

# Distributed Estimation and Detection under Local Information

F. Pasqualetti\* R. Carli\* A. Bicchi\*\* F. Bullo\*

\* *Center for Control, Dynamical Systems and Computation, University of California, Santa Barbara, USA*

{fabiopas,carlirug,bullo}@engineering.ucsb.edu

\*\* *Centro I. R. "E. Piaggio", Università di Pisa, Pisa, Italy*  
bicchi@ing.unipi.it

---

**Abstract:** This work considers the problem of obtaining optimal estimates via distributed computation in a large scale system. The electric power system, the transportation system, and generally any computer or network system, are examples of large scale systems: a decentralized estimation of signals based on observations acquired by spatially distributed sensors is the basis for a wide range of important applications. In this work, we focus on the problem of reconstructing the initial state of a linear network in the presence of process and measurement noise. We consider a local model information setup, in which the entire dynamical and measurement model is nowhere available and cannot be reconstructed for the computation. Our estimation procedure relies upon a novel technique to solve a consistent system of linear equations, for which we prove correctness and convergence. In the second part of the paper we consider the problem of detecting anomalies in a large scale network driven by noise. Despite the theoretical advances in this field of research, the currently available procedures to enforce security in large scale systems are computationally inefficient and numerically unreliable. Using our optimal estimation scheme, we describe a distributed procedure with performance guarantees that only requires local knowledge of the system model.

---

## 1. INTRODUCTION

Large scale distributed systems, such as the electrical power network and the telecommunication network, are receiving increasing attention from researchers in different fields. The wide spatial distribution and the high dimensionality of network systems forbid the use of centralized solutions to tackle classical estimation, control, and fault detection problems, and they require, instead, the development of new decentralized techniques. One possibility to overcome these issues is to geographically deploy some leaders in the network, each one responsible for a different subpart of the whole system. Local estimation and control schemes can successively be used, together with an information exchange mechanism to recover the performance of a centralized scheme.

We focus here on the problem of reconstructing and computing a function of the initial state starting from local observations. This very basic estimation problem arises often in sensor networks and in multi-agent systems. For instance, many formation and coordination problems require the agents to compute the average of the initial states, or simply a consensus on a variable of interest, e.g., see Ren et al. (2005) and the references therein. The procedures proposed in these works require little knowledge of the network structure, and they can be implemented on

agents with very limited computation capabilities. On the other hand, their convergence is usually very slow, and it depends upon the dimension of the network, e.g., see Olshevsky and Tsitsiklis (2009). A different approach is taken in Sundaram and Hadjicostis (2008), where, assuming that the network is observable in a system theoretic sense, a finite time algorithm is derived to both recover the initial state and evaluate a function of it. This method requires the knowledge by each agent of the entire network topology, and it suffers from numerical instabilities as the network size grows. Clearly, a strong tradeoff exists between the performance of a consensus or estimation algorithm and the work load on each single agent. In this paper, we further exploit this tradeoff by proposing a hierarchical structure, in which only a subset of the agents is in charge of the computational burden. By doing so, we maintain the performance and increase the stability of the finite time algorithms, while preserving, for the majority of the agents, the simplicity of the asymptotic procedures.

We allow for the presence of process and measurement noise in the network system. The state estimation problem in a distributed setting and in the presence of noise is receiving increasing attention in the last years, e.g., see Carli et al. (2008) and the references therein. A distributed version of the Kalman filter is often proposed, whose structure depends upon the assumptions on the noise covariance matrix and upon the knowledge each agent needs to have. Besides the fact that, under a general error model, no finite time distributed algorithm is available,

---

\* This material is based in part upon work supported by ICB ARO grant DAAD19-03-D-0004 and AFOSR MURI grant FA9550-07-1-0528, and in part upon the EC Contract IST 224428 "CHAT".

these solutions are better suited to track the evolution of the system, rather than computing a function of the initial state, and they constitute therefore a complementary area of research with respect to this work.

Because of the decentralized nature of network systems, sensor and actuator failures, as well as cyber attacks compromising the availability of resources, constitute an actual threat. As for the state estimation problem, the large dimension of the system under consideration prevents the use of a centralized detection system, hence ruling out classical solutions as presented in Ding (2008). Few techniques have been presented so far that assume only local knowledge of the system model. In Chung et al. (2001) a method based on decentralized estimation and game theoretic fault detection theory is proposed. In Li et al. (2009) an LMI based approach is used to design the structure of local residual generators, in a way that their behavior imitate the output of a centralized detection filter. Although appealing, these solutions have no performance guarantees, and it is a difficult matter to characterize the class of anomalies that can successfully be detected. A different approach is taken in Pasqualetti et al. (2010a), where, by exploiting the presence in the network of weakly interconnected subparts, a bound on the detection performance is characterized.

