DISTRIBUTED OPTIMIZATION OF QUADRATIC COSTS WITH A GROUP-SPARSITY REGULARIZATION VIA PDMM

Kangwei Hu, Danqi Jin, Wen Zhang, Jie Chen

Centre of Intelligent Acoustics and Immersive Communications
School of Marine Science and Technology, Northwestern Polytechinical University, China
kangweihu@mail.nwpu.edu.cn danqijin@mail.nwpu.edu.cn wen.zhang@nwpu.edu.cn dr.jie.chen@ieee.org

ABSTRACT

Structural sparsity is useful for variable and node selection in distributed networks. In this paper, we propose a distributed algorithm to solve the problem of a quadratic cost function with mixed $\ell_{1,2}$-norm regularization to promote the group-sparsity of the solution. By introducing virtual pair nodes to each actual node and by decomposing the cost function to each nodes, we obtain a distributed optimization problem on an extended graph model, which is further solved via the PDMM algorithm. Numerical simulation results illustrate the accurate convergence of the proposed algorithm to the centralized solution.

Index Terms— Distributed optimization, group-sparsity, $\ell_{1,2}$-norm regularization, primal-dual algorithm, PDMM.

1. INTRODUCTION

In recent years, distributed optimization has drawn increasing attentions with the advances and developments of sensor networks, cloud computing, big data and neural networks [1–3]. It endows networks with abilities of parallel computation, local data accessing and exchange, privacy protection, etc.

A variety of distributed optimization algorithms have been proposed in the literature. Typical algorithms and strategies include the dual-averaging algorithm [4], subgradient algorithm [5], and Gossip algorithm [6], consensus strategy [7–9], incremental strategy [10, 11], and diffusion strategy [12–15]. In recent years, the primal-dual type algorithms are also widely used for distributed optimization [16, 17]. Alternating Direction Methods of Multipliers (ADMM) is a typical algorithm of this class, and it has received much attention due to its wide range of applications [18]. By utilizing the Douglas-Rachford splitting [19] for augmented Lagrangian to approach the saddle point, ADMM provides an efficient solution for the distributed optimization problems. Alternatively, utilizing a similar Douglas-Rachford splitting for the so-called augmented primal-dual Lagrangian leads to Primal-Dual Method of Multipliers. In several scenarios, PDMM exhibits a faster convergence rate over ADMM, meanwhile provides a general framework for distributed optimization with synchronous and asynchronous updating schemes [20].

Several applications based on PDMM have been proposed, one typical example is on the design of distributed speech enhancement algorithms. A distributed implementation of the robust linearly constrained minimum variance (LCMV) beamformer is proposed in [21]. A distributed minimum variance distortionless response (MVDR) beamformer [22] and maximum signal to interference-plus-noise ratio (max-SINR) beamformer are proposed for an ad hoc microphone array [23]. The work [24] proposes a distributed MVDR with node selection, where a quadratic cost function regularized by the ℓ_1-norm is minimized via PDMM in a distributed manner.

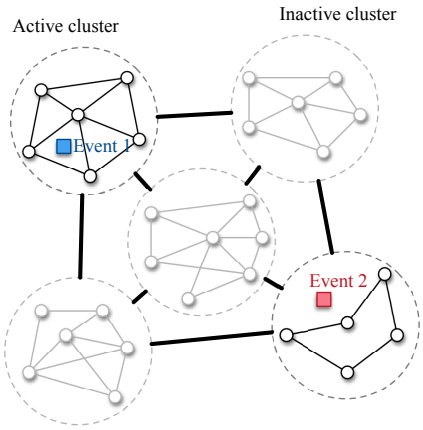
Inspired by [24], we further consider in this work the distributed optimization problem with the mixed $\ell_{1,2}$-norm regularization [25–27]. This regularization is used to encourage the parameter to have structural sparsity, so that nodes within a same cluster tend to be activated or deactivated simultaneously. By decomposing quadratic costs to each node and introducing virtual nodes for the term, we propose to solve this problem in a distributed manner via PDMM. Numerical simulation results illustrate the effectiveness and accurate convergence of the proposed algorithm.

