 4 and numerical examples are given in Section 5. Finally, the conclusion of this paper is offered in Section 6.

2. PRELIMINARIES

In this section, we introduce preliminaries of convex analysis and graph theory, which will be further used in the analysis of our distributed stochastic mirror-descent algorithm for solving $AXB = C$.

2.1. Convex Analysis

First, we present several properties of convex functions.

Definition 1. (Convexity and Lipschitz continuity, subgradient)

(a) A function $f : \mathbb{R}^{mr} \to \mathbb{R}$ is be κ-Lipschtiz continuous for a constant $\kappa > 0$ if $\|f(x_1) - f(x_2)\| \leqslant \kappa\|x_1 - x_2\|$, $\forall x_1, x_2 \in \mathbb{R}^{mr}$.

(b) For $x \in \Omega$, define by $\partial f(x)$ the subdifferential of a nonsmooth convex function f at x. For any $x_1, x_2 \in \Omega$, the following inequality holds

$$f(x_2) - f(x_1) \geqslant \langle \nabla f(x_1), x_2 - x_1 \rangle, \quad \forall \nabla f \in \partial f(x),$$

where $\nabla f(x)$ is a subgradient of function f at x.

Next, we give the concept of Bregeman distance function, which is important for distributed first-order methods as a generalization of Euclidean projections.

Definition 2. (Bregman divergence [1]) Given a strongly convex and differentiable funcion $\phi : \mathbb{R}^{mr} \to \mathbb{R}$, the Bregman divergence $B_i(\cdot, \cdot)$ is introduced by function ϕ as

$$B_i(x, z) = \phi(x) - \phi(z) - \langle \nabla\phi(z), x - z \rangle \geqslant 0.$$

The following assumption on the Bregman function $B(\cdot, \cdot)$ is adopted in this paper, which is widely used in [6, 15].

Assumption 1. (a) Suppose $B_i(x, z)$ is strongly convex with module 1, and then $\langle \nabla\phi(x) - \nabla\phi(z), x - z \rangle \geqslant \|x - z\|^2$, $\forall x, z \in \mathbb{R}^{mr}$ and $B(x, z) \geqslant \frac{1}{2}\|x - z\|^2$, $\forall x, z \in \mathbb{R}^{mr}$.

(b) Suppose $B_i(x, z)$ is growing quadratically with quadratic constant ρ in Ω_i, i.e., for all $x, z \in \Omega_i$ and $i = 1, \dots, m$, there exists a constant $\rho > 0$ such that $B(x, z) \leqslant \frac{\rho}{2}\|x - z\|^2$.

Remark 3. We provide some choices of convex function $\phi(\cdot)$ and its corresponding Bregman divergence function $B(\cdot, \cdot)$ [1, 15]:

(a) $\phi(x) = \frac{1}{2}||x||_2^2$, the corresponding $B(x, y) = \frac{1}{2}||x - y||_2^2$.

(b) $\phi(x) = \sum_{p=1}^{d} [x]_p \log[x]_p$, the corresponding $B(x, y) = \sum_{p=1}^{d} [x]_p \log \frac{[x]_p}{[y]_p}$, which is known as the Kullback-Leibler divergence.

(c) $\phi(x) = -\log x$, the corresponding $B(x, y) = \frac{x}{y} - \log\left(\frac{x}{y}\right) - 1$, which is known as the itakura-saito divergence.

2.2. Graph theory

We still need to consider the communication between agents. The communication topology between agents is described by an undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{A})$, where $\mathcal{V} = \{1, 2, ..., n\}$ denotes the agent set, $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ denotes the edge set between agents, and matrix $\mathcal{A}$ as the adjacency matrix, whose elements are given by

$$\begin{cases} a_{i,j} & = a_{j,i} > 0, \quad \text{if} \quad (i,j) \in \mathcal{E}, j \neq i \\ a_{i,j} & = 0, \quad \text{otherwise.} \end{cases}$$

Denote by $\mathcal{L} = \mathcal{D} - \mathcal{A}$ the weighted Laplacian matrix of graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$, where $\mathcal{D}$ is an $m \times m$ diagonal matrix whose diagonal elements are $\mathcal{D}_{ii} = \sum_{j=1}^{m} a_{i,j}, i \in \mathcal{V}$.

An edge $(j, i) \in \mathcal{E}$ means that agent j can communicate with agent i. A path of $\mathcal{G}$ is a sequence of distinct agents where any pair of consecutive agents in the sequence has an edge in $\mathcal{E}$. Agent j is connected to agent i if there exists a path from agent j to agent i. The graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ is a connected graph if any two agents over the network are connected. The communication graph between agents satisfies the following assumption.

Assumption 2. The communication graph is connected.

3. PROBLEM DESCRIPTION AND ALGORITHM DESIGN

In this section, we formulate a standard distributed form of solving matrix equation $AXB = C$ and design a distributed stochastic mirror descent algorithm for solving it.

3.1. Problem description

Consider the distributed computation of a solution to the following matrix equation:

$$\min_X \|AXB - C\|^2, X \in \mathcal{X} \tag{1}$$

where $A \in \mathbb{R}^{m\times p}$, $B \in \mathbb{R}^{q\times r}$, and $C \in \mathbb{R}^{m\times r}$ are known matrices in piror knowledge, and $X \in \mathbb{R}^{p\times q}$ is an unknown matrix to be calculated. Though the solution to matrix equation (1) may not exist, there always exists a least squares solution to equation (1), which is defined as follows.

Definition 4. A *least squares solution* to matrix equation (1) is a solution to the optimization problem

$$\min_X \|AXB - C\|^2.$$

In this paper, we assume that each agent i over the network has prior knowledge to $A_{vi} \in \mathbb{R}^{m_i\times p}$, $B_{li} \in \mathbb{R}^{q\times r_i}$, and $C_{li} \in \mathbb{R}^{m\times r_i}$, $\sum_{i=1}^n m_i = m$, $\sum_{i=1}^n r_i = r$, where A_{vi}, B_{li}, and C_{li} are given through the following equations:

$$\begin{aligned} A &= \begin{bmatrix} A_{v1} \\ \vdots \\ A_{vn} \end{bmatrix} \in \mathbb{R}^{m\times p}, \\ B &= \begin{bmatrix} B_{l1}, \dots, B_{ln} \end{bmatrix} \in \mathbb{R}^{q\times r} \\ C &= \begin{bmatrix} C_{l1}, \dots, C_{ln} \end{bmatrix} \in \mathbb{R}^{m\times r}. \end{aligned} \tag{2}$$

Through neighbourhood information interchange over the network, agents can solve the least squares solution to equation (1) cooperatively.

Remark 5. A special case $B^\top X^\top = C^\top$ of equation (1) has been widely investigated in [8, 9, 11] and references therein. In this special setting, $X, C \in \mathbb{R}$ with $A = 1$. Still, each agent has prior knowledge to a row sub-block of $B^\top$ and vector $C^\top$. However, in the original setting (1), sub-blocks of matrices A, B, and X are coupled and hence we need to investigate new distributed computing technologies in our algorithm design.