The main contributions of this work are threefold. First, we describe a novel technique to compute the minimum norm solution as well as the weighted least squares solution to a set of linear equations. The proposed linear system solver belongs to the class of row-projection methods, and it is shown to converge in a finite number of steps. Second, we introduce a hierarchical structure in the network by allowing for the presence of a subset of agents with better computation and communication capabilities. We describe a distributed procedure for the leaders to reconstruct the network initial state and to compute a linear function of it. Our procedure converges in a finite number of steps, which we characterize, and it requires local knowledge of the network topology. In the absence of noise in the system, our method reconstructs exactly the initial state of the system. In the presence of noise, instead, the linear minimum variance estimate of the initial state is computed. Third and finally, we use our estimation procedure to design a distributed and finite time detection scheme, in which each leader operates under local model information. A condition to avoid incorrect detection is given, and, consequently, the set of detectable anomalies is characterized. The proposed detection method is, to the best of our knowledge, the first decentralized detection method with performance guarantees that works on any network structure and in the presence of noise.

The rest of the paper is organized as follows. Section 2 contains the notation. Our linear system solver is in Section 3. Sections 4 and 5 contain our main results. Our numerical studies and conclusion are in Section 6 and 7.

## 2. PROBLEM SETUP

We introduce in this section the notation and the problems under consideration. Let  $G$  denote a directed graph with vertex set  $V = \{1, \dots, n\}$  and edge set  $E \subset V \times V$ . The in-neighbor set of a node  $i \in V$ , i.e., all the nodes  $j \in V$

such that the pair  $(j, i) \in E$ , is denoted with  $N_i$ . We let each vertex  $j \in V$  denote an autonomous agent, and we associate a real number  $x_j$  with each agent  $j$ . Let the vector  $x$  contain the values  $x_j$ . A linear iteration over  $G$  is an update rule for the state vector  $x$  and it is described by the linear discrete time system

$$x(t+1) = Ax(t).$$

We refer to the graph  $G$  as to the communication graph associated with the network matrix  $A$ , in which the  $(i, j)$ -th entry is nonzero if and only if  $(j, i) \in E$ . An output matrix  $C_j$  describes the information about the state of the network that is directly available to  $j$ . In particular,  $y_j(t) = C_j x(t)$ ,  $C_j = [e_{n_1} \dots e_{n_p}]^T$ , and  $\{n_1, \dots, n_p\} = N_j$ . In this work, we consider the noisy network

$$\begin{aligned} x(t+1) &= Ax(t) + v(t), \\ y_j(t) &= C_j x(t) + w_j(t), \end{aligned} \quad (1)$$

where  $v(t) \in \mathbb{R}^n$  and  $w_j(t) \in \mathbb{R}^p$  for all  $j \in \{1, \dots, n\}$ . We also make the standard assumptions that

$$\mathbb{E}[v(t)w_j(t)] = 0, \quad \mathbb{E}[w_i(t)w_j(t)] = 0, \quad (2)$$

and that, being  $\delta$  the Kronecker delta function and  $I$  the identity matrix of appropriate dimension,

$$\begin{aligned} \mathbb{E}[v(t)] &= 0, \quad \mathbb{E}[v(t)v(\tau)^T] = \sigma_v^2 \delta(t-\tau)I, \\ \mathbb{E}[w_j(t)] &= 0, \quad \mathbb{E}[w_j(t)w_j(\tau)^T] = \sigma_w^2 \delta(t-\tau)I, \end{aligned} \quad (3)$$

for all  $t, \tau \in \mathbb{R}$  and for all  $i, j \in \{1, \dots, n\}$  with  $i \neq j$ . We introduce a hierarchy in the network by assuming that some agents, denoted as *leaders*, have better computation and communication capabilities. Let  $V^{(\ell)} = \{\ell_1, \dots, \ell_m\} \subseteq V$  be the leader set, and let  $G^{(\ell)} = (V^{(\ell)}, E^{(\ell)})$  be the *leader graph*, where  $E^{(\ell)} \subseteq V^{(\ell)} \times V^{(\ell)}$  describes the feasible communications among the leaders. Let  $N_i^{(\ell)}$  denote the set of the neighbors of the leader  $\ell_i$  in  $G^{(\ell)}$ . The information of the state  $x(t)$  directly available to the leader  $\ell_i$  is given by  $y_{\ell_i}(t) = C_{\ell_i} x(t)$ , where  $C_{\ell_i}$  is defined according to the neighbor set  $N_{\ell_i}$  in  $G$ . We assume that the leader graph  $G^{(\ell)}$  is strongly connected.

The problem of obtaining the linear minimum-variance estimate of the initial state of the network (1) is considered in this work. Observe that, if the agent  $j$  has complete knowledge of the network matrix  $A$ , and if the pair  $(A, C_j)$  is observable, see Ding (2008), then an optimal estimate of the initial state can be obtained by collecting a sufficient number of output measurements, and by solving a standard weighted least squares problem as described in Luenberger (1969). We consider in this paper the case in which every agent (resp. leader) has only local knowledge of the network model. In other words, we assume that the matrix  $A$  is nowhere available and, furthermore, cannot be reconstructed, so that the centralized solution described above is inapplicable.

*Problem 1. (Distributed minimum-variance estimate)* Design a distributed algorithm for the leaders to obtain the linear minimum-variance estimate of  $x(0)$ .

