Distributed Fault Detection with Sensor Networks using Pareto-Optimal Dynamic Estimation Method

Yilun Zhou, Francesca Boem, Carlo Fischione and Thomas Parisini

***Abstract*—In this paper, a distributed method for fault detection using sensor networks is proposed. Each sensor communicates only with neighboring nodes to compute locally an estimate of the state of the system to monitor. A residual is defined and suitable stochastic thresholds are designed, allowing to set the parameters so to guarantee a maximum false alarms probability. The main novelty and challenge of the proposed approach consists in addressing the individual correlations between the state, the measurements, and the noise components, thus significantly generalising the estimation methodology compared to previous results. No assumptions on the probability distribution family are needed for the noise variables. Simulation results show the effectiveness of the proposed method, including an extensive sensitivity analysis with respect to fault magnitude and measurement noise.**

I. Introduction

In recent years, the research on large-scale and/or distributed systems has become always more important, due to the relevant applications involving collaboration and communication among subsystems. In order to satisfy the growing demand for intelligent and reliable operations in distributed systems, distributed fault detection approaches have gained significant attention thanks to their advantages on feasibility, reliability and scalability. Several distributed model-based fault detection methods have been developed based on observers ([1], [2], [3], [4], [5], [6]). An interesting instrument for the monitoring of large-scale systems can be represented by sensor networks. There exist many works in the literature dealing with distributed estimation methods using sensor networks (see, as examples [7], [8], [9]). While many papers deal with the problem of sensor fault detection for sensor networks ([10], [11] as example), the use of sensor networks for the monitoring of processes is a relatively new field of research. An exception is [12], where a distributed fault detection and isolation technique is designed, relying on decentralized Kalman filtering and based on a distributed hypothesis testing method. [13] uses sensor networks for distributed estimation, but the fault detection decision is centralized. In this paper, a sensor network monitors a system characterized by stochastic uncertainties using noisy measurements of the state. A residual signal is defined and probabilistic thresholds are designed according to a chosen maximum false alarm probability. The proposed fault detection scheme takes advantage of the distributed estimation approach over sensor networks introduced in [14]. The advantage of using this distributed prediction method based on a Pareto optimization framework is that filtering and prediction weights are computed at each time step so to minimize both the bias and the variance of the prediction error. Moreover, thanks to the way it is designed, it is possible to estimate at each time step the mean and the variance of the prediction error, thus allowing to obtain suitable thresholds for some residual signals with known confidence levels and guaranteed maximum rate of false alarms. Furthermore, since the distributed Pareto estimator does not require to know the distribution of process and measurements noises (only mean and variance), it can be applied to any noise distribution (having mean and variance). In this paper, differently from [14], the goal is fault detection. In [15], a distributed fault detection scheme is proposed based on the distributed filtering method presented in [16], where only the filtering weights are computed by solving a Pareto optimization problem. In this paper, differently from [15], we take into account the correlations existing between different state variables and process noise components. This represents a more challenging framework and a much more complicated analysis in order to compute the stochastic thresholds. Moreover, in addition to [15], both the filtering and the prediction weights are jointly optimized. In fact, each node in the sensor network acts as a local estimation and detection agent, where the local stochastic thresholds are updated at every iteration. The rest of the paper is organized as follows. In Section II we introduce the problem formulation. In Section III, the distributed state estimation method is briefly presented. Then the distributed fault detection algorithm is illustrated and analyzed in Section IV. Simulation results are given in Section V and some concluding remarks in Section VI.

Notation Given a stochastic variable x, we define $\mathbb{E}x$ its expected value. By $\mathbf{1}_m$ and I_m we denote the m elements vector $(1,\dots,1)^\top$ and the identity matrix of dimension $m \times m$, respectively. Finally, the operator $\otimes$ represents the Kronecker product and $\circ$ the component-by-component product.

II. Problem Formulation

The process whose state dynamics we want to monitor is modeled as

$$x(t+1) = Ax(t) + \bar{w}(t) + \tilde{w}(t) + \beta(t-t_0)\phi(x(t),t), \quad (1)$$

Y. Zhou and F. Boem are with the Dept. of Electrical and Electronic Engineering at Imperial College London, UK. (yilun.zhou11@imperial.ac.uk, f.boem@imperial.ac.uk)

C. Fischione is with ACCESS Linnaeus Center, KTH Royal Institute of Technology, Stockholm, Sweden. (carlofi@kth.se)

T. Parisini is with the Dept. of Electrical and Electronic Engineering at Imperial College London, UK, and with Dept. of Engineering and Architecture at University of Trieste, Italy (t.parisini@gmail.com)

This work has been conducted as part of the research project *Stability and Control of Power Networks with Energy Storage* (STABLE-NET) which is funded by the RCUK‘s Energy Programme (contract no: EP/L014343/1).

where t is the discrete time, $x \in \mathbb{R}^m$ denotes the state vector, $\bar{w} \in \mathbb{R}^m$ represents any known time-varying modeling error including possibly some known process disturbances and modeling errors, while $\tilde{w}(t)$ models uncertainties and process noises, and $\beta(t-t_0)\phi(x(t),t)$ represents the dynamics of a process fault occurring at the unknown time t_0, with $\beta(t-t_0)$ being the time profile of the fault function.

Assumption 1: We assume that $\tilde{w}(t)$ is a zero-mean process with covariance matrix $\Sigma_w(t)$.

A *sensor network*, composed of n nodes, monitors system (1) by taking noisy measurements, so that, for each sensor node i, with $i = 1, \ldots, n$, we have:

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

where $y_i \in \mathbb{R}^m$ denotes the measurements vector taken by sensor i and $v_i \in \mathbb{R}^m$ is the measurement noise.

Assumption 2: We assume that $v_i(t)$ is a zero-mean[1] measurement noise, with covariance matrix Σ_{v_i}. The measurement noises in different nodes are not correlated with each other.

