Parameter estimation in wireless sensor networks with faulty transducers: a distributed EM approach

Silvana Silva Pereira[a,*], Roberto López-Valcarce[b], *EURASIP Member*, Alba Pagès-Zamora[a,**]

[a] *SPCOM Group, Universitat Politècnica de Catalunya - BarcelonaTech (UPC), C/Jordi Girona 31, 08034, Barcelona, Spain (e-mail: alba.pages@upc.edu).*
[b] *GPSC Group, Universidade de Vigo, Rua Maxwell s/n 36310, Spain (e-mail: valcarce@gts.uvigo.es).*

Abstract

We address the problem of distributed estimation of a vector-valued parameter performed by a wireless sensor network in the presence of noisy observations which may be unreliable due to faulty transducers. The proposed distributed estimator is based on the Expectation-Maximization (EM) algorithm and combines consensus and diffusion techniques: a term for information diffusion is gradually turned off, while a term for updated information averaging is turned on so that all nodes in the network approach the same value of the estimate. The proposed method requires only local exchanges of information among network nodes and, in contrast with previous approaches, it does not assume knowledge of the a priori probability of transducer failures or the noise variance. A convergence analysis is provided, showing that the convergent points of the centralized EM iteration are locally asymptotically convergent points of the proposed distributed scheme. Numerical examples show that the distributed algorithm asymptotically attains the performance of the centralized EM method.

Keywords: Consensus averaging, diffusion strategies, distributed estimation, expectation-maximization, maximum-likelihood, soft detection, wireless sensor networks.

1. Introduction

Wireless sensor networks (WSNs) consist of many small, spatially distributed autonomous nodes, equipped with one or more on-board sensors to collect information from the surrounding environment, and which collaborate to jointly perform a variety of inference and information processing tasks. Applications include environmental and healthcare monitoring, event detection, target classification, and industrial automation [1, 2]. Distributed processing, by which computations are carried out within the network in order to avoid raw data transmission to a fusion center, is a desirable feature of WSNs since it usually results in energy savings and

*Present Address: CNS Group, Universitat Pompeu Fabra, C/Ramon Trias Fargas, 25-27, 08005 Barcelona, Spain (e-mail: silvana.silva@upf.edu)
**Corresponding author

improved robustness [3, 4]. In particular, distributed estimation of unknown parameters in WSNs is an important problem which has been extensively considered over the past few years [5, 6, 7, 8, 9, 10, 11].

In practice, estimation performance may be severely degraded when the information collected by the nodes becomes unreliable due to sensor malfunction [12, 13, 14, 15], and therefore it is important to efficiently identify faulty nodes [16, 17]. Given that nodes are typically deployed in outdoor, potentially harsh environments, sensor malfunction effects should not be lightly dismissed. We consider the problem of distributed estimation of a vector-valued parameter from the observations collected by a WSN where some nodes may be subject to random transducer faults, so that their reports contain only noise [13, 18]. In the presence of such unreliable observations, one possibility is to run a node classification stage previously to the estimation stage [19]; however, this entails increased computational complexity and communication cost. In relation to algorithms based on prior detection of faulty nodes, the Mixed Detection and Estimation (MDE) scheme in [18] performs the node classification and estimation tasks in a jointly distributed manner. However, since MDE classifies nodes based on hard decisions, it is prone to decision errors whenever the signal-to-noise ratio (SNR) is not sufficiently high. To avoid this problem, we adopt an approach in which a *soft classification* of the data is performed by means of the expectation-maximization (EM) algorithm, a well-known method for computing the maximum likelihood (ML) estimate in the presence of hidden variables [20, 21]. The EM algorithm implicitly and iteratively produces estimates of the class probabilities, alternating between an expectation step (E-step), where access to the whole network dataset is required, and a maximization step (M-step), where updated estimates are obtained.

Distributed implementations of the EM algorithm for Gaussian mixture density estimation and clustering have been previously proposed. For example, in *incremental* approaches [22, 23, 24, 25], computations involving global network information at the E-step are addressed via aggregation strategies, assigning routing paths or junction trees within the network. This problem is avoided in [26, 27, 29], which apply full-blown gossip- or consensus-based schemes at each E-step so that all nodes arrive at an agreement about *every* intermediate estimate. The main drawback of these methods, however, is the need to exchange a large amount of information among neighbor nodes, with the consequent penalty in energy efficiency. In [28] a distributed EM algorithm based on the alternating direction method of multipliers (ADMM) is proposed for clustering. In this scheme the communication overhead is reduced but at the cost of significantly increasing the computational cost since each node has to solve a convex optimization problem via, e.g., interior point methods at each iteration. A potential way to overcome these problems is the use of diffusion strategies [11], by which nodes exchange local information only once per EM iteration and perform averaging over the values in their neighborhoods [30, 31, 32] (see [33]

for an extension to general mixture models). Convergence analyses of these schemes either assume that an infinite amount of data is available at each node [30, 32], or adopt a stochastic framework under an independence assumption [31].

The algorithm proposed in this paper is based on a different diffusion-based approach [34, 35], in which the propagation of information throughout the network is embedded in the iterative parameter update. This is done by appropriately combining two terms for information diffusion and information averaging (consensus) in the update equations. The resulting iteration, termed *diffusion-averaging distributed Expectation-Maximization* (DA-DEM), is reminiscent of so-called *consensus+innovations* (C+I) algorithms for distributed estimation in linear models [36], whose updates combine a consensus term and a local innovation term; nevertheless, several important differences should be highlighted. First, the model underlying C+I schemes is linear, but in our setting this property does not apply due to the potential presence of faulty nodes. Second, C+I schemes are usually designed for on-line adaptation, i.e., sensors keep acquiring new observations as time progresses, whereas the DA-DEM algorithm is of batch type in which a single measurement is available to each sensor. Thus, in our setting, the "innovation" provided by the diffusion term does not correspond to information provided by new measurements, but rather to that provided by the iterative refinement of the estimates. Third, in contrast with [18, 34, 35, 36] where the diffusion and averaging terms have different asymptotic decay rates, thus leading to mixed time-scale recursions, in DA-DEM both terms have the same rate. In contrast with [30, 31, 32], this feature allows for the development of a local convergence analysis under a deterministic setting with a finite amount of data, showing that any convergent point of the centralized EM iteration, and therefore a (possibly local) maximum of the likelihood function, must be an asymptotically convergent point of DA-DEM. Numerical examples show that the DA-DEM estimator asymptotically attains the performance of centralized EM in terms of mean square error (MSE). In addition to the aforementioned convergence analysis, further contributions with respect to [35] include lack of knowledge about the a priori probability of a sensor fault and the consideration of vector-valued parameter. In contrast with incremental strategies, DA-DEM does not require the computation and management of routing paths through the network, resulting in sizable reduction in convergence time and thus leading to energy savings.

The paper is organized as follows. Sec. 2 describes the signal model, and Sec. 3 presents the centralized EM-based estimator, the starting point for the development of the distributed implementation in Sec. 4. The convergence analysis of DA-DEM is developed in Sec. 5. Finally, simulation results and conclusions are presented in Secs. 6 and 7 respectively.

Notation: We use lowercase, bold lowercase, and bold uppercase symbols to respectively denote scalars, vectors and matrices. The transpose and inverse of matrix $\boldsymbol{A}$ are denoted by

$\boldsymbol{A}^T$ and $\boldsymbol{A}^{-1}$ respectively. The 2-norm of a vector $\boldsymbol{v}$ is denoted by $\|\boldsymbol{v}\|$, whereas for a matrix $\boldsymbol{A}$, $\|\boldsymbol{A}\|_F$ denotes its Frobenius norm, $\|\boldsymbol{A}\|$ its spectral norm (i.e., its largest singular value) and, for $\boldsymbol{A}$ square, $\rho(\boldsymbol{A})$ is the spectral radius (largest of the moduli of the eigenvalues). For an $n \times n$ symmetric matrix $\boldsymbol{S}$, $\mathrm{vec}\{\boldsymbol{S}\}$ is a vector of size $n(n+1)/2$ obtained by stacking the entries of the upper triangular part of $\boldsymbol{S}$. The composition of two functions f and g is denoted by $f \circ g$, so that $(f \circ g)(x) = f(g(x))$, and $\mathbb{E}\{\cdot\}$ denotes statistical expectation.

2. Problem statement

We consider the problem of estimating a parameter vector $\boldsymbol{x} \in \mathbb{R}^{L\times 1}$ based on a set of $N \gg L$ independent observations given by

$$y_i = a_i \boldsymbol{h}_i^T \boldsymbol{x} + w_i, \qquad i = 1, \ldots, N, \tag{1}$$

where $\boldsymbol{h}_i = [h_i(1) \cdots h_i(L)]^T$ are assumed known $\forall i$, $\{w_i, \forall i\}$ are independent, identically distributed (i.i.d.) zero-mean Gaussian random variables with variance σ^2, modeling the observation noise, and $\{a_i, \forall i\}$ are i.i.d. Bernoulli random variables with $\Pr(a_i = 1) = p$, independent of $w_j, \forall\{i,j\}$. A value of $a_i = 1$ indicates that node i has actually sensed the parameter vector $\boldsymbol{x}$, whereas $a_i = 0$ indicates a transducer failure, i.e. the measurement contains only noise. The equations for the N observations can be written in vector form as

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{H}\boldsymbol{x} + \boldsymbol{w}, \tag{2}$$

where $\boldsymbol{A} = \mathrm{diag}\{\boldsymbol{a}\}$, $\boldsymbol{a} = [a_1 \cdots a_N]^T$, and

$$\boldsymbol{y} \triangleq \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}, \quad \boldsymbol{H} \triangleq \begin{bmatrix} \boldsymbol{h}_1^T \\ \vdots \\ \boldsymbol{h}_N^T \end{bmatrix}, \quad \boldsymbol{w} \triangleq \begin{bmatrix} w_1 \\ \vdots \\ w_N \end{bmatrix}.$$

Assuming for the moment a centralized framework, in which all N observations in $\boldsymbol{y}$ are available at the processing entity, a *clairvoyant* (CV) estimator, i.e., an estimator with knowledge of $\boldsymbol{A}$, should average only those observations y_i for which $a_i = 1$. The corresponding ML estimate of $\boldsymbol{x}$ is therefore

$$\hat{\boldsymbol{x}}_{\mathrm{CV}} = (\boldsymbol{H}^T\boldsymbol{A}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{A}\boldsymbol{y}, \tag{3}$$

where we have used $\boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{A}$. Since in practice knowledge of $\boldsymbol{A}$ is not available, a different approach must be followed. For instance, the Least Squares (LS) estimate is obtained by neglecting the fact that transducer faults may be present, assuming $\boldsymbol{A} = \boldsymbol{I}$ in (3):

$$\hat{\boldsymbol{x}}_{\mathrm{LS}} = (\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{y}. \tag{4}$$

Note that $\mathbb{E}\{\boldsymbol{y}\} = p\boldsymbol{H}\boldsymbol{x}$, such that the LS estimate is biased. If the probability p were known, this bias could be readily removed using

$$\hat{\boldsymbol{x}}_{\mathrm{BLUE}} = \frac{1}{p}\hat{\boldsymbol{x}}_{\mathrm{LS}}, \tag{5}$$

which, for asymptotically small SNR, constitutes the Best Linear Unbiased Estimator (BLUE)[1] [37].

Alternatively, we consider ML estimation of $\boldsymbol{x}$ under model (2). The ML estimator has the desirable properties of being asymptotically unbiased and efficient as the number of samples goes to infinity. Since the observations are i.i.d., the probability density function (pdf) of $\boldsymbol{y}$ in (2) is parameterized by $\boldsymbol{\theta} = [\boldsymbol{x}^T \ \sigma^2 \ p]^T$ and given by

$$f(\boldsymbol{y}|\boldsymbol{\theta}) = \frac{1}{(2\pi\sigma^2)^{\frac{N}{2}}} \prod_{i=1}^{N} \left[p\, \mathrm{e}^{-\frac{(y_i - \boldsymbol{h}_i^T \boldsymbol{x})^2}{2\sigma^2}} + (1-p)\, \mathrm{e}^{-\frac{y_i^2}{2\sigma^2}} \right]. \tag{6}$$

Whereas the matrix of regressors $\boldsymbol{H}$ is assumed perfectly known, the noise variance σ^2 and the a priori probability p are regarded as unknown *nuisance* parameters. Maximizing (6) w.r.t. $\boldsymbol{\theta}$ in closed form is not possible, and one has to resort to numerical methods. Since the EM algorithm is particularly well suited to problems like the one at hand, we start deriving a centralized EM estimator which implicitly performs a *soft* detection of the fault events and requires neither knowledge of the noise variance σ^2 nor of the a priori probability p. Then, a *distributed* version suitable for WSNs is derived, in which each node has access to a single observation y_i and there is no central processing unit.

3. Centralized EM Estimator

Starting from an initial estimate, the EM algorithm alternates between an E-step, where the expected log-likelihood function (LLF) of the observations is computed using the current estimates, and an M-step, where the parameters maximizing the expected LLF are obtained; under mild conditions, the EM will converge to a maximum, possibly local, of the LLF [20, 21]. Consider the observation vector in (2) with pdf given by (6). We regard $\boldsymbol{y}$ as the *incomplete* observation and $\{\boldsymbol{y}, \boldsymbol{a}\}$ as the *complete* one. Assuming that all the observations are available, at iteration t one performs the following:

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

$$Q(\tilde{\boldsymbol{\theta}}\,;\, \hat{\boldsymbol{\theta}}_t) = \mathbb{E}_{\boldsymbol{a}} \left\{ \log f(\boldsymbol{y}, \boldsymbol{a} \,|\, \tilde{\boldsymbol{\theta}}) \;\middle|\; \hat{\boldsymbol{\theta}}_t, \boldsymbol{y} \right\}, \tag{7}$$

where $\tilde{\boldsymbol{\theta}}$ denotes a trial value of $\boldsymbol{\theta}$.

[1] In the medium/high SNR regime, the BLUE only exists for $L = 1$, since for $L > 1$ it would depend on the unknown parameter $\boldsymbol{x}$. Thus, the subscript in $\hat{\boldsymbol{x}}_{\mathrm{BLUE}}$ is slightly abusing notation.

