A Diffusion-Based EM Algorithm for Distributed Estimation in Unreliable Sensor Networks

Silvana Silva Pereira, *Member, IEEE*, Roberto López-Valcarce, *Member, IEEE*, and Alba Pagès-Zamora, *Member, IEEE*

***Abstract*—We address the problem of distributed estimation of a parameter from a set of noisy observations collected by a sensor network, assuming that some sensors may be subject to data failures and report only noise. In such scenario, simple schemes such as the Best Linear Unbiased Estimator result in an error floor in moderate and high signal-to-noise ratio (SNR), whereas previously proposed methods based on hard decisions on data failure events degrade as the SNR decreases. Aiming at optimal performance within the whole range of SNRs, we adopt a Maximum Likelihood framework based on the Expectation-Maximization (EM) algorithm. The statistical model and the iterative nature of the EM method allow for a diffusion-based distributed implementation, whereby the information propagation is embedded in the iterative update of the parameters. Numerical examples show that the proposed algorithm practically attains the Cramer–Rao Lower Bound at all SNR values and compares favorably with other approaches.**

***Index Terms*—Consensus averaging, diffusion strategies, distributed estimation, expectation-maximization, maximum-likelihood, sensor networks, soft detection.**

I. Introduction

DISTRIBUTED estimation of unknown parameters is one of the fundamental problems in wireless sensor networks (WSNs), in which a large number of nodes favors the use of decentralized architectures to reduce complexity and power consumption, as well as to increase scalability and robustness to node failures. In practice, the data collected by the nodes may be unreliable due to for instance, external malicious attacks aimed at jeopardizing the application [1], or incorrect sensing due to sensor failures [2]. For instance, in the aircraft control field, one of the concerns is the detection of actuator and sensor failures [3]. All these *data fault events* pose an added difficulty to the distributed estimation problem, and methods based on an initial data classification stage to discard invalid observations shall perform poorly if the Signal-to-Noise Ratio (SNR) is not sufficiently high. We consider the problem of distributed estimation under the simple assumption that nodes subject to a data fault do not measure the parameter of interest and report only noise, thus modeling a transducer failure [4].

A convenient framework when dealing with unreliable measurements is to assume hidden random variables that govern the occurrence of a data fault event at each sensor. The approach adopted here aims at the computation of the Maximum Likelihood (ML) estimator by using the well-known EM algorithm [5], which amounts to a *soft classification* of the data, avoiding error-prone hard classification stages. The EM algorithm is an iterative scheme that alternates between an expectation step (E-step), where certain conditional expectation of the log-likelihood function of the observations is computed, and a maximization step (M-step), at which said function is maximized to update the estimates. The M-step requires access to the whole network dataset, as would be the case in a centralized approach. Hence, for a distributed implementation in a WSN, the global information needed at the M-step must be shared among the nodes, for instance by means of diffusion strategies [6]. Related contributions on distributed EM implementations based on diffusion include [7], where the authors use the results from [8] to show that their implementation is a Robbins-Monro stochastic approximation to the centralized scheme. A similar approach is proposed in [9] for tracking applications using particle filtering. In [10] a diffusion adaptation scheme is proposed for learning in the presence of Gaussian mixture models.

Our main contribution is a novel *distributed* version of the EM algorithm to be implemented in a WSN with a decentralized architecture using diffusion-like strategies, termed therefore *diffusion-based distributed EM* (DB-DEM). The main novelty of DB-DEM with respect to existing schemes [7]–[10] resides in the fact that the propagation of information across the network is embedded in the iterative update of the parameters, where a faster term for information diffusion is combined with a slower term for information averaging. The relative speed of these two terms is controlled by assigning them appropriate time-varying step-size sequences. Therefore, the DB-DEM algorithm differentiates from adaptive diffusion techniques [6], [10], where adaptive algorithms run over networks in a distributed fashion. Numerical examples show that, given a sufficient number of iterations, the mean square error (MSE) for the DB-DEM is equal to that of the centralized EM estimator at all SNR values, both practically attaining the Cramer-Rao Lower Bound (CRLB).

