A distributed method for state estimation and false data detection in power networks

Fabio Pasqualetti, Ruggero Carli, and Francesco Bullo

***Abstract*—This work presents a distributed method for control centers in a power network to estimate the operating condition of the power plant, and to ultimately determine the occurrence of threatening situations. Specifically 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 voltages 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 assumed, whereas, in the second method, a *diffusive* interaction is implemented. These estimation strategies are provably convergent in a finite number of steps to the weighted least-squares solution and are exploited to develop a distributed algorithm to detect corrupted data among the network measurements.**

I. Introduction

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 in time, the network model and the phasor voltages at every system bus need 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 measurement data is used for control and optimization functions, such as contingency analysis, automatic generation control, load forecasting, optimal power flow computation, and reactive power dispatch [1]. Various sources of uncertainties, e.g., measurement and communication noise, lead to inaccuracies in the received data, which may affect the performance of the control and optimization algorithms, and, ultimately, the stability of the power plant. This concern was first recognized and addressed in [2] 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 systems state estimation problem, several centralized and parallel solutions have been developed in the last decades, e.g., see [3]. Being an online function, computational issues, storage requirements, and numerical robustness of the solution algorithm need to be taken into account. Within this regard, distributed algorithms based on network partitioning techniques are to be preferred over centralized ones. On the other hand, as outlined in [4] concerning the blackout of August 2003, an estimation of the entire network is essential to prevent networks damages, so that, even in a decentralized setting, the whole state vector should be preferably estimated by every unit. In [5], [6] the idea is explored of using a global control center to coordinate estimates obtained locally by several local control centers. In this work, we improve the results along these directions, by proposing a fully decentralized and distributed estimation algorithm, which, by only assuming local knowledge of the network structure by the local control centers, allows them to obtain in finite-time an optimal estimate of the network state. Our procedure appears scalable against the power network dimension, and, furthermore, numerically reliable and accurate.

Because of the increasing reliance of nowadays power systems on communication networks, the possibility of cyber-attacks to networked systems is an actual threat [7]. 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 [8]. This important type of attack is often referred in the power systems literature to as *false data injection attack*. 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, and 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 the above

This material is based in part upon work supported by ICB ARO grant W911NF-09-D-0001 and NSF CNS-0834446, and in part upon the EC Contract IST 224428 "CHAT".

Fabio Pasqualetti and Francesco Bullo are with the Center for Control, Dynamical Systems and Computation, University of California, Santa Barbara {fabiopas, bullo}@engineering.ucsb.edu

Ruggero Carli is with the Department of Information Engineering, University of Padova carlirug@dei.unipd.it

dimensionality problem by distributing the false data detection problem among several control centers.

The contributions of this work are twofold. First, we adopt the static state network estimation model, in which the state vector is linearly related to the network measurements. We develop a method for a group of connected control centers to compute an optimal estimate of the system state via distributed computation. Our estimation method is iterative, and it is shown to converge in a finite number of steps. Furthermore, it requires only local information for its implementation, because each control center uses only the measurements coming from a local neighborhood of the whole network. Second, we describe a finite-time algorithm to detect corrupted data in the network measurements via distributed computation. Our detection method is based upon our state estimation algorithm, and it inherits its convergence properties. Additionally, we characterize a class of corrupted vectors that are guaranteed to be detected by our procedure. Finally, we show the effectiveness of our procedures on the IEEE 118 bus system.

The rest of the paper is organized as follows. In Section II we introduce the problem under consideration, and we describe the mathematical setup. Section III contains our main results on the state estimation and on the detection problem, as well as our algorithmic procedures. In Section IV we study the IEEE 118 bus system, and we present some simulation results. Finally, Section V contains our conclusion.