We make the following efforts so as to handle the couplings between sub-blocks of matrices A, B, and X in equation (1). First, a substitutional variable Y which is equivalent to $Y = AX$ and $YB_{li} = C_{li}$ for $i \in \{1, \ldots, n\}$ is introduced. Define $X_i \in \mathbb{R}^{p\times q}$ and $Y_i \in \mathbb{R}^{m\times q}$ as agent $i \in \{1, \ldots, n\}$'s estimates for X and Y, respectively. A consensus-based substitutional decomposition which requires the agreements between both variables X_i and Y_i is required:

$$Y_i B_{li} = C_{li}, \quad Y_i = Y_j, \quad i, j \in \{1, \ldots, n\}, \tag{3}$$

$$AX_i = Y_i, \quad X_i = X_j. \tag{4}$$

However, (4) is not in a fully distributed form for the global information matrix A need to be known for each agent. Further, define $Y_i \triangleq \begin{bmatrix} Y_i^{v1} \\ \vdots \\ Y_i^{vn} \end{bmatrix}$, where $Y_i^{vj} \in \mathbb{R}^{m_j \times q}$ for all $i, j \in \{1, \ldots, n\}$ to decompose (4). Since $Y_i = Y_j$ holds for all $i, j \in \{1, \ldots, n\}$ in (3), (4) can be rewritten as

$$A_{vi} X_i = Y_i^{vi}, \quad X_i = X_j, \quad i, j \in \{1, \ldots, n\}. \tag{5}$$

Hence, the matrix equation (1) is equivalent to linear matrix equations (3) and (5). Define extended matrices $X_{\mathrm{E}} = [X_1^{\mathrm{T}}, \ldots, X_n^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{np\times q}$ and $Y_{\mathrm{E}} = [Y_1^{\mathrm{T}}, \ldots, Y_n^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{nm\times q}$. Based on (3) and (5), we reformulate the distributed computation of (1) as the following distributed optimization problem as done in [17]:

$$\min_{X_{\mathrm{E}}, Y_{\mathrm{E}}} \quad \sum_{i=1}^{n} \|Y_i B_{li} - C_{li}\|^2, \tag{6a}$$

$$\text{s. t.} \quad X_i = X_j,\, Y_i = Y_j,\, A_{vi} X_i = Y_i^{vi},\, i, j \in \{1, \ldots, n\}, \tag{6b}$$

where agent i has prior knowledge to sub-blocks A_{vi}, B_{li}, C_{li}, and estimates X_i and Y_i with its local information.

3.2. Algorithm design

Here we propose a distributed algorithm with stochastic gradient to solve the problem (6), which is called DSMD, where the stochastic gradients $\nabla g_i(x_i, \xi_i)$ satisfy the following assumption:

Assumption 3. For each $i \in \mathcal{V}$, the stochastic gradients $\nabla g_i(Y_i, \xi_i)$ and stochastic function $g_i(Y_i, \xi_i)$ satisfies:

(a) $\mathbb{E}_{\xi_i^k}[\nabla g_i(Y_i^k, \xi_i^k) \big| Y_i^k] = B_{li}^{\top}\big(Y_i^k B_{li} - C_{li}\big)$.

(b) $\mathbb{E}_{\xi_i^k}\Big[\big\|\nabla g_i(Y_i^k, \xi_i^k) - \nabla f_i(x_i^k)\big\|^2 \Big| x_i^k\Big] \leqslant \delta_k^2$, $\sum_{k=1}^{\infty} \delta_k^2 < \infty$.

(c) $\mathbb{E}_{Y_i^k}[g_i(Y_i^k, \xi_i^k) \big| Y_i^k] = \frac{1}{2}\|Y_i^k B_{li} - C_{li}\|^2$.

Different from [6, 10], where the variance of gradient noises are assumed to be uniformly bounded, in Assumption 3, we require the variance of gradient noises to be summable. Infinite sampling (see [7]) is a guarantee for Assumption 3. The special choice of gradient noises guarantees the sublinear convergence of DSMD converges to the optimal solution of problem (6) with fixed stepsizes. ‘

Algorithm DSMD Distributed Stochastic Mirror Descent Algorithm for Solving $AXB = C$

1: Initialization of $X_i \in \mathcal{X}$ for all $i = 1, 2, \ldots n$.

2: Update $\hat{Z}_i^k = (\hat{X}_i^k, \hat{Y}_i^k)$ according to

$$\hat{X}_i^k = 2X_i^{k-1} - X_i^{k-2} \tag{7}$$

$$\hat{Y}_i^k = 2Y_i^{k-1} - Y_i^{k-2} \tag{8}$$

3: Update $\Lambda_i^k = (\Lambda_{1i}^k, \Lambda_{2i}^k, \Lambda_{3i}^k)$ according to

$$\Lambda_{1i}^k = \Lambda_{1i}^{k-1} + \frac{1}{\alpha_i}\Big[A_{vi}\hat{X}_i^k - \hat{Y}_i^{vi,k}\Big] \tag{9}$$

$$\Lambda_{2i}^k = \Lambda_{2i}^{k-1} + \frac{1}{\beta_i}\sum_{j\in N_i} l_{i,j}\hat{X}_j^k \tag{10}$$

$$\Lambda_{3i}^k = \Lambda_{3i}^{k-1} + \frac{1}{\gamma_i}\sum_{j\in N_i} l_{i,j}\hat{Y}_j^k \tag{11}$$

4: Update $Z_i^k = (Z_{1i}^k, Z_{2i}^k) = (X_i^k, Y_i^k)$ according to

$$X_i^k = \arg\min_{X_i}\left\{\Big\langle \Lambda_{1i}^k, A_{vi}X_i - \hat{Y}_i^{vi,k}\Big\rangle + \Big\langle \sum_{j\in N_i} l_{i,j}\Lambda_{2i}^k, X_i\Big\rangle + \mu_{1i}B_i\Big(X_i^{k-1}, X_i\Big)\right\} \tag{12}$$

$$Y_i^k = \arg\min_{Y_i}\left\{g_i(Y_i, \xi_i^k) + \Big\langle \sum_{j\in N_i} l_{i,j}\Lambda_{3i}^k, Y_i\Big\rangle + \Big\langle \Lambda_{1i}^k, A_{vi}X_i^k - Y_i^{vi}\Big\rangle + \mu_{2i}B_i\Big(Y_i^{k-1}, Y_i\Big)\right\} \tag{13}$$

5: Check the end condition of algorithm. If the condition is satisfied, then the algorithm is terminated. Otherwise, $k := k + 1$ and go to Step 2.

Explanations for DSMD are given as follows.

(a) In each iteration step of DSMD, (10), (11), (12) and (13) involve the neighborhood communication between agents, (7), (8) and (9) can be performed by each agents separately. Through proper choosing of stepsizes $\{\alpha_i\}$, $\{\beta_i\}$, $\{\gamma_i\}$, $\{\mu_{1i}$ and $\{\mu_{2i}$, we can find ϵ-solutions of DSMD, i.e. $\mathbf{x}^* \in \Omega^m$ such that $\frac{1}{2}\sum_{i=1}^n \mathbb{E}\Big\|Y_iB_{li} - C_{li}\Big\|^2 \leqslant \epsilon$

within $O(1/\epsilon)$ iterations.

(b) The basic idea of DSMD is inspired by distributed primal-dual algorithm [17] (deterministic and continuous-time algorithm). However, the refined steps (7) – (8) and the existence of Bregman penalty term $B\big(x_i, x_i^{k-1}\big)$ and $B\big(Y_i, Y_i^{k-1}\big)$ guarantee an ϵ-solution of DSMD within $(1/\epsilon)$ iterations.

4. MAIN RESULTS

In this section, we provide the convergence results of Algorithm DSMD, whose analysis are given in Appendix.

