Bias Estimation in Sensor Networks

Mingming Shi, Claudio De Persis, *Member, IEEE*, Pietro Tesi, *Member, IEEE*, and Nima Monshizadeh

***Abstract*—This article investigates the problem of estimating biases affecting relative state measurements in a sensor network. Each sensor measures the relative states of its neighbors, and this measurement is corrupted by a constant bias. We analyze under what conditions on the network topology and the maximum number of biased sensors the biases can be correctly estimated. We show that, for nonbipartite graphs, the biases can always be determined even when all the sensors are corrupted, whereas for bipartite graphs, more than half of the sensors should be unbiased to ensure the correctness of the bias estimation. If the biases are heterogeneous, then the number of unbiased sensors can be reduced to two. Based on these conditions, we propose three algorithms to estimate the biases.**

***Index Terms*—Bipartite graph, compressed sensing, estimation, linear programming, wireless sensor networks.**

I. Introduction

THE normal operation of many large-scale systems relies on networks of sensors that provide information by monitoring and managing system operating conditions [1]–[6]. However, when measuring the variables of interest, sensors may generate unreliable results due to the low quality of the hardware, environmental variations, or adversary attacks. This introduces measurement errors, which can degrade the system performance and even lead to major disruptions [5]–[11].

In this article, we consider networks in which each sensor measures the relative state of its neighbors (the difference between its state and that of its neighbors) and aim to characterize the conditions under which the biases corrupting the measurements can be accurately estimated and provide methods for their estimation. The problem is important since biases on the relative measurements can cause issues in several application domains. In sensor localization, relative measurements are used widely [8], [12]–[18]. Sensor biases of the kind considered in this article can lead to inaccurate estimates by the sensors. In the context of formation control, biases in relative measurements can result in the distortion of the formation shape and the excursion of the positions of the vehicles from the prescribed ones [10], [11], [19]–[21]. In statistical ranking, pairwise assessments are used to determine the rank of a product within a group of similars, such as movies or websites [18], [22]. Biases in these pairwise assessments may lead to incorrect ranking and mislead consumers.

Given erroneous relative measurements, providing precise estimates of the relative states can be considered as a complementary problem to the one of estimating biases. Many papers [8], [12]–[18] have provided methods for estimating the states of the sensors from noisy relative measurements by solving linear or nonlinear least-square problems. These methods cannot precisely estimate the state since the least-square approach has no robustness to the measurement error and any error can make the estimation of the unknown deviate from the actual value [24].

The formulation of the problem considered in this article covers the situation where the biases are constant but with arbitrary magnitude, thus allowing the presence of outliers. Similar problems have been addressed recently in [16] and [18], where the focus was on the state estimation problem. However, neither of the papers gives results on how the sparsity of the measurement errors affects the state estimation. On the other hand, computing biases from relative measurements received comparably less attention. Reference [25] proposed algorithms to estimate sensor offsets in wireless sensor networks. These methods only partially compensate the offsets. In problems that use the angle of arrival (AOA) measurements, if the local frame is unaligned with the global frame, then the unknown orientation of the local frame can be regarded as a bias. Ahn *et al.* [26], [27] use the consensus algorithm to estimate the orientation. However, similar to that in [25], the estimation error of their algorithms never vanishes.

In this article, we reduce the bias estimation problem to the solution of linear equations (LEs). Several algorithms have been devoted to the distributed solution of LEs, with a focus on asynchronous implementations [28], [29]; graph connectivity conditions [30]; and secure computing [31]; to name a few. However, in these algorithms, each node needs to find all entries of the vector of the unknowns, which, if employed in our problem, would require the nodes to know the network size. Instead, we exploit a suitable sparsity condition on the biases to ensure they can be uniquely determined, which is an important problem in

Manuscript received October 30, 2019; revised November 2, 2019, January 20, 2020, January 22, 2020, and March 1, 2020; accepted March 18, 2020. Date of publication April 2, 2020; date of current version September 10, 2020. The work of Mingming Shi was supported by the Chinese Scholarship Council (CSC) under Grant 201506120032. Recommended by Associate Editor Y. Wan. *(Corresponding author: Mingming Shi.)*

Mingming Shi is with the ICTEAM, Université Catholique de Louvain, B-1348 Louvain-la-Neuve, Belgium (e-mail: mingming.shi@uclouvain.be).

Claudio De Persis and Nima Monshizadeh are with the ENTEG, University of Groningen, 9747 AG Groningen, The Netherlands (e-mail: c.de.persis@rug.nl; n.monshizadeh@rug.nl).

Pietro Tesi is with the ENTEG, University of Groningen, 9747 AG Groningen, The Netherlands, and also with the DINFO, University of Florence, 50139 Firenze, Italy (e-mail: p.tesi@rug.nl).

Digital Object Identifier 10.1109/TCNS.2020.2984684

compressive sensing [32]–[39], and is related to secure state estimation [5], [40]–[43].

Several papers also investigate the problem of achieving consensus or a prescribed formation in the presence of inconsistent or biased measurements. Meng *et al.* [11] use estimators to counteract compass mismatches, while requiring each node to measure the relative positions of all the edges. De Marina *et al.* [10] address the rigid formation control problem where the agents disagree on the prescribed interagent distances. In our setup, this method would require that for each pair of adjacent nodes, at least one of the nodes is bias-free, which is a very stringent requirement. A similar setup is also adopted in [19]. For second-order consensus, Sukumar *et al.* [20] propose an adaptive compensator to prevent the state unboundedness caused by the biases. The proposed compensator cannot make the system achieve exact consensus.

Our contribution: Given relative state measurements that are affected by biases, we find conditions under which the biases are identified so that the actual relative states can be exactly reconstructed. Similar to that in [1], [8], [13], [25]–[27], and [44], we assume that biased measurements can be exchanged among the neighboring nodes. We assume that each node has one sensor; hence, the relative measurements taken by the node are affected by the same bias. In [16] and [20], biases are on the relative measurements. The form of the system of LEs to which we reduce the problem is different from the one formulated in papers involving range or AOA measurements [11], [25]–[27]. In our problem (see Section III), the biases affect the relative state measurements, whereas for problems involving range or AOA measurements, the biases affect the absolute value of or the pointing of the vector of the relative measurements (distances or bearings). The LEs of the form considered in [11], [25]–[27] also appears in papers that studied problems of sensor synchronization [45] and multiagent fault estimation [40].

We provide conditions under which the biases are uniquely determined from the proposed system of LEs. Our results answer the question: "What is the maximum number of sensor biases that can be estimated from erroneous relative state measurements?" For nonbipartite graphs, the answer is "all the nodes", and we provide a distributed algorithm to estimate the biases. In the algorithm, each sensor only needs to estimate its own bias, leading to a reduction of the computational resources and memory sizes required at each node, a solution that is different from those in [28]–[31].

For bipartite graphs, similar to secure state estimation problems [5], [40]–[43], we show that the biases can be correctly computed when less than half of the sensors are biased. Furthermore, we prove that the maximum number of biased sensors can be increased if the biases are heterogeneous. This reduces the number of unbiased sensors to only two and improves the results in secure state estimation. We provide two algorithms to compute the biases. By exploiting the heterogeneous assumption and a coordinator to coordinate the sensors, the first algorithm we propose computes the biases in a finite number of steps. To remove the coordinator and make the estimation fully distributed, in the second algorithm, we solve a relaxed ℓ_1-norm optimization problem as in [40] and [42]. We show an interesting result that the actual vector of biases is the unique solution of the ℓ_1-norm optimization problem if less than half of the sensors are biased, which does not worsen the bound on the sparsity condition of the biases for the nonrelaxed problem.

We also apply the bias estimation algorithms to a consensus problem. Different from the work in [20], we can prove that the system achieves exact consensus. Our algorithms do not require each node to measure the relative states of all the edges, in contrast to the work in [11].

The rest of the article is organized as follows. In Section II, we introduce the notation, some general notions about graphs, and few specialized results on bipartite graphs. We formulate the problem and provide a useful lemma in Section III. Section IV deals with the bias estimation algorithm for nonbipartite graphs. In Section V, we introduce the sparsity condition on biases that ensures the correctness of the bias estimation, and we provide two bias estimation algorithms and show consensus using one of the proposed algorithms. Section VI presents numerical experiments to validate the theoretical findings.

II. Preliminaries

A. Notation

For a vector $z \in \mathbb{R}^p$, $\mathrm{diag}\{z\}$ represents the diagonal matrix with the ith diagonal entry equal to the ith element of z. We denote by $\mathcal{S}_z$ the support of z, which is the set of indices that correspond to the nonzero entries of z, and by $\|z\|_0$ the 0-norm of z, which is the number of elements in $\mathcal{S}_z$. We let $\mathbb{1}_m$ and $\mathbb{0}_m$ denote the m-dimensional vectors with all elements equal to 1 and 0, respectively. Given a matrix A, A_i represents its ith row and a_{ij} represents its element in the ith row and jth column. The cardinality of a set S is denoted by $|S|$. For two sets S and M, we let $S \setminus M = \{x \in S \mid x \notin M\}$ represent the complement of M in S.

B. Graph-Theoretic Notions

For a network with n nodes, let its topology be represented by an undirected and connected graph $G = \{V, E\}$, with $V = \{1, 2, \ldots, n\}$ being the set of nodes and $E \subseteq V \times V$ be the set of edges, where $\{i, j\} \in E$, or equivalently, node i is a neighbor of node j, means that node i can receive information from node j and vice versa. We denote the set of neighbors of node i by $\mathcal{N}_i$, and let $d_i = |\mathcal{N}_i|$.