II. Problem definition and preliminary notions

The state of a power network 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 [2] refers to the procedure of estimating the state of a power network given a set of measurements of the network variables, such as, for instance, 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 vector. Then we have

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

where $h(\cdot)$ is a nonlinear measurement function, and η is a zero mean random vector satisfying $\mathbb{E}[\eta\eta^{\mathsf{T}}] = \Sigma_\eta = \Sigma_\eta^{\mathsf{T}} > 0$. An optimal estimate of the network state coincides with the *most likely* vector $\hat{x}$ that solves equation (1). It should be observed that the network state could be obtained by measuring directly the voltage phasors by means of phasor measurement devices.[1] Such an approach, however, would be economically expensive, and it would be very vulnerable to communication failures [1]. In this work, we adopt the approximated estimation model presented in [9], which follows from the linearization around the origin of equation (1). In particular, letting $H \in \mathbb{R}^{p\times n}$, $\mathbb{E}[v] = 0$, and $E[vv^{\mathsf{T}}] = \Sigma = \Sigma^{\mathsf{T}} > 0$, the state estimation problem becomes

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

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

Observe that, because of the interconnection structure of a power network, the measurement matrix H is sparse. Let $\mathrm{Ker}(H)$ denote the null space defined by the matrix H. For the equation (2), without affecting generality, assume $\mathrm{Ker}(H) = \{0\}$, and recall from [10] that the vector

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

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

The centralized computation of the minimum variance estimate to (2) assumes the complete knowledge of the matrices H and Σ, and it requires the inversion of the matrix $H^{\mathsf{T}}\Sigma^{-1}H$. 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. Because the performance of the network control and optimization algorithms depend upon the precision of the state estimate, a limitation on the 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 as

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

where, for $i \in \{1, \dots, m\}$, $m_i \in \mathbb{N}$, $H_i \in \mathbb{R}^{m_i\times n}$, $\Sigma_i \in \mathbb{R}^{m_i\times n}$, $z_i \in \mathbb{R}^{m_i}$, and $\sum_{i=1}^m m_i = p$. Let $G = (V, \mathcal{E})$ be a connected graph in which each vertex $i \in V = \{1, \dots, m\}$ denotes a monitor. Assume that each 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.*

Given the distributed nature of a power system and the increasing reliance on local area networks to transmit data to a control center, there exists the possibility for an attacker to compromise the network functionality 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, hence, it is unknown and unmeasurable by any of the monitoring

stations. We refer to the vector w to 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 bad data vector is intentionally chosen such that $w \in \mathrm{Im}(H)$, then the attack cannot be detected through the measurements z, because, denoting with † the pseudoinverse operation, the vector $x + H^\dagger w$ is a valid network state. In this work, 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.*

III. 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.

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, precisely, the *incremental* and the *diffusive* interaction. 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 [11], induces a cyclic interaction graph among the processors. In an diffusive strategy, instead, each node exchange 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 communication round. This section is divided into three parts. In Section III-A, we develop a distributed incremental method to solve a minimum variance estimation problem. In Section III-B we derive a diffusive strategy which is amenable to asynchronous implementation. Finally, in Section III-C we propose a distributed algorithm for the detection of false data among the measurements.

A. Incremental state estimation via distributed computation

We start by introducing a distributed incremental procedure to compute the minimum norm solution to a set of liner equations. This procedure constitutes the key ingredient of the incremental method we later propose to solve the minimum variance estimation problem.

Let $H \in \mathbb{R}^{p\times m}$, and let $z \in \mathrm{Im}(H)$, where $\mathrm{Im}(H)$ 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 $\mathrm{Ker}(H)$ of H [10]. 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 $\mathrm{Basis}(\mathcal{V})$ to denote any full rank matrix whose columns span the subspace $\mathcal{V}$.

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

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

The next theorem states the convergence properties of the *Incremental minimum norm solution* algorithm.

Theorem III.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 $\hat{x} \perp \mathrm{Ker}(H)$.*[2]

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 $\mathbb{E}(vv^\mathsf{T}) = \Sigma = \Sigma^\mathsf{T} > 0$. Consider the system of equations

$$z = Hx + v, \tag{5}$$

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

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

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

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

[2] A proof of the results contained in this paper can be found at `http://arxiv.org/abs/1103.0579`.

[3] 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 estimation x_{wls} and $\hat{x}(\varepsilon)$.