For the convenience of denoting, we redefine some related variables of Algorithm DSMD as follows

$$\begin{cases} D_{1i}^{(\cdot)} & = A_{vi}X_i^{(\cdot)} - Y_i^{vi,\cdot} \\ D_{2i}^{(\cdot)} & = \sum_{j\in N_i} l_{i,j}X_j^{(\cdot)} \\ D_{3i}^{(\cdot)} & = \sum_{j\in N_i} l_{i,j}Y_j^{(\cdot)} \\ \Lambda^{(\cdot)} & = \Big(\Lambda_1^{(\cdot)}, \Lambda_2^{(\cdot)}, \Lambda_3^{(\cdot)}\Big) \end{cases} \tag{14}$$

Based on the redefined variables in (14), the Lagrangian function of optimization problem in (6) can be formulated as

$$L\Big(Z_{\mathrm{E}}, \Lambda\Big) = \frac{1}{2}\sum_{i=1}^{n}\Big\|Y_iB_{li} - C_{li}\Big\|^2 + \sum_{r=1}^{3}\sum_{i=1}^{n}\langle\Lambda_{ri}, D_{ri}\rangle \tag{15}$$

For a pair of feasible solutions $Z_E = \big(X_E, Y_E\big)$ and $\Lambda = \big(\Lambda^1, \Lambda^2, \Lambda^3\big)$ of the Lagrangian function $L(X_{\mathrm{E}}, Y_{\mathrm{E}}, \Lambda^1, \Lambda^2, \Lambda^3)$, we define the primal-dual gap function $Q(Z_E, \bar{\Lambda})$ by

$$Q\Big[\bar{Z}_E, \bar{\Lambda}; Z_E, \Lambda\Big] = L\Big(\bar{Z}_{\mathrm{E}}, \Lambda\Big) - L\Big(Z_{\mathrm{E}}, \bar{\Lambda}\Big). \tag{16}$$

We first provide a lemma to characterize the solution between primal steps (12) – (13) and dual steps (9) – (11).

Lemma 6. (Lan et al. [6]) Let $\phi : U \to \mathbb{R}$ be a differentiable convex function and $B(x,z) = \phi(x) - \phi(z) - \langle\nabla\phi(x), x - z\rangle$. Let the convex function $q : U \to \mathbb{R}$, points $\bar{x}, \bar{y} \in U$ and the scalars $\mu_1, \mu_2 \in \mathbb{R}$ be given. If

$$u^* \in \arg\min\Big\{q(u) + \mu_1B(\bar{x}, u) + \mu_2B(\bar{y}, u)\Big\},$$

then for any $u \in U$, we have

$$q(u^*) + \mu_1B(\bar{x}, u^*) + \mu_2B(\bar{y}, u^* \leqslant q(u) + \mu_1B(\bar{x}, u) + \mu_2B(\bar{y}, u) - (x + y)B(u^*, u).$$

In the following, we provide a selection of stepsizes α_i, β_i, γ_i, μ_{1i} and μ_{2i} and establish the complexity of DSMD through computing the ϵ-solution of problem 1.

Theorem 7. Suppose that stepsizes α_i, β_i and γ_i, μ_{1i} and μ_{2i} are set to

$$\alpha_i = \beta_i = \gamma_i = 1, \mu_{1i} = 2\|A_i\|^2 + 2\|L_i\|^2, \mu_{1i} = 2 + 2\|L_i\|^2. \tag{17}$$

Then for any $K \geqslant 1$, we have

$$\begin{aligned} &\frac{1}{2}\mathbb{E}\|\bar{Y}_i^k B_{li} - C_{li}\|^2 - \frac{1}{2}\mathbb{E}\|Y_i^* B_{li} - C_{li}\|^2 \\ \leqslant &\frac{1}{K}\sum_{i=1}^{n}\left[\mu_{1i}\mathbb{E}B_i\Big(X_i^0, X_i\Big) + \sum_{i=1}^{n}\mu_{2i}\mathbb{E}B_i\Big(Y_i^0, Y_i\Big) + \frac{\alpha_i}{2}\mathbb{E}\|\Lambda_{1i}^0\|^2 + \frac{\beta_i}{2}\mathbb{E}\|\Lambda_{2i}^0\|^2 + \frac{\gamma_i}{2}\mathbb{E}\|\Lambda_{3i}^0\|^2\right] \end{aligned} \tag{18}$$

and

$$\begin{aligned} &\mathbb{E}\|\frac{1}{K}\sum_{k=1}^{K}\rho_{ii}^K\| \\ \leqslant &\frac{1}{K}\sum_{i=1}^{n}\left[\mu_{1i}B_i\Big(X_i^0, X_i\Big) + \sum_{i=1}^{n}\mu_{2i}\mathbb{E}B_i\Big(Y_i^0, Y_i\Big) + \frac{\alpha_i}{2}\mathbb{E}\|\Lambda_{1i}^0\|^2 + \frac{\beta_i}{2}\mathbb{E}\|\Lambda_{2i}^0\|^2 + \frac{\gamma_i}{2}\mathbb{E}\|\Lambda_{3i}^0\|^2\right] \end{aligned} \tag{19}$$

where

$$\begin{cases} \rho_{1i}^K = D_{1i}^K - D_{1i}^{K-1} + \alpha_i\big(\Lambda_{1i}^K - \Lambda_{1i}^0\big) \\ \rho_{2i}^K = D_{2i}^K - D_{2i}^{K-1} + \beta_i\big(\Lambda_{2i}^K - \Lambda_{2i}^0\big) \\ \rho_{3i}^K = D_{3i}^K - D_{3i}^{K-1} + \alpha_i\big(\Lambda_{3i}^K - \Lambda_{3i}^0\big). \end{cases}$$

Proof. The proof of Theorem 7 can be derived directly from Lemma 10 and Lemma 11. □

From Theorem 7, we can see that the complexity of Algorithm DSMD for computing an expected (ϵ, δ)-solution is $O(\frac{1}{\epsilon})$ for the primal functional optimality and $O(\frac{1}{\epsilon})$ for the constraint violation. Since each iteration involves a constant number of communication rounds and the number of communications between agents required is also in the same order.

5. SIMULATION

Consider a 10-agents network, whose communication topology is given in Figure 1.

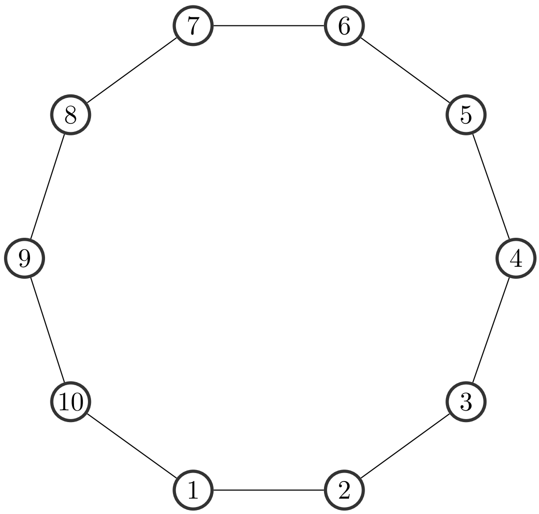

Fig. 1. Topology between agents over the network.