Each node communicates with neighbors by means of a communication network modeled by an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{N}_i = \{j \in \mathcal{V} : (j,i) \in \mathcal{E}\} \cup \{i\}$ is the set of neighbors of node $i \in \mathcal{V}$ plus the node itself. Each node of the network observes the entire state vector and exchanges measurements and predictions with the neighboring nodes, as presented in [14]. The computed state prediction is then used for fault detection. The objective of the proposed approach is to monitor the state dynamics of (1), taking into account the modeling and measurement uncertainty. Moreover, we want to guarantee some properties on the fault detection performance in terms of sensitivity and false alarms rate. As already mentioned, this analysis is rather challenging due to the fact that the state and noises components may be individually correlated. In the next section, for the reader's convenience, the distributed state prediction method introduced in [14] is briefly recalled.

III. Distributed State Prediction

Each node i of the sensor network implements a two steps estimator: a *filtering* phase and a *prediction* step. First, by communicating only with neighboring nodes, the estimator filters the measurements noise by computing a linear combination of its own and neighboring measurements and predictions:

$$\bar{x}_i(t) = \sum_{j \in \mathcal{N}_i} k_{i,j}(t)\hat{x}_j(t) + \sum_{j \in \mathcal{N}_i} h_{i,j}(t) y_j(t), \tag{3}$$

where $k_{i,j}(t)$ and $h_{i,j}(t)$ are some properly designed time-varying filtering weights.

Secondly, each node implements a classical model-based prediction, using filtered measurements (obtained from the previous step), instead of the observed measurements. The one-step-ahead prediction, considering healthy dynamics, is

$$\hat{x}_i(t+1) = A\bar{x}_i(t) + \bar{w}(t) + \lambda'_i(t)(\hat{x}_i(t) - \bar{x}_i(t)), \tag{4}$$

where $\lambda'_i(t) \in \mathbb{R}^{m \times m}$ is a time-varying matrix collecting the properly designed prediction parameters. The term $\lambda'_i(t)(\hat{x}_i(t) - \bar{x}_i(t))$ represents a correction of the previous prediction errors. To simplify the methodology and its analysis, we denote the filter parameter matrix $\lambda'_i(t) = \lambda_i(t)A$, where $\lambda_i(t)$ is the time varying diagonal matrix that we design. Filtering weights $k_{i,j}(t)$ and $h_{i,j}(t)$ and prediction parameters $\lambda_i(t)$ are computed at each time step by each node in a distributed way so to jointly minimize the bias and the variance of the prediction error $\hat{E}_i(t) = \hat{x}_i(t) - x(t)$, for all $i = 1, \ldots, n$, in a Pareto optimization framework.

We rewrite Eqs. (3) and (4) as

$$\begin{aligned} \bar{x}_i(t) &= \kappa_i(t)\hat{x}_i^{\text{reg}}(t) + \eta_i(t) y_i^{\text{reg}}(t) \\ \hat{x}_i(t+1) &= A\bar{x}_i(t) + \bar{w}(t) + \lambda_i(t)(A\hat{x}_i(t) - A\bar{x}_i(t)), \end{aligned} \tag{5}$$

where $\hat{x}_i^{\text{reg}}$ and y_i^{reg} are two column vectors collecting the prediction vectors and the measurements vectors respectively available at node i, ordered according to their indexes: $\hat{x}_i^{\text{reg}} = (\hat{x}_{i_1}^\top, \ldots, \hat{x}_{i_{N_i}}^\top)^\top$, $i_1 < \ldots < i_{N_i}$, and $y_i^{\text{reg}} = (y_{i_1}^\top, \ldots, y_{i_{N_i}}^\top)^\top$, $i_1 < \ldots < i_{N_i}$, with $N_i = |\mathcal{N}_i|$ denoting the number of neighbors of i plus i itself. Moreover, $\kappa_i(t)$ and $\eta_i(t)$ are the time varying $1 \times N_i$ block matrices collecting the diagonal matrices $k_{i,j}(t)$ and $h_{i,j}(t)$ with $j \in \mathcal{N}_i$, respectively.

The optimal time-varying weights κ_i, η_i and λ_i at each time t are the solutions of the following Pareto optimization problem:

$$\min_{\kappa_i, \eta_i, \lambda_i} \ \text{tr}\ [\rho_i B_i^2 + (1-\rho_i)V_i] \tag{6a}$$

$$\text{s.t.}\ (\kappa_i(t) + \eta_i(t))\mathbf{1}_{m_{N_i}} = \mathbf{1}_m, \tag{6b}$$

$$\text{convergence conditions}, \tag{6c}$$

where $m_{N_i} = m \times N_i$, $0 \leq \rho_i \leq 1$ is the Pareto parameter, $B_i = \mathbb{E}\hat{E}_i(t+1)$ is the expected value of prediction error, $V_i = \mathbb{E}[(\hat{E}_i(t+1) - \mathbb{E}\hat{E}_i(t+1))(\hat{E}_i(t+1) - \mathbb{E}\hat{E}_i(t+1))^\top]$ is the variance of prediction error. The constraints represent some local conditions on the decisional variables, derived in [14] to guarantee the convergence of the prediction error.

We define $\kappa_i^{\text{vec}}(t) = \kappa_i^\top(t)\mathbf{1}_m$, collecting all the decisional variables of the diagonals of $\kappa_i(t)$ into a vector. Similarly, $\eta_i^{\text{vec}}(t) = \eta_i^\top(t)\mathbf{1}_m$, $\lambda_i^{\text{vec}}(t) = \lambda_i^\top(t)\mathbf{1}_m$ and $\nu_i^E(t) = [\mathbf{1}_{m_{N_i}} \nu_i^\top] \circ (\mathbf{1} \otimes I)\mathbf{1}_m$, collecting N_i times vector ν_i, which is a Lagrangian dual variable. By analyzing the Karush-Kuhn-Tucker conditions related to problem 6, in [14] the optimal weights are derived as

$$\kappa_i^{\text{vec}} = -\left(2M_i \circ \mathbf{D}_i\right)^{-1} \nu_i^E\,, \tag{7a}$$

