A Trust Based Distributed Kalman Filtering Approach for Mode Estimation in Power Systems

Tao Jiang Ion Matei John S. Baras
Institute for Systems Research
and Department of Electrical and Computer Engineering
University of Maryland
College Park, MD
{tjiang,imatei,baras}@umd.edu

***Abstract*—We consider distributed mode estimation in power systems. The measurements are observed by PMUs (Power Management Units). We introduce a novel model of trust, using weights on the graph links and nodes that represent the networked PMUs. We describe two algorithms that integrate distributed Kalman filtering with these trust weights. We consider two interpretations of these trust weights as information accuracy and reliability. We show that by appropriate use of these weights the distributed estimation algorithm avoids using information from untrusted PMUs. Simulation experiments further demonstrate the behavior of these algorithms.**

I. Introduction

The digital control and protection of power systems require the collection of huge amounts of data to estimate various parameters in real-time. For instance, when a short circuit occurs in a power transmission line, the steady state values of the post-fault currents and voltages must be estimated to locate the fault location. Furthermore, next generation power grids involve large interconnected power networks, resulting in greater emphasis on reliable and secure operations [1]. The large scale communication networks underlying the power grids make it impossible to collect data and control power systems in a centralized manner. The new power systems must have a *distributed* communication and control system in the face of an ever changing environment such as equipment failures and even attacks (e.g. cyber-attacks).

Because the new communication and control system enables many more interactions between many more participants, it has security requirements beyond the conventional Confidentiality, Integrity and Availability properties provided by conventional security systems. For example, integrity and confidentiality have nothing to say about the quality of the data obtained from various substations. Nor does confidentiality protect against disclosure of a measurement by an intended recipient. As the community of participants in the power grids operations grows, properties that involve the behavior of participants become increasingly critical for reliable operations and difficult to deal with.

One crucial question is: how the control system can trust the data provided by the communication network? Our research efforts are motivated by two key observations. First, due to the distributed and dynamic nature of the power systems, the uncertainty of data accuracy has to be taken into consideration. Second, PMUs in the power grids often operate unattended in physically insecure environments, and are designed with an emphasis on numbers and low cost which makes security measures such as tamper-proof hardware not cost effective. Therefore, we cannot only resort to costly cryptography to guarantee reliable operations. In this paper, the concept of trust is used in a specific problem of power systems: mode estimation. We propose a trust based distributed Kalman filtering approach to estimate the modes of power systems. We show that by establishing appropriate trust relations, the estimation is more resilient to attacks.

II. Problem formulation

Large interconnected power networks are often associated with inter-area oscillations between clusters of generators. These inter-area oscillations are of critical importance in system stability and require on-line observation and control [2]. The inter-area oscillations (often referred to as modes) are damped sinusoids which all have a particular frequency and damping factor. The damping factor determines the transient ability of the system to stabilize post disturbance. Therefore, it is critical to have a rapid and good estimation of the damping factor in large distributed power systems.

This work addresses automatic detection of oscillations in power systems using dynamic data such as currents, voltages and angle differences measured across transmission lines given that some measurements are false. The measurements are provided on-line by the PMUs distributed throughout the large-area power system. The power system is assumed being driven by disturbances around nominal operating points ([3]), therefore linear models can be used to linearize the system and to model oscillations.

The linearizaton method used in this paper is based on the work by Lee and Poon [4]. Disturbance inputs in a power system (such as load changes) consist of M frequency modes and, with the initial steady-state value eliminated, can be generalized over a specific time period as

$$f(t) \quad = \quad a_1 \exp(\sigma_1 t) \cos(\omega_1 t)$$

$$+\sum_{j=2}^{M} a_j \exp(\sigma_j t)\cos(\omega_j t+\phi_i) \quad (1)$$

where a_i are oscillation amplitudes, σ_i are damping constants, ω_i are the oscillation frequencies and ϕ_i are phase angles of the oscillations. Without loss of generality, we consider two modes in Eqn. (1), given by

