2012 20th Mediterranean Conference on Control & Automation (MED)
Barcelona, Spain, July 3-6, 2012

Trust-Based Multi-Agent Filtering for Increased Smart Grid Security

Ion Matei, John S. Baras, Vijay Srinivasan

***Abstract*— We address the problem of state estimation of the power system for the Smart Grid. We assume that the monitoring of the electrical grid is done by a network of agents with both computing and communication capabilities. We propose a security mechanism aimed at protecting the state estimation process against false data injections originating from faulty equipment or cyber-attacks. Our approach is based on a multi-agent filtering scheme, where in addition to taking measurements, the agents are also computing local estimates based on their own measurements and on the estimates of the neighboring agents. We combine the multi-agent filtering scheme with a *trust-based mechanism* under which each agent associates a trust metric to each of its neighbors. These trust metrics are taken into account in the filtering scheme so that information transmitted from agents with low trust is disregarded. In addition, a mechanism for the trust metric update is also introduced, which ensures that agents that diverge considerably from their expected behavior have their trust values lowered.**

I. Introduction

Smart Grid refers to the modernization of the electric system through the integration of new information-age technologies and new strategic public policies. It is based on adding and integrating new digital computing and communication technologies and services with the power-delivery infrastructure. Some of the characteristics of the Smart Grid include an increased use of digital information and controls technology to improve reliability, security, and efficiency of the electric grid, dynamic optimization of grid operations and resources, with full cyber security [1]. It is widely recognized that one of the most challenging tasks in implementing the Smart Grid is putting in place security policies that address security threats to the infrastructure. The main goal of a cyber-security strategy is the prevention of damages to, unauthorized use of, exploitation of electronic information and communication systems and services, and to ensure confidentiality, integrity and availability [2].

The part of the Smart Grid infrastructure used to control the electricity generation and transmission is represented by the Supervisory Control and Data Acquisition (SCADA) systems. A SCADA system receives measurements of the state of the power grid and computes an estimate on the state of the power grid on which the control strategy is based. Power blackout events due to the failure of the SCADA systems to recognize load and stability restrictions, human errors, faults, etc. suggest the need for improved system-wide monitoring, alarms and power system state estimation programs [9].

In this paper we focus on the *security of the state estimation* of the power system. We propose a strategy aimed at improving the security of the SCADA systems, by implementing algorithmic policies aimed at protecting the estimation process against false data injection generated by faulty equipment or cyber attacks. Our approach is based on a *multi-agent filtering scheme*, where intelligent agents, geographically distributed in the power grid, receive (local) measurements on the state of the power grid and compute local estimates based on their own measurements and on the estimates of the neighboring agents as well. We combine the multi-agent filtering scheme with a *trust-based mechanism* under which each agent associates a trust metric to each of its neighbors. These trust metrics are taken into account in the filtering scheme so that information transmitted from agents with low trust is disregarded. In addition, a mechanism for the trust metric update is introduced, which is based on the long-term behavior characteristics of the agents. Other approaches for dealing with false data injections on SCADA systems can be found in [5], for example. Note that a similar problem setup was proposed in [11]. However, in the current paper we consider a different update mechanism for the trust metrics attributed to the agents.

II. Problem formulation

State estimation of power systems, using real-time measurements of active and reactive power flows in the network, is used to build the model for the observable part of the power grid. It was introduced to help the system operators having a good image on the state of the power system in order to increase the ability to tackle contingency conditions. Rather than centralizing the measurements to the SCADA systems for state estimate computations, we propose a strategy to decentralize the estimation process, under which a network of intelligent agents computes local estimates based on their measurements and the estimates computed by other agents.

Thanks to their simplicity, the most common models used to represent the dynamics of the power grid are linear [8], [10], [22]. Similarly, throughout this paper we consider the

Ion Matei is with the Engineering Laboratory at the National Institute of Standards and Technology, Gaithersburg, MD 20899 (ion.matei@nist.gov) and with the Institute for Research in Electronics and Applied Physics, University of Maryland, College Park, MD 20742 (imatei@umd.edu); John S. Baras is with the Institute for Systems Research at the University of Maryland, College Park (baras@umd.edu); Vijay Srinivasan is with the Engineering Laboratory at the National Institute of Standards and Technology, Gaithersburg, MD 20899 (vsrini@nist.gov).