Theorem III.2 (Convergence with ε) *Consider the system of linear equations $z = Hx + v$. Let $\mathbb{E}(v) = 0$ and $\mathbb{E}(vv^{\mathsf{T}}) = \Sigma = BB^{\mathsf{T}} > 0$, for a full row rank matrix B. Let*

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

Then

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

where $x(\varepsilon) = \left(H^{\dagger} - \varepsilon H^{\dagger}B(C^{\dagger} + D)\right)z$. Moreover, for a fixed value of ε, we have

$$x_{\text{wls}} - \hat{x}(\varepsilon) = \varepsilon H^{\dagger}BDz.$$

For the system of equations (5), the estimate $\hat{x}(\varepsilon)$ can be computed by means of Algorithm 1 with input $[H_1\ \varepsilon B_i], \ldots, [H_m\ \varepsilon B_m]$ and $z_1, \ldots, z_m$, where, being BB^{T} is the noise covariance matrix, we have

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

and, for $i \in \{1, \ldots, m\}$, $m_i \in \mathbb{N}$, $H_i \in \mathbb{R}^{m_i \times n}$, $B_i \in \mathbb{R}^{m_i \times n}$, $z_i \in \mathbb{R}^{m_i}$, and $\varepsilon > 0$.

To conclude this section, we characterize the estimation residual $z - H\hat{x}$. This quantity will play a important role for the synthesis of a distributed bad data detection algorithm.

Corollary III.1 (Estimation residual) *Consider the system $z = Hx + v$, and let $\mathbb{E}[vv^{\mathsf{T}}] = \Sigma = \Sigma^{\mathsf{T}} > 0$. Then*[4]

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

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

B. Diffusive state estimation via distributed computation

The implementation of the incremental state estimation algorithm described in Section III-A 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.[5] To be more precise, let $V = \{1, \ldots, m\}$ 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 equation (8). During the h-th iteration of the algorithm, monitor i, with $i \in \{1, \ldots, N\}$, 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 next show the convergence of Algorithm 2 to the minimum variance estimate.

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

Input : H_i, εB_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;
    $\hat{x}_i := \hat{x}_i + [K_i\ \ 0][-K_i\ \ K_j]^{\dagger}(\hat{x}_i - \hat{x}_j)$;
    $K_i := \text{Basis}(\text{Im}(K_i) \cap \text{Im}(K_j))$;
  transmit $\hat{x}_i$ and K_i;

Theorem III.3 (Convergence of Algorithm 2) *Consider the system of linear equations $z = Hx + v$, where $\mathbb{E}[v] = 0$ and $\mathbb{E}[vv^{\mathsf{T}}] = BB^{\mathsf{T}}$. 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 at most d steps.*

As a consequence of Theorem III.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 [13], [14] and the references therein for a thorough discussion.

C. Detection of false data via distributed computation

In the previous sections we have shown how to compute an optimal estimate of the system state via distributed computation. We focus now on the problem of detecting the presence of false data among the measurements in a distributed manner. When the measurements are corrupted, the state to measurements relation becomes

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

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

[5]An undirected graph is said to be connected if there exists a path between any two vertices [12].

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

Input : H_i, εB_i, Γ;

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**
  False data detected;

where w is the false data vector. Following Corollary III.1, the vector w is detectable if it affects significantly the estimation residual, i.e., if

$$\lim_{\varepsilon \to 0} \|z - H\hat{x}(\varepsilon)\| > \Gamma,$$

where threshold Γ depends upon the magnitude of the noise v. Notice that, because the 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 $z(t)$ be the measurements vector at a given time instant t, and let $\mathbb{E}[z(t_1)z^{\mathsf{T}}(t_2)] = 0$ for all $t_1 \neq t_2$. Based on this considerations, our distributed detection procedure is in Algorithm 3, where the matrices H_i and εB_i are as defined in equation (8), and Γ is a predefined threshold.

In Algorithm 3, the value of the threshold Γ determines the false alarm and the misdetection rate. Clearly, if $\Gamma \geq \|(I - HW)\|\|v(t)\|$ and ε is sufficiently small, then no false alarm is triggered, at the expenses of the misdetection 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 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, once the estimation $\hat{x}(t)$ has been computed, the condition $\|z(t) - H\hat{x}(t)\|_\infty > \Gamma$ can be checked by each leader without any further communication. A related example is presented in the next section.