$$\begin{aligned} f(t) &= a_1\exp(\sigma_1 t)\cos(\omega_1 t)+ \\ & \quad a_2\exp(\sigma_2 t)\cos(\omega_2 t+\phi_2), \end{aligned} \quad (2)$$

which is a nonlinear function of the parameters a_i, σ_i and ϕ_i. Using the first two terms in the Taylor series expansion of the exponential function and expanding the trigonometric functions, we have that

$$\begin{aligned} f(t) &= a_1(1+\sigma_1 t)\cos(\omega_1 t)+ \\ & \quad a_2(1+\sigma_2 t)[\cos\phi_2\cos(\omega_2 t)-\sin\phi_2\sin(\omega_2 t)]. \end{aligned} \quad (3)$$

We introduce the notation:

$$\begin{array}{ll} x_1 = a_1 & x_2 = a_1\sigma_1 \\ x_2 = a_2\cos\phi_2 & x_4 = a_2\sigma_2\cos\phi_2 \\ x_5 = a_2\sin\phi_2 & x_6 = a_2\sigma_2\sin\phi_2 \end{array}$$

and

$$\begin{array}{ll} c_{11} = \cos(\omega_1 t) & c_{12} = t\cos(\omega_1 t) \\ c_{13} = \cos(\omega_2 t) & c_{14} = t\cos(\omega_2 t) \\ c_{15} = -\sin(\omega_2 t) & c_{16} = -t\sin(\omega_2 t) \end{array}$$

Then we have

$$f(t) = \sum_{i=1}^{6} c_{1i}(t)x_i(t). \quad (4)$$

The power system is sampled at a preselected rate, say every Δt seconds. Eqn. (4) can be written in discrete time k, $k = 1,\ldots,K$. We have the linear measurement model as the following:

$$y_i(k) = C_i x(k) + v_i(k), \quad (5)$$

where $y_i(k)$ is the measurement of the state $x(k)$ made by PMU i, and $v_i(k)$ is the measurement noise assumed Gaussian with zero mean and covariance matrix R_i.

For N measurements, Eqn. (5) can be written in vector form as

$$y(k) = Cy(k) + v(k). \quad (6)$$

The state transition matrix $A(k)$, which relates the state $x(k)$ to $x(k-1)$ is the identity matrix. The state space equation is given by

$$x(k+1) = A(k)x(k) + w(k), \quad (7)$$

where $w(k) \in \mathbb{R}^n$ is the state noise, assumed Gaussian with zero mean and covariance matrix Q. The initial state x_0 has a Gaussian distribution, with mean μ_0 and covariance matrix P_0. Eqn. (6) and (7) form a linear random process that can be estimated using the Kalman filter algorithm.

Having estimated the parameter vector $x(k)$, the amplitude, damping constant, and phase angle can be calculated at any time step k using the following equations:

$$a_1(k) = x_1(k) \quad (8)$$

$$\sigma_1(k) = \frac{x_2(k)}{x_1(k)} \quad (9)$$

$$a_2(k) = [x_3^2(k)+x_5^2(k)]^{1/2} \quad (10)$$

$$\sigma_2(k) = \left[\frac{x_4^2(k)+x_6^2(k)}{x_3^2(k)+x_5^2(k)}\right]^{1/2} \quad (11)$$

$$\phi_2(k) = \tan^{-1}\left[\frac{x_6(k)}{x_4(k)}\right] = \tan^{-1}\left[\frac{x_5(k)}{x_3(k)}\right]. \quad (12)$$

Fig. 1 shows a power system with several PMUs. Measurements from the entire grid are synchronized via a satellite. As