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

$$\hat{\boldsymbol{\theta}}_{t+1} = \arg\max_{\tilde{\boldsymbol{\theta}}} Q(\tilde{\boldsymbol{\theta}}\,;\,\hat{\boldsymbol{\theta}}_t). \tag{8}$$

The conditional pdf of $\{\boldsymbol{y}, \boldsymbol{a}\}$ is given by

$$\begin{aligned} f(\boldsymbol{y}, \boldsymbol{a}\,|\,\tilde{\boldsymbol{\theta}}) &= f(\boldsymbol{y}\,|\,\tilde{\boldsymbol{\theta}}, \boldsymbol{a}) \cdot f(\boldsymbol{a}\,|\,\tilde{\boldsymbol{\theta}}) \\ &= \frac{1}{(2\pi\tilde{\sigma}^2)^{\frac{N}{2}}} \cdot \exp\left\{-\frac{\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{H}\tilde{\boldsymbol{x}}\|^2}{2\tilde{\sigma}^2}\right\} \\ &\quad \cdot \prod_{i=1}^{N} \tilde{p}^{\,a_i}\,(1-\tilde{p})^{1-a_i} \end{aligned} \tag{9}$$

Taking the logarithm yields

$$\begin{aligned} \log f(\boldsymbol{y}, \boldsymbol{a}\,|\,\tilde{\boldsymbol{\theta}}) \;\propto\; & -\frac{N}{2}\log\tilde{\sigma}^2 - \frac{1}{2\tilde{\sigma}^2}\left[\|\boldsymbol{y}\|^2 + \tilde{\boldsymbol{x}}^T\boldsymbol{H}^T\boldsymbol{A}\boldsymbol{H}\tilde{\boldsymbol{x}}\right. \\ & \left. - 2\tilde{\boldsymbol{x}}^T\boldsymbol{H}^T\boldsymbol{A}\boldsymbol{y}\right] + \sum_{i=1}^{N}\left[a_i\log\tilde{p} + (1-a_i)\log(1-\tilde{p})\right]. \end{aligned} \tag{10}$$

In order to obtain $Q(\tilde{\boldsymbol{\theta}};\hat{\boldsymbol{\theta}}_t)$ we must take the expectation over $\boldsymbol{a}$ of (10) conditioned on the observations $\boldsymbol{y}$ and on the previous estimate $\hat{\boldsymbol{\theta}}_t$. To this end, let $\hat{a}_{i,t} = \mathbb{E}_{\boldsymbol{a}}[a_i\,|\,\hat{\boldsymbol{\theta}}_t, y_i] = \Pr\left\{a_i = 1\,|\,\hat{\boldsymbol{\theta}}_t, y_i\right\}$ denote the a posteriori expected value of a_i at time t, and let $\hat{\boldsymbol{A}}_t = \mathrm{diag}\{\hat{\boldsymbol{a}}_t\}$ with $\hat{\boldsymbol{a}}_t = [\hat{a}_{1,t}\,\cdots\,\hat{a}_{N,t}]^T$. The a posteriori expected value $\hat{a}_{i,t}$ can be found using Bayes' rule as follows:

$$\begin{aligned} \hat{a}_{i,t} &= \frac{f(y_i\,|\,\hat{\boldsymbol{\theta}}_t, a_i = 1)\cdot\Pr\left\{a_i = 1\,|\,\hat{\boldsymbol{\theta}}_t\right\}}{f(y_i\,|\,\hat{\boldsymbol{\theta}}_t)} \\ &= \frac{\hat{p}_t\cdot\exp\left\{-\frac{(y_i - \boldsymbol{h}_i^T\hat{\boldsymbol{x}}_t)^2}{2\hat{\sigma}_t^2}\right\}}{\hat{p}_t\cdot\exp\left\{-\frac{(y_i - \boldsymbol{h}_i^T\hat{\boldsymbol{x}}_t)^2}{2\hat{\sigma}_t^2}\right\} + (1-\hat{p}_t)\cdot\exp\left\{-\frac{y_i^2}{2\hat{\sigma}_t^2}\right\}}. \end{aligned} \tag{11}$$

Then, from (10) we have

$$\begin{aligned} Q(\tilde{\boldsymbol{\theta}}\,;\,\hat{\boldsymbol{\theta}}_t) \;\propto\; & -\frac{N}{2}\log\tilde{\sigma}^2 \\ & - \frac{1}{2\tilde{\sigma}^2}\left[\|\boldsymbol{y}\|^2 + \tilde{\boldsymbol{x}}^T\hat{\boldsymbol{\Gamma}}_t\tilde{\boldsymbol{x}} - 2\tilde{\boldsymbol{x}}^T\hat{\boldsymbol{\psi}}_t\right] \\ & + \sum_{i=1}^{N}\hat{a}_{i,t}\log\tilde{p} + \left(N - \sum_{i=1}^{N}\hat{a}_{i,t}\right)\log(1-\tilde{p}), \end{aligned} \tag{12}$$

where for convenience we have defined

$$\hat{\boldsymbol{\Gamma}}_t \;\triangleq\; \boldsymbol{H}^T\hat{\boldsymbol{A}}_t\boldsymbol{H} = \sum_{i=1}^{N}\hat{a}_{i,t}\boldsymbol{h}_i\boldsymbol{h}_i^T, \tag{13}$$

$$\hat{\boldsymbol{\psi}}_t \;\triangleq\; \boldsymbol{H}^T\hat{\boldsymbol{A}}_t\boldsymbol{y} = \sum_{i=1}^{N}\hat{a}_{i,t}y_i\boldsymbol{h}_i. \tag{14}$$

The joint maximization of (12) w.r.t. $\{\tilde{\boldsymbol{x}}, \tilde{p}, \tilde{\sigma}^2\}$ can be solved as follows. First, maximization of (12) w.r.t. $\tilde{\boldsymbol{x}}$ is a weighted LS problem, whose solution $\hat{\boldsymbol{x}}_{t+1}$ is that of the linear system

$$\hat{\boldsymbol{\Gamma}}_t \hat{\boldsymbol{x}}_{t+1} = \hat{\boldsymbol{\psi}}_t. \tag{15}$$

Then, maximization of (12) w.r.t $\tilde{p}$ and $\tilde{\sigma}^2$ yields

$$\begin{aligned} \hat{p}_{t+1} &= \frac{1}{N}\sum_{i=1}^{N} \hat{a}_{i,t}, & (16)\\ \hat{\sigma}^2_{t+1} &= \frac{1}{N}[\boldsymbol{y}^T\boldsymbol{y} - \hat{\boldsymbol{\psi}}_t^T \hat{\boldsymbol{x}}_{t+1}] \\ &= \frac{1}{N}\sum_{i=1}^{N} \left[y_i^2 - \hat{a}_{i,t} y_i \boldsymbol{h}_i^T \hat{\boldsymbol{x}}_{t+1}\right]. & (17) \end{aligned}$$

Observe that global information is required in order to compute (15)-(17), i.e., one needs $\{y_i, \hat{a}_{i,t}, \boldsymbol{h}_i\}$ for all i. In Sec. 4 we will introduce a distributed implementation of the EM algorithm which is based on the combination of diffusion and consensus strategies.

To close this section, we rephrase the centralized EM iteration above in a way that will be useful in the sequel. Let $P = \frac{L(L+3)}{2} + 2$, and introduce the $P \times 1$ vector

$$\hat{\boldsymbol{\chi}}_t \triangleq \frac{1}{N}\left[\begin{array}{cccc} \|\boldsymbol{y}\|^2 & \mathbf{1}^T\hat{\boldsymbol{a}}_t & \hat{\boldsymbol{\psi}}_t^T & \mathrm{vec}\left\{\hat{\boldsymbol{\Gamma}}_t\right\}^T \end{array}\right]^T. \tag{18}$$

Then, given $\hat{\boldsymbol{\theta}}_t$, one computes $\hat{\boldsymbol{a}}_t$ by means of (11), after which $\hat{\boldsymbol{\chi}}_t$ is obtained via (13)-(14). Thus, we can write $\hat{\boldsymbol{\chi}}_t = g_1(\hat{\boldsymbol{\theta}}_t)$. On the other hand, it is seen from (15)-(17) that the parameter estimate $\hat{\boldsymbol{\theta}}_{t+1}$ can be directly computed from $\hat{\boldsymbol{\chi}}_t$, i.e., $\hat{\boldsymbol{\theta}}_{t+1} = g_2(\hat{\boldsymbol{\chi}}_t)$. Putting it all together, we can rewrite (8) as $\hat{\boldsymbol{\theta}}_{t+1} = (g_2 \circ g_1)(\hat{\boldsymbol{\theta}}_t)$ or, in terms of $\hat{\boldsymbol{\chi}}_t$, as

$$\hat{\boldsymbol{\chi}}_{t+1} = g(\hat{\boldsymbol{\chi}}_t) \quad \text{with} \quad g \triangleq g_1 \circ g_2. \tag{19}$$

Suppose that $\hat{\boldsymbol{\theta}}_\star$ is a fixed point of the EM iteration: $\hat{\boldsymbol{\theta}}_\star = (g_2 \circ g_1)(\hat{\boldsymbol{\theta}}_\star)$. Then $\hat{\boldsymbol{\chi}}_\star = g_1(\hat{\boldsymbol{\theta}}_\star)$ is a fixed point of (19). Moreover, if $\hat{\boldsymbol{\theta}}_\star$ is asymptotically convergent, so is $\hat{\boldsymbol{\chi}}_\star$[46]. As it will be seen later in Section 5, this alternative way of expressing the centralized EM iteration as an update of the entries of vector $\hat{\boldsymbol{\chi}}_t$ through the mapping $g(\cdot)$ will be used to analize the convergence of the proposed distributed implementation of the EM algorithm.

4. A Diffusion-Averaging Distributed EM Estimator

The proposed distributed implementation of the EM estimator hinges on the fact that in the centralized version the information from the different nodes is aggregated by means of averages, as can be seen in (13)-(17). This property is similar to that used in [38] for distributed computation of a Least Squares estimate. However, in contrast with [38], in our estimation

problem not all of the quantities to be averaged are available at the nodes from the very beginning; rather, they depend on the variables $\hat{a}_{i,t}$ which are updated over time. Because of this, it becomes necessary to incorporate a *diffusion* mechanism together with a *consensus averaging* procedure, analogous to that from [38], as described next.

Thus, consider a WSN with N nodes, such that each node can only communicate with neighboring nodes located within a small area. The information flow among the nodes of the network is described by means of an undirected graph $\mathcal{G}=\{V,E\}$, where V is the set of vertices or nodes and E is the set of bidirectional edges or links e_{ij} $\forall\{i,j\}\in V$ with $e_{ij}=e_{ji}$ [39]. The set of neighbors of node i is denoted as $\mathcal{N}_i=\{j\in V : e_{ij}\in E\}$ for all $i\in\{1,\cdots,N\}$. We further assume that the network is connected, such that there exists a path between any pair of nodes $\{i,j\}\in V$. Consider then a *weight matrix* $\boldsymbol{W}\in\mathbb{R}^{N\times N}$, related to the topology of the underlying graph model, with a nonzero $\{i,j\}^{\text{th}}$ entry W_{ij} only if $j\in\mathcal{N}_i$, and satisfying the following conditions [40]:

Assumption 1. *The weight matrix $\boldsymbol{W}$ is symmetric and satisfies:*

$$\boldsymbol{W}\mathbf{1}=\mathbf{1},\quad \rho(\boldsymbol{W}-\boldsymbol{J})<1, \tag{20}$$

where $\mathbf{1}$ is an all-ones vector of length N, and

$$\boldsymbol{J}\triangleq\frac{1}{N}\mathbf{1}\mathbf{1}^T \tag{21}$$

is the orthogonal projector onto the one-dimensional subspace spanned by $\mathbf{1}$.

Thus, the largest eigenvalue of $\boldsymbol{W}$ equals 1 with algebraic multiplicity one, a fact that is key to ensuring that a global consensus is achieved throughout the network. A right eigenvector $\mathbf{1}$ associated with the eigenvalue 1 implies that after reaching a consensus the network will remain in consensus, and a left eigenvector $\mathbf{1}$ implies that the average of the state vector is preserved from iteration to iteration. Moreover, the symmetry of $\boldsymbol{W}$ reflects the fact that the information flows in both directions of a link.

The proposed diffusion-averaging scheme is as follows. Each node i keeps track of local estimates $\hat{\boldsymbol{x}}_{i,k}$, $\hat{\sigma}^2_{i,k}$, $\hat{p}_{i,k}$ at every iteration[2] k. From these, a *soft* estimate $\hat{\varphi}_{i,k}$ of the a posteriori expected value of a_i at node i and at time k is computed as follows:

$$\hat{\varphi}_{i,k}=\frac{\hat{p}_{i,k}\cdot\exp\left\{-\frac{\left(y_i-\boldsymbol{h}_i^T\hat{\boldsymbol{x}}_{i,k}\right)^2}{2\hat{\sigma}^2_{i,k}}\right\}}{\hat{p}_{i,k}\cdot\exp\left\{-\frac{\left(y_i-\boldsymbol{h}_i^T\hat{\boldsymbol{x}}_{i,k}\right)^2}{2\hat{\sigma}^2_{i,k}}\right\}+(1-\hat{p}_{i,k})\cdot\exp\left\{-\frac{y_i^2}{2\hat{\sigma}^2_{i,k}}\right\}}. \tag{22}$$

[2]To stress the difference with respect to the centralized approach, the iteration index for the distributed algorithm is denoted by k rather than t.

Notice the main difference between $\hat{a}_{i,t}$ in (11) and $\hat{\varphi}_{i,k}$ in (22): whereas $\hat{a}_{i,t}$ is computed using *global* estimates, that is $\hat{\boldsymbol{x}}_t$, $\hat{\sigma}_t^2$ and $\hat{p}_t$ in (15)-(17), computation of $\hat{\varphi}_{i,k}$ only uses *local* information, namely the initially available $\{y_i, \boldsymbol{h}_i\}$ and the current local estimates $\hat{\boldsymbol{x}}_{i,k}$, $\hat{\sigma}_{i,k}^2$ and $\hat{p}_{i,k}$.

