



Distributed estimation via iterative projections with application to power network monitoring[☆]

Fabio Pasqualetti^{a,1}, Ruggero Carli^b, Francesco Bullo^a

^a Center for Control Dynamical Systems and Computation, University of California, Santa Barbara, USA

^b Department of Information Engineering, University of Padova, Padova, Italy

ARTICLE INFO

Article history:

Received 2 March 2011

Received in revised form

22 September 2011

Accepted 27 September 2011

Available online 15 March 2012

Keywords:

Power systems

State estimation

Monitoring

Algebraic approaches

Distributed detection

System security

ABSTRACT

This work presents a distributed method for control centers to monitor the operating condition of a power network, i.e., to estimate the network state, and to ultimately determine the occurrence of threatening situations. State estimation has been recognized to be a fundamental task for network control centers to operate safely and reliably a power grid. We consider (static) state estimation problems, in which the state vector consists of the voltage magnitude and angle at all network buses. We consider the state to be linearly related to network measurements, which include power flows, current injections, and voltage phasors at some buses. We admit the presence of several cooperating control centers, and we design two distributed methods for them to compute the minimum variance estimate of the state, given the network measurements. The two distributed methods rely on different modes of cooperation among control centers: in the first method an *incremental* mode of cooperation is used, whereas, in the second method, a *diffusive* interaction is implemented. Our procedures, which require each control center to know only the measurements and the structure of a subpart of the whole network, are computationally efficient and scalable with respect to the network dimension, provided that the number of control centers also increases with the network cardinality. Additionally, a *finite-memory* approximation of our diffusive algorithm is proposed, and its accuracy is characterized. Finally, our estimation methods are exploited to develop a distributed algorithm to detect corrupted network measurements.

Published by Elsevier Ltd

1. Introduction

Large-scale complex systems, such as the electrical power grid and the telecommunication system, are receiving increasing attention from researchers in different fields. The wide spatial distribution and the high dimensionality of these systems preclude 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 monitors

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.

1.1. Control centers, state estimation and cyber security in power networks

Power systems are operated by system operators from the area control center. The main goal of the system operator is to maintain the network in a secure operating condition, in which all the loads are supplied power by the generators without violating the operational limits on the transmission lines. In order to accomplish this goal, at a given point of time, the network model and the phasor voltage at every system bus needs to be determined, and preventive actions have to be taken if the system is found in an insecure state. For the determination of the operating state, remote terminal units and measuring devices are deployed in the network to gather measurements. These devices are then connected via a local area network to a SCADA (Supervisory Control and Data Acquisition) terminal, which supports the communication of the collected measurements to a control center. At the control center, the measured data is used for control and optimization functions,

[☆] This material is based in part upon work supported by ICB ARO grant W911NF-09-D-0001, NSF grant CNS-0834446, NSF grant CPS-1135819, European Union Seventh Framework Programme [FP7/2007-2013] under grant agreement n°257462 HYCON2 Network of excellence, and in part upon the EC Contract IST 224428 “CHAT”. The material in this paper was partially presented at 2nd IFAC Workshop on Distributed Estimation and Control in Networked Systems, September 13–14, 2010, Annecy, France. This paper was recommended for publication in revised form by Associate Editor James Lam under the direction of Editor Ian R. Petersen.

E-mail addresses: fabiopas@engineering.ucsb.edu (F. Pasqualetti), carlirug@dei.unipd.it (R. Carli), bullo@engineering.ucsb.edu (F. Bullo).

¹ Tel.: +1 805 259 5979.

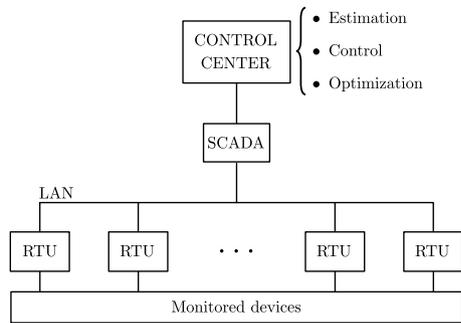


Fig. 1. In this figure remote terminal units transmit their measurements to a SCADA terminal through a local area network. The data is then sent to a control center to implement estimation, control, and optimization procedures.

such as contingency analysis, automatic generation control, load forecasting, optimal power flow computation, and reactive power dispatch (Abur & Exposito, 2004). A diagram representing the interconnections between remote terminal units and the control center is reported in Fig. 1.

Various sources of uncertainties, including measurement and communication noise, lead to inaccuracies in the received data. These inaccuracies may affect the performance of control and optimization algorithms and, ultimately, the stability of the power plant. This concern was recognized and addressed in (Schweppe & Wildes, 1970a,b,c) by introducing the idea of (static) state estimation in power systems.

Power network state estimators are broadly used to obtain an optimal estimate from redundant noisy measurements, and to estimate the state of a network branch which, for economical or computational reasons, is not directly monitored. For the power system state estimation problem, several centralized and parallel solutions have been developed in the last decades, e.g., see Falcao, Wu, and Murphy (1995), Monticelli (1999), Shahidehpour and Wang (2003). Being an online function, computational issues, storage requirements, and numerical robustness of the estimation algorithm need to be taken into account. Within this regard, distributed algorithms based on network partitioning techniques are to be preferred over centralized ones. Moreover, even in a decentralized setting, the work (NERC, 2004) on the blackout of August 2003 suggests that an estimate of the state of the entire network by every control center is essential to prevent networks damages. The Refs. (Jiang, Vittal, & Heydt, 2007; Zhao & Abur, 2005) explore the idea of using a global control center to coordinate estimates obtained locally by several local control centers. In this work, we improve upon these prior results by proposing a fully decentralized and distributed estimation algorithm that only assumes local knowledge of the network structure by local control centers, and that allows them to obtain an optimal estimate of the network state in finite time. Since the computation is distributed among the control centers, our procedure appears scalable against the power network dimension, and, furthermore, numerically reliable and accurate.

A second focus of this paper is false data detection and cyber attacks in power systems. Because of the increasing reliance of modern power systems on communication networks, the possibility of cyber attacks is a real threat (Meserve, 2007). One possibility for the attacker is to corrupt the data coming from the measuring units and directed to the control center, in order to introduce arbitrary errors in the estimated state, and, consequently, to compromise the performance of control and optimization algorithms (Liu, Reiter & Ning, 2009). This important type of attack is often referred in the power systems literature to as *false data injection attack*. In Xie, Mo, and Sinopoli (2010) it is

shown that a false data injection attack, in addition to destabilizing the grid, may also lead to fluctuations in the electricity market, causing significant economical losses. The presence of false data is classically checked by analyzing the statistical properties of the *estimation residual* $z - H\hat{x}$, where z is the measurements vector, \hat{x} is a state estimate, and H is the state to measurements matrix. For an attack to be successful, the residual needs to remain within a certain confidence level. Accordingly, one approach to circumvent false data injection attacks is to increase the number of measurements so as to obtain a more accurate confidence bound. Clearly, by increasing the number of measurements, the data to be transmitted to the control center increases, and the dimension of the estimation problem grows. By means of our estimation method, we address this dimensionality problem by distributing the false data detection problem among several control centers.

1.2. Related work on distributed estimation and projection methods

Starting from the eighties, the problem of distributed estimation has attracted intense attention from the scientific community, generating along the years a very rich literature. More recently, because of the advent of highly integrated and low-cost wireless devices as key components of large autonomous networks, the interest for this classical topic has been renewed. For a wireless sensor network, novel applications requiring efficient distributed estimation procedures include, for instance, environment monitoring, surveillance, localization, and target tracking. Considerable effort has been devoted to the development of distributed and adaptive filtering schemes, which generalize the notion of adaptive estimation to a setup involving networked sensing and processing devices (Cattivelli, Lopes, & Sayed, 2008). In this context, relevant methods include incremental Least Mean-Square (Lopes & Sayed, 2007), incremental Recursive Least-Square (Sayed & Lopes, 2007), Diffusive Least Mean-Square (Sayed & Lopes, 2007), and Diffusive Recursive Least-Square (Cattivelli et al., 2008). Diffusion Kalman filtering and smoothing algorithms are proposed, for instance, in Carli, Chiuso, Schenato, and Zampieri (2008), Cattivelli and Sayed (2010), and consensus based techniques in (Schizas, Ribeiro, & Giannakis, 2007; Schizas, Mateos, & Giannakis, 2009). We remark that the strategies proposed in the aforementioned references could be adapted for the solution of the power network static estimation problem. Their performance, however, appear to be not well suited in our context for the following reasons. First, the convergence of the above estimation algorithms is only asymptotic, and it depends upon the communication topology. As a matter of fact, for many communication topologies, such as Cayley graphs and random geometric graphs, the convergence rate is very slow and scales badly with the network dimension. Such slow convergence rate is clearly undesirable because a delayed state estimation could lead the power plant to instability. Second, approaches based on Kalman filtering require the knowledge of the global state and observation model by all the components of the network, and they violate therefore our assumptions. An exception is constituted by Stankovic, Stankovic, and Stipanovic (2009), where an estimation technique based on local Kalman filters and a consensus strategy is developed. This latter method, however, besides exhibiting asymptotic convergence, does not offer guarantees on the final estimation error. Third and finally, the application of these methods to the detection of cyber attacks, which is also our goal, is not straightforward, especially when detection guarantees are required.

Our estimation technique belongs to the family of Kaczmarz (row-projection) methods for the solution of a linear system of equations. See Censor (1981), Gordon, Bender, and Herman (1970), Kaczmarz (1937), Tanabe (1971) for a detailed discussion. Differently from the existing row-action methods, our algorithms exhibit finite time convergence, and they can be used to compute the weighted least squares solution to a system of linear equations with arbitrary weights.

1.3. Our contributions

The contributions of this work are threefold. First, we adopt the static state network estimation model, in which the state vector is linearly related to the network measurements. We develop two methods for a group of interconnected control centers to compute an optimal estimate of the system state via distributed computation. Our first estimation algorithm assumes an *incremental* mode of cooperation among the control centers, while our second estimation algorithm is based upon a *diffusive* strategy. Both methods are shown to converge in a finite number of iterations, and to require only local information for their implementation. Differently from Rakpenthai, Premrudeepreechacharn, Uatrongjit, and Watson (2005), our estimation procedures assume neither the measurement error covariance nor the measurements matrix to be diagonal. Furthermore, our algorithms are advantageous from a communication perspective, since they reduce the distance between remote terminal units and the associated control center, and from a computational perspective, since they distribute the measurements to be processed among the control centers. Second, as a minor contribution, we describe a finite-time algorithm to detect via distributed computation if the measurements have been corrupted by a malignant agent. Our detection method is based upon our state estimation technique, and it inherits its convergence properties. Notice that, since we assume the measurements to be corrupted by noise, the possibility exists for an attacker to compromise the network measurements while remaining undetected (by injecting for instance a vector with the same noise statistics). With respect to this limitation, we characterize the class of corrupted vectors that are guaranteed to be detected by our procedure, and we show optimality with respect to a centralized detection algorithm. Third, we study the scalability of our methods in networks of increasing dimension, and we derive a finite-memory approximation of our diffusive estimation strategy. For this approximation procedure we show that, under a reasonable set of assumptions and independently of the network dimension, each control center is able to recover a good approximation of the state of a certain subnetwork through little computation. Moreover, we provide bounds on the approximation error for each subnetwork. Finally, we illustrate the effectiveness of our procedures on the IEEE 118 bus system.