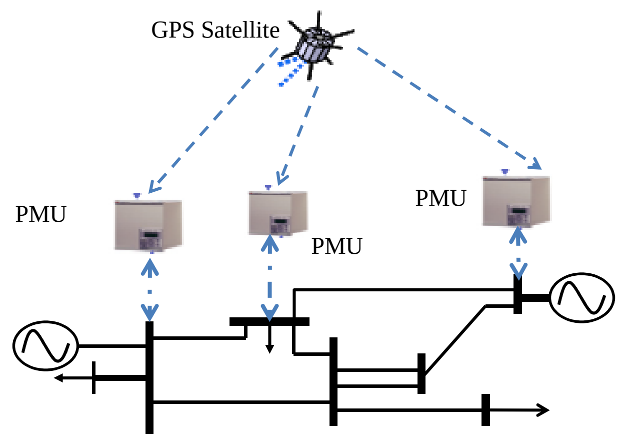

Fig. 1. An Overview of the Monitoring System

we discussed in Section I, distributed computation and communication are needed given the large scale communication networks underlying the power grid. We consider a power system with N multiple recording sites (PMUs) to measure the output signals, indexed by i. The goal of each PMU i is to compute an accurate estimation of the state $x(k)$, using: the local measurements $y_i(k)$; the information received from the PMUs in its communication neighborhood (e.g. measurements and estimates); and the confidence in the information received from other PMUs provided by the trust model described in the following sections.

Each PMU i has a communication neighborhood containing PMUs with whom the PMU can exchange information. Let $\mathcal{N}_i$ denote such a communication neighborhood:

$$\mathcal{N}_i = \{j \mid i \text{ exchanges information with } j\}.$$

The communication neighborhoods of the PMUs determine a communication graph with N vertices, such that a link from i to j exists if PMU i sends information to PMU j.

We attach a positive value T_{ij} to each link (j,i) which represents the confidence value that PMU i places on the information coming from PMU j. The value T_{ij} represents a measure of the trust PMU i has in the information received from PMU j.

There are many different definitions of "trust" depending on the particular domains. An operational definition of "trust" for information, mainly considers two aspects: information *accuracy* and *reliability*. Accuracy reflects the deviation of the information from truth, and reliability is confidence in the assessment of accuracy. In this paper, we apply trust weights to the distributed estimation problem where these two aspects of trust are investigated separately.

III. Distributed Kalman filtering

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 [5]. In what follows, we use a simplified version of the algorithm proposed in [5].

Algorithm 1: Distributed Kalman Filtering algorithm with consensus step on estimates [5]

Input: μ_0, P_0

1 Initialization: $\xi_i = \mu_0$, $P_i = P_0$

2 **while** new data exists

3 Compute the intermediate Kalman estimate of the target state:

$$M_i = P_i^{-1} + C_i' R_i^{-1} C_i$$
$$L_i = M_i C_i R_i^{-1}$$
$$\varphi_i = \xi_i + L_i(y_i - C_i \xi_i)$$

4 Estimate the state after a consensus step:

$$\hat{x}_i = \varphi_i + \epsilon \textstyle\sum_{N_i \cup \{i\}} (\varphi_j - \varphi_i)$$

5 Update the state of the local Kalman filter:

$$P_i = A M_i A' + Q$$
$$\xi_i = A \hat{x}_i$$

For simplicity we omitted the time index in Algorithm 1. Notice that with the exception of line 4, the above algorithm is the standard linear Kalman filter. In line 4, the local information is linearly combined with information received from neighbors. We will refer to line 4 as either the *information fusion step* or the *consensus step*. We will focus our analysis on the values of the weights w_{ij} . In fact they will play the role of the confidence values introduced in the previous section. Unlike the original algorithm [5], we assume that only local estimates are exchanged and not output measurements as well.

IV. Distributed Kalman filtering with trust dependent weights in the consensus step

In this section we develop the distributed filtering equations that take into account the confidence (trust) of the PMUs. We address two cases reflecting what the confidence values represent. In the first case, we assume that the weights w_{ij} are a measure of the *information accuracy*, i.e. the larger the value of w_{ij} is, the more accurate the information received by i from j is. In the second case, the weights w_{ij} are a measure of the *trustworthiness* of the data received by PMU i from PMU j. It may be the case that either a PMU or a link were compromised, so that the information received from the respective PMU or through the respective link is not trustworthy.