Manuscript received February 14, 2013; revised April 15, 2013; accepted April 16, 2013. Date of publication April 26, 2013; date of current version May 02, 2013. The work of S. S. Pereira and A. Pagès-Zamora was supported by the Spanish Government, ERDF funds (CONSOLIDER-INGENIO 2010 CSD2008-00010 COMONSENS and TEC2010-19171 MOSAIC) and the Spanish Government (2009SGR- 01236 AGAUR). R. López-Valcarce work is supported by the Spanish Government, ERDF funds (TEC2010-21245-C02-02/TCM DYNACS, CONSOLIDER-INGENIO 2010 CSD2008-00010 COMONSENS) and the Galician Government (CN 2012/260 AtlantTIC). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Gesualdo Scutari.

S. S. Pereira and A. Pagès-Zamora are with the SPCOM Group, Universitat Politècnica de Catalunya - Barcelona Tech (UPC), 08034 Barcelona, Spain (e-mail: silvana.silva@upc.edu; alba.pages@upc.edu).

R. López-Valcarce is with the GPSC Group, Universidade de Vigo, 36310 Vigo, Spain (e-mail: valcarce@gts.uvigo.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2013.2260329

The letter is organized as follows. Section II presents the signal model, the CRLB and its asymptotic behavior, and a brief discussion on previous estimators for this problem. The centralized EM estimator is presented in Section III, leading to the distributed version (DB-DEM) in Section IV. Simulation results and conclusions are presented in Sections V and VI respectively.

II. Signal Model and Cramer-Rao Lower Bound

Consider a set of N sensors collecting N observations

$$y_i = a_i x + w_i, \qquad i = 1, \dots, N, \tag{1}$$

where x is the parameter of interest, $\{a_i, \forall i\} = \{0, 1\}$ are independent identically distributed (i.i.d.) Bernoulli random variables with probability $p \triangleq \Pr\{a_i = 1\}$, and $\{w_i, \forall i\}$ are i.i.d. zero-mean Gaussian with variance σ^2 and independent of $\{a_i, \forall i\}$. Thus, a value of $a_i = 1$ indicates that node i is actually sensing the parameter x (corrupted by noise), whereas $a_i = 0$ indicates a transducer failure such that sensor i measures only noise. Rewriting (1) for the entire network in vector form yields $\boldsymbol{y} = \boldsymbol{a}x + \boldsymbol{w}$ where $\boldsymbol{y} = [y_1, \cdots, y_N]^T$, $\boldsymbol{a} = [a_1, \cdots, a_N]^T$ and $\boldsymbol{w} = [w_1, \cdots, w_N]^T$. Since the observations are i.i.d., the probability density function (pdf) of $\boldsymbol{y}$ is

$$f(\boldsymbol{y}|\boldsymbol{\theta}) = \prod_{i=1}^{N} \left[\frac{p}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - x)^2}{2\sigma^2}} + \frac{1-p}{\sqrt{2\pi\sigma^2}} e^{-\frac{y_i^2}{2\sigma^2}} \right], \tag{2}$$

where $\boldsymbol{\theta} = [x, \sigma^2]^T$. The parameter to be estimated is x, whereas σ^2 is regarded as an unknown *nuisance* parameter; the a priori probability p is assumed known throughout. Closed-form maximization of $f(\boldsymbol{y}|\boldsymbol{\theta})$ is not possible, and one must resort to numerical methods. As a first step we compute the CRLB for the estimation of x under model (1), in order to benchmark the performance of different estimators.

Let $\gamma^2 \triangleq x^2/\sigma^2$ denote the SNR. After some algebra, the *normalized* CRLB for the estimation of x is found to be

$$\frac{\mathrm{CRLB}\{x\}}{x^2} = \frac{1}{Np\gamma^2} \cdot \frac{1}{1 - \varrho I_{11}(\gamma) - \varrho^2 \gamma^2 \frac{I_{12}^2(\gamma)}{I_{22}(\gamma)}} \tag{3}$$

where $\varrho \triangleq (1-p)p^{-1}$ and the functions $I_{mn}(\gamma)$ are defined in terms of a zero-mean, unit-variance Gaussian distributed random variable z as follows, with $g_\gamma(z) \triangleq e^{\gamma z + \gamma^2/2}$:

$$I_{11}(\gamma) = \mathbb{E}\left\{ \frac{z^2}{g_\gamma(z) + \varrho} \right\} = \int_{-\infty}^{\infty} \frac{z^2}{g_\gamma(z) + \varrho} \cdot \frac{e^{-z^2/2}}{\sqrt{2\pi}} \mathrm{d}z, \tag{4}$$

$$I_{12}(\gamma) = \mathbb{E}\left\{ \frac{\gamma z + 2z^2}{g_\gamma(z) + \varrho} \right\}, \tag{5}$$

$$I_{22}(\gamma) = \mathbb{E}\left\{ \frac{\left[(z^2 - 1) g_\gamma(z) + \varrho \left((z+\gamma)^2 - 1 \right) \right]^2}{(g_\gamma(z) + \varrho) g_\gamma(z)} \right\}. \tag{6}$$

For low SNR, it can be readily checked that (3) behaves as

$$\frac{\mathrm{CRLB}\{x\}}{x^2} \approx \frac{1}{Np^2\gamma^2} \qquad \text{for } \gamma^2 \to 0. \tag{7}$$

This is the performance achieved by the Best Linear Unbiased Estimator (BLUE) at low SNR, given by

$$\hat{x}_{\mathrm{BLUE}} = \frac{\sum_{i=1}^{N} y_i}{Np}, \quad \frac{\mathrm{Var}\{\hat{x}_{\mathrm{BLUE}}\}}{x^2} = \frac{1}{Np^2\gamma^2} + \frac{1-p}{Np}. \tag{8}$$

Thus, the BLUE becomes asymptotically efficient for low SNR.

In the high SNR regime, it follows from (3) that

$$\frac{\mathrm{CRLB}\{x\}}{x^2} \approx \frac{1}{Np\gamma^2} \qquad \text{for } \gamma^2 \to \infty. \tag{9}$$

Comparing (9) with (8), it is apparent that the BLUE is not efficient in high SNR. On the other hand, (9) coincides with the performance achieved by a *clairvoyant* estimator with knowledge of the values $\{a_i, \forall i\}$, which is given by

$$\hat{x}_{\mathrm{CV}} = \frac{\sum_{i=1}^{N} a_i y_i}{\sum_{i=1}^{N} a_i}. \tag{10}$$

This is not unexpected since it should be feasible, for sufficiently high SNR, to decide with vanishingly small error probability which nodes are sensing only noise: the corresponding measurements are then removed from the averaging operation. This is the principle behind the mixed detection-estimation (MDE) scheme of [4], which is based on *local hard decisions* for distributed estimation of x. Assuming knowledge of the noise variance σ^2, the MDE scheme iteratively switches between Maximum A Posteriori (MAP) detection of the faulty sensors, taking the parameter estimate from the previous iteration as the true value of x, and ML estimation assuming the detected values of the $\{a_i\}$ as the true ones. Although asymptotically efficient for high SNR, MDE becomes severely biased for medium and low SNR values due to errors in the hard decisions on a_i.

The CRLB asymptotes (7) and (9) motivate the search for better estimators performing well at *all* SNR values, as opposed to MDE and BLUE. Next we derive a centralized EM-based ML estimator, to be followed by a distributed implementation for WSNs.

III. ML Estimation: Centralized EM

ML estimators have the desirable properties of being asymptotically unbiased and efficient as the number of samples goes to infinity. The EM algorithm is a numerical method to compute the ML estimate in the presence of incomplete observations. We regard the observation vector $\boldsymbol{y}$ as the *incomplete* observation and $\{\boldsymbol{y}, \boldsymbol{a}\}$ as the *complete* observation. Assuming for the time being a centralized approach in which $\boldsymbol{y}$ is available, at iteration t one performs the following:

1) *E-step*: given an estimate $\hat{\boldsymbol{\theta}}(t) = [\hat{x}(t), \sigma^2(t)]^T$, compute the conditional expectation

$$Q\left(\tilde{\boldsymbol{\theta}}; \hat{\boldsymbol{\theta}}(t)\right) = \mathbb{E}_{\boldsymbol{a}}\left\{ \log f(\boldsymbol{y}, \boldsymbol{a}|\tilde{\boldsymbol{\theta}}) | \hat{\boldsymbol{\theta}}(t), \boldsymbol{y} \right\}, \tag{11}$$

where $\tilde{\boldsymbol{\theta}} = [\tilde{x}, \tilde{\sigma}^2]^T$ denotes a trial value of $\boldsymbol{\theta}$.

2) *M-step*: obtain the estimate for the next iteration as

$$\hat{\boldsymbol{\theta}}(t+1) = \arg\max_{\tilde{\boldsymbol{\theta}}} Q\left(\tilde{\boldsymbol{\theta}}; \hat{\boldsymbol{\theta}}(t)\right). \tag{12}$$

Let $\varphi_i(t) \triangleq \Pr\{a_i = 1 | \hat{\boldsymbol{\theta}}(t), y_i\}$. By virtue of Bayes' theorem, the E-step results in computing

$$\varphi_i(t) = \frac{p \cdot \exp\left\{ -\frac{(y_i - \hat{x}(t))^2}{2\hat{\sigma}^2(t)} \right\}}{p \cdot \exp\left\{ -\frac{(y_i - \hat{x}(t))^2}{2\hat{\sigma}^2(t)} \right\} + (1-p)\exp\left\{ -\frac{y_i^2}{2\hat{\sigma}^2(t)} \right\}}. \tag{13}$$

Then, after some algebra (11) becomes

$$Q\left(\tilde{\boldsymbol{\theta}};\hat{\boldsymbol{\theta}}(t)\right) \propto -\frac{N}{2}\log 2\pi\tilde{\sigma}^2 - \frac{1}{2\tilde{\sigma}^2}\left[\|\boldsymbol{y}\|^2 + \tilde{x}^2 \mathbf{1}^T \boldsymbol{\varphi}(t) - 2\tilde{x}\boldsymbol{y}^T\boldsymbol{\varphi}(t)\right] \tag{14}$$

where $\boldsymbol{\varphi}(t) = [\varphi_1(t), \cdots, \varphi_N(t)]^T$. The M-step is now easily accomplished: maximizing (14) with respect to $\tilde{\boldsymbol{\theta}}$ yields

$$\hat{x}(t+1) = \frac{\boldsymbol{y}^T\boldsymbol{\varphi}(t)}{\mathbf{1}^T\boldsymbol{\varphi}(t)} \tag{15}$$

$$\hat{\sigma}^2(t+1) = \frac{1}{N}\left[\|\boldsymbol{y}\|^2 - \frac{\left(\boldsymbol{y}^T\boldsymbol{\varphi}(t)\right)^2}{\mathbf{1}^T\boldsymbol{\varphi}(t)}\right]. \tag{16}$$

From the Cauchy-Schwarz inequality and the fact that $0 \leq \varphi_i(t) \leq 1$ for all $\{i, t\}$, it follows that $\hat{\sigma}^2(t) \geq 0$ for all t, which is a desirable property.

Obtaining a distributed implementation of the EM estimator entails the computation of (15)–(16), which require global information, in a decentralized fashion. A similar problem is considered in [4] assuming σ^2 known and relying on hard decisions on $\{a_i, \forall i\}$. The EM approach, in contrast, does not require knowledge of σ^2 and implicitly makes soft decisions on a_i, which is the key to obtaining good performance at all SNR values.

IV. A Diffusion-based Distributed EM Estimator

We propose a scheme where the information needed for the computation of (15)–(16) is spread over the network by diffusion-like strategies. Consider a WSN with nodes indexed from 1 to N, where the communications are restricted to a neighborhood of each node. The network is assumed connected, i.e., there exist a path between any pair of nodes $\{i, j\}$. Let $\boldsymbol{W} \in \mathbb{R}^{N\times N}$ denote a *weight matrix* with a nonzero entry W_{ij} only if node i can receive information from node j. It is assumed that $\boldsymbol{W}$ is symmetric and that it satisfies

$$i)\quad \boldsymbol{W}\mathbf{1} = \mathbf{1}, \quad \mathbf{1}^T\boldsymbol{W} = \mathbf{1}^T, \quad \rho\left(\boldsymbol{W} - \frac{\mathbf{1}\mathbf{1}^T}{N}\right) < 1 \tag{17}$$

where $\mathbf{1} \in \mathbb{R}^N$ is an all-ones vector and $\rho(\cdot)$ denotes the spectral radius. Further, we assume that the measurements are bounded. At each time instant[1] k, node i keeps local variables $\hat{x}_i(k)$ and $\hat{\sigma}_i^2(k)$, as well as the following auxiliary variables:

$$\begin{aligned} f_{yp}(i,k) &= y_i\hat{p}_i(k), & f_y(i,k) &= y_i^2, \\ f_p(i,k) &= \hat{p}_i(k), & f_1(i,k) &= 1, \end{aligned} \tag{18}$$

where $\hat{p}_i(k)$ is the a posteriori probability of a_i given the observation y_i and the local estimates, given by

$$\hat{p}_i(k) = \frac{p\cdot\exp\left\{-\frac{(y_i-\hat{x}_i(k))^2}{2\hat{\sigma}_i^2(k)}\right\}}{p\cdot\exp\left\{-\frac{(y_i-\hat{x}_i(k))^2}{2\hat{\sigma}_i(k)}\right\} + (1-p)\exp\left\{-\frac{y_i^2}{2\hat{\sigma}_i(k)}\right\}}. \tag{19}$$

The main difference between $\varphi_i(t)$ in (13) and $\hat{p}_i(k)$ in (19) is that $\varphi_i(t)$ is computed using the global estimates in (15)–(16), whereas computing $\hat{p}_i(k)$ only requires the local estimates $\hat{x}_i(k)$ and $\hat{\sigma}_i^3(k)$ at each node i. By means of local communications, the local information in $\hat{p}_i(k)$ is appropriately diffused over the network. Thus, each node can in turn update its local estimates $\hat{x}_j(k+1)$ and $\hat{\sigma}_j^2(k+1)$, so that, in the limit, all nodes reach an agreement on their values. The computation of these local estimates is detailed in Table I, which summarizes the DB-DEM algorithm. Briefly, the DB-DEM iteration for node i at time k can be described as follows. First, the a posteriori probability in (19) is computed using the current estimates $\hat{x}_i(k)$ and $\hat{\sigma}_i^2(k)$. Information is then exchanged with neighboring nodes to update the intermediate variables $\phi_\nu(i,k)$. Finally, the local estimates $\hat{x}_i(k)$ and $\hat{\sigma}_i^2(k)$ are updated using the intermediate variables $\phi_\nu(i,k)$. This procedure is repeated until convergence.

TABLE I
Diffusion-Based Distributed EM Algorithm

For $i = 1, \cdots, N$

1) *Initialize:*

$$\begin{aligned} f_{yp}(i,1) &= y_i p & & \\ f_p(i,1) &= p & \hat{x}_i(1) &= \textstyle\sum_{j=1}^N W_{ij}y_j \\ f_y(i,1) &= y_i^2 & \hat{\sigma}_i^2(1) &= \textstyle\sum_{j=1}^N W_{ij}y_j^2 - \hat{x}_i^2(1) \\ f_1(i,1) &= 1 & & \end{aligned}$$

where p is the a priori probability.

For $k \geq 1$,

2) *E-Step*: given $\hat{x}_i(k)$ and $\hat{\sigma}_i^2(k)$, compute $\hat{p}_i(k)$ as in (19).

3) *M-Step*: for every subindex $\nu \in \{yp, p, y, 1\}$, and with $f_\nu(j,k)$ given by (18), compute the intermediate variables

$$\phi_\nu(i,k) = (1-\beta(k))\sum_{j=1}^N W_{ij}\phi_\nu(j,k-1) + \alpha(k)\sum_{j=1}^N W_{ij}f_\nu(j,k) \tag{20}$$

where

$$\alpha(k) = \frac{1}{k}, \quad \beta(k) = \frac{1}{k^\delta}, \quad 0 < \delta < 1, \quad k = 1, 2, \cdots \tag{21}$$

and then update

$$\begin{aligned} \hat{x}_i(k+1) &= \frac{\phi_{yp}(i,k)}{\phi_p(i,k)} \\ \hat{\sigma}_i^2(k+1) &= \frac{\phi_y(i,k)}{\phi_1(i,k)} - \frac{\phi_{yp}^2(i,k)}{\phi_p(i,k)\phi_1(i,k)} \end{aligned} \tag{22}$$

4) *Repeat* steps 2 and 3 until convergence.