Next, the information at each node is appropriately diffused over the network via local communication among neighbors, so that each node can in turn update its local estimates and reach an agreement asymptotically. To this end, node i computes the following $P = \frac{L(L+3)}{2} + 2$ auxiliary variables:

$$\begin{aligned} f_{i,k}^{y} &= y_i^2, \qquad & f_{i,k}^{\hat{\boldsymbol{\psi}}(l)} &= \hat{\varphi}_{i,k} y_i h_i(l), \\ f_{i,k}^{a} &= \hat{\varphi}_{i,k}, & f_{i,k}^{\hat{\boldsymbol{\Gamma}}(l,m)} &= \hat{\varphi}_{i,k} h_i(l) h_i(m), \end{aligned} \tag{23}$$

with $1 \leq l \leq m \leq L$. These can be seen as local contributions, up to a factor of $\frac{1}{N}$, to the entries of the vector $\hat{\boldsymbol{\chi}}_t$ featuring in the centralized EM iteration, see (18). For each of these variables $f_{i,k}^{\nu}$, with the index $\nu \in \mathcal{V}$ belonging in the set

$$\mathcal{V} = \{y, a, \hat{\boldsymbol{\psi}}(1), \cdots, \hat{\boldsymbol{\psi}}(L), \hat{\boldsymbol{\Gamma}}(1,1), \hat{\boldsymbol{\Gamma}}(1,2), \cdots, \hat{\boldsymbol{\Gamma}}(L,L)\}, \tag{24}$$

a corresponding variable $\phi_{i,k}^{\nu}$ is kept. Then, given $f_{i,k}^{\nu}$ and the previous value $\phi_{i,k-1}^{\nu}$, node i computes

$$\phi_{i,k-1}^{\nu} + \alpha_k (f_{i,k}^{\nu} - \phi_{i,k-1}^{\nu}) \tag{25}$$

with $\alpha_k > 0$ a suitable stepsize sequence. The values in (25) are then exchanged among neighboring nodes, after which $\phi_{i,k}^{\nu}$ is updated at node i via spatial averaging as follows:

$$\phi_{i,k}^{\nu} = \sum_{j \in \mathcal{N}_i} W_{ij} \left(\phi_{j,k-1}^{\nu} + \alpha_k (f_{j,k}^{\nu} - \phi_{j,k-1}^{\nu}) \right). \tag{26}$$

Thus, each node i computes a pair of local variables $(f_{i,k}^{\nu}, \phi_{i,k}^{\nu})$ $\forall \nu \in \mathcal{V}$ for each one of the entries of vector $\hat{\boldsymbol{\chi}}_t$ in (18). Whereas variables $f_{i,k}^{\nu}$ are the local contribution to the corresponding entries of vector $\hat{\boldsymbol{\chi}}_t$ upon substituting $\hat{a}_{i,t}$ by $\hat{\varphi}_{i,k}$, variables $\phi_{i,k}^{\nu}$ are their counterparts after combining the values from neighboring nodes via (26). Once (26) are computed for all $\nu \in \mathcal{V}$, the local estimates $\hat{\boldsymbol{x}}_{i,k+1}$, $\hat{\sigma}_{i,k+1}^2$ and $\hat{p}_{i,k+1}$ are updated as follows:

$$\hat{\boldsymbol{\Gamma}}_{i,k} \hat{\boldsymbol{x}}_{i,k+1} = \hat{\boldsymbol{\psi}}_{i,k}, \tag{27}$$

$$\hat{p}_{i,k+1} = \phi_{i,k}^{a}, \tag{28}$$

$$\hat{\sigma}_{i,k+1}^2 = \phi_{i,k}^{y} - \hat{\boldsymbol{\psi}}_{i,k}^T \hat{\boldsymbol{x}}_{i,k+1}. \tag{29}$$

where

$$\hat{\boldsymbol{\Gamma}}_{i,k}(l,m) = \phi_{i,k}^{\hat{\boldsymbol{\Gamma}}(l,m)} \text{ and } \hat{\boldsymbol{\psi}}_{i,k}(l) = \phi_{i,k}^{\hat{\boldsymbol{\psi}}(l)}. \tag{30}$$

for $1 \leq l \leq m \leq L$, and with $\hat{\boldsymbol{\Gamma}}_{i,k}^T = \hat{\boldsymbol{\Gamma}}_{i,k}$. This procedure is repeated until convergence. For the sake of clarity, Table 1 summarizes the proposed DA-DEM algorithm. For the initialization, in

the absence of any a priori knowledge about the probability p of a transducer failure, we choose to set $\hat{\varphi}_{i,0} = \frac{1}{2}\ \forall i$.

Note that DA-DEM requires an exchange of $O(L^2)$ scalar quantities per iteration among neighboring nodes that is carried out at the so-called *Diffusion-Averaging* step (see Table 1). The distributed EM in [22] would need the same communication overhead as DA-DEM of $O(L^2)$ parameters at each iteration but, whereas DA-DEM just requires a connected graph, the sequential updating strategy used in [22] demands for a cyclic topology. With regard to the other relevant distributed EM method in [28], we note that it has a lower communication overhead of $O(L)$ parameters per iteration, but at the cost of a much higher computational load. That is, whereas in DA-DEM each node has to solve a linear equation system with a typical cost of $O(L^3)$ operations, the distributed EM method in [28] is based on the ADMM algorithm and requires each node to solve an optimization problem with $O(L)$ unknowns via, e.g. interior point methods. Note that the methods in [22, 28] address a different problem of Gaussian mixture density estimation and clustering and, therefore, the communication overhead and computational cost comparison has been done assuming they were appropriately modified to solve the estimation of $\boldsymbol{x}$ in (2).

In order to gain some insight into the behavior of the DA-DEM algorithm, let us define for each $\nu \in \mathcal{V}$ in (24) the vectors gathering the local variables at time k, i.e.,

$$\boldsymbol{\phi}_k^\nu \triangleq \left[\begin{array}{cccc} \phi_{1,k}^\nu & \phi_{2,k}^\nu & \cdots & \phi_{N,k}^\nu \end{array} \right]^T, \tag{31}$$

$$\boldsymbol{f}_k^\nu \triangleq [\begin{array}{cccc} f_{1,k}^\nu & f_{2,k}^\nu & \cdots & f_{N,k}^\nu \end{array}]^T. \tag{32}$$

According to (26), $\boldsymbol{\phi}_k^\nu$ evolves as follows:

$$\begin{aligned} \boldsymbol{\phi}_k^\nu &= \boldsymbol{W}\left((1-\alpha_k)\boldsymbol{\phi}_{k-1}^\nu + \alpha_k \boldsymbol{f}_k^\nu\right) \\ &= (1-\alpha_k)\boldsymbol{W}\boldsymbol{\phi}_{k-1}^\nu + \alpha_k \boldsymbol{W}\boldsymbol{f}_k^\nu, \quad k \geq 1 \end{aligned} \tag{33}$$

where $\alpha_1 = 1$ and $\alpha_k \to 0$. Although initialization of $\boldsymbol{\phi}_k^\nu$ is irrelevant as long as $\alpha_1 = 1$, we assume for convenience that $\boldsymbol{\phi}_0^\nu = \mathbf{0}$ for all $\nu \in \mathcal{V}$. As seen in (33), $\boldsymbol{\phi}_k^\nu$ is a convex combination of two terms, $\boldsymbol{W}\boldsymbol{\phi}_{k-1}^\nu$ and $\boldsymbol{W}\boldsymbol{f}_k^\nu$. The term $\boldsymbol{W}\boldsymbol{f}_k^\nu$ is responsible for the diffusion over the network of the updated local information. On the other hand, the purpose of the term $\boldsymbol{W}\boldsymbol{\phi}_{k-1}^\nu$ is to drive the state vector $\boldsymbol{\phi}_k^\nu$ toward a consensus, so that all nodes reach the same values for their estimates (27)-(29). With $\alpha_1 = 1$ and $\alpha_k \to 0$, the diffusion term in (33) is dominant at the beginning of the process. Then, as time progresses, this diffusion term gradually "turns off" and the consensus term becomes dominant, in order to drive the network towards agreement.

It must be emphasized that, once the observations $\{y_i\}$ are given, and assuming a deterministic schedule for the stepsize sequence $\{\alpha_k\}$, the DA-DEM algorithm as detailed in Table 1

Table 1: The Diffusion-Averaging Distributed EM (DA-DEM) Algorithm

For $i = 1, \cdots, N$, initialize $\hat{\varphi}_{i,0} = \frac{1}{2}$ and

$$\hat{p}_{i,1} = \hat{\varphi}_{i,0}, \quad \hat{\boldsymbol{x}}_{i,1} = \frac{y_i \boldsymbol{h}_i}{\boldsymbol{h}_i^T \boldsymbol{h}_i}, \quad \hat{\sigma}^2_{i,1} = y_i^2(1 - \hat{\varphi}_{i,0}).$$

For $k \geq 1$ and $\forall i$

1. *E-Step*: given $\hat{\boldsymbol{x}}_{i,k}, \hat{\sigma}^2_{i,k}$ and $\hat{p}_{i,k}$, compute the a posteriori probabilities $\hat{\varphi}_{i,k}$ as

$$\hat{\varphi}_{i,k} \;=\; \frac{\hat{p}_{i,k} \cdot \exp\left\{-\frac{\left(y_i - \boldsymbol{h}_i^T \hat{\boldsymbol{x}}_{i,k}\right)^2}{2\hat{\sigma}^2_{i,k}}\right\}}{\hat{p}_{i,k} \cdot \exp\left\{-\frac{\left(y_i - \boldsymbol{h}_i^T \hat{\boldsymbol{x}}_{i,k}\right)^2}{2\hat{\sigma}^2_{i,k}}\right\} + (1 - \hat{p}_{i,k}) \cdot \exp\left\{-\frac{y_i^2}{2\hat{\sigma}^2_{i,k}}\right\}}.$$

2. *Diffusion-Averaging Step*: for each index $\nu \in \mathcal{V}$, being

$$\mathcal{V} = \{y, a, \hat{\boldsymbol{\psi}}(1), \cdots, \hat{\boldsymbol{\psi}}(L), \hat{\boldsymbol{\Gamma}}(1,1), \hat{\boldsymbol{\Gamma}}(1,2), \cdots, \hat{\boldsymbol{\Gamma}}(L,L)\},$$

compute the auxiliary variables $f^{\nu}_{i,k}$ as

$$\begin{aligned} f^{y}_{i,k} &= y_i^2, & f^{\hat{\boldsymbol{\psi}}(l)}_{i,k} &= \hat{\varphi}_{i,k} y_i h_i(l), \\ f^{a}_{i,k} &= \hat{\varphi}_{i,k}, & f^{\hat{\boldsymbol{\Gamma}}(l,m)}_{i,k} &= \hat{\varphi}_{i,k} h_i(l) h_i(m), \end{aligned}$$

for $1 \leq l \leq m \leq L$, and then update

$$\phi^{\nu}_{i,k} \;=\; \sum_{j=1}^{N} W_{ij}\big((1-\alpha_k)\phi^{\nu}_{j,k-1} + \alpha_k f^{\nu}_{j,k}\big)\,,$$

for suitable nonnegative stepsizes $\alpha_k \to 0$ with $\alpha_1 = 1$. Note that this step entails the exchange of local variables among neighbouring nodes.

3. *M-Step*: for $1 \leq l \leq m \leq L$, set $\hat{\boldsymbol{\Gamma}}_{i,k}(l,m) = \phi^{\hat{\boldsymbol{\Gamma}}(l,m)}_{i,k}$ and $\hat{\boldsymbol{\psi}}_{i,k}(l) = \phi^{\hat{\boldsymbol{\psi}}(l)}_{i,k}$. Solve for $\hat{\boldsymbol{x}}_{i,k+1}$ in the linear system

$$\hat{\boldsymbol{\Gamma}}_{i,k}\, \hat{\boldsymbol{x}}_{i,k+1} = \hat{\boldsymbol{\psi}}_{i,k},$$

with $\hat{\boldsymbol{\Gamma}}^T_{i,k} = \hat{\boldsymbol{\Gamma}}_{i,k}$, and update

$$\hat{p}_{i,k+1} = \phi^{a}_{i,k}, \quad \hat{\sigma}^2_{i,k+1} = \phi^{y}_{i,k} - \hat{\boldsymbol{\psi}}^T_{i,k} \hat{\boldsymbol{x}}_{i,k+1}.$$

4. *Repeat* steps 1, 2 and 3 until convergence.

is a completely deterministic process. Consequently, the convergence analysis presented in the following section is carried out under a purely deterministic framework.

5. Local Convergence Analysis

We analyze now the convergence properties of the DA-DEM algorithm derived in Sec. 4. Recall from (33) that the step-size sequence α_k governs the diffusion/consensus process, gradually switching from one to the other as long as this sequence converges to zero. The use of vanishing step-sizes is common in stochastic approximation [47] and it is found also in consensus applications with noisy signals [42, 43]. In particular, we consider the following choice:

$$\alpha_k = \frac{\rho}{k+\rho-1}, \quad \rho > 0, \quad k = 1, 2, \ldots \tag{34}$$

Note that $\alpha_1 = 1$ and that α_k is positive and monotonically decreasing to zero at a rate of k^{-1}. The larger the value of the user-selectable constant ρ, the more slowly α_k decays to zero, thus delaying the onset of the consensus averaging process in (33).

We note that the choice of stepsize sequence (34) is fundamentally different from those in [18, 35], which replace the term $1-\alpha_k$ in (33) by $1-\beta_k$, with β_k converging to zero at a slower rate than α_k. This choice has important and far-reaching consequences, because it results in the state variables $\boldsymbol{\phi}_k^\nu$ in (33) not converging to zero as $k \to \infty$, which was the case with the method from [35]. This difference in behavior is due to the alternative choice of stepsize sequence (34) with respect to that in [35].

The convergence analysis is carried out in two steps. First, Theorem 1 shows that the state variables $\phi_{i,k}^\nu$ asymptotically converge to a consensus among the nodes. Then Theorem 2 shows that, under a mild technical requirement, an asymptotically stable equilibrium of the centralized EM iteration of Section 3 is an asymptotically convergent point of the DA-DEM algorithm. In order to proceed, let us first introduce the folowing decomposition of $\boldsymbol{\phi}_k^\nu$:

$$\boldsymbol{\phi}_k^\nu = \boldsymbol{\eta}_k^\nu + \boldsymbol{\zeta}_k^\nu, \quad \text{with} \begin{cases} \boldsymbol{\eta}_k^\nu & \triangleq & \boldsymbol{J}\boldsymbol{\phi}_k^\nu, \\ \boldsymbol{\zeta}_k^\nu & \triangleq & (\boldsymbol{I}-\boldsymbol{J})\boldsymbol{\phi}_k^\nu. \end{cases} \tag{35}$$

Note that this decomposition is orthogonal, i.e., $(\boldsymbol{\eta}_k^\nu)^T\boldsymbol{\zeta}_k^\nu = 0$, and that one can write $\boldsymbol{\eta}_k^\nu = \bar{\phi}_k^\nu \mathbf{1}$, where

$$\bar{\phi}_k^\nu \triangleq \frac{1}{N}\mathbf{1}^T\boldsymbol{\phi}_k^\nu \tag{36}$$

is the average of the values of $\phi_{i,k}^\nu$ across nodes. Therefore, $\boldsymbol{\eta}_k^\nu$ can be thought of as the "consensus" component of vector $\boldsymbol{\phi}_k^\nu$, whereas $\boldsymbol{\zeta}_k^\nu$ represents the"consensus error" or "deviation from consensus" component.

The following result given by Theorem 1, whose proof is in Appendix A, states that, for all ν, the consensus error sequences $\boldsymbol{\zeta}_k^\nu$ approach zero as $k \to \infty$. Or, equivalently, that for all ν

the sequences $\phi_{i,k}^{\nu}$, $i = 1, \ldots, N$ tend to a consensus as $k \to \infty$, which is given by the average of the entries of $\boldsymbol{\phi}_k^{\nu}$.

Theorem 1. *Consider the DA-DEM algorithm from Table 1 with the choice of stepsize* (34). *Then, under Assumption 1,*

$$\lim_{k\to\infty} \boldsymbol{\zeta}_k^{\nu} = \lim_{k\to\infty} [\boldsymbol{\phi}_k^{\nu} - \boldsymbol{J}\boldsymbol{\phi}_k^{\nu}] = \mathbf{0} \tag{37}$$

for all $\nu \in \mathcal{V}$ *with* $\mathcal{V}$ *as in* (24).

After establishing asymptotic consensus via Theorem 1, we now focus on the asymptotic properties of $\bar{\phi}_k^{\nu}$ in (36) as $k \to \infty$, which are ultimately provided in Theorem 2. Before that, however, we establish a relation between the mapping of both the centralized EM iteration and the DA-DEM iteration. In order to do so, first let $\bar{\boldsymbol{\phi}}_k \in \mathbb{R}^{P\times 1}$ comprise all of these average variables $\{\bar{\phi}_k^{\nu}\,,\, \nu \in \mathcal{V}\}$. Premultiplying (33) by $\frac{1}{N}\mathbf{1}^T$, it is readily found that

$$\bar{\boldsymbol{\phi}}_k = (1-\alpha_k)\bar{\boldsymbol{\phi}}_{k-1} + \alpha_k \bar{\boldsymbol{f}}_k, \tag{38}$$

where $\bar{\boldsymbol{f}}_k \in \mathbb{R}^{P\times 1}$ comprises P variables $\{\bar{f}_k^{\nu}\,,\, \nu \in \mathcal{V}\}$ defined, similarly to (36), as the average of the entries of $\boldsymbol{f}_k^{\nu}$:

$$\bar{f}_k^{\nu} \triangleq \frac{1}{N}\mathbf{1}^T \boldsymbol{f}_k^{\nu}. \tag{39}$$

Note that $\bar{\boldsymbol{f}}_k$ can be seen as the counterpart of $\hat{\boldsymbol{\chi}}_t$ from (18), but using the local variables $\hat{\varphi}_{i,k}$ rather than the $\hat{a}_{i,t}$ variables of the centralized EM method. Indeed, upon defining $\hat{\boldsymbol{\varphi}}_k \triangleq [\,\hat{\varphi}_{1,k}\ \hat{\varphi}_{2,k}\ \cdots\ \hat{\varphi}_{N,k}\,]^T$ and $\boldsymbol{\Phi}_k \triangleq \mathrm{diag}\,\{\,\hat{\varphi}_{1,k}\ \hat{\varphi}_{2,k}\ \cdots\ \hat{\varphi}_{N,k}\,\}$, in view of (23) one can write

$$\bar{\boldsymbol{f}}_k = \frac{1}{N}\left[\ \|\boldsymbol{y}\|^2\ \ \mathbf{1}^T\hat{\boldsymbol{\varphi}}_k\ \ (\boldsymbol{H}^T\boldsymbol{\Phi}_k\boldsymbol{y})^T\ \ \mathrm{vec}\left\{\boldsymbol{H}^T\boldsymbol{\Phi}_k\boldsymbol{H}\right\}^T\ \right]^T, \tag{40}$$

which is seen to have the same structure as $\hat{\boldsymbol{\chi}}_t$ in (18). Given that the centralized EM iteration can be written in terms of $\hat{\boldsymbol{\chi}}_t$ via the mapping $g(\cdot)$ in (19), it is one's hope that DA-DEM will drive $\bar{\boldsymbol{f}}_k$ toward a fixed point of (19), i.e., a fixed point of the centralized EM method. To this end, first we expose the relationship between $\bar{\boldsymbol{f}}_k$ and $\bar{\boldsymbol{\phi}}_{k-1}$ through the mapping $g(\cdot)$ in the following lemma, whose proof is given in Appendix B.

Lemma 1. *Let* $g : \mathbb{R}^P \to \mathbb{R}^P$ *be the map of the centralized EM iteration as defined in* (19). *The vector sequence* $\{\bar{\boldsymbol{f}}_k\}$ *satisfies the relation*

$$\bar{\boldsymbol{f}}_k = g(\bar{\boldsymbol{\phi}}_{k-1}) + \bar{\boldsymbol{\xi}}_{k-1}, \tag{41}$$

where the sequence $\bar{\boldsymbol{\xi}}_k$ *converges to zero:*

$$\lim_{k\to\infty} \bar{\boldsymbol{\xi}}_k = \mathbf{0}. \tag{42}$$

It follows from Lemma (1) that the sequence $\bar{\boldsymbol{f}}_k$ converges if $\bar{\boldsymbol{\phi}}_k$ converges; moreover, if $\bar{\boldsymbol{\phi}}_k$ converges to a fixed point of g, then $\bar{\boldsymbol{f}}_k$ will converge to the same point. Substituting now (41) in (38), one has

$$\bar{\boldsymbol{\phi}}_k = (1-\alpha_k)\bar{\boldsymbol{\phi}}_{k-1} + \alpha_k g(\bar{\boldsymbol{\phi}}_{k-1}) + \alpha_k \bar{\boldsymbol{\xi}}_{k-1}, \tag{43}$$

which constitutes a *nonlinear, nonautonomous* (i.e., time-varying), *forced* discrete-time dynamical system [46] with state $\bar{\boldsymbol{\phi}}_{k-1}$ and input $\bar{\boldsymbol{\xi}}_{k-1}$. The associated *unforced* system is given by

$$\begin{aligned} \bar{\boldsymbol{\phi}}_k &= (1-\alpha_k)\bar{\boldsymbol{\phi}}_{k-1} + \alpha_k g(\bar{\boldsymbol{\phi}}_{k-1}) \\ &\triangleq g_k(\bar{\boldsymbol{\phi}}_{k-1}). \end{aligned} \tag{44}$$

It is readily seen that if $\bar{\boldsymbol{\phi}}_\star$ is a fixed point of g, then it is also an equilibrium of the unforced system (44), since $g_k(\bar{\boldsymbol{\phi}}_\star) = (1-\alpha_k)\bar{\boldsymbol{\phi}}_\star + \alpha_k g(\bar{\boldsymbol{\phi}}_\star) = (1-\alpha_k)\bar{\boldsymbol{\phi}}_\star + \alpha_k \bar{\boldsymbol{\phi}}_\star = \bar{\boldsymbol{\phi}}_\star$ for all k. Note that the same is not true for the forced system (43), i.e., having $\bar{\boldsymbol{\phi}}_{k-1} = \bar{\boldsymbol{\phi}}_\star$ does *not* imply $\bar{\boldsymbol{\phi}}_k = \bar{\boldsymbol{\phi}}_\star$. Nevertheless, one could expect such property to hold asymptotically because, in view of Lemma 1, the input $\bar{\boldsymbol{\xi}}_{k-1}$ of the forced system (43) converges to zero. In fact, the following result shows that if $\bar{\boldsymbol{\phi}}_\star$ is an *attractive* fixed point of g, then it is also an asymptotically convergent point of the DA-DEM algorithm. The proof is given in Appendix C.

Theorem 2. *Let $\bar{\boldsymbol{\phi}}_\star$ be an asymptotically stable equilibrium of the dynamical system $\bar{\boldsymbol{\phi}}_k = g(\bar{\boldsymbol{\phi}}_{k-1})$, and assume that:*

1. *The stepsize α_k is given by* (34)*.*
2. *The Jacobian of g evaluated at $\bar{\boldsymbol{\phi}}_\star$ has all eigenvalues with magnitude less than one.*

Then $\bar{\boldsymbol{\phi}}_\star$ is an asymptotically convergent point of (43)*, in the sense that there exist an integer k_1 and a constant $\delta > 0$ such that*

$$\|\bar{\boldsymbol{\phi}}_k - \bar{\boldsymbol{\phi}}_\star\| \leq \delta \quad \textit{for some } k \geq k_1 \quad \Rightarrow \quad \lim_{n\to\infty} \bar{\boldsymbol{\phi}}_n = \bar{\boldsymbol{\phi}}_\star. \tag{45}$$

Recall from (19) that the set of attractive fixed points of g correspond to the set of convergent points of the centralized EM iteration. Hence, under the additional condition on the eigenvalues of the Jacobian matrix, it follows that these points are locally asymptotically convergent for the DA-DEM scheme with the proposed stepsize (34). Note that for an asymptotically stable equilibrium $\bar{\boldsymbol{\phi}}_\star$ of the centralized EM iteration, these eigenvalues necessarily have magnitude no larger than one [46]. Having magnitudes strictly less than one is a technical requirement for the linearization approach used in the proof given in Appendix C, and due to the fact that the linearization method is inconclusive when the Jacobian matrix presents eigenvalues with magnitude no larger than 1, with some of them having magnitude exactly 1 [46]. Whether it is possible in practice to find settings in which at least one eigenvalue has magnitude 1, and yet

the fixed point $\bar{\boldsymbol{\phi}}_\star$ of the centralized EM iteration remains asymptotically stable, is difficult to ascertain. Note that even in that case, Theorem 2 does not necessarily imply instability of the DA-DEM scheme.

6. Simulation Results

The theoretical results from Sec. 5 are supported here with computer simulations of a network composed of $N=100$ nodes randomly deployed over a unit square with connectivity radius $r_c=0.18$. The nodes sense a unit-norm parameter vector $\boldsymbol{x}\in\mathbb{R}^{L\times 1}$ with $L=3$, randomly generated and fixed throughout the simulation. Each node has access to one measurement $y_i=a_i\boldsymbol{h}_i^T\boldsymbol{x}+w_i$, with $w_i\sim\mathcal{N}(0,\sigma^2)$, $\boldsymbol{x}$ is assumed sensed with probability $p=\{0.7,0.9\}$ and $\boldsymbol{W}$ is taken as a Metropolis weight matrix [38]. In each run, the matrix $\boldsymbol{H}$ is randomly generated with zero-mean i.i.d. Gaussian entries and the a_i's are generated as Bernoulli random variables. Conditioned on $\boldsymbol{H}$ and assuming $p=1$, the SNR is

$$\text{SNR} \;=\; \frac{\boldsymbol{x}^T\boldsymbol{H}^T\boldsymbol{H}\boldsymbol{x}}{N\sigma^2} \le \frac{\|\boldsymbol{x}\|^2\|\boldsymbol{H}\|_F^2}{N\sigma^2}. \tag{46}$$

We take the upper bound in (46) as the SNR in the simulations, as it only depends on $\|\boldsymbol{H}\|_F$ and $\|\boldsymbol{x}\|$. The performance metrics used are the normalized MSE and the normalized bias, defined respectively as

$$\begin{aligned}\mathsf{NMSE}\{\hat{\boldsymbol{x}}\} &= \frac{1}{N\|\boldsymbol{x}\|_2^2}\sum_{i=1}^{N}\mathbb{E}\left[\|\hat{\boldsymbol{x}}_{i,k}-\boldsymbol{x}\|_2^2\right],\\ \mathsf{NBias}\{\hat{\boldsymbol{x}}\} &= \frac{1}{N\|\boldsymbol{x}\|_2^2}\sum_{i=1}^{N}\|\mathbb{E}[\hat{\boldsymbol{x}}_{i,k}]-\boldsymbol{x}\|_2.\end{aligned}$$

Results are averaged over 100 independent realizations for each SNR value.

Fig. 1 and Fig. 2 show respectively the NMSE and the NBias in terms of the SNR $=[5,25]$ dB for the centralized clairvoyant (CV) estimator in (3), the LS estimator in (4), the BLUE for low SNR in (5), the centralized EM (CEM) after $t=500$ iterations, and DA-DEM with $\rho=1$ and $p=0.7$ after $k=10\,000$ iterations. We use these iteration numbers to guarantee the NMSE and NBias are computed once the algorithms have converged for small SNRs. Results for the distributed algorithm based on the MDE scheme from [18] are also included, which addresses the same problem of estimating $\boldsymbol{x}$ in (2). Notice that the original MDE assumes knowledge of both p and σ^2, and relies on hard decisions on the variables a_i to estimate a scalar variable x. For the sake of comparison, the MDE results shown here are obtained with a modified version of MDE adapted to the signal model in (1), so that p and σ^2 are estimated jointly with $\boldsymbol{x}$ exactly as in Table 1 but substituting $f_{i,k}^a$ in (23) by the hard decision on a_i that MDE takes at each iteration. Observe from Fig. 1 that, whereas LS and BLUE exhibit a flooring effect

with increasing SNR due to the bias, the performance of CEM approaches that of the CV estimator. As expected, DA-DEM approaches the centralized EM solution with a slight deviation for low SNR values. The reason for this discrepancy is twofold. First, the convergence speed of DA-DEM slows down as the SNR decreases, so that a larger number of iterations is required to get as close to the asymptotic values. Second, at low SNR more realizations are needed to obtain reliable results for both CEM and DA-DEM. Still, the number of realizations were limited to 100 due to the overwhelming computational load involved in the simulation of the whole network. It can be also observed that MDE performs significantly worse than DA-DEM in terms of both NBias and NMSE.