A. Distributed Kalman Filtering with accuracy dependent consensus step

We attach to each PMU a trust value. In this subsection, the trust refers to the accuracy of information. The larger the trust value is, the more accurate the information received from the respective PMU is. The information exchanged between PMUs is represented by estimates. As previously mentioned, we denote by T_{ij} the trust PMU i has in information received from PMU j. We propose to choose the trust values to be inversely proportional to the estimation error, according to the formula:

$$T_{ij} = \frac{1}{trace(M_j)},\ j \in \mathcal{N}_i, \tag{13}$$

where M_j represents the covariance matrix of the estimation error from the standard Kalman filter step. The properties of this matrix will be affected by how *observable* the state is from PMU j, (such as the rank of matrix C_j) and how noisy the measurements are, i.e. the variance of the measurements' noise R_j. We can expect the variance of the estimation error, given by the trace of M_j, to be small for highly observable measurements with low noise. Therefore, we computed the weight values in the information fusion step, by normalizing the trust values T_{ij}:

$$w_{ij} = \frac{T_{ij}}{\sum_k T_{ik}}. \tag{14}$$

This way, we assign a larger influence to the more accurate estimates, directing the resulting average towards estimates with high accuracy. Note however that the matrix M_j is not the actual covariance matrix of the estimation error for the current estimate $\hat{x}_j$, but the covariance error given by the standard Kalman filter. In does however reflect the observability properties of the PMU, making it a good candidate for constructing the weight values. We summarize the idea introduced above in Algorithm 2.

B. Distributed estimation with reliability dependent consensus step

In this subsection we propose a distributed estimation scheme where the averaging operation depends on the reliability of the PMUs. We assume that PMUs may be compromised and may send data aimed at modifying the result of the estimation process. The update mechanism for the trust values T_{ij} is based on the notion of *belief divergence* [6]:

$$d_i = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} \|\hat{x}_i - \hat{x}_j\|^2, \tag{15}$$

where we denoted by $\hat{x}_i$ the current estimates.

Algorithm 2: Distributed Kalman Filtering Algorithm with accuracy dependent consensus step on estimates

Input: $\mu_0,\ P_0$

Initialization: $\xi_i = \mu_0,\ P_i = P_0$

while new data exists

Compute the intermediate Kalman estimate of the target state:

$$M_i = P_i^{-1} + C_i' R_i^{-1} C_i$$
$$L_i = M_i C_i R_i^{-1}$$
$$\varphi_i = \xi_i + L_i(y_i - C_i \xi_i)$$

Compute the consensus weight values:

$$T_{ij} = \frac{1}{trace(M_j)}$$
$$w_{ij} = \frac{\bar{w}_{ij}}{\sum_k \bar{w}_{ik}}$$

Estimate the state after a consensus step:

$$\hat{x}_i = \sum_{j \in \mathcal{N}_i \cup \{i\}} w_{ij} \varphi_j$$

Update the state of the local Kalman filter:

$$P_i = A M_i A' + Q$$
$$\xi_i = A \hat{x}_i$$

The belief divergence d_i, gives to PMU i a measure of how different its own estimate is with respect to the estimates of the other PMUs within its communication neighborhood.

Since the PMUs exchange only state estimates, every PMU will compute a belief divergence, d_{ij}, for each PMU in his neighborhood, according to the formula:

$$d_{ij} = \frac{1}{\mathcal{N}_i - 1} \sum_{k \in \mathcal{N}_i} \|\hat{x}_j - \hat{x}_k\|^2. \tag{16}$$

This metric shows how far a received estimate is from the other received estimates in some neighborhood. Note that in the fusion step, estimates far from their real values are prone to hurt more. However, if enough neighbors provide reliable information, the belief divergence for a PMU sending false information is going to by high. We use the locally computed belief divergence metric, to update the trust values T_{ij}. We first choose a positive constant c_i, satisfying:

$$c_i > \max\{d_{ij} \mid j \in \mathcal{N}_i\}.$$

We use the constant c_i in the following formula for updating the trust values:

$$T_{ij} = c_i - d_j,\ j \in \mathcal{N}_i \tag{17}$$

Notice that the parameters c_i were chosen so that the trust value T_{ij} are nonnegative. Moreover, c_i are discriminating in the sense that they influence the ratios T_{ij}/T_{ik}. Typically, the smaller c_i is, the more PMUs with large values of the belief divergence are penalized. From (17) we note that we favor the PMU whose estimate is close to the other estimates in its neighborhood, in a sense 'accelerating convergence' to consensus. We denote by p_{ij} the normalized versions of the trust values T_{ij}, computed according to the formula:

$$p_{ij} = \frac{T_{ij}}{\sum_{k \in \mathcal{N}_i} T_{ik}}, \tag{18}$$

which may be interpreted as the "probability the data received by PMU i from j are accurate". Note from the above formulas that, although small, the normalized trust values are not necessarily zero for PMUs with large belief divergence. Therefore if the value of a false estimate is large compared with the others, it will still influence negatively the information fusion step. That is why we introduce a thresholding scheme on the normalized trust values. Let p_i^{min} be the minimum value accepted for p_{ij}. If $p_{ij} < p_i^{min}$ the trust value T_{ij} will be set to zero, hence filtering out information that is not considered sufficiently trustworthy. The lower bound p_i^{min} should be chosen to be inversely proportional to the size (cardinality) of the neighborhood.

The updated trust values are next used to compute the weights in the consensus step:

$$w_{ij} = \frac{T_{ij}}{\sum_{k \in \mathcal{N}_i} T_{ik}}. \tag{19}$$

The distributed estimation algorithm with a reliability dependent averaging scheme is presented in Algorithm 3 below. The intuition behind our proposed algorithm is that if a node j sends false data, the other nodes will compute large belief divergence values, and hence low trust values, which together with the thresholding scheme will eliminate the node from the information flow. The consensus step has the role of producing a new state estimate by averaging the estimates on neighborhoods. If an estimate is not accurate enough, it may drag the updated estimate towards the wrong direction. By computing the consensus weight values using a trust dependent mechanism, we try to minimize the possibility of an estimate update moving in the wrong direction. By adjusting the minimum accepted value for the normalized trust values, p_i^{min} , the PMUs can control their sensibility with respect to the received data.

V. Simulations

In this section, we report results on simulations and test of our implementation of the disitributed Kalman filter algorithm to estimate the oscillation amplititudes and the damping coefficients of a practical example, given in [4]. It is noted that it has two modes at $\omega_1 = 0.4Hz$ and $\omega_2 = 0.5Hz$. A model of power system was used as shown in Figure 2. The model employs five measurements, where each PMU is installed over a line connected to one generator.

We first test Algorithm 2 against Algorithm 1, where independent white noise with different SNR was added to each measurement before feeding them into the estimation procedure. For computing the weights w_{ij} in Algorithm 1 we used the original scheme proposed in [5], the value for

Algorithm 3: Distributed Kalman Filtering Algorithm with a reliability dependent consensus step on estimates

Input: $\mu_0,\ P_0$

Initialization: $\xi_i = \mu_0,\ P_i = P_0$

while new data exists

Compute the intermediate Kalman estimate of the target state:

$$M_i = P_i^{-1} + C_i' R_i^{-1} C_i$$
$$L_i = M_i C_i R_i^{-1}$$
$$\varphi_i = \xi_i + L_i(y_i - C_i \xi_i)$$

Compute locally the belief divergence:

$$d_{ij} = \frac{1}{\mathcal{N}_i - 1} \sum_{k \in \mathcal{N}_i} \|\varphi_j - \varphi_k\|^2$$

Compute the trust values:

$$T_{ij} = c_i - \bar{d}_{ij},\ j \in \mathcal{N}_i$$

Compute the normalized trust values:

$$p_{ij} = \frac{T_{ij}}{\sum_k T_{ik}}$$

Eliminate insufficiently accurate data by setting T_{ij} to zero if $p_{ij} < p_i^{min}$