This material is based in part upon work supported by the NIST-ARRA Measurement Science and Engineering Fellowship Program award 70NANB10H026, through the University of Maryland, by the U.S. Air Force of Scientific Research MURI grants FA9550-09-1-0538 and FA9550-10-1-0573, by the National Institute of Standards and Technology (NIST) grant award 70NANB11H148, and by the Defense Advanced Research Projects Agency (DARPA) and the SRC FCRP grant SA00007007.

978-1-4673-2531-8/12/$31.00 ©2012 IEEE

following stochastic linear equation as an approximate model for the power grid dynamics

$$x(k+1) = A(k)x(k) + w(k), \tag{1}$$

where k denotes the discrete time, $x(k) \in \mathbb{R}^n$ is the state vector, $A(k)$ is the matrix connecting the current state with the one at the next time instant, and $w(k) \in \mathbb{R}^n$ is the state noise, assumed Gaussian with zero mean and covariance matrix Q. The state variables usually include *nodal voltages* (voltage magnitudes, voltage angles), *transformer ratios*, and *complex power flows* (active and reactive power flows) [20]. The initial state x_0 has a Gaussian distribution, with mean μ_0 and covariance matrix P_0. The parameters of the model, $A(k)$ and Q, are assumed to be determined a priori through a parameter identification process [8]. Usually, the matrix $A(k)$ is assumed slow time-varying; and, for simplicity in this paper we assume $A(k)$ to be constant, i.e., $A(k) = A$ for all k.

In the context of this paper an agent is an intelligent device with computation and communication capabilities; it can receive measurements on the state of the power grid and it is able to compute state estimates. A candidate for playing the role of an agent is the Phasor Data Concentrator (PDC), which collects the measurements from a set of GPS synchronized Phasor Measurement Units (PMUs). Their ability to measure positive sequence voltages at network busses and positive sequence currents in transmission lines and transformers led to an improvement of the state estimation capabilities [19]. The estimation setup considered in this paper is somewhat similar to the two-level estimator framework proposed in [25], where the bus system is split into different areas where SCADA-like systems compute separately estimates. These estimates are centralized and combined to obtain an overall estimate of the entire power system. Unlike the aforementioned approach, we do not centralize the estimates computed by different areas, but use local collaboration to obtain an overall estimate of the power grid. We assume that the agents (PDCs) can communicate with each other and the SCADA system, thereby forming a communication network. We denote by

$$\mathcal{N}_i = \{j \text{ such that } i \text{ communicates with } j\}$$

the neighborhood of agent i, i.e., all agents it is capable to communicate with (by convention i belongs to $\mathcal{N}_i$). We assume linear sensing models for the agents, given by

$$y_i(k) = C_i x(k) + v_i(k), \tag{2}$$

where $y_i(k) \in \mathbb{R}^{p_i}$ is the observation of the state $x(k)$ made by the agent i and $v_i(k) \in \mathbb{R}^{p_i}$ is the measurement noise, assumed Gaussian with zero mean and covariance matrix R_i. The measurements that are normally included in practical state estimators are voltage magnitudes and angle differences, active and reactive powers, current magnitude flows, magnitude of turn ratios, phase shift angles of transformers, and active and reactive power flows [20]. The above model is usually obtained as a linearization of a nonlinear sensing model, where the matrix C_i is the Hessian of a nonlinear function $c_i(x)$ computed at some nominal point [10], relating the measurements to the states.

Remark 2.1: We would like to point out that the entries of matrix C_i reflect only components from the state vector $x(k)$ related to the bus(es) the agent i monitors. This implies that the agent i has only a local view on the state of the network, but through collaboration with other agents, this view can be potentially enlarged.

We denote by $\hat{x}_i(k)$ and by $e_i(k) \triangleq x(k) - \hat{x}_i(k)$ the local estimate and the estimation error computed by agent i, respectively. Each agent i associates with each of its neighbors a *trust metric* denoted by T_{ij}, for $j \in \mathcal{N}_i$. Intuitively, the trust metrics designate the weight agent i gives to the information received from its neighbors.

Problem: We assume that some agents can become faulty or under the control of non-authorized entities that can cause the respective agents to spread false data on the power grid to the other agents. This false information can affect the computation of control strategies for the power generation and transmission needed to cope with changes in the state of the power grid. Our goal is to propose a strategy aimed at limiting the effect of false data injection on the state estimate computation, based on the notion of *trust*.