The adjacency matrix A of G is defined as $a_{ij} = 1$ if node j is the neighbor of node i and $a_{ij} = 0$ otherwise. For an undirected graph G, we can assign arbitrary orientations to the edges such that each edge $\{i, j\} \in E$ has a head and a tail. The edge-node incidence matrix $B \in \mathbb{R}^{m \times n}$ of G, with $m = |E|$, is defined as $b_{ij} = 1$ if j is the head node of the edge $i \in E$, $b_{ij} = -1$ if j is the tail node, and $b_{ij} = 0$ otherwise. The Laplacian matrix L of G is an $n \times n$ matrix given by $l_{ij} = -a_{ij}$ for $j \neq i$ and $l_{ii} = \sum_{j \in \mathcal{N}_i} a_{ij} = d_i$. Since G is undirected, then $L = B^\top B$. The incidence matrix can be decomposed as the head incidence matrix $B_+ \in \mathbb{R}^{m \times n}$ and the tail incidence matrix $B_- \in \mathbb{R}^{m \times n}$. The former is defined by letting $b_{+,ij} = 1$ if node j is the head of edge $i \in \mathcal{E}$ and 0 otherwise, whereas in the latter, $b_{-,ij} = -1$

if node j is the tail of edge $i \in \mathcal{E}$ and 0 otherwise. We also let R denote the signless edge-node incidence matrix with $r_{ij} = |b_{ij}|$. It is easy to verify that $B = B_+ + B_-$ and $R = B_+ - B_-$. Let $d = [d_1\ d_2\ \dots\ d_n]^\top$ and $D = \text{diag}\{d\}$. The matrix $A + D$ is called the signless Laplacian matrix. When G is undirected, $A + D = R^\top R$. Hence, $A + D$ is positive semidefinite and all its eigenvalues $\mu_1 \leq \mu_2 \leq \cdots \leq \mu_n$ are real and nonnegative.

A path $\mathcal{P}_{ij}$ from node i to node j is a sequence of nodes and edges such that each successive pair of nodes in the sequence is adjacent. The length of a path is the number of edges in the path. The distance between nodes i and j is the length of the shortest path from i to j. We denote by D_G the diameter of G, which is the maximum distance between any two nodes.

C. Bipartite Graphs

A graph G is bipartite if the vertex set V can be partitioned into two sets V_+ and V_- in such a way that no two vertices from the same set are adjacent. The sets V_+ and V_- are called the color classes of G, and (V_+, V_-) is a bipartition of G. For a bipartite graph, the following result holds:

Theorem 1 (See: [46]) A graph G is bipartite if and only if G has no cycle of odd length.

An algebraic property of bipartite graphs is provided next.

Lemma 1: An undirected and connected graph G is bipartite if and only if the signless incidence matrix R does not have full-column rank. Moreover, if G is bipartite, then any $n - 1$ columns of R are linearly independent.

We give a proof of this lemma in [47], and omit it here due to lack of space. For later use, from the proof in [47], we note that for a bipartite graph with bipartition (V_+, V_-), it holds

$$Rv = \mathbb{0}_n \iff \exists a \in \mathbb{R} \text{ s.t. } v_i = \begin{cases} a & i \in V_+ \\ -a & i \in V_-. \end{cases} \tag{1}$$

Lemma 2 (See: [48]) The smallest eigenvalue of the signless Laplacian matrix $A + D$ of an undirected and connected graph is equal to zero if and only if the graph is bipartite. In case the graph is bipartite, zero is a simple eigenvalue.

D. Compressed Sensing

In the field of compressed sensing or sparse signal recovery, one of the most important problems is how to find the sparsest solution from the number-deficient measurements. Formally, consider the following LE:

$$y = Fx \tag{2}$$

where $x \in \mathbb{R}^n$ is the vector of unknown variables, $y \in \mathbb{R}^p$ is the vector of known values, and $F \in \mathbb{R}^{p\times n}$ is a matrix defining the linear relation from x to y. It is assumed that $p < n$, thus (2) is underdetermined. It is then of interest to find solutions x such that $\|x\|_0 \ll n$, and in particular to seek for the sparsest solution of (2). Let us define the set of k-sparse vectors as

$$\mathcal{W}_k := \{x \in \mathbb{R}^n \mid \|x\|_0 \leq k\}. \tag{3}$$

The following result provides a sufficient condition under which the solution of (2) can be uniquely determined.

Lemma 3: Given an integer $s \geq 0$, let $2\,s \leq p$, and assume that any matrix made of $2s$ columns of F is full-column rank. If $x \in \mathcal{W}_s$ is a solution of (2), then there exists no other solution of (2) in $\mathcal{W}_s$.

Remark 1: *(On Lemma 3)* Under the assumptions of the lemma, the solution $x \in \mathcal{W}_s$ of (2) is also the solution to

$$\min_{\overline{x}\in\mathbb{R}^n} \|\overline{x}\|_0 \qquad \text{s.t. } y = F\overline{x} \tag{4}$$

that is, the sparsest solution to (2). The proof of Lemma 3 descends from [33, Lemma 1]. ■

However, solving x from (2) under the assumption that $\|x\|_0 \leq s$ is cumbersome when s is not small, as it requires to combinatorially search for s columns of F whose span contains y. A typical way to avoid this exhaustive search is to change the problem into the following ℓ_1-norm optimization problem [32]:

$$\min_{\overline{x}\in\mathbb{R}^n} \|\overline{x}\|_1 \qquad \text{s.t. } y = F\overline{x} \tag{5}$$

where y is the vector of known values in (2) and the objective function and the constraint are both convex. Problem (5) can be solved by linear programming [35]. The ℓ_1-norm minimization may return a solution $\overline{x}_*$ different from the solution x of (2). The following definition and result characterize the relation between the matrix F, (2), and the ℓ_1-norm minimization problem.

Definition 1 (Nullspace Property): A matrix $F \in \mathbb{R}^{p\times n}$ is said to satisfy the nullspace property of order s, with s being a positive integer, if for any set $S \subset V = \{1, 2, \dots, n\}$ with $|S| \leq s$ and any nonzero vector v in the nullspace of F, the following condition holds:

$$\|v_S\|_1 < \|v_{S_c}\|_1 \tag{6}$$

where $v_S \in \mathbb{R}^{|S|}$ and $v_{S_c} \in \mathbb{R}^{|S_c|}$ are subvectors of v whose elements are indexed by S and S_c, respectively, and $S_c = V \setminus S$.

The nullspace property is usually difficult to verify and a more restrictive but more conveniently checkable condition known as restricted isometry property is considered [35, p. 8]. Yet, in the special cases that are of interest to us, the nullspace property can be easily confirmed (cf., Theorem 7), and we will persist with it in the sequel.

Theorem 2: [35, Th. 2.3] Every vector $x \in \mathcal{W}_s$ is the unique solution of the ℓ_1-norm minimization problem (5), with $y = Fx$, if and only if F satisfies the nullspace property of order s.

We highlight the role of this theorem explicitly in connection with (2). For a given $y \in \mathbb{R}^p$, let $x \in \mathbb{R}^n$ be a solution of (2). Assume that $\|x\|_0 \leq s$ and F satisfies the nullspace property of order s, with $0 < s < n$. By Theorem 2, x is the unique solution of (5), with $y = Fx$. Stated directly, there exists a unique solution $\overline{x}_*$ of (5), with $y = Fx$, and it satisfies $\overline{x}_* = x$. Hence, under the given condition of s-sparsity of the vector x and the nullspace property of order s of the matrix F, solving the optimization problem (5), with $y = Fx$, univocally Returns x.

III. Problem Formulation—Biases Estimation in Sensor Networks

We consider a sensor network where each sensor is identified with a node in a measurement graph $G = (V, E)$ with V the set of nodes, $|V| = n \geq 2$ and E the set of edges. Throughout the article, we assume that G is connected and undirected. Apart from the measurement graph, there exists a communication network through which the nodes can communicate with each other without any imperfection. The communication network can have a different topology from that of the measurement graph. Nevertheless, to facilitate the presentation, we assume that the graph of the communication network is the same as the measurement graph.

A state variable $x_i \in \mathbb{R}$ is associated to each node $i \in V$. Each sensor $i \in V$ can measure the relative information $x_j - x_i$ for all $j \in \mathcal{N}_i$. We are interested in a scenario where the measurements taken by the sensor network may be subject to constant biases. As a result of the bias, the relative information read by the sensor i, will be modified as

$$z_{ij} = x_j - x_i + w_i \quad \forall j \in \mathcal{N}_i \tag{7}$$

where $w_i \in \mathbb{R}$ is an unknown constant term accounted for the bias of sensor i. In case a sensor is bias-free, we set $w_i = 0$.

The presence of biases deteriorates the performance of the network, and may even raise stability issues. Thus, it is of interest to estimate the biases, and possibly counteract their effect in the network.

To formulate the problem, we first rearrange the equalities in (7) in a suitable vector form. After assigning arbitrary orientation to G, we collect in the vector $\zeta \in \mathbb{R}^m$ all the measurements z_{ij} for which node $i \in V$ is the head of the edge $\{i, j\} \in E$, which gives $\zeta = -Bx + B_+ w$, with B_+ denoting the head incidence matrix. Similarly, we collect, in the vector $\eta \in \mathbb{R}^m$, all the measurements z_{ij} for which node $i \in V$ is the tail of the edge $\{i, j\} \in E$, and obtain $\eta = Bx - B_- w$, where B_- is the tail incidence matrix. Hence

$$z := \begin{bmatrix} \zeta \\ \eta \end{bmatrix} = \begin{bmatrix} -B \\ B \end{bmatrix} x + \begin{bmatrix} B_+ \\ -B_- \end{bmatrix} w. \tag{8}$$

Note that, by construction, we have $z \in \operatorname{im}(\mathcal{B})$, where

$$\mathcal{B} := \begin{bmatrix} -B & B_+ \\ B & -B_- \end{bmatrix}$$

and $\operatorname{im}(\cdot)$ denotes the column span of a matrix.

For a given measurement z, we are interested in finding the bias vector w in a set $\mathcal{W} \subseteq \mathbb{R}^n$ of admissible biases, which is defined more precisely later. To avoid ambiguity, we first introduce the definition of a solution of (8) with respect to w.

Definition 2 (Solution of (8) in $\mathcal{W}$): Given $z \in \operatorname{im}(\mathcal{B})$ and a set $\mathcal{W} \subseteq \mathbb{R}^n$ of admissible biases, the vector $\overline{w} \in \mathcal{W}$ solves (8) if there exists $\overline{x} \in \mathbb{R}^n$ such that (8) is satisfied with $(x, w) = (\overline{x}, \overline{w})$. In this case, we say $\overline{w}$ solves (8) in $\mathcal{W}$, or $\overline{w}$ is a solution of (8) in $\mathcal{W}$.