In the second part of the paper we use our distributed estimation algorithm to address the problem of detecting failures and anomalies in the network. We model the presence of an anomaly as an unknown and unmeasurable exogenous input in the system (1). This model has been shown to be general, since it considers both sensor and

actuator failures, as well as malicious external attacks, e.g., see Pasqualetti et al. (2010a). The following problem is also addressed in this paper.

*Problem 2.* Design an algorithm for the leaders to detect an anomaly in the network.

We next describe a method to solve a system of linear equations, which will be used to derive our main results.

### 3. AN ITERATIVE LINEAR SYSTEM SOLVER AMENABLE TO DISTRIBUTED IMPLEMENTATION

We consider in this section the problem of determining the minimum norm solution as well as the weighted least squares solution to a set of linear equations. As it will be clear in the following sections, the procedure we propose is particularly convenient in a decentralized setup, in which the knowledge of the whole set of linear equations is nowhere available. We start with the computation of the minimum norm solution. Consider the system of linear equations  $y = Ax$ , where  $A \in \mathbb{R}^{l \times n}$ ,  $y \in \mathbb{R}^l$ ,  $x \in \mathbb{R}^n$ , and assume that  $y \in \text{Im}(A)$ , where  $\text{Im}(A)$  denotes the range space spanned by the matrix  $A$ . The minimum norm solution to  $y = Ax$  coincides with the vector  $\hat{x}$  such that  $y = A\hat{x}$  and  $\|\hat{x}\|_2$  is minimum. It is known that  $\|\hat{x}\|_2$  being minimum corresponds to  $\hat{x}$  being orthogonal to  $\text{Ker}(A)$ , where  $\text{Ker}(A)$  denotes the null space of the matrix  $A$ . Let

$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_m \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}, \quad (4)$$

where  $A_i \in \mathbb{R}^{m_i \times n}$ ,  $y_i \in \mathbb{R}^{m_i}$ , for  $i \in \{1, \dots, m\}$ , and  $\sum_{i=1}^m m_i = l$ . Our procedure to compute the minimum norm solution is in Algorithm 1, where, given a subspace  $\mathcal{V}$ ,  $\text{Basis}(\mathcal{V})$  denotes any full rank orthonormal matrix whose columns span the subspace  $\mathcal{V}$ . The reader familiar with row-action methods, a detailed discussion of which is in Censor (1981), may notice a similarity with the procedure we propose. It should be observed that, differently from the existing row-action methods, our algorithm converges in finite time, and, as it is shown later, it can be used to compute any weighted least squares solution to a system of linear equations. For  $1 \leq k \leq m$ , let

$$A^k = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_k \end{bmatrix}, \quad y^k = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix}.$$

We next show that the computation of the vector  $\hat{x}_i$  and of the matrix  $K_i$  in Algorithm 1 is such that  $y^i = A^i \hat{x}_i$  and  $\hat{x}_i \perp \text{Ker}(A^i)$ , so that the minimum norm solution to the system  $y = Ax$  is finally returned.

*Theorem 1. (Convergence of Algorithm 1)* Consider the system of linear equations  $y = Ax$ , where  $A$  and  $y$  are partitioned in  $m$  blocks as in (4). Assume that  $y \in \text{Im}(A)$ . Then Algorithm 1 returns in  $m$  steps the vector  $\hat{x}$  such that  $y = A\hat{x}$  and  $\hat{x} \perp \text{Ker}(A)$ .

**Proof.** We refer the reader to Pasqualetti et al. (2010b) for a proof of the results contained in this paper.  $\square$

We focus now on the computation of the weighted least squares solution to a set of linear equations. Let  $w$  be an

---

#### Algorithm 1 Sequential system solver

---

**Input:**  $A_1, \dots, A_m, y_1, \dots, y_m$ ;

**Require:**  $y \in \text{Im}(A)$ ;

$\hat{x}_0 := 0, K_0 := I_n$ ;

**for**  $i = 1 : m$  **do**

$\hat{x}_i := \hat{x}_{i-1} + K_{i-1}(A_i K_{i-1})^\dagger (y_i - A_i \hat{x}_{i-1})$ ;

$K_i := \text{Basis}(K_{i-1} \text{Ker}(A_i K_{i-1}))$ ;

**end for**

**return**  $\hat{x}_m$ ;

---

unknown and unmeasurable  $n$ -dimensional random vector, with  $\mathbb{E}[w] = 0$  and  $\mathbb{E}[ww^T] = Q = Q^T > 0$ . Let

$$y = Ax + w, \quad (5)$$

where  $\text{Ker}(A) = 0$ , and recall from Luenberger (1969) that the linear minimum variance estimate of  $x$  equals

$$x_{\text{wls}} = (A^T Q^{-1} A)^{-1} A^T Q^{-1} y.$$

We aim at computing the vector  $x_{\text{wls}}$  by means of Algorithm 1. Because the matrix  $Q$  is symmetric and positive definite, there exists<sup>1</sup> a full row rank matrix  $B$  such that  $Q = BB^T$ . Notice that equation (5) can be rewritten as

$$y = [A \ \varepsilon B] \begin{bmatrix} x \\ \bar{w} \end{bmatrix}, \quad (6)$$

where  $\varepsilon > 0$ ,  $\mathbb{E}[\bar{w}] = 0$  and  $\mathbb{E}[\bar{w}\bar{w}^T] = \varepsilon^{-2}I$ . Because  $B$  has full row rank, the system (6) is underdetermined, i.e.,  $y \in \text{Im}([A \ \varepsilon B])$  and  $\text{Ker}([A \ \varepsilon B]) \neq 0$ . Let

$$\begin{bmatrix} \hat{x}(\varepsilon) \\ \bar{w} \end{bmatrix} = [A \ \varepsilon B]^\dagger y.$$