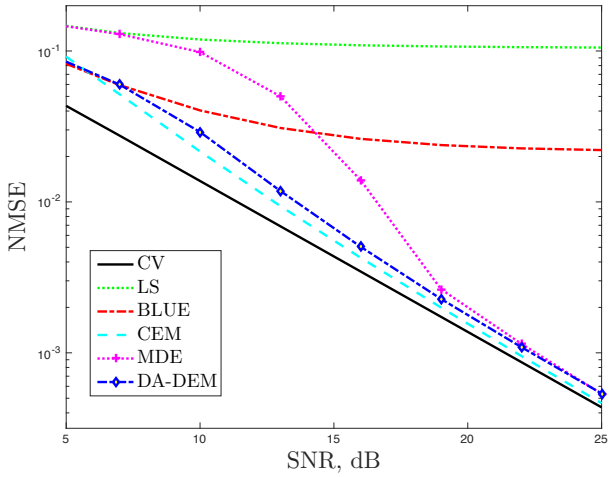

Figure 1: NMSE vs. SNR for the centralized estimators: CV, LS, BLUE and CEM, and for the distributed ones: DA-DEM and MDE.

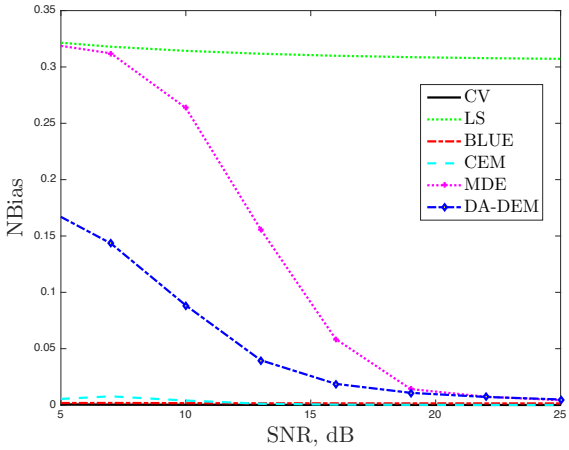

Figure 2: NBias vs. SNR for CV, LS, BLUE and CEM, and for DA-DEM and MDE.

6.1. Effect of parameter ρ

Although, as stated by Theorem 2, CEM convergent points are DA-DEM convergent points for all $\rho > 0$, the value of ρ does have an impact on the convergence speed of DA-DEM. In

this section we investigate this impact and its relation to the connectivity of the network. Fig. 3 shows the results of a single realization of DA-DEM with SNR = 20 dB and $p = 0.9$ for $\rho = 0.1$, $\rho = 1$ and $\rho = 100$. Fig. 3 (a) show the convergence of the local estimates of the three components of $\boldsymbol{x}$ for all nodes, and for illustrative purposes, the CEM estimates are depicted at the last iteration ('o'). Fig. 3 (b) show the convergence of the consensus components to the CEM estimates, $\|\boldsymbol{\eta}_k - \boldsymbol{\eta}^\star\|$, where $\boldsymbol{\eta}_k^\nu$ is defined in (35) and $\boldsymbol{\eta}^*$ is a vector containing the CEM estimates. Fig. 35 (c) show the evolution of $\|\boldsymbol{\zeta}_k\|$, i.e., the deviation from the consensus component defined in (35), vs. iterations. For the smallest ρ on top of Fig. 3 (a) the nodes reach consensus very fast, but this average is far from the CEM estimate. This bias decays slowly and is noticeable even after 10 000 iterations. With $\rho = 1$ we can see that the nodes not only reach an agreement on the estimated values, but also converge to the CEM estimate significantly faster. With $\rho = 100$, the nodes converge in average to the CEM estimate much more quickly, but with a large inter-node variability. This is because consensus among nodes becomes delayed further in time for large values of ρ, resulting in a higher variance. This is in agreement with our discussion in Sec. 5, i.e. the value of ρ should strike the right balance between allowing sufficient time for the information to diffuse over the network in the initial stage, and the kickoff of the consensus process in the final stage. Moreover, we observe that as the value of ρ increases, the convergence of the consensus components gets faster, while the convergence of the consensus component error slows down for this set of parameters.

Fig. 4 shows the NMSE curves of DA-DEM averaged over 100 independent realizations for different values of $\rho = \{0.1, 0.5, 1, 2, 3, 4\}$, with SNR $= \{10, 20\}$ dB and for two different connectivities: a more connected one with $r_c = 0.18$ and average number of neighbors $\mathcal{N}_{ave} = 8$, and a less connected one with $r_c = 0.1$ and $\mathcal{N}_{ave} = 3.3$. Fig. 4 (a) SNR $= 10$ and $r_c = 0.18$, (b) SNR $= 10$ and $r_c = 0.1$, (c) SNR $= 20$ and $r_c = 0.18$, (d) SNR $= 20$ and $r_c = 0.1$. In low SNR scenarios (Fig. 4 (a, b)), after 10 000 iterations the NMSE has not reached yet its asymptotic value (given by the NMSE obtained by CEM, shown as benchmark). In the high SNR case (Fig. 4 (c, d)), convergence of the NMSE to its asymptotic value can be observed within the simulation window of 10 000 iterations if the value of the parameter ρ is appropriately chosen. Again, a reduction in network connectivity results in slower convergence and increased sensitivity to large values of ρ, which turn on the adaptive consensus process later in time. Convergence is slower for the less connected network (Fig. 4 (b, d)), since with low network connectivity, consensus is intrinsically delayed and more iterations are needed to reach an agreement. This results in a slower decrease in NMSE due to a higher dispersion of estimates among the nodes. For the more connected network, we see in Fig. 4 (a) that $\rho = 1$ provides fastest convergence, whereas for the less connected one in Fig. 4 (b), the best value of ρ is smaller, i.e. $\rho = 0.5$. A smaller ρ speeds up the consensus process and somehow compensates for the slowdown due to

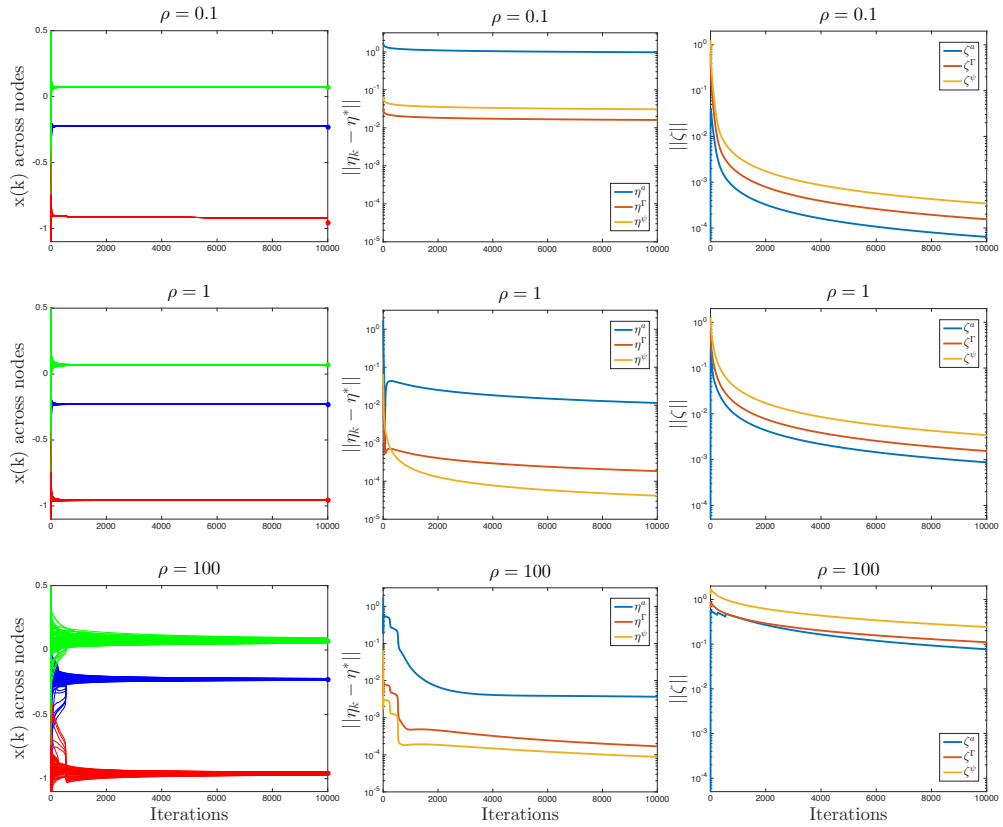

Figure 3: (a) DA-DEM estimates $\{\hat{\boldsymbol{x}}_{i,k}; \forall i = 1, \cdots, N\}$ vs. k obtained with $\rho = 0.1$, $\rho = 1$ and $\rho = 100$. CEM estimates $\hat{\boldsymbol{x}}_t$ obtained after $t = 500$ iterations are included at $k = 10\,000$ ('o'). (b) Evolution of $||\boldsymbol{\eta}_k^{\nu} - \boldsymbol{\eta}^{\star}||$ vs. k for $\rho = \{0.1, 1, 100\}$ and $\nu = \{a, \boldsymbol{\psi}, \boldsymbol{\Gamma}\}$. (c) Evolution of $\|\boldsymbol{\zeta}_k^{\nu}\|$ vs. k for $\rho = \{0.1, 1, 100\}$ and $\nu = \{a, \boldsymbol{\psi}, \boldsymbol{\Gamma}\}$.

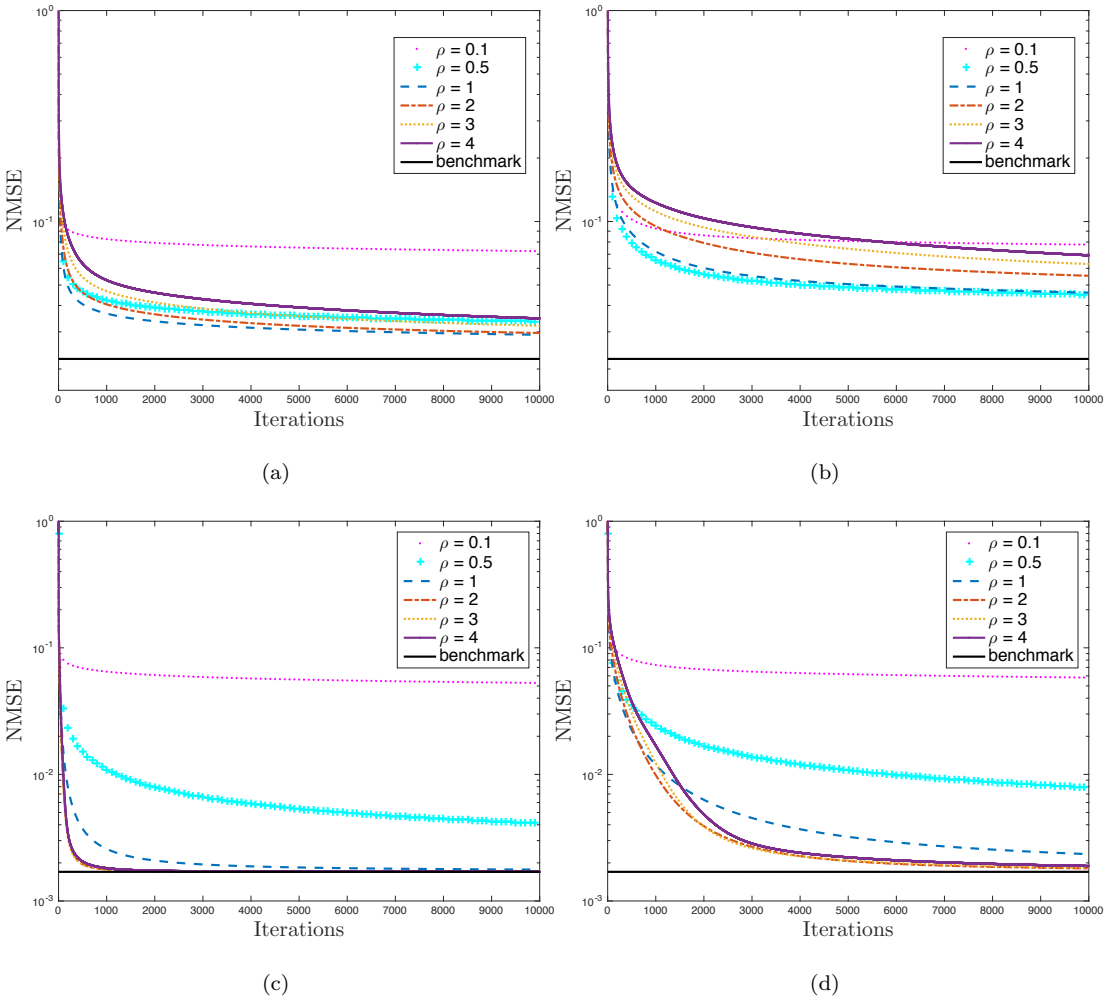

Figure 4: NMSE of the DA-DEM algorithm for $p = 0.7$ and different values of ρ. The NMSE reached by the CEM is included as a benchmark. (a) SNR = 10 dB and $r_c = 0.18$. (b) SNR = 10 dB and $r_c = 0.1$. (c) SNR = 20 and $r_c = 0.18$. (d) SNR = 20 and $r_c = 0.1$.

a reduction in connectivity.