III. Trust Model

Trust appears in various ways and meanings. We can refer to the reduced trustworthiness of a sensor, meaning that the sensor may have been compromised, or we can refer to the trustworthiness of the data transmitted by a sensor. Similarly, we can refer to a compromised link due to jamming, which reduces the trustworthiness of the link. Thus trust in sensor networks, and more generally in hybrid networks consisting of collaborating humans and automated agents (sensors, actuators, computers) is a composite entity, represented by several metrics and/or parameters.

There are various ways to represent trust weights numerically. In some trust schemes, continuous or discrete numerical values are assigned to measure the level of trustworthiness. For example, in [18], an entity's opinion about the trustworthiness of a certificate is described by a continuous value in [0, 1]. In [24], a 2-tuple in $[0,1]^2$ describes the trust opinion. In [12], the metric is a triplet in $[0,1]^3$, where the elements in the triplet represent belief, disbelief, and uncertainty, respectively (we denoted by $[0,1]^n$ the n times Cartesian product of the set $[0,1]$). Trust can also be interpreted as probability. In [13], subjective probability is employed, while objective probability is used in [14]. As a concept of uncertainty, entropy in information theory is a natural measurement of trust as well [23]. In the extreme case, trust can be binary: trust (trust weight=1) or distrust (trust weight=0); because either there is 100% security in the network or the approach to evaluate trust is very coarse.

In this paper, we assume each agent i assigns a trust metric to each of its neighbors j, denoted by T_{ij}, which refers to the reliability of data received from agent j. We represent trust values as non-negative real numbers taking values in

the interval [0, T_{max}], for some positive real T_{max}. In the following, these trust values will be used in conjunction with a multi-agent estimation algorithm to limit the negative effect on the state estimation process caused by false data injection.

IV. Trust-based Multi Agent State Estimation

In this section we present the trust-based multi-agent filtering scheme aimed at improving the security of the state estimation process in the power grid. The section is divided into three parts. First, we present a multi-agent filtering scheme. Second, we describe the update mechanism for the trust values, based on the behavior of the agents. Third, we combine the multi-agent filtering scheme with the trust update mechanism, which ensures that agents with low trust values have limited influence on the estimation process.

A. Distributed Estimation

A fundamental problem in sensor networks is developing multi-agent (distributed) algorithms for the state estimation of a process of interest. Generically, a process is observed by a group of sensors organized in a network. The goal of each sensor is to compute accurate state estimates. The distributed filtering (estimation) problem has received a lot of attention during the past years, starting with the contributions made by Borkar and Varaiya [6]. The main idea behind distributed estimation, found in most of the papers addressing this problem, consists of using a standard Kalman filter locally, together with a consensus step in order to ensure that the local estimates agree [7], [16], [21].

In what follows, we use a simplified version of the algorithm proposed in [21], which is described next. For simplicity we omitted the time index in Algorithm 1. There are several approaches for computing the filtering gains. In [21], the authors propose the filtering gains to be computed using the local Kalman filter equations. Other ideas include the off-line computation of the (stationary) filtering gains using Linear Matrix Inequalities techniques [16], which takes into account the topology of the network. In line 5 of Algorithm 1, the local information is linearly combined with information received from neighbors. Unlike the algorithm introduced in [21], we assume that only local estimates are exchanged but not output measurements. We will refer to line 5 as either the *information fusion step* or the *consensus step*. Further on in the paper, we will focus our analysis on the values of the weights w_{ij}, which are positive values summing up to one. Through these weights each agent controls how the information received from neighbors is used.

Algorithm 1: Distributed Filtering

Input: μ_0, P_0

1 Initialization: $\hat{x}_i = \mu_0$, $P_i = P_0$

2 **while** new data exists

3 Compute the filtering gain L_i

4 Compute the intermediate estimate of the state:

$$\varphi_i = \hat{x}_i + L_i(y_i - C_i\hat{x}_i)$$

5 Estimate the state after a Consensus step:

$$\xi_i = \sum_{j\in\mathcal{N}_i} w_{ij}\varphi_j$$

6 Update the state of the local filter:

$$\hat{x}_i = A\xi_i$$

Remark 4.1: It is reasonable to assume that the agents cannot observe the entire state of the grid. However, through collaboration (line 5 of Algorithm 1), provided that the power grid is globally observable, the agents will potentially have a global view on the state of the power grid.