$$\eta_i^{\text{vec}} = -\left(2S_i \circ \mathbf{D}_i\right)^{-1} \nu_i^E\,, \tag{7b}$$

$$\begin{aligned} \nu_i = &- 2(I_{m \times m_{N_i}}((M_i \circ \mathbf{D}_i)^{-1} + \\ &(S_i \circ \mathbf{D}_i)^{-1})I_{m \times m_{N_i}}^\top)^{-1}\mathbf{1}_m\,, \end{aligned} \tag{7c}$$

$$\begin{aligned} \lambda_i^{\text{vec}} = &\left(2(M_i^{\text{loc}} + \kappa_i M_i \kappa_i^\top + \eta_i S_i \eta_i^\top) \circ (A^\top A)\right)^{-1} \\ &\left(\text{diag}\left(2(\kappa_i M_i \kappa_i^\top + \eta_i S_i \eta_i^\top) A^\top A\right)\right)\,, \end{aligned} \tag{7d}$$

with $\mathbf{D}_i = \mathbf{1}_{N_i}\mathbf{1}_{N_i}^\top \otimes D_i$ and $D_i := (I - \lambda_i)^\top A^\top A(I - \lambda_i)$,

$$M_i(t, \rho_i) = \rho_i \mathbb{E}\hat{\epsilon}_i(t)\mathbb{E}\hat{\epsilon}_i^\top(t) + (1-\rho_i)\Gamma_{\hat{\epsilon}_i}(t),$$

[1] The extension to the non-zero mean case is trivial.

$$M_i(t,\rho_i)^{\mathrm{loc}} = \rho_i \mathbb{E}\hat{E}_i(t)\mathbb{E}\hat{E}_i^\top(t) + (1-\rho_i)\Gamma_{\hat{E}_i}(t),$$
$$S_i(t,\rho_i) = (1-\rho_i)\Sigma_{v_{\hat{\epsilon}_i}}(t)\,.$$

$\hat{\epsilon}_i$ collects the prediction error vectors $\hat{E}_j$, $j \in \mathcal{N}_i$, available at node i and $\Sigma_{v_{\hat{\epsilon}_i}}(t)$ is a block matrix collecting on the diagonal the measurement noise covariance matrices $\Sigma_{v_j}(t)$, $j \in \mathcal{N}_i$, related to the measurements communicated at node i, both ordered according to the indexes, and

$$\Gamma_{\hat{\epsilon}_i}(t) = \mathbb{E}[(\hat{\epsilon}_i(t) - \mathbb{E}\hat{\epsilon}_i(t))(\hat{\epsilon}_i(t) - \mathbb{E}\hat{\epsilon}_i(t))^\top]$$

is the covariance matrix of the prediction errors. The Pareto parameter ρ_i is set locally.

In [14], a computational method is proposed (and also an analytical solution of an approximated problem is proposed) so that each node can compute locally at each time step the filtering and prediction weights. Moreover, it is possible to locally compute the estimates $\hat{m}_{\epsilon_i}$ and $\hat{\Gamma}_i$ respectively of the expected value $\mathbb{E}\hat{\epsilon}_i$ and variance $\Gamma_{\hat{\epsilon}_i}$ using at each time step the new samples of local measurements y_i^{reg} and state predictions $\hat{x}_i^{\mathrm{reg}}$ (see Algorithm 1 for details about the estimation from samples).

IV. Distributed Fault Detection

In the previous section, we have introduced the one-step ahead prediction method. Based on those results, in this section, a distributed fault detection scheme is designed by analyzing the dynamics of a stochastic residual. Hence, the fault detection probabilistic thresholds are derived for the generated residual.

A. Local Residual Analysis

In this subsection we introduce the residual signal and we analyze its stochastic features. The local residual is defined as $r_i(t) = \bar{x}_i(t) - \hat{x}_i(t)$. Its dynamics can be described as:

$$r_i(t+1) = \lambda_i(t)Ar_i(t) + \bar{e}_i(t+1) - A\bar{e}_i(t) + \tilde{w}(t)\,, \quad (8)$$

where $\bar{e}_i(t) = \bar{x}_i(t) - x(t)$ is the local filtering error.

We define the stochastic part of the residual $r_i(t)$ as

$$\chi_i(t) = \bar{e}_i(t) - A\bar{e}_i(t-1) + \tilde{w}(t-1), \quad (9)$$

which is a stochastic variable at time t. We then analyze the stochastic features of the residual. The mean of χ_i is

$$\mu_{\chi_i}(t) = \mathbb{E}\chi_i(t) = \mathbb{E}\bar{e}_i(t) - A\mathbb{E}\bar{e}_i(t-1)\,. \quad (10)$$

Since, using the definition, the filtering error $\bar{e}_i(t)$ can be computed as

$$\bar{e}_i(t) = \kappa_i(t)\hat{\epsilon}_i(t) + \eta_i(t)v_{\epsilon_i}(t), \quad (11)$$

where v_{ϵ_i} collects the measurement noise vectors v_j, $j \in \mathcal{N}_i$, then we derive $\mathbb{E}\bar{e}_i(t) = \kappa_i(t)\mathbb{E}\hat{\epsilon}_i(t)$. We need the following equation:

$$\begin{aligned}\mathbb{E}\hat{E}_i(t+1) =& A(I-\lambda_i(t))\kappa_i(t)\mathbb{E}\hat{\epsilon}_i(t) + A\lambda_i(t)\mathbb{E}\hat{E}_i(t) \\ =& A[(I-\lambda_i(t))\kappa_i(t) + \lambda_i^0(t)]\mathbb{E}\hat{\epsilon}_i(t)\,, \quad (12)\end{aligned}$$

where λ_i^0 is a $1 \times N_i$ block matrix with only one-zero component λ_i, following the same index as in the set $\mathcal{N}_i$.

We require the following assumption:

Assumption 3: The measurement noise v and process disturbances w are not correlated.

We now derive the variance of the stochastic variable χ_i.