Notice that the system (6) can be partitioned as in (4), and that Algorithm 1 can be used to compute the vectors  $\hat{x}(\varepsilon)$  and  $\bar{w}$ . We next show how the vector  $\hat{x}(\varepsilon)$  is related to the minimum variance estimate  $x_{\text{wls}}$ .

*Theorem 2. (Asymptotic convergence)* Consider the system of linear equations (5). Let  $\mathbb{E}[w] = 0$  and  $\mathbb{E}[ww^T] = Q = Q^T = BB^T > 0$ , for a full row rank matrix  $B$ . Let

$$C = \varepsilon(I - AA^\dagger)B,$$

$$E = I - C^\dagger C,$$

$$D = \varepsilon E [I + \varepsilon^2 EB^T (AA^T)^\dagger BE]^{-1} B^T (AA^T)^\dagger (I - \varepsilon BC^\dagger).$$

Then

$$[A \ \varepsilon B]^\dagger = \begin{bmatrix} A^\dagger - \varepsilon A^\dagger B (C^\dagger + D) \\ C^\dagger + D \end{bmatrix},$$

and

$$\lim_{\varepsilon \rightarrow 0^+} A^\dagger - \varepsilon A^\dagger B (C^\dagger + D) = (A^T Q^{-1} A)^{-1} A^T Q^{-1}.$$

Throughout the paper, let  $\hat{x}(\varepsilon)$  be the estimate of  $x$  returned by Algorithm 1 applied to (6), and let

$$\hat{x} = \lim_{\varepsilon \rightarrow 0^+} \hat{x}(\varepsilon).$$

Theorem 2 guarantees that the minimum variance estimate  $x_{\text{wls}}$  coincides with the limit of  $\hat{x}(\varepsilon)$  as  $\varepsilon \rightarrow 0^+$ . When the parameter  $\varepsilon$  is fixed, the estimate  $\hat{x}(\varepsilon)$  differs from the minimum variance estimate  $\hat{x}_{\text{wls}}$ . We next characterize the approximation error  $x_{\text{wls}} - \hat{x}(\varepsilon)$ .

*Corollary 3. (Approximation error)* Consider the system (5), and let  $\mathbb{E}[ww^T] = BB^T > 0$ . Then

$$x_{\text{wls}} - \hat{x}(\varepsilon) = \varepsilon A^\dagger B D y,$$

where  $B$  and  $D$  are as in Theorem 2.

<sup>1</sup> Choose for instance  $B = W\Lambda^{1/2}$ , where  $W$  is a basis of eigenvectors and  $\Lambda$  is the corresponding diagonal matrix of the eigenvalues.

To conclude this section, we characterize the residual  $y - A\hat{x}$ . This quantity plays an important role in Section 5 for the synthesis of a distributed detection algorithm.

*Corollary 4. (Residual)* Consider the system (5), and let  $\mathbb{E}[ww^T] = Q = Q^T > 0$ . Then<sup>2</sup>

$$\lim_{\varepsilon \rightarrow 0^+} \|y - A\hat{x}\| \leq \|(I - AW)\| \|w\|,$$

where  $W = (A^T Q^{-1} A)^{-1} A^T Q^{-1}$ .

#### 4. OPTIMAL FUNCTION EVALUATION

The problem of obtaining the minimum variance estimate of the initial state of the noisy network (1) is considered in this section. Precisely, we exploit the hierarchic levels in the network, and we describe a distributed procedure for the leaders to recover the desired estimation under the assumption that no leader has knowledge of the entire network structure. We first introduce some notation. Let  $V$  denote the set of agents, and let  $\{\ell_1, \dots, \ell_m\} \subseteq V$  denote the subset of leaders. Let  $C_{\ell_i}$  be the output matrix associated with the leader  $\ell_i$ , and let  $w_{\ell_i}$  be the corresponding measurement noise. The composite information available to the set of leaders can be conveniently described by the output matrix  $C^{(\ell)} = [C_{\ell_1}^T \ \dots \ C_{\ell_m}^T]^T$ , so that the evolution of the network with the leaders observations becomes

$$\begin{aligned} x(t+1) &= Ax(t) + v(t), \\ y(t) &= C^{(\ell)}x(t) + w(t), \end{aligned} \quad (7)$$