The uniqueness of the solution of (8) is defined in the following.

Definition 3 (Unique solution of (8) in $\mathcal{W}$): A solution $\overline{w}$ of (8) in $\mathcal{W}$ is unique if there exists no vector $\overline{w}'$, with $\overline{w}' \neq \overline{w}$, which is a solution of (8) in $\mathcal{W}$. In this case, we say $\overline{w}$ uniquely solves (8) in $\mathcal{W}$.

We then formulate the problem, which is of interest in this article.

Problem formulation: Given the vector of biased measurements $z \in \operatorname{im}(\mathcal{B})$ and a set $\mathcal{W} \subseteq \mathbb{R}^n$ of admissible biases, find conditions under which the vector of actual sensor biases w is the unique solution of (8) in $\mathcal{W}$, and design algorithms for estimating it.

Remark 2: *(Noiseless measurements)* An implicit assumption on the measurements (7) is that sensors are only subject to biases. In reality, this may be quite restrictive since the reading of sensors may be affected by noise, e.g., zero mean noise with bounded variance [1], [13], [15]. However, the goal of this article is to provide a theoretical answer to the question of quantifying the maximum number of sensors that can be biased while allowing for methods that estimate the biases. Assuming to have noiseless measurements allows us to provide a neat answer to the previous question. Since it is difficult to distinguish between the value of noise and of the bias, the presence of noise can induce errors in the bias estimation methods developed later, and even make some results (e.g., Theorem 4) inapplicable. A thorough analysis of how the noise affects the bias estimation error is important but out of the scope of the present article. Methods as those in [36, Theorem 6.12], [40] can inspire such analysis. ■

Remark 3: *(Constant versus time-varying biases)* The article focuses on constant biases. Measurements with such biases have disruptive effects on the tasks for which they are used [19], [20], [25], [49]. Constant biases can also be used to estimate (slowly) time-varying biases to first approximation. Namely, we could apply the estimation methods proposed in the ensuing sections to estimate the sensor bias at a given time and then repeat the same estimation at any sampling time. If the time-varying bias is sufficiently slow compared to the estimation speed of the algorithms, then the repeated application of the estimation process at the sampling times would lead to an approximate reconstruction of the bias. A similar idea is also presented in the context of secure state estimation and fault detection of multiagent systems [40]. Even in those cases in which we are not able to obtain an accurate estimate of time-varying biases, obtaining an estimate of the bias at a certain time and identifying the biased sensors is a useful activity for monitoring the sensor network. ■

A. Useful Lemma

Note that $\mathcal{W}$ should always contain the bias vector w and by construction at least one solution to (8) exists. Determining conditions under which the solution to (8) is unique implies we can correctly estimate the vector of actual biases affecting the measurements. To prove the uniqueness of the solution of (8), we will rely on a reduced form of (8) provided in the following result.

Lemma 4: Consider the vector of biased measurements $z \in \operatorname{im}(\mathcal{B})$ and a set $\mathcal{W} \subseteq \mathbb{R}^n$ of admissible biases. Consider the equality

$$Rw = \tilde{z} \tag{9}$$

where $R = B_+ - B_-$ is the signless edge-node incidence matrix, $\tilde{z} = Fz$, and $F = [I_m \; I_m]$ is the left annihilator of the matrix $[-B^\top \; B^\top]^\top$. Then, the following two statements hold:

i) The vector w is a solution of (8) in $\mathcal{W}$ if and only if $w \in \mathcal{W}$ is a solution of (9).
ii) The vector w is the unique solution of (8) in $\mathcal{W}$ if and only if w is the unique solution of (9) in $\mathcal{W}$.

Proof: i). (Only if) If w is a solution of (8) in $\mathcal{W}$, then premultiplying (8) by F leads to $\tilde{z} = Rw$. Hence, $w \in \mathcal{W}$ is also a solution of (9).

(If) Since w is a solution of (9) in $\mathcal{W}$, then $\zeta + \eta = Rw$, with $w \in \mathcal{W}$. Since $z \in \operatorname{im}(\mathcal{B})$, there should exist a vector $x' \in \mathbb{R}^n$ and $w' \in \mathbb{R}^n$ such that

$$z = \begin{bmatrix} -B \\ B \end{bmatrix} x' + \begin{bmatrix} B_+ \\ -B_- \end{bmatrix} w'. \tag{10}$$

Premultiplying the previous equality by F leads to $\tilde{z} = Rw'$. Combining this with $\tilde{z} = Rw$, we have $R(w' - w) = \mathbb{0}_m$. We continue the proof considering the following two distinct cases.

Case 1. G is not bipartite: Since G is not bipartite, by Lemma 1 the matrix R is full-column rank, which implies $w' = w \in \mathcal{W}$. Hence, $w \in \mathcal{W}$ is a solution of (8).

Case 2. G is bipartite: Since G is bipartite, there should exist a bipartition $V = \{V_+, V_-\}$. Let $|V_+| = p$, label the nodes in V such that $V_+ = \{1, 2, \ldots, p\}$, $V_- = \{p+1, \ldots, n\}$ and define the orientations of the edges in such a way that the head node of each edge in E belongs to V_+. Bearing in mind the identity $R(w' - w) = \mathbb{0}_m$ above, and noting (1) we have

$$w' = w + fa, \quad f = \begin{bmatrix} \mathbb{1}_p \\ -\mathbb{1}_{n-p} \end{bmatrix} \tag{11}$$

for some $a \in \mathbb{R}$. Substituting this back to (10) yields

$$z = \begin{bmatrix} -B \\ B \end{bmatrix} x' + \begin{bmatrix} B_+ \\ -B_- \end{bmatrix} w + \begin{bmatrix} B_+ \\ -B_- \end{bmatrix} fa. \tag{12}$$

To prove that w is a solution of (8) in $\mathcal{W}$, in view of Definition 2, we need to show that

$$z - \begin{bmatrix} B_+ \\ -B_- \end{bmatrix} w \in \operatorname{im} \begin{bmatrix} -B \\ B \end{bmatrix}$$

which, by (12), reduces to

$$\begin{bmatrix} B_+ \\ -B_- \end{bmatrix} f \in \operatorname{im} \begin{bmatrix} -B \\ B \end{bmatrix}. \tag{13}$$

Let B_+ and B_- be decomposed as

$$B_+ = \begin{bmatrix} \tilde{B}_+ & \mathbb{0}_{m\times(n-p)} \end{bmatrix}, \quad B_- = \begin{bmatrix} \mathbb{0}_{m\times p} & \tilde{B}_- \end{bmatrix}$$

for some matrices $\tilde{B}_+$ and $\tilde{B}_-$. Then, (13) can be written as

$$\begin{bmatrix} \tilde{B}_+ & \mathbb{0}_{m\times(n-p)} \\ \mathbb{0}_{m\times p} & -\tilde{B}_- \end{bmatrix} f \in \operatorname{im} \begin{bmatrix} -\tilde{B}_+ & -\tilde{B}_- \\ \tilde{B}_+ & \tilde{B}_- \end{bmatrix}$$

where we have used the fact that $B = B_+ + B_-$. Noting that $\tilde{B}_+ \mathbb{1}_p = -\tilde{B}_- \mathbb{1}_{n-p}$, it is easy to verify that the aforementioned relationship is satisfied since

$$\begin{bmatrix} \tilde{B}_+ & \mathbb{0}_{m\times(n-p)} \\ \mathbb{0}_{m\times p} & -\tilde{B}_- \end{bmatrix} f = \begin{bmatrix} -\tilde{B}_+ & -\tilde{B}_- \\ \tilde{B}_+ & \tilde{B}_- \end{bmatrix} \begin{bmatrix} \mathbb{0}_p \\ \mathbb{1}_{n-p} \end{bmatrix}.$$

This completes the proof of part (i).

ii). We only prove the "if" part since the converse implication can be shown similarly. Assume w is a unique solution of (9) in $\mathcal{W}$, then by (i), we have w is also a solution of (8) in $\mathcal{W}$. Now if there exists another vector $w' \in \mathcal{W}$, with $w' \neq w$ is a solution of (8), it should also be a solution of (9) by the first statement. This contradicts the uniqueness assumption. ■

The result of Lemma 4 will be used in some of the derivations of the main results in the sequel.

To study the conditions guaranteeing the uniqueness of the solution of (8) in $\mathcal{W}$, we differentiate between bipartite and nonbipartite graphs.

IV. Nonbipartite Graphs

In this section, we present the results for the case when the measurement graph G is not bipartite.

A. Condition for Correct Bias Estimation

The following result shows that w can be determined uniquely from (8) if the graph is not bipartite.

Theorem 3: Consider a graph G, let $z \in \operatorname{im}(\mathcal{B})$ be the vector of biased measurements, and $\mathcal{W} = \mathbb{R}^n$ be the set of admissible biases. Then, w is the unique solution of (8) in $\mathcal{W} = \mathbb{R}^n$ if and only if G is not bipartite.

Proof: In view of Lemma 4, we need to show that the bias vector w is the unique solution of (9) if and only if G is not bipartite. This holds since, by Lemma 1, the matrix R has full-column rank if and only if G is not bipartite. ■

B. Distributed Bias Estimation

In this section, we propose a distributed method to estimate the biases. A simple way to solve for w is to first reformulate (9) as the optimization problem

$$\min_{w \in \mathbb{R}^n} \frac{1}{2} \|Rw - \tilde{z}\|_2^2. \tag{14}$$

In fact, since for nonbipartite graphs R is full-column rank (Lemma 1), the unique minimizer is given by $w_* = (R^\top R)^{-1} R^\top \tilde{z}$.