2. PROBLEM FORMULATION AND SYSTEM MODEL

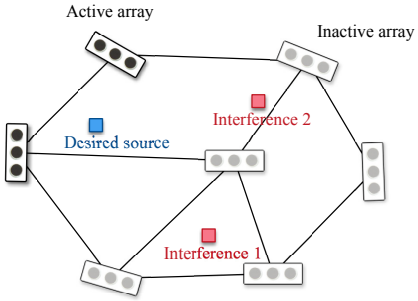
Notation. Normal font x denotes scalars. Boldface small letters $\boldsymbol{x}$ and capital letters $\boldsymbol{X}$ denote column vectors and matrices, respectively. The superscript $(\cdot)^\top$ denotes the transpose of vectors or matrices. Operators $\bigcup$ and $\bigcap$ calculate the union and intersection of two sets, respectively. Operator $\mathrm{col}\{\cdot\}$ stacks the column vectors entries on top of each other. Operator $\max\{\cdot,\cdot\}$ takes the maximum of its two arguments. $\mathcal{N}_k$ denotes the neighbors of node k, including k, with cardinality $|\mathcal{N}_k|$.

Without loss of generality, we focus on the following real-

This work was supported in part by National Natural Science Foundation of China (NSFC) under grants 61671382 and 61671380.



(a) When detecting local events, clusters of nodes near to these events are supposed to be activated as a whole.



(b) Microphone arrays near to the desired source are activated to perform the beamforming task, while others can be deactivated.

Fig. 1. Motivation of promoting the group-sparsity structure in networks.

valued optimization problem[1]:

$$\min_{\boldsymbol{x}} \; f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top \boldsymbol{R}\boldsymbol{x} - \boldsymbol{d}^\top \boldsymbol{x} + \alpha \left\|\boldsymbol{x}\right\|_{1,2}, \tag{1}$$

where $\boldsymbol{R} \in \mathbb{R}^{NM\times NM}$ is a symmetric, positive definite matrix, $\boldsymbol{d}$ and $\boldsymbol{x}$ are column vectors of length NM,

$$\boldsymbol{d} = [d_1, \, \cdots, \, d_{NM}]^\top \tag{2}$$

$$\boldsymbol{x} = [x_1, \, \cdots, \, x_{NM}]^\top, \tag{3}$$

and the mixed $\ell_{1,2}$-norm is used to promote the group-sparsity of the solution, with the regularization parameter $\alpha > 0$. The $\ell_{1,2}$-norm of a vector $\boldsymbol{x}$ is defined as

$$\left\|\boldsymbol{x}\right\|_{1,2} = \sum_{j=1}^{J} \left\|\boldsymbol{x}_{\mathcal{G}_j}\right\|_2, \tag{4}$$

where $\{\mathcal{G}_j\}_{j=1}^{J}$ are subsets of the whole index set $\mathcal{G} = \{1, 2, \ldots, NM\}$, satisfying:

$$\bigcup_{j=1}^{J} \mathcal{G}_j = \mathcal{G} \;\text{ and }\; \mathcal{G}_j \bigcap \mathcal{G}_l = \emptyset \;\text{ for } j \neq l.$$

Such group sparsity is desired when we intend to activate a cluster of nodes as a whole to detect some events in the network, while other clusters are deactivated, see Fig. 1(a). Another example is the distributed beamforming with microphone arrays. Each array is then considered as a node in the network, and it is supposed to activate some arrays instead of individually activating some microphones, see Fig. 1(b).

[1]For complex-valued $\boldsymbol{R}$, $\boldsymbol{d}$ and $\boldsymbol{x}$, by resorting to the mapping in [28] to transform quantities from C^{MN} to $\mathbb{R}^{2MN}$, we actually obtain an optimization problem in real domain.

For simplicity, we assume uniform groups with group size M, resulting in a group number $J = N$. We aim to solve (1) in a distributed manner. Now Consider a network of N nodes, with each node is associated with a node variable $\overline{\boldsymbol{x}}_k$, consisting of a group of M entries of $\boldsymbol{x}$. Without loss of generality, the NM entries of $\boldsymbol{x}$ are assigned in order, with elements $(k-1)M+1$ to kM being assigned to node k. Besides, the nodes are connected by edges according to network topology denoted by a undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of the nodes, $\mathcal{E}$ is the set of the edges, with cardinalities $V = |\mathcal{V}| = N$ and $E = |\mathcal{E}|$, respectively. If there exists an edge between nodes k and ℓ, we say $(k,\ell) \in \mathcal{E}$. The binary connection matrix of the network is denoted by $\boldsymbol{C}_{0,1}$, where the (k,ℓ)-th element is 1 if $(k,\ell) \in \mathcal{E}$, and 0 elsewise.