Notice that the terms including $\alpha(k), \beta(k)$ in the update equation (20) converge to zero. The use of vanishing control parameters is common in stochastic adaptive signal processing and control [11], and also in consensus applications with noisy signals [12], [13]. For $\nu \in \{yp, p, y, 1\}$, the vector $\boldsymbol{\phi}_\nu(k) \triangleq [\phi_\nu(1,k), \cdots, \phi_\nu(N,k)]^T$ evolves for $k \geq 1$ as

$$\boldsymbol{\phi}_\nu(k) = (1-\beta(k))\boldsymbol{W}\boldsymbol{\phi}_\nu(k-1) + \alpha(k)\boldsymbol{W}\boldsymbol{f}_\nu(k) \tag{23}$$

where $\boldsymbol{f}_\nu(k) \triangleq [f_\nu(1,k), \cdots, f_\nu(N,k)]^T$. The last term on the right-hand side of (23) is responsible for propagating information over the network, whereas the first term drives the state vector $\boldsymbol{\phi}_\nu(k)$ toward a consensus (identical entries) once the information has propagated through the network, so that all nodes reach an agreement on the values of their estimates in (22). Thus, intuitively, one should have $1 - \beta(k) \ll \alpha(k)$ for small k, tending to the reverse situation as k increases. The parameter δ allows to control this relationship and to tune the onset

[1]We use the index k for the distributed implementation to avoid confusion with the centralized approach of Section III, for which we use the index t.

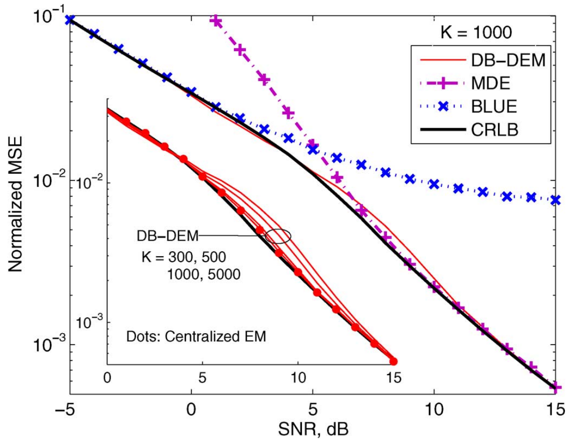

Fig. 1. NMSE vs. SNR for the DB-DEM, MDE, BLUE and centralized EM estimators. The CRLB is depicted as a reference.

of the averaging process. In general, more connected topologies may benefit form values of δ closer to one to increase the convergence rate. It must be emphasized that, once the observations $\{y_i\}$ are given and assuming a deterministic schedule for the control parameters $\alpha(k)$ and $\beta(k)$, the DB-DEM algorithm in Table I is a completely deterministic process. A preliminary convergence analysis, not included due to lack of space, shows that, under assumption i) and with the choice in (21), if convergence of the DB-DEM algorithm takes place, then it must be to a fixed point of the centralized EM scheme, upon which all the nodes agree.

V. Simulation Results

We simulate an example network composed of $N = 100$ nodes randomly deployed over a unit square with connectivity radius $r_c = 0.25$. The nodes sense the parameter $x = 1$ with probability $p = 0.6$ and SNR values in the range [−5,15] dB. The DB-DEM algorithm from Table I is run over the network with $\delta = 0.8$ and a Metropolis weight matrix $\boldsymbol{W}$ [14]. A comparison is made with MDE [4], BLUE (8) and the centralized EM of Section III, using the normalized MSE, defined as

$$\mathsf{NMSE}\{\hat{x}\} = \frac{1}{Nx^2} \cdot \sum_{i=1}^{N} \mathbb{E}\left[\|\hat{x}_i(k) - x\|_2^2\right]$$