Fig. 5 shows the results for the same values of ρ and SNR in both deployments but considering instead $p = 0.9$. We observe that the NMSE is reduced in all scenarios with respect to the previous results: (a) SNR $= 10$ and $r_c = 0.18$, (b) SNR $= 10$ and $r_c = 0.1$, (c) SNR $= 20$ and $r_c = 0.18$, and (d) SNR $= 20$ and $r_c = 0.1$. Whereas the behavior of the NMSE according to the parameters is consistent with the previous results, the NMSE is clearly reduced in all cases when the probability p is higher.

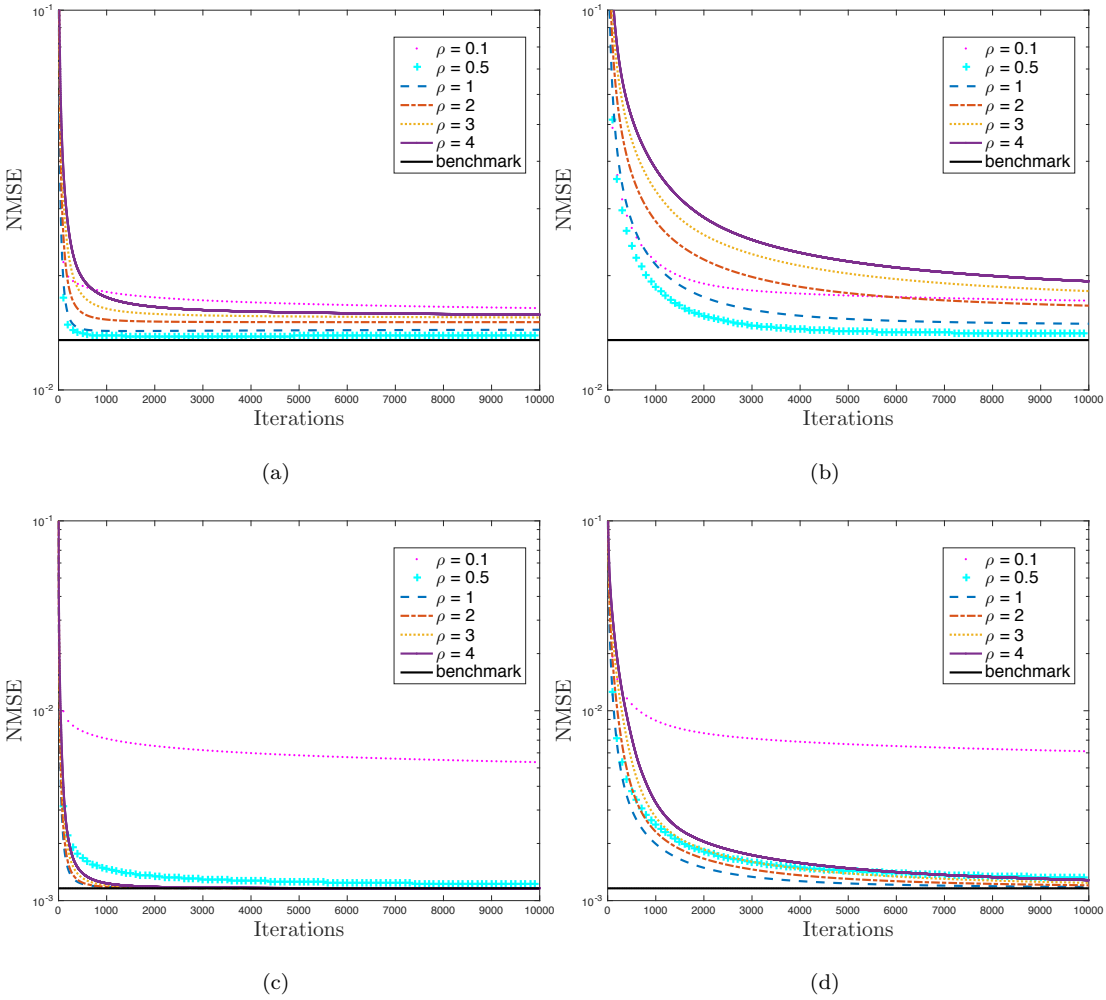

Figure 5: NMSE of the DA-DEM algorithm for $p = 0.9$ and different values of ρ. The NMSE reached by the CEM is included as a benchmark. (a) SNR = 10 dB and $r_c = 0.18$. (b) SNR = 10 dB and $r_c = 0.1$. (c) SNR = 20 and $r_c = 0.18$. (d) SNR = 20 and $r_c = 0.1$.

7. Conclusion

We have proposed a diffusion-averaging distributed EM algorithm for estimation of a vector-valued parameter with a wireless sensor network in the presence of noisy observations and with potentially faulty transducers. The DA-DEM recursion combines an initial period where the process of information diffusion is gradually switched off at the same time as an information averaging process is gradually switched on. The switching mechanism is controlled by proper choice of vanishing step-size sequences. The method requires only local exchanges of information among network nodes and, in contrast with previous approaches, it does not assume knowledge of the a priori probability of transducer failures or the noise variance.

The convergence analysis provided shows that the convergent points of the centralized EM iteration are locally asymptotically convergent points of DA-DEM. Numerical results show that with a properly tuned DA-DEM scheme it is possible to attain the performance of the centralized EM estimator at all SNR values. Ongoing work is addressing the applicability of the DA-DEM principle to more sophisticated data models.

Appendix A. Proof of Theorem 1

The update equation for the vector $\boldsymbol{\phi}_k^\nu$ defined in (33) can be expressed as

$$\boldsymbol{\phi}_k^\nu = \boldsymbol{W}\boldsymbol{\phi}_{k-1}^\nu + \alpha_k \boldsymbol{W}\left(\boldsymbol{f}_k^\nu - \boldsymbol{\phi}_{k-1}^\nu\right). \tag{A.1}$$

Introducing the weight sequence

$$w_k(n) \triangleq \alpha_n \prod_{l=n+1}^{k} (1-\alpha_l), \quad 1 \le n \le k, \tag{A.2}$$

it can be checked that the recursion above yields

$$\boldsymbol{\phi}_k^\nu = \sum_{n=1}^{k} w_k(n) \boldsymbol{W}^{k-n+1} \boldsymbol{f}_n^\nu. \tag{A.3}$$

For the choice of stepsize (34), the weights (A.2) can be written explicitly as

$$w_k(n) = \frac{\rho\Gamma(k)}{\Gamma(k+\rho)} \cdot \frac{\Gamma(n+\rho-1)}{\Gamma(n)}, \tag{A.4}$$

where $\Gamma(x) = \int_0^\infty t^{x-1}\mathrm{e}^{-t}\mathrm{d}t$ is the gamma function.

The deviation of (A.3) with respect to $\boldsymbol{J}\boldsymbol{\phi}_k^\nu$ is then

$$\boldsymbol{\zeta}_k^\nu = \boldsymbol{\phi}_k^\nu - \boldsymbol{J}\boldsymbol{\phi}_k^\nu = \sum_{n=1}^{k} w_k(n)\left(\boldsymbol{W}^{k-n+1} - \boldsymbol{J}\right)\boldsymbol{f}_n^\nu, \tag{A.5}$$

where we have used the fact that $\boldsymbol{JW} = \boldsymbol{J}$. We will show next that the right-hand side of (A.5) converges to zero. To do so, consider the eigenvalue decomposition of the symmetric weight matrix $\boldsymbol{W} = \frac{1}{N}\mathbf{1}\mathbf{1}^T + \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T$, where

$$\boldsymbol{\Lambda} = \text{diag}\,\{\lambda_2 \cdots \lambda_N\} \quad \text{with} \quad 1 > |\lambda_2| \geq \cdots \geq |\lambda_N|.$$

The inequalities above hold because $\rho(\boldsymbol{W} - \boldsymbol{J}) < 1$ by Assumption 1. The matrix $\boldsymbol{U} = [\ \boldsymbol{u}_2 \quad \boldsymbol{u}_3 \quad \cdots \quad \boldsymbol{u}_N\ ] \in \mathbb{R}^{N\times(N-1)}$ has orthonormal columns, and satisfies $\boldsymbol{U}^T\mathbf{1} = \mathbf{0}$. Therefore, for any integer n, it holds that $\boldsymbol{W}^n = \boldsymbol{J} + \boldsymbol{U}\boldsymbol{\Lambda}^n\boldsymbol{U}^T$. Using this in (A.5), and introducing

$$\tilde{\boldsymbol{f}}_k^\nu \triangleq \boldsymbol{U}^T\boldsymbol{f}_k^\nu, \qquad \tilde{\boldsymbol{s}}_k^\nu \triangleq \sum_{n=1}^{k} w_k(n)\boldsymbol{\Lambda}^{k-n+1}\tilde{\boldsymbol{f}}_n^\nu, \tag{A.6}$$

it is found that

$$\boldsymbol{\zeta}_k^\nu = \ \boldsymbol{\phi}_k^\nu - \boldsymbol{J}\boldsymbol{\phi}_k^\nu = \boldsymbol{U}\tilde{\boldsymbol{s}}_k^\nu. \tag{A.7}$$

We now show that $\tilde{\boldsymbol{s}}_k^\nu \to \mathbf{0}$. This vector can be written component-wise as

$$\tilde{s}_{i,k}^\nu = \sum_{n=1}^{k} w_k(n)\lambda_{i+1}^{k-n+1}\tilde{f}_{i,n}^\nu, \qquad i = 1, \cdots, N-1. \tag{A.8}$$

Now note that in view of (22), it holds that $0 \leq \hat{\varphi}_{i,k} \leq 1$ for all k. This in turn implies that the sequences $\{\boldsymbol{f}_k^\nu\}$ are bounded, see (23), and therefore $\tilde{\boldsymbol{f}}_k^\nu = \boldsymbol{U}^T\boldsymbol{f}_k^\nu$ are bounded as well. Thus, there exist constants $c_\nu > 0$ such that $|\tilde{f}_{i,k}^\nu| < c_\nu$ for all $\{i,k\}$. Using (A.4), it follows that, for $i = 1, \cdots, N-1$,

$$\begin{aligned} |\tilde{s}_{i,k}^\nu| \ &\leq \ c_\nu \left[\frac{\rho\Gamma(k)}{\Gamma(k+\rho)}\right] \\ &\quad \times \left[\sum_{n=1}^{k} \frac{\Gamma(n+\rho-1)}{\Gamma(n)}|\lambda_{i+1}|^{k-n+1}\right]. \end{aligned} \tag{A.9}$$

Using the following property of the gamma function [44]:

$$\lim_{x\to\infty} \frac{\Gamma(x+\alpha)}{\Gamma(x)x^\alpha} = 1, \qquad \alpha \in \mathbb{R}, \tag{A.10}$$

it follows that the first term in brackets in (A.9) goes to zero as $1/k^\rho$. To deal with the second term, we use the fact that

$$\begin{aligned} \sum_{n=1}^{k} \frac{\Gamma(n+\rho-1)}{\Gamma(n)}a^{k-n+1} \ &= \ \left(\frac{a}{a-1}\right)^\rho \left[a^k\Gamma(\rho)\right. \\ &\quad \left. - \frac{\Gamma(k+\rho)}{\Gamma(k+1)}{}_2F_1\left(k, 1-\rho; k+1; \frac{1}{a}\right)\right] \end{aligned} \tag{A.11}$$

where ${}_2F_1$ is the hypergeometric function [45]. Since

$$\lim_{k\to\infty} \frac{\Gamma(k+\rho)}{\Gamma(k+1)k^{\rho-1}} \ = \ 1, \tag{A.12}$$

$$\lim_{k\to\infty} {}_2F_1\left(k, 1-\rho; k+1; \frac{1}{a}\right) \ = \ \left(\frac{a-1}{a}\right)^{\rho-1}, \tag{A.13}$$

and given that $|\lambda_{i+1}| < 1$, $i = 1, \ldots, N-1$, it follows that

$$\sum_{n=1}^{k} \frac{\Gamma(n+\rho-1)}{\Gamma(n)} |\lambda_{i+1}|^{k-n+1} = \frac{|\lambda_{i+1}|\ k^{\rho-1}}{1-|\lambda_{i+1}|} + O(k^{\rho-2}). \tag{A.14}$$

Hence, the right-hand side of (A.9) goes to zero at a rate of k^{-1}. Thus,

$$\lim_{k\to\infty} \tilde{s}_{i,k}^{\nu} = 0, \qquad i = 1, \cdots, N-1, \tag{A.15}$$

yielding $\lim_{k\to\infty} [\boldsymbol{\phi}_k^{\nu} - \boldsymbol{J}\boldsymbol{\phi}_k^{\nu}] = \mathbf{0}$, in view of (A.7). ∎

Appendix B. Proof of Lemma 1

Let $\boldsymbol{\phi}_k \in \mathbb{R}^{PN\times 1}$ be formed by stacking all vectors $\{\boldsymbol{\phi}_k^{\nu}\,,\, \nu \in \mathcal{V}\}$; in view of (35), $\boldsymbol{\phi}_k$ is given by

$$\boldsymbol{\phi}_k = \boldsymbol{\eta}_k + \boldsymbol{\zeta}_k, \tag{B.1}$$

where $\boldsymbol{\eta}_k \in \mathbb{R}^{PN\times 1}$ and $\boldsymbol{\zeta}_k \in \mathbb{R}^{PN\times 1}$ are analogously formed by stacking the P vectors $\{\boldsymbol{\eta}_k^{\nu}\,,\, \nu \in \mathcal{V}\}$ and $\{\boldsymbol{\zeta}_k^{\nu}\,,\, \nu \in \mathcal{V}\}$ from (35), respectively. Note that, given $\boldsymbol{\phi}_{k-1}$, the i-th node (i) obtains its local estimates $\hat{\boldsymbol{x}}_{i,k}$, $\hat{p}_{i,k}$ and $\hat{\sigma}_{i,k}^2$ via (27)-(29); (ii) from these, it obtains $\hat{\varphi}_{i,k}$ via (22); and then (iii) it finally computes $f_{i,k}^{\nu}$ for $\nu \in \mathcal{V}$ as per (23). We summarize all these operations in the maps $\mathcal{G}_i^{\nu} : \mathbb{R}^{PN\times 1} \to \mathbb{R}$, $\nu \in \mathcal{V}$, so that

$$\begin{aligned} f_{i,k}^{\nu} = \mathcal{G}_i^{\nu}(\boldsymbol{\phi}_{k-1}) &= \mathcal{G}_i^{\nu}(\boldsymbol{\eta}_{k-1} + \boldsymbol{\zeta}_{k-1}) \\ &= \mathcal{G}_i^{\nu}(\boldsymbol{\eta}_{k-1}) + \xi_{i,k-1}^{\nu}, \end{aligned} \tag{B.2}$$

where in the second step we have substituted (B.1), and in the third step we have introduced the quantity

$$\xi_{i,k}^{\nu} \triangleq \mathcal{G}_i^{\nu}(\boldsymbol{\eta}_k + \boldsymbol{\zeta}_k) - \mathcal{G}_i^{\nu}(\boldsymbol{\eta}_k). \tag{B.3}$$