Consider to solve linear matrix equation (1), where

$$A = \begin{bmatrix} 2 & 1 \\ 4 & 3 \\ 1 & 3 \\ 2 & 4 \\ 3 & 1 \\ 0 & 5 \\ 3 & 0 \\ 2 & 1 \\ 3 & 2 \\ 5 & 2 \end{bmatrix} \quad B = \begin{bmatrix} 1 & -1 \\ 2 & 1 \\ 3 & 1 \\ 2 & 3 \\ 2 & 1 \\ 3 & 1 \\ 1 & 2 \\ 2 & 3 \\ 5 & 1 \\ 1 & 4 \end{bmatrix} \quad \text{and } C = \begin{bmatrix} 0 & 0 \\ 2 & 1 \\ 3 & 5 \\ 1 & 4 \\ 1 & 2 \\ 3 & 2 \\ 2 & 4 \\ 2 & 1 \\ 1 & 3 \\ 2 & 4 \end{bmatrix}.$$

We solve a least squares solution with Algorithm DSMD

$$\begin{aligned} &X \\ =&[X_{l1}; X_{l2}; X_{l3}; X_{l4}; X_{l5}; X_{l6}; X_{l7}; X_{l8}; X_{19}; X_{l10}] \\ =&\begin{bmatrix} -0.0187 & 0.0064 & 0.0022 & 0.0356 & 0.0064 & 0.0022 & 0.0251 & 0.0356 & -0.0060 & 0.0543 \\ -0.0084 & 0.0150 & 0.0172 & 0.0361 & 0.0150 & 0.0172 & 0.0234 & 0.0361 & 0.0216 & 0.0445 \end{bmatrix} \end{aligned}$$

where agent i estimates X_{li} for $i = 1, 2 \ldots 10$. Figure 2 shows that the trajectory of $\|AX_i^k B - C\|_F$ for 4 agents converges to a least squares solution.

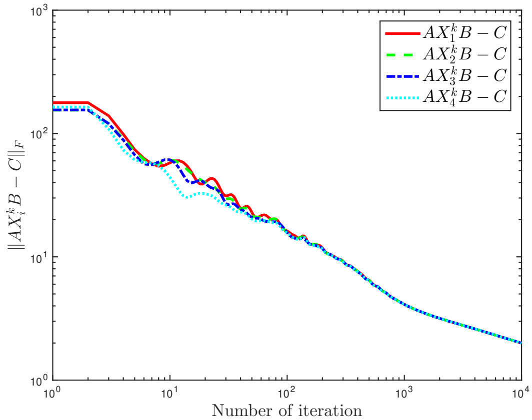

Fig. 2. The trajectories of $\|AX_i^kB - C\|$ for each agent i

.

6. CONCLUSION

A distributed stochastic mirror descent algorithm for solving $AXB = C$ was proposed in this paper. Sublinear convergence to the optimal solution and the $O(\frac{1}{\epsilon})$ complexity were also given to the proposed algorithm. Finally, simulations were given to the proposed algorithm to verify its effectiveness.

ACKNOWLEDGEMENT

This work was supported by the State Key Laboratory of Intelligent Control and Decision of Complex Systems.

(Received September 6, 2020)

REFERENCES

[1] L. M. Bregman: The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. USSR Computational Mathematics and Mathematical Physics *7* (1967), 200–217. DOI:10.1016/0041-5553(67)90152-8

[2] G. Chen, X. Zeng, and Y. Hong: Distributed optimisation design for solving the Stein equation with constraints. IET Control Theory Appl. *13* (2019), 2492–2499. DOI:10.1049/iet-cta.2019.0140

[3] S. Cheng and S. Liang: Distributed optimization for multi-agent system over unbalanced graphs with linear convergence rate. Kybernetika *56* (2020), 559–577. DOI:10.14736/kyb-2020-3-0559

[4] W. Deng, X. Zeng, and Y. Hong: Distributed computation for solving the Sylvester equation based on optimization. IEEE Control Systems Lett. *4* (2019), 414–419. DOI:10.1109/LCSYS.2019.2942711

[5] M. R. Gholami, M. Jansson, E. G. Ström et al.: Diffusion estimation over cooperative multi-agent networks with missing data. IEEE Trans. Signal Inform. Process. over Networks *2* (2016), 27–289. DOI:10.1109/tsipn.2016.2570679

[6] G. Lan, S. Lee, and Y. Zhou: Communication-efficient algorithms for decentralized and stochastic optimization. Math. Programm. *180* (2020), 237–284. DOI:10.1007/s10107-018-1355-4

[7] J. Lei, U. V. Shanbhag, J. S. Pang et al.: On synchronous, asynchronous, and randomized best-response schemes for stochastic Nash games. Math. Oper. Res. *45* (2020), 157–190. DOI:10.1287/moor.2018.0986

[8] J. Liu, A. S. Morse, A. Nedic, et a.: Exponential convergence of a distributed algorithm for solving linear algebraic equations. Automatica *83* (2017), 37–46. DOI:10.1016/j.automatica.2017.05.004

[9] S. Mou, J. Liu, and A. S. Morse: A distributed algorithm for solving a linear algebraic equation. IEEE Trans. Automat. Control *60* (2015), 2863–2878. DOI:10.1109/TAC.2015.2414771

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

[11] G. Shi, B. D. O. Anderson, and U. Helmke: Network flows that solve linear equations. IEEE Trans. Automat. Control *62* (2016), 2659–2674. DOI:10.1109/TAC.2016.2612819

[12] Y. Wang, P. Lin, and Y. Hong: Distributed regression estimation with incomplete data in multi-agent networks. Science China Inform. Sci. *61* (2018), 092202.

[13] Y. Wang, P. Lin, and H. Qin: Distributed classification learning based on nonlinear vector support machines for switching networks. Kybernetika *53* (2017), 595–611. DOI:10.14736/kyb-2017-4-0595

[14] Y. Wang, W. Zhao, Y. Hong et al.: Distributed subgradient-free stochastic optimization algorithm for nonsmooth convex functions over time-varying networks. SIAM J. Control Optim. *57* (2019), 2821–2842. DOI:/10.1137/18M119046X

[15] D. Yuan, Y. Hong, D. W. C. Ho et al.: Optimal distributed stochastic mirror descent for strongly convex optimization. Automatica *90*(2018), 196–203. DOI:10.1016/j.automatica.2017.12.053

[16] D. Yuan, Y. Hong, D. W. C. Ho et al.: Distributed mirror descent for online composite optimization. IEEE Trans. Automat. Control (2020).

[17] X. Zeng, S. Liang, Y. Hong et al.: Distributed computation of linear matrix equations: An optimization perspective. IEEE Trans. Automat. Control *64* (2018), 1858–1873. DOI:10.1109/TAC.2018.2847603

Yinghui Wang, School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083. P. R. China.
e-mail: wangyinghuisdu@163.com

Songsong Cheng, (Corresponding Author) School of Electrical Engineering and Automation, Anhui University, Hefei 230601. P. R. China.
e-mail: sscheng@amss.ac.cn

APPENDIX

Proof of Theorem 7

Before we provide proofs for Theorem 7, we still need to present the following results which characterize the convergence properties of Algorithm DSMD.

Define

$$\begin{cases} P_{1i}\Big(X_i, Y_i, \Lambda_i\Big) & = \Big\langle \Lambda_{1i}, A_{vi}X_i - Y_i^{vi}\Big\rangle + \Big\langle \sum_{j\in N_i} l_{i,j}\Lambda_{2i}, X_i\Big\rangle \\ P_{2i}\Big(X_i, Y_i, \Lambda_i\Big) & = \frac{1}{2}\|Y_iB_{li} - C_{li}\|^2 + \Big\langle \sum_{j\in N_i} l_{i,j}\Lambda_{3i}, Y_i\Big\rangle + \Big\langle \Lambda_{1i}, A_{vi}X_i - Y_i^{vi}\Big\rangle \end{cases} \tag{20}$$

The following lemma gives upper bounds of $P_{1i}\Big(X_i, Y_i, \Lambda_i\Big)$ and $P_{2i}\Big(X_i, Y_i, \Lambda_i\Big)$, which is useful in the estimate of $\mathbb{E}Q\Big[\bar{Z}_E^K, \bar{\Lambda}^K; Z_E, \Lambda\Big]$.