The rest of the paper is organized as follows. In Section 2 we introduce the problem under consideration, and we describe the mathematical setup. Section 3 contains our results on state estimation and false data detection, and it presents our algorithms. Section 4 describes our approximated state estimation algorithm. In Section 5 we study the IEEE 118 bus system, and we present some simulation results. Finally, Section 6 contains our conclusion.

2. Problem setup and preliminary notions

For a power network, an example of which is reported in Fig. 2, the state at a certain instant of time consists of the voltage angles and magnitudes at all the system buses. The (static) state estimation problem introduced in the seminal work by Schweppe and Wildes (1970a) refers to the procedure of estimating the state of a power network given a set of measurements of the network variables, such as voltages, currents, and power flows along the transmission lines. To be more precise, let $x \in \mathbb{R}^n$ and $z \in \mathbb{R}^p$ be, respectively, the state and measurements vectors. Then, the vectors x and z are related by the relation

$$z = h(x) + \eta, \tag{1}$$

where $h(\cdot)$ is a nonlinear measurement function, and where η , which is traditionally assumed to be a zero mean random vector satisfying $\mathbb{E}[\eta\eta^T] = \Sigma_\eta = \Sigma_\eta^T > 0$, is the measurements

noise. An optimal estimate of the network state coincides with the *most likely* vector \hat{x} that solves Eq. (1). It should be observed that, instead of by solving the above estimation problem, the network state could be obtained by measuring directly the voltage phasors by means of phasor measurement devices.² Such an approach, however, would be economically expensive, since it requires a phasor measurement device at each network bus, and it would be very vulnerable to communication failures (Abur & Exposito, 2004). In this work, we adopt the approximated estimation model presented in Schweppe and Wildes (1970b), which follows from the linearization around an operating point of Eq. (1). Specifically, we have

$$z = Hx + v, \tag{2}$$

where $H \in \mathbb{R}^{p \times n}$ and where v , the measurements noise, is such that $\mathbb{E}[v] = 0$ and $E[vv^T] = \Sigma = \Sigma^T > 0$. Observe that, because of the interconnection structure of a power network, the measurement matrix H is usually sparse. Let $\text{Ker}(H)$ denote the null space of H , and assume $\text{Ker}(H) = \{0\}$. Recall from Luenberger (1969) that the vector

$$x_{\text{wls}} = (H^T \Sigma^{-1} H)^{-1} H^T \Sigma^{-1} z \tag{3}$$

minimizes the weighted variance of the estimation error, i.e., $x_{\text{wls}} = \arg \min_{\hat{x}} (z - H\hat{x})^T \Sigma^{-1} (z - H\hat{x})$.

The centralized computation of x_{wls} assumes the complete knowledge of the matrices H and Σ , and it requires the inversion of the matrix $H^T \Sigma^{-1} H$. For a large power network, such computation imposes a limitation on the dimension of the matrix H , and hence on the number of measurements that can be efficiently processed to obtain a real-time state estimate. Since the performance of network control and optimization algorithms depend upon the precision of the state estimate, a limitation on the number of network measurements constitutes a bottleneck toward the development of a more efficient power grid. A possible solution to address this complexity problem is to distribute the computation of x_{wls} among geographically deployed control centers (monitors), in a way that each monitor is responsible for a subpart of the whole network. To be more precise, let the matrices H and Σ , and the vector z be partitioned as³

$$H = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_m \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \Sigma_1 \\ \Sigma_2 \\ \vdots \\ \Sigma_m \end{bmatrix}, \quad z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix}, \tag{4}$$

where, for $i \in \{1, \dots, m\}$, $m_i \in \mathbb{N}$, $H_i \in \mathbb{R}^{m_i \times n}$, $\Sigma_i \in \mathbb{R}^{m_i \times p}$, $z_i \in \mathbb{R}^{m_i}$, and $\sum_{i=1}^m m_i = p$. Let $G = (V, \mathcal{E})$ be the connected graph in which each vertex $i \in V = \{1, \dots, m\}$ denotes a monitor, and $\mathcal{E} \in V \times V$ denotes the set of monitors interconnections. For $i \in \{1, \dots, m\}$, assume that monitor i knows the matrices H_i , Σ_i , and the vector z_i , and that two neighboring monitors are allowed to cooperate by exchanging information. Notice that, if the full matrices H and Σ are nowhere available, and if they cannot be used for the computation of x_{wls} , then, with no cooperation among the monitors, the vector x_{wls} cannot be computed by any of the monitor. Hence we consider the following problem.

Problem 1 (Distributed State Estimation). Design an algorithm for the monitors to compute the minimum variance estimate of the network state via distributed computation.

² Phasor measurement units are devices that synchronize by using GPS signals, and that allow for a direct measurement of voltage and current phasors.

³ In most application the error covariance matrix is assumed to be diagonal, so that each submatrix Σ_i is very sparse. However, we do not impose any particular structure on the error covariance matrix.

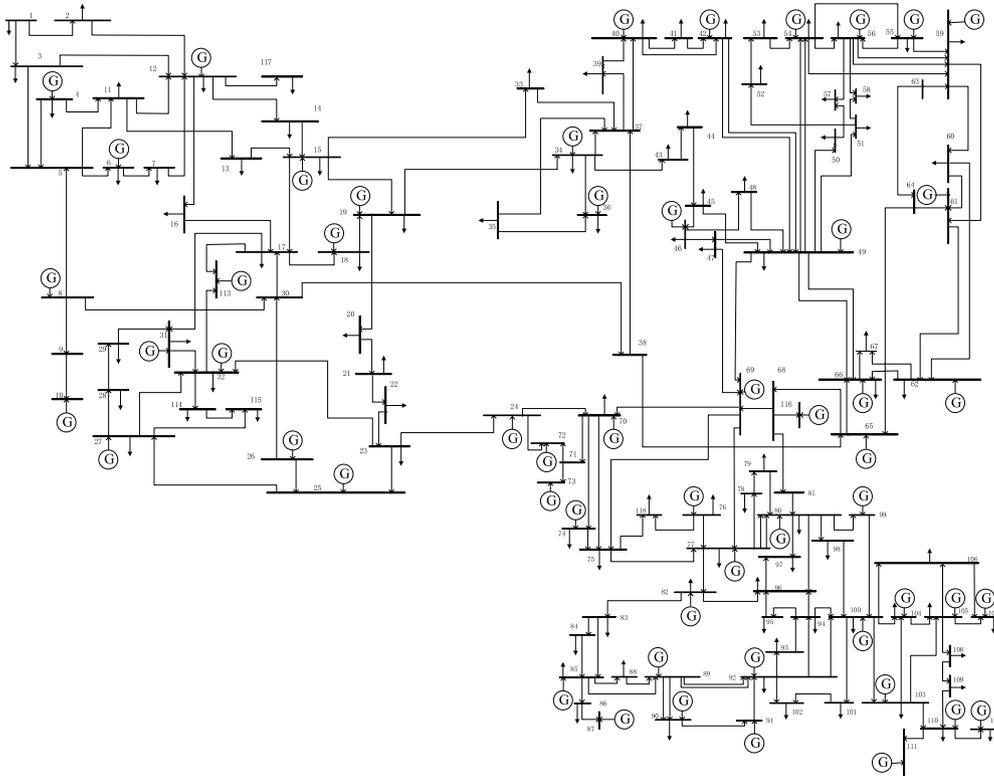


Fig. 2. This figure shows the diagram of the IEEE 118 bus system (courtesy of the IIT Power Group). The network has 118 buses, 186 branches, 99 loads, and 54 generators.

We now introduce the second problem addressed in this work. Given the distributed nature of a power system and the increasing reliance on local area networks to transmit data to control centers, there exists the possibility for an attacker to compromise the network functionalities by corrupting the measurements vector. When a malignant agent corrupts some of the measurements, the state to measurements relation becomes

$$z = Hx + v + w,$$

where the vector $w \in \mathbb{R}^p$ is chosen by the attacker, and consequently, it is unknown and unmeasurable by any of the monitoring stations. We refer to the vector w as *false data*. From the above equation, it should be observed that there exist vectors w that cannot be detected through the measurements z . For instance, if the false data vector is intentionally chosen such that $w \in \text{Im}(H)$, then the attack cannot be detected through the measurements z . Indeed, denoting with † the pseudoinverse operation, the vector $x + H^\dagger w$ is a valid network state. We assume that the vector w is detectable from the measurements z , and we consider the following problem.

Problem 2 (Distributed Detection). Design an algorithm for the monitors to detect the presence of false data in the measurements via distributed computation.

As it will be clear in the sequel, the complexity of our methods depends upon the dimension of the state, as well as the number of monitors. In particular, few monitors should be used in the absence of severe computation and communication constraints, while many monitors are preferred otherwise. We believe that a suitable choice of the number of monitors depends upon the specific scenario, and it is not further discussed in this work.

Remark 1 (Generality of Our Methods). In this paper we focus on the state estimation and the false data detection problems for

power systems. The methods described in the following sections, however, are general, and they have applicability beyond the power network scenario. For instance, our procedures can be used for state estimation and false data detection in dynamical system, as described in Pasqualetti, Carli, Bicchi, and Bullo (2010) for the case of sensors networks.

3. Optimal state estimation and false data detection via distributed computation

The objective of this section is the design of distributed methods to compute an optimal state estimate from measurements. With respect to a centralized method, in which a powerful central processor is in charge of processing all the data, our procedures require the computing units to have access to only a subset of the measurements, and are shown to reduce significantly the computational burden. In addition to being convenient for the implementation, our methods are also optimal, in the sense that they maintain the same estimation accuracy of a centralized method.