Compute the consensus weight values:

$$w_{ij} = \frac{T_{ij}}{\sum_k T_{ik}}$$

Estimate the state after a consensus step:

$$\hat{x}_i = \sum_{j \in \mathcal{N}_i \cup \{i\}} w_{ij} \varphi_j$$

Update the state of the local Kalman filter:

$$P_i = A M_i A' + Q$$
$$\xi_i = A \hat{x}_i$$

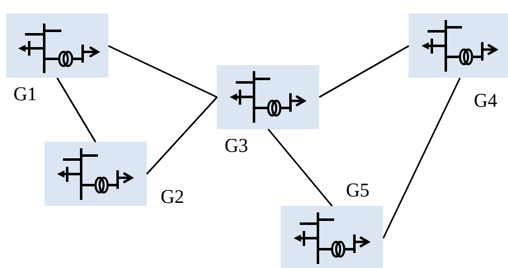

Fig. 2. Power System for Simulations

ϵ being chosen such that the average estimation error per node was as small as posssible. More precisely we want to compare the average estimation errors per node, given by the two algorithms. Since the trust weights are computed so that more weight is given to information coming from PMUs with smaller variance of the estimation error, we would expect Algorithm 2 to perform better, in the sense that the average estimation error per node should converge to a smaller value.

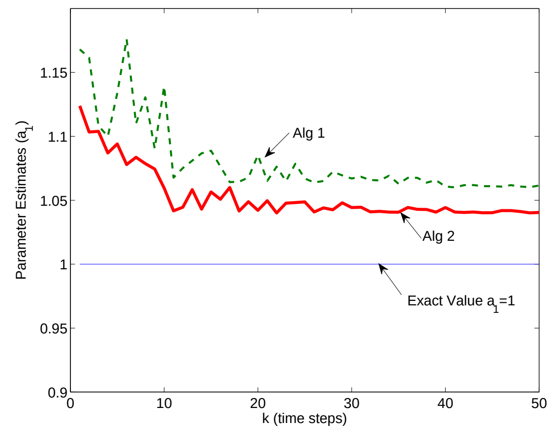

Fig. 3. Comparison of estimating parameter a_1 given by Alg 1 and Alg 2 respectively

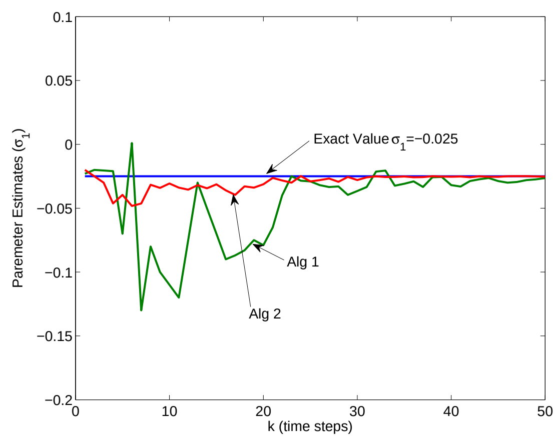

Fig. 4. Comparison of estimating parameter σ_1 given by Alg 1 and Alg 2 respectively

The comparison results for estimating parameters a_1 and σ_1 are shown in Fig. 3 and 4. The results for a_2 and σ_2 are similar. We observe that Algorithm 2, as expected, performs better. This is mainly due to the fact that in the estimation fusion step, we move the estimate updates closer to the local estimate with better observability and lower measurement noise.

For testing Algorithm 3, we assume that the measurements from the PMU connecting G3 were compromised and send false information to all the other PMUs. The goal of the PMU in G3 is to shift the estimates of other nodes away from their true values. We consider the case when the PMU connecting G3 sends to its neighbors a white noise with standard deviation equal to 0.1. The PMU connecting G3 is chosen because it is centered and has potential to do a lot of damage since it is connected to all other PMUs. We compare the results using Algorithm 1 and Algorithm 3. The results for estimating parameter a_1 and σ_1 are shown in Figure 5 and Figure 6 respectively.