where  $w(t)^T = [w_{\ell_1}(t)^T \ \dots \ w_{\ell_m}(t)^T]^T$ , and  $v(t)$ ,  $w(t)$  satisfy the assumptions (2) and (3) at all instants  $t$ . Let  $I_n$  denote the identity matrix of dimension  $n \times n$ , and let  $0_{m,n}$  denote the zero matrix of dimension  $m \times n$ . Let  $|V| = n$ , and let  $|N_{\ell_i}| = p_{\ell_i}$ . For  $s \in \mathbb{Z}_{>0}$ , let

$$O_i^s = \begin{bmatrix} C_{\ell_i} \\ C_{\ell_i}A \\ C_{\ell_i}A^2 \\ \vdots \\ C_{\ell_i}A^{s-1} \end{bmatrix}, \quad Y_i^s = \begin{bmatrix} y_{\ell_i}(0) \\ y_{\ell_i}(1) \\ y_{\ell_i}(2) \\ \vdots \\ y_{\ell_i}(s-1) \end{bmatrix},$$

and

$$F_i^s = \begin{bmatrix} D_{\ell_i} & 0 & \dots & \dots & 0 \\ C_{\ell_i}B & D_{\ell_i} & \ddots & \ddots & 0 \\ C_{\ell_i}AB & C_{\ell_i}B & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ C_{\ell_i}A^{s-2}B & C_{\ell_i}A^{s-3}B & \dots & C_{\ell_i}B & D_{\ell_i} \end{bmatrix},$$

where

$$\begin{aligned} B &= [I_n \ 0_{n,p_{\ell_1}} \ \dots \ 0_{n,p_{\ell_m}}], \\ D_{\ell_i} &= [0_{p_{\ell_i},n} \ \dots \ 0_{p_{\ell_i},p_{\ell_{i-1}}} \ I_{p_{\ell_i}} \ 0_{p_{\ell_i},p_{\ell_{i+1}}} \ \dots \ 0_{p_{\ell_i},p_{\ell_m}}]. \end{aligned}$$

Finally, let

$$O^s = \begin{bmatrix} O_1^s \\ O_2^s \\ \vdots \\ O_m^s \end{bmatrix}, \quad Y^s = \begin{bmatrix} Y_1^s \\ Y_2^s \\ \vdots \\ Y_m^s \end{bmatrix}, \quad F^s = \begin{bmatrix} F_1^s \\ F_2^s \\ \vdots \\ F_m^s \end{bmatrix},$$

<sup>2</sup> Given a vector  $w$  and a matrix  $A$ , we denote by  $\|w\|$  any vector norm, and by  $\|A\|$  the corresponding induced matrix norm.

---

#### Algorithm 2 Sequential state estimator (Leader $\ell_i$ )

---

**Input:**  $O_i^s, F_i^s \Lambda^{1/2}, y_i, \varepsilon$ ;  
**if**  $i = 1$  **then**  
     $\hat{x} := 0$  and  $K := I$ ;  
**else**  
    receive  $\hat{x}$  and  $K$  from leader  $\ell_{i-1}$ ;  
**end if**  
 $\hat{x} := \hat{x} + K([O_i^s \ \varepsilon F_i^s \Lambda^{1/2}]K)^\dagger (y_i - [O_i^s \ \varepsilon F_i^s \Lambda^{1/2}]\hat{x})$ ;  
 $K := \text{Basis}(K \text{ Ker}([O_i^s \ \varepsilon F_i^s \Lambda^{1/2}]K))$ ;  
**if**  $i = m$  **then**  
    transmit  $\hat{x}$  to every leader;  
**return**  $\hat{x}$ ;  
**else**  
    transmit  $\hat{x}$  and  $K$  to leader  $\ell_{i+1}$ ;  
**end if**

---

and, for  $t \geq 0$ ,

$$U(t) = \begin{bmatrix} v(t) \\ w_{\ell_1}(t) \\ \vdots \\ w_{\ell_m}(t) \end{bmatrix}, \quad U^s = \begin{bmatrix} U(0) \\ U(1) \\ \vdots \\ U(s-1) \end{bmatrix}.$$

Observe that

$$Y^s = O^s x(0) + F^s U^s. \quad (8)$$

Assume that  $\text{Ker}(O^s) = 0$ , and notice that  $F^s$  has full row rank. The minimum variance estimate of  $x(0)$  according to equation (8) is given by

$$\hat{x}(0) = ((O^s)^T Q^{-1} O^s)^{-1} (O^s)^T Q^{-1} Y^s,$$

where

$$Q = \mathbb{E}[(F^s U^s)(F^s U^s)^T] = (F^s \Lambda^{\frac{1}{2}})(\Lambda^{\frac{1}{2}}(F^s)^T),$$

and  $\Lambda = \mathbb{E}[U^s(U^s)^T]$ . Observe that  $\Lambda$  is diagonal and positive definite. We now show how Algorithm 1 can be used to compute the minimum variance estimate of  $x(0)$ . Let  $\varepsilon \neq 0$ , and consider the system of linear equations

$$Y^s = \begin{bmatrix} O^s & \varepsilon F^s \Lambda^{\frac{1}{2}} \end{bmatrix} \begin{bmatrix} x(0) \\ \bar{U}^s \end{bmatrix},$$

where  $\bar{U}^s$  is such that  $U^s = \varepsilon \Lambda^{\frac{1}{2}} \bar{U}^s$ ,  $\mathbb{E}[\bar{U}^s(\bar{U}^s)^T] = \varepsilon^{-2} I$ . Assume that an ordering of the leaders has been defined in a way that  $\ell_i \in N_{\ell_{i+1}}^{(\ell)}$  for all  $i \in \{1, \dots, m-1\}$ . A procedure for the leaders to estimate  $x(0)$  is in Algorithm 2.