For a distributed method to be implemented, the interaction structure among the computing units needs to be defined. Here we consider two modes of cooperations among the computing units, and, namely, the *incremental* and the *diffusive* interactions. In an incremental mode of cooperation, information flows in a sequential manner from one node to the adjacent one. This setting, which usually requires the least amount of communications (Rabbat & Nowak, 2005), induces a cyclic interaction graph among the processors. In a diffusive strategy, instead, each node exchanges information with all (or a subset of) its neighbors as defined by an interaction graph. In this case, the amount of communication and computation is higher than in the incremental case, but each node possesses a good estimate before the termination of the algorithm, since it improves its estimate at each

Algorithm 1 Incremental minimum norm solution (*i*-th monitor)

Input: H_i, z_i ;
Require: $[z_1^T \dots z_m^T]^T \in \text{Im}([H_1^T \dots H_m^T]^T)$;
if $i = 1$ **then**
 $\hat{x}_0 := 0, K_0 := I_n$;
else
 receive \hat{x}_{i-1} and K_{i-1} from monitor $i - 1$;
end if
 $\hat{x}_i := \hat{x}_{i-1} + K_{i-1}(H_i K_{i-1})^\dagger (z_i - H_i \hat{x}_{i-1})$;
 $K_i := \text{Basis}(K_{i-1} \text{Ker}(H_i K_{i-1}))$;
if $i < m$ **then**
 transmit \hat{x}_i and K_i to monitor $i + 1$;
else
 return \hat{x}_m ;
end if

communication round. This section is divided into three parts. In Section 3.2, we first develop a distributed incremental method to compute the minimum norm solution to a set of linear equations, and then exploit such method to solve a minimum variance estimation problem. In Section 3.3 we derive a diffusive strategy which is amenable to asynchronous implementation. Finally, in Section 3.4 we propose a distributed algorithm for the detection of false data among the measurements. Our detection procedure requires the computation of the minimum variance state estimate, for which either the incremental or the diffusive strategy can be used.

3.1. Incremental solution to a set of linear equations

We start by introducing a distributed incremental procedure to compute the minimum norm solution to a set of linear equations. This procedure constitutes the key ingredient of the incremental method we later propose to solve a minimum variance estimation problem. Let $H \in \mathbb{R}^{p \times m}$ and let $z \in \text{Im}(H)$, where $\text{Im}(H)$ denotes the range space spanned by the matrix H . Consider the system of linear equations $z = Hx$, and recall that the unique minimum norm solution to $z = Hx$ coincides with the vector \hat{x} such that $z = H\hat{x}$ and $\|\hat{x}\|_2$ is minimum. It can be shown that $\|\hat{x}\|_2$ being minimum corresponds to \hat{x} being orthogonal to the null space $\text{Ker}(H)$ (Luenberger, 1969). Let H and z be partitioned in m blocks as in (4), and let $G = (V, \mathcal{E})$ be a directed graph such that $V = \{1, \dots, m\}$ corresponds to the set of monitors, and, denoting with (i, j) the directed edge from j to i , $\mathcal{E} = \{(i + 1, i) : i = 1, \dots, m - 1\} \cup \{(1, m)\}$. Our incremental procedure to compute the minimum norm solution to $z = H\hat{x}$ is in Algorithm 1, where, given a subspace \mathcal{V} , we write $\text{Basis}(\mathcal{V})$ to denote any full rank matrix whose columns span the subspace \mathcal{V} . We now proceed with the analysis of the convergence properties of the *Incremental minimum norm solution* algorithm.

Theorem 3.1 (Convergence of Algorithm 1). *Let $z = Hx$, where H and z are partitioned in m row-blocks as in (4). In Algorithm 1, the m -th monitor returns the vector \hat{x} such that $z = H\hat{x}$ and $\hat{x} \perp \text{Ker}(H)$.*

Proof. See Appendix A.1. \square

It should be observed that the dimension of K_i decreases, in general, when the index i increases. In particular, $K_m = \{0\}$ and $K_1 = \text{Ker}(H_1)$. To reduce the communication burden of the algorithm, monitor i could transmit the smallest among $\text{Basis}(K_{i-1} \text{Ker}(H_i K_{i-1}))$ and $\text{Basis}(K_{i-1} \text{Ker}(H_i K_{i-1})^\perp)$, together with a packet containing the type of the transmitted basis.

Table 1
Computational complexity of Algorithm 1.

Block size	<i>i</i> -th complexity	Total complexity
$k \leq n$	$O(k^2 n)$	$O(mkn)$
$k > n$	$O(kn^2)$	$O(mn^2)$

Remark 2 (Computational Complexity of Algorithm 1). In Algorithm 1, the main operation to be performed by the *i*-th agent is a singular value decomposition (SVD).⁴ Indeed, since the range space and the null space of a matrix can be obtained through its SVD, both the matrices $(H_i K_{i-1})^\dagger$ and $\text{Basis}(K_{i-1} \text{Ker}(H_i K_{i-1}))$ can be recovered from the SVD of $H_i K_{i-1}$. Let $H \in \mathbb{R}^{m \times n}$, $m > n$, and assume the presence of $\lceil m/k \rceil$ monitors, $1 \leq k \leq m$. Recall that, for a matrix $M \in \mathbb{R}^{k \times p}$, the singular value decomposition can be performed with complexity $O(\min\{kp^2, k^2 p\})$ (Golub & van Loan, 1989). Hence, the computational complexity of computing a minimum norm solution to the system $z = Hx$ is $O(mn^2)$. In Table 1 we report the computational complexity of Algorithm 1 as a function of the block size k .

The following observations are in order. First, if $k \leq n$, then the computational complexity sustained by the *i*-th monitor is much smaller than the complexity of a centralized implementation, i.e., $O(k^2 n) \ll O(mn^2)$. Second, the complexity of the entire algorithm is optimal, since, in the worst case, it maintains the computational complexity of a centralized solution, i.e., $O(mkn) \leq O(mn^2)$. Third and finally, a compromise exists between the blocks size k and the number of communications needed to terminate Algorithm 1. In particular, if $k = m$, then no communication is needed, while, if $k = 1$, then $m - 1$ communication rounds are necessary to terminate the estimation algorithm.⁵

3.2. Incremental estimation via distributed computation

We now focus on the computation of the weighted least squares solution to a set of linear equations. Let v be an unknown and unmeasurable random vector, with $\mathbb{E}(v) = 0$ and $\mathbb{E}(vv^T) = \Sigma = \Sigma^T > 0$. Consider the system of equations

$$z = Hx + v, \quad (5)$$

and assume $\text{Ker}(H) = 0$. Notice that, because of the noise vector v , we generally have $z \notin \text{Im}(H)$, so that Algorithm 1 cannot be directly employed to compute the vector x_{wls} defined in (3). It is possible, however, to recast the above weighted least squares estimation problem to be solvable with Algorithm 1. Note that, because the matrix Σ is symmetric and positive definite, there exists⁶ a full row rank matrix B such that $\Sigma = BB^T$. Then, Eq. (5) can be rewritten as

$$z = \begin{bmatrix} H & \varepsilon B \end{bmatrix} \begin{bmatrix} x \\ \bar{v} \end{bmatrix}, \quad (6)$$

where $\varepsilon \in \mathbb{R}_{>0}$, $\mathbb{E}[\bar{v}] = 0$ and $\mathbb{E}[\bar{v}\bar{v}^T] = \varepsilon^{-2}I$. Observe that, because B has full row rank, the system (6) is underdetermined, i.e., $z \in \text{Im}([H \ \varepsilon B])$ and $\text{Ker}([H \ \varepsilon B]) \neq 0$. Let

$$\begin{bmatrix} \hat{x}(\varepsilon) \\ \hat{\bar{v}} \end{bmatrix} = [H \ \varepsilon B]^\dagger z. \quad (7)$$

⁴ The matrix H is usually very sparse, since it reflects the network interconnection structure. If the matrices $H_i K_{i-1}$ are also sparse, then efficient SVD algorithms for very large sparse matrices can be employed (cf. SVDPACK). In general, if the dimension of $H_i K_{i-1}$ is too large for an SVD algorithm to be numerically reliable, then additional monitors should be used to reduce the dimension of $H_i K_{i-1}$.

⁵ Additional $m - 1$ communication rounds are needed to transmit the estimate to all monitors.

⁶ Choose for instance $B = W\Lambda^{1/2}$, where W is a basis of eigenvectors of Σ and Λ is the corresponding diagonal matrix of the eigenvalues.

The following theorem characterizes the relation between the minimum variance estimate x_{wls} and $\hat{x}(\varepsilon)$.

Theorem 3.2 (Convergence with ε). Consider the system of linear equations $z = Hx + v$. Let $\mathbb{E}[v] = 0$ and $\mathbb{E}[vv^T] = \Sigma = BB^T > 0$ for a full row rank matrix B . Let

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

Then

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

and

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

Proof. See Appendix A.2. \square

Throughout the paper, let $\hat{x}(\varepsilon)$ be the vector defined in (7), and notice that Theorem 3.2 implies that

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

Remark 3 (Incremental State Estimation). For the system of equations $z = Hx + v$, let BB^T be the covariance matrix of the noise vector v , and let

$$H = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_m \end{bmatrix}, \quad B = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_m \end{bmatrix}, \quad z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix}, \quad (8)$$

where $m_i \in \mathbb{N}$, $H_i \in \mathbb{R}^{m_i \times n}$, $B_i \in \mathbb{R}^{m_i \times p}$, and $z_i \in \mathbb{R}^{m_i}$. For $\varepsilon > 0$, the estimate $\hat{x}(\varepsilon)$ of the weighted least squares solution to $z = Hx + v$ can be computed by means of Algorithm 1 with input $[H_i \ \varepsilon B_i]$ and z_i .

Observe now that the estimate $\hat{x}(\varepsilon)$ coincides with \hat{x}_{wls} only in the limit for $\varepsilon \rightarrow 0^+$. When the parameter ε is fixed, the estimate $\hat{x}(\varepsilon)$ differs from the minimum variance estimate \hat{x}_{wls} . We next characterize the approximation error $x_{\text{wls}} - \hat{x}(\varepsilon)$.

Corollary 3.1 (Approximation Error). Consider the system $z = Hx + v$, and let $\mathbb{E}[vv^T] = BB^T$ for a full row rank matrix B . Then

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

where D is as in Theorem 3.2.