Hence, the explicit computation of the minimizer, that is, the explicit computation of the bias vector w in (9), can be carried out via a gradient descent method

$$\dot{\hat{w}} = -R^\top R \hat{w} + R^\top \tilde{z} \tag{15}$$

where $\hat{w}$ is the estimate of w. Fortunately, in the present case, the gradient descent leads to a fully distributed algorithm. In fact, we first notice that the gradient of the objective function satisfies $R^\top (Rw - \tilde{z}) = (A + D)w - R^\top \tilde{z}$, where $-(A + D)$ is a Hurwitz matrix by Lemma 2. When we write the right-hand side of the identity earlier component by component with w replaced by $\hat{w}$, we obtain

$$[R^\top]_i \tilde{z} - (A_i + D_i)\hat{w} = \sum_{j\in\mathcal{N}_i} \tilde{z}_{ij} - d_i \hat{w}_i - \sum_{j\in\mathcal{N}_i} \hat{w}_j$$
$$= \sum_{j\in\mathcal{N}_i} (\tilde{z}_{ij} - \hat{w}_i - \hat{w}_j) = \sum_{j\in\mathcal{N}_i} (z_{ij} + z_{ji} - \hat{w}_i - \hat{w}_j) \tag{16}$$

where $[R^\top]_i$ is the ith row of $R^\top$. Hence, we conclude that, for each node $i \in V$, the estimation variable $\hat{w}_i$ evolves as

$$\dot{\hat{w}}_i = \sum_{j\in\mathcal{N}_i} (z_{ij} + z_{ji} - \hat{w}_i - \hat{w}_j). \tag{17}$$

Node i uses the biased measurements z_{ij} and z_{ji}, and the bias estimates $\hat{w}_i$ and $\hat{w}_j$. Note that the values of z_{ji} and $\hat{w}_j$ are communicated to node i via the link $\{j, i\}$. Hence, the algorithm is fully distributed.

The following result shows exponential convergence of the estimates to the actual biases.

Proposition 1: The estimate vector $\hat{w}$ generated by (17) converges exponentially fast to the vector w of the actual biases if the measurement graph G is not bipartite.

Proof: In view of the arguments earlier, bearing in mind the vectorized form (15) of (17), the proof follows immediately from Lyapunov arguments for the gradient descent algorithm using the Lyapunov function $V(\hat{w}) = \frac{1}{2}\|\hat{w} - w_*\|^2$, where $w_* = (R^\top R)^{-1} R^\top \tilde{z}$. Details are standard and, therefore, omitted. ■

An alternative way to solve for w in (9) is to use the block partition method in [16], [44], and [50]. When applied to the problem under investigation in this article, the method requires each node to estimate not only its own bias but also those of its neighbors. In contrast, the estimation dynamics (17) only requires each node to store and transmit its own estimate; hence, it reduces the memory space and communication burden.

C. Example of Use: Rejecting Biases in a Consensus Network

In this section, we investigate the possibility of removing the effect of relative state measurement biases from a consensus process. By exploiting the bias estimation method provided in the previous section, we devise a compensator that asymptotically rejects the biases. To this end, let

$$\begin{aligned}\dot{x}_i &= \sum_{j\in\mathcal{N}_i} z_{ij} + u_i^c \\ &= \sum_{j\in\mathcal{N}_i} (x_j - x_i) + d_i w_i + u_i^c \;\forall i \in V \end{aligned}\tag{18}$$

where u_i^c is an additional control input available to the designer. Note that without a proper compensation, i.e., $u_i^c = 0$, solutions of (18) can be unbounded. Let u_i^c be given by

$$u_i^c = -d_i \hat{w}_i \quad \forall i \in V \tag{19}$$

where $\hat{w}_i$ is given by (17). This results in the closed-loop dynamics

$$\begin{aligned}\dot{x}_i &= \sum_{j\in\mathcal{N}_i} y_{ij} - u_i^c \\ &= \sum_{j\in\mathcal{N}_i} (x_j - x_i + w_i - \hat{w}_i) \\ &= \sum_{j\in\mathcal{N}_i} (x_j - x_i - e_i)\end{aligned} \tag{20}$$

which can be written compactly as

$$\dot{x} = -Lx - De \tag{21}$$

where L is the Laplacian matrix of G, $e_i = \hat{w}_i - w_i$ is the estimation error for the bias w_i and $e \in \mathbb{R}^n$ is the vectorized form of all e_i. Differentiating e and using (15) and the relation $A + D = R^\top R$ lead to

$$\begin{aligned}\dot{e} &= R^\top \tilde{z} - (A + D)\hat{w} \\ &= R^\top R w - (A + D)\hat{w} \\ &= -(A + D)e\end{aligned} \tag{22}$$

where the second equality follows from (9).

In case of a nonbipartite graph, the vector of biases can be asymptotically rejected and consensus can be achieved.

Proposition 2: Let G be a nonbipartite graph. Then, solutions (e, x) of (22) and (21) exponentially converge to the point (e^*, x^*), where $x^* \in \mathrm{im}(\mathbb{1}_n)$ and $e^* = 0$. If $\hat{w}$ is initialized at zero, equivalently $e(0) = -w$, then we have

$$x_i^* = \frac{\mathbb{1}_n^\top}{n} D(A + D)^{-1} w + \frac{\mathbb{1}_n^\top x(0)}{n} \tag{23}$$

for each $i \in V$.

Proof: Equation (21) can be seen as the conventional consensus dynamics driven by the bias estimation error. Let $0 = \lambda_1 < \lambda_2 \le \lambda_3 \le \cdots \le \lambda_n$ be the eigenvalues of L along with the basis of orthonormal eigenvectors $\{\frac{\mathbb{1}_n}{\sqrt{n}}, v_2, \ldots, v_n\}$. Define $\Lambda = \mathrm{diag}\,[\lambda_1, \ldots, \lambda_n]$, $U = [\frac{\mathbb{1}_n}{\sqrt{n}}\; U_2]$ with $U_2 = [v_2 \;\cdots\; v_n]$ and apply the state transformation $z = U^\top x$. In the new coordinates, we have

$$\dot{z} = -\Lambda z - U^\top D e \tag{24}$$

where z_1 is the solution of $\dot{z}_1 = -\frac{\mathbb{1}_n^\top D}{\sqrt{n}} e$ and $z_{[2:n]} := [\, z_2 \;\ldots\; z_n\,]^\top$ follows

$$\dot{z}_{[2:n]} = -\overline{\Lambda} z_{[2:n]} - U_2^\top D e \tag{25}$$

with $\overline{\Lambda} = \mathrm{diag}\,[\lambda_2, \ldots, \lambda_n]$. By Proposition 1, if G is not bipartite, then the estimation errors satisfy

$$e(t) = e^{-(A+D)t} e(0) \tag{26}$$

from which we have

$$z_1(t) = -\frac{\mathbb{1}_n^\top}{\sqrt{n}} D(A + D)^{-1}(1 - e^{-(A+D)t})e(0) + z_1(0)$$

which shows $\lim_{t\to+\infty} z_1(t) = -\frac{\mathbb{1}_n^\top}{\sqrt{n}} D(A + D)^{-1} e(0) + z_1(0)$. Since $\overline{\Lambda} > 0$, the vector $z_{2:n}(t)$ converges to zero exponentially fast. Hence, we find that x exponentially converges to $c\mathbb{1}_n$ for some $c \in \mathbb{R}$. It is easy to see that

$$c = \frac{1}{\sqrt{n}} \lim_{t\to+\infty} z_1(t).$$

If $e(0) = -w$, then $c = x_i^*$ given by (23), for each $i \in V$. ■

Although the system with bias compensation achieves consensus, the exact consensus value to which the agents converge is not predictable since it depends both on the initial state and the bias of the sensors. For those problems where it is of primary interest to converge to the average consensus, alternatively one can first run the algorithm (17) over a sufficiently large time horizon to obtain a sufficiently accurate estimate of the biases, and then directly remove the biases from the measurements used in the consensus process.

V. Bipartite Graphs

In this section, we consider the case where the measurement graph G is bipartite.

A. Conditions for Bias Estimation

For bipartite graphs, the following result gives a general condition that ensures that the vector of biases can be correctly estimated from the measurement (8).

Theorem 4: Consider a bipartite graph G, if a vector w solves (8) in[1] $\mathcal{W}_k$, with $k = \lfloor \frac{n-1}{2} \rfloor$, then it uniquely solves (8) in $\mathcal{W}_k$.

Proof: Since G is bipartite, by Lemma 1, any submatrix of R with $n-1$ columns has full-column rank. Hence, by Lemma 3, if there exists a solution $w \in \mathcal{W}_k$ of (9), then it is unique in $\mathcal{W}_k$. The proof ends by noticing that if w is a unique solution of (9) in $\mathcal{W}_k$, then it is the unique solution of (8) in $\mathcal{W}_k$ (see Lemma 4). ■

To ensure uniqueness of the solution in (8), approximately half of the sensors are required to be bias-free by Theorem 4. Similar requirements have been observed in the problem of secure state estimation [5], [43] and distributed classification [51]. Next, we introduce rather mild restrictions on the admissible set of biases $\mathcal{W}$ in order to obtain more relaxed conditions on the number of bias-free sensors.

Definition 4:

i) The set $\mathcal{W}_k^h$, with $2 \le k \le n$, of heterogeneous k-sparse bias vectors is the set of all vectors $w \in \mathcal{W}_k$ such that their nonzero entries are different from each other, namely $w_i \neq w_j$ for any $i, j \in V$ with $w_i \neq 0$ and $w_j \neq 0$.

ii) The set $\mathcal{W}_k^a$, with $2 \le k \le n$, of *absolutely* heterogeneous k-sparse bias vectors is the set of all vectors $w \in \mathcal{W}_k$ such that their nonzero entries in *absolute value* are different from each other, namely $|w_i| \neq |w_j|$ for any $i, j \in V$ with $w_i \neq 0$ and $w_j \neq 0$.

Note that we have $\mathcal{W}_k^a \subset \mathcal{W}_k^h \subset \mathcal{W}_k$, for each $k = 2, 3, \dots, n$.

Theorem 5: Consider a bipartite graph G,

i) If there exists w that solves (8) in $\mathcal{W}_{n-3}^h$, then it uniquely solves (8) in $\mathcal{W}_{n-3}$.

ii) If there exists w that solves (8) in $\mathcal{W}_{n-2}^a$, then it uniquely solves (8) in $\mathcal{W}_{n-2}$.