Lemma 1: The variance of $\chi_i(t)$ defined in (9) is

$$\begin{aligned}\sigma^2_{\chi_i}(t) =& \Gamma_{\bar{e}_i}(t) + A\Gamma_{\bar{e}_i}(t-1)A^\top + \Sigma_w(t-1) \\ &- \mathrm{Cov}(\bar{e}_i(t), \bar{e}_i(t-1))A^\top - (\mathbf{1}_{N_i}^\top \otimes \Sigma_w(t-1))\kappa_i^\top \\ &- A\mathrm{Cov}(\bar{e}_i(t-1), \bar{e}_i(t)) - \kappa_i(\mathbf{1}_{N_i} \otimes \Sigma_w(t-1)), \quad (13)\end{aligned}$$

where $\Gamma_{\bar{e}_i}(t) = \mathbb{E}[(\bar{e}_i(t) - \mathbb{E}\bar{e}_i(t))(\bar{e}_i(t) - \mathbb{E}\bar{e}_i(t))^\top]$ denotes the variance of local filtering error and can be computed as

$$\Gamma_{\bar{e}_i}(t) = \kappa_i(t)\Gamma_{\hat{\epsilon}_i}(t)\kappa_i(t)^\top + \eta_i(t)\Sigma_{v_{\hat{\epsilon}_i}}\eta_i(t)^\top. \quad (14)$$

Due to space constraints, the proof is omitted.

The local autocorrelation matrix is defined as

$$\begin{aligned}&\mathrm{Cov}(\bar{e}_i(t), \bar{e}_i(t-1)) = \\ &\quad \mathbb{E}[(\bar{e}_i(t) - \mathbb{E}\bar{e}_i(t))(\bar{e}_i(t-1) - \mathbb{E}\bar{e}_i(t-1))^\top]. \quad (15)\end{aligned}$$

In the next subsection, we explain how to compute it locally.

B. Local autocorrelation matrix

$\mathrm{Cov}(\bar{e}_i(t), \bar{e}_i(t-1))$ is the i^{th} diagonal component of global covariance matrix $\mathrm{Cov}(\bar{e}(t), \bar{e}(t-1))$. To facilitate the analysis, the distributed estimation formulation can be expressed in an extended vector form including all the nodes in the sensor network:

$$\begin{aligned}\bar{x}(t) &= K(t)\hat{x}(t) + H(t)y(t) \\ \hat{x}(t+1) &= A_E\bar{x}(t) + \bar{w}_E(t) + \lambda'(t)(\hat{x}(t) - \bar{x}(t)) \quad (16) \\ y(t) &= x_E(t) + v(t)\,,\end{aligned}$$

where $\bar{x}$, $\hat{x}$, y and v are column vectors $\in \mathbb{R}^{mn\times 1}$ collecting the local vectors $\bar{x}_i$, $\hat{x}_i$, y_i and v_i, respectively, with $i = 1, \ldots, n$; $A_E = I_n \otimes A$ is a diagonal block matrix, with each block on the diagonal equal to A; $\bar{w}_E = \mathbf{1}_n \otimes \bar{w}$ and $x_E = \mathbf{1}_n \otimes x$ are both column vectors of appropriate length, where the process disturbance vector $\bar{w}$ and x are repeated n times, respectively. $K(t)$ and $H(t)$ are block matrices, where each $(i,j)^{\mathrm{th}}$ block, with $i, j = 1, \ldots, n$, collects on the diagonal the weights with which the j^{th} sensor weights measurements or estimates components developed by the j^{th} node, if they are neighbors; the block is a null matrix if i and j are not neighboring nodes. $\lambda'(t) = A_E\lambda(t)$ is a block matrix, collecting the matrices λ_i', $i = 1, \ldots, n$, on the diagonal.

Therefore, the global filtering error $\bar{e}(t) = \bar{x}(t) - x_E(t)$ and the global prediction error $\hat{E}(t) = \hat{x}(t) - x_E(t)$ can be computed as follows:

$$\bar{e}(t) = K(t)\hat{E}(t) + H(t)v(t), \quad (17)$$

$$\begin{aligned}\hat{E}(t+1) =& A_E\left[(I - \lambda(t))K(t) + \lambda(t)\right]\hat{E}(t) \\ &+ A_E(I - \lambda(t))H(t)v(t) - \tilde{w}_E(t)\,, \quad (18)\end{aligned}$$

with $\tilde{w}_E = \mathbf{1}_n \otimes \tilde{w}$, and the first constraint of problem (6) is used. Let us compute mean and variance of the global estimation and prediction errors.

$$\mathbb{E}\bar{e}(t) = K(t)\mathbb{E}\hat{E}(t), \quad (19)$$

$$\mathbb{E}\hat{E}(t+1) = A_E\left[(I-\lambda(t))K(t)+\lambda(t)\right]\mathbb{E}\hat{E}(t). \qquad (20)$$

Since the local variance can be computed as

$$\begin{aligned}&\mathbb{E}[(\hat{E}_i(t+1)-\mathbb{E}\hat{E}_i(t+1))(\hat{E}_i(t+1)-\mathbb{E}\hat{E}_i(t+1))^\top]\\ &= W_i^{(1)}(t)\Gamma_{\hat{\epsilon}_i}(t)W_i^{(1)}(t)^\top + W_i^{(2)}(t)\Sigma_{v_{\hat{\epsilon}_i}}W_i^{(2)}(t)^\top + \Sigma_w(t),\end{aligned}$$

where $W_i^{(1)}(t) = A[(I-\lambda_i(t))\kappa_i(t)+\lambda_i^0(t)]$ and $W_i^{(2)}(t) = A(I-\lambda_i(t))\eta_i(t)$, then, the global variance matrix $\Gamma^{\hat{E}}(t+1)$ can be derived:

$$\begin{aligned}&\mathbb{E}\left[(\hat{E}(t+1)-\mathbb{E}\hat{E}(t+1))(\hat{E}(t+1)-\mathbb{E}\hat{E}(t+1))^\top\right]\\ &= W^{(1)}(t)\Gamma^{\hat{E}}(t)W^{(1)}(t)^\top + W^{(2)}(t)\Sigma^v(t)W^{(2)}(t)^\top + \Sigma_{w_E}(t),\end{aligned} \qquad (21)$$