B. Trust Update Algorithm

Due to the dynamic nature of agents (some of them may become faulty, receive measurements from faulty equipment, or may come under the control of unauthorized entities), agents need to implement a mechanism for updating the trust values T_{ij}. Examples of trust update mechanisms are presented in [15], in the context of reputation systems where the update is based on the notion of *belief divergence* and in [17] in the context of distributed estimation.

In this paper we pursue a different avenue of investigation. We assume that the agents "learn" the behavior patterns of their neighbors, and when they determine significant changes in these patterns, they adjust the trust values accordingly, i.e., by decreasing them. We assume that the learning period takes place during the initial operation of the system, when it is reasonable to assume that the agents function properly, i.e., the information provided by them is correct.

As mentioned earlier, agent i receives from its neighbors their local estimates $\hat{x}_j(k)$, for $j \in \mathcal{N}_i$. Let us denote by $e_{ij}(k)$ the difference between the estimates of agents i and j, i.e.

$$e_{ij}(k) = \hat{x}_i(k) - \hat{x}_j(k).$$

We note that $e_{ij}(k)$ can be equivalently written as

$$e_{ij}(k) = e_j(k) - e_i(k),$$

where $e_i(k) = x(k) - \hat{x}_i(k)$ is the estimation error at agent i. If we define the vector $\mathbf{e}(k) = (e_i(k))$, then it can be easily shown that this vector has a multivariate normal distribution, for all k, since $\mathbf{e}(k)$ is updated according to linear dynamics with initial state normally distributed. Consequently, it can be shown that the vectors $e_{ij}(k)$ have multivariate normal distributions as well. From the equations of the distributed estimation algorithm we get that $e_i(k)$ have zero means, and therefore $e_{ij}(k)$ have zero means as well, for all $k \geq 0$. Let us denote by $P_{ij}(k)$ the covariance matrices of the vectors $e_{ij}(k)$. Unfortunately, due to cross-correlations, computing these matrices exactly is intractable for large values of k (in fact the complexity increases exponentially with time).

We consider that *the statistics* of the vectors $e_{ij}(k)$ determine the *behavior patterns* of the agent i's neighbors. We can define confidence regions for $e_{ij}(k)$ based on the *chi-square distribution*. It is well known that for a multivariate normally distributed vector X in $\mathbb{R}^n$, with mean μ and covariance Σ, the region

$$\{x \mid (x-\mu)'\Sigma^{-1}(x-\mu) \leq \chi^2(\alpha)\}, \tag{3}$$

contains $(1-\alpha)100\%$ of the probability in the distribution, where $\chi^2(\alpha)$ is the chi-square distribution with n degrees of freedom, computed at α. By varying α we can define different confidence regions.

The update mechanism of the trust values is based on the following idea. Every time $e_{ij}(k)$ is outside the confidence region determined by parameter α, the trust value T_{ij} is decreased up to a zero value, while every time $e_{ij}(k)$ is inside the confidence region, the trust value T_{ij} is increased up to a maximum value T_{max}.

Learning - As mentioned earlier, the behavior patterns are determined by the covariance matrices $P_{ij}(k)$ whose exact computation is unfortunately not tractable for large values of k (computing $P_{ij}(k)$ is in the same spirit as computing the gains in a decentralized control problem; problem which is still open for decades [4]). Therefore, the agents need to approximate these matrices. Denoting by $\hat{P}_{ij}(k)$ the estimate of $P_{ij}(k)$, we propose the use of a time averaging filter given by

$$\hat{P}_{ij}(k+1) = \frac{1}{k+1} e_{ij}(k) e_{ij}(k)' + \frac{k}{k+1} \hat{P}_{ij}(k), \qquad (4)$$

to approximate the covariances matrices. The intuition behind this approach is the following. Under the assumption that the parameters of the stochastic process are time-invariant (and under the assumption that the estimation errors are mean square stable), the distributions of the estimation errors $e_i(k)$ will converge to some stationary distributions. Consequently, the same will happen for the random vectors $e_{ij}(k)$. The covariance matrix of the stationary distribution of $e_{ij}(k)$ can be approximated by taking the average over a sufficiently large numbers of samples. Writing the averaging operation iteratively, it results in a filter as in (4). Let $\bar{P}_{ij}$ be the approximation of the covariance matrix of the stationary distribution of the vector $e_{ij}(k)$, given by

$$\bar{P}_{ij} = \hat{P}_{ij}(K),$$

where K is sufficiently large and represents the learning horizon. Algorithm 2 summarizes the trust update mechanism implemented by agent i (for brevity, the time index is ignored).