Proof: Noting Lemma 4, we work with equation (9) to prove uniqueness of the solution.

i) We prove this part by contradiction. Suppose there exists another solution $w' \neq w$ of (9), satisfying $w' \in \mathcal{W}_{n-3}$. Then

$$R(w - w') = 0. \tag{27}$$

By (1), this implies that $w = w' + fa$, where f is given by (11) and $a \in \mathbb{R}$.

Let $\mathcal{S}_w$ and $\mathcal{S}_{w'}$ be the support of w and w'. If $V \setminus (\mathcal{S}_w \cup \mathcal{S}_{w'})$ is nonempty, i.e., there exists at least one index $i \in V$ such that $w_i - w'_i = 0$, then $a = 0$. This implies $w = w'$ and leads to a contradiction. If $\mathcal{S}_w \cup \mathcal{S}_{w'} = V$, we have that $\mathcal{S}_w \setminus \mathcal{S}_{w'} = (\mathcal{S}_w \cup \mathcal{S}_{w'}) \setminus \mathcal{S}_{w'} = V \setminus \mathcal{S}_{w'}$ should have at least three elements since $\|w'\|_0 \le n-3$. However, this would imply that there exist at least three distinct indices $i, j, k \in \mathcal{S}_w \setminus \mathcal{S}_{w'}$, such that each one of w_i, w_j, w_k is either equal to a or $-a$, with $a \neq 0$. Hence, at least two elements in the set $\{w_i, w_j, w_k\}$ must be the same, which contradicts the heterogeneity assumption $w \in \mathcal{W}_{n-3}^h$. This completes the proof of uniqueness for part (i).

ii) Suppose by contradiction that there exists another solution $w' \neq w$ of (9), satisfying $w' \in \mathcal{W}_{n-2}$. Analogous to the proof of a), if $V \setminus (\mathcal{S}_w \cup \mathcal{S}_{w'})$ is nonempty, then $w = w'$, whereas if $\mathcal{S}_w \cup \mathcal{S}_{w'} = V$, the set $\mathcal{S}_w \setminus \mathcal{S}_{w'}$ has at least two elements since $\|w\|_0 \le n-2$. This would imply that there exist at least two distinct indices $i, j \in \mathcal{S}_w \setminus \mathcal{S}_{w'}$, such that each one of w_i and w_j is equal to either a or $-a$, with $a \neq 0$. This results in $|w_i| = |w_j|$, thus contradicting the absolute heterogeneity assumption $w \in \mathcal{W}_{n-2}^a$. This completes the proof of (8). ■

Thus, focusing the attention on the class of heterogeneous biases in the sense of Definition 4 considerably increases the number of allowable biased sensors.

Remark 4: *(On heterogeneous biases)* The results in Theorem 5 hinge on the assumptions that the (absolute) values of the nonzero biases are heterogeneous. Violation of these assumptions makes the conclusions in Theorem 5 invalid. However, the violation would imply that at least two nonzero biases have the same (absolute) value, which is an unlikely event.

Hence, assuming heterogeneous biases is a mild assumption. A similar assumption is also adopted in [52, Remark 7], which, differently from this article, studies the sparse recovery property of the vertex-edge incidence matrix of graphs.

B. Distributed Bias Computation With a Coordinator

In this section, we focus on algorithms for computing the actual vector of biases w. We propose the use of a coordinator that delegates the computation of the biases to the nodes while organizing the execution of their commands. Compared to a centralized solution, the distributed computation with a coordinator eases the analysis and does not require to know the network topology.

We consider the case when $w \in \mathcal{W}_{n-2}^a$ and use the result established in Theorem 5(ii). When $w \in \mathcal{W}_{n-2}^a$, there exist at least two (bias-free) nodes $i, j \in V$, $i \neq j$, satisfying $w_i = w_j = 0$. The essence of the algorithm here is to find such a bias-free pair. To this end, some additional notation is needed. For a pair of nodes $i, j \in V$ with $i \neq j$, let $\mathcal{P}_{ij}$ be a path connecting them, namely $\mathcal{P}_{ij} = \{k_0, k_1, \dots, k_{d_{ij}}\}$, with $k_0 = i$, $k_{d_{ij}} = j$, and d_{ij} the length of the path. Moreover, we collect the measurements that are indexed by $\mathcal{P}_{ij}$ as

$$Z_{ij} := \begin{bmatrix} z_{k_0 k_1} + z_{k_1 k_0} \\ z_{k_1 k_2} + z_{k_2 k_1} \\ \vdots \\ z_{k_{d_{ij}-1} k_{d_{ij}}} + z_{k_{d_{ij}} k_{d_{ij}-1}} \end{bmatrix}. \tag{28}$$

[1] Recall the definition (3) of the set $\mathcal{W}_k$ of k-sparse vectors.

Finally, we let

$$e_{d_{ij}} = [(-1)^{d_{ij}-1} \ (-1)^{d_{ij}-2} \ \dots \ (-1)^1 \ (-1)^0]^\top. \quad (29)$$

We then have the following result.

Proposition 3: Consider a bipartite graph G, let w be the vector of biases and assume that $w \in \mathcal{W}^a_{n-2}$. For a given pair of nodes $i, j \in V$, with $i \neq j$, and a path $\mathcal{P}_{ij}$ connecting them, we have the following:

i) $I_{ij} := e^\top_{d_{ij}} Z_{ij} = 0$ if and only if $w_i = w_j = 0$, i.e., the pair $i, j \in V$ is bias-free.

ii) If $w_i = 0$, then $I_{ij} = w_j$.

iii) $I_{ik_\ell} = -I_{ik_{\ell-1}} + (z_{k_{\ell-1}k_\ell} + z_{k_\ell k_{\ell-1}})$ for $\ell \in \{2, \dots, d_{ij}\}$, where $I_{ik_\ell}, I_{ik_{\ell-1}}$ are defined similarly to I_{ij}.

Proof: i) By (7), the vector Z_{ij} equals

$$Z_{ij} = \begin{bmatrix} w_{k_0} + w_{k_1} \\ \vdots \\ w_{k_{d_{ij}-1}} + w_{k_{d_{ij}}} \end{bmatrix} \quad (30)$$

from which

$$I_{ij} = e^\top_{d_{ij}} Z_{ij} = \sum_{\ell=1}^{d_{ij}} (-1)^{d_{ij}-\ell}(w_{k_{\ell-1}} + w_{k_\ell})$$
$$= (-1)^{d_{ij}-1} w_{k_0} + w_{k_{d_{ij}}} = (-1)^{d_{ij}-1} w_i + w_j. \quad (31)$$

Noting $w \in \mathcal{W}^a_{n-2}$, we find that $e^\top Z_{ij} = 0$ if and only if $w_i = w_j = 0$, as claimed.

ii) By (31), we immediately obtain that $I_{ij} = w_j$ if $w_i = 0$.

iii) The conclusion is straightforward to obtain by the definition of I_{ij} and (29). ■

From Proposition 3(i), no matter along which path the quantity I_{ij} is computed, the identity $I_{ij} = 0$ holds if and only if the pair $i, j \in V$ is bias-free. Hence, I_{ij} is an indicator of whether or not a pair of nodes are bias-free. In addition, by Proposition 3(iii), if node $k \in \mathcal{N}_j$ knows I_{ij}, then it can compute I_{ik}. In turn, by Proposition 3(ii), if $w_i = 0$, then the variable I_{ik} equals the bias w_k. Based on Proposition 3, searching the bias-free nodes and solving the bias can be concurrently carried out by the nodes in a distributed fashion coordinated by a coordinator. The idea is to let the coordinator make $n-1$ selections of a candidate bias-free node i and let the other nodes j compute the variables I_{ij} with respect to the selected node. As soon as a zero I_{ij} is observed at a node j, then that node informs the coordinator to terminate the search. At this stage, every node has computed the value of its bias via the indicator variable, namely $I_{ij} = w_j$.

The commands executed by the coordinator are summarized in Algorithm 1, whereas the commands executed by the nodes are listed in Algorithm 2. Algorithm 2 comprises two stages, *the node pair test stage*, in which the coordinator and the nodes cooperate to check whether or not a given pair of nodes is bias-free, and *the bias computing stage* during which the biases are explicitly computed. In Algorithm 2, we assume that each node has access to the data $\{z_{ij} + z_{ji}\}_{j \in \mathcal{N}_i}$, which can be achieved by letting all the nodes collect the measurements from their neighbors, before running Algorithms 1 and 2.

Algorithm 1: Coordinator.

```
Data: Set of nodes V and counter T;
Initialize: T := 0;
for i = 1 : n − 1 do
    Inform all the nodes in V to start the Node pair test
      stage in Algorithm 2;
    Inform node i that it is selected and nodes j ∈ V \ i
      that they need to calculate and send back the
      variable I_ij to the coordinator;
    T = T + 1;
    Once Node pair test stage is completed by all the
      nodes, receive I_ij and t_j from all j ∈ V \ i;
    Compute T = T + max_{j∈V\{i}} {t_j};
    if there exists one I_ij = 0 then
        Stop the for iteration;
    end if
end for
Inform all the nodes to start the Bias computing stage;
```

Algorithm 2: Node j.

```
Data: Set of neighbors N_j, measurement data
  {z_jk + z_kj}_{k∈N_j} and counter t_j;
if informed to start the Node pair test stage then
    /* Node pair test stage */
    if node j is selected in iteration i, i.e. j = i then
        Set the auxiliary variable I_jj = 0 and t_j = 1;
        Send (I_jj, t_j) to all k ∈ N_j;
        Stop accepting data from the neighbors;
    else
        Once (I_ik, t_k), for some k ∈ N_j, are received,
          pick any one of (I_ik, t_k) and compute
          I_ij := −I_ik + (z_jk + z_kj), t_j = t_k + 1;
        Send (I_ij, t_j) to all k ∈ N_j and the coordinator;
        Stop accepting data from the neighbors;
    end if
if informed to start the Bias computing stage then
    /* Bias computing stage */
    w_j = I_ij;
end if
```

To measure the number of executed instructions required by the algorithms to terminate the computation, we introduce counters that store integer values. In Algorithm 1, the sequence of actions by the coordinator consisting of informing node i that it has been selected, and asking nodes $j \in V \setminus i$ to calculate and send back the variable I_{ij} is considered as one instruction, which increases the counter T by 1 unit. The single action of informing all the nodes to start the *Bias computing stage*, is regarded as another instruction, and again results in an increase of T by 1 unit. In Algorithm 2, at each iteration i, the variable t_j, $j \in V$, stores the number of instructions executed from the moment that node i is selected by the coordinator till when j computes I_{ij}. The counters t_j, $j \in V$, are communicated to the coordinator and used to update the counter T, which, therefore, contains the total number of instructions executed before the bias-free node pair is found. Note that the counters are only introduced to store the number of instructions needed for the computation of the solution, as formalized in Theorem 6, but do not play any role in the computation of the solution itself.