Proof. With the same notation as in the proof of Theorem 3.2, for every value of $\varepsilon > 0$, the difference $x_{\text{wls}} - \hat{x}(\varepsilon)$ equals

$$((H^T \Sigma^{-1} H)^{-1} H^T \Sigma^{-1} - H^\dagger + \varepsilon H^\dagger B(C^\dagger + D)) z.$$

Since $(H^T \Sigma^{-1} H)^{-1} H^T \Sigma^{-1} - H^\dagger + \varepsilon H^\dagger B C^\dagger = 0$ for every $\varepsilon > 0$, it follows $x_{\text{wls}} - \hat{x}(\varepsilon) = \varepsilon H^\dagger B D z$. \square

Therefore, for the solution of system (5) by means of Algorithm 1, the parameter ε is chosen according to Corollary 3.1 to meet a desired estimation accuracy. It should be observed that, even if the entire matrix H needs to be known for the computation of the exact parameter ε , the advantages of our estimation technique are preserved. Indeed, if the matrix H is unknown and an upper bound for $\|H^\dagger B D z\|$ is known, then a value for ε can still be computed that guarantees the desired estimation accuracy. On the other hand, if H is entirely known, it may be inefficient to use H to perform a centralized state estimation over time. Instead, a suitable parameter ε needs to be computed only once. To conclude this section we characterize the estimation residual $z - H\hat{x}$. This quantity plays an important role for the synthesis of a distributed false data detection algorithm.

Corollary 3.2 (Estimation Residual). Consider the system $z = Hx + v$, and let $\mathbb{E}[vv^T] = \Sigma = \Sigma^T > 0$. Then⁷

$$\lim_{\varepsilon \rightarrow 0^+} \|z - H\hat{x}(\varepsilon)\| \leq \|(I - HW)\| \|v\|,$$

where $W = (H^T \Sigma^{-1} H)^{-1} H^T \Sigma^{-1}$.

Proof. By virtue of Theorem 3.2 we have

$$\lim_{\varepsilon \rightarrow 0^+} \hat{x}(\varepsilon) = x_{\text{wls}} = (H^T \Sigma^{-1} H)^{-1} H^T \Sigma^{-1} z = Wz.$$

Observe that $HWH = H$, and recall that $z = Hx + v$. For any matrix norm, we have

$$\|z - Hx_{\text{wls}}\| = \|z - HWz\| = \|(I - HW)(Hx + v)\| \\ = \|Hx - Hx + (I - HW)v\| \\ \leq \|(I - HW)\| \|v\|,$$

and the theorem follows. \square

3.3. Diffusive estimation via distributed computation

The implementation of the incremental state estimation algorithm described in Section 3.2 requires a certain degree of coordination among the control centers. For instance, an ordering of the monitors is necessary, such that the i -th monitor transmits its estimate to the $(i + 1)$ -th monitor. This requirement imposes a constraint on the monitors interconnection structure, which may be undesirable, and, potentially, less robust to link failures. In this section, we overcome this limitation by presenting a diffusive implementation of Algorithm 1, which only requires the monitors interconnection structure to be connected.⁸ To be more precise, let $V = \{1, \dots, m\}$ be the set of monitors, and let $G = (V, E)$ be the undirected graph describing the monitors interconnection structure, where $E \subseteq V \times V$, and $(i, j) \in E$ if and only if the monitors i and j are connected. The neighbor set of node i is defined as $N_i = \{j \in V: (i, j) \in E\}$. We assume that G is connected, and we let the distance between two vertices be the minimum number of edges in a path connecting them. Finally, the diameter of a graph G , in short $\text{diam}(G)$, equals the greatest distance between any pair of vertices. Our diffusive procedure is described in Algorithm 2, where the matrices H_i and εB_i are as defined in Eq. (8). During the h -th iteration of the algorithm, monitor i , with $i \in \{1, \dots, N\}$, performs the following three actions in order:

- (i) transmits its current estimates \hat{x}_i and K_i to all its neighbors;
- (ii) receives the estimates \hat{x}_j from neighbors N_i ; and
- (iii) updates \hat{x}_i and K_i as in the *for* loop of Algorithm 2.

We now show the convergence properties of Algorithm 2.

Theorem 3.3 (Convergence of Algorithm 2). Consider the system of linear equations $z = Hx + v$, where $\mathbb{E}[v] = 0$ and $\mathbb{E}[vv^T] = BB^T$. Let H , B and z be partitioned as in (8), and let $\varepsilon > 0$. Let the monitors communication graph be connected, let d be its diameter, and let the monitors execute the Diffusive state estimation algorithm. Then, each monitor computes the estimate $\hat{x}(\varepsilon)$ of x in d steps.

Proof. Let \hat{x}_i be the estimate of the monitor i , and let K_i be such that $[x^T \ v^T]^T - \hat{x}_i \in \text{Im}(K_i)$, where x denotes the network state, v is the measurements noise vector, and $\hat{x}_i \perp \text{Im}(K_i)$. Notice that $z_i = [H_i \ \varepsilon B_i] \hat{x}_i$, where z_i is the i -th measurements vector. Let i and j be two neighboring monitors. Notice that there exist vectors v_i and

Algorithm 2 Diffusive state estimation (*i*-th monitor)

Input: $H_i, \varepsilon B_i, z_i$;
 $\hat{x}_i := [H_i \ \varepsilon B_i]^\dagger z_i$;
 $K_i := \text{Basis}(\text{Ker}([H_i \ \varepsilon B_i]));$
while $K_i \neq 0$ **do**
 for $j \in N_i$ **do**
 receive \hat{x}_j and K_j ;
 $\hat{x}_i := \hat{x}_i + [K_i \ 0][-K_i \ K_j]^\dagger(\hat{x}_i - \hat{x}_j)$;
 $K_i := \text{Basis}(\text{Im}(K_i) \cap \text{Im}(K_j))$;
 end for
 transmit \hat{x}_i and K_i ;
end while

v_j such that $\hat{x}_i + K_i v_i = \hat{x}_j + K_j v_j$. In particular, those vectors can be chosen as

$$\begin{bmatrix} v_i \\ v_j \end{bmatrix} = [-K_i \ K_j]^\dagger(\hat{x}_i - \hat{x}_j).$$

It follows that the vector

$$\hat{x}_i^+ = \hat{x}_i + [K_i \ 0][-K_i \ K_j]^\dagger(\hat{x}_i - \hat{x}_j)$$

is such that $z_i = [H_i \ \varepsilon B_i] \hat{x}_i^+$ and $z_j = [H_j \ \varepsilon B_j] \hat{x}_j^+$. Moreover we have $\hat{x}_i^+ \perp (\text{Im}(K_i) \cap \text{Im}(K_j))$. Indeed, notice that

$$\begin{bmatrix} v_i \\ v_j \end{bmatrix} \perp \text{Ker}([-K_i \ K_j]) \supseteq \left\{ \begin{bmatrix} w_i \\ w_j \end{bmatrix} : K_i w_i = K_j w_j \right\}.$$

We now show that $K_i v_i \perp \text{Im}(K_j)$. By contradiction, if $K_i v_i \not\perp \text{Im}(K_j)$, then $v_i = \tilde{v}_i + \bar{v}_i$, with $K_i \tilde{v}_i \perp \text{Im}(K_j)$ and $K_i \bar{v}_i \in \text{Im}(K_j)$. Let $\bar{v}_j = K_j^\dagger K_i \bar{v}_i$, and $\tilde{v}_j = v_j - \bar{v}_j$. Then, $[\bar{v}_i^\top \ \bar{v}_j^\top]^\top \in \text{Ker}([-K_i \ K_j])$, and hence $[v_i^\top \ v_j^\top]^\top \not\perp \text{Ker}([-K_i \ K_j])$, which contradicts the hypothesis. We conclude that $[K_i \ 0][-K_i \ K_j]^\dagger(\hat{x}_i - \hat{x}_j) \perp \text{Im}(K_j)$, and, since $\hat{x}_i \perp \text{Im}(K_i)$, it follows $\hat{x}_i^+ \perp (\text{Im}(K_i) \cap \text{Im}(K_j))$. The theorem follows from the fact that after a number of steps equal to the diameter of the monitors communication graph, each vector \hat{x}_i verifies all the measurements, and $\hat{x}_i \perp \text{Im}(K_1) \cap \dots \cap \text{Im}(K_m)$. \square

As a consequence of [Theorem 3.2](#), in the limit for ε to zero, Algorithm 2 returns the minimum variance estimate of the state vector, being therefore the diffusive counterpart of Algorithm 1. A detailed comparison between incremental and diffusive methods is beyond the purpose of this work, and we refer the interested reader to [Lopes and Sayed \(2007, 2008\)](#) and the references therein for a thorough discussion. Here we only underline some key differences. While Algorithm 1 requires less operations, being therefore computationally more efficient, Algorithm 2 does not constraint the monitors communication graph. Additionally, Algorithm 2 can be implemented adopting general asynchronous communication protocols. For instance, consider the *Asynchronous (diffusive) state estimation* algorithm, where, at any given instant of time, at most one monitor, say j , sends its current estimates to its neighbors, and where, for $i \in N_j$, monitor i performs the following operations:

- (i) $\hat{x}_i := \hat{x}_i + [K_i \ 0][-K_i \ K_j]^\dagger(\hat{x}_i - \hat{x}_j)$,
- (ii) $K_i := \text{Basis}(\text{Im}(K_i) \cap \text{Im}(K_j))$.

Corollary 3.3 (*Asynchronous Estimation*). *Consider the system of linear equations $z = Hx + v$, where $\mathbb{E}[v] = 0$ and $\mathbb{E}[vv^\top] = BB^\top$. Let H, B and z be partitioned as in (8), and let $\varepsilon > 0$. Let the monitors*

⁷ Given a vector v and a matrix H , we denote by $\|v\|$ any vector norm, and by $\|H\|$ the corresponding induced matrix norm.

⁸ An undirected graph is said to be connected if there exists a path between any two vertices ([Godsil & Royle, 2001](#)).

Algorithm 3 False data detection (*i*-th monitor)

Input: $H_i, \varepsilon B_i, \Gamma$;
while True **do**
 collect measurements $z_i(t)$;
 estimate network state $\hat{x}(t)$ via Algorithm 1 or 2;
 if $\|z_i(t) - H_i \hat{x}(t)\|_\infty > \Gamma$ **then**
 return false data detected;
 end if
end while

communication graph be connected, let d be its diameter, and let the monitors execute the *Asynchronous (diffusive) state estimation* algorithm. Assume that there exists a duration $T \in \mathbb{R}$ such that, within each time interval of duration T , each monitor transmits its current estimates to its neighbors. Then, each monitor computes the estimate $\hat{x}(\varepsilon)$ of x within time dT .