According to the aforementioned algorithm, every time the error $e_{ij}(k)$ lies in the confidence region $\{x \mid x' \bar{P}_{ij}^{-1} x \le \chi^2(\alpha)\}$, the trust value T_{ij} is increased by δ_1 (a positive real value) up to T_{max}, while every time the trust values lie outside the aforementioned confidence region, they are decreased by δ_2 (assumed positive). We note that even a correctly functioning agent j can have its trust value decreased. In the long run, the trust value T_{ij} is decreased approximately $\alpha\chi^2(\alpha)100\%$ of the total time. A good idea is to choose the parameters δ_1 and δ_2 so that they reflect how close or far the errors are from the regions of type (3). For example, if $e'_{ij} \bar{P}_{ij}^{-1} e_{ij}$ is large compared to $\chi^2(\alpha)$, then δ_2 should be chosen large as well, so that the corresponding trust value is decreased rapidly. We chose an incremental procedure for updating the trust values, through the parameters δ_1 and δ_2. We can also envisage a model where the increase/decrease rate of the trust values is exponential. In such a case, we would have

$$T_{ij} = \begin{cases} \min\{\delta_1 T_{ij}, T_{max}\} & e'_{ij} \bar{P}_{ij}^{-1} e_{ij} \le \chi^2(\alpha) \\ \delta_2 T_{ij} & e'_{ij} \bar{P}_{ij}^{-1} e_{ij} > \chi^2(\alpha) \end{cases}$$

where $\delta_1 > 1$ and $0 < \delta_2 < 1$.

Algorithm 2: Trust update

Input: $\bar{P}_{ij}$, T_{max}, δ_1, δ_2, α

while new data exists

Compute the errors between estimates:

$$e_{ij} = \hat{x}_i - \hat{x}_j$$

Update the trust values:

$$T_{ij} = \begin{cases} \min\{T_{ij} + \delta_1, T_{max}\} & e'_{ij} \bar{P}_{ij}^{-1} e_{ij} \le \chi^2(\alpha) \\ \max\{T_{ij} - \delta_2, 0\} & e'_{ij} \bar{P}_{ij}^{-1} e_{ij} > \chi^2(\alpha) \end{cases}$$

end while

C. Trust-based Distributed Estimation

In this subsection we introduce an estimation algorithm where the distributed estimation scheme presented in Algorithm 1 is combined with the trust update mechanism presented in Algorithm 2.

We note that the weights w_{ij} control how much the neighbors influence the update of the estimates $\hat{x}_i(k)$. A small value of the weight w_{ij} means that agent j will have little influence on the agent i. Therefore, it makes sense to choose the weights w_{ij} to be proportional to the trust values T_{ij}. We propose to choose the weights w_{ij} as weighted trust values, so that they sum up to one, i.e.,

$$w_{ij} = \frac{T_{ij}}{\sum_{j \in N_i} T_{ij}}.$$

This way the weights decrease with the trust values, so that agents with low reliability will have little influence in the computation of the local estimates. Algorithm 3 presents the trust-based distributed estimation algorithm. We would like to emphasize that Algorithm 3 acts on two levels. On one level it extends the scope of the agents' view with respect to the grid by making the estimation process distributed. This way, through collaboration, the agents can potentially have a more accurate image on the global state of the grid, which otherwise would be more difficult. On another level, the agents limit the effect of false data injection by updating the agents' trust in their neighbors, according to their recorded behavior. Thus, the estimation process becomes more robust.

V. Numerical example

We consider an example of a power grid with three generators and nine buses [3], shown in Figure 1. We assume that a PMU is placed at each bus that measures the complex voltages and currents (in the case of adjacent buses). We use an estimation model similar to the one presented in [26]. Under this model, the state vector is formed by the voltages measured at buses, i.e., $X = (U_i)$, where U_i is the complex voltage at bus i. The measurement models are as follows. In

Algorithm 3: Trust-based Distributed Estimation Algorithm

Input: μ_0, P_0,T_{max}, α, δ_1, δ_2, K

Initialization: $\hat{x}_i = \mu_0$, $P_i = P_0$, $T_{ij} = T_{max}$

For a learning horizon K, apply Algorithm 1 together with (4) to approximate the covariance matrices $P_{ij}(k)$.