with $\Sigma^v(t) = \mathbb{E}\left[(v(t)-\mathbb{E}v(t))(v(t)-\mathbb{E}v(t))^\top\right]$, $\Sigma_{w_E}(t) = \mathbb{E}\left[\tilde{w}_E(t)-\mathbb{E}\tilde{w}_E(t)\right]^2$,

$$W^{(1)}(t) = A_E\left[(I-\lambda(t))K(t)+\lambda(t)\right],$$

$$W^{(2)}(t) = A_E(I-\lambda(t))H(t),$$

where $W^{(1)}(t)$, $W^{(2)}(t)$, $\Sigma^v(t)$ and $\Sigma_{w_E}(t)$ are $n \times n$ block matrices, being $\mathrm{Cov}(\hat{E}(t), v(t)) = 0$, $\mathrm{Cov}(\hat{E}(t), \tilde{w}_E(t)) = 0$ and $\mathrm{Cov}(v(t), \tilde{w}_E(t)) = 0$, using Assumption 3. Given (17) – (20), we have:

$$\begin{aligned}\bar{e}(t)-\mathbb{E}\bar{e}(t) =& K(t)(\hat{E}(t)-\mathbb{E}\hat{E}(t)) + H(t)v(t),\\ \hat{E}(t)-\mathbb{E}\hat{E}(t) =& W^{(1)}(t-1)(\hat{E}(t-1)-\mathbb{E}\hat{E}(t-1))\\ &+ W^{(2)}(t-1)v(t-1) - \tilde{w}_E(t-1).\end{aligned}$$

Hence, the global covariance matrix, collecting on the diagonal the matrices (15) needed to compute (13), is

$$\begin{aligned}\mathrm{Cov}(\bar{e}(t), \bar{e}(t-1)) =& K(t)W^{(1)}(t-1)\Gamma^{\hat{E}}(t-1)K(t-1)^\top\\ &+ K(t)W^{(2)}(t-1)\Sigma^v(t-1)H(t-1)^\top.\end{aligned}$$

Proposition 4.1: Let us define two block matrices $P(t) = W^{(1)}(t)\Gamma^{\hat{E}}(t)$ and $Q(t) = W^{(2)}(t)\Sigma^v(t)$. The covariance matrix $\mathrm{Cov}(\bar{e}_i(t), \bar{e}_i(t-1))$ can be computed locally as

$$\begin{aligned}&\mathrm{Cov}(\bar{e}_i(t), \bar{e}_i(t-1))\\ &= \sum_{j'\in\mathcal{N}_i}\sum_{j\in\mathcal{N}_i} k_{i,j}(t)P_{j,j'}(t-1)k_{i,j'}(t-1)^\top\\ &\quad + \sum_{j'\in\mathcal{N}_i}\sum_{j\in\mathcal{N}_i} k_{i,j}(t)Q_{j,j'}(t-1)h_{i,j'}(t-1)^\top,\end{aligned} \qquad (22)$$

where subscripts denote the indexes of the blocks in the block matrices.

Due to space constraints, the proof is omitted. At each time step t all the variables can be computed locally just communicating with neighboring nodes. The needed information is collected from the set of nodes $\mathcal{N}_i^{\mathrm{all}} = \bigcup_{l\in\mathcal{N}_i}\mathcal{N}_l$, denoting the neighbours of neighbours of node i, in two time steps, thus involving only local communication.

C. *The fault detection thresholds*

In this section, the objective is to design the time-varying thresholds that bound with a guaranteed probability each residual component when the system is healthy.

Proposition 4.2: Let us consider the following time-varying upper and lower thresholds for the residual $r_i(t+1)$:

$$\begin{aligned}\bar{r}_i^{(k)+}(t+1) =& \lambda_i(t)\bar{r}_i^{(k)+}(t) + \mu_{\chi_i}^{(k)}(t+1) + \alpha\sigma_{\chi_i}^{(k)}(t+1)\\ \bar{r}_i^{(k)-}(t+1) =& \lambda_i(t)\bar{r}_i^{(k)-}(t) + \mu_{\chi_i}^{(k)}(t+1) - \alpha\sigma_{\chi_i}^{(k)}(t+1),\end{aligned}$$

where $\chi_i^{(k)}$ and $\mu_{\chi_i}^{(k)}$ are the k^{th} component of vectors χ_i and μ_{χ_i}, defined in (10), respectively; $\sigma_{\chi_i}^{(k)}$ is the square root of the k-th diagonal component of matrix $\sigma_{\chi_i}^2$ in (13). Hence, there exists a ‘tube’ between the two thresholds

$$\bar{r}_i^{(k)-}(t) \leq r_i^{(k)}(t) \leq \bar{r}_i^{(k)+}(t). \qquad (23)$$

where, in healthy conditions, the residual lies with probability greater than $1-1/\alpha^2$.

The proof is based on Chebyshev‘s inequality and is omitted due to space constraints. Thanks to the way they are designed, the proposed thresholds guarantee that the false-alarms rate is lower than $1/\alpha^2$. According to (23), a fault is *detected* by a node $i \in \{1,\ldots,n\}$ when at least one component $r_i^{(k)}$ of the residual r_i exceeds the corresponding thresholds $\bar{r}_i^{(k)+}$ or $\bar{r}_i^{(k)-}$.

The α parameter can be set according to the wanted maximum false alarms probability, taking into account the trade-off existing between robustness and sensitivity of the thresholds. Less conservative thresholds can be designed if we assume the noises $\tilde{w}$ and v_i are normally distributed.

D. *The algorithm*

The proposed distributed fault detection method is illustrated by Algorithm 1. In this paper the Pareto parameter ρ is set to 0.5. In [16] it is explained how to alternatively compute it at each step in order to reach the knee-point of the Pareto optimization. $\hat{m}_{\bar{e}_i}$ denotes the local estimate of $\mathbb{E}\bar{e}_i$. Roman numbers indicate the steps where information communication is required.