3. DISTRIBUTED SOLUTION BASED ON PDMM

Without ambiguity, we denote $f(\boldsymbol{x})$ in (1) by f for simplicity. To derive a distributed solution, we firstly decompose the cost function $f(\boldsymbol{x})$ to local costs attached to each node. Considering to restrict communications within intermediate neighboring nodes, and utilizing the decomposition techniques proposed in [24, 29] for the first two terms on the right-hand-side of (1), f is equivalently formulated as

$$\min_{\boldsymbol{x},\boldsymbol{x}_k,k\in\mathcal{V}} f = \sum_{k\in\mathcal{V}} \left(\frac{1}{2}\boldsymbol{x}_k^\top \boldsymbol{R}_k \boldsymbol{x}_k - \boldsymbol{d}_k^\top \boldsymbol{x}_k\right) + \alpha \left\|\boldsymbol{x}\right\|_{1,2}$$
$$\text{subject to} \quad \boldsymbol{A}_{k\ell}\boldsymbol{x}_k + \boldsymbol{A}_{\ell k}\boldsymbol{x}_\ell = \boldsymbol{0}, \quad \forall (k,\ell) \in \mathcal{E} \tag{5}$$

where $\boldsymbol{x}_k \in \mathbb{R}^{M\cdot|\mathcal{N}_k|}$ is the local estimate of $\boldsymbol{x}$ at node k over its neighbor $\mathcal{N}_k$, i.e., $\boldsymbol{x}_k = \mathrm{col}\{\overline{\boldsymbol{x}}_\ell\}_{\ell\in\mathcal{N}_k}$, $\boldsymbol{d}_k \in \mathbb{R}^{M\cdot|\mathcal{N}_k|}$ is a vector of all zeros except M elements equals to d_m with index m ranging from $(k-1)M+1$ to kM, corresponding

to elements of node k itself, i.e.

$$\boldsymbol{d}_k = [0 \ \cdots \ 0 \ \ d_{(k-1)M+1} \ \cdots \ d_{kM} \ \ 0 \ \cdots \ 0]^\top, \quad (6)$$

and $\boldsymbol{R}_k \in \mathbb{R}^{M\cdot|\mathcal{N}_k|\times M\cdot|\mathcal{N}_k|}$ is a local matrix defined by

$$\boldsymbol{R}_k = (\boldsymbol{C}_{0,1}^{2\dagger} \otimes \boldsymbol{E}_M) \circ \boldsymbol{R}, \quad (7)$$

with entries only belonging to $\mathcal{N}_k$ preserved in order so as to have dimension $M|\mathcal{N}_k| \times M|\mathcal{N}_k|$. $\boldsymbol{C}_{0,1}^{2\dagger}$ denotes the element-wise inverse of the square of the network connection matrix $\boldsymbol{C}_{0,1}$, $\boldsymbol{E}_M$ is an $M \times M$ matrix with all elements equal to one, symbols $\otimes$ and $\circ$ denote Kronecker product and Hadamard product, respectively, and $\boldsymbol{A}_{k\ell}$ is a matrix with elements $\{1, 0, -1\}$ to keep the corresponding entries (node variable) of $\boldsymbol{x}_k$ and $\boldsymbol{x}_\ell$ belonging to same nodes to be consistent so as to keep consensus.

In order to facilitate distributed processing, the regularization term is specifically treated by introducing virtual nodes for each actual node. Considering the expression of the regularization term, we assume that each node k has a virtual pair node with index $k + N$, connecting to itself only by a virtual edge. The set of virtual nodes is denoted by $\mathcal{V}_v$, with node indices from $N + 1$ to $2N$, and there are N virtual edges in total denoted by set $\mathcal{E}_v$. Since the $\ell_{1,2}$-norm is separable over N groups, we may assign for each virtual node $N+k$ the vector $\boldsymbol{x}_{N+k} = [x_{(k-1)M+1} \ \cdots \ x_{kM}]^\top$ and its ℓ_2-norm as their node variables and corresponding cost functions, along with the consensus constraints to keep the estimate of node variables at a virtual node and the corresponding actual node to be consistent, that is, $\boldsymbol{x}_{N+k} + \boldsymbol{D}_{\ell(N+k)}\boldsymbol{x}_\ell = \boldsymbol{0}$ with $\boldsymbol{D}_{\ell(N+k)}$ consisting of $\{1, 0, -1\}$. Therefore, (5) can be written equivalently as