Set $\bar{P}_{ij} = \hat{P}_{ij}(k)$

while new data exists

Compute the filtering gain L_i

Compute the intermediate estimate of the state:

$$\varphi_i = \hat{x}_i + L_i(y_i - C_i\hat{x}_i)$$

Compute the errors between estimates:

$$e_{ij} = \hat{x}_i - \hat{x}_j$$

Update the trust values:

$$T_{ij} = \begin{cases} \min\{T_{ij}+\delta_1, T_{max}\} & e'_{ij}\bar{P}_{ij}^{-1}e_{ij} \le \chi^2(\alpha) \\ \max\{T_{ij}-\delta_2, 0\} & e'_{ij}\bar{P}_{ij}^{-1}e_{ij} > \chi^2(\alpha) \end{cases}$$

Update the consensus weights

$$w_{ij} = \frac{T_{ij}}{\sum_{j\in\mathcal{N}_i} T_{ij}}.$$

Compute the state after a Consensus step:

$$\xi_i = \sum_{j\in\mathcal{N}_i} w_{ij}\varphi_j$$

Update the state of the local filter:

$$\hat{x}_i = A\xi_i$$

end while

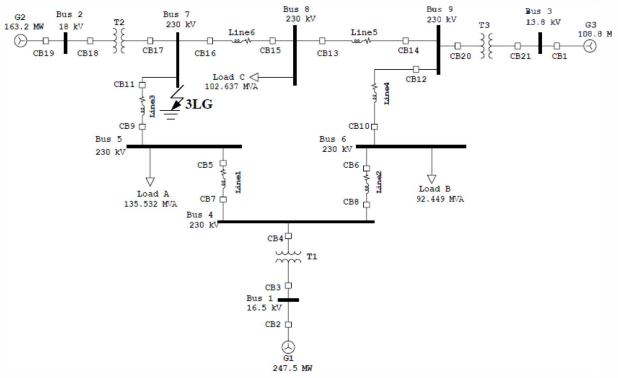


Fig. 1. The 3-generators, 9-bus system

the case where the buses (i,j) are adjacent (e.g., $(2,7)$, $(3,9)$ and $(1,4)$ in Figure 1), the complex measurement model for each PMU is given by

$$Z_i(k) = \begin{pmatrix} 1 & 0 \\ Y_{ij} & -Y_{ij} \end{pmatrix}\begin{pmatrix} X_i(k) \\ X_j(k) \end{pmatrix} + V_i(k),$$

where the measurement vector $Z_i(k)' = (U_i(k), I_{ij}(k))$ encompasses the complex voltage at bus i and the complex current on the line (i,j), Y_{ij} is the admittance of line (i,j) and $V_i(k)$ is the complex measurement noise. In the case bus i has no other adjacent bus, the measurement model gets simplified to

$$Z_i(k) = X_i(k) + V_i(k).$$

Thus, we can generically represent the complex measurement model as

$$Z_i(k) = H_iX(k) + V_i(k).$$

The real valued measurement model is given by

$$y_i(k) = C_ix(k) + v_i(k),$$

where

$$y_i = \begin{pmatrix} Re(Z_i) \\ Im(Z_i) \end{pmatrix}, C_i = \begin{pmatrix} Re(H_i) & -Im(H_i) \\ Im(H_i) & Re(H_i) \end{pmatrix}, v_i = \begin{pmatrix} Re(V_i) \\ Im(V_i) \end{pmatrix},$$

and $x' = (Re(X'), Im(X'))$.

We consider the power system to be reasonably stable, and where the oscillations in the state variables are assumed to be small and induced by a white Gaussian noise. Thus, we model the dynamics of the power system by

$$x(k+1) = x(k) + w(k),\ x(0) = x_0,$$

where a solution to the power grid in Figure 1 is used as initial state for the dynamics of the state variables.

We assume that each PMU plays the role of a PDC and that they form a communication network, as shown in Figure 2, where each node represents a PMU.