Lemma 8. Let the iterate $\{Z_E^k\}$ be generated by Algorithm DSMD. $P_{1i}\Big(X_i, Y_i, \Lambda_i\Big)$ and $P_{2i}\Big(X_i, Y_i, \Lambda_i\Big)$ are defined in (20). Then,

$$\begin{cases} \sum_{k=1}^K \mathbb{E}\Big[P_{1i}\Big(X_i^k, Y_i^k, \Lambda_i^k\Big) - P_{1i}\Big(X_i, Y_i^k, \Lambda_i^k\Big)\Big] \leqslant & \mu_{1i}\mathbb{E}B_i\Big(X_i^0, X_i\Big) - \sum_{k=1}^K \mu_{1i}\mathbb{E}B_i\Big(X_i^{k-1}, X_i^k\Big) \\ \sum_{k=1}^K \mathbb{E}\Big[P_{2i}\Big(X_i^k, Y_i^k, \Lambda_i^k\Big) - P_{2i}\Big(X_i^k, Y_i, \Lambda_i^k\Big)\Big] \leqslant & \mu_{2i}\mathbb{E}B_i\Big(Y_i^0, Y_i\Big) - \sum_{k=1}^K \mu_{2i}\mathbb{E}B_i\Big(Y_i^{k-1}, Y_i^k\Big). \end{cases} \tag{21}$$

P r o o f . Applying Lemma 6 to (12) and (13), for all $\Lambda_{1i} \in \mathbb{R}^{m_i\times q}$, $\Lambda_{2i} \in \mathbb{R}^{p\times q}$, $\Lambda_{3i} \in \mathbb{R}^{m\times q}$, $X_i \in \mathbb{R}^{p\times q}$ and $Y_i \in \mathbb{R}^{m\times q}$, we can get

$$\mathbb{E}P_{1i}\Big(X_i^k, Y_i^k, \Lambda_i^k\Big) - \mathbb{E}P_{1i}\Big(X_i, Y_i^k, \Lambda_i^k\Big) \leqslant \mu_{1i}\Big[\mathbb{E}B_i\Big(X_i^{k-1}, X_i\Big) - \mathbb{E}B_i\Big(X_i^k, X_i\Big) - \mathbb{E}B_i\Big(X_i^{k-1}, X_i^k\Big)\Big], \tag{22}$$

and

$$\mathbb{E}P_{2i}\Big(X_i^k, Y_i^k, \Lambda_i^k\Big) - \mathbb{E}P_{2i}\Big(X_i^k, Y_i, \Lambda_i^k\Big) \leqslant \mu_{2i}\Big[\mathbb{E}B_i\Big(Y_i^{k-1}, Y_i\Big) - \mathbb{E}B_i\Big(Y_i^k, Y_i\Big) - \mathbb{E}B_i\Big(Y_i^{k-1}, Y_i^k\Big)\Big]. \tag{23}$$

Summing both sides of (22) from $k = 1$ to $k = K$, it yields

$$\sum_{k=1}^K \mathbb{E}\Big[P_{1i}\Big(X_i^k, Y_i^k, \Lambda_i^k\Big) - P_{1i}\Big(X_i, Y_i^k, \Lambda_i^k\Big)\Big] \leqslant \mu_{1i}\mathbb{E}B_i\Big(X_i^0, X_i\Big) - \sum_{k=1}^K \mu_{1i}\mathbb{E}B_i\Big(X_i^{k-1}, X_i^k\Big), \tag{24}$$

where the inequality can be deduced by the face that $B_i\Big(X_i^T, X_i\Big) \geqslant 0$. Analogously, we can give an upper bound for $\sum_{k=1}^K \mathbb{E}\Big[P_{2i}\Big(X_i^k, Y_i^k, \Lambda_i^k\Big) - P_{2i}\Big(X_i^k, Y_i, \Lambda_i^k\Big)\Big]$, which completes the proof. □

Define

$$\Gamma_{ri}^k = \langle \Lambda_{ri} - \Lambda_{ri}^k, D_{ri}^k\rangle, \quad r = 1, 2, 3 \tag{25}$$

Next, we give estimates of Γ_{ri}^k, $r = 1, 2, 3$.

Lemma 9. Let the iterate $\{\Lambda^k\}$ be generated by Algorithm DSMD. Γ_{ri}^k, $r=1,2,3$ are defined in (25). Then,

$$\begin{cases} \sum_{k=1}^K \mathbb{E}\Gamma_{1i}^k \leqslant & \frac{\alpha_i}{2}\mathbb{E}\Big[\|\Lambda_{1i}-\Lambda_{1i}^0\|^2-\|\Lambda_{1i}-\Lambda_{1i}^K\|^2\Big]+\mathbb{E}\Big\langle D_{1i}^K-D_{1i}^{K-1},\Lambda_{1i}-\Lambda_{1i}^K\Big\rangle \\ & +\sum_{k=1}^K \mathbb{E}\Big[\frac{\|A_i\|^2}{2\alpha_{i1}}\big\|X_i^{k-1}-X_i^{k-2}\big\|^2+\frac{1}{2\alpha_{i2}}\big\|Y_i^{k-1}-Y_i^{k-2}\big\|^2\Big], \\ \sum_{k=1}^K \mathbb{E}\Gamma_{2i}^k \leqslant & \frac{\beta_i}{2}\mathbb{E}\Big[\|\Lambda_{2i}-\Lambda_{2i}^0\|^2-\|\Lambda_{2i}-\Lambda_{2i}^K\|^2\Big]+\mathbb{E}\Big\langle D_{2i}^K-D_{2i}^{K-1},\Lambda_{2i}-\Lambda_{2i}^K\Big\rangle \\ & +\sum_{k=1}^K \mathbb{E}\Big[\frac{\|L_i\|^2}{2\beta_i}\big\|X_i^{k-1}-X_i^{k-2}\big\|^2\Big] \\ \sum_{k=1}^K \mathbb{E}\Gamma_{3i}^k \leqslant & \frac{\gamma_i}{2}\mathbb{E}\Big[\|\Lambda_{3i}-\Lambda_{3i}^0\|^2-\|\Lambda_{3i}-\Lambda_{3i}^K\|^2\Big]+\mathbb{E}\Big\langle D_{3i}^K-D_{3i}^{K-1},\Lambda_{3i}-\Lambda_{3i}^K\Big\rangle \\ & +\sum_{k=1}^K \mathbb{E}\Big[\frac{\|L_i\|^2}{2\gamma_i}\big\|Y_i^{k-1}-Y_i^{k-2}\big\|^2\Big], \end{cases}$$

where $\alpha_i=\alpha_{i1}+\alpha_{i2}$.