Proof. The proof follows from the following two facts. First, the intersection of subspaces is a commutative operation. Second, since each monitor performs a data transmission within any time interval of length T , it follows that, at time dT , the information related to one monitor has propagated through the network to all monitors. \square

3.4. Detection of false data via distributed computation

In the previous sections we have shown how to compute an optimal state estimate via distributed computation. A rather straightforward application of the proposed state estimation technique is the detection of false data among the measurements. When the measurements are corrupted, the state to measurements relation becomes

$$z = Hx + v + w,$$

where w is the false data vector. As a consequence of [Corollary 3.2](#), the vector w is detectable if it affects significantly the estimation residual, i.e., if $\lim_{\varepsilon \rightarrow 0} \|z - H\hat{x}(\varepsilon)\| > \Gamma$, where the threshold Γ depends upon the magnitude of the noise v . Notice that, because false data can be injected at any time by a malignant agent, the detection algorithm needs to be executed over time by the control centers. Let $z(t) = Hx(t) + v(t) + w(t)$ be the measurements vector at a given time instant t . Based on these considerations, our distributed detection procedure is in Algorithm 3, where the matrices H_i and εB_i are as defined in Eq. (8), and Γ is a predefined threshold.

In Algorithm 3, the value of the threshold Γ determines the false alarm and the missed detection rates. Clearly, if $\Gamma \geq \|(I - HW)\| \|v(t)\|$ at all times t , and ε is sufficiently small, then no false alarm is triggered, at the expenses of the missed detection rate. By decreasing the value of Γ the sensitivity to failures increases together with the false alarm rate. Notice that, if the magnitude of the noise signals is upper bounded by γ , then a reasonable choice of the threshold is $\Gamma = \gamma \|(I - HW)\|_\infty$, where the use of the infinity norm in Algorithm 3 is also convenient for the implementation. Indeed, since the condition $\|z(t) - H\hat{x}(t)\|_\infty > \Gamma$ is equivalent to $\|z_i(t) - H_i \hat{x}(t)\|_\infty > \Gamma$ for some monitor i , the presence of false data can be independently checked by each monitor without further computation. Notice that an eventual alarm message needs to be propagated to all control centers.

Remark 4 (*Statistical Detection*). A different strategy for the detection of false data relies on statistical techniques, e.g., see [Abur and Exposito \(2004\)](#). In the interest of brevity, we do not consider these methods, and we only remark that, once the estimation residual has been computed by each monitor, the implementation of a (distributed) statistical procedure, such as, for instance, the (distributed) χ^2 -Test, is a straightforward task.

4. A finite-memory estimation technique

The procedure described in Algorithm 1 allows each agent to compute an optimal estimate of the whole network state in finite time. In this section, we allow each agent to handle only local and of small dimension vectors, and we develop a procedure to recover an estimate of only a certain subnetwork. We envision that the knowledge of only a subnetwork may be sufficient to implement distributed estimation and control strategies.

We start by introducing the necessary notation. Let the measurements matrix H be partitioned into m^2 blocks, being m the number of monitors in the network, as

$$H = \begin{bmatrix} H_{11} & \cdots & H_{1m} \\ \vdots & & \vdots \\ H_{m1} & \cdots & H_{mm} \end{bmatrix}, \quad (9)$$

where $H_{ij} \in \mathbb{R}^{m_i \times m_j}$ for all $i, j \in \{1, \dots, m\}$. The above partitioning reflects a division of the whole network into competence regions: we let each monitor be responsible for the correct functionality of the subnetwork defined by its blocks. Additionally, we assume that the union of the different regions covers the whole network, and that different competence regions may overlap. Observe that, in most of the practical situations, the matrix H has a sparse structure, so that many blocks H_{ij} have only zero entries. We associate an undirected graph G_h with the matrix H , in a way that G_h reflects the interconnection structure of the blocks H_{ij} . To be more precise, we let $G_h = (V_h, \mathcal{E}_h)$, where $V_h = \{1, \dots, m\}$ denotes the set of monitors, and where, denoting by (i, j) the undirected edge from j to i , it holds $(i, j) \in \mathcal{E}_h$ if and only if $\|H_{ij}\| \neq 0$ or $\|H_{ji}\| \neq 0$. Noticed that the structure of the graph G_h , besides reflecting the sparsity pattern of the measurement matrix, describes also the monitors interconnections. By using the same partitioning as in (9), the Moore–Penrose pseudoinverse of H can be written as

$$H^\dagger = \tilde{H} = \begin{bmatrix} \tilde{H}_{11} & \cdots & \tilde{H}_{1m} \\ \vdots & & \vdots \\ \tilde{H}_{m1} & \cdots & \tilde{H}_{mm} \end{bmatrix}, \quad (10)$$

where $\tilde{H}_{ij} \in \mathbb{R}^{n_i \times m_j}$. Assume that H has full row rank,⁹ and observe that $H^\dagger = H^\top(HH^\top)^{-1}$. Consider the equation $z = Hx$, and let $H^\dagger z = \hat{x} = [\hat{x}_1^\top \cdots \hat{x}_m^\top]^\top$, where, for all $i \in \{1, \dots, m\}$, $\hat{x}_i \in \mathbb{R}^{n_i}$. We employ Algorithm 2 for the computation of the vector \hat{x} , and we let

$$\hat{x}^{(i,h)} = \begin{bmatrix} \hat{x}_1^{(i,h)} \\ \vdots \\ \hat{x}_m^{(i,h)} \end{bmatrix}$$

be the estimate vector of the i -th monitor after h iterations of Algorithm 2, i.e., after h executions of the *while* loop in Algorithm 2. In what follows, we will show that, for a sufficiently sparse matrix H , the error $\|\hat{x}_i - \hat{x}_i^{(i,h)}\|$ has an exponential decay when h increases, so that it becomes negligible before the termination of Algorithm 2, i.e., when $h < \text{diam}(G_h)$. The main result of this section is next stated.

Theorem 4.1 (Local Estimation). *Let the full-row rank matrix H be partitioned as in (9). Let $[a, b]$, with $a < b$, be the smallest interval containing the spectrum of HH^\top . Then, for $i \in \{1, \dots, m\}$ and $h \in \mathbb{N}$, there exist $C \in \mathbb{R}_{>0}$ and $q \in (0, 1)$ such that*

$$\|\hat{x}_i - \hat{x}_i^{(i,h)}\| \leq Cq^{\frac{h}{2}+1}.$$

For the readers convenience, before proving the above result, we recall the following definitions and results. Given an invertible matrix M of dimension n , let us define the *support sets*

$$S_h(M) = \bigcup_{k=0}^h \{(i, j) : M^k(i, j) \neq 0\},$$

being $M^k(i, j)$ the (i, j) -th entry of M^k , and the *decay sets*

$$D_h(M) = (\{1, \dots, n\} \times \{1, \dots, n\}) \setminus S_h(M).$$

Theorem 4.2 (Decay Rate Demko, Moss, & Smith, 1984). *Let M be of full row rank, and let $[a, b]$, with $a < b$, be the smallest interval containing the spectrum of M . There exist $C \in \mathbb{R}_{>0}$ and $q \in (0, 1)$ such that*

$$\sup \{|M^\dagger(i, j)| : (i, j) \in D_h(MM^\top)\} \leq Cq^{h+1}.$$

For a graph G_h and two nodes i and j , let $\text{dist}(i, j)$ denote the smallest number of edges in a path from j to i in G_h . The following result will be used to prove Theorem 4.1. Recall that, for a matrix M , we have $\|M\|_{\max} = \max\{|M(i, j)|\}$.

Lemma 4.1 (Decay Sets and Local Neighborhood). *Let the matrix H be partitioned as in (9), and let G_h be the graph associated with H . For $i, j \in \{1, \dots, m\}$, if $\text{dist}(i, j) = h$, then*

$$\|H_{ij}^\dagger\|_{\max} \leq Cq^{\frac{h}{2}+1}.$$

Proof. A proof of this result follows from simple inspection and it is omitted here. \square

Lemma 4.1 establishes a relationship between the decay sets of an invertible matrix and the distance among the vertices of a graph associated with the same matrix. By using this result, we are now ready to prove Theorem 4.1.

Proof of Theorem 4.1. Notice that, after h iterations of Algorithm 2, the i -th monitor has received data from the monitors within distance h from i , i.e., from the monitors T such that, for each $j \in T$, there exists a path of length up to h from j to i in the graph associated with H . Reorder the rows of H such that the i -th block come first and the T -th blocks second. Let $H = [H_1^\top H_2^\top H_3^\top]^\top$ be the resulting matrix. Accordingly, let $z = [z_1^\top z_2^\top z_3^\top]^\top$, and let $x = [x_1^\top x_2^\top x_3^\top]^\top$, where $z = Hx$.

Because H has full row rank, we have

$$\begin{bmatrix} H_1 \\ H_2 \\ H_3 \end{bmatrix} \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix} = \begin{bmatrix} I_1 & 0 & 0 \\ 0 & I_2 & 0 \\ 0 & 0 & I_3 \end{bmatrix},$$

where I_1, I_2 , and I_3 are identity matrices of appropriate dimension, and

$$H^\dagger = \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix}.$$

For a matrix M , let $\text{col}(M)$ denote the number of columns of M . Let $T_1 = \{1, \dots, \text{col}(P_{11})\}$, $T_2 = \{1 + \text{col}(P_{11}), \dots, \text{col}([P_{11} P_{12}])\}$, and $T_3 = \{1 + \text{col}([P_{11} P_{12}]), \dots, \text{col}([P_{11} P_{12} P_{13}])\}$.

Let T_1, T_2 , and T_3 , be, respectively, the indices of the columns of P_{11}, P_{12} , and P_{13} . Notice that, by construction, if $i \in T_1$ and $j \in T_3$, then $\text{dist}(i, j) > h$. Then, by virtue of Lemma 4.1 and Theorem 4.2, the magnitude of each entry of P_{13} is bounded by $\bar{C}\bar{q}^{\lfloor \frac{h}{2} \rfloor + 1}$, for $\bar{C}, \bar{q} \in \mathbb{R}$.

Because H has full row rank, from Theorem 3.1 we have that

$$\hat{x} = H^\dagger z = \hat{x} + K_1(H_3 K_1)^\dagger (z_3 - H_3 \hat{x}^1), \quad (11)$$

⁹ The case of a full-column rank matrix is treated analogously.