$$\min_{\boldsymbol{x}_k, \boldsymbol{x}_j, k\in\mathcal{V}, j\in\mathcal{V}_v} f = \sum_{k\in\mathcal{V}} \left(\frac{1}{2}\boldsymbol{x}_k^\top \boldsymbol{R}_k \boldsymbol{x}_k - \boldsymbol{d}_k^\top \boldsymbol{x}_k \right) + \sum_{j\in\mathcal{V}_v} \alpha \|\boldsymbol{x}_j\|_2$$
$$\text{subject to} \quad \boldsymbol{A}_{k\ell}\boldsymbol{x}_k + \boldsymbol{A}_{\ell k}\boldsymbol{x}_\ell = \boldsymbol{0}, \quad \forall (k,\ell) \in \mathcal{E}$$
$$\boldsymbol{x}_j + \boldsymbol{D}_{\ell j}\boldsymbol{x}_\ell = \boldsymbol{0}, \quad \forall (j,\ell) \in \mathcal{E}_v. \quad (8)$$

By introducing virtual nodes and virtual edges, we have an extended graph model $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$, with $\mathcal{V}' = \mathcal{V} \bigcup \mathcal{V}_v$ and $\mathcal{E}' = \mathcal{E} \bigcup \mathcal{E}_v$, thus we can rewrite (8) in a more compact form via the extended graph model $\mathcal{G}'$ as

$$\min_{\boldsymbol{x}_k, k\in\mathcal{V}'} f = \sum_{k\in\mathcal{V}'} f_k(\boldsymbol{x}_k)$$
$$\text{subject to} \quad \boldsymbol{B}_{k\ell}\boldsymbol{x}_k + \boldsymbol{B}_{\ell k}\boldsymbol{x}_\ell = \boldsymbol{0}, \quad \forall (k,\ell) \in \mathcal{E}' \quad (9)$$

where

$$f_k(\boldsymbol{x}_k) = \begin{cases} \frac{1}{2}\boldsymbol{x}_k^\top \boldsymbol{R}_k \boldsymbol{x}_k - \boldsymbol{d}_k^\top \boldsymbol{x}_k & \text{for} \quad k \in \mathcal{V} \\ \alpha \|\boldsymbol{x}_k\|_2 & \text{for} \quad k \in \mathcal{V}_v, \end{cases} \quad (10)$$

and matrices $\boldsymbol{A}_{k\ell}$ and $\boldsymbol{D}_{k\ell}$ are replaced by $\boldsymbol{B}_{k\ell}$ correspondingly.

After that, we propose to solve problem (9) over the extended graph model $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$ via PDMM. We make no distinction between virtual nodes and actual nodes in graph $\mathcal{G}'$, and $\mathcal{N}_k$ includes virtual node $N + k$. By introducing for each edge $(k, \ell) \in \mathcal{E}'$ two auxiliary node-dependent variables $\boldsymbol{\lambda}_{k|\ell}$ and $\boldsymbol{\lambda}_{\ell|k}$, one for each node k and ℓ, the augmented primal-dual Lagrangian of PDMM is written as

$$L_{\mathcal{P}}(\boldsymbol{x}, \boldsymbol{\lambda}) = \sum_{k\in\mathcal{V}'} \Big[f_k(\boldsymbol{x}_k) - \sum_{\ell\in\mathcal{N}_k} \boldsymbol{\lambda}_{\ell|k}^\top (\boldsymbol{B}_{k\ell}\boldsymbol{x}_k) - f_k^*(\boldsymbol{B}_k^\top \boldsymbol{\lambda}_k) \Big]$$
$$+ \sum_{(k,\ell)\in\mathcal{E}'} \frac{1}{2}\|\boldsymbol{B}_{k\ell}\boldsymbol{x}_k + \boldsymbol{B}_{\ell k}\boldsymbol{x}_\ell\|^2 - \sum_{(k,\ell)\in\mathcal{E}'} \frac{1}{2}\|\boldsymbol{\lambda}_{k|\ell} - \boldsymbol{\lambda}_{\ell|k}\|^2 \quad (11)$$

where f_k^* is the convex conjugate function of f_k [30], $\boldsymbol{\lambda}_k$ is obtained by vertically concatenating all $\boldsymbol{\lambda}_{k|\ell}$ for $\ell \in \mathcal{N}_k$, and $\boldsymbol{B}_k^\top$ is obtained by horizontally concatenating all $\boldsymbol{B}_{k\ell}^\top$ for $\ell \in \mathcal{N}_k$. Utilizing the updating scheme of PDMM to approach the saddle point of $L_{\mathcal{P}}(\boldsymbol{x}, \boldsymbol{\lambda})$, and resorting to proximity operator [31] to derive a closed-form expression, the updating equations of $\boldsymbol{x}_k$ and $\boldsymbol{\lambda}_{k|\ell}$ are given as follows (with superscript i denoting iteration index)