IV. An illustrative example

The IEEE 118 bus system represents a portion of the American Electric Power System as of December, 1962. This test case system is composed of 118 buses, 186 branches, 54 generators, and 99 loads. Following [9], the voltage angles and the power injections at the network buses are 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. Let $z = P_{\text{bus}} - v$ be the measurements vector, where $\mathbb{E}[v] = 0$ and $\mathbb{E}[vv^{\mathsf{T}}] = \sigma^2 I$, $\sigma \in \mathbb{R}$. Then, following the notation in Theorem III.2, the minimum variance estimate of θ_{bus} can be recovered as

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

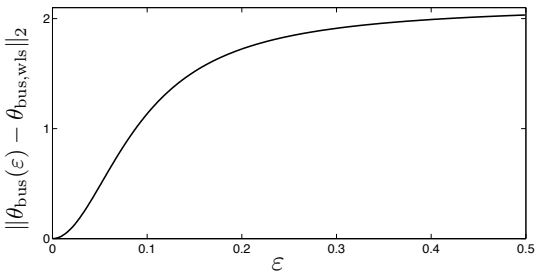

Fig. 1. The normalized Euclidean norm of the error vector $\theta_{\text{bus}}(\varepsilon) - \theta_{\text{bus,wls}}$ is plotted as a function of the parameter ε, where $\theta_{\text{bus}}(\varepsilon)$ is the estimation vector computed according to Theorem III.2, and $\theta_{\text{bus,wls}}$ is the minimum variance estimate of θ_{bus}. As ε decreases, the vector $\theta_{\text{bus}}(\varepsilon)$ converges to the minimum variance estimate $\theta_{\text{bus,wls}}$.

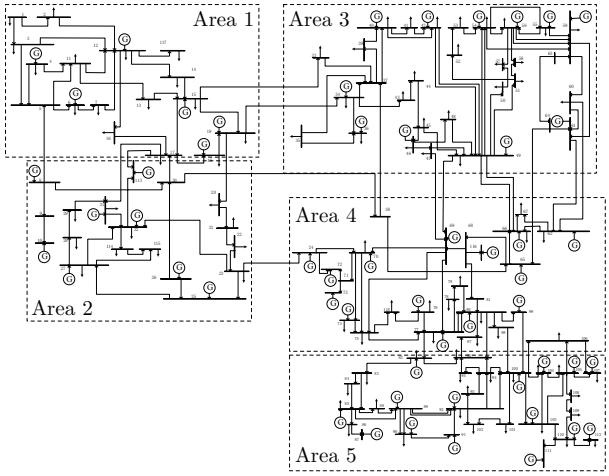

Fig. 2. The IEEE 118 bus system has been divided into 5 areas. Each area is monitored and operated by a control center. The control centers cooperate to estimate the state and to assess the functionality of the whole network.

In Fig. 1 we show that, as ε decreases, the estimation vector computed according to Theorem III.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 IEEE 118 network, each one responsible for a subpart of the entire network. The situation is depicted in Fig. 2. 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 $\mathbb{E}[v_i v_i^{\mathsf{T}}] = \sigma_i^2 I$, be 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 III, the control center can compute an optimal estimate of θ_{bus} by means of Algorithm 1 or 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 computation burden and memory requirement. Indeed, the i-th monitor uses n_i measurements instead of N. Let $\bar{N}$ be the maximum

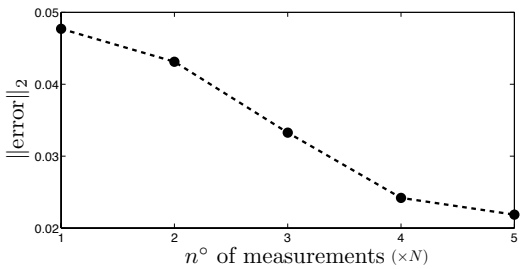

Fig. 3. For a fixed value of ε, the plot shows the norm of the error with respect to the true state vector of the estimate obtained by means of Algorithm 1 (solid) and of the centralized minimum variance estimation with $\bar{N}$ measurements (dashed). The x axis denotes the number of measurements used for the distributed estimation. Because of the presence of several control centers, the distributed algorithm is more accurate while maintaining the same (or smaller) computational complexity of the centralized estimation.