Proof. Applying Lemma 6 to (9), (10) and (11) respectively, for all $\Lambda_{1i}\in\mathbb{R}^{m_i\times q}$, $\Lambda_{2i}\in\mathbb{R}^{p\times q}$, $\Lambda_{3i}\in\mathbb{R}^{m\times q}$, $X_i\in\mathbb{R}^{p\times q}$ and $Y_i\in\mathbb{R}^{m\times q}$, we have

$$\mathbb{E}\Big\langle A_{vi}\hat{X}_i^k-\hat{Y}_i^{vi,k},\Lambda_{1i}-\Lambda_{1i}^k\Big\rangle\leqslant\frac{\alpha_i}{2}\mathbb{E}\Big[\|\Lambda_{1i}-\Lambda_{1i}^{k-1}\|^2-\|\Lambda_{1i}-\Lambda_{1i}^k\|^2-\|\Lambda_{1i}^{k-1}-\Lambda_{1i}^k\|^2\Big],\tag{26}$$

$$\mathbb{E}\Big\langle l_{i,j}\hat{X}_j^k,\Lambda_{2i}-\Lambda_{2i}^k\Big\rangle\leqslant\frac{\beta_i}{2}\mathbb{E}\Big[\|\Lambda_{2i}-\Lambda_{2i}^{k-1}\|^2-\|\Lambda_{2i}-\Lambda_{2i}^k\|^2-\|\Lambda_{2i}^{k-1}-\Lambda_{2i}^k\|^2\Big],\tag{27}$$

$$\mathbb{E}\Big\langle l_{i,j}\hat{Y}_j^k,\Lambda_{3i}-\Lambda_{3i}^k\Big\rangle\leqslant\frac{\gamma_i}{2}\mathbb{E}\Big[\|\Lambda_{3i}-\Lambda_{3i}^{k-1}\|^2-\|\Lambda_{3i}-\Lambda_{3i}^k\|^2-\|\Lambda_{3i}^{k-1}-\Lambda_{3i}^k\|^2\Big].\tag{28}$$

Summing Γ_{ri}^k from $k=1$ to $k=K$ yields

$$\begin{aligned} &\sum_{k=1}^K\mathbb{E}\Gamma_{1i}^k \\ =&\sum_{k=1}^K\mathbb{E}\Big\langle A_{vi}X_i^k-Y_i^{vi,k},\Lambda_{1i}-\Lambda_{1i}^k\Big\rangle+\sum_{k=1}^K\mathbb{E}\Big\langle A_{vi}\big(X_i^k-\hat{X}_i^k\big)-\big(Y_i^{vi,k}-\hat{Y}_i^{vi,k}\big),\Lambda_{1i}-\Lambda_{1i}^k\Big\rangle \\ \leqslant&\frac{\alpha_i}{2}\mathbb{E}\Big[\|\Lambda_{1i}-\Lambda_{1i}^0\|^2-\|\Lambda_{1i}-\Lambda_{1i}^K\|^2\Big]+\mathbb{E}\Big\langle A_{vi}\big(X_i^K-X_i^{K-1}\big)-\big(Y_i^{vi,K}-Y_i^{vi,K-1}\big),\Lambda_{1i}-\Lambda_{1i}^K\Big\rangle \\ &-\sum_{k=1}^K\mathbb{E}\Big[\frac{\alpha_i}{2}\|\Lambda_{1i}^{k-1}-\Lambda_{1i}^k\|^2-\Big\langle A_{vi}\big(X_i^{k-1}-X_i^{k-2}\big)-\big(Y_i^{vi,k-1}-Y_i^{vi,k-2}\big),\Lambda_{1i}^k-\Lambda_{1i}^{k-1}\Big\rangle\Big] \\ \leqslant&\frac{\alpha_i}{2}\mathbb{E}\Big[\|\Lambda_{1i}-\Lambda_{1i}^0\|^2-\|\Lambda_{1i}-\Lambda_{1i}^K\|^2\Big]+\mathbb{E}\Big\langle A_{vi}\big(X_i^K-X_i^{K-1}\big)-\big(Y_i^{vi,K}-Y_i^{vi,K-1}\big),\Lambda_{1i}-\Lambda_{1i}^K\Big\rangle \\ &+\sum_{k=1}^K\mathbb{E}\Big[\frac{\|A_i\|^2}{2\alpha_{i1}}\big\|X_i^{k-1}-X_i^{k-2}\big\|^2+\frac{1}{2\alpha_{i1}}\big\|Y_i^{k-1}-Y_i^{k-2}\big\|^2\Big], \end{aligned}\tag{29}$$

where the second inequality follows from the fact that $a\langle u,v\rangle-\frac{a^2}{2b}\|v\|^2\leqslant\frac{b}{2}\|u\|^2$, $\forall u,v\in\mathbb{R}^{p\times q}$. Similarly, we can also give the upper bounds of $\sum_{k=1}^K\mathbb{E}\Gamma_{2i}^k$ and $\sum_{k=1}^K\mathbb{E}\Gamma_{3i}^k$, which complete the proof. □

Then, we present estimates on $\mathbb{E}Q\Big[\bar{Z}_E^K,\bar{\Lambda}^K;Z_E,\Lambda\Big]$ defined in (16) together with stepsizes α_i, β_i, γ_i, μ_{1i} and μ_{2i}, which are used to provide convergence rate of AlgorithmDSMD.

Lemma 10. Let the iterates $\left\{Z_E^k,\Lambda^k\right\}$ be generated by Algorithm DSMD. With

$$\begin{cases}\mu_{1i}-\frac{\|A_i\|^2}{\alpha_{i1}}-\frac{\|L_i\|^2}{\beta_i}\geqslant 0\\ \mu_{2i}-\frac{1}{\alpha_{i1}}-\frac{\|L_i\|^2}{\gamma_i}\geqslant 0\end{cases}\tag{30}$$

and defining $\bar{Z}_E^K=\frac{1}{K}\sum_{k=1}^K Z_E^k$, $\bar{\Lambda}^K=\frac{1}{K}\sum_{k=1}^K\Lambda^k$, we have

$$\begin{aligned}&\mathbb{E}Q\Big[\bar{Z}_E^K,\bar{\Lambda}^K;Z_E,\Lambda\Big]-\frac{1}{K}\mathbb{E}\sum_{r=1}^3\sum_{i=1}^n\Big\langle\Lambda_{ri},\rho_{ri}^K\Big\rangle\\ \leqslant&\frac{1}{K}\sum_{i=1}^n\Big[\mu_{1i}\mathbb{E}B_i\Big(X_i^0,X_i\Big)+\sum_{i=1}^n\mu_{2i}\mathbb{E}B_i\Big(Y_i^0,Y_i\Big)+\frac{\alpha_i}{2}\mathbb{E}\|\Lambda_{1i}^0\|^2+\frac{\beta_i}{2}\mathbb{E}\|\Lambda_{2i}^0\|^2+\frac{\gamma_i}{2}\mathbb{E}\|\Lambda_{3i}^0\|^2\Big]\end{aligned}\tag{31}$$

where

$$\begin{cases}\rho_{1i}^K=D_{1i}^K-D_{1i}^{K-1}+\alpha_i\big(\Lambda_{1i}^K-\Lambda_{1i}^0\big)\\ \rho_{2i}^K=D_{2i}^K-D_{2i}^{K-1}+\beta_i\big(\Lambda_{2i}^K-\Lambda_{2i}^0\big)\\ \rho_{3i}^K=D_{3i}^K-D_{3i}^{K-1}+\alpha_i\big(\Lambda_{3i}^K-\Lambda_{3i}^0\big).\end{cases}$$

Proof. In view of the definition of $Q\Big[\bar{Z}_E,\bar{\Lambda};Z_E,\Lambda\Big]$, $D_{ri}^{(\cdot)}$ and $\Gamma_{ri}^{(\cdot)}$ in (16), (14), and (25), respectively, we can get

$$\begin{aligned}&\mathbb{E}Q\Big[Z_E^k,\Lambda^k;Z_E,\Lambda\Big]\\ =&\sum_{i=1}^n\mathbb{E}\Big[P_{1i}\Big(X_i^k,Y_i^k,\Lambda_i^k\Big)-P_{1i}\Big(X_i,Y_i^k,\Lambda_i^k\Big)\Big]+\sum_{i=1}^n\mathbb{E}\Big[P_{2i}\Big(X_i^k,Y_i^k,\Lambda_i^k\Big)-P_{1i}\Big(X_i^k,Y_i,\Lambda_i^k\Big)\Big]\\ &+\sum_{r=1}^3\sum_{i=1}^n\mathbb{E}\Gamma_{ri}^k.\end{aligned}\tag{32}$$

