A DIFFUSION-BASED DISTRIBUTED EM ALGORITHM FOR DENSITY ESTIMATION IN WIRELESS SENSOR NETWORKS

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

*SPCOM Group, Universitat Politècnica de Catalunya-Barcelona Tech, Spain.
†GPSC, Universidade de Vigo, Spain.

ABSTRACT

Distributed implementations of the Expectation-Maximization (EM) algorithm reported in literature have been proposed for applications to solve specific problems. In general, a primary requirement to derive a distributed solution is that the structure of the centralized version enables the computation involving global information in a distributed fashion. This paper treats the problem of distributed estimation of Gaussian densities by means of the EM algorithm in wireless sensor networks using diffusion strategies, where the information is gradually diffused across the network for the computation of the global functions. The low-complexity implementation presented here is based on a two time scale operation for information averaging and diffusion. The convergence to a fixed point of the centralized solution has been studied and the appealing results motivates our choice for this model. Numerical examples provided show that the performance of the distributed EM is, in practice, equal to that of the centralized scheme.

Index Terms— Distributed algorithms, expectation-maximization algorithms, Gaussian distribution, unsupervised learning, wireless sensor networks.

1. INTRODUCTION

Density estimation belongs to the general class of unsupervised learning problems and is widely studied in fields like data mining and machine learning, particularly in applications like pattern classification and pattern recognition for image or speech analysis, but also for clustering. In presence of hidden variables, the EM algorithm provides a means to iteratively compute the maximum likelihood (ML) estimator when the data follows an exponential distribution [1]. Starting from an initial set of estimates, the method alternates between an expectation (E) step, where the expected log-likelihood function of the observations is evaluated using the current estimates, and a maximization (M) step, where the log-likelihood function of the E-step is maximized with respect to the estimates. Whereas the E-step performs computations using local information only, the M-step performs computations using global information. A distributed implementation of the algorithm for a wireless sensor network (WSN) entails therefore a reformulation of the computations such that they can be performed locally at each node. Related contributions to the literature propose distributed EM implementations where the global sufficient statistics are computed using for instance aggregation [2–5], or using a consensus-based scheme [6–8]. Aggregation strategies require the assignment of routing paths or junction trees within the network, whereas consensus type strategies result in an increase in the total number of iterations until convergence, since a consensus algorithm [9, 10] is fully executed at each M-step. Based on diffusion strategies [11–13], a distributed EM is proposed in [14] where the authors use the results from [15] to show that their implementation is a Robbins-Monro stochastic approximation to the centralized EM approach. A similar scheme is proposed in [16] for tracking applications using particle filtering, and a diffusion adaptation scheme is proposed in [17] for learning in Gaussian mixture models (GMM).

In this paper we present a novel Diffusion-Based Distributed EM (DB-DEM) algorithm for density estimation and classification in WSNs. The method is based on a distributed algorithm derived for ML estimation in presence of unreliable observations [18]. Whereas the problem of unreliable observations in [18] could be seen as a particular case of a GMM, we address here the general case of density estimation under the assumption of GMM's. The main idea behind the DB-DEM algorithm is that the diffusion of the information across the network is embedded in the iterative update of the parameter estimates. Therefore, the M-step could be seen as a two time scale operation where a faster term for information diffusion is combined with a slower term for information averaging, and differing therefore from [14–17] which assume a single time scale operation. As opposed to most diffusion adaptive techniques which consider constant step-sizes, e.g. [16, 17], we consider here time-varying step-sizes

in order to converge to the desired values. The advantage of the DB-DEM with respect to the consensus-based EM algorithm in [6–8] is a significant reduction in the total number of iterations, since only one averaging operation is performed at each M-step. This reduction in the number of iterations can be translated into energy savings, a critical issue specially in large-scale deployments. Although not included due to lack of space, the convergence of the DB-DEM algorithm to a fixed point of the centralized EM solution has been studied for different choices of the control parameters. Numerical examples provided here show that the performance of the DB-DEM is, in practice, equal to that of the centralized scheme. The paper is organized as follows. In section 2 we describe the observation model and derive the expressions for the centralized EM algorithm. Section 3 presents the DB-DEM for density estimation in WSN's under the assumption of GMM's. Simulations results and conclusions are presented in sections 4 and 5 respectively.