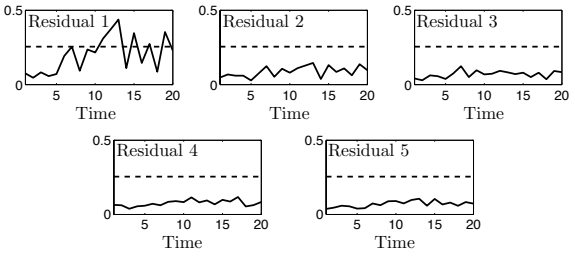

Fig. 4. Residual functions computed by the 5 control centers. Since the first residual is greater than the threshold value, the presence of false data is correctly detected by the first control center. A form of regional identification is possible by simple identifying the residuals above the security threshold.

number of measurements that, due to hardware or numerical constraints, a control center can efficiently handle for the state estimation problem. In Fig. 3, 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 [1]. 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 $\bar{z}_i = z_i + e_1 w_i$, where e_1 is the first canonical vector, and w_i is a random variable. For the simulation we choose w_i to be uniformly distributed in the interval $[0, w_{\max}]$, 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 $2\sigma\|I - H(H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1}\|_\infty$.[6] The residual functions $\|z_i - H\hat{x}\|_\infty$ are reported in Fig. 4. 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 $\{2, \ldots, 5\}$ are below the threshold value, the corrupted data is likely to be among the measurements of the first area. This important aspect is left, however, as the subject of future research.

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

V. Conclusion

Two algorithms have been presented to solve the static state estimation problem in a power network via distributed computation. Our first algorithm is incremental, and it converges in a number of steps equal to the number of units involved in the computation. Our second algorithm is diffuse: it requires a larger number of operations compared to our first algorithm, but (i) it allows for an asynchronous update of the estimate by the computing units, and (ii) it does not impose any particular interconnection structure among the computing units. Based on these estimation procedures, a distributed algorithm to detect false data among the network measurements is also proposed.

References

[1] A. Abur and A. G. Exposito, *Power System State Estimation: Theory and Implementation*. CRC Press, 2004.

[2] F. C. Schweppe and J. Wildes, "Power system static-state estimation, Part I: Exact model," *IEEE Transactions on Power Apparatus and Systems*, vol. 89, no. 1, pp. 120–125, 1970.

[3] A. Monticelli, *State Estimation in Electric Power Systems: A Generalized Approach*. Springer, 1999.

[4] NERC, "Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendations," Apr. 2004, Available at http://www.nerc.com/filez/blackout.html.

[5] L. Zhao and A. Abur, "Multi area state estimation using synchronized phasor measurements," *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 611–617, 2005.

[6] W. Jiang, V. Vittal, and G. T. Heydt, "A distributed state estimator utilizing synchronized phasor measurements," *IEEE Transactions on Power Systems*, vol. 22, no. 2, pp. 563–571, 2007.

[7] J. Meserve, "Sources: Staged cyber attack reveals vulnerability in power grid," http://cnn.com, September 26, 2007.

[8] Y. Liu, M. K. Reiter, and P. Ning, "False data injection attacks against state estimation in electric power grids," in *ACM Conference on Computer and Communications Security*, Chicago, IL, USA, Nov. 2009, pp. 21–32.

[9] F. C. Schweppe and J. Wildes, "Power system static-state estimation, Part II: Approximate model," *IEEE Transactions on Power Apparatus and Systems*, vol. 89, no. 1, pp. 125–130, 1970.

[10] D. G. Luenberger, *Optimization by Vector Space Methods*. Wiley, 1969.

[11] M. G. Rabbat and R. D. Nowak, "Quantized incremental algorithms for distributed optimization," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 798–808, 2005.

[12] C. D. Godsil and G. F. Royle, *Algebraic Graph Theory*, ser. Graduate Texts in Mathematics. Springer, 2001, vol. 207.

[13] C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Transactions on Signal Processing*, vol. 55, no. 8, pp. 4064–4077, 2007.

[14] ——, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3122–3136, 2008.