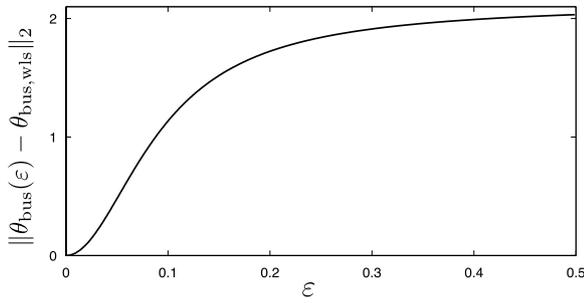


Fig. 3. In this figure the normalized euclidean norm of the error vector $\theta_{\text{bus}}(\varepsilon) - \theta_{\text{bus,wls}}$ is plotted as a function of the parameter ε , where $\theta_{\text{bus}}(\varepsilon)$ is the estimation vector computed according to Theorem 3.2, and $\theta_{\text{bus,wls}}$ is the minimum variance estimate of θ_{bus} . As ε decreases, the vector $\theta_{\text{bus}}(\varepsilon)$ converges to the minimum variance estimate $\theta_{\text{bus,wls}}$.

where

$$\hat{x} = [H_1^T \ H_2^T]^T \dagger [z_1^T \ z_2^T]^T \quad \text{and} \quad K_1 = \text{Basis} \left(\text{Ker} \left([H_1^T \ H_2^T]^T \right) \right).$$

With the same partitioning as before, let $\hat{x} = [\hat{x}_1^T \ \hat{x}_2^T \ \hat{x}_3^T]^T$. In order to prove the theorem, we need to show that there exists $C \in \mathbb{R}_{>0}$ and $q \in (0, 1)$ such that

$$\|\hat{x}_1 - \hat{x}_1\| \leq Cq^{L \frac{n}{2} + 1}.$$

Notice that, for (11) to hold, the matrix K_1 can be any basis of $\text{Ker}([H_1^T \ H_2^T]^T)$. Hence, let $K_1 = [P_{13}^T \ P_{23}^T \ P_{33}^T]$. Because every entry of P_{13} decays exponentially, the theorem follows. \square

In Section 5.2 we provide an example to clarify the exponential decay described in Theorem 4.1.

5. An illustrative example

The effectiveness of the methods developed in the previous sections is now shown through some examples.

5.1. Estimation and detection for the IEEE 118 system

The IEEE 118 bus system represents a portion of the American Electric Power System as of December, 1962. This test case system, whose diagram is reported in Fig. 2, is composed of 118 buses, 186 branches, 54 generators, and 99 loads. The voltage angles θ_{bus} and the power injections P_{bus} at the network buses are assumed to be related through the linear relation

$$P_{\text{bus}} = H_{\text{bus}} \theta_{\text{bus}},$$

where the matrix H_{bus} depends upon the network interconnection structure and the network admittance matrix. For the network in Fig. 2, let $z = P_{\text{bus}} - v$ be the measurements vector, where $\mathbb{E}[v] = 0$ and $\mathbb{E}[vv^T] = \sigma^2 I$, $\sigma \in \mathbb{R}$. Then, following the notation in Theorem 3.2, the minimum variance estimate of θ_{bus} can be recovered as

$$\lim_{\varepsilon \rightarrow 0^+} [H_{\text{bus}} \ \varepsilon \sigma I]^\dagger z.$$

In Fig. 3 we show that, as ε decreases, the estimation vector computed according to Theorem 3.2 converges to the minimum variance estimate of θ_{bus} .

In order to demonstrate the advantage of our decentralized estimation algorithm, we assume the presence of 5 control centers in the network of Fig. 2, each one responsible for a subpart of the entire network. The situation is depicted in Fig. 4. Assume that each control center measures the real power injected at the buses in its area, and let $z_i = P_{\text{bus},i} - v_i$, with $\mathbb{E}[v_i] = 0$ and $\mathbb{E}[v_i v_i^T] = \sigma_i^2 I$, be

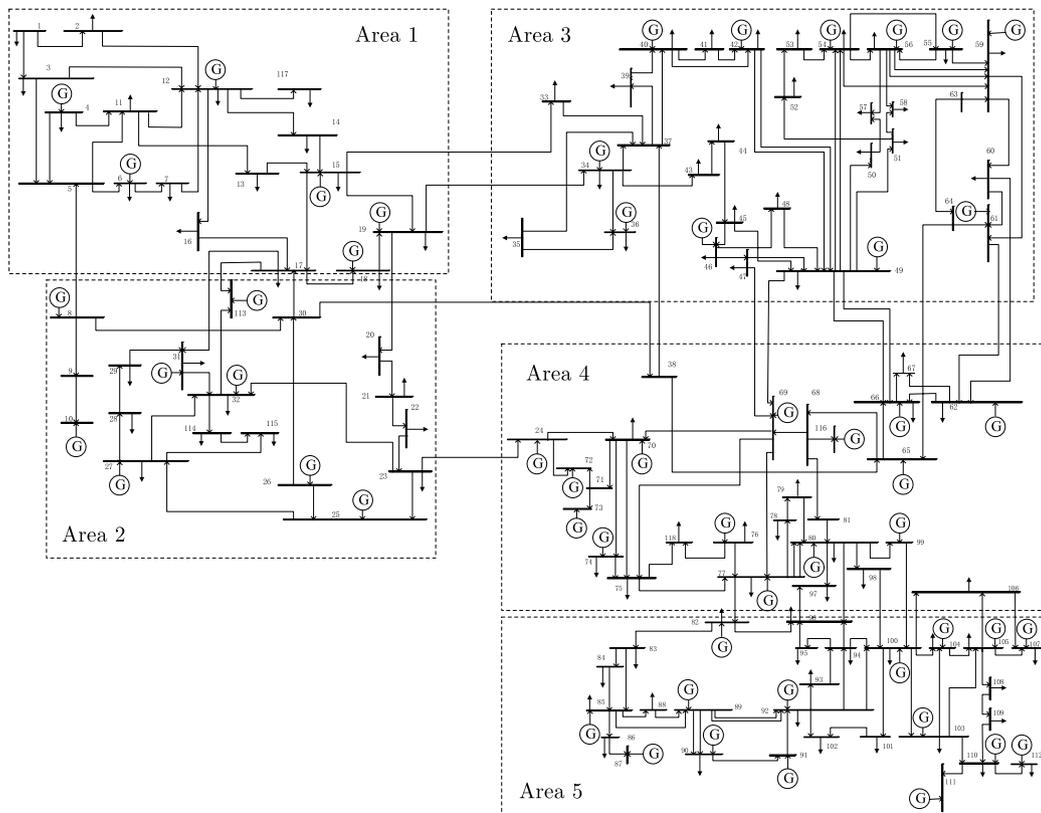


Fig. 4. In this figure the IEEE 118 bus system has been divided into 5 areas. Each area is monitored and operated by a control center. The control centers cooperate to estimate the state and to assess the functionality of the whole network.

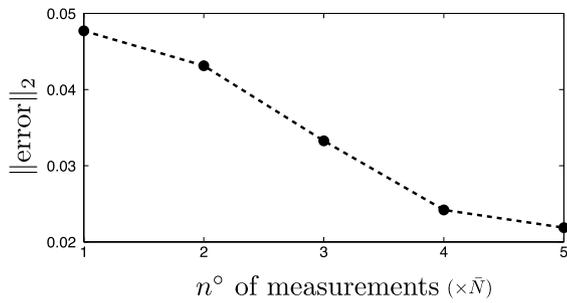


Fig. 5. For a fixed value of ε , Fig. 5 shows the average (over 100 tests) of the norm of the error (with respect to the network state) of the estimate obtained by means of Algorithm 1. The estimation error decreases with the number of measurements. Because of the presence of several control centers, our algorithm processes more measurements (up to $5\bar{N}$) while maintaining the same (or smaller) computational complexity of a centralized estimation with \bar{N} measurements.

the measurements vector of the i -th area. Finally, assume that the i -th control center knows the matrix $H_{\text{bus},i}$ such that $z_i = H_{\text{bus},i}\theta_{\text{bus}} + v_i$. Then, as discussed in Section 3, the control centers can compute an optimal estimate of θ_{bus} by means of Algorithm 1 or Algorithm 2. Let n_i be the number of measurements of the i -th area, and let $N = \sum_{i=1}^5 n_i$. Notice that, with respect to a centralized computation of the minimum variance estimate of the state vector, our estimation procedure obtains the same estimation accuracy while requiring a smaller computational burden and memory requirement. Indeed, the i -th monitor uses n_i measurements instead of N . Let \bar{N} be the maximum number of measurements that, due to hardware or numerical constraints, a control center can efficiently handle for the state estimation problem. In Fig. 5, we increase the number of measurements taken by a control center, so that $n_i \leq \bar{N}$, and we show how the accuracy of the state estimate increases with respect to a single control center with \bar{N} measurements.

To conclude this section, we consider a security application, in which the control centers aim at detecting the presence of false data among the network measurements via distributed computation. For this example, we assume that each control center measures the real power injection as well the current magnitude at some of the buses of its area. By doing so, a sufficient redundancy in the measurements is obtained for the detection to be feasible (Abur & Exposito, 2004). Suppose that the measurements of the power injection at the first bus of the first area is corrupted by a malignant agent. To be more precise, let the measurements vector of the first area be $\bar{z}_1 = z_1 + e_1 w_1$, where e_1 is the first canonical vector, and w_1 is a random variable. For the simulation we choose w_1 to be uniformly distributed in the interval $[0, w_{\text{max}}]$, where w_{max} corresponds approximately to the 10% of the nominal real injection value. In order to detect the presence of false data among the measurements, the control centers implement Algorithm 3, where, being H the measurements matrix, and σ , Σ the noise standard deviation and covariance matrix, the threshold value Γ is chosen as $2\sigma\|I - H(H^T\Sigma^{-1}H)^{-1}H^T\Sigma^{-1}\|_\infty$.¹⁰ The residual functions $\|z_i - H\hat{x}\|_\infty$ are reported in Fig. 6. Observe that, since the first residual is greater than the threshold Γ , the control centers successfully detect the false data. Regarding the identification of the corrupted measurements, we remark that a regional identification may be possible by simply analyzing the residual functions. In this example, for instance, since the residuals 2–5 are below the threshold value, the corrupted data is likely to be among the measurements of the first area. This important aspect is left as the subject of future research.

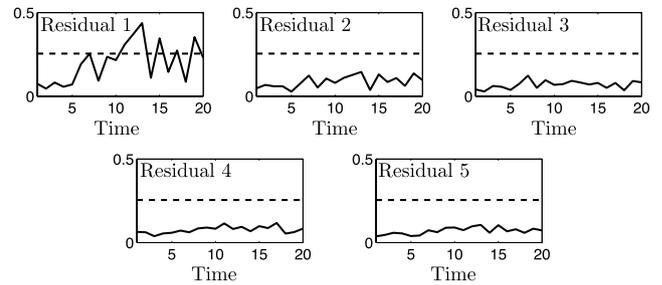


Fig. 6. Fig. 6 shows the residual functions computed by the 5 control centers. Since the first residual is greater than the threshold value, the presence of false data is correctly detected by the first control center. A form of regional identification is possible by simply identifying the residuals above the security threshold.