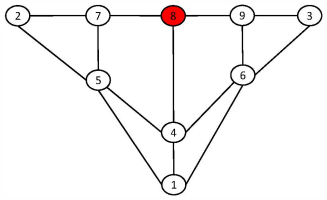
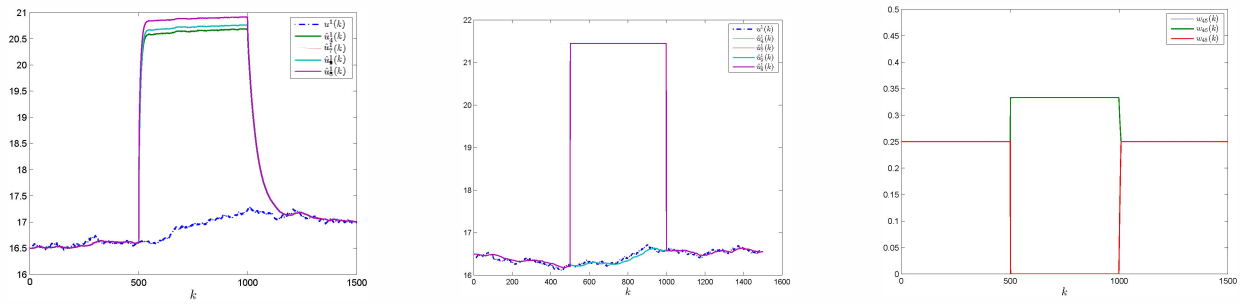


Fig. 2. PMU network

In the numerical simulations that follow, we assume the estimation is performed over 1500 time units and that during the interval $[500, 1000]$ agent (PDC) 8 (color red in Figure 2) shares false data with the neighbors.

Figure 3(a) shows the voltage at bus 1 ($u^1(k)$) and the estimates of the voltage at bus 1 made by agents 4, 7 and 9 ($\hat{u}^1_4(k), \hat{u}^1_7(k), \hat{u}^1_9(k)$), together with the false data injected by agent 8 ($\hat{u}^1_8(k)$), when Algorithm 1 is used (and assigning equal consensus weights), i.e., the trust update mechanism is *not applied*. We note that the estimates of the aforementioned agents are significantly affected. We have repeated the numerical simulation using Algorithm 3. The results are presented in Figure 3(b). We note that although agent 8 shares false information, its neighbors are not affected this time. This is because the neighbors of agent 8 adjust their trust values so that the data coming from agent 8 are rejected. Figure 3(c) shows the time evolution of the consensus weights of agent 4. We note that between the interval $[500, 1000]$ the weight $w_{4,8}(k)$ is lowered to zero, as a result of decreased trust in agent 8.

Remark 5.1: The local filtering gains were computed using only the observable part of the pairs (A, C_i). However,



(a) (b) (c)

Fig. 3. (a) Estimates of the voltage at bus 1 using Algorithm 1, with agent 8 injecting false data; (b) Estimates of the voltage at bus 1 using Algorithm 3, with agent 8 injecting false data; (c) The evolution of agent 4's weights.

as it can be seen from Figure 3(b), through collaboration (consensus step) even agents that do not measure directly the voltage at bus 1 are still able to compute good estimates for the aforementioned voltage.

VI. Conclusions

In this paper we proposed an algorithm for the state estimation of the power grid, aimed at making the estimation process robust to false data injection. Our approach consisted of combining a multi-agent filtering algorithm with a trust metric, where agents with low trust values have little influence on the computation of the estimates. In addition, we proposed a trust update mechanism so that the trust values of the agents are updated according to their recorded behavior.

References

[1] *Energy Independence and Security Act of 2007*. Public Law No: 110-140, Title XIII, Sec. 1301, 2007.

[2] *Report to NIST on the Smart Grid Interoperability Standards Roadmap*. Electric Power Research Institute, August 2009.

[3] P.M. Anderson and A.A. Fouad. *Power System Control and Stability*. Vol. 1, The Iowa State University Press, Iowa, USA, 1977.

[4] V.D. Blondel and A. Megretski. *Unsolved Problems in Mathematical Systems and Control Theory*. Princeton University Press, New Jersey, USA, 2004.

[5] R.B. Bobba, K.M. Rogers, Q. Wang, H. Khurana, K. Nahrstedt, and T.J. Overbye. Detecting false data injection attacks on DC state estimation. *Proceedings of the First Workshop on Secure Control Systems (SCS 2010), CPSWEEK2010*, April 2010.