V. Simulation Results

In this section, some simulation results are presented to illustrate the fault detection performance. We consider a 15-nodes sensor network monitoring the four-tanks system introduced in [17]. LQR control is applied to regulate the levels in the tanks. The nonlinear system is linearized around the operation points. Zero-order hold method is used to discretize the system with a sample time $\mathrm{Ts} = 1\mathrm{s}$. Given $Q = I_4$ and $R = 2I_4$ and using the parameters as in [17], the input $u(t)$ is generated so that system matrix A in (1) is

$$A = \begin{bmatrix} 0.9430 & -0.0031 & 0.0262 & -0.0118\\ -0.0036 & 0.9579 & 0.0121 & 0.0213\\ -0.0025 & -0.0233 & 0.9500 & -0.0084\\ -0.0153 & -0.0010 & -0.0053 & 0.9629 \end{bmatrix}.$$

Algorithm 1 Fault detection algorithm for node i

```
$t := 0\,,\; N_i := |\mathcal{N}_i|\;,\; N_i^{\mathrm{all}} := \left|\mathcal{N}_i^{\mathrm{all}}\right|\;,\; \rho_i := 0.5$
$\hat{m}_{\epsilon_i}^{\mathrm{all}}(0) := \mathbf{0}$
$\hat{\Gamma}_i^{\mathrm{all}}(0) := \Sigma_{v_{\hat{\epsilon}_i}^{\mathrm{all}}}$    ▷ I
Extract $\hat{m}_{\epsilon_i}(0)$ from $\hat{m}_{\epsilon_i}^{\mathrm{all}}(0)$
Extract $\hat{\Gamma}_i(0)$ from $\hat{\Gamma}_i^{\mathrm{all}}(0)$
$\bar{x}_i(0) := y_i(0)$
$\hat{x}_i(1) := A\bar{x}_i(0)$
repeat
    $t := t+1$
    Collect $\hat{\mathbf{x}}_i^{\mathrm{reg}}(t) := (\hat{x}_{i_1}(t),\dots,\hat{x}_{i_{N_i}}(t))^\top$ and $\mathbf{y}_i^{\mathrm{reg}}(t) := (y_{i_1}(t),\dots,y_{i_{N_i}}(t))^\top$ where $\{i_1,\dots,i_{N_i}\}\in\mathcal{N}_i$    ▷ II
    $\hat{\epsilon}_i(t) := \frac{\hat{\mathbf{x}}_i}{1+\nu} - \frac{\nu\mathbf{1}^\top\hat{\mathbf{x}}_i+(1+\nu)\mathbf{1}^\top\mathbf{y}_i}{N_i(1+2\nu)(1+\nu)}\mathbf{1}$
    $\hat{m}_{\epsilon_i}(t) := \frac{t-1}{t}\hat{m}_{\epsilon_i}(t-1) + \frac{1}{t}\hat{\epsilon}_i(t)$
    $\hat{\Gamma}_i(t) := \frac{t-1}{t}\hat{\Gamma}_i(t-1) + \frac{1}{t}(\hat{\epsilon}_i(t)-\hat{m}_{\epsilon_i}(t))(\hat{\epsilon}_i^{\mathrm{all}}(t)-\hat{m}_{\epsilon_i}(t))^\top$
    $S_i := (1-\rho_i)\Sigma_{v_{\hat{\epsilon}_i}}(t)$    ▷ III
    $M_i := \rho_i(\hat{m}_{\epsilon_i}(t)\hat{m}_{\epsilon_i}^\top(t)) + (1-\rho_i)\hat{\Gamma}_i(t)$
    Compute $\kappa_i(t)\,,\eta_i(t)\,,\lambda_i(t)$ as in Eqs. (7)
    $\bar{x}_i(t) = \kappa_i(t)\hat{\mathbf{x}}_i^{\mathrm{reg}}(t)(t) + \eta_i(t)\mathbf{y}_i^{\mathrm{reg}}(t)$
    $\hat{x}_i(t+1) = A\bar{x}_i(t) + \bar{w}(t) + \lambda_i(t)(A\hat{x}_i(t) - A\bar{x}_i(t))$
    $r_i(t) := \bar{x}_i(t) - \hat{x}_i(t)$
    $\hat{m}_{\bar{e}_i}(t) := \kappa_i(t)\hat{m}_{\epsilon_i}(t)$
    $\Gamma_{\bar{e}_i}(t) := \kappa_i(t)\hat{\Gamma}_i(t)\kappa_i(t)^\top + \eta_i(t)\Sigma_{v_{\hat{\epsilon}_i}}\eta_i(t)^\top$
    if $t = 1$ then
        $\bar{r}_i^+(1) := r_i(1) + 2|r_i(1)|$
        $\bar{r}_i^-(1) := r_i(1) - 2|r_i(1)|$
    else
        Collect $\hat{\mathbf{x}}_i^{\mathrm{all}}(t-1) := (\hat{x}_{i_1}(t-1),\dots,\hat{x}_{i_{N_i^{\mathrm{all}}}}(t-1))^\top$ and $\mathbf{y}_i^{\mathrm{all}}(t-1) := (y_{i_1}(t-1),\dots,y_{i_{N_i^{\mathrm{all}}}}(t-1))^\top$ where $\{i_1,\dots,i_{N_i^{\mathrm{all}}}\}\in\mathcal{N}_i^{\mathrm{all}}$    ▷ IV
        $\hat{\epsilon}_i^{\mathrm{all}}(t-1) := \frac{\hat{\mathbf{x}}_i^{\mathrm{all}}(t-1)}{1+\nu} - \frac{\nu\mathbf{1}^\top\hat{\mathbf{x}}_i^{\mathrm{all}}(t-1)+(1+\nu)\mathbf{1}^\top\mathbf{y}_i^{\mathrm{all}}(t-1)}{N_i^{\mathrm{all}}(1+2\nu)(1+\nu)}\mathbf{1}$
        $\hat{m}_{\epsilon_i}^{\mathrm{all}}(t-1) := \frac{t-2}{t-1}\hat{m}_{\epsilon_i}^{\mathrm{all}}(t-2) + \frac{1}{t-1}\hat{\epsilon}_i^{\mathrm{all}}(t-1)$
        $\hat{\Gamma}_i^{\mathrm{all}}(t-1) := \frac{t-2}{t-1}\hat{\Gamma}_i^{\mathrm{all}}(t-2) + \frac{1}{t-1}(\hat{\epsilon}_i^{\mathrm{all}}(t-1)-\hat{m}_{\epsilon_i}^{\mathrm{all}}(t-1))(\hat{\epsilon}_i^{\mathrm{all}}(t-1)-\hat{m}_{\epsilon_i}^{\mathrm{all}}(t-1))^\top$
        Compute $P_{i,i'}(t-1)$ with $i'\in\mathcal{N}_i^{\mathrm{all}}$ and send to $j\in\mathcal{N}_i$    ▷ V
        Compute $Q_{i,i'}(t-1)$ with $i'\in\mathcal{N}_i^{\mathrm{all}}$ and send to $j\in\mathcal{N}_i$    ▷ VI
        Collect $P_{j,j'}(t-1)\,,\;Q_{j,j'}(t-1)$ with $j,j'\in\mathcal{N}_i$
        Compute $\sigma_{\chi_i}^2(t)$ as in (13)
        $\mu_{\chi_i}(t) = \hat{m}_{\bar{e}_i}(t) - A\hat{m}_{\bar{e}_i}(t-1)$
        $\bar{r}_i^+(t) := \lambda_i(t-1)\bar{r}_i^+(t-1) + \mu_{\chi_i}(t) + \alpha\sigma_{\chi_i}(t)$
        $\bar{r}_i^-(t) := \lambda_i(t-1)\bar{r}_i^-(t-1) + \mu_{\chi_i}(t) - \alpha\sigma_{\chi_i}(t)$
    end if
until $\exists\, k : r_i^{(k)}(t) > r_i^{(k)+}(t) \;\vee\; r_i^{(k)}(t) < r_i^{(k)-}(t)$
A fault is detected at time $t$.
```