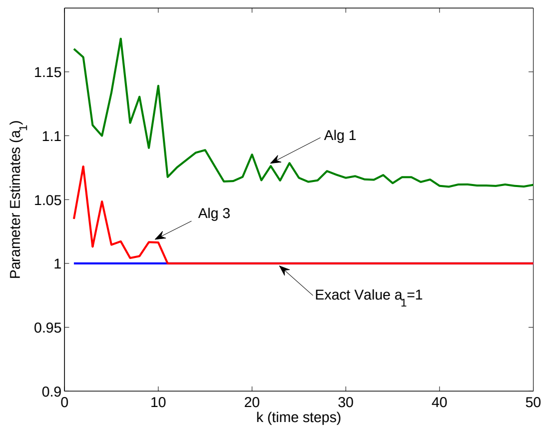

Fig. 5. Distributed Kalmann filtering with constant false information, estimating a_1

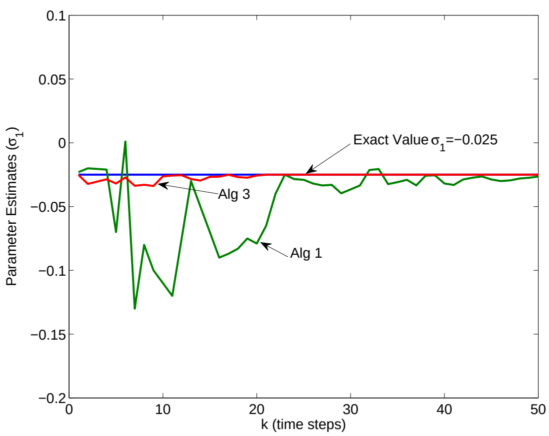

Fig. 6. Distributed Kalmann filtering with constant false information, estimating σ_1

We observe that Algorithm 3 is able to detect the false data provided by the PMU connecting G3 and eliminate it from further participation in the processing. The other PMUs are able to estimate closely the parameters. However, the false data does have influence on how fast the estmates converge to the real value at the beginning, since the false data are not immediately detected and rejected, the PMUs are able to compute parameter estimates that are close to the state values.

VI. Conclusion

In this paper, a distributed Kalman filtering approach is used to estimate oscillation modes in power systems that have false measurements and even under attacks. We proposed two modified distributed Kalman filtering algorithms, which incorporate the notion of trust. The first algorithm uses the trust notion to quantify the estimation errors in terms of observation and measurement noise. The second algorithm interpreted trust in terms of security. The low trusted PMUs are excluded from the estimation procedure. Via simulations, we compared our trust based algorithms with the original distributed Kalman filtering algorithm and showed that our modified algorithms perform better when there are large noises in the system and are able to detect malicious data.

Acknowledgement

Research partially supported by the Defense Advanced Research Projects Agency (DARPA) under award number 013641-001 for the Multi-Scale Systems Center (MuSyC), through the FRCP of SRC and DARPA. The authors acknowledge useful discussions and suggestions received through their participation in the EU project VIKING.

References

[1] V. Vittal, ”Consequence and Impact of Electric Utility Industry Restructuring on Transient Stability and Small-signal Stability Analysis”, *Proc. IEEE*, vol. 88, no. 2, pp. 196-207, Feb. 2000.

[2] M. Klein, G. J. Rogers and P. Kundur, ”A Fundamental Study of Interarea Oscillations in Power Systems”, *IEEE Trans. Power Syst.*, vol. 6, no. 3, pp. 914-921, Aug. 1991.

[3] G. Ledwich and E. Palmer, ”Modal estimates from normal operation of power systems”, *2000 IEEE Power Eng. Soc. Winter Meeting. Conf. Proc.*, Singapore, 2000, vol. 2, pp. 15271531.

[4] K. C. Lee and K.P. Poon, ”Analysis of power system dynamic oscillation with beat phenomenon by Fourier transformation”, *IEEE Trans. Power Syst.*, vol. 5, no. 1, pages 148-153, 1990.

[5] R. Olfati-Saber, ”Distributed Kalman Filtering for Sensor Networks”, *Proceedings of the* 46^{th} *IEEE Conference on Decision and Control*, pages 5492-5498, 2007

[6] C. De Kerchove and P. Van Doren, ”Iterative filtering for a dynamical reputation system”, arXiv, 2007.