[6] V. Borkar and P. Varaiya. Asymptotic agreement in distributed estimation. *IEEE Trans. Autom. Control*, AC-27(3):650–655, Jun 1982.

[7] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri. Distributed Kalman filtering based on consensus strategies. *IEEE Journal on Selected Area in Communication*, 26(4):622–633, May 2008.

[8] A.S. Debs and R.E. Larson. A dynamic estimator for tracking the state of a power system. *IEEE Transactions on Power Apparatus and Systems*, PAS 89(7), Semptember 1970.

[9] S. Horowitz, A.G. Phadke, and B.A. Renz. The future of power transmission. *Taking the Measure of the Smart Grid around the World, Special Issue of the IEEE Power abd Energy Magazine*, pages 4–10, 2010.

[10] A. Jain and N.R. Shivakumar. Power system tracking and dynamic state estimation. *IEEE PES Power Systems Conference Exposition (PSCE)*, March 2009.

[11] T. Jiang, I. Matei, and J.S. Baras. Trust based distributed Kalman filtering approach for mode estimation in power systems. In *Proceedings of the First Workshop on Secure Control Systems (SCS 2010)*, April 2010.

[12] A. Josang. A logic for uncertain probabilities. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(3):279–311, June 2001.

[13] A. Josang and R. Ismail. The beta reputation system. *Proceedings of the 15th Bled Conference on Electronic Commerce*, 2002.

[14] S.D. Kamvar, M.T. Schlosser, and H. Garcia-Molina. The EigenTrust algorithm for reputation management in p2p networks. *Proceedings of the 12th Initernational World Wid Web Conference*, pages 640–651, 2003.

[15] C. De Kerchove and P. Van Dooren. Iterative filtering for a dynamical reputation system. *SIAM Journal of Matrix Analysis and its Applications*, 5(4):1–28, 2010.

[16] I. Matei and J.S. Baras. Consensus-based linear filtering. *Proceedings of the 49th IEEE Conference on Decision and Control*, pages 7009–7014, December 2010.

[17] I. Matei, T. Jiang, and J.S. Baras. A composite trust model and its applications to collaborative distributed information fusion. *Proceedings of the 12th Internation Conference on Information Fusion (Fusion 2009)*, pages 1950–1957, 2009.

[18] U. Maurer. Modelling a public-key infrastructure. *Proceedings of the 1996 European Symposium on Research in Computer Security (ESORICS'96)*, pages 325–350, 1996.

[19] A. P. Meliopoulos, G. J. Cokkinides, F. Galvan, B. Fardanesh, and P. Myrda. Advances in the SuperCalibrator Concept - Practical Implementations. *Proceedings of the 40st Annual Hawaii International Conference on System Sciences*, January 2007.

[20] A. Monticelli. Electric power system state estimation. *Proceedings of the IEEE*, 88(2):262 –282, February 2000.

[21] R. Olfati-Saber. Distributed Kalman filtering for sensor networks. *Proceedings of the 46th IEEE Conference on Decision and Control*, pages 5492–5498, 2007.

[22] N. R. Shivakumar and A. Jain. Including phasor measurements in dynamic state estimation of power systems. *Proceedings of the 2008 International Conference on Power System Analysis Control and Optimization(PSACO)*, March 2008.

[23] Y.L. Sun, Z. Han, W. Yu, and K.J. Ray Liu. A trust evaluation framework in distributed networks: Vulnerability analysis and defense against attacks. *Proceedings of 25th IEEE International Conference on Computer Communications (INFOCOM'06)*, pages 1–13, April 2006.

[24] G. Theodorakopoulos and J.S. Baras. On trust models and trust evaluation metrics for ad-hoc networks. *IEEE Journal on Selected Areas in Communications, Security in Wireless Ad-Hoc Networks*, 24(2):318–328, February 2006.

[25] G. Valverde and V. Terzija. PMU-based multi-area state estimation with low data exchange. In *Innovative Smart Grid Technologies Conference Europe (ISGT Europe), 2010 IEEE PES*, pages 1 –7, October 2010.

[26] J. Zhang, G. Welch, and G. Bishop. Observability and estimation uncertainty analysis for PMU placement analternatives. In *North American Power Symposium (NAPS), 2010*, pages 1 –8, September 2010.