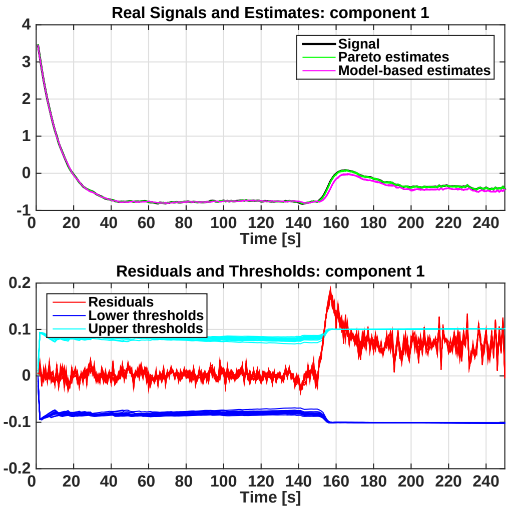

Fig. 1. Estimates, Residuals and Thresholds for state component 1 .

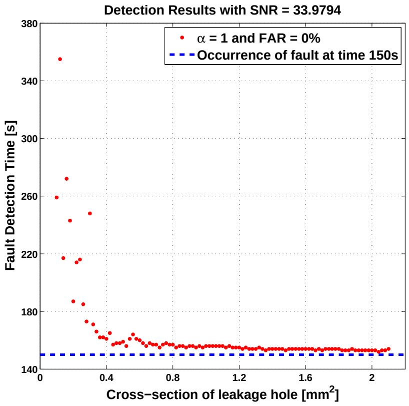

Fig. 2. Detection time Vs leakage size.

The process noise $\tilde{w}(t)$ is a zero-mean Gaussian noise with $\sigma_{\tilde{w}}^2 = 0.0001\ \mathrm{I}_4$. The initial state is $x(0) = [4.81; 4.70; 1.0; 1.0]^\top$. Each sensor in the sensor network can measure the entire state. The measurement noise v_i in each node i is an independent zero-mean Gaussian noise with $\sigma_{v_i}^2 = 0.0004\ \mathrm{I}_4$ and the Signal over Noise Ratio $\mathrm{SNR} = 33.9794$. The 15-nodes sensor network is obtained by distributing the nodes randomly over a squared area of size $n/2$. The graph $\mathcal{G}$ is generated under the rule that two nodes are connected if their relative distance is less than $1.7\sqrt{n}$. The parameter α is chosen equal to 1. We assume that a leakage occurs at the bottom of $\mathrm{Tank}\,3$ at time $t_0 = 150s$ with a fault matrix Φ given by

$$\Phi = \begin{bmatrix} 0 & 0 & 0.0188 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -0.0190 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Fig. 1 shows the estimation performance and residuals and thresholds in one of the state components affected by the fault. We can see that the fault is detected at 153s after 3 seconds in each node. The sensitivity with respect to the fault magnitude is then analyzed. We run 101 experiments with different randomly generated graphs. The cross-section of the leakage in Tank 3, Δa_3, is assigned evenly every 0.02 in the interval $[0.1, 2.1]\mathrm{mm}^2$. The detection times for each experiment, considering a measurement noise such that $\mathrm{SNR} = 33.9794$, are shown in Fig. 2. We can see

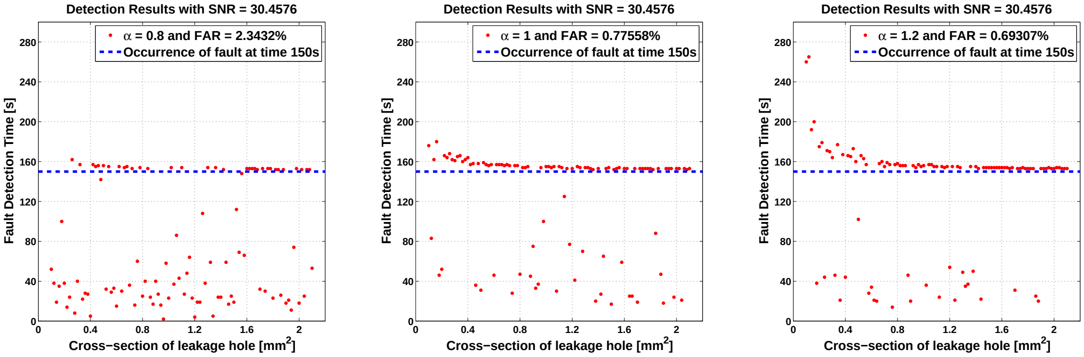

Fig. 3. Detection time Vs leakage size for different α values.

that the detection time decreases when the cross-section of the leakage increases, reaching then a minimum time of detection. We then consider the influence of the α parameter for fault detection. The detection times are plotted in Fig. 3 for the scenario with the measurement noise so that $\mathrm{SNR} = 30.4576$, considering three different α values. Since the measurements are affected by a higher noise, we can see in this scenario the presence of false-alarms. Considering different values of the α parameter, generally speaking, we can see that increasing the α value allows to obtain a smaller number of false-alarms but the detection time increases. We define the False-Alarms Rate (FAR) indicator as:

$$\mathrm{FAR} = \frac{\text{number of false alarms}}{101 \times \mathrm{n} \times \mathrm{m}} \times 100\% \, .$$

In Table I we show the FAR indicator value for nine different scenarios, considering three different SNR and three different α values. We can see that, as expected, the false alarm rate increases with the noise and decreases for larger α values. This confirms that the α parameter has to be chosen according to the noise features of the process and the wanted false-alarms rate.

TABLE I

FALSE-ALARMS RATE

α	0.8	1.0	1.2
$\mathrm{SNR} = 33.98$	0%	0%	0%
$\mathrm{SNR} = 32.04$	0.173%	0.08%	0.03%
$\mathrm{SNR} = 30.46$	2.343%	0.776%	0.693%

VI. CONCLUDING REMARKS

In this paper we have extended the distributed estimation method proposed in [14] for distributed fault detection. As a future work, we intend to further extend the framework to the non-completely measurable state case and we will analyze observability and detectability properties.

REFERENCES

[1] C. Keliris, M. M. Polycarpou, and T. Parisini, “A robust nonlinear observer-based approach for distributed fault detection of input-output interconnected systems,” *Automatica*, vol. 53, pp. 408 – 415, 2015.

[2] F. Boem, R. M. Ferrari, and T. Parisini, “Distributed fault detection and isolation of continuous-time non-linear systems,” *European Journal of Control*, vol. 17, no. 5, pp. 603–620, 2011.

[3] T. Chen, C. Wang, and D. J. Hill, “Rapid oscillation fault detection and isolation for distributed systems via deterministic learning,” *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 25, no. 6, pp. 1187–1199, 2014.

[4] V. Reppa, M. M. Polycarpou, and C. G. Panayiotou, “Distributed sensor fault diagnosis for a network of interconnected cyberphysical systems,” *Control of Network Systems, IEEE Transactions on*, vol. 2, no. 1, pp. 11–23, 2015.

[5] Q. Zhang and X. Zhang, “Distributed sensor fault diagnosis in a class of interconnected nonlinear uncertain systems,” in *Fault Detection, Supervision and Safety of Technical Processes*, vol. 8, 2012, pp. 1101–1106.

[6] I. Shames, A. M. Teixeira, H. Sandberg, and K. H. Johansson, “Distributed fault detection for interconnected second-order systems,” *Automatica*, vol. 47, no. 12, pp. 2757 – 2764, 2011.

[7] H. Dong, Z. Wang, S. X. Ding, and H. Gao, “A survey on distributed filtering and fault detection for sensor networks,” *Mathematical Problems in Engineering*, vol. 2014, p. 7, 2014.

[8] D. Ding, Z. Wang, and B. Shen, “Recent advances on distributed filtering for stochastic systems over sensor networks,” *International Journal of General Systems*, vol. 43, no. 3-4, pp. 372–386, 2014.

[9] M. S. Mahmoud and H. M. Khalid, “Distributed Kalman filtering: a bibliographic review,” *Control Theory & Applications, IET*, vol. 7, no. 4, 2013.

[10] T. Wang, L. Chang, and P. Chen, “A collaborative sensor-fault detection scheme for robust distributed estimation in sensor networks,” *Communications, IEEE Transactions on*, vol. 57, no. 10, pp. 3045–3058, October 2009.

[11] M. Nandi, A. Nayak, B. Roy, and S. Sarkar, “Hypothesis testing and decision theoretic approach for fault detection in wireless sensor networks,” *International Journal of Parallel, Emergent and Distributed Systems*, vol. 30, no. 4, pp. 262–285, 2015.

[12] E. Franco, R. Olfati-Saber, T. Parisini, and M. Polycarpou, “Distributed fault diagnosis using sensor networks and consensus-based filters,” in *Decision and Control, 45th IEEE Conference on*, 2006, pp. 386 –391.

[13] Q. Cheng, P. Varshney, J. Michels, and C. Belcastro, “Distributed fault detection with correlated decision fusion,” *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 45, no. 4, pp. 1448–1465, Oct 2009.

[14] F. Boem, Y. Xu, C. Fischione, and T. Parisini, “A distributed Pareto-optimal dynamic estimation method,” in *European Control Conference*, 2015.

[15] ——, “Distributed fault detection using sensor networks and Pareto estimation,” in *European Control Conference*, 2013, pp. 932–937.

[16] ——, “A distributed estimation method for sensor networks based on Pareto optimization,” in *Decision and Control, 51th IEEE Conference on*, 2012, pp. 775–781.

[17] K. Johansson, “The quadruple-tank process: a multivariable laboratory process with an adjustable zero,” *Control Systems Technology, IEEE Transactions on*, vol. 8, no. 3, pp. 456–465, May 2000.