$$\boldsymbol{x}_k^{i+1} = \begin{cases} \big(\boldsymbol{R}_k + \sum_{\ell\in\mathcal{N}_k} \boldsymbol{B}_{k\ell}^\top \boldsymbol{B}_{k\ell}\big)^{-1} \times \\ \big(\boldsymbol{d}_k + \sum_{\ell\in\mathcal{N}_k} \boldsymbol{B}_{k\ell}^\top \boldsymbol{\lambda}_{\ell|k}^i - \sum_{\ell\in\mathcal{N}_k} \boldsymbol{B}_{k\ell}^\top \boldsymbol{B}_{k\ell} \boldsymbol{x}_\ell^i\big) \\ \qquad\qquad\qquad\qquad\qquad \text{for} \quad k \in \mathcal{V} \\ \max\{0, \ (1 - \alpha/\|\boldsymbol{p}^i\|_2)\} \cdot \boldsymbol{p}^i \qquad \text{for} \quad k \in \mathcal{V}_v \end{cases} \quad (12)$$

with $\boldsymbol{p}^i = \boldsymbol{\lambda}_{\ell|k}^i - \boldsymbol{B}_{\ell k}\boldsymbol{x}_\ell^i$, and

$$\boldsymbol{\lambda}_{k|\ell}^{i+1} = \boldsymbol{\lambda}_{\ell|k}^i - (\boldsymbol{B}_{\ell k}\boldsymbol{x}_\ell^i + \boldsymbol{B}_{k\ell}\boldsymbol{x}_k^{i+1}) \quad \text{for} \quad k \in \mathcal{V}' \quad (13)$$

with all $(k, \ell) \in \mathcal{E}'$.

Finally, a fusion step is necessary to collect the estimate of node variables at each actual node to obtain the final result. However, this procedure is performed only once. Besides, since an actual node k and the corresponding virtual node $k + N$ are the same node physically, when used in asynchronous updating scheme, the update of the actual node k is followed by the update of the corresponding virtual node $k + N$ immediately. In addition, when there is only one element in each group, i.e., $M = 1$, the problem with group-sparse regularization reduces to the case with the sparse regularization derived in [24]. The complete algorithm is summarized in Algorithm 1.

4. SIMULATION RESULTS

Now we present simulation results to validate the proposed algorithm over a distributed network. We generate a network consisting of $N = 64$ nodes with node variables of dimension $M = 2$, and the nodes are connected according to an

Algorithm 1 The proposed algorithm.

Input: The matrix $\boldsymbol{R}$ and column vector $\boldsymbol{d}$

Output: Group-sparse vector $\boldsymbol{x}$

1: Set up the graph, decompose the cost function.
2: Initialize $i := 0$, $\boldsymbol{x}_k$ and $\boldsymbol{\lambda}_k$
3: **while** not converged **do**
4: **for** $\forall k \in \mathcal{V}'$ **do**
5: Update $\boldsymbol{x}_k$ by

$$\boldsymbol{x}_k^{i+1} = \begin{cases} \boldsymbol{S}_k^{-1}\boldsymbol{t}_k^i, & \text{for} \quad k \in \mathcal{V}. \\ \max\{0,\ (1-\alpha/\|\boldsymbol{p}^i\|_2)\} \cdot \boldsymbol{p}^i, & \text{for} \quad k \in \mathcal{V}_v. \end{cases}$$

where

$$\boldsymbol{S}_k = \boldsymbol{R}_k + \sum_{\ell \in \mathcal{N}_k} \boldsymbol{B}_{k\ell}^\top \boldsymbol{B}_{k\ell},$$

$$\boldsymbol{t}_k^i = \boldsymbol{d}_k + \sum_{\ell \in \mathcal{N}_k} \boldsymbol{B}_{k\ell}^\top \boldsymbol{\lambda}_{\ell|k}^i - \sum_{\ell \in \mathcal{N}_k} \boldsymbol{B}_{k\ell}^\top \boldsymbol{B}_{k\ell} \boldsymbol{x}_\ell^i,$$

$$\boldsymbol{p}^i = \boldsymbol{\lambda}_{\ell|k}^i - \boldsymbol{B}_{\ell k} \boldsymbol{x}_\ell^i.$$

6: Update $\boldsymbol{\lambda}_k$ by

$$\boldsymbol{\lambda}_{k|\ell}^{i+1} = \boldsymbol{\lambda}_{\ell|k}^i - (\boldsymbol{B}_{\ell k}\boldsymbol{x}_\ell^i + \boldsymbol{B}_{k\ell}\boldsymbol{x}_k^{i+1}) \quad \text{for} \quad k \in \mathcal{V}'.$$

7: **end for**
8: Check stop criterion
9: $i \leftarrow i+1$
10: **end while**
11: Fuse $\boldsymbol{x}_k$

arbitrary manner. Thus the quantities $\boldsymbol{R}$ and $\boldsymbol{d}$ used in the simulation are of dimension 128×128 and 128×1, respectively, and the group size is 2. Besides, to better illustrate the degree of group-sparsity varying with increased regularization parameter α, we set first the positive definite matrix $\boldsymbol{R}$ and the optimal solution $\boldsymbol{x}^\star$ of (1) without a regularization term, then obtained vector $\boldsymbol{d}$ via $\boldsymbol{d} = \boldsymbol{R}\boldsymbol{x}^\star$, and the entries of $\boldsymbol{x}^\star$ were chosen to grow slowly along its dimension. And the nodes with non-zero valued node variables were named as active nodes.

In the first experiment, the obtained distributed solution $\boldsymbol{x}_{\rm d}^i$ at each iteration i was compared with the centralized solution $\boldsymbol{x}_{\rm c}$, which is the solution of problem (1) computed via the proximal gradient method after 100 iterations. Fig. 2 shows the iteration of $\text{Error}(i) = \|\boldsymbol{x}_{\rm d}^i - \boldsymbol{x}_{\rm c}\|^2$ over time instant i with the regularization parameter $\alpha = 0, 3, 6, 9$, respectively, and Fig. 3 illustrates the iterations of the corresponding cost values (1) after logarithm transform. Both Fig. 2 and Fig. 3 illustrate the accurate convergence of the distributed solution

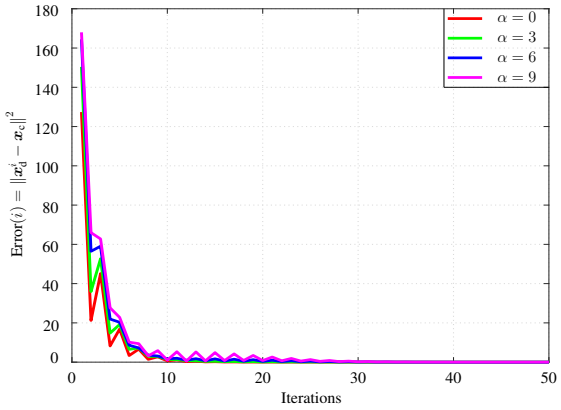


Fig. 2. Convergence of the distributed solution $\boldsymbol{x}_{\rm d}^i$ towards the centralized solution $\boldsymbol{x}_{\rm c}$ over iterations with the regularization parameter $\alpha = 0, 3, 6, 9$, respectively.

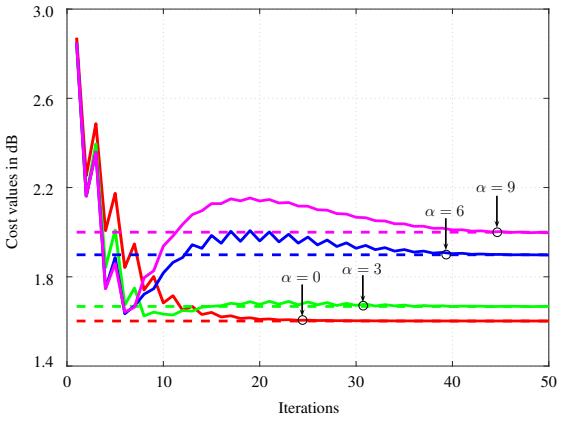


Fig. 3. Convergence of the cost value for distributed solution (full line) towards that of the centralized solution (dotted line, obtained after 100 iterations via the proximal gradient method) with the regularization parameter $\alpha = 0, 3, 6, 9$, respectively.

to the centralized solution. Besides, for different regularization parameters, the number of iterations required for the distributed solution to converge to the centralized solution within the precision range of 10^{-2} are given in Fig. 4. It can be observed that the number of iterations grows with the increasing of regularization parameter α in general.

In the second experiment, we examine the influence of the regularization parameter α to the degree of the group-sparsity in the distributed solution $\boldsymbol{x}_{\rm d}$ and the resulting bias ΔJ from the minimal cost $J_{\min} = \frac{1}{2}\boldsymbol{x}^{\star\top}\boldsymbol{R}\boldsymbol{x}^\star - \boldsymbol{d}^\top\boldsymbol{x}^\star$, where the bias is defined as $\Delta J = (\frac{1}{2}\boldsymbol{x}_{\rm d}^\top\boldsymbol{R}\boldsymbol{x}_{\rm d} - \boldsymbol{d}^\top\boldsymbol{x}_{\rm d}) - J_{\min}$. The results are shown in Fig. 5, we can see that the number of active nodes decreases with the increase of regularization parameter α, which means the degree of group-sparsity grows, while the resulting bias ΔJ becomes larger. Besides, for a sufficiently

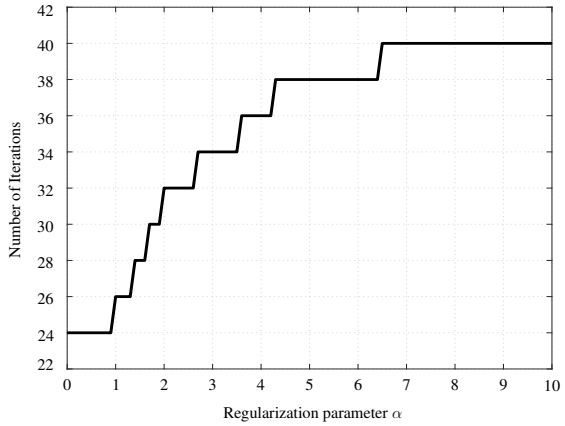


Fig. 4. The number of iterations required for the distributed solution $\boldsymbol{x}_{\mathrm{d}}$ to converge to the centralized solution $\boldsymbol{x}_{\mathrm{c}}$ within the precision range of 10^{-2} for different regularization parameters.

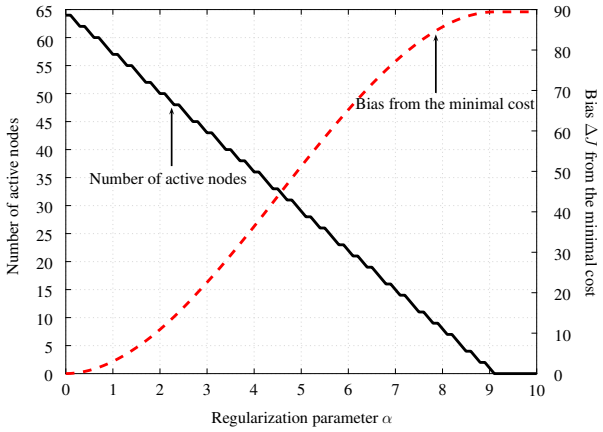


Fig. 5. The number of active nodes (black, full line) and the bias ΔJ from the minimal cost (red, dotted line) for different regularization parameters.

large regularization parameter, the distributed solution $\boldsymbol{x}_{\mathrm{d}}$ becomes an all-zero vector. This implies that the proposed algorithm can achieve the desired degree of group-sparsity as long as a proper regularization parameter is chosen.

5. CONCLUSIONS

In this paper, we proposed a distributed algorithm to solve the problem of a quadratic cost function with $\ell_{1,2}$-norm regularization. By introducing virtual pair nodes to each actual node and decomposing the cost function, we obtained an extended graph model and solved the optimization problem on the graph via PDMM. Numerical simulation results showed the accurate convergence of the proposed algorithm to the centralized solution. In future works, the proposed algorithm will be used to design distributed MVDR beamformers with the group-sparsity property.

6. REFERENCES

[1] Valmir C. Barbosa, *Massively Parallel Models of Computation: Distributed Parallel Processing in Artificial Intelligence and Optimisation*, Ellis Horwood, Upper Saddle River, NJ, USA, 1993.

[2] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, A. Senior, P. Tucker, K. Yang, Q. V. Le, et al., "Large scale distributed deep networks," in *Advances in neural information processing systems*, 2012, pp. 1223–1231.

[3] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," in *Proceedings of the 3rd international symposium on Information processing in sensor networks*, 2004, pp. 20–27.

[4] J. C. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: Convergence analysis and network scaling," *Trans. Autom. Control*, vol. 57, no. 3, pp. 592–606, Mar. 2012.

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

[6] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing," *Proc. of the IEEE*, vol. 98, no. 11, pp. 1847–1864, Nov. 2010.

[7] P. Braca, S. Marano, and V. Matta, "Enforcing consensus while monitoring the environment in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3375–3380, Jul. 2008.

[8] S. Kar and J. M. F. Moura, "Distributed consensus algorithms in sensor networks: Link failures and channel noise," *IEEE Trans. Signal Process.*, vol. 57, no. 1, pp. 355–369, Jan. 2009.

[9] S. Kar and J. M. F. Moura, "Distributed consensus algorithms in sensor networks: Quantized data and random link failures," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1383–1400, Mar. 2010.

[10] C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans. Signal Process.*, vol. 55, no. 8, pp. 4064–4077, Aug. 2007.

[11] F. S. Cattivelli and A. H. Sayed, "Analysis of spatial and incremental lms processing for distributed estimation," *IEEE Trans. Signal Process.*, vol. 59, no. 4, pp. 1465–1480, Apr. 2011.

[12] A. H. Sayed, "Diffusion adaptation over networks," in *Academic Press Library in Signal Processing*, vol. 3, pp. 323–453. Elsevier, 2014.

[13] J. Chen and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4289–4305, Aug. 2012.

[14] J. Chen, C. Richard, and A. H. Sayed, "Multitask diffusion adaptation over networks," *IEEE Trans. Signal Process.*, vol. 62, no. 16, pp. 4129–4144, Aug. 2014.

[15] J. Chen, C. Richard, and A. H. Sayed, "Diffusion LMS over multitask networks," *IEEE Trans. Signal Process.*, vol. 63, no. 11, pp. 2733–2748, Jun. 2015.

[16] D. Yuan, D. W. C. Ho, and S. Xu, "Regularized primal–dual subgradient method for distributed constrained optimization," *IEEE transactions on cybernetics*, vol. 46, no. 9, pp. 2109–2118, 2016.

[17] M. Badiei and N. Li, "Distributed regularized primal-dual method," in *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2016, pp. 540–544.

[18] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[19] J. Douglas and H. H. Rachford, "On the numerical solution of heat conduction problems in two and three space variables," *Transactions of the American mathematical Society*, vol. 82, no. 2, pp. 421–439, 1956.

[20] G. Zhang and R. Heusdens, "Distributed optimization using the primal-dual method of multipliers," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 4, no. 1, pp. 173–187, 2018.

[21] T. Sherson, W. B. Kleijn, and R. Heusdens, "A distributed algorithm for robust LCMV beamforming," in *proceedings of the IEEE ICASSP*, 2016, pp. 101–105.

[22] V. M. Tavakoli, J. R. Jensen, R. Heusdens, J. Benesty, and M. G. Christensen, "Ad hoc microphone array beamforming using the primal-dual method of multipliers," in *proceedings of the IEEE EUSIPCO*, 2016, pp. 1088–1092.

[23] V. M. Tavakoli, J. R. Jensen, R. Heusdens, J. Benesty, and M. G. Christensen, "Distributed max-SINR speech enhancement with ad hoc microphone arrays," in *proceedings of the IEEE ICASSP*, 2017, pp. 151–155.

[24] M. O'Connor, W. B. Kleijn, and T. Abhayapala, "Distributed sparse MVDR beamforming using the bi-alternating direction method of multipliers," in *proceedings of the IEEE ICASSP*, 2016, pp. 106–110.

[25] J. Huang and T. Zhang, "The benefit of group sparsity," *The Annals of Statistics*, vol. 38, no. 4, pp. 1978–2004, Aug. 2010.

[26] Y. Chen, Y. Gu, and A. O. Hero, "Regularized least-mean-square algorithms," 2010, Arxiv preprint arXiv:1012.5066.

[27] D. Jin, J. Chen, C. Richard, and J. Chen, "Adaptive parameters adjustment for group reweighted zero-attracting LMS," in *proceedings of the IEEE ICASSP*, Apr. 2018, pp. 4294–4298.

[28] J. K. Au, *An ab initio approach to the inverse problem-based design of photonic bandgap devices*, Ph.D. thesis, California Institute of Technology, 2007.

[29] S. L. Lauritzen, *Graphical models*, vol. 17, Clarendon Press, 1996.

[30] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge university press, 2004.

[31] N. Parikh, S. Boyd, et al., "Proximal algorithms," *Foundations and Trends® in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.