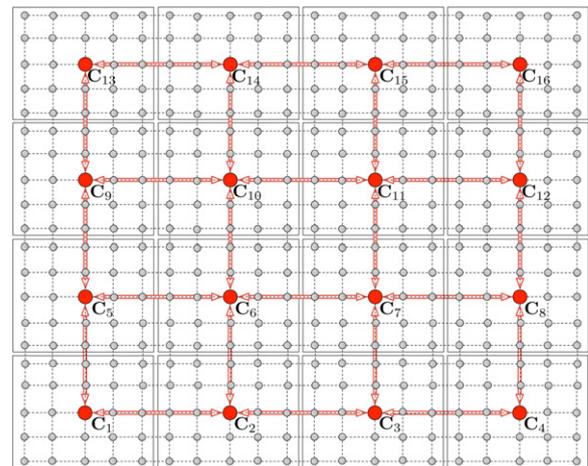


Fig. 7. In Fig. 7, a two dimensional power grid with 400 buses. The network is operated by 16 control centers, each one responsible for a different subnetwork. Control centers cooperate through the red communication graph. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

5.2. Scalability property of our finite-memory estimation technique

Consider an electrical network with $(ab)^2$ buses, where $a, b \in \mathbb{N}$. Let the buses interconnection structure be a two dimensional lattice, and let G be the graph whose vertices are the $(ab)^2$ buses, and whose edges are the network branches. Let G be partitioned into b^2 identical blocks containing a^2 vertices each, and assume the presence of b^2 control centers, each one responsible for a different network part. We assume the control centers to be interconnected through an undirected graph. In particular, being V_i the set of buses assigned to the control center C_i , we let the control centers C_i and C_j be connected if there exists a network branch linking a bus in V_i to a bus in V_j . An example with $b = 4$ and $a = 5$ is in Fig. 7. In order to show the effectiveness of our approximation procedure, suppose that each control center C_i aims at estimating the vector of the voltage angles at the buses in its region. We assume also that the control centers cooperate, and that each of them receives the measurements of the real power injected at only the buses in its region. Algorithm 2 is implemented by the control centers to solve the estimation problem. In Fig. 8 we report the estimation error during the iterations of the algorithm. Notice that, as predicted by Theorem 4.1, each leader has a good estimate of the state of its region before the termination of the algorithm.

6. Conclusion

Two distributed algorithms for network control centers to compute the minimum variance estimate of the network state

¹⁰ For a Gaussian distribution with mean μ and variance σ^2 , about 95% of the realizations are contained in $[\mu - 2\sigma, \mu + 2\sigma]$.

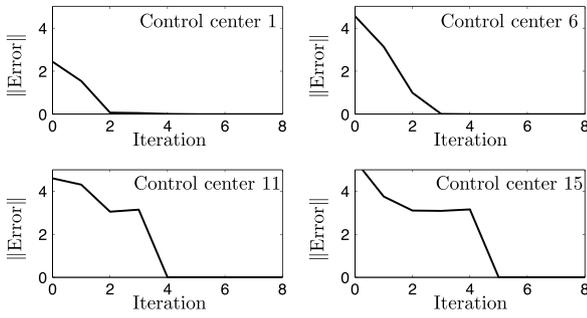


Fig. 8. Fig. 8 shows the norm of the estimation error of the local subnetwork as a function of the number of iterations of Algorithm 2. The considered monitors are C_1, C_6, C_{11} , and C_{15} . As predicted by Theorem 4.1, the local estimation error becomes negligible before the termination of the algorithm.

given noisy measurements have been proposed. The two methods differ in the mode of cooperation of the control centers: the first method implements an incremental mode of cooperation, while the second uses a diffusive interaction. Both methods converge in finite time, which we characterize, and they require only local measurements and model knowledge to be implemented. Additionally, an asynchronous and scalable implementation of our diffusive estimation method has been described, and its efficiency has been shown through a rigorous analysis and through a practical example. Based on these estimation methods, an algorithm to detect cyber-attacks against the network measurements has also been developed, and its detection performance has been characterized.

Appendix

A.1. Proof of Theorem 3.1

Proof. Let $H^i = [H_1^T \dots H_i^T]^T$, $z^i = [z_1^T \dots z_i^T]^T$. We show by induction that $z^i = H^i \hat{x}_i$, $K_i = \text{Basis}(\text{Ker}(H^i))$, and $\hat{x}_i \perp \text{Ker}(H^i)$. Note that the statements are trivially verified for $i = 1$. Suppose that they are verified up to i , then we need to show that $K_{i+1} = \text{Basis}(\text{Ker}(H^{i+1}))$, $\hat{x}_{i+1} \perp \text{Ker}(H^{i+1})$, and $z^{i+1} = H^{i+1} \hat{x}_{i+1}$.

We start by proving that $K_{i+1} = \text{Basis}(\text{Ker}(H^{i+1}))$. Observe that $\text{Ker}(K_i) = 0$ for all i , and that

$$\text{Ker}(H_{i+1}K_i) = \{v: K_iv \in \text{Ker}(H_{i+1})\}. \quad (\text{A.1})$$

Hence,

$$\begin{aligned} \text{Im}(K_{i+1}) &= \text{Im}(K_i \text{Ker}(H_{i+1}K_i)) \\ &= \text{Im}(K_i) \cap \text{Ker}(H_{i+1}) \\ &= \text{Ker}(H^i) \cap \text{Ker}(H_{i+1}) = \text{Ker}(H^{i+1}). \end{aligned}$$

We now show that $x_{i+1} \perp \text{Ker}(H^{i+1})$, which is equivalent to

$$(\hat{x}_i + K_i(H_{i+1}K_i)^\dagger(z_{i+1} - H_{i+1}\hat{x}_i)) \in \text{Ker}(H^{i+1})^\perp.$$

Note that

$$\text{Ker}(H^{i+1}) \subseteq \text{Ker}(H^i) \Leftrightarrow \text{Ker}(H^{i+1})^\perp \supseteq \text{Ker}(H^i)^\perp.$$

By the induction hypothesis we have $\hat{x}_i \in \text{Ker}(H^i)^\perp$, and hence $\hat{x}_i \in \text{Ker}(H^{i+1})^\perp$. Therefore, we need to show that

$$K_i(H_{i+1}K_i)^\dagger(z_{i+1} - H_{i+1}\hat{x}_i) \in \text{Ker}(H^{i+1})^\perp.$$

Let $w = (H_{i+1}K_i)^\dagger(z_{i+1} - H_{i+1}\hat{x}_i)$, and notice that $w \in \text{Ker}(H_{i+1}K_i)^\perp$ due to the properties of the pseudoinverse operation. Suppose that $K_iw \notin \text{Ker}(H_{i+1})^\perp$. Since $\text{Ker}(K_i) = \{0\}$, the vector w can be written as $w = w_1 + w_2$, where $K_iw_1 \in \text{Ker}(H_{i+1})^\perp$ and $K_iw_2 = K_iw - K_iw_1 \neq 0$, $K_iw_2 \in \text{Ker}(H_{i+1})$. Then, it holds $H_{i+1}K_iw_2 = 0$,

and hence $w_2 \in \text{Ker}(H_{i+1}K_i)$, which contradicts the hypothesis $w \in \text{Ker}(H_{i+1}K_i)^\perp$. Finally $K_iw \in \text{Ker}(H_{i+1})^\perp \subseteq \text{Ker}(H^{i+1})^\perp$.

We now show that $z^{i+1} = H^{i+1}\hat{x}_{i+1}$. Because of the consistency of the system of linear equations, and because $z^i = H^i\hat{x}_i$ by the induction hypothesis, there exists a vector $v_i \in \text{Ker}(H^i) = \text{Im}(K_i)$ such that $z^{i+1} = H^{i+1}(\hat{x}_i + v_i)$, and hence that $z_{i+1} = H_{i+1}(\hat{x}_i + v_i)$. We conclude that $(z_{i+1} - H_{i+1}\hat{x}_i) \in \text{Im}(H_{i+1}K_i)$, and finally that $z^{i+1} = H^{i+1}\hat{x}_{i+1}$. \square

A.2. Proof of Theorem 3.2

Before proceeding with the proof of the above theorem, we recall the following fact in linear algebra.

Lemma A.1. Let $H \in \mathbb{R}^{n \times m}$. Then $\text{Ker}((H^\dagger)^\top) = \text{Ker}(H)$.

Proof. We first show that $\text{Ker}((H^\dagger)^\top) \subseteq \text{Ker}(H)$. Recall from Bernstein (2009) that $H = HH^\top(H^\dagger)^\top$. Let x be such that $(H^\dagger)^\top x = 0$, then $Hx = HH^\top(H^\dagger)^\top x = 0$, so that $\text{Ker}((H^\dagger)^\top) \subseteq \text{Ker}(H)$. We now show that $\text{Ker}(H) \subseteq \text{Ker}((H^\dagger)^\top)$. Recall that $(H^\dagger)^\top = (H^\top)^\dagger = (HH^\top)^\dagger H$. Let x be such that $Hx = 0$, then $(H^\dagger)^\top x = (HH^\top)^\dagger Hx = 0$, so that $\text{Ker}(H) \subseteq \text{Ker}((H^\dagger)^\top)$, which concludes the proof. \square

We are now ready to prove Theorem 3.2.