as performance metric. Fig. 1 shows the residual NMSE vs. SNR averaged over 1000 independent realizations after $K = 1000$ iterations for the different estimators. The CRLB (3) is included as a reference (solid thick line). As anticipated in Section II, the MDE ('$+$') (which assumes known σ^2) and BLUE ('$\times$') estimators are efficient in the high- and low-SNR regimes respectively. The MDE suffers from a large bias in the low-SNR regime, and hence a large NMSE, due to frequent errors in its MAP-based hard decision stage. An embedded higher-resolution plot further shows the results for $K = \{300, 500, 1000, 5000\}$ along with the results for the centralized EM (dots) and the CRLB (solid thick line). The centralized EM is efficient over practically the whole SNR range, except for a slight deviation which can be observed for intermediate values, and a similar behavior is observed for the DB-DEM. The explanation for this phenomenon is the fact that the MLE is not necessarily efficient for finite N. For some SNR values, the DB-DEM variance remain slightly below the CRLB indicating a residual bias, which nevertheless vanishes with increasing K. Therefore, we can conclude that the DB-DEM practically attains the performance of the centralized EM for a sufficient number of iterations K.

VI. Conclusion

A diffusion-based EM algorithm has been proposed for distributed estimation in WSNs in the presence of noisy measurements and data faults. The novelty of the proposed scheme, denoted DB-DEM, is that the propagation of information over the network is embedded in the iterative update of parameters. In this way, DB-DEM combines two operations, where an initial period for information diffusion is gradually switched off at the same time as an averaging process is gradually switched on. Numerical results show that the proposed scheme is able to practically attain the CRLB at all SNR values, outperforming similar algorithms in terms of MSE performance. Ongoing research is addressing the applicability of the DB-DEM principle to more advanced data models.

References

[1] Z. Benenson, P. M. Cholewinski, and F. C. Freiling, "Vulnerabilities and Attacks in Wireless Sensor Networks," in *Wireless Sensor Networks Security*, ser. Cryptology & Information Security Series (CIS), J. Lopez and J. Zhou, Eds. Philadelphia, PA, USA: IOS Press, 2008.
[2] K. Ni *et al.*, "Sensor network data fault types," *ACM Trans. Sensor Netw.*, vol. 5, no. 3, 2009.
[3] A. S. Willsky, "A survey for design methods for failure detection in dynamic systems," *Automatica*, vol. 12, no. 6, pp. 601–611, 1976.
[4] Q. Zhou, S. Kar, L. Huie, and S. Cui, "Distributed estimation in sensor networks with imperfect model information: An adaptive learning-based approach," *Proc. IEEE ICASSP*, 2012.
[5] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Stat. Soc.*, ser. B, vol. 39, no. 1, pp. 1–38, 1977.
[6] A. H. Sayed, "Diffusion adaptation over networks," in *E-Reference Signal Processing*, R. Chellapa and S. Theodoridis, Eds. Amsterdam, The Netherlands: Elsevier, 2013.
[7] Y. Weng, L. Xie, and W. Xiao, "Diffusion scheme of distributed EM algorithm for gaussian mixtures over random networks," in *Proc. IEEE ICCA*, 2009, pp. 1529–1534.
[8] M. A. Sato and S. Ishii, "On-line EM Algorithm for the normalized Gaussian network," *Neural Comput.*, vol. 12, no. 2, pp. 407–432, Feb. 2000.
[9] D. Gu, "Distributed EM algorithm for Gaussian mixtures in sensor networks," *IEEE Trans. Neural Networks*, vol. 19, no. 7, pp. 1154–1166, 2008.
[10] Z. J. Towfic, J. Chen, and A. Sayed, "Collaborative learning of mixture models using diffusion adaptation," *IEEE MLSP*, 2011.
[11] L. Ljung and T. Söderström*, Theory and Practice of Recursive Identification.*. Cambridge, MA, USA: MIT Press, 1983.
[12] Y. Hatano, A. Das, and M. Mesbahi, "Agreement in presence of noise: Pseudogradients on random geometric networks," in *Proc. IEEE DCC and ECC*, 2005, pp. 6382–6387.
[13] S. Kar and J. Moura, "Distributed consensus algorithms in sensor networks with imperfect communication: Link failures and channel noise," *IEEE Trans. Signal Process.*, vol. 57, no. 1, pp. 355–369, 2009.
[14] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. IPSN*, 2005, pp. 63–70.