2. PROBLEM FORMULATION

2.1. Observation Model

Consider a set of N independent observations indexed with $i = 1, \cdots, N$. A D-dimensional observation vector $\boldsymbol{y}_i$ is formed with variables drawn from a set of M Gaussian densities with probability distribution $\mathcal{N}(\boldsymbol{y}_i|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$, where $\boldsymbol{\mu}_m \in \mathbb{R}^{D \times 1}$ is the vector of means and $\boldsymbol{\Sigma}_m \in \mathbb{R}^{D \times D}$ is the covariance matrix $\forall m = \{1, \cdots, M\}$. Let $\boldsymbol{z} \in \mathbb{R}^{N \times 1}$ be an indicator vector with the ith entry $z_i = m$ if the ith observation belongs to the mth density. The probability density function (pdf) for $\boldsymbol{y}_i$ is given by

$$f(\boldsymbol{y}_i) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{y}_i|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \qquad (1)$$

where π_m is the prior probability of observing the mth Gaussian, i.e., $\pi_m = \Pr\{z_i = m\}$ for all i with $\sum_{m=1}^{M} \pi_m = 1$. We regard the vector $\boldsymbol{y} = [\boldsymbol{y}_1^T \cdots \boldsymbol{y}_N^T]^T$ as the *incomplete* observation and $\{\boldsymbol{y}, \boldsymbol{z}\}$ as the *complete* observation. Since $\boldsymbol{z}$ is unknown, we treat it as a random variable (r.v.). The set of unknown parameters to be estimated is therefore the tuple $\boldsymbol{\theta} = \{\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}\}$, where $\boldsymbol{\pi} = \{\pi_1, \cdots, \pi_M\}$, $\boldsymbol{\mu} = \{\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_M\}$, and $\boldsymbol{\Sigma} = \{\boldsymbol{\Sigma}_1, \cdots, \boldsymbol{\Sigma}_M\}$.

2.2. Centralized EM Algorithm

The ML estimator has the desirable properties of being unbiased and asymptotically efficient as the number of samples goes to infinity, and the EM algorithm is a numerical method to iteratively compute the ML estimates in the presence of incomplete observations. Under mild conditions, it is guaranteed to converge to a -local- maximum of the likelihood function [19]. Assuming a centralized approach in which the vector of observations $\boldsymbol{y}$ is available, at time t one performs

the following:

1. *E-step:* given an estimate $\hat{\boldsymbol{\theta}}^{t-1} = \{\hat{\boldsymbol{\pi}}^{t-1}, \hat{\boldsymbol{\mu}}^{t-1}, \hat{\boldsymbol{\Sigma}}^{t-1}\}$, where $\hat{\boldsymbol{\pi}}^t = \{\hat{\pi}_1^t, \cdots, \hat{\pi}_M^t\}, \hat{\boldsymbol{\mu}} = \{\hat{\boldsymbol{\mu}}_1^t, \cdots, \hat{\boldsymbol{\mu}}_M^t\}, \hat{\boldsymbol{\Sigma}} = \{\hat{\boldsymbol{\Sigma}}_1^t, \cdots, \hat{\boldsymbol{\Sigma}}_M^t\}$, compute the conditional expectation

$$Q(\tilde{\boldsymbol{\theta}}\,;\,\hat{\boldsymbol{\theta}}^{t-1}) = \mathbb{E}_{\boldsymbol{z}}\{\log f(\boldsymbol{y}, \boldsymbol{z}\,|\,\tilde{\boldsymbol{\theta}})\,|\,\hat{\boldsymbol{\theta}}^{t-1}, \boldsymbol{y}\} \qquad (2)$$

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

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

$$\hat{\boldsymbol{\theta}}^t = \arg\max_{\tilde{\boldsymbol{\theta}}} Q(\tilde{\boldsymbol{\theta}}\,;\,\hat{\boldsymbol{\theta}}^{t-1}).$$

Note that we can write $Q(\tilde{\boldsymbol{\theta}}\,;\,\hat{\boldsymbol{\theta}}^{t-1}) = \sum_{i=1}^{N} Q_i(\tilde{\boldsymbol{\theta}}\,;\,\hat{\boldsymbol{\theta}}^{t-1})$ due to independence of the pairs $\{z_i, \boldsymbol{y}_i, \forall i\}$, where

$$Q_i(\tilde{\boldsymbol{\theta}}\,;\,\hat{\boldsymbol{\theta}}^{t-1}) = \mathbb{E}_{z_i}\{\log f(\boldsymbol{y}_i, z_i\,|\,\tilde{\boldsymbol{\theta}})\,|\,\hat{\boldsymbol{\theta}}^{t-1}, \boldsymbol{y}_i\}$$