With Lemmas 8-9 and (30), $\sum_{k=1}^K Q\Big[Z_E^k,\Lambda^k;Z_E,\Lambda\Big]$ can be bounded as

$$\sum_{k=1}^K\mathbb{E}Q\Big[Z_E^k,\Lambda^k;Z_E,\Lambda\Big]\leqslant\sum_{s=1}^2\sum_{i=1}^n\mathbb{E}\Delta_{si},\tag{33}$$

where

$$\begin{aligned}\Delta_{1i}=&\mu_{1i}B_i\Big(X_i^0,X_i\Big)-\mu_{1i}B_i\Big(X_i^{K-1},X_i^K\Big)+\frac{\alpha_{i1}}{2}\|\Lambda_{1i}-\Lambda_{1i}^0\|^2-\frac{\alpha_{i1}}{2}\|\Lambda_{1i}-\Lambda_{1i}^K\|^2+\frac{\beta_i}{2}\|\Lambda_{2i}-\Lambda_{2i}^0\|^2\\ &-\frac{\beta_i}{2}\|\Lambda_{2i}-\Lambda_{2i}^K\|^2+\Big\langle A_{vi}\big(X_i^K-X_i^{K-1}\big),\Lambda_{1i}-\Lambda_{1i}^K\Big\rangle+\Big\langle\sum_{j=1}^n l_{i,j}\big(X_i^K-X_i^{K-1}\big),\Lambda_{2i}-\Lambda_{2i}^K\Big\rangle\\ \Delta_{2i}=&\mu_{2i}B_i\Big(Y_i^0,Y_i\Big)-\mu_{2i}B_i\Big(Y_i^{K-1},Y_i^K\Big)+\frac{\alpha_{i2}}{2}\|\Lambda_{1i}-\Lambda_{1i}^0\|^2-\frac{\alpha_{i2}}{2}\|\Lambda_{1i}-\Lambda_{1i}^K\|^2+\frac{\gamma_i}{2}\|\Lambda_{3i}-\Lambda_{3i}^0\|^2\\ &-\frac{\gamma_i}{2}\|\Lambda_{3i}-\Lambda_{3i}^K\|^2-\Big\langle Y_i^{vi,K}-Y_i^{vi,K-1},\Lambda_{1i}-\Lambda_{1i}^K\Big\rangle+\Big\langle\sum_{j=1}^n l_{i,j}\big(Y_i^K-Y_i^{K-1}\big),\Lambda_{3i}-\Lambda_{3i}^K\Big\rangle.\end{aligned}$$

Since $\|\Lambda_{si}-\Lambda_{si}^0\|^2-\|\Lambda_{si}-\Lambda_{si}^K\|^2=\|\Lambda_{si}^0\|^2-\|\Lambda_{si}^K\|^2-2\langle\Lambda_{si},\Lambda_{si}^0-\Lambda_{si}^K\rangle$, Δ_{si} can be modified as

$$\begin{aligned}\Delta_{1i}\leqslant&\mu_{1i}B_i\Big(X_i^0,X_i\Big)-\mu_{1i}B_i\Big(X_i^{K-1},X_i^K\Big)+\frac{\alpha_{i1}}{2}\|\Lambda_{1i}^0\|^2-\frac{\alpha_{i1}}{2}\|\Lambda_{1i}^K\|^2-\alpha_{i1}\Big\langle\Lambda_{1i}^0-\Lambda_{1i}^K,\Lambda_{1i}\Big\rangle\\&+\frac{\beta_i}{2}\|\Lambda_{2i}^0\|^2-\frac{\beta_i}{2}\|\Lambda_{2i}^K\|^2-\beta_i\Big\langle\Lambda_{2i}^0-\Lambda_{2i}^K,\Lambda_{2i}\Big\rangle+\Big\langle A_{vi}\big(X_i^K-X_i^{K-1}\big),\Lambda_{1i}-\Lambda_{1i}^K\Big\rangle\\&+\Big\langle\sum_{j=1}^n l_{i,j}\big(X_i^K-X_i^{K-1}\big),\Lambda_{2i}-\Lambda_{2i}^K\Big\rangle\\\leqslant&\mu_{1i}B_i\Big(X_i^0,X_i\Big)+\frac{\alpha_{i1}}{2}\|\Lambda_{1i}^0\|^2+\frac{\beta_i}{2}\|\Lambda_{2i}^0\|^2+\Big\langle A_{vi}\big(X_i^K-X_i^{K-1}\big)+\alpha_{i1}\big(\Lambda_{1i}^K-\Lambda_{1i}^0\big),\Lambda_{1i}\Big\rangle\\&+\Big\langle\sum_{j=1}^n l_{i,j}\big(X_i^K-X_i^{K-1}\big)+\beta_i\big(\Lambda_{2i}^K-\Lambda_{2i}^0\big),\Lambda_{2i}\Big\rangle,\end{aligned}\tag{34}$$

where the second inequality is based on $\mu_{1i}-\frac{\|A_i\|^2}{\alpha_{i1}}-\frac{\|L_i\|^2}{\beta_i}\geqslant 0$ and $a\langle X,Y\rangle-\frac{a^2}{2b}\leqslant\frac{b}{2}\|X\|^2$. Similarly, we have

$$\begin{aligned}\Delta_{2i}\leqslant&\mu_{2i}B_i\Big(Y_i^0,Y_i\Big)+\frac{\alpha_{i2}}{2}\|\Lambda_{1i}^0\|^2+\frac{\gamma_i}{2}\|\Lambda_{3i}^0\|^2+\Big\langle\big(Y_i^{vi,K-1}-Y_i^{vi,K}\big)+\alpha_{i2}\big(\Lambda_{1i}^K-\Lambda_{1i}^0\big),\Lambda_{1i}\Big\rangle\\&+\Big\langle\sum_{j=1}^n l_{i,j}\big(Y_i^K-Y_i^{K-1}\big)+\gamma_i\big(\Lambda_{3i}^K-\Lambda_{3i}^0\big),\Lambda_{3i}\Big\rangle.\end{aligned}\tag{35}$$

Combining (34) and (35), we obtain

$$\begin{aligned}&\sum_{k=1}^K\mathbb{E}Q\Big[Z_E^k,\Lambda^k;Z_E,\Lambda\Big]\\\leqslant&\sum_{i=1}^n\mathbb{E}\Big[\mu_{1i}B_i\Big(X_i^0,X_i\Big)+\mu_{2i}B_i\Big(Y_i^0,Y_i\Big)+\frac{\alpha_i}{2}\|\Lambda_{1i}^0\|^2+\frac{\beta_i}{2}\|\Lambda_{2i}^0\|^2+\frac{\gamma_i}{2}\|\Lambda_{3i}^0\|^2+\sum_{r=1}^3\Big\langle D_{ri}^K\\&-D_{ri}^{k-1},\Lambda_{ri}\Big\rangle+\Big\langle\alpha_i\big(\Lambda_{1i}^K-\Lambda_{1i}^0\big),\Lambda_{1i}\Big\rangle+\Big\langle\beta_i\big(\Lambda_{2i}^K-\Lambda_{2i}^0\big),\Lambda_{2i}\Big\rangle+\Big\langle\gamma_i\big(\Lambda_{3i}^K-\Lambda_{3i}^0\big),\Lambda_{3i}\Big\rangle\Big].\end{aligned}\tag{36}$$