Proof. The first property follows directly from Bernstein (2009) (cfr. p. 427). To show the second property, observe that $C^\dagger = \frac{1}{\varepsilon}((I - HH^\dagger)B)^\dagger$, so that

$$\lim_{\varepsilon \rightarrow 0^+} \varepsilon D = 0.$$

For the theorem to hold, we need to verify that

$$H^\dagger - H^\dagger B((I - HH^\dagger)B)^\dagger = (H^\top \Sigma^{-1}H)^{-1}H^\top \Sigma^{-1},$$

or, equivalently, that

$$\begin{aligned} (H^\dagger - H^\dagger B((I - HH^\dagger)B)^\dagger)HH^\dagger \\ = (H^\top \Sigma^{-1}H)^{-1}H^\top \Sigma^{-1}HH^\dagger, \end{aligned} \quad (\text{A.2})$$

and

$$\begin{aligned} (H^\dagger - H^\dagger B((I - HH^\dagger)B)^\dagger)(I - HH^\dagger) \\ = (H^\top \Sigma^{-1}H)^{-1}H^\top \Sigma^{-1}(I - HH^\dagger). \end{aligned} \quad (\text{A.3})$$

Consider Eq. (A.2). After simple manipulation, we have

$$H^\dagger - H^\dagger B((I - HH^\dagger)B)^\dagger HH^\dagger = H^\dagger,$$

so that we need to show only that

$$H^\dagger B((I - HH^\dagger)B)^\dagger HH^\dagger = 0.$$

Recall that for a matrix W it holds $W^\dagger = (W^\top W)^\dagger W^\top$. Then the term $((I - HH^\dagger)B)^\dagger HH^\dagger$ equals

$$(((I - HH^\dagger)B)^\top ((I - HH^\dagger)B))^\dagger B^\top (I - HH^\dagger)HH^\dagger = 0,$$

because $(I - HH^\dagger)HH^\dagger = 0$. We conclude that Eq. (A.2) holds. Consider now Eq. (A.3). Observe that $HH^\dagger(I - HH^\dagger) = 0$. Because B has full row rank, and $\Sigma = BB^\top$, simple manipulation yields

$$\begin{aligned} -H^\top (BB^\top)^{-1}HH^\dagger B[(I - HH^\dagger)B]^\dagger (I - HH^\dagger)B \\ = H^\top (BB^\top)^{-1}(I - HH^\dagger)B, \end{aligned}$$

and hence

$$H^\top (BB^\top)^{-1} \left\{ I + HH^\dagger B[(I - HH^\dagger)B]^\dagger \right\} (I - HH^\dagger)B = 0.$$

Since $HH^\dagger = I - (I - HH^\dagger)$, we obtain

$$H^\top (BB^\top)^{-1}B[(I - HH^\dagger)B]^\dagger (I - HH^\dagger)B = 0.$$

A sufficient condition for the above equation to be true is

$$\left([(I - HH^\dagger)B]^\dagger \right)^\top B^\top (BB^\top)^{-1} H = 0.$$

From Lemma A.1 we have.

$$\text{Ker} \left(\left([(I - AA^\dagger)B]^\dagger \right)^\top \right) = \text{Ker}((I - AA^\dagger)B).$$

Since

$$(I - HH^\dagger)BB^\top (BB^\top)^{-1} H = (I - HH^\dagger)H = 0,$$

we have that

$$H^\top (BB^\top)^{-1} B [(I - HH^\dagger)B]^\dagger (I - HH^\dagger)B = 0,$$

and that Eq. (A.3) holds. This concludes the proof. \square

References

- Abur, A., & Exposito, A. G. (2004). *Power system state estimation: theory and implementation*. CRC Press.
- Bernstein, D. S. (2009). *Matrix mathematics* (2nd ed.). Princeton University Press.
- Carli, R., Chiuso, A., Schenato, L., & Zampieri, S. (2008). Distributed Kalman filtering based on consensus strategies. *IEEE Journal on Selected Areas in Communications*, 26(4), 622–633.
- Cattivelli, F. S., Lopes, C. G., & Sayed, A. H. (2008). Diffusion recursive least-squares for distributed estimation over adaptive networks. *IEEE Transactions on Signal Processing*, 56(5), 1865–1877.
- Cattivelli, F. S., & Sayed, A. H. (2010). Diffusion strategies for distributed Kalman filtering and smoothing. *IEEE Transactions on Automatic Control*, 55(9), 2069–2084.
- Censor, Y. (1981). Row-action methods for huge and sparse systems and their applications. *SIAM Review*, 23(4), 444–466.
- Demco, S., Moss, W. F., & Smith, P. W. (1984). Decay rates for inverses of band matrices. *Mathematics of Computation*, 43(168), 491–499.
- Falcao, D. M., Wu, F. F., & Murphy, L. (1995). Parallel and distributed state estimation. *IEEE Transactions on Power Systems*, 10(2), 724–730.
- Godsil, C. D., & Royle, G. F. (2001). *Graduate texts in mathematics: Vol. 207. Algebraic graph theory*. Springer.
- Golub, G. H., & van Loan, C. F. (1989). *Matrix computations* (2nd ed.). Johns Hopkins University Press.
- Gordon, R., Bender, R., & Herman, G. T. (1970). Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography. *Journal of Theoretical Biology*, 29(3), 471–481.
- Jiang, W., Vittal, V., & Heydt, G. T. (2007). A distributed state estimator utilizing synchronized phasor measurements. *IEEE Transactions on Power Systems*, 22(2), 563–571.
- Kaczmarz, S. (1937). Angenäherte Auflösung von Systemen linearer Gleichungen. *Bulletin de l'Academie Polonaise des Sciences Letters A*, 35, 355–357.
- Liu, Y., Reiter, M. K., & Ning, P. (2009). False data injection attacks against state estimation in electric power grids. In *ACM conference on computer and communications security* (pp. 21–32). USA: Chicago, IL.
- Lopes, C. G., & Sayed, A. H. (2007). Incremental adaptive strategies over distributed networks. *IEEE Transactions on Signal Processing*, 55(8), 4064–4077.
- Lopes, C. G., & Sayed, A. H. (2008). Diffusion least-mean squares over adaptive networks: formulation and performance analysis. *IEEE Transactions on Signal Processing*, 56(7), 3122–3136.
- Luenberger, D. G. (1969). *Optimization by vector space methods*. Wiley.
- Meserve, J. (2007). Sources: staged cyber attack reveals vulnerability in power grid. September 26. <http://cnn.com>.
- Monticelli, A. (1999). *State estimation in electric power systems: a generalized approach*. Springer.
- NERC, (2004). Final Report on the August 14, 2003 Blackout in the United States and Canada: causes and recommendations. Available at <http://www.nerc.com/filez/blackout.html>.
- Pasqualetti, F., Carli, R., Bicchì, A., & Bullo, F. (2010). Distributed estimation and detection under local information. In *IFAC workshop on distributed estimation and control in networked systems* (pp. 263–268). France: Anney.
- Rabbat, M. G., & Nowak, R. D. (2005). Quantized incremental algorithms for distributed optimization. *IEEE Journal on Selected Areas in Communications*, 23(4), 798–808.
- Rakpenthai, C., Premrudeepreechacharn, S., Uatrongjit, S., & Watson, N. R. (2005). Measurement placement for power system state estimation using decomposition technique. *Electric Power Systems Research*, 75(1), 41–49.
- Sayed, A. H., & Lopes, C. G. (2007). Adaptive processing over distributed networks. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E90-A(8), 1504–1510.
- Schizas, I., Ribeiro, A., & Giannakis, G. (2007). Consensus in ad hoc WSNs with noisy links—Part I: distributed estimation of deterministic signals. *IEEE Transactions on Signal Processing*, 56(1), 350–364.
- Schizas, I. D., Mateos, G., & Giannakis, G. B. (2009). Distributed LMS for consensus-based in-network adaptive processing. *IEEE Transactions on Signal Processing*, 57(6), 2365–2382.
- Schweppe, F. C., & Wildes, J. (1970a). Power system static-state estimation, Part I: exact model. *IEEE Transactions on Power Apparatus and Systems*, 89(1), 120–125.
- Schweppe, F. C., & Wildes, J. (1970b). Power system static-state estimation, Part II: approximate model. *IEEE Transactions on Power Apparatus and Systems*, 89(1), 125–130.
- Schweppe, F. C., & Wildes, J. (1970c). Power system static-state estimation, Part III: implementation. *IEEE Transactions on Power Apparatus and Systems*, 89(1), 130–135.
- Shahidehpour, M., & Wang, Y. (2003). *Communication and control in electric power systems: applications of parallel and distributed processing*. Wiley-IEEE Press.
- Stankovic, S. S., Stankovic, M. S., & Stipanovic, D. M. (2009). Consensus based overlapping decentralized estimation with missing observations and communication faults. *Automatica*, 45(6), 1397–1406.
- Tanabe, K. (1971). Projection method for solving a singular system of linear equations and its applications. *Numerische Mathematik*, 17(3), 203–214.
- Xie, L., Mo, Y., & Sinopoli, B. (2010). False data injection attacks in electricity markets. In *IEEE international conference on smart grid communications*. (pp. 226–231). Gaithersburg, MD, USA.
- Zhao, L., & Abur, A. (2005). Multi area state estimation using synchronized phasor measurements. *IEEE Transactions on Power Systems*, 20(2), 611–617.



Fabio Pasqualetti (S'07) is a Doctoral Candidate in Mechanical Engineering at the University of California, Santa Barbara. He received the Laurea degree “summa cum laude” in Computer Engineering from the University of Pisa, Pisa, Italy, in 2004, and the Laurea Magistrale degree “summa cum laude” in Automation Engineering from the University of Pisa, Pisa, Italy, in 2007.



His main research interest is in secure control systems, with application to multi-agent networks, distributed computing and power networks. Other interests include vehicle routing and combinatorial optimization, with application to distributed area patrolling and persistent surveillance.

Ruggero Carli received the Laurea Degree in 2004 and the Ph.D. degree in 2007 in Information engineering from the University of Padova, Padova, Italy. From 2008 through 2010 he was a Post Doctoral Fellow with the Mechanical Engineering Department at the University of California in Santa Barbara. He is currently an Assistant Professor with the Department of Information Engineering, University of Padova. His research interests include control theory and, in particular, control under communication constraints, cooperative control and distributed estimation.



Francesco Bullo (S'95-M'99-SM'03-F'10) received the Laurea degree summa cum laude in electrical engineering from the University of Padova, Padova, Italy, in 1994 and the Ph.D. degree in control and dynamical systems from the California Institute of Technology, Pasadena, in 1999. From 1998 to 2004, he was an Assistant Professor with the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana. He is currently a Professor with the Mechanical Engineering Department and the Center for Control, Dynamical Systems and Computation, University of California in Santa Barbara. His main research interests are multiagent networks with application to robotic coordination, distributed computing and power networks; he has worked on problems in vehicle routing, geometric control, and motion planning. He is the coauthor of *Geometric Control of Mechanical Systems* (Springer, 2004, ISBN: 0-387- 22195-6) and *Distributed Control of Robotic Networks* (Princeton, 2009, ISBN: 978-0-691-14195-4). He has published more than 200 papers in international journals, books, and refereed conferences. Dr. Bullo is a recipient of the Office of Naval Research (ONR) Young Investigator Award, the Outstanding Paper Award for the IEEE Control Systems Magazine, and the AACC Hugo Schuck Best Paper Award. His students papers were finalists for the Best Student Paper Award at the IEEE Conference on Decision and Control (2002, 2005, 2007) and the American Control Conference (2005, 2006, 2010).