Now, according to (39) and using (B.2), the average values $\bar{f}_k^{\nu}$ satisfy

$$\begin{aligned} \bar{f}_k^{\nu} &= \frac{1}{N}\sum_{i=1}^{N} \mathcal{G}_i^{\nu}(\boldsymbol{\phi}_{k-1}) \\ &= \frac{1}{N}\sum_{i=1}^{N} \mathcal{G}_i^{\nu}(\boldsymbol{\eta}_{k-1}) + \frac{1}{N}\sum_{i=1}^{N} \xi_{i,k-1}^{\nu}. \end{aligned} \tag{B.4}$$

Let now $\mathcal{G}_i : \mathbb{R}^{PN\times 1} \to \mathbb{R}^P$ denote the map whose ν-th component is $\mathcal{G}_i^{\nu}$. Also, let $\boldsymbol{\xi}_{i,k} \in \mathbb{R}^{P\times 1}$ comprise the P variables $\{\xi_{i,k}^{\nu},\, \nu \in \mathcal{V}\}$, and define $\bar{\boldsymbol{\xi}}_k \triangleq \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\xi}_{i,k}$. Then, from (B.4), the vector $\bar{\boldsymbol{f}}_k \in \mathbb{R}^{P\times 1}$ comprising $\{\bar{f}_k^{\nu},\, \nu \in \mathcal{V}\}$ can be written as

$$\begin{aligned} \bar{\boldsymbol{f}}_k &= \frac{1}{N}\sum_{i=1}^{N} \mathcal{G}_i(\boldsymbol{\phi}_{k-1}) \\ &= \frac{1}{N}\sum_{i=1}^{N} \mathcal{G}_i(\boldsymbol{\eta}_{k-1}) + \bar{\boldsymbol{\xi}}_{k-1}. \end{aligned} \tag{B.5}$$

Regarding the first term in the right-hand side of (B.5), note from (35)-(36) that $\boldsymbol{\eta}_k^\nu = \boldsymbol{J}\boldsymbol{\phi}_k^\nu = \bar{\phi}_k^\nu \mathbf{1}$, so that $\boldsymbol{\eta}_{k-1}$ can be written as the Kronecker product $\boldsymbol{\eta}_{k-1} = \bar{\boldsymbol{\phi}}_{k-1} \otimes \mathbf{1}$. By inspecting eqs. (22)-(23) and (27)-(30), and in view of (40), it is readily found that

$$\frac{1}{N}\sum_{i=1}^{N} \mathcal{G}_i(\bar{\boldsymbol{\phi}}_{k-1} \otimes \mathbf{1}) = g(\bar{\boldsymbol{\phi}}_{k-1}), \tag{B.6}$$

where $g : \mathbb{R}^P \to \mathbb{R}^P$ is the map featuring in the centralized EM iteration (19). Therefore, from (B.5) and (B.6), the sought relationship (41) between $\bar{\boldsymbol{f}}_k$ and $\bar{\boldsymbol{\phi}}_{k-1}$ is obtained. Finally, from Theorem 1 one has $\lim_{k\to\infty} \boldsymbol{\zeta}_k = \mathbf{0}$; thus, since the maps $\mathcal{G}_i^\nu$ are continuous, $\lim_{k\to\infty} \xi_{i,k}^\nu = 0$ in view of the definition (B.3), and then $\bar{\boldsymbol{\xi}}_{k-1}$ in (B.5) converges to zero as stated in the lemma. ■

Appendix C. Proof of Theorem 2

Denoting the deviation of the state vector from $\bar{\boldsymbol{\phi}}_\star$ by $\boldsymbol{z}_k \triangleq \bar{\boldsymbol{\phi}}_k - \bar{\boldsymbol{\phi}}_\star$, the forced system (43) can be rewritten as

$$\boldsymbol{z}_k = (1-\alpha_k)\boldsymbol{z}_{k-1} + \alpha_k f(\boldsymbol{z}_{k-1}) + \alpha_k \bar{\boldsymbol{\xi}}_{k-1}, \tag{C.1}$$

where $f(\boldsymbol{z}) \triangleq g(\boldsymbol{z} + \bar{\boldsymbol{\phi}}_\star) - \bar{\boldsymbol{\phi}}_\star$. Let $\boldsymbol{B}$ be the Jacobian of g evaluated at $\bar{\boldsymbol{\phi}}_\star$:

$$\boldsymbol{B} = \left[\frac{\partial g}{\partial \bar{\boldsymbol{\phi}}}\right]_{\bar{\boldsymbol{\phi}}=\bar{\boldsymbol{\phi}}_\star}. \tag{C.2}$$

The fact that $\bar{\boldsymbol{\phi}}_\star$ is an asymptotically stable equilibrium of the iteration $\bar{\boldsymbol{\phi}}_k = g(\bar{\boldsymbol{\phi}}_{k-1})$ implies that (i) $g(\bar{\boldsymbol{\phi}}_\star) = \bar{\boldsymbol{\phi}}_\star$, and (ii) all eigenvalues of $\boldsymbol{B}$ have magnitude no larger than one [46]. In addition, these magnitudes are strictly less than one by assumption, i.e., $\boldsymbol{B}$ is a *stable* matrix.

Note that $f(\mathbf{0}) = \mathbf{0}$, and that the Jacobian of f at $\boldsymbol{z} = \mathbf{0}$ is also given by $\boldsymbol{B}$. Therefore, there exist positive constants c_z, δ_z such that $f_0(\boldsymbol{z}) \triangleq f(\boldsymbol{z}) - \boldsymbol{B}\boldsymbol{z}$ satisfies

$$\|\boldsymbol{z}\| \le \delta_z \quad \Rightarrow \quad \|f_0(\boldsymbol{z})\| \le c_z \cdot \|\boldsymbol{z}\|^2. \tag{C.3}$$

Our goal is to show that (C.1) asymptotically converges to the origin. We can rewrite (C.1) as

$$\boldsymbol{z}_k = \left[(1-\alpha_k)\boldsymbol{I} + \alpha_k \boldsymbol{B}\right]\boldsymbol{z}_{k-1} + \alpha_k f_0(\boldsymbol{z}_{k-1}) + \alpha_k \bar{\boldsymbol{\xi}}_{k-1}. \tag{C.4}$$

Since $\boldsymbol{B}$ is stable, there exists a symmetric positive definite matrix $\boldsymbol{P}$ such that $\boldsymbol{B}^T\boldsymbol{P}\boldsymbol{B} - \boldsymbol{P} = -\boldsymbol{I}$ [46]. Let $\boldsymbol{Q}$ be the symmetric square root of $\boldsymbol{P}$, i. e., $\boldsymbol{P} = \boldsymbol{Q}\boldsymbol{Q}^T = \boldsymbol{Q}^2$, and consider the change of variables $\boldsymbol{v}_k = \boldsymbol{Q}\boldsymbol{z}_k$. Then (C.4) becomes

$$\boldsymbol{v}_k = \left[(1-\alpha_k)\boldsymbol{I} + \alpha_k \tilde{\boldsymbol{B}}\right]\boldsymbol{v}_{k-1} + \alpha_k \tilde{f}_0(\boldsymbol{v}_{k-1}) + \alpha_k \tilde{\boldsymbol{\xi}}_{k-1}, \tag{C.5}$$

where $\tilde{\boldsymbol{\xi}}_{k-1} \triangleq \boldsymbol{Q}\bar{\boldsymbol{\xi}}_{k-1}$, and

$$\tilde{\boldsymbol{B}} \triangleq \boldsymbol{Q}\boldsymbol{B}\boldsymbol{Q}^{-1}, \qquad \tilde{f}_0(\boldsymbol{v}) \triangleq \boldsymbol{Q}f_0(\boldsymbol{Q}^{-1}\boldsymbol{v}). \tag{C.6}$$

Now, in view of (C.3), if we let

$$c_v = c_z \cdot \|\boldsymbol{Q}\| \cdot \|\boldsymbol{Q}^{-1}\|^2, \qquad \delta_v = \frac{\delta_z}{\|\boldsymbol{Q}^{-1}\|}, \tag{C.7}$$

then it holds that

$$\|\boldsymbol{v}\| \le \delta_v \quad \Rightarrow \quad \|\tilde{f}_0(\boldsymbol{v})\| \le c_v \cdot \|\boldsymbol{v}\|^2. \tag{C.8}$$

In addition, one has

$$\begin{aligned} \tilde{\boldsymbol{B}}^T\tilde{\boldsymbol{B}} &= \boldsymbol{Q}^{-1}\boldsymbol{B}^T\boldsymbol{Q} \cdot \boldsymbol{Q}\boldsymbol{B}\boldsymbol{Q}^{-1} \\ &= \boldsymbol{Q}^{-1}(\boldsymbol{P}-\boldsymbol{I})\boldsymbol{Q}^{-1} \quad = \boldsymbol{I}-\boldsymbol{P}^{-1}, \end{aligned} \tag{C.9}$$

showing that $\|\tilde{\boldsymbol{B}}\| < 1$. Now we can proceed to bound the norm of $\boldsymbol{v}_k$ in (C.5) as follows:

$$\begin{aligned} \|\boldsymbol{v}_k\| &\le \left\|(1-\alpha_k)\boldsymbol{I} + \alpha_k\tilde{\boldsymbol{B}}\right\| \|\boldsymbol{v}_{k-1}\| \\ &\quad + \alpha_k\|\tilde{f}_0(\boldsymbol{v}_{k-1})\| + \alpha_k\|\tilde{\boldsymbol{\xi}}_{k-1}\| \\ &\le [1-\alpha_k(1-\|\tilde{\boldsymbol{B}}\|)]\|\boldsymbol{v}_{k-1}\| \\ &\quad + \alpha_k\|\tilde{f}_0(\boldsymbol{v}_{k-1})\| + \alpha_k\|\tilde{\boldsymbol{\xi}}_{k-1}\|. \end{aligned} \tag{C.10}$$

Let $\mu \triangleq 1 - \|\tilde{\boldsymbol{B}}\| \in (0,1]$. Then, if $\|\boldsymbol{v}_{k-1}\| < \delta_v$, one has

$$\|\boldsymbol{v}_k\| \le (1-\mu\alpha_k)\|\boldsymbol{v}_{k-1}\| + \alpha_k c_v\|\boldsymbol{v}_{k-1}\|^2 + \alpha_k\|\tilde{\boldsymbol{\xi}}_{k-1}\|. \tag{C.11}$$

Now pick ϵ such that $0 < \epsilon < \mu$. Since $\lim_{k\to\infty}\tilde{\boldsymbol{\xi}}_k = \boldsymbol{0}$, there exists an integer k_1 such that

$$\|\tilde{\boldsymbol{\xi}}_k\| < \epsilon \cdot \min\left\{\delta_v, \frac{\mu-\epsilon}{2c_v}\right\} \quad \text{for all} \quad k \ge k_1. \tag{C.12}$$

Now let

$$\delta = \frac{\min\{\delta_v, \frac{\mu-\epsilon}{2c_v}\}}{\|\boldsymbol{Q}\|}, \tag{C.13}$$

and assume that $\|\boldsymbol{z}_{k_0}\| < \delta$ for some $k_0 \ge k_1$. We will show that this implies $\boldsymbol{z}_k \to \boldsymbol{0}$.

Note that $\|\boldsymbol{v}_{k_0}\| \le \|\boldsymbol{Q}\| \cdot \|\boldsymbol{z}_{k_0}\| < \min\{\delta_v, \frac{\mu-\epsilon}{2c_v}\}$. Let $k \ge k_0$ and assume that $\|\boldsymbol{v}_k\| \le \min\{\delta_v, \frac{\mu-\epsilon}{2c_v}\}$. Consider then the following two possible cases:

1. $\|\boldsymbol{v}_k\| < \|\tilde{\boldsymbol{\xi}}_k\|/\epsilon$. It then follows from (C.11) and (C.12) that

$$\begin{aligned} \|\boldsymbol{v}_{k+1}\| &\le \left[1-\alpha_{k+1}\left(\mu-\epsilon-c_v\frac{\|\tilde{\boldsymbol{\xi}}_k\|}{\epsilon}\right)\right]\frac{\|\tilde{\boldsymbol{\xi}}_k\|}{\epsilon} \\ &\le \left(1-\alpha_{k+1}\frac{\mu-\epsilon}{2}\right)\frac{\|\tilde{\boldsymbol{\xi}}_k\|}{\epsilon} \\ &\le \frac{\|\tilde{\boldsymbol{\xi}}_k\|}{\epsilon}. \end{aligned} \tag{C.14}$$

In particular, from (C.12), one has $\|\boldsymbol{v}_{k+1}\| \le \min\{\delta_v, \frac{\mu-\epsilon}{2c_v}\}$.

2. $\|\boldsymbol{v}_k\| \geq \|\tilde{\boldsymbol{\xi}}_k\|/\epsilon$. Then from (C.11),

$$\begin{aligned} \|\boldsymbol{v}_{k+1}\| &\leq [1 - \alpha_{k+1}(\mu - \epsilon - c_v\|\boldsymbol{v}_k\|)]\,\|\boldsymbol{v}_k\| \\ &\leq \left(1 - \alpha_{k+1}\frac{\mu-\epsilon}{2}\right)\|\boldsymbol{v}_k\| \qquad \text{(C.15)} \\ &\leq \|\boldsymbol{v}_k\|, \qquad \text{(C.16)} \end{aligned}$$

so that $\|\boldsymbol{v}_{k+1}\| \leq \min\{\delta_v, \frac{\mu-\epsilon}{2c_v}\}$ holds in this case as well.

By induction in k, it follows that

$$\|\boldsymbol{v}_k\| \leq \min\left\{\delta_v, \frac{\mu-\epsilon}{2c_v}\right\} \quad \text{for all} \quad k \geq k_0. \tag{C.17}$$

Now for each $k \geq k_0$, let us define the set

$$\mathcal{S}_k = \{n \in \mathbb{N} \mid k_0 \leq n \leq k,\ \|\boldsymbol{v}_n\| < \|\tilde{\boldsymbol{\xi}}_n\|/\epsilon\}, \tag{C.18}$$

and then let

$$j_\star(k) \quad = \quad \begin{cases} k_0, & \text{if } \mathcal{S}_k = \varnothing, \\ \max_n\{n \in \mathcal{S}_k\}, & \text{otherwise.} \end{cases} \tag{C.19}$$

Then, in view of (C.15), for $n = j_\star(k)+1, \ldots, k$ one has

$$\|\boldsymbol{v}_{n+1}\| \leq \left(1 - \alpha_{n+1}\frac{\mu-\epsilon}{2}\right)\|\boldsymbol{v}_n\|, \tag{C.20}$$

so that

$$\|\boldsymbol{v}_{k+1}\| \leq \left[\prod_{n=j_\star(k)+1}^{k}\left(1 - \alpha_{n+1}\frac{\mu-\epsilon}{2}\right)\right]\|\boldsymbol{v}_{j_\star(k)+1}\|. \tag{C.21}$$

If there exists $k' \geq k_0$ such that $\mathcal{S}_{k'}$ is nonempty (the case when no such k' exists will be dealt with shortly), then (C.14) and (C.21) yield

$$\|\boldsymbol{v}_{k+1}\| \leq \left[\prod_{n=j_\star(k)+1}^{k}\left(1 - \alpha_{n+1}\frac{\mu-\epsilon}{2}\right)\right]\frac{\|\tilde{\boldsymbol{\xi}}_{j_\star(k)}\|}{\epsilon} \tag{C.22}$$

for all $k \geq k'$. The product in brackets is always less than or equal to 1 (because each factor is), and it is to be taken as 1 whenever $j_\star(k) = k$. Substituting the stepsize values (34), this product can be written as

$$\prod_{n=j_\star(k)+1}^{k}\left(1 - \alpha_{n+1}\frac{\mu-\epsilon}{2}\right) = \frac{q(k)}{q(j_\star(k))} \leq 1, \tag{C.23}$$

where

$$q(n) \triangleq \frac{\Gamma(n+1+a\rho)}{\Gamma(n+1+\rho)} \quad \text{with} \quad a \triangleq 1 - \frac{\mu-\epsilon}{2}. \tag{C.24}$$

Observe that the sequence $j_\star(k)$ either has a limit or goes to infinity. We now analyze the behavior of $\|\boldsymbol{v}_{k+1}\|$ as $k \to \infty$ in both cases.

Suppose first that $\lim_{k\to\infty} j_\star(k) = \infty$. Then it holds that $\lim_{k\to\infty} \|\tilde{\boldsymbol{\xi}}_{j_\star(k)}\| = 0$, and since (C.22) implies $\|\boldsymbol{v}_{k+1}\| \leq \|\tilde{\boldsymbol{\xi}}_{j_\star(k)}\|/\epsilon$, we conclude that $\lim_{k\to\infty} \|\boldsymbol{v}_{k+1}\| = 0$, as desired.

On the other hand, if $\lim_{k\to\infty} j_\star(k) = j_\star < \infty$, then from (C.22)-(C.23),

$$\lim_{k\to\infty} \|\boldsymbol{v}_{k+1}\| \leq \frac{\|\tilde{\boldsymbol{\xi}}_{j_\star}\|}{\epsilon q(j_\star)} \lim_{k\to\infty} q(k). \tag{C.25}$$

Using property (A.10), and since $1 - a > 0$, $\rho > 0$, it is seen that $q(k)$ goes to zero for $k \to \infty$ as $1/(k+\rho+1)^{(1-a)\rho}$. Hence $\lim_{k\to\infty} \|\boldsymbol{v}_{k+1}\| = 0$.

Finally, if $\mathcal{S}_k = \varnothing$ (and thus $j_\star(k) = k_0$) for all $k \geq k_0$, then (C.16) and (C.21) yield, for all $k \geq k_0$,

$$\begin{aligned} \|\boldsymbol{v}_{k+1}\| &\leq \left[\prod_{n=j_\star(k)+1}^{k} \left(1 - \alpha_{n+1}\frac{\mu-\epsilon}{2}\right)\right] \|\boldsymbol{v}_{k_0}\| \\ &= \frac{q(k)}{q(k_0)}\|\boldsymbol{v}_{k_0}\|, \end{aligned} \tag{C.26}$$

which goes to zero as $k \to \infty$, similarly to (C.25).

Since $\boldsymbol{v}_k \to \boldsymbol{0}$ and $\boldsymbol{z}_k = \boldsymbol{Q}^{-1}\boldsymbol{v}_k$, we conclude that $\boldsymbol{z}_k$ goes to zero asymptotically. ∎

Acknowledgements

This work was supported by the Ministerio de Economía y Competitividad of the Spanish Government, ERDF funds [TEC2013-41315-R,TEC2015-69648-REDC,TEC2016-75067-C4-2-R,TEC2013-47020-C2-1-R, TEC2016-76409-C2-2-R]; and the Galician Government [Atlant-TIC, GRC2013/009, R2014/037].

References

[1] F. Zhao and L. Guibas, *Wireless sensor networks: an information processing approach.* San Mateo, CA: Morgan Kaufmann, 2004.

[2] I. F. Akyildiz and M. Vuran, *Wireless Sensor Networks.* New York, NY, USA: Wiley, 2010.

[3] G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing," *Proc. IEEE,* vol. 98, no. 11, pp. 1847–1864, 2010.

[4] R. Olfati-Saber, J. A. Fax and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE,* vol. 95, no. 1, pp. 215–233, 2007.

[5] J.-J. Xiao, A. Ribeiro, Z.-Q. Luo and G. B. Giannakis, "Distributed compression-estimation using wireless sensor networks," *IEEE Signal Process. Mag.,* vol. 23, no. 4, pp. 27–41, 2006.

[6] S. Boyd, A. Ghosh, B. Prabhakar and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Info. Theory,* vol. 52, no. 6, pp. 2508–2530, 2006.

[7] S. Barbarossa and G. Scutari, "Decentralized maximum-likelihood estimation for sensor networks composed of nonlinearly coupled dynamical systems," *IEEE Trans. Signal Process.,* vol. 55, no. 7, pp. 3456–3470, 2007.

[8] T. Zhao and A. Nehorai, "Information-driven distributed maximum likelihood estimation based on Gauss-Newton method in wireless sensor networks," *IEEE Trans. Signal Process.,* vol. 55, no. 9, pp. 4669–4682, 2007.

[9] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in ad hoc WSNs with noisy links - Part I: distributed estimation of deterministic signals," *IEEE Trans. Signal Process.,* vol. 56, no. 1, pp. 350–364, 2008.

[10] S. S. Stanković, M. S. Stanković, and D. M. Stipanović, "Decentralized parameter estimation by consensus based stochastic approximation," *IEEE Trans. Autom. Control,* vol. 56, no. 3, pp. 531–543, 2011.

[11] A. H. Sayed, "Diffusion adaptation over networks," in *Academic Press Library in Signal Processing, vol. 1,* R. Chellapa and S. Theodoridis, eds., Elsevier, 2014.

[12] A. S. Willsky, "A survey for design methods for failure detection in dynamic systems," *Automatica,* vol. 12, no. 6, pp. 601–611, 1976.

[13] Y. Zhang and X. Rong Li, "Detection and diagnosis of sensor and actuator failures using IMM estimator," *IEEE Trans. Aerosp. Electron. Syst.,* vol. 34, no. 4, pp. 1293–1313, 1998.

[14] K. Ni *et al.,* "Sensor network data fault types," *ACM Trans. Sensor Networks,* vol. 5, no. 3, 2009.

[15] T.-Y. Wang, L.-Y. Chang and P.-Y. Chen, "A collaborative sensor-fault detection scheme for robust distributed estimation in sensor networks," *IEEE Trans. Commun.,* vol. 57, no. 10, pp. 3045–3058, 2009.

[16] A. Mahapatro and P. M. Khilar, "Fault diagnosis in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.,* vol. 15, no. 4, pp. 2000–2026, Mar. 2013.

[17] W. Li, F. Bassi, D. Dardari, M. Kieffer and G. Pasolini, "Defective sensor identification for WSNs involving generic local outlier detection tests," *IEEE Trans. Signal Inf. Process. Netw.,* vol. 2, no. 1, pp. 29–48, Mar. 2016.

[18] Q. Zhou, S. Kar, L. Huie, and S. Cui, "Distributed estimation in sensor networks with imperfect model information: an adaptive learning-based approach," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.,* pp. 3109–3112, 2012.

[19] G. Bianchin, A. Cenedese, M. Luvisotto and G. Michieletto, "Distributed fault detection in sensor networks via clustering and consensus," *Proc. IEEE Conf. Decision and Control,* pp. 3828–3833, 2015.

[20] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from incomplete data via the EM algorithm," *J. Royal Stat. Soc.,* Series B, vol. 39, no. 1, pp. 1–38, 1977.

[21] T. K. Moon, "The Expectation-Maximization algorithm," *IEEE Signal Process. Mag.,* vol. 13, no. 6, pp. 47–60, 1996.

[22] R. D. Nowak, "Distributed EM algorithms for density estimation and clustering in sensor networks," *IEEE Trans. Signal Process.,* vol. 51, no. 8, pp. 2245–2253, 2003.

[23] J. Wolfe, A. Haghighi, and D. Klein, "Fully distributed EM for very large datasets," in *Proc. Int. Conf. Machine Learning,* pp. 1184–1191, 2008.

[24] K. Bhaduri and A. N. Srivastava, "A local scalable distributed expectation maximization algorithm for large peer-to-peer networks," in *Proc. IEEE Int. Conf. Data Mining,* pp. 31–40, 2009.

[25] B. Safarinejadian, M. B. Menhaj, and M. Karrari, "A distributed EM algorithm to estimate the parameters of a finite mixture of components," *Knowl. Inf. Syst.*, vol. 23, no. 3, pp. 267–292, 2010.

[26] W. Kowalczyk and N. Vlassis, "Newscast EM," *Adv. Neural Inf. Process. Syst.,* MIT Press, pp. 713–720, 2005.

[27] P. A. Forero, A. Cano, G. B. Giannakis, "Consensus-based distributed expectation-maximization algorithm for density estimation and classification using wireless sensor networks", in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.,* pp. 1989–1992, 2008.

[28] P. A. Forero, A. Cano, G. B. Giannakis, "Distributed clustering using wireless sensor networks", in *IEEE Journal of Sel. Top. in Signal Processing* vol. 5, no. 4, pp. 707–724, 2011.

[29] S. Silva Pereira, S. Barbarossa, and A. Pagès-Zamora, "Consensus for distributed EM-based clustering in WSNs," in *Proc. IEEE Sensor Array Multichannel Signal Process. Workshop,* pp. 45–48, 2010.

[30] D. Gu, "Distributed EM algorithm for Gaussian mixtures in sensor networks," *IEEE Trans. Neural Networks*, vol. 19, no. 7, pp. 1154–1166, 2008.

[31] B. Safarinejadian, M. B. Menhaj, and M. Karrari, "Distributed unsupervised Gaussian mixture learning for density estimation in sensor networks," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 9, pp. 2250–2260, Sep. 2010.

[32] Y. Weng, W. Xiao and L. Xie, "Diffusion-based EM algorithm for distributed estimation of Gaussian mixtures in wireless sensor networks," *Sensors,* vol. 11, no. 6, pp. 6297–316, Jan. 2011.

[33] Z. J. Towfic, J. Chen and A. H. Sayed, "Collaborative learning of mixture models using diffusion adaptation," in *Proc. IEEE Workshop Machine Learning for Signal Process.,* 2011.

[34] S. Silva Pereira, A. Pagès-Zamora and R. López-Valcarce, "A diffusion-based distributed EM algorithm for density estimation in wireless sensor networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.,* 2013.

[35] S. Silva Pereira, R. López-Valcarce, A. Pagès-Zamora, "A diffusion-based EM algorithm for distributed estimation in unreliable sensor networks," *IEEE Signal Process. Lett.*, vol. 20, pp. 595–598, 2013.

[36] S. Kar and J. M. F. Moura, "Consensus + Innovations distributed inference over networks: cooperation and sensing in networked systems," *IEEE Signal Process. Mag.,* vol. 30, no. 3, pp. 99–109, May 2013.

[37] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory.* Upper Saddle River, NJ: Prentice-Hall, 1993.

[38] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. Int. Symp. Inf. Process. Sensor Networks*, pp. 63–70, 2005.

[39] C. Godsil and G. Royle, *Algebraic graph theory.* Graduate Texts in Mathematics. Berlin, Germany: Springer-Verlag, vol. 207, 2001.

[40] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," in *Proc. IEEE Conf. Decision and Control,* vol. 5, pp. 4997–5002, 2003.

[41] L. Ljung and T. Söderström, *Theory and practice of recursive identification.* Cambridge, MA: MIT Press, 1983.

[42] Y. Hatano, A. Das and M. Mesbahi, "Agreement in presence of noise: pseudogradients on random geometric networks," in *Proc. IEEE Decision and Control Conf. and European Control Conf.,* pp. 6382–6387, 2005.

[43] 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.

[44] W. Freeden and M. Gutting, *Special functions of mathematical (geo-) physics.* Basel: Springer, 2013.

[45] M. Abramowitz and I. A. Stegun, eds. *Handbook of mathematical functions.* New York: Dover, 1965.

[46] M. Vidyasagar, *Nonlinear systems analysis.* Englewood Cliffs, NJ: Prentice-Hall, 1993.

[47] H. Kushner and G. G. Yin, *Stochastic approximation and recursive algorithms and applications.* Springer Science & Business Media, 2003.