Therefore,

$$Q_i(\tilde{\boldsymbol{\theta}}\,;\,\hat{\boldsymbol{\theta}}^{t-1}) = \mathbb{E}_{z_i}\left\{\log\left(f(\boldsymbol{y}_i\,|\,z_i, \tilde{\boldsymbol{\theta}}) \cdot \Pr(z_i\,|\,\tilde{\boldsymbol{\theta}})\right)\,|\,\hat{\boldsymbol{\theta}}^{t-1}, \boldsymbol{y}_i\right\}$$
$$= \sum_{m=1}^{M} \hat{\zeta}_{m,i}^t \log\left[\tilde{\pi}_m \mathcal{N}(\boldsymbol{y}_i|\tilde{\boldsymbol{\mu}}_m, \tilde{\boldsymbol{\Sigma}}_m)\right] \qquad (3)$$

where

$$\hat{\zeta}_{m,i}^t = \Pr(z_i = m\,|\,\hat{\boldsymbol{\theta}}^{t-1}, \boldsymbol{y}_i)$$
$$= \frac{\hat{\pi}_m^{t-1}\mathcal{N}(\boldsymbol{y}_i|\hat{\boldsymbol{\mu}}_m^{t-1}, \hat{\boldsymbol{\Sigma}}_m^{t-1})}{\sum_{l=1}^{M} \hat{\pi}_l^{t-1}\mathcal{N}(\boldsymbol{y}_i|\hat{\boldsymbol{\mu}}_l^{t-1}, \hat{\boldsymbol{\Sigma}}_l^{t-1})} \qquad (4)$$

is the (a posteriori) probability of having observed the mth Gaussian density given $\boldsymbol{y}_i$ and the estimates from the previous iteration. Hence, the expression in (2) is equal to

$$\sum_{m=1}^{M} \sum_{i=1}^{N} \hat{\zeta}_{m,i}^t \log \tilde{\pi}_m + \sum_{m=1}^{M} \sum_{i=1}^{N} \hat{\zeta}_{m,i}^t \log \mathcal{N}(\boldsymbol{y}_i|\tilde{\boldsymbol{\mu}}_m, \tilde{\boldsymbol{\Sigma}}_m). \quad (5)$$

The estimator for the a priori probability for the mth density is obtained maximizing the expression above with respect to $\tilde{\pi}_m$ subject to the constraint $\sum_{m=1}^{M} \hat{\pi}_m^t = 1$, whereas the estimators for the mean and the covariance matrix are obtained maximizing it with respect to $\tilde{\boldsymbol{\mu}}_m$ and $\tilde{\boldsymbol{\Sigma}}_m$, i.e.,

$$\hat{\pi}_m^t = \frac{1}{N} \sum_{i=1}^{N} \hat{\zeta}_{m,i}^t \qquad (6a)$$

$$\hat{\boldsymbol{\mu}}_m^t = \frac{\sum_{i=1}^{N} \boldsymbol{y}_i \hat{\zeta}_{m,i}^t}{\sum_{i=1}^{N} \hat{\zeta}_{m,i}^t} \qquad (6b)$$

$$\hat{\boldsymbol{\Sigma}}_m^t = \frac{\sum_{i=1}^{N} (\boldsymbol{y}_i - \hat{\boldsymbol{\mu}}_m^t)(\boldsymbol{y}_i - \hat{\boldsymbol{\mu}}_m^t)^T \hat{\zeta}_{m,i}^t}{\sum_{i=1}^{N} \hat{\zeta}_{m,i}^t} \qquad (6c)$$

It is well known that the EM algorithm is guaranteed to converge to a local maximum of the likelihood function, but it is

sensitive to the initialization of the parameters. Therefore, in order to start up the iteration, a suitable initializer is needed. Notice that the a posteriori probabilities in (4) require knowledge of *local* information only, whereas the estimates in (6) require knowledge of global information. Therefore, a distributed implementation of the EM algorithm entails the computation of (6) in a distributed fashion.

3. DIFFUSION-BASED EM ALGORITHM

Based on the distributed implementation of the EM algorithm in [18], we propose a scheme where the summations among all observations in (6) are computed by the network via diffusion. Let us assume a WSN composