Signal Processing 153 (2018) 24–33

Contents lists available at ScienceDirect

Signal Processing

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

Robust decentralized dynamic optimization at presence of malfunctioning agents☆

Wei Xu[a], Zhengqing Li[a], Qing Ling[b,*]

[a] Department of Automation, University of Science and Technology of China, China
[b] School of Data and Computer Science, Sun Yat-Sen University, China

ARTICLE INFO

Article history:
Received 16 January 2018
Revised 27 June 2018
Accepted 30 June 2018
Available online 2 July 2018

Keywords:
Decentralized optimization
Dynamic optimization
Robust optimization

ABSTRACT

This paper considers the problem of tracking a network-wide solution that dynamically minimizes the summation of time-varying local cost functions of network agents, when some of the agents are malfunctioning. The malfunctioning agents broadcast faulty values to their neighbors, and lead the optimization process to a wrong direction. To mitigate the influence of the malfunctioning agents, we propose a total variation (TV) norm regularized formulation that drives the local variables of the regular agents to be close, while allows them to be different with the faulty values broadcast by the malfunctioning agents. We give a sufficient condition under which consensus of the regular agents is guaranteed, and bound the gap between the consensual solution and the optimal solution we pursue as if the malfunctioning agents do not exist. A fully decentralized subgradient algorithm is proposed to solve the TV norm regularized problem in a dynamic manner. At every time, every regular agent only needs one subgradient evaluation of its current local cost function, in addition to combining messages received from neighboring regular and malfunctioning agents. The tracking error is proved to be bounded, given that variation of the optimal solution is bounded. Numerical experiments demonstrate the robust tracking performance of the proposed algorithm at presence of the malfunctioning agents.

1. Introduction

Consider an undirected network consisting of n agents, which at time k try to cooperatively solve a decentralized dynamic optimization problem

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

Here $f_i^k : \mathbb{R}^p \to \mathbb{R}$ is a convex and differentiable local cost function only available to agent i at time k and $\tilde{x}^k \in \mathbb{R}^p$ is the common optimization variable to all agents. At time k, every agent is allowed to exchange its current local iterate with network neighbors, followed by local computation so as to track the dynamic optimal solution. The purpose of this paper is to develop a robust decentralized dynamic optimization algorithm that solves (1) at presence of malfunctioning agents. By malfunctioning agents, we mean those who, instead of transmitting local iterates to neighbors, send wrong values (for example, faulty constants or random variables) due to failures of communication or computation units.

Decentralized dynamic optimization problems in the form of (1) are popular in multi-agent networks with time-varying tasks [2–5]. Examples include adaptive filtering and estimation in a wireless sensor network [6–8], target tracking using a group of robots [9–11], dynamic resource allocation over a communication network [12–14], voltage control of a power network [15,16], to name a few. Existing algorithms to solve (1) are (sub)gradient methods [8,15], mirror descent method [5], alternating direction method of multipliers [2,14], as well as gradient, Newton, and interior point methods based on the prediction-correction scheme [3,4].

Nevertheless, most of the existing works assume that the agents faithfully follow prescribed optimization protocols: accessing dynamic local cost functions, exchanging local iterates, and performing local computations. This assumption does not always hold true since some of the agents might be malfunctioning in practice – some may send malicious information to their neighbors so as to deliberately guide the optimization process to a wrong direction that they expect to reach, whilst some may send faulty values to

☆ This work is supported by the China National Science Foundation under Grant 61573331 and the Anhui Provincial Natural Science Foundation under Grant 1608085QF130. Part of this paper has been accepted to the 43th International Conference on Acoustics, Speech, and Signal Processing, Calgary, Canada, April 15–20, 2018 [1].

* Corresponding author.
E-mail address: lingqing556@mail.sysu.edu.cn (Q. Ling).

https://doi.org/10.1016/j.sigpro.2018.06.024

their neighbors, not deliberately but due to failures of communication or computation units. This paper focuses on mitigating the impact of malfunctioning agents in decentralized dynamic optimization.

The impact of malfunctioning agents has been analyzed in the context of average consensus over a social network [17–19]. It is shown that the malfunctioning agents shall bias the network opinions from the consensual state of the regular agents [17], and the locations of the malfunctioning agents are critical to evolution of the network opinions [18]. Decentralized detection and localization methods are proposed in [19] to identify the malfunctioning agents. To the best of our knowledge, there is no existing work that considers the influence of the malfunctioning agents on the tracking performance of decentralized dynamic optimization.

Our work is tightly related to [20], whose goal is decentralized *static* optimization at presence of the malfunctioning agents. Different from the *dynamic* case studied in this paper, Ben-Ameur et al. [20] assumes that the local cost functions f_i^k are invariant across time k. To handle the faulty values broadcast by the malfunctioning agents, the total variation (TV) norm of the vector that stacks all the local variables is penalized. Through minimizing the summation of the local cost functions and the TV norm, most local variables (from the regular agents) are able to reach consensus and those outliers (from the malfunctioning agents) are tolerated. A subgradient method is proposed to solve this robust decentralized *static* optimization problem. Our work also adopts the TV norm penalty to handle the malfunctioning agents and a subgradient algorithm as the optimization tool, but extends their applications to the *dynamic* regime. We give a sufficient condition under which consensus of the regular agents is guaranteed, and also give an upper bound on the tracking error of the regular agents. These results are different to those developed for the *static* case in [20] due to the *dynamic* environment, and provide theoretical guarantees to the tracking performance of the subgradient method at presence of the malfunctioning agents.

Another related work is [21], which considers decentralized *stochastic* optimization. Instead of tracking a dynamic optimal solution, Koppel et al. [21] minimizes the summation of the local cost functions f_i^k across all nodes i and all times k. Therefore, the local iterates are expected to reach a steady-state consensual solution, given that the stochastic noise of the local cost functions is bounded. To allow for data heterogeneity across the network, Koppel et al. [21] introduces proximity constraints such that neighboring local variables are close enough, but not necessarily consensual. Though not explicitly claimed in [21], this approach is also able to alleviate the influence of the malfunctioning agents. A saddle point method is proposed to solve this constrained stochastic optimization problem. Our work is different from [21] in terms of problem setting (dynamic versus stochastic), mathematical formulation (TV norm penalty versus proximity constraints), and algorithm design (subgradient versus saddle point).

The main contributions of this paper are as follows.

1. We formulate a TV norm regularized problem, which is robust to presence of the malfunctioning agents (Section 2). We give a sufficient condition under which consensus of the regular agents is guaranteed, and bound the gap between the consensual solution and the optimal solution we pursue as if the malfunctioning agents do not exist (Section 3.2).
2. We propose a fully decentralized subgradient algorithm to solve the TV norm regularized problem in a dynamic manner. At every time, every regular agent only needs one subgradient evaluation, in addition to combining messages from neighboring regular and malfunctioning agents (Section 2). We prove that the tracking error is bounded, given that the variation of the optimal solution is bounded (Section 3.3).
3. We provide extensive numerical experiments, demonstrating the robust tracking performance of the proposed algorithm at presence of the malfunctioning agents (Section 4).

2. Problem formulation and algorithm design

Let us consider a connected undirected network of n agents $\mathcal{V} = \{1, \cdots, n\}$ with $n = |\mathcal{V}|$, and a set of edges $\mathcal{A}$. If an edge $(i, j) \in \mathcal{A}$, then agents i and j are neighbors, and can communicate with each other. We denote the set of agent i's neighbors as $\mathcal{N}_i$. The agents aim at solving the decentralized dynamic optimization problem in the form of (1). We assume that the network is synchronized, and at time k every agent i strictly conforms to the following protocol:

Step 1. Accessing local cost function f_i^k.

Step 2. Computing local iterate $x_i^k \in \mathbb{R}^p$.

Step 3. Broadcasting local iterate x_i^k to neighbors $j \in \mathcal{N}_i$.

However, some of the agents in the network are malfunctioning, meaning that they broadcast faulty values other than local iterates. To be specific, denote $\mathcal{M}$ as the set of malfunctioning agents and $\mathcal{R} := \mathcal{V} \backslash \mathcal{M}$ as the set of regular agents. Define $r := |\mathcal{R}|$ and $m := |\mathcal{M}|$. The subset of edges connecting the regular agents in $\mathcal{V}$ is denoted by $\mathcal{E} \subseteq \mathcal{A}$. At time k, malfunctioning agent $i \in \mathcal{M}$ broadcasts a variable $z_i^k \in \mathbb{R}^p$, instead of x_i^k, to its neighbors $j \in \mathcal{N}_i$. The faulty value may come from deliberate malicious attack, failure of the computation unit, or breakdown of the communication unit. Different from [17–20] that assume the faulty values are constant across time k, we also allow that they are time-varying (for example, random variables or values generated from certain functions of time). Although identifying the malfunctioning agents is possible in decentralized *static* optimization [19], their detection and localization are much more challenging for the *dynamic* task, especially when the faulty values are time-varying.

Observe that at presence of the malfunctioning agents, it is meaningless to solve (1), which minimizes the summation of all agents' local cost functions. For example, in multi-robot tracking, when several robots are malfunctioning, taking their information into consideration shall bias the tracking result. Therefore, at time k, our goal is no longer solving (1) but finding the dynamic optimal solution that minimizes the summation of the regular agents' local cost functions

$$\tilde{x}^{k*} := \arg\min_{\tilde{x}^k} \sum_{i\in\mathcal{R}} f_i^k(\tilde{x}^k). \tag{2}$$

Directly solving (2) is intractable because the identities of malfunctioning agents are not available in advance. To address this issue, we introduce a TV norm penalty on the transmitted values, which include the local iterates of the regular agents and the faulty values from the malfunctioning agents. For agent i, define $\mathcal{R}_i$ as the set of its regular neighbors and $\mathcal{M}_i := \mathcal{N}_i \backslash \mathcal{R}_i$ as the set of its malfunctioning neighbors. At time k, we expect to approximately solve

$$\begin{aligned} x^{k*} := [x_i^{k*}] = \arg\min_{x^k := [x_i^k]} & \sum_{i\in\mathcal{R}} f_i^k(x_i^k) \\ & + \lambda \sum_{i\in\mathcal{R}} \left(\frac{1}{2} \sum_{j\in\mathcal{R}_i} \|x_i^k - x_j^k\|_1 + \sum_{j\in\mathcal{M}_i} \|x_i^k - z_j^k\|_1 \right), \end{aligned} \tag{3}$$

where $x^k := [x_i^k] \in \mathbb{R}^{rp}$ is a vector that stacks all the local variables x_i^k of regular agents, $x^{k*} := [x_i^{k*}] \in \mathbb{R}^{rp}$ is the optimal solution of (3), and λ is a positive constant penalty factor. The second term in the cost function of (3) is the TV norm penalty on the transmitted values, whose minimization forces every x_i^k to be close to most of the received values on agent i, but allows it to be different to those received outliers [20]. Therefore, when the malfunctioning agents are sparse within the network, the TV norm penalty helps mitigate their negative influence. For the applications of TV norm in identifying sparse outliers, readers are referred to [22,23].

We propose a subgradient method to approximately solve (3) in a decentralized and dynamic manner. The subgradient of the cost function in (3) with respect to x_i^k is

$$\nabla f_i^k(x_i^k) + \lambda\left(\sum_{j\in\mathcal{R}_i} sign(x_i^k - x_j^k) + \sum_{j\in\mathcal{M}_i} sign(x_i^k - z_j^k)\right),$$

where $sign(\cdot)$ is an element-wise sign function. Given $a \in \mathbb{R}$, $sign(a)$ equals to 1 when $a > 0$, -1 when $a < 0$, and an arbitrary value within $[-1, 1]$ when $a = 0$. Note that this subgradient can be easily generalized to the case that f_i^k is nondifferentiable, as long as we replace $\nabla f_i^k(x_i^k)$ by a subgradient of f_i^k at x_i^k. For every regular agent i, its subgradient update at time k is

$$x_i^k = x_i^{k-1} - \alpha\nabla f_i^k(x_i^{k-1}) - \alpha\lambda\left(\sum_{j\in\mathcal{R}_i} sign(x_i^{k-1} - x_j^{k-1}) + \sum_{j\in\mathcal{M}_i} sign(x_i^{k-1} - z_j^{k-1})\right), \quad (4)$$

where α is a positive constant stepsize. We use a constant stepsize, other than a diminishing one, for the purpose of adapting to the dynamic cost functions [24].

Algorithm 1 A subgradient method for robust decentralized dynamic optimization.

Input: $x_i^0 \in \mathbb{R}^p$ for $i \in \mathcal{R}$, $\lambda > 0$ and $\alpha > 0$
1: **for** Time $k = 1, 2, \cdots$, every regular agent $i \in \mathcal{R}$ **do**
2: Receive x_j^{k-1} from regularneighbors $j \in \mathcal{R}_i$ and z_j^{k-1} from malfunctioning neighbors $j \in \mathcal{M}_i$.
3: Access local cost function f_i^k.
4: Update local iterate x_i^k according to(4).
5: **end for**

The subgradient method to solve the robust decentralized dynamic optimization problem is outlined in Algorithm 1. The algorithm has two parameters, penalty factor λ and stepsize α. Every regular agent $i \in \mathcal{R}$ initializes its local iterate as x_i^0. At time k, it accesses the local cost function f_i^k, after receiving local iterates x_j^{k-1} from regular neighbors $j \in \mathcal{R}_i$ and broadcast values z_j^{k-1} from malfunctioning neighbors $j \in \mathcal{M}_i$. Note that agent i does not need to know which neighbors are regular or malfunctioning; it only receives broadcast values without distinction. Having all these information at hand, it updates the local iterate x_i^k according to (4). For regular agent i, its communication cost per iteration consists of broadcasting a p-dimensional vector to and receiving $|\mathcal{N}_i|$ p-dimensional vectors from its neighbors. The computation cost, which mainly comes from evaluating the local gradient $\nabla f_i^k(x_i^{k-1})$, is lightweight.

3. Performance analysis

This section analyzes the tracking performance of the proposed robust decentralized dynamic optimization algorithm at presence of the malfunctioning agents. Section 3.1 lists basic assumptions for the analysis. Section 3.2 investigates the TV norm regularized problem (3) at any time k, showing the condition under which the optimal solution of (3) is consensual and its gap from the optimal solution of (2) is bounded. Then in Section 3.3, we bound the tracking error of Algorithm 1, given that the variation of the dynamic optimal solution to (2) is bounded.

3.1. Assumptions

We make the following assumptions on the dynamic local cost functions f_i^k, which are normal for convex analysis.

Assumption 1 (Lipschitz continuous gradients). Local cost functions f_i^k are differentiable and have Lipschitz continuous gradients with Lipschitz constants $M_{f_i^k} < M_{f_i}$, where $M_{f_i} > 0$ are constants, for all regular agents $i \in \mathcal{R}$ and times k; namely, for any pair of points x_i and y_i, it holds $\|\nabla f_i^k(x_i) - \nabla f_i^k(y_i)\| \le M_{f_i^k}\|x_i - y_i\|$.

Assumption 2 (Strong convexity). Local cost functions f_i^k are strongly convex with strong convexity constants $m_{f_i^k} > m_{f_i}$, where $m_{f_i} > 0$ are constants, for all regular agents $i \in \mathcal{R}$ and times k; namely, for any pair of points x_i and y_i, it holds $[x_i - y_i]^T[\nabla f_i^k(x_i) - \nabla f_i^k(y_i)] \ge m_{f_i^k}\|x_i - y_i\|^2$.

Assumption 3 (Bounded Gradients at Optimum). Local cost functions f_i^k have bounded gradients at $\tilde{x}^{k*}$, the dynamic optimal solution to (2), for all regular agents $i \in \mathcal{R}$ and times k; namely, $\|\nabla f_i^k(\tilde{x}^{k*})\| < \infty$.

We also assume that the network of the regular agents is bidirectionally connected. Otherwise, consensus among regular agents is generally impossible.

Assumption 4 (Network connectivity). The network consisting of all regular agents $i \in \mathcal{R}$, denoted by $(\mathcal{R}, \mathcal{E})$, is bidirectionally connected.

For future usage, define the node-edge incidence matrix $A = [a_{ie}] \in \mathbb{R}^{r\times|\mathcal{E}|}$ of the network with all regular agents. If $e = (i, j) \in \mathcal{E}$, then we set $a_{ie} = 1$ and $a_{je} = -1$ (the order of i and j is arbitrary, but by default we consider the ordered edge (i, j) with $i < j$). If an agent i is not attached to an edge e, then $a_{ie} = 0$.

The last assumption is about the variation of the dynamic optimal solution of (2) over time, which must be bounded to guarantee reasonable tracking performance.

Assumption 5. For any two successive times $k-1$ and k, the variation of the dynamic optimal solution of (2) is bounded by a positive constant Θ; namely

$$\|\tilde{x}^{k*} - \tilde{x}^{(k-1)*}\| \le \Theta. \quad (5)$$

3.2. Gap between (2) and (3)

Before stating the main result, we introduce an auxiliary problem in the form of

$$\tilde{y}^{k*} := \arg\min_{\tilde{y}^k} \sum_{i\in\mathcal{R}} f_i^k(\tilde{y}^k) + \lambda\sum_{i\in\mathcal{R}}\sum_{j\in\mathcal{M}_i}\|\tilde{y}^k - z_j^k\|_1. \quad (6)$$

The first-order optimality condition of (6) is that, for any malfunctioning agent j, there exists $v_j \in \mathbb{R}^p$ whose value satisfies the definition of $sign(\tilde{y}^{k*} - z_j^k)$ such that

$$\frac{1}{\lambda}\sum_{i\in\mathcal{R}}\nabla f_i^k(\tilde{y}^{k*}) + \sum_{i\in\mathcal{R}}\sum_{j\in\mathcal{M}_i} v_j = 0. \quad (7)$$

Indeed, v_j is a the subgradient of $\|\tilde{y}^k - z_j^k\|_1$ at the point $\tilde{y}^k = \tilde{y}^{k*}$. For notational simplicity, define a vector $b^k := [b_i^k] \in \mathbb{R}^{rp}$ whose ith block $b_i^k := \nabla f_i^k(\tilde{y}^{k*})/\lambda + \sum_{j\in\mathcal{M}_i} v_j$. Thus, (7) is equivalent to $\sum_{i\in\mathcal{R}} b_i^k = 0$. An immediate implication of Assumptions 1–3, given by the following lemma, is that b^k is upper bounded. Its proof is in Appendix A.

Lemma 1. *Under Assumptions 1–3, b^k are upper bounded for all times k; namely, $\|b^k\| < \infty$.*

The following lemma provides a sufficient condition under which (3) and the auxiliary problem (6) are equivalent. Its proof is in Appendix B.

Lemma 2. *Define a vector* $u := [u_e] \in \mathbb{R}^{|\mathcal{E}|p}$ *whose eth block is* $u_e \in \mathbb{R}^p$. *If there exists u whose elements are within the range of* $[-1, 1]$, *such that* $A \otimes I_p u + b^k = 0$ *holds, where* I_p *is a* $p \times p$ *identity matrix, then under Assumption 2, the optimal solution* $x^{k*} := [x_i^{k*}]$ *of (3) is consensual and all the blocks* x_i^{k*} *equal to the optimal solution* $\tilde{y}^{k*}$ *of (6).*

Remark 1. Observe that Lemma 2 only provides a sufficient condition, which is not necessarily tight, for the equivalence of (3) and (6). Now we show that $A \otimes I_p u + b^k = 0$ may have a solution whose elements are within the range of $[-1, 1]$.

Since by Assumption 4, the network consisting of all regular agents $(\mathcal{R}, \mathcal{E})$ is bidirectionally connected, the node-edge incidence matrix A is with rank $r-1$ and the null space of A^T is $\text{span}(1_r)$, where 1_r is an all-one r-dimensional vector. Therefore, any nonzero r-dimensional vector with summation being 0 is not in the null space of A^T. Consequently, the condition $\sum_{i\in\mathcal{R}} b_i^k = 0$ guarantees that b^k is not in the null space of $A^T \otimes I_p$ and the linear equation $A \otimes I_p u + b^k = 0$ has at least one solution. By Lemma 1, this solution is bounded because $\|b^k\| < \infty$.

To further guarantee that the elements of the solution are within the range of $[-1, 1]$, the magnitude of b^k must be small enough. Note that ith block of b^k is $b_i^k = \nabla f_i^k(\tilde{y}^{k*})/\lambda + \sum_{j\in\mathcal{M}_i} v_j$. The magnitude of the first term $\nabla f_i^k(\tilde{y}^{k*})/\lambda$ is small as long as λ is sufficiently large. Because $v_j \in \mathbb{R}^p$ is a subgradient and its entries are within the range of $[-1, 1]$, the second term is bounded by $\|\sum_{j\in\mathcal{M}_i} v_j\| \le \sqrt{p}|\mathcal{M}_i|$, meaning that the number of malfunctioning agents attached to every regular agent must be small enough.

In summary, Lemma 2 implies that (3) and (6) are equivalent, given that the regularization factor λ is large enough and the number of malfunctioning agents is small enough.

We proceed to show that the optimal solutions of (2) and (6) have a bounded gap in Lemma 3. Its proof is in Appendix C.

Lemma 3. *Under Assumption 2, the distance between the dynamic optimal solution* $\tilde{x}^{k*}$ *of (2) and the optimal solution* $\tilde{y}^{k*}$ *of (6) is bounded by*

$$\|\tilde{x}^{k*} - \tilde{y}^{k*}\| \le \Delta^k := \frac{\lambda\sqrt{p}}{\sum_{i\in\mathcal{R}} m_{f_i^k}} \sum_{i\in\mathcal{R}} |\mathcal{M}_i|. \tag{8}$$

In Lemma 3, the gap Δ^k is proportional to λ, meaning that large λ yields large approximation error. Meanwhile, Remark 1 asserts that large λ enhances consensus among the regular agents. Therefore, setting a proper λ helps achieve the tradeoff between network consensus and approximation accuracy. Further, from Remark 1 and Lemma 3, if the number of malfunctioning neighbors is large, network consensus is difficult to reach and approximation error is also remarkable. This makes sense because the number of malfunctioning agents dictates the performance of the TV norm regularized problem.

Summarizing Lemmas 2 and 3 immediately yields the main theorem on the bounded gap between (2) and (3). Therefore, (3) is a good surrogate of (2), which minimizes the summation of the regular agents' local cost functions.

Theorem 1. *Suppose that Assumptions 1–4 hold true. Define a vector* $u := [u_e] \in \mathbb{R}^{|\mathcal{E}|p}$ *whose eth block is* $u_e \in \mathbb{R}^p$. *If there exists u whose elements are within the range of* $[-1, 1]$, *such that* $A \otimes I_p u + b^k = 0$ *holds, then the optimal solution* $x^{k*} := [x_i^{k*}]$ *of (3) is consensual and the distance between every block* x_i^{k*} *and the optimal solution* $\tilde{x}^{k*}$ *of (2) satisfies* $\|x_i^{k*} - \tilde{x}^{k*}\| \le \Delta^k$, *where* Δ^k *is defined in (8).*

Below we provide two simple examples to illustrate the theoretical results.

Example 1. Consider a network consisting of 3 fully connected regular agents 1, 2 and 3, where two malfunctioning agents 4 and 5 are attached to 1 and 2, respectively. For regular agent i, its local cost function is $f_i^k(\tilde{x}) = (\tilde{x}^k - i)^2/2$. For malfunctioning agent j, we suppose that it always sends a constant $z_j^k = 10$ to its neighbor. The optimal solution of (6) is $\tilde{y}^{k*} = 2 + 2\lambda/3$ if $0 < \lambda < 12$ and $\tilde{y}^{k*} = 10$ if $\lambda \ge 12$. The optimal solution of (3), if reaching consensus, is $x_i^{k*} = \tilde{y}^{k*}$ for all $i \in \mathcal{R}$. Numerical experiments show that such a consensus is attainable when $\lambda \ge 0.44$.

Now we check the theoretical condition given by Theorem 1. The node-edge incidence matrix of the regular agents is

$$A = \begin{pmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & -1 \end{pmatrix}.$$

The vector $b^k = [1/\lambda - 1/3; -1/3; -1/\lambda + 2/3]^T$ if $0 < \lambda < 12$ and $b^k = [-3/\lambda; -4/\lambda; 7/\lambda]^T$ if $\lambda \ge 12$. Because b^k is not in the null space of A^T, a sufficient condition for $Au + b^k = 0$ to have a solution whose elements are within the range of $[-1, 1]$ is that $\|b^k\|/\sigma_{\min} \le 1$, where $\sigma_{\min}$ is the smallest nonzero singular value of A. For this case, $\sigma_{\min} = \sqrt{3}$. Thus, it can be predicted that when $\lambda \ge 0.59$, the optimal solution of (3) is consensual and every block is the same as that of (6). This theoretical threshold is close to the numerical threshold 0.44.

Then we check the gap between $\tilde{x}^{k*}$ and $\tilde{y}^{k*}$. Because $\tilde{x}^{k*} = 2$, $\|\tilde{x}^{k*} - \tilde{y}^{k*}\| = 2\lambda/3$ if $0 < \lambda < 12$ and $\|\tilde{x}^{k*} - \tilde{y}^{k*}\| = 8$ if $\lambda \ge 12$. This is close to the theoretical gap $2\sqrt{3}\lambda/3$ as predicted by Theorem 2.

Example 2. Consider another example whose setting is the same as that in Example 1, except that regular agents 2 and 3 are not connected. For this case, the node-edge incidence matrix of the regular agents is

$$A = \begin{pmatrix} 1 & 1 \\ -1 & 0 \\ 0 & -1 \end{pmatrix},$$

whose smallest nonzero singular value is 1. Similar to the discussion in Example 1, the theoretical condition for the equivalence of (3) and (6) is $\lambda \ge 0.87$, while the numerical result is $\lambda \ge 0.75$. The gap between $\tilde{x}^{k*}$ and $\tilde{y}^{k*}$ is the same as that in Example 1.

With particular note, Ben-Ameur et al. [20] analyzes conditions under which (3) achieves consensus, as well as (2) and (3) are equivalent, given that all the agents are regular. Here we extend the result to the more challenging case that the malfunctioning agents exist and play negative roles. In addition, Ben-Ameur et al. [20] considers the *static* TV norm regularized problem, while we further investigate the *dynamic* tracking performance, as shown below.

3.3. Tracking performance

Theorem 1 asserts that (3) is a good approximation of (2). Now we further show that Algorithm 1, which approximately solves (3) using one subgradient evaluation per time, other than running an inner loop of multiple subgradient evaluations at each time index, tracks the optimal solution of (3) with bounded error.

Theorem 2. *Given that the conditions of Theorem 1 as well as Assumption 5 hold true, the tracking error between* x^k *and the optimal solution of (3) is upper bounded by*

$$\|x^k - x^{k*}\| \le c^k\|x^0 - x^{0*}\| + \frac{1}{1-c}\left(2c\sqrt{r}\max_k \Delta^k + c\sqrt{r}\Theta + d\right), \quad (9)$$

if the stepsize $\alpha < \min_k \quad 1/(\min_{i\in\mathcal{R}} m_{f_i^k} + \max_{i\in\mathcal{R}} M_{f_i^k})$. *Here* $c := \max_k \quad (1 - 2\alpha m_{f^k} M_{f^k}/(m_{f^k} + M_{f^k}))^{1/2}$ *and* $d := (8\alpha^2\lambda^2 p \sum_{i\in\mathcal{R}} |\mathcal{N}_i|^2)^{1/2}$ *are two constants.*

Since c is a constant within the range of (0,1), Theorem 2 implies that the influence of the initial tracking error $\|x^0 - x^{0*}\|$ vanishes at an exponential rate. The steady-state tracking error, as $k \to \infty$, is proportional to $\max_k \Delta^k$ (the gap between the optimal solutions of (2) and (6)), Θ (the variation of the dynamic optimal solution of (2)), as well as $(\sum_{i\in\mathcal{R}} |\mathcal{N}_i|^2)^{1/2}$ (a constant determined by the topology of regular and malfunctioning agents).

Note that the term d in (9) depends on $\mathcal{N}_i$, not $\mathcal{M}_i$, due to the proof techniques. To be specific, the proof in Appendix D needs to bound the subgradient of the cost function in (3), which relies on not only $\mathcal{M}_i$ (malfunctioning agents), but also $\mathcal{R}_i$ (regular agents). On the other hand, when we increase the connections between regular agents, the conditions in Theorem 2 can be satisfied for a smaller λ. Therefore, the value of d could be reduced through selecting a smaller λ so as to reduce the upper bound of the tracking error.

Theorem 2 shows how Algorithm 1 tracks the optimal solution of (3). Combining Theorems 1 and 2, it is straightforward assert that Algorithm 1 is also able to track the dynamic optimal solution of (2) with bounded error.

Corollary 1. *Given that the conditions of Theorem 1 as well as Assumption 5 hold true, the tracking error between* x^k *and the optimal solution of (2) is upper bounded by*

$$\|x^k - [\tilde{x}^{k*}]\| \le c^k\|x^0 - x^{0*}\| + \frac{1}{1-c}\left((1+c)\sqrt{r}\max_k \Delta^k + c\sqrt{r}\Theta + d\right), \quad (10)$$

if the stepsize $\alpha < \min_k \quad 1/(\min_{i\in\mathcal{R}} m_{f_i^k} + \max_{i\in\mathcal{R}} M_{f_i^k})$. *Here* $c := \max_k \quad (1 - 2\alpha m_{f^k} M_{f^k}/(m_{f^k} + M_{f^k}))^{1/2}$ *and* $d := (8\alpha^2\lambda^2 p \sum_{i\in\mathcal{R}} |\mathcal{N}_i|^2)^{1/2}$ *are two constants, and* $[\tilde{x}^{k*}] \in \mathbb{R}^{rp}$ *stacks* r *optimal solution* $\tilde{x}^{k*}$ *of (2).*

4. Numerical experiments

This section provides numerical experiments to demonstrate robustness of the proposed decentralized dynamic optimization algorithm at presence of the malfunctioning agents. We compare the proposed algorithm with the dynamic version of the celebrated decentralized gradient descent (DGD) method [25], which does not consider mitigating the influence of the malfunctioning agents. At time k, DGD updates the local variables as

$$x_i^{k+1} = \sum_{j\in\mathcal{R}_i} w_{ij}x_j^k + \sum_{j\in\mathcal{M}_i} w_{ij}z_j^k - \beta\nabla f_i(x_i^k), \quad (11)$$

for all $i \in \mathcal{R}$. Here β is a positive constant stepsize, and $W = [w_{ij}] \in \mathbb{R}^{n\times n}$ is the mixing matrix of the entire network including both regular and malfunctioning agents. In the numerical experiments, we choose W according to the maximum-degree rule [26].

We consider two kinds graphs, both with $n = 100$ agents. The first one is a random geometric graph, which uniformly randomly places 100 agents in a two-dimensional area $[0, 3] \times [0, 3]$ and treats two agents as neighbors if and only if their distance is less than 1. The second one is a line graph. The agents track a moving target whose true position $\tilde{x}^k \in \mathbb{R}^2$ evolves along a 3/4 circle, starting from (0,0), heading to $(-3, 3)$ and then (6,0), and ending at (3,3). The velocity of the target is constant and each 1/4 circle takes 100 time slots. At time k, regular agent i measures a true position $\tilde{x}^k$ through a linear observation function $y_i^k = H_i^k\tilde{x}^k + e_i^k$, where elements of the measurement matrix $H_i^k \in \mathbb{R}^{2\times 2}$ follow normal distribution $\mathcal{N}(0, 1)$ and elements of the measurement noise $e_i^k \in \mathbb{R}^2$ follow normal distribution $\mathcal{N}(0, 1)$. Thus, the regular agents aim at finding $\tilde{x}^{k*} := \arg\min \sum_{i\in\mathcal{R}} f_i^k(\tilde{x}^k)$, where $f_i^k(\tilde{x}^k) = \|H_i^k\tilde{x}^k - y_i^k\|^2/2$. The performance metric is tracking error defined by $\sum_{i\in\mathcal{R}} \|x_i^k - \tilde{x}^{k*}\|/r$.

4.1. Comparison with DGD

We first compare Algorithm 1 with DGD in terms of their robustness to malfunctioning agents over both random geometric graph and line graph.

4.1.1. Random geometric graph

Randomly choose $m = 3$ malfunctioning agents among $n = 100$ agents, but guarantee that the network of regular agents is connected. Suppose that the malfunctioning agents broadcast the same faulty vectors. We consider three different settings for faulty vectors: for all k and for all $i \in \mathcal{M}$, $z_i^k = [5; 5]$, $z_i^k = [10; 10]$, or $z_i^k = [20; 20]$.

Performance of DGD with stepsize $\beta = 0.2$, which is hand-tuned to yield balanced tracking performance, is illustrated in Fig. 1. The left plot compares the true signal $\tilde{x}^k$ and the decentralized estimates of a randomly chosen regular agent for different levels of faulty values. When the magnitude of the faulty vectors becomes larger, the bias between the decentralized estimate and the true signal also becomes more significant. The impact of the faulty vectors can be further observed from the right plot, which shows the overall tracking error of the network. As the faulty vectors vary from [5,5] to [20,20], the steady-state tracking error increases from around 0.5 to around 2.

Performance of Algorithm 1 with stepsize $\alpha = 0.1$ and regularization parameter $\lambda = 0.1$ is illustrated in Fig. 2. Thanks to the TV norm regularization term, the network is not sensitive to the faulty vectors broadcast by the malfunctioning agents. The left plot shows that, no matter how the faulty vectors vary, the decentralized estimate of a randomly chosen regular agent is always close to the true signal $\tilde{x}^k$. The right plot depicts that the steady-state tracking errors are always around 0.3, which are much smaller than those of DGD, for all the three cases.

4.1.2. Line graph

The line graph connects agent i with agent $i+1$ from $i = 1$ to $i = 99$. Choose $m = 2$ malfunctioning agents labelled as 1 and 100, which are the end nodes of the line. Similar to the previous setting, we consider three different levels for faulty vectors: for all k and for all $i \in \mathcal{M}$, $z_i^k = [5; 5]$, $z_i^k = [10; 10]$, or $z_i^k = [20; 20]$.

Fig. 3 shows performance of DGD with stepsize $\beta = 0.2$. The left plot demonstrates the true signal $\tilde{x}^k$ and the decentralized estimates of regular agent 2, which is directly connected with malfunctioning agent 1. The presence of a neighbor that constantly broadcasts faulty vectors has significant influence on the estimates of agent 2. When the faulty vectors are as large as [20; 20], agent 2 is almost unable to track the true trajectory. Interestingly, the average impact of the malfunctioning agents does not vary too much when the magnitude of the faulty vectors increases, as observed from the overall tracking error in the right plot. We conjecture that the faulty vectors have to propagate from the end nodes to the middle of the network, and their impact decays when the line graph is large. Therefore, a large number of regular agents are still able to track the true trajectory well in this case (observed in the experiments but not shown here), even though the magnitude of the faulty vectors is large.

We test Algorithm 1 with stepsize $\alpha = 0.1$ and regularization parameter $\lambda = 0.1$ over the line graph, as shown in Fig. 4. The left

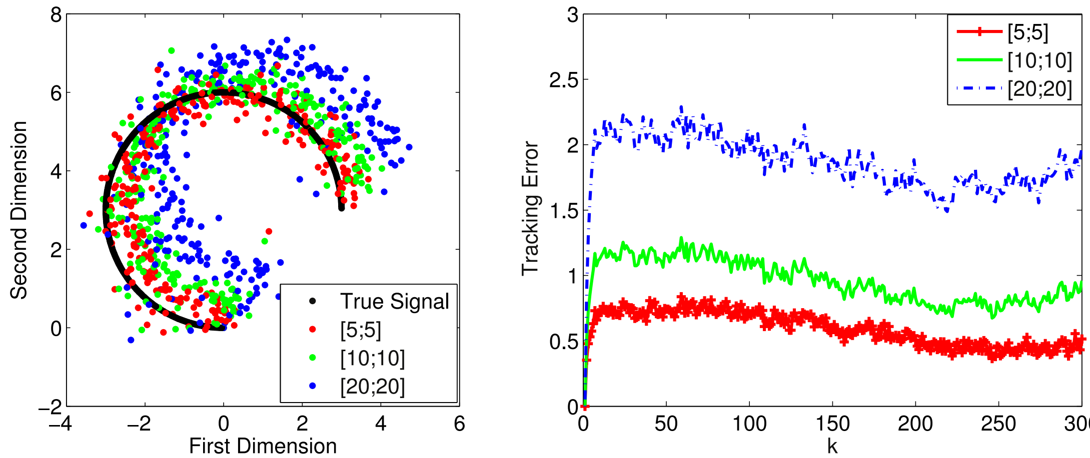

Fig. 1. DGD with three malfunctioning agents broadcasting the same faulty vectors, [5; 5], [10; 10] or [20; 20], over a random geometric graph. Left: True signal and decentralized estimates of a randomly chosen regular agent. Right: Tracking error of the network.

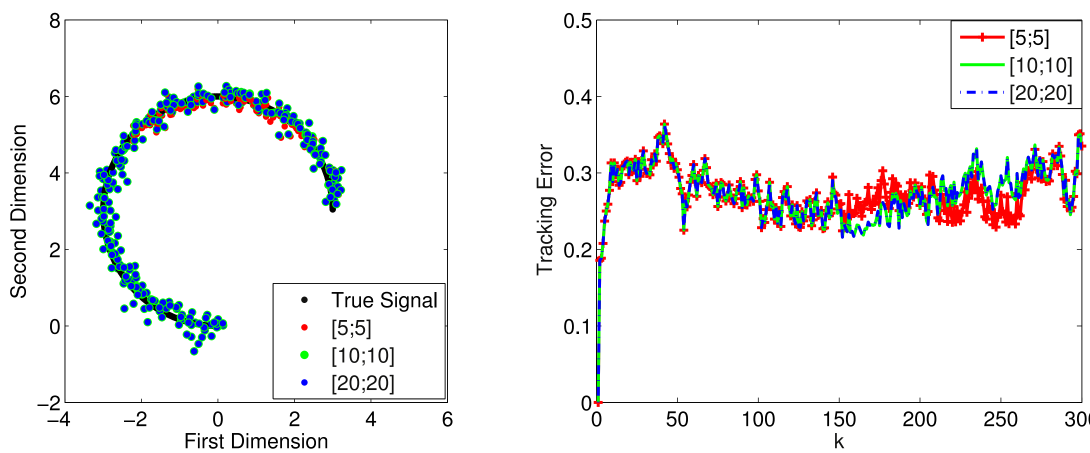

Fig. 2. Algorithm 1 with three malfunctioning agents broadcasting the same faulty vectors, [5; 5], [10; 10] or [20; 20], over a random geometric graph. Left: True signal and decentralized estimates of a randomly chosen regular agent. Right: Tracking error of the network.

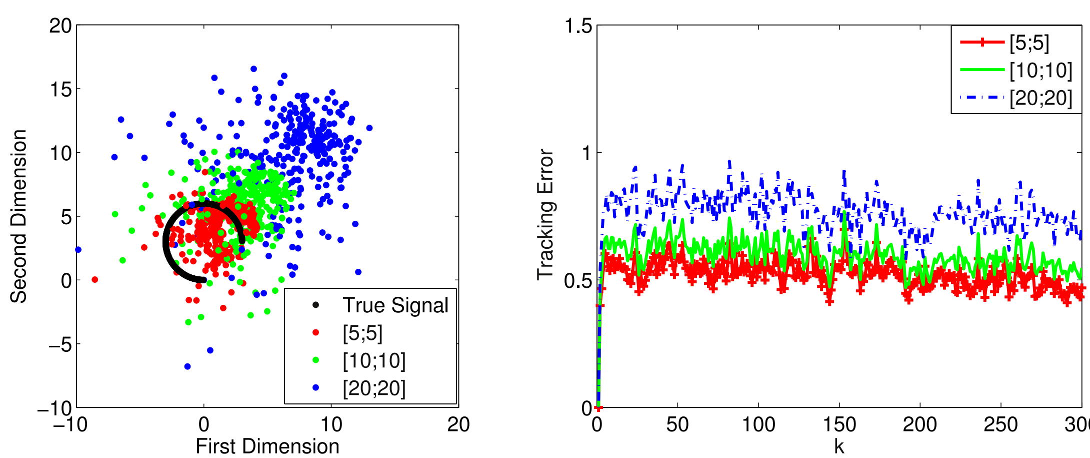

Fig. 3. DGD with malfunctioning agents 1 and 100 broadcasting the same faulty vectors, [5; 5], [10; 10] or [20; 20], over a line graph. Left: True signal and decentralized estimates of regular agent 2. Right: Tracking error of the network.

plot demonstrate that, even constantly disturbed by a malfunctioning agent, regular agent 2 still gives satisfactory decentralized estimates that are close to the true signal $\tilde{x}^k$. The steady-state tracking errors depicted in the right plot are lower than 0.4, which are much smaller than those of DGD.

Note that for the line graph, there does not exist any u whose elements are within the range of $[-1, 1]$, such that $A \otimes I_p u + b^k = 0$ holds. Though there is no theoretical guarantee, the proposed robust decentralized dynamic optimization algorithm still performs well, demonstrating its adaptability to network topology.

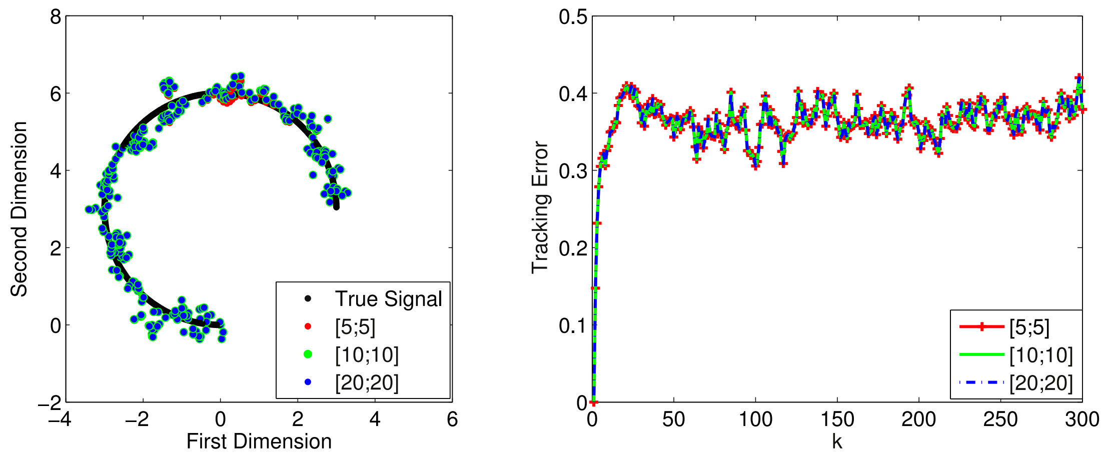

Fig. 4. Algorithm 1 with malfunctioning agents 1 and 100 broadcasting the same faulty vectors, [5; 5], [10; 10] or [20; 20], over a line graph. Left: True signal and decentralized estimates of regular agent 2. Right: Tracking error of the network.

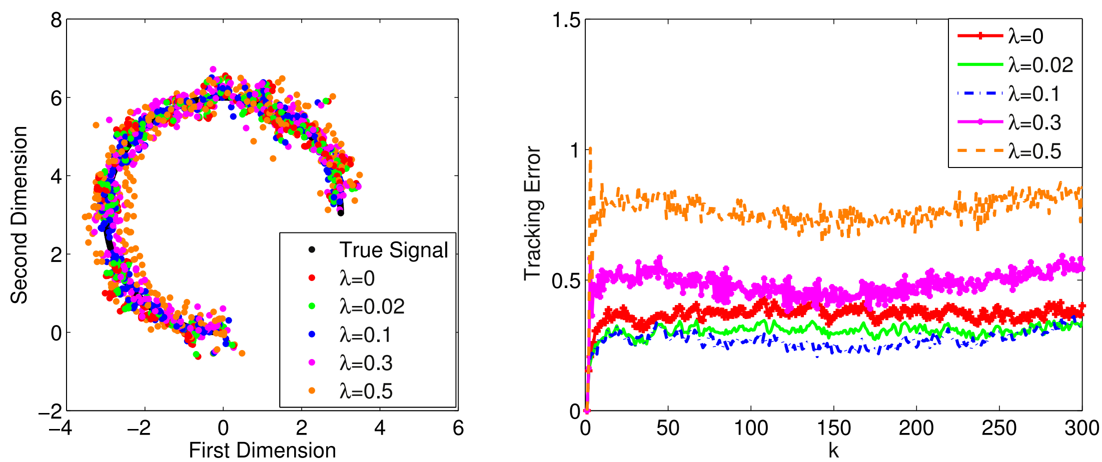

Fig. 5. Algorithm 1 with three malfunctioning agents broadcasting the same faulty vectors [10; 10] over a random geometric graph, while the regularization parameter λ varies from 0, 0.02, 0.1, 0.3 to 0.5. Left: True signal and decentralized estimates of a randomly chosen regular agent. Right: Tracking error of the network.

4.2. Impact of various factors on Algorithm 1

Now we investigate the impact of various factors, including the regularization parameter λ, the number of malfunctioning agents m and the form of faulty values, on the performance of Algorithm 1. All the experiments are conducted over the random geometric graph with $m = 100$ agents.

4.2.1. Impact of regularization parameter λ

Randomly choose $m = 3$ malfunctioning agents among $n = 100$ agents and suppose that the malfunctioning agents broadcast the same faulty vectors [10; 10]. The stepsize remains to be $\alpha = 0.1$ but the regularization parameter λ varies from 0, 0.02, 0.1, 0.3 to 0.5. Note that $\lambda = 0$ corresponds to that the malfunctioning agents do not collaborate with any others, no matter regular or malfunctioning, and independently optimize their own local cost functions. The left plot of Fig. 5 shows the decentralized estimates of a random regular agent, which are not far away from the true signal and are robust to the setting of λ. Observing the right plot, we can see that too large or too small λ both yield large steady-state tracking error.

In practice, selecting a proper λ so as to minimize the tracking error is a challenging task. Indeed, Ben-Ameur et al. [20] provides guidelines of selecting λ for two specific problems (average consensus and medium consensus), when the environment is static and the malfunctioning agents are absent. For a general problem under dynamic environment and at presence of malfunctioning agents, calculating the optimal λ relies on the global knowledge of the network topology and the local cost functions. However, the theoretical analysis still provides clues of selecting λ. As we have discussed after Lemma 3, a large λ helps consensus of the regular agents, but the reached consensus is not necessarily close to the dynamic optimal solution. Therefore, to achieve the tradeoff between consensus and approximation accuracy, we recommend to select a relatively small λ, which allows the regular agents to be "selfish" such that network-wide consensus is only slightly violated. Nevertheless, when the measurement noise becomes larger, the best λ should also be larger, because "selfish" decisions inevitably lead to significant tracking errors.

For every λ, we also count the number of time steps that the theoretical condition is satisfied, namely, there exists u whose elements are within the range of $[-1, 1]$ such that $A \otimes I_p u + b^k = 0$ holds. When $\lambda = 0.3$ and $\lambda = 0.5$, 12 and 208 steps out of 300 satisfy the condition, respectively. The condition cannot be satisfied when $\lambda \le 0.1$ and is always satisfied when $\lambda \ge 1$. Note that the condition is sufficient and not necessarily tight. We can observe that the proposed algorithm is robust even when there is no theoretical guarantee. How to establish a tighter condition remains an open problem.

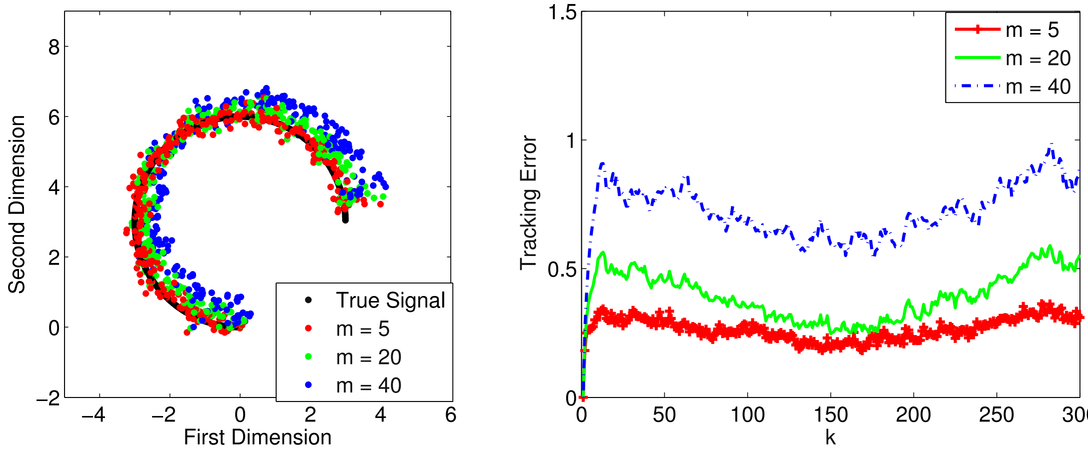

Fig. 6. Algorithm 1 with m malfunctioning agents broadcasting the same faulty vectors [10; 10] over a random geometric graph, while the number of malfunctioning agents m varies from 5, 20 to 40. Left: True signal and decentralized estimates of a randomly chosen regular agent. Right: Tracking error of the network.

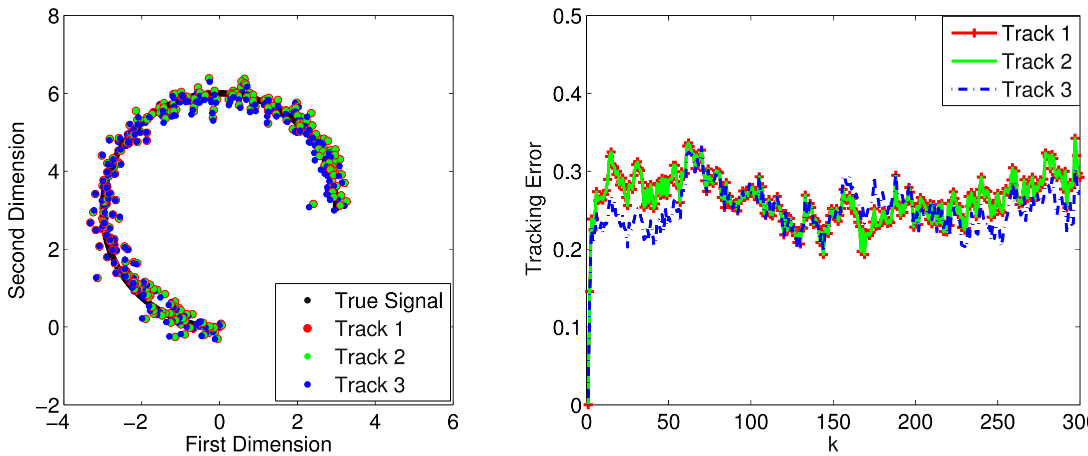

Fig. 7. Algorithm 1 with three malfunctioning agents broadcasting different forms of faulty values over a random geometric graph. Left: True signal and decentralized estimates of a randomly chosen regular agent. Right: Tracking error of the network.

4.2.2. Impact of number of malfunctioning agents m

Setting stepsize $\alpha = 0.1$ and regularization parameter $\lambda = 0.1$ in Algorithm 1, we vary the number of malfunctioning agents m from 5, 20 to 40. The m malfunctioning agents are randomly chosen, but guaranteeing that the network of regular agents remains connected. The malfunctioning agents broadcast the same faulty vectors [10; 10]. The left plot of Fig. 6 shows the true signal and the decentralized estimates of one agent, which is always regular in the experiments. Surprisingly, even at presence of 40 malfunctioning agents, the regular agents are still able to track the true signal well. The tracking error in the right plot increases from around 0.3 to around 0.8, when m increases from 5 to 40. Therefore, the TV norm regularization is robust even when the malfunctioning agents are no longer sparse within the network.

4.2.3. Impact of form of faulty values

Finally, we validate that Algorithm 1 is robust to different forms of faulty values. Also consider the random geometric graph with $m = 3$ malfunctioning agents out of $n = 100$. The stepsize is $\alpha = 0.1$ and the regularization parameter $\lambda = 0.1$. We consider three forms of faulty values. Track 1: the malfunctioning agents broadcast fixed faulty vectors [20; 20]. Track 2: the malfunctioning agents broadcast random faulty vectors whose elements are uniformly randomly chosen within [10, 15]. Track 3: each faulty vector also evolves along a 3/4 circle, but with an overshoot of 1/4 circle – for example, the faulty vector is $(-3, 3)$ when the true signal is (0,0). Observing the decentralized estimates of a randomly chosen regular agent in the left plot as well as the tracking error of the network in the right plot of Fig. 7, we conclude that the proposed TV norm regularization technique and the subgradient method are insensitive to the form of faulty values.

5. Conclusion

Dynamically minimizing the summation of time-varying local cost functions over a network is of particular interest in various applications, such as target tracking and adaptive filtering. However, some of the network agents could be malfunctioning due to the failures of their computation and/or communication units. When the regular agents broadcast their current iterates, the malfunctioning agents broadcast faulty values, and hence lead the optimization process to a wrong direction. Through introducing TV norm regularization that has been proved to be a powerful tool in decentralized *static* optimization in [20] into decentralized *dynamic* optimization, we force the local variables of the regular agents to be close while allows them to be different with the faulty values broadcast by the malfunctioning agents. A fully decentralized subgradient algorithm is proposed to dynamically solve the TV norm regularized problem. The tracking error is proved to be bounded, given that the variation of the optimal solution is also bounded. Numerical experiments demonstrate that, at presence of the malfunctioning agents, the proposed algorithm is superior to the dynamic version of DGD in terms of the tracking error.

In the future work, we shall try to intelligently identify the malfunctioning agents, and correct the optimization process using the identification result so as to further enhance the robust tracking performance. This is related to the adversarial agent identification problem in randomized gossiping [19] and distributed detection [27,28]. Another direction of research is to analyze the performance of the proposed algorithm when the faulty messages are not arbitrary but follow certain rules, which could enable us to establish a tighter bound of the tracking error.

Appendix A. Proof of Lemma 1

Proof. By Assumption 2, $\sum_{i\in\mathcal{R}} f_i^k(\tilde{x}^k)$ is also strongly convex with constant $\sum_{i\in\mathcal{R}} m_{f_i^k}$. Therefore, we have

$$\|\tilde{x}^{k*}-\tilde{y}^{k*}\| \le \frac{1}{\sum_{i\in\mathcal{R}} m_{f_i^k}} \|\sum_{i\in\mathcal{R}} \nabla f_i^k(\tilde{x}^{k*}) - \sum_{i\in\mathcal{R}} \nabla f_i^k(\tilde{y}^{k*})\|. \tag{A.1}$$

From the first-order optimality condition of (2), $\sum_{i\in\mathcal{R}} \nabla f_i^k(\tilde{x}^{k*})=0$. From (7), the first-order optimality condition of (6), $\sum_{i\in\mathcal{R}} \nabla f_i^k(\tilde{y}^{k*}) = -\lambda \sum_{i\in\mathcal{R}}\sum_{j\in\mathcal{M}_i} v_j$. Thus, we can rewrite the right-hand side of (A.1) and obtain

$$\begin{aligned}\|\tilde{x}^{k*}-\tilde{y}^{k*}\| &\le \frac{1}{\sum_{i\in\mathcal{R}} m_{f_i^k}} \|\lambda \sum_{i\in\mathcal{R}}\sum_{j\in\mathcal{M}_i} v_j\| \\ &\le \frac{\lambda\sqrt{p}\sum_{i\in\mathcal{R}} |\mathcal{M}_i|}{\sum_{i\in\mathcal{R}} m_{f_i^k}}.\end{aligned} \tag{A.2}$$

By Assumption 1, f_i^k has Lipschitz continuous gradients with constant $M_{f_i^k}$ such that $\|\nabla f_i^k(\tilde{x}^{k*}) - \nabla f_i^k(\tilde{y}^{k*})\| \le M_{f_i^k}\|\tilde{x}^{k*}-\tilde{y}^{k*}\|$. This fact and (A.2) imply that

$$\|\nabla f_i^k(\tilde{x}^{k*}) - \nabla f_i^k(\tilde{y}^{k*})\| \le \frac{\lambda\sqrt{p}M_{f_i^k}\sum_{i\in\mathcal{R}} |\mathcal{M}_i|}{\sum_{i\in\mathcal{R}} m_{f_i^k}}. \tag{A.3}$$