The following result summarizes the properties of the algorithms.

Theorem 6: Consider a bipartite graph G, with its diameter given by D_G, let w be the vector of biases and assume that $w \in \mathcal{W}^a_{n-2}$. If the coordinator uses Algorithm 1 and the nodes Algorithm 2, then a bias-free node can be identified in T instructions and the vector of biases w can be reconstructed in $T+2$ instructions with $T \leq (n-1)(D_G+2)$.

We omit the proof of this theorem due to space limitations and we refer the reader to [47]. A few remarks are in the following order.

1) If $w \in \mathcal{W}^a_n \setminus \mathcal{W}^a_{n-2}$, so that the assumption $w \in \mathcal{W}^a_{n-2}$ in Theorem 6 is not satisfied, then $I_{ij} = 0$ will not be observed at any node, and the coordinator infers that there is no pair of bias-free nodes.
2) If $w \in \mathbb{R}^n \setminus \mathcal{W}^a_n$, namely there exist at least two sensors whose biases are nonzero with the same absolute value, the central coordinator would deem those sensors to be bias-free, which would lead to an incorrect bias estimation result. However, the situation in which $w \in \mathbb{R}^n \setminus \mathcal{W}^a_n$ is highly unlikely, as mentioned in Remark 4.
3) In Algorithm 1, the coordinator is only responsible for coordinating the nodes, namely initializing each iteration, whereas all computations are performed at the nodes in a distributed fashion. Moreover, note that the coordinator does not need to know the topology of the network, apart from the node set V.
4) Another method to compute the vector of biases when $w \in \mathcal{W}^a_{n-2}$ is to combinatorially search the pair of nodes that is bias-free, as in [5] and [53]. Specifically, for each pair of indices $i, j \in V$, with $i \neq j$, one could look for a solution of the modified equation $Rw^{(i,j)} = \tilde{z}$, where $w^{(i,j)}$ is a vector whose entries i and j are set to zero. If a solution to this modified equation exists, then by construction, it satisfies the sparsity condition $\|w^{(i,j)}\|_0 \leq n-2$, and by Theorem 5(ii), it will be equal to the vector of actual biases. Hence, the determination of the vector of biases w satisfying (8) is reduced to considering the $n(n-1)/2$ systems of equations and check if each of these equations admit a solution. Note, however, that such an approach would require that the unit carrying out the combinatorial search has access to the network topology and possesses enough computational power.

C. Distributed Bias Estimation Without Coordinator

In the previous section, we assumed the existence of a coordinator that supervises the nodes checking the conditions of Proposition 3. In this section, we seek a method that estimates the biases in a distributed manner without resorting to a coordinator, since communication with the coordinator may not exist [10], [19], [26], or be inefficient when the size of the sensor network is large [14], [16]. We show that this is achievable provided that we restrict the class of admissible biases. To this end, by Section II-D and (9), we consider the following ℓ_1-norm minimization problem:

$$\min_{\overline{w}\in\mathbb{R}^n} \|\overline{w}\|_1, \qquad \text{s.t. } R\,\overline{w} = \tilde{z} \tag{32}$$

where $\tilde{z}$ is the vector of known values appearing in (9). As mentioned in Section II-D, solving the ℓ_1-norm minimization problem may yield a solution that is different from the vector of actual biases w. The sparsity condition under which the solution of (32) coincides with w is provided in the following theorem.

Theorem 7: For a bipartite graph G, the vector of biases w is the unique solution of the ℓ_1-norm minimization problem (32) if the number of biased sensors is not greater than $\lfloor \frac{n-1}{2} \rfloor$, i.e., $w \in \mathcal{W}_{\lfloor \frac{n-1}{2} \rfloor}$.

Proof: Since the graph is bipartite, then (1) holds. Hence, inequality (6), in this case, is given by

$$\sum_{i\in S, |S|=s} |v_i| < \sum_{j\in S_c} |v_j| \iff s|a| < (n-s)|a|, \ a \neq 0$$

which is satisfied if and only if $s < \frac{n}{2}$. Hence, the matrix R satisfies the nullspace property of order s, with $s = \lfloor \frac{n-1}{2} \rfloor$.

Therefore, by (9), Theorem 2 and the discussion following it, if the vector of biases w in (9) satisfies $w \in \mathcal{W}_{\lfloor \frac{n-1}{2} \rfloor}$, then there exists a unique solution of the optimization problem (32), with $\tilde{z} = Rw$, and it is equal to w. ■

This theorem shows that for bipartite graphs, the ℓ_1-norm minimization does not decrease the maximum number of allowed biased sensors obtained in Theorem 4. On the other hand, in the case where the vector of biases w belongs to the set of heterogeneous biases $\mathcal{W}^h_{n-3}$ or $\mathcal{W}^a_{n-2}$ considered in Theorem 5, examples can be found where the solution of the ℓ_1-norm minimization problem does not give the correct bias estimation. Hence, in the following, we only discuss the solution of (32) for the case of bipartite graphs with a number of biased sensors as characterized in Theorem 7.

The ℓ_1-norm optimization problem (32) can be solved directly in a distributed manner by the methods in [39] and [54]. In this article, we reformulate it as a linear programming problem as [38]

$$\min_{\eta\in\mathbb{R}^{2n}} \mathbb{1}^\top_{2n}\eta, \qquad \text{s.t. } H\eta = \tilde{z}, \ \eta \geq 0$$

where η is the decision variable and $H = [R, -R]$. Under the sparsity condition in Theorem 7, if η^* is the solution of (33), the vector of biases can be computed as $w = \begin{bmatrix} I_n & -I_n \end{bmatrix}\eta^*$.

The aforementioned linear programming problem can be solved by various distributed methods available in the literature, see, e.g., [55]–[57]. In particular, using the result of [57], the bias estimation dynamics takes the form

$$\begin{aligned}
\hat{w} &= \begin{bmatrix} I_n & -I_n \end{bmatrix}\eta \\
\dot{\eta}_i &= \begin{cases} f_i(\eta,\lambda), & \text{if } \eta_i > 0 \\ \max\{0, f_i(\eta,\lambda)\} & \text{if } \eta_i = 0 \end{cases}, \ i \in V \\
\dot{\lambda} &= H\eta - \tilde{z}
\end{aligned} \tag{33}$$

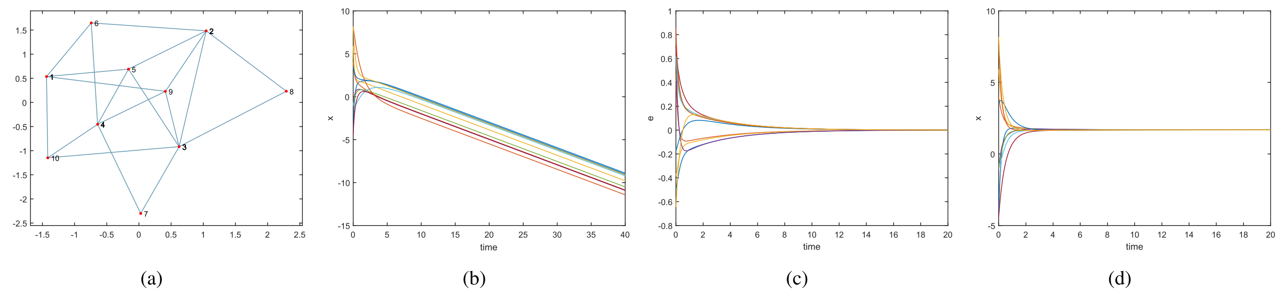


Fig. 1. Bias estimation and consensus evolution for a nonbipartite graph. (a) Topology for a ten-node nonbipartite graph. (b) State evolution of the consensus dynamics (18) without bias compensation. (c) Bias estimation error e generated by bias estimator (17). (d) State evolution of the consensus dynamics (18) with bias compensator (19).

with $f(\eta, \lambda) = -\mathbb{1}_{2n} - H^\top(\lambda + H\eta - \tilde{z})$, and where $\lambda \in \mathbb{R}^m$ is the dual variable and the initial condition satisfies $\eta_i(0) \geq 0$ for all $i \in V$.

For this method, we have the following result.

Proposition 4: The estimate $\hat{w}$ generated by the bias estimation method (33) converges asymptotically to the vector of biases w if G is bipartite and $w \in \mathcal{W}_{\lfloor \frac{n-1}{2} \rfloor}$.

Proof: This result follows directly from [57, Proposition IV.4] noting that the linear program (33) has a unique solution. ■

Remark 5: *(Compensating for biases in bipartite graphs)* Similarly to Section IV-C, one could use the estimate $\hat{w}$ generated by the algorithm (33) in the compensator (19) to reject the effect of the biases and achieve consensus. In fact, the consensus dynamics (21) driven by the estimation error e continues to be valid and an analysis similar to the one in Proposition 2 can be carried out. In the case of bipartite graphs, however, we cannot provide the estimate of the new consensus value, due to the lack of the exponential convergence of the estimation error. ■

Remark 6: *(Comparison of the methods: With and without coordinator)* Both the bias estimation method (33) and Algorithms 1 and 2 are proposed for bipartite graphs. The former requires more than half of the sensors to be unbiased, whereas the latter needs only two bias-free sensors. Moreover, compared with Algorithms 1 and 2, which is implemented in discrete-time, the bias estimator in (33) evolves in continuous-time. ■

For the problem at hand, the algorithm (33) has some advantages when compared with possible alternatives, such as the one provided in the recent paper [54], where a new distributed algorithm for solving the ℓ_1-norm minimization problem with linear equality constraints is proposed. However, in this method, each node needs to reconstruct all the elements of the solution of the ℓ_1-norm minimization problem, which implies that each node stores and communicates a vector with the same dimension as the (unknown) solution. Moreover, an implicit requirement for the method in [54] is that each agent must know the number of columns of the coefficient matrix, which translates to knowing the network size in our setting. In the method given by (33), on the other hand, each node reconstructs only one element of w by communicating suitable variables with its neighbor. The latter is done without relying on any global information including the size of the network.

VI. Numerical Simulations

In this section, we provide numerical simulations to illustrate the results for bias estimation and compensation for both nonbipartite graph and bipartite graph.

A. Nonbipartite Graphs

We consider a network with ten nodes and each node takes a sensor. The associated graph is nonbipartite and given in Fig. 1(a). The initial state $x_i(0)$ and the bias w_i of each node are generated randomly within the intervals $[-10, 10]$ and $[-1, 1]$, respectively.

We simulate the consensus dynamic (18) with the bias estimator (17) and the bias compensator (19), where the initial condition for the bias estimate is $\hat{w} = \mathbb{0}_{10}$. The numerical simulation for a specific example is provided in Fig. 1, where Fig. 1(b) and (d) shows the system state evolution without bias compensation and with bias compensation, respectively, and Fig. 1(c) shows the bias estimation error e. As can be seen in Fig. 1(b), if the biases are not compensated, the nodes will not achieve exact consensus and the state of each node x_i drifts away under the influence of the measurement biases. On the contrary, using the bias estimator (17) and the compensator (19), the bias error e vanishes and all x_i variables converge to the same finite value.

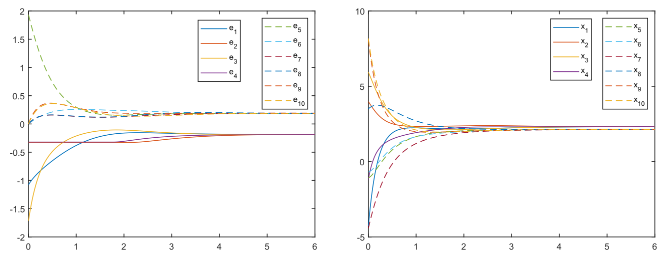
B. Bipartite Graphs

Now, we consider a bipartite graph, which is obtained from the graph in the last section removing the edge $\{2, 3\}$. In this case, one can verify that (V_+, V_-), with $V_+ = \{1, 2, 3, 4\}$ and $V_- = \{5, 6, 7, 8, 9, 10\}$, is a bipartition of the graph.

We first show that if more than $\lfloor \frac{n-1}{2} \rfloor$ sensors of nodes are biased, the ℓ_1 minimization (32) may fail to find the vector of the actual biases w for bipartite graphs. We assume that the sensors of the first five nodes are biased and

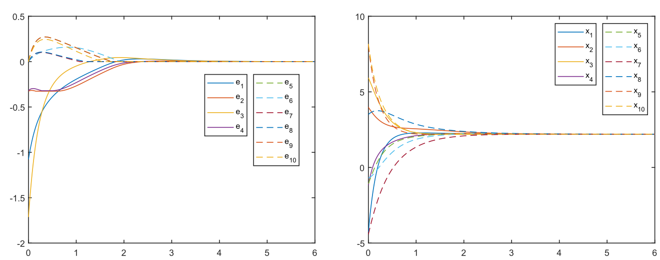
$$w = [1.076\ 0.326\ 1.713\ 0.320\ \ -1.932\ 0\ 0\ 0\ 0\ 0]^\top. \tag{34}$$

We simulate the consensus dynamics (18) with the bias estimator (33) and the bias compensator (19). The initial conditions for η and λ are set to zero. The result is given in Fig. 2, where the solid lines and dashed lines represent the nodes in V_+ and V_-, respectively. From Fig. 2, one can see that the entries of the bias estimation error e corresponding to V_+ and V_- converge to two



(a) (b)

Fig. 2. Bias estimation and consensus evolution for a ten-node bipartite graph, with the bipartition $V_+ = \{1, 2, 3, 4\}$ and $V_- = \{5, 6, 7, 8, 9, 10\}$. Five sensors are biased; hence, the condition of Theorem 7 is violated. The nodes apply the bias estimator (33) and the bias compensator (19). (a) Bias estimation error. (b) State evolution.



(a) (b)

Fig. 3. Bias estimation and consensus evolution for a ten-node bipartite graph, with the bipartition $V_+ = \{1, 2, 3, 4\}$ and $V_- = \{5, 6, 7, 8, 9, 10\}$. Four sensors are biased; hence, the condition of Theorem 7 is satisfied. The nodes apply the bias estimator (33) and the bias compensator (19). (a) Bias estimation error e. (b) State evolution with compensation.

values with the same absolute value but opposite signs; thus, the biases are not correctly estimated and consensus is not achieved.

We then let the sensor of the fifth node also to be unbiased, namely the last six entries of w in (34) are all zero. The condition of Theorem 7 is now satisfied. The result is depicted in Fig. 3, which shows that the bias estimation error decays to zero and the system achieves consensus.

VII. Conclusion

In this article, we studied the problem of estimating the biases in sensor networks from relative state measurements, with an application to the problem of consensus with biased relative state measurement. Without any sparsity constraint on the biases, we show that the biases can be accurately estimated if and only if the graph is nonbipartite. For bipartite graphs, we show that the biases can be uniquely determined from the measurements if less than half of the sensors are biased. The number of biased sensors can be increased when the biases are heterogeneous, i.e., different from each other, or absolutely heterogeneous, i.e., with absolute values different from each other. For both nonbipartite and bipartite graphs, we propose distributed methods to compute the biases.

The problem considered in this article can be further investigated. First, if the sensors are affected by noise in addition to biases, one could study how noise impacts the accuracy of the estimation of the biases [53]. Second, the result for bipartite graphs could be also used for problems where the range and AOA measurements are affected by biases [11], [25]–[40]. Finally, given a sensor network, we do not have a way to identify *a priori* to which admissible set the biases (see Definition 2) belong. This poses another question for future research.

References

[1] V. Kekatos and G. B. Giannakis, "Distributed robust power system state estimation," *IEEE Trans. Power Syst.*, vol. 28, no. 2, pp. 1617–1626, May 2013.
[2] A. Mitra and S. Sundaram, "Distributed observers for LTI systems," *IEEE Trans. Autom. Control*, vol. 63, no. 11, pp. 3689–3704, Nov. 2018.
[3] I. Coddington, W. C. Swann, L. Nenadovic, and N. R. Newbury, "Rapid and precise absolute distance measurements at long range," *Nature Photon.*, vol. 3, no. 6, pp. 351–356, 2009.
[4] F. Fagnani, S. Fosson, and C. Ravazzi, "A distributed classification/estimation algorithm for sensor networks," *SIAM J. Control Optim.*, vol. 52, pp. 189–218, 2014.
[5] H. Fawzi, P. Tabuada, and S. N. Diggavi, "Secure estimation and control for cyber-physical systems under adversarial attacks," *IEEE Trans. Autom. Control*, vol. 59, no. 6, pp. 1454–1467, Jun. 2014.
[6] J. Cortés, "Global and robust formation-shape stabilization of relative sensing networks," *Automatica*, vol. 45, pp. 2754–2762, 2009.
[7] Y. Mo and B. Sinopoli, "On the performance degradation of cyber-physical systems under stealthy integrity attacks," *IEEE Trans. Autom. Control*, vol. 61, no. 9, pp. 2618–2624, Sep. 2016.
[8] S. Zhao and D. Zelazo, "Localizability and distributed protocols for bearing-based network localization in arbitrary dimensions," *Automatica*, vol. 69, pp. 334–341, 2016.
[9] X. Li and A. Scaglione, "Robust decentralized state estimation and tracking for power systems via network gossiping," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 7, pp. 1184–1194, Jul. 2013.
[10] H. G. de Marina, M. Cao, and B. Jayawardhana, "Controlling rigid formations of mobile agents under inconsistent measurements," *IEEE Trans. Robot.*, vol. 31, no. 1, pp. 31–39, Feb. 2015.
[11] Z. Meng, B. D. O. Anderson, and S. Hirche, "Formation control with mismatched compasses," *Automatica*, vol. 69, pp. 232–241, 2016.
[12] P. Barooah and J. Hespanha, "Estimation on graphs from relative measurements," *IEEE Control Syst.*, vol. 27, no. 4, pp. 57–74, Aug. 2007.
[13] A. Carron, M. Todescato, R. Carli, and L. Schenato, "An asynchronous consensus-based algorithm for estimation from noisy relative measurements," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 3, pp. 283–295, Sep. 2014.
[14] S. Safavi, U. A. Khan, S. Kar, and J. M. F. Moura, "Distributed localization: A linear theory," *Proc. IEEE*, vol. 106, no. 7, pp. 1204–1223, Jul. 2018.
[15] B. Wang and Y.-P. Tian, "Distributed network localization: Accurate estimation with noisy measurement and communication information," *IEEE Trans. Signal Process.*, vol. 66, no. 22, pp. 5927–5940, Nov. 2018.
[16] N. Bof, M. Todescato, R. Carli, and L. Schenato, "Robust distributed estimation for localization in lossy sensor networks," *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 250–255, 2016.
[17] C. Ravazzi, P. Frasca, H. Ishii, and R. Tempo, "A distributed randomized algorithm for relative localization in sensor networks," in *Proc. Eur. Control Conf.*, 2013, pp. 1776–1781.
[18] C. Ravazzi, N. P. K. Chan, and P. Frasca, "Distributed estimation from relative measurements of heterogeneous and uncertain quality," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 5, no. 2, pp. 203–217, Jun. 2019.
[19] F. Liu, A. S. Morse, and B. D. O. Anderson, "Robust control of undirected rigid formations with constant measurement bias in relative positions," in *Proc. Amer. Control Conf.*, 2016, pp. 143–147.
[20] S. Sukumar, E. Panteley, A. Loría, and W. Pasillas, "On consensus of double integrators over directed graphs and with relative measurement bias," in *Proc. IEEE Conf. Decision Control*, 2018, pp. 4147–4152.
[21] S. Knorn, A. Donaire, J. C. Agüero, and R. H. Middleton, "Scalability of bidirectional vehicle strings with static and dynamic measurement errors," *Automatica*, vol. 62, pp. 208–212, 2015.
[22] X. Jiang, L.-H. Lim, Y. Yao, and Y. Ye, "Statistical ranking and combinatorial Hodge theory," *Math. Program.*, vol. 127, no. 1, pp. 203–244, Mar. 2011.

[23] I. Shames, A. N. Bishop, and B. D. O. Anderson, "Analysis of noisy bearing-only network localization," *IEEE Trans. Autom. Control*, vol. 58, no. 1, pp. 247–252, Jan. 2013.
[24] Y. Sharon, J. Wright, and Y. Ma, "Minimum sum of distances estimator: Robustness and stability," in *Proc. Amer. Control Conf.*, Jun. 2009, pp. 524–530.
[25] S. Bolognani, S. Del Favero, L. Schenato, and D. Varagnolo, "Consensus-based distributed sensor calibration and least-square parameter identification in WSNs," *Int. J. Robust Nonlinear Control: IFAC-Affiliated J.*, vol. 20, no. 2, pp. 176–193, 2010.
[26] B.-H. Lee and H.-S. Ahn, "Distributed formation control via global orientation estimation," *Automatica*, vol. 73, pp. 125–129, 2016.
[27] K.-K. Oh and H.-S. Ahn, "Formation control and network localization via orientation alignment," *IEEE Trans. Autom. Control*, vol. 59, no. 2, pp. 540–545, Feb. 2014.
[28] J. Lu and C. Y. Tang, "Distributed asynchronous algorithms for solving positive definite linear equations over networks—Part I: Agent networks," *IFAC Proc. Volumes*, vol. 42, no. 20, pp. 252–257, 2009.
[29] J. Liu, S. Mou, and A. S. Morse, "Asynchronous distributed algorithms for solving linear algebraic equations," *IEEE Trans. Autom. Control*, vol. 63, no. 2, pp. 372–385, Feb. 2018.
[30] G. Shi and B. D. O. Anderson, "Network flows that solve linear equations," *IEEE Trans. Autom. Control*, vol. 62, no. 6, pp. 2659–2674, Jun. 2017.
[31] W. Shen, B. Yin, X. Cao, and Y. Cheng, "A distributed secure outsourcing scheme for solving linear algebraic equations in ad hoc clouds," *IEEE Trans. Cloud Comput.*, vol. 7, no. 2, pp. 415–430, Apr.–Jun. 1, 2019.
[32] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.
[33] D. Hayden, Y. H. Chang, J. Goncalves, and C. J. Tomlin, "Sparse network identifiability via compressed sensing," *Automatica*, vol. 68, pp. 9–17, 2016.
[34] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution," *Commun. Pure Appl. Math.: J. Issued Courant Inst. Math. Sci.*, vol. 59, no. 6, pp. 797–829, 2006.
[35] H. Rauhut, "Compressive sensing and structured random matrices," in, *Theoretical Foundations Numerical Methods Sparse Recovery*. Berlin, Germany: De Gruyter, vol. 9, pp. 1–92, 2010.
[36] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*. Basel, Switzerland: Birkhäuser, 2013, vol. 1, no. 3.
[37] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.
[38] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.
[39] J. F. Mota, J. M. F. Xavier, P. M. Q. Aguiar, and M. Püschel, "Distributed basis pursuit," *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 1942–1956, Apr. 2012.
[40] K. Hashimoto, M. Chong, and D. V. Dimarogonas, "Distributed l1-state-and-fault estimation for multi-agent systems," 2018, *arXiv:1810.08756*.
[41] Y. H. Chang, Q. Hu, and C. J. Tomlin, "Secure estimation based Kalman filter for cyber-physical systems against adversarial attacks," 2015, *arXiv:1512.03853*.
[42] Q. Hu, D. Fooladivanda, Y. H. Chang, and C. J. Tomlin, "Secure state estimation and control for cyber security of the nonlinear power systems," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 3, pp. 1310–1321, Sep. 2018.
[43] Y. Shoukry and P. Tabuada, "Event-triggered state observers for sparse sensor noise/attacks," *IEEE Trans. Autom. Control*, vol. 61, no. 8, pp. 2079–2091, Aug. 2016.
[44] M. Todescato, G. Cavraro, R. Carli, and L. Schenato, "A robust block-Jacobi algorithm for quadratic programming under lossy communications," *IFAC-PapersOnLine*, vol. 48, no. 22, pp. 126–131, 2015.
[45] A. Giridhar and P. R. Kumar, "Distributed clock synchronization over wireless networks: Algorithms and analysis," in *Proc. 45th IEEE Conf. Decision Control*, 2006, pp. 4915–4920.
[46] A. S. Asratian, T. M. Denley, and R. Häggkvist, *Bipartite Graphs and Their Applications*. Cambridge, U.K.: Cambridge Univ. Press, 1998, vol. 131.
[47] M. Shi, C. De Persis, N. Monshizadeh, and P. Tesi, "Bias estimation in sensor networks," 2019. [Online]. Available: http://depersis.altervista.org/papers/2019-Shi-DePersis-Tesi-preprint.pdf
[48] D. Cvetković, P. Rowlinson, and S. K. Simić, "Signless Laplacians of finite graphs," *Linear Algebra Appl.*, vol. 423, no. 1, pp. 155–171, 2007.
[49] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "A secure control framework for resource-limited adversaries," *Automatica*, vol. 51, pp. 135–148, 2015.
[50] I. Notarnicola, R. Carli, and G. Notarstefano, "Distributed partitioned big-data optimization via asynchronous dual decomposition," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 4, pp. 1910–1919, Dec. 2018.
[51] A. Chiuso, F. Fagnani, L. Schenato, and S. Zampieri, "Gossip algorithms for simultaneous distributed estimation and classification in sensor networks," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 4, pp. 691–706, Aug. 2011.
[52] M. Zhao, M. D. Kaba, R. Vidal, D. P. Robinson, and E. Mallada, "Sparse recovery over graph incidence matrices," in *Proc. IEEE Conf. Decision Control*, Dec. 2018, pp. 364–371.
[53] C. Lee, H. Shim, and Y. Eun, "Secure and robust state estimation under sensor attacks, measurement noises, and process disturbances: Observer-based combinatorial approach," in *Proc. IEEE Eur. Control Conf.*, 2015, pp. 1872–1877.
[54] J. Zhou, W. Xuan, S. Mou, and B. D. Anderson, "Distributed algorithm for achieving minimum l 1 norm solutions of linear equation," in *Proc. IEEE Annu. Amer. Control Conf.*, 2018, pp. 5857–5862.
[55] D. Feijer and F. Paganini, "Stability of primal-dual gradient dynamics and applications to network optimization," *Automatica*, vol. 46, pp. 1974–1981, 2010.
[56] M. Bürger, G. Notarstefano, F. Bullo, and F. Allgöwer, "A distributed simplex algorithm for degenerate linear programs and multi-agent assignments," *Automatica*, vol. 48, pp. 2298–2304, 2012.
[57] D. Richert and J. Cortés, "Robust distributed linear programming," *IEEE Trans. Autom. Control*, vol. 60, no. 10, pp. 2567–2582, Oct. 2015.

Mingming Shi received the B.Eng. degree in spacecraft design and engineering and the M.Eng. degree in aeronautical and astronautical science and technology from the Harbin Institute of Technology, Harbin, China, in 2013 and 2015, respectively, and the Ph.D. degree from the Faculty of Science and Engineering, University of Groningen, Groningen, The Netherlands, in 2019.

He is currently a Postdoctoral Researcher with the Université Catholique de Louvain, Ottignies-Louvain-la-Neuve, Belgium. His research interests include networked control systems and sensor networks.

Claudio De Persis (Member, IEEE) received the Laurea degree in electronic engineering and the Ph.D. degree in system engineering from the University of Rome "La Sapienza," Rome, Italy, in 1996 and 2000, respectively.

He is currently a Professor with the Faculty of Science and Engineering, Engineering and Technology Institute, University of Groningen, Groningen, The Netherlands. Previously, he held faculty positions with the Department of Mechanical Automation and Mechatronics, University of Twente, and the Department of Computer, Control, and Management Engineering, University of Rome "La Sapienza". He was a Research Associate with the Department of Systems Science and Mathematics, Washington University, St. Louis, MO, USA, during 2000–2001, and with the Department of Electrical Engineering, Yale University, New Haven, CT, USA, during 2001–2002. His current research interests include learning, data, and control, mainly focusing on control theory.

Prof. De Persis was an Editor of the *International Journal of Robust and Nonlinear Control* from 2006 to 2013, an Associate Editor for the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY from 2010 to 2015, and for the IEEE TRANSACTIONS ON AUTOMATIC CONTROL from 2012 to 2015. He has been an Associate Editor for *Automatica* since 2013 and for *IEEE Control Systems Letters* since 2017.

Pietro Tesi (Member, IEEE) received the Ph.D. degree in computer and control engineering from the University of Florence, Florence, Italy, in 2010.

He is currently an Associate Professor with the University of Florence. He also holds a Guest Research Position with the University of Groningen, Groningen, The Netherlands. His main research interests include adaptive and learning systems, networked control systems, and data-driven control design.

Prof. Tesi has been an Editor for the *International Journal of Robust and Nonlinear Control* since 2016, and the *IEEE Control Systems Letters* since 2018.

Nima Monshizadeh received the B.Sc. degree in electrical engineering from the University of Tehran, Tehran, Iran, in 2006, the M.Sc. degree in control engineering from the K.N. Toosi University of Technology, Tehran, in 2009, and the Ph.D. degree with honors (*cum laude*) from the Johann Bernoulli Institute for Mathematics and Computer Science, University of Groningen, Groningen, The Netherlands, in 2013.

Afterward, he was appointed a Postdoctoral Researcher with the Engineering and Technology Institute, University of Groningen. He was a Research Associate with the Control Group of the University of Cambridge, before taking a tenure track position at the University of Groningen in 2018. His research interests include optimization and control of complex networks, model reduction, and power networks.