*Theorem 5. (Minimum variance estimate)* Consider the noisy network (7). Let  $\varepsilon > 0$ , and assume that each leader knows the matrices  $O_i^s$  and  $F_i^s \Lambda^{1/2}$ . Then Algorithm 2 returns an estimate  $\hat{x}(\varepsilon)$  such that  $\|x_{wls} - \hat{x}(\varepsilon)\| = \varepsilon c$ , where the constant  $c$  depends on  $O^s$ ,  $F^s \Lambda^{1/2}$ , and  $Y^s$ .

It should be noticed that if the parameter  $\varepsilon$  converges to zero, then the estimate obtained with Algorithm 2 coincides with the minimum variance estimate of the network initial state. Finally, it can be shown that a leader based procedure, such as Algorithm 2, is numerically more reliable with respect to classical centralized solutions.

*Remark 1. (Function evaluation).* The proposed method can be effectively employed to achieve the minimum variance estimate of a linear function of the network initial state. Indeed, suppose that the function  $f(x(0))$  needs to be computed by all (or a subset of) the leaders, and recall

from Luenberger (1969) that the minimum variance estimate of  $f(x(0))$  is  $f(\hat{x})$ , where  $\hat{x}$  is the minimum variance estimate of  $x(0)$ . Hence, the function evaluation problem is a direct extension of the state estimation problem.

*Remark 2.* (Network knowledge). The computation of the matrices  $O_i^s$  and  $F_i^s \Lambda^{1/2}$ ,  $i \in V^{(\ell)}$ , does not require the knowledge of the entire network model. Let a path be a sequence of vertices, such that any two consecutive vertices in the sequence are connected through an edge. Let the length of a path equal the number of its edges. Let  $A$  be the network matrix, and observe that the  $(i, j)$ -th entry of  $A^k$ , with  $k \in \mathbb{Z}$ , is nonzero if and only if there exists a path of length  $k$  connecting the agent  $j$  to  $i$ . Let  $N_{\ell_i}^s \subseteq V$  denote the set of the neighbors connected to  $\ell_i$  through a path of length at most  $s$ . It can be shown that the matrix  $O_i^s$  and  $F_i^s \Lambda^{1/2}$  can be computed by only knowing the sub-matrix of  $A$  and  $\Lambda^{1/2}$  with rows and columns in  $N_{\ell_i}^s$ .

## 5. ANOMALY DETECTION

Because of the absence of a centralized and omniscient entity that monitors the behavior of the network, the system (1) is prone to malfunctions and external attacks. It can be shown that the presence of sensor and actuator failures, as well as of external attacks, can be modeled by adding an unknown and unmeasurable input to the network, e.g., see Pasqualetti et al. (2010a). We say that the network (1) undergoes an anomaly if it is affected by an exogenous input that steers the network along state trajectories not belonging to (1). Precisely, the network model in the presence of an anomaly  $e$  becomes

$$x(t+1) = Ax(t) + Ee(t) + v(t), \quad (9)$$

where the matrix  $E$  is referred to as the anomaly or failure signature. We show in this section how our estimation procedure can be used to allow the leaders to detect an anomaly in the network by only using local model information. It should be noticed that the presence of the noise  $v$  and  $w$  limit the class of detectable anomalies.

Because the failure signature is a priori unknown, the leaders need to detect the anomaly by means of only the past observations. Let  $Y_{i,t}^s$  denote the vector containing the measurements taken by the leader  $\ell_i$  from time  $t$  up to time  $t+s-1$ . Analogously, let  $U_t^s$  and  $U_{e,t}^s$  contain, respectively, the noise value and the exogenous input from time  $t$  up to time  $t+s-1$ . Then, with the notation introduced in the previous section, we have

$$\begin{bmatrix} Y_{1,t}^s \\ Y_{2,t}^s \\ \vdots \\ Y_{m,t}^s \end{bmatrix} = \begin{bmatrix} O_1^s \\ O_2^s \\ \vdots \\ O_m^s \end{bmatrix} x(t) + \begin{bmatrix} F_1^s \\ F_2^s \\ \vdots \\ F_m^s \end{bmatrix} U_t^s + \begin{bmatrix} F_1^s(E) \\ F_2^s(E) \\ \vdots \\ F_m^s(E) \end{bmatrix} U_{e,t}^s, \quad (10)$$

where, for  $i \in \{1, \dots, m\}$ ,

$$F_i^s(E) = \begin{bmatrix} 0 & 0 & \cdots & \cdots & 0 \\ C_{\ell_i} E & 0 & \ddots & \ddots & 0 \\ C_{\ell_i} E A & C_{\ell_i} E & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ C_{\ell_i} A^{s-2} E & C_{\ell_i} A^{s-1} E & \cdots & C_{\ell_i} E & 0 \end{bmatrix}.$$