Also noticing that $\|\sum_{j\in\mathcal{M}_i} v_j\| \le \sqrt{p}|\mathcal{M}_i|$ because $v_j\in\mathbb{R}^p$ is a subgradient and its entries are within the range of $[-1,1]$, we can bound $b_i^k = \nabla f_i^k(\tilde{y}^{k*})/\lambda + \sum_{j\in\mathcal{M}_i} v_j$ by

$$\begin{aligned}\|b_i^k\| &\le \frac{1}{\lambda}\|\nabla f_i^k(\tilde{x}^{k*})\| + \frac{1}{\lambda}\|\nabla f_i^k(\tilde{x}^{k*}) - \nabla f_i^k(\tilde{y}^{k*})\| + \|\sum_{j\in\mathcal{M}_i} v_j\| \\ &\le \frac{1}{\lambda}\|\nabla f_i^k(\tilde{x}^{k*})\| + \frac{\sqrt{p}M_{f_i^k}\sum_{i\in\mathcal{R}}|\mathcal{M}_i|}{\sum_{i\in\mathcal{R}} m_{f_i^k}} + \sqrt{p}|\mathcal{M}_i|.\end{aligned} \tag{A.4}$$

Because $\|\nabla f_i^k(\tilde{x}^{k*})\| < \infty$ according to Assumption 3, we have $\|b_i^k\| \le \infty$, which implies that $\|b^k\| \le \infty$ and completes the proof. □

Appendix B. Proof of Lemma 2

Proof. The optimality condition of (3) is that

$$\frac{1}{\lambda}\nabla f_i^k(x_i^{k*}) + \sum_{j\in\mathcal{R}_i} u_{ij} + \sum_{j\in\mathcal{M}_i} v_{ij} = 0, \quad \forall i\in\mathcal{R}, \tag{B.1}$$

where the value of u_{ij} satisfies the definition of $sign(x_i^{k*}-x_j^{k*})$ for every $i\in\mathcal{R}$ and $j\in\mathcal{R}_i$, and the value of v_{ij} satisfies the definition of $sign(x_i^{k*}-z_j^k)$ for every $i\in\mathcal{R}$ and $j\in\mathcal{M}_i$. The optimal solution x^{k*} is unique since f_i^k is strongly convex by Assumption 2.

By hypothesis, there exist a group of variables u_e whose elements are within the range of $[-1,1]$ for all edges e, such that $A\otimes I_p u + b^k = 0$ with $u=[u_e]$. Substituting the definitions of A and b^k yields

$$\frac{1}{\lambda}\nabla f_i^k(\tilde{y}^{k*}) + \sum_{j:\ e=(i,j)} u_e - \sum_{j:\ e=(j,i)} u_e + \sum_{j\in\mathcal{M}_i} v_j = 0, \tag{B.2}$$

for all $i\in\mathcal{R}$. Since the elements of u_e are within the range of $[-1,1]$, the value of u_e satisfies the definition of $sign(0)$ for every edge $e=(i,j)$ or $e=(j,i)$, where $i\in\mathcal{R}$ and $j\in\mathcal{R}_i$. Meanwhile, The value of v_{ij} satisfies the definition of $sign(\tilde{y}^{k*}-z_j^k)$ for every $i\in\mathcal{R}$ and $j\in\mathcal{M}_i$. Thus, there exists a group of variables $x_i^{k*}=\tilde{y}^{k*}$ for all $i\in\mathcal{R}$, $u_{ij}=u_e$ for all $e=(i,j)$ with $i\in\mathcal{R}$, $j\in\mathcal{R}_i$ and $i<j$, $u_{ij}=-u_e$ for all $e=(i,j)$ with $i\in\mathcal{R}$, $j\in\mathcal{R}_i$ and $i>j$, as well as $v_{ij}=v_j$ for all $i\in\mathcal{R}$ and $j\in\mathcal{M}_i$, such that (B.1) holds. Thus, we conclude that $x_i^{k*}=\tilde{y}^{k*}, \forall i\in\mathcal{R}$ is the optimal solution of (3), where $\tilde{y}^{k*}$ is the optimal solution of (6). □

Appendix C. Proof of Lemma 3

Proof. The first-order optimality condition of (2) is

$$\sum_{i\in\mathcal{R}} \nabla f_i^k(\tilde{x}^{k*}) = 0. \tag{C.1}$$

The first-order optimality condition of (6) is that, for any malfunctioning agent j, there exists $v_j\in\mathbb{R}^p$ whose value satisfies the definition of $sign(\tilde{y}^{k*}-z_j^k)$ such that

$$\sum_{i\in\mathcal{R}} \nabla f_i^k(\tilde{y}^{k*}) + \lambda\sum_{i\in\mathcal{R}}\sum_{j\in\mathcal{M}_i} v_j = 0. \tag{C.2}$$

Subtracting (C.1) and (C.2), we have

$$\begin{aligned}\lambda\|\sum_{i\in\mathcal{R}}\sum_{j\in\mathcal{M}_i} v_j\| &= \|\sum_{i\in\mathcal{R}}\left(\nabla f_i^k(\tilde{x}^{k*}) - \nabla f_i^k(\tilde{y}^{k*})\right)\| \\ &\ge \sum_{i\in\mathcal{R}} m_{f_i^k}\|\tilde{x}^{k*}-\tilde{y}^{k*}\|.\end{aligned} \tag{C.3}$$

The last inequality holds because $\sum_{i\in\mathcal{R}} f_i^k(\tilde{x}^k)$ is strongly convex with constant $\sum_{i\in\mathcal{R}} m_{f_i^k}$. Applying the inequality $\|\sum_{i\in\mathcal{R}}\sum_{j\in\mathcal{M}_i} v_j\| \le \sqrt{p}\sum_{i\in\mathcal{R}}|\mathcal{M}_i|$ to (C.3) yields (8) and completes the proof. □

Appendix D. Proof of Theorem 1

Proof. For notational simplicity, define $x^k := [x_i^k]\in\mathbb{R}^{rp}$ as a vector that stacks all the local variables x_i^k of regular agents. Also define two functions $f^k(x^k) := \sum_{i\in\mathcal{R}} f_i^k(x_i^k)$ and $g^k(x^k) := (\lambda/2)\sum_{i\in\mathcal{R}}\sum_{j\in\mathcal{R}_i}\|x_i^k-x_j^k\|_1 + \lambda\sum_{i\in\mathcal{R}}\sum_{j\in\mathcal{M}_i}\|x_i^k-z_j^k\|_1$. With these definitions, the update of x^k in Algorithm 1 is

$$x^k = x^{k-1} - \alpha\left(\nabla f^k(x^{k-1}) + \partial g^k(x^{k-1})\right), \tag{D.1}$$

where $\partial g^k(x^{k-1})$ is a subgradient of g^k at x^{k-1}. Subtracting both sides of (D.1) by x^{k*} and taking squares, we have

$$\begin{aligned}\|x^k-x^{k*}\|^2 = \|x^{k-1}-x^{k*}\|^2 - 2\alpha\langle\nabla f^k(x^{k-1}) + \partial g^k(x^{k-1}), x^{k-1}-x^{k*}\rangle \\ + \alpha^2\|\nabla f^k(x^{k-1}) + \partial g^k(x^{k-1})\|^2.\end{aligned} \tag{D.2}$$

Now we process the second term at the right-hand side of (D.2). Using the fact $\nabla f^k(x^{k*}) + \partial g^k(x^{k*}) = 0$ to split this term as

$$\begin{aligned}&-2\alpha\langle\nabla f^k(x^{k-1}) + \partial g^k(x^{k-1}), x^{k-1}-x^{k*}\rangle \\ &= -2\alpha\langle\nabla f^k(x^{k-1}) - \nabla f^k(x^{k*}), x^{k-1}-x^{k*}\rangle \\ &\quad -2\alpha\langle\nabla\partial g^k(x^{k-1}) - \partial g^k(x^{k*}), x^{k-1}-x^{k*}\rangle.\end{aligned} \tag{D.3}$$

According to Assumptions 1 and 2, $f^k(x)$ is strongly convex with constant $m_{f^k} = \min_{i\in\mathcal{R}} m_{f_i^k}$ and has Lipschitz continuous gradients with constant $M_{f^k} = \max_{i\in\mathcal{R}} M_{f_i^k}$. Thus, we have

$$\begin{aligned} & -2\alpha\langle\nabla f^k(x^{k-1}) - \nabla f^k(x^{k*}), x^{k-1} - x^{k*}\rangle \\ & \leq -\frac{2\alpha m_{f^k} M_{f^k}}{m_{f^k} + M_{f^k}}\|x^{k-1} - x^{k*}\|^2 \\ & -\frac{2\alpha}{m_{f^k} + M_{f^k}}\|\nabla f^k(x^{k-1}) - \nabla f^k(x^{k*})\|^2. \end{aligned} \tag{D.4}$$

By the convexity of $g^k(x)$, we also have

$$-2\alpha\langle\nabla\partial g^k(x^{k-1}) - \partial g^k(x^{k*}), x^{k-1} - x^{k*}\rangle \leq 0. \tag{D.5}$$

Substituting (D.4) and (D.5) into (D.3) yields

$$\begin{aligned} & -2\alpha\langle\nabla f^k(x^{k-1}) + \partial g^k(x^{k-1}), x^{k-1} - x^{k*}\rangle \\ & \leq -\frac{2\alpha m_{f^k} M_{f^k}}{m_{f^k} + M_{f^k}}\|x^{k-1} - x^{k*}\|^2 \\ & -\frac{2\alpha}{m_{f^k} + M_{f^k}}\|\nabla f^k(x^{k-1}) - \nabla f^k(x^{k*})\|^2. \end{aligned} \tag{D.6}$$

In addition, using the fact of $\nabla f^k(x^{k*}) + \partial g^k(x^{k*}) = 0$ again, the third term at the right-hand side of (D.2) satisfies

$$\begin{aligned} & \alpha^2\|\nabla f^k(x^{k-1}) + \partial g^k(x^{k-1})\|^2 \\ & = \alpha^2\|\nabla f^k(x^{k-1}) - \nabla f^k(x^{k*}) + \partial g^k(x^{k-1}) - \partial g(x^{k*})\|^2 \\ & \leq 2\alpha^2\|\nabla f^k(x^{k-1}) - \nabla f^k(x^{k*})\|^2 + 2\alpha^2\|\partial g^k(x^{k-1}) \\ & -\partial g^k(x^{k*})\|^2. \end{aligned} \tag{D.7}$$