Following the convexity of $Q\Big[Z_E^k,\Lambda^k;Z_E,\Lambda\Big]$ with respect to Z_E^k and Λ^k, it follows

$$\begin{aligned}&\mathbb{E}Q\Big[\bar Z_E^K,\bar\Lambda^K;Z_E,\Lambda\Big]-\frac{1}{K}\sum_{r=1}^3\sum_{i=1}^n\mathbb{E}\Big\langle\Lambda_{ri},\rho_{ri}^K\Big\rangle\\\leqslant&\frac{1}{K}\sum_{i=1}^n\mathbb{E}\Big[\mu_{1i}B_i\Big(X_i^0,X_i\Big)+\sum_{i=1}^n\mu_{2i}B_i\Big(Y_i^0,Y_i\Big)+\frac{\alpha_i}{2}\|\Lambda_{1i}^0\|^2+\frac{\beta_i}{2}\|\Lambda_{2i}^0\|^2+\frac{\gamma_i}{2}\|\Lambda_{3i}^0\|^2\Big]\end{aligned}\tag{37}$$

□

Next, we present estamates of $\mathbb{E}\|\Lambda_{1i}^*-\Lambda_{1i}^K\|^2$, $\mathbb{E}\|\Lambda_{2i}^*-\Lambda_{2i}^K\|^2$, $\mathbb{E}\|\Lambda_{3i}^*-\Lambda_{3i}^K\|^2$, $\mathbb{E}\|X_i^K-X_i^{K-1}\|^2$ and $\mathbb{E}\|Y_i^K-Y_i^{K-1}\|^2$.

Lemma 11. Let the iterates $\big\{Z_E^k,\Lambda^k\big\}$ be generated by Algorithm DSMD. Assuming that

$$\begin{cases}\frac{\alpha_i}{2}-\frac{\|A_i\|^2}{2\mu_{1i1}}-\frac{1}{2\mu_{2i1}}\geqslant 0\\\frac{\beta_i}{2}-\frac{\|L_i\|^2}{2\mu_{1i2}}\geqslant 0\\\frac{\gamma_i}{2}-\frac{\|L_i\|^2}{2\mu_{2i2}}\geqslant 0\end{cases}\tag{38}$$

with $\mu_{1i1} + \mu_{1i2} = \mu_{1i}$ and $\mu_{2i1} + \mu_{2i2} = \mu_{2i}$, we have

$$\begin{aligned} &\max \mathbb{E}\big\{\|\Lambda_{1i}^* - \Lambda_{1i}^K\|^2, \|\Lambda_{2i}^* - \Lambda_{2i}^K\|^2, \|\Lambda_{3i}^* - \Lambda_{3i}^K\|^2, \|X_i^K - X_i^{K-1}\|^2, \|Y_i^K - Y_i^{K-1}\|^2\big\} \\ \leqslant &\frac{1}{\sigma}\mathbb{E}\Big[\mu_{1i}B_i\Big(X_i^0, X_i^*\Big) + \mu_{2i}B_i\Big(Y_i^0, Y_i^*\Big) + \frac{\alpha_i}{2}\|\Lambda_{1i}^* - \Lambda_{1i}^0\|^2 + \frac{\beta_i}{2}\|\Lambda_{2i}^* - \Lambda_{2i}^0\|^2 + \frac{\gamma_i}{2}\|\Lambda_{3i}^* - \Lambda_{3i}^0\|^2\Big]. \end{aligned} \tag{39}$$

P r o o f. Recalling (32) and $Q\Big[Z_E^k, \Lambda^k; Z_E^*, \Lambda^*\Big] \geqslant 0$, it yields

$$\begin{aligned} &\mu_{1i}\mathbb{E}B_i\Big(X_i^{K-1}, X_i^K\Big) + \mu_{2i}B_i\mathbb{E}\Big(Y_i^{K-1}, Y_i^K\Big) \\ \leqslant &\mu_{1i}\mathbb{E}B_i\Big(X_i^0, X_i^*\Big) + \mu_{2i}\mathbb{E}B_i\Big(Y_i^0, Y_i^*\Big) + \frac{\alpha_i}{2}\mathbb{E}\|\Lambda_{1i}^* - \Lambda_{1i}^0\|^2 + \frac{\beta_i}{2}\mathbb{E}\|\Lambda_{2i}^* - \Lambda_{2i}^0\|^2 + \frac{\gamma_i}{2}\mathbb{E}\|\Lambda_{3i}^* - \Lambda_{3i}^0\|^2 \\ &+ \left[\frac{\|A_i\|^2}{2\alpha_{i1}} + \frac{\|L_i\|^2}{2\beta_i}\right]\mathbb{E}\|X_i^K - X_i^{K-1}\|^2 + \left[\frac{1}{2\alpha_{i2}} + \frac{\|L_i\|^2}{2\gamma_i}\right]\mathbb{E}\|Y_i^K - Y_i^{K-1}\|^2. \end{aligned} \tag{40}$$

Rerange terms in (40), we get

$$\begin{aligned} &\xi_{1i}\mathbb{E}\|X_i^K - X_i^{K-1}\|^2 + \xi_{2i}\mathbb{E}\|Y_i^K - Y_i^{K-1}\|^2 \\ \leqslant &\mu_{1i}\mathbb{E}B_i\Big(X_i^0, X_i^*\Big) + \mu_{2i}\mathbb{E}B_i\Big(Y_i^0, Y_i^*\Big) + \frac{\alpha_i}{2}\mathbb{E}\|\Lambda_{1i}^* - \Lambda_{1i}^0\|^2 + \frac{\beta_i}{2}\mathbb{E}\|\Lambda_{2i}^* - \Lambda_{2i}^0\|^2 + \frac{\gamma_i}{2}\mathbb{E}\|\Lambda_{3i}^* - \Lambda_{3i}^0\|^2, \end{aligned} \tag{41}$$

where $\xi_{1i} = \frac{\mu_{1i}}{2} - \frac{\|A_i\|^2}{2\alpha_{i1}} - \frac{\|L_i\|^2}{2\beta_i}$ and $\xi_{2i} = \frac{\mu_{2i}}{2} - \frac{1}{2\alpha_{i2}} - \frac{\|L_i\|^2}{2\gamma_i}$. Similarly, we have

$$\begin{aligned} &\sum_{r=1}^{3} \tau_{ri}\mathbb{E}\|\Lambda_{ri}^* - \Lambda_{ri}^K\|^2 \\ \leqslant &\mu_{1i}\mathbb{E}B_i\Big(X_i^0, X_i^*\Big) + \mu_{2i}\mathbb{E}B_i\Big(Y_i^0, Y_i^*\Big) + \frac{\alpha_i}{2}\mathbb{E}\|\Lambda_{1i}^* - \Lambda_{1i}^0\|^2 + \frac{\beta_i}{2}\mathbb{E}\|\Lambda_{2i}^* - \Lambda_{2i}^0\|^2 + \frac{\gamma_i}{2}\mathbb{E}\|\Lambda_{3i}^* - \Lambda_{3i}^0\|^2, \end{aligned} \tag{42}$$

where $\tau_{1i} = \frac{\alpha_i}{2} - \frac{\|A_i\|^2}{2\mu_{1i1}} - \frac{1}{2\mu_{2i1}}$, $\tau_{2i} = \frac{\beta_i}{2} - \frac{\|L_i\|^2}{2\mu_{1i2}}$ and $\tau_{3i} = \frac{\gamma_i}{2} - \frac{\|L_i\|^2}{2\mu_{2i2}}$. With (41) and (42) and defining σ as follows

$$\sigma = \min\big\{\tau_{1i}, \tau_{2i}, \tau_{3i}, \xi_{1i}, \xi_{2i}\big\}, \tag{43}$$

the proof is completed. □