---

### Algorithm 3 Sequential anomaly detector (Leader $\ell_i$ )

---

**Input:**  $O_i^s, \varepsilon F_i^s \Lambda^{1/2}, \Gamma$ ;  
**loop**  
  collect the measurements  $Y_{i,t}^s$ ;  
  execute Algorithm 2;  
  compute  $r_i = \|Y_{i,t}^s - O_i^s \hat{x}\|_\infty$ ;  
  **if**  $r_i > \Gamma$  **then**  
    **return** *Anomaly*;  
  **end if**  
**end loop**

---

Assume that  $\text{Ker}(O^s) = 0$ , and recall that the noise covariance matrix is  $Q = Q^T = (F^s \Lambda^{\frac{1}{2}})(\Lambda^{\frac{1}{2}}(F^s)^T)$ . The following condition ensures the detectability of the anomaly from the measurements.

*Theorem 6.* (Detectability). Consider the network (9). Let  $\hat{x}(t)$  be the minimum variance estimate of  $x(t)$  according to the measurements  $Y_t^s$ . An anomaly is present if

$$\|Y_t^s - O^s \hat{x}(t)\| > \|(I - O^s W)\| \|U_t^s\|,$$

where  $W = ((O^s)^T Q^{-1} O^s)^{-1} (O^s)^T Q^{-1}$ .

Following Theorem 6, a procedure to detect the presence of an anomaly in the network via distributed computation is in Algorithm 3. Notice that the value of the threshold  $\Gamma$  determines the false alarm and misdetection rate. Clearly, if  $\Gamma \geq \|(I - O^s W)\| \|U_t^s\|$  then no false alarm is triggered, at the expenses of the misdetection rate. By decreasing the value of  $\Gamma$  the sensitivity to failures increases together with the false alarm rate. Observe that, if the magnitude of the noise signals is bounded by  $\gamma$ , then a reasonable choice of the threshold is  $\Gamma = \gamma \|(I - O^s W)\|_\infty$ . We present a related example in the next section, and we leave the case of unbounded noise, e.g., gaussian noise, as future work. Finally, observe that the use of the infinity norm in Algorithm 3 is also convenient for the implementation. Indeed, once the estimation  $\hat{x}(t)$  has been computed, the condition  $\|Y_{i,t}^s - O_i^s \hat{x}(t)\|_\infty > \Gamma$  can be checked individually by each leader without any further communication.

## 6. SIMULATIONS

In this section two numerical studies are considered to demonstrate the effectiveness of our numerical procedures. First, following Theorem 5, we show that, as the parameter  $\varepsilon$  decreases, the estimation of the network initial state converges to the minimum variance estimate of the initial state. Second, we demonstrate the performance of our Anomaly detection procedure. Let  $G$  denote the graph of Fig. 1, and construct the matrix  $A = [a_{ij}]$  as follows:

- (i) if the agents  $i$  and  $j$  are connected, then set the value of  $a_{ij} = a_{ji}$  according to a uniform distribution<sup>3</sup> in the interval  $(0, 1)$ , and set  $a_{ij} = a_{ji} = 0$  otherwise;
- (ii) normalize the rows of  $A$  such that  $\|A\|_\infty = 1$ .

Let  $V^{(\ell)}$  denote the leader set, and let  $C^{(\ell)}$  be the composite observation matrix. The evolution of the network is described by (7), where the random vectors  $v(t)$  and  $w(t)$  satisfy the assumptions (2) and (3) at all instants

<sup>3</sup> If the entries of  $A$  are chosen uniformly and independently in the interval  $(0, 1)$ , then the pair  $(A, C_j)$  is observable with probability one for every output matrix  $C_j$ .

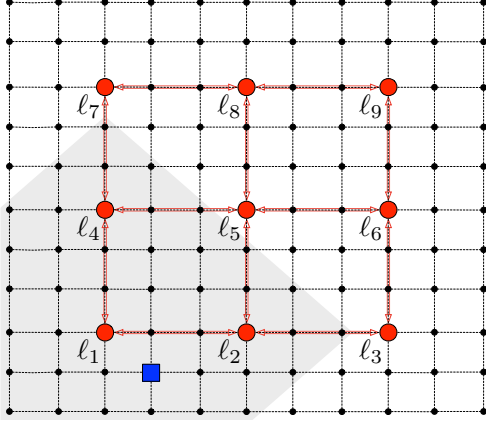


Fig. 1. A grid network with 121 agents. The 9 leaders and the leaders graph are in red. The misbehaving agent is in blue (square). The shaded region denotes the subnetwork known to the leader  $l_1$ .

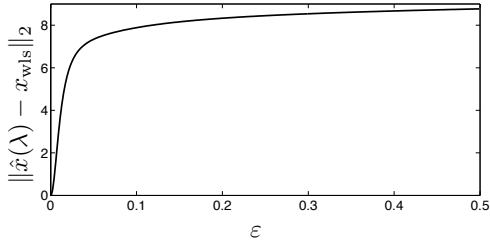


Fig. 2. The estimation error  $\|\hat{x}(\varepsilon) - x_{\text{wis}}\|_2$  decreases to 0 with the parameter  $\varepsilon$ .