Substituting (D.6) and (D.7) into (D.2), we have

$$\begin{aligned} \|x^k - x^{k*}\|^2 & \leq \left(1 - \frac{2\alpha m_{f^k} M_{f^k}}{m_{f^k} + M_{f^k}}\right)\|x^{k-1} \\ & - x^{k*}\|^2 + 2\alpha^2\|\partial g^k(x^{k-1}) - \partial g^k(x^{k*})\|^2 \\ & + \left(2\alpha^2 - \frac{2\alpha}{m_{f^k} + M_{f^k}}\right)\|\nabla f^k(x^{k-1}) - \nabla f^k(x^{k*})\|^2. \end{aligned} \tag{D.8}$$

According to the definitions of $g^k(x)$ and its subgradient, we know $\|\partial g^k(x)\|^2 \leq \lambda^2 p \sum_{i\in\mathcal{R}} |\mathcal{N}_i|^2$ such that $2\alpha^2\|\partial g^k(x^{k-1}) - \partial g^k(x^{k*})\|^2 \leq 8\alpha^2\lambda^2 p \sum_{i\in\mathcal{R}} |\mathcal{N}_i|^2$. Meanwhile, because $\alpha \leq \min_k \; 1/(m_{f^k} + M_{f^k})$, the coefficients $1 - 2\alpha m_{f^k} M_{f^k}/(m_{f^k} + M_{f^k}) \geq 0$ and $2\alpha^2 - 2\alpha/(m_{f^k} + M_{f^k}) \leq 0$. Therefore, (D.8) becomes

$$\|x^k - x^{k*}\|^2 \leq \left(1 - \frac{2\alpha m_{f^k} M_{f^k}}{m_{f^k} + M_{f^k}}\right)\|x^{k-1} - x^{k*}\|^2 + 8\alpha^2\lambda^2 p \sum_{i\in\mathcal{R}} |\mathcal{N}_i|^2, \tag{D.9}$$

and consequently

$$\begin{aligned} \|x^k - x^{k*}\| & \leq \left(1 - \frac{2\alpha m_{f^k} M_{f^k}}{m_{f^k} + M_{f^k}}\right)^{1/2}\|x^{k-1} - x^{k*}\| \\ & + \left(8\alpha^2\lambda^2 p \sum_{i\in\mathcal{R}} |\mathcal{N}_i|^2\right)^{1/2} \leq c\|x^{k-1} - x^{k*}\| + d. \end{aligned} \tag{D.10}$$

Here, for notational simplicity define two constants $c := \max_k \; (1 - 2\alpha m_{f^k} M_{f^k}/(m_{f^k} + M_{f^k}))^{1/2}$ and $d := (8\alpha^2\lambda^2 p \sum_{i\in\mathcal{R}} |\mathcal{N}_i|^2)^{1/2}$. Applying the triangle inequality to (D.10) yields

$$\begin{aligned} \|x^k - x^{k*}\| & \leq c\|x^{k-1} - x^{(k-1)*}\| + c\|x^{(k-1)*} - [\tilde{x}^{(k-1)*}]\| + c\|[\tilde{x}^{(k-1)*}] \\ & - [\tilde{x}^{k*}]\| + c\|[\tilde{x}^{k*}] - x^{k*}\| + d. \end{aligned} \tag{D.11}$$

Here $[\tilde{x}^{k*}] \in \mathbb{R}^{rp}$ stacks r dynamic optimal solution $\tilde{x}^{k*}$ of (2). According to Theorem 1, $\|x_i^{k*} - \tilde{x}^{k*}\| \leq \Delta^k$, and hence $\|x^{k*} - [\tilde{x}^{k*}]\| \leq \sqrt{r}\Delta^k \leq \sqrt{r}\max_k \Delta^k$. This inequality also holds true for time $k-1$ such that $\|x^{(k-1)*} - [\tilde{x}^{(k-1)*}]\| \leq \sqrt{r}\max_k \Delta^k$. By Assumption 5, $\|\tilde{x}^{k*} - \tilde{x}^{(k-1)*}\| \leq \Theta$, which implies $\|[\tilde{x}^{k*}] - [\tilde{x}^{(k-1)*}]\| \leq \sqrt{r}\Theta$. Therefore, from (D.11) we have

$$\|x^k - x^{k*}\| \leq c\|x^{k-1} - x^{(k-1)*}\| + 2c\sqrt{r}\max_k \Delta^k + c\sqrt{r}\Theta + d \tag{D.12}$$

Multiplying c^{k-t} to the two sides of (D.12) for time $k = t$, summing up from time 1 to time k, and applying telescopic cancellation, we obtain (9) and complete the proof. □

References

[1] W. Xu, Z. Li, Q. Ling, Robust decentralized dynamic optimization, in: Proceedings of ICASSP, 2018.
[2] Q. Ling, A. Ribeiro, Decentralized dynamic optimization through the alternating direction method of multipliers, IEEE Trans. Signal Process. 62 (5) (2014) 1185–1197.
[3] A. Simonetto, A. Koppel, A. Mokhtari, G. Leus, A. Ribeiro, Decentralized prediction-correction methods for networked time-varying convex optimization, IEEE Trans. Autom. Control 62 (11) (2017) 5724–5738.
[4] M. Fazlyab, S. Paternain, V. Preciado, A. Ribeiro, Prediction-correction interior-point method for time-varying convex optimization, 2016, Arxiv:1608.07544.
[5] S. Shahin, A. Jadbabaie, Distributed online optimization in dynamic environments using mirror descent, IEEE Trans. Autom. Control 63 (3) (2018) 714–725.
[6] S. Tu, A. Sayed, Mobile adaptive networks, IEEE J. Sel. Top Signal Process. 5 (4) (2011) 649–664.
[7] F. Jakubiec, A. Ribeiro, D-MAP: distributed maximum a posteriori probability estimation of dynamic systems, IEEE Trans. Signal Process. 61 (2) (2013) 450–466.
[8] R. Cavalcante, S. Stanczak, A distributed subgradient method for dynamic convex optimization problems under noisy information exchange, IEEE J. Sel. Top Signal Process. 7 (2) (2013) 243–256.
[9] K. Zhou, S. Roumeliotis, Multirobot active target tracking with combinations of relative observations, IEEE Trans. Rob. 27 (4) (2010) 678–695.
[10] M. Ye, G. Hu, Distributed optimization for systems with time-varying quadratic objective functions, in: Proceedings of CDC, 2015.
[11] S. Rahili, W. Ren, Distributed continuous-time convex optimization with time–varying cost functions, IEEE Trans. Autom. Control 62 (4) (2017) 1590–1605.
[12] C. Enyioha, A. Jadbabaie, V. Preciado, G. Pappas, Distributed resource allocation for epidemic control, 2015, arXiv:1501.01701.
[13] T. Chen, Q. Ling, G. Giannakis, An online convex optimization approach to proactive network resource allocation, IEEE Trans. Signal Process. 65 (24) (2017) 6350–6364.
[14] M. Maros, J. Jalden, ADMM for distributed dynamic beam-forming, IEEE Trans. Signal Inf. Process. Over Netw. (2017).
[15] H. Liu, W. Shi, H. Zhu, Decentralized dynamic optimization for power network voltage control, IEEE Trans. Signal Inf.Process. Over Netw. (2017).
[16] C. Enyioha, S. Magnusson, K. Heal, N. Li, C. Fischione, V. Tarokh, On variability of renewable energy and online power allocation, IEEE Trans. Power Syst. (2017).
[17] D. Acemoglu, G. Como, F. Fagnani, A. Ozdaglar, Opinion fluctuations and disagreement in social networks, Math. Oper. Res. 38 (1) (2013) 1–27.
[18] E. Yildiz, A. Ozdaglar, D. Acemoglu, A. Scaglione, Binary opinion dynamics with stubborn agents, ACM Trans. Econ. Comput. 1 (4) (2013) 19.
[19] R. Gentz, S. Wu, H. Wai, A. Scaglione, A. Leshem, Data injection attacks in randomized gossiping, IEEE Trans. Signal Inf. Process. Netw. 2 (4) (2016) 523–538.
[20] W. Ben-Ameur, P. Bianchi, J. Jakubowicz, Robust distributed consensus using total variation, IEEE Trans. Autom. Control 61 (6) (2016) 1550–1564.
[21] A. Koppel, B. Sadler, A. Ribeiro, Proximity without consensus in online multi--agent optimization, IEEE Trans. Signal Process. 65 (12) (2017) 3062–3077.
[22] L. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Physica D 60 (1–4) (1992) 259–268.
[23] A. Chambolle, An algorithm for total variation minimization and applications, J. Math. Imaging Vis. 20 (1–2) (2004) 89–97.
[24] A. Sayed, Adaptive networks, Proc. IEEE 102 (4) (2014) 460–497.
[25] A. Nedic, A. Ozdaglar, Distributed subgradient methods for multiagent optimization, IEEE Trans. Autom. Control 54 (1) (2009) 48–61.
[26] S. Boyd, P. Diaconis, L. Xiao, Fastest mixing Markov chain on a graph, SIAM Rev. 46 (4) (2004) 667–689.
[27] B. Kailkhura, S. Brahma, P. Varshney, Data falsification attacks on consensus-based detection systems, IEEE Trans. Signal Inf. Process. Netw. 3 (1) (2017) 145–158.
[28] E. Nurellari, D. McLernon, M. Ghogho, A secure optimum distributed detection scheme in under-attack wireless sensor networks, IEEE Transactions on Signal and Information Processing over Networks, 2017.