$t$ , and their entries are uniformly distributed in the interval  $[-\gamma, \gamma]$ . For the simulation we let  $\gamma = 0.15$  and, consequently,  $\sigma_v^2 = \sigma_w^2 = \gamma^2/12$ . As in Section 4, we have  $Y^s = O^s x(0) + F^s U^s$ , and  $Q = \mathbb{E}[(F^s U^s)(F^s U^s)^T] = (F^s \Lambda^{1/2})(\Lambda^{1/2}(F^s)^T)$ . The estimation error  $\|\hat{x}(\varepsilon) - x_{\text{wis}}\|_2$  is plotted in Fig. 2 as a function of the parameter  $\varepsilon$ .

We consider now the problem of detecting an anomaly in the system. Suppose that the blue square agent in Fig. 1 injects the signal  $e(t) = \bar{e} + e_w(t)$ , for  $t \geq \bar{t}$ , into the network, where  $\bar{e}$  is a positive constant, and  $\mathbb{E}[e_w(t)] = 0$ ,  $\mathbb{E}[e_w(t)e_w(\tau)^T] = \sigma_e^2 \delta(t - \tau)$ . For the simulation, we let  $\bar{e} = 0.4$ ,  $\sigma_e^2 = 0.16$ , and  $\bar{t} = 15$ , and we assume that the anomaly signature  $E$  coincides with the vector of the canonical basis corresponding to the index of the misbehaving agent in the network. The residual functions computed by the leaders are in Fig. 3, where the threshold  $\Gamma$  is approximately 0.1. Notice that the behavior of the residual functions can also be used to locate the misbehaving agent in the network. Indeed, because the residuals computed by the leaders  $l_2, \dots, l_9$  do not cross the detection threshold, the misbehaving agent is more likely to be closer to  $l_1$  and  $l_2$ . This conjecture, together with a systematic procedure to identify the misbehaving components, is left as the subject of future research.

## 7. CONCLUSION

The problems of reconstructing the initial state and of detecting anomalies in a linear network driven by process

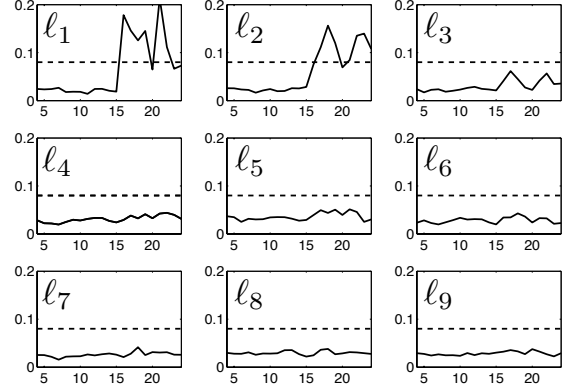


Fig. 3. The residual functions (continuous) and the decision threshold (dashed). The anomaly is correctly detected by the leaders  $l_1$  and  $l_2$ .

and measurement noise have been considered. Finite time algorithms that only assume local knowledge of the network model have been proposed and characterized.

## REFERENCES

- Carli, R., Chiuso, A., Schenato, L., and Zampieri, S. (2008). Distributed Kalman filtering based on consensus strategies. *IEEE Journal on Selected Areas in Communications*. To appear.
- Censor, Y. (1981). Row-action methods for huge and sparse systems and their applications. *SIAM Review*, 444–466.
- Chung, W.H., Speyer, J.L., and Chen, R.H. (2001). A decentralized fault detection filter. *ASME Journal on Dynamic Systems, Measurement, and Control*, 123(3), 237–248.
- Ding, S.X. (2008). *Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*. Springer.
- Li, W., Gui, W.H., Xie, Y.F., and Ding, S.X. (2009). Decentralized fault detection system design for large-scale interconnected systems. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*. Barcelona, Spain.
- Luenberger, D.G. (1969). *Optimization by Vector Space Methods*. Wiley.
- Olshevsky, A. and Tsitsiklis, J.N. (2009). Convergence speed in distributed consensus and averaging. *SIAM Journal on Control and Optimization*, 48(1), 33–55.
- Pasqualetti, F., Bicchi, A., and Bullo, F. (2010a). Consensus computation in unreliable networks: A system theoretic approach. *IEEE Transactions on Automatic Control*. Submitted.
- Pasqualetti, F., Carli, R., Bicchi, A., and Bullo, F. (2010b). Distributed estimation and detection under local information - Proof of results. Available at <http://fabiopas.it/papers/FP-RC-AB-FB-10b-bis.pdf>.
- Ren, W., Beard, R.W., and Atkins, E.M. (2005). A survey of consensus problems in multi-agent coordination. In *American Control Conference*, 1859–1864. Portland, OR.
- Sundaram, S. and Hadjicostis, C.N. (2008). Distributed function calculation and consensus using linear iterative strategies. *IEEE Journal on Selected Areas in Communications*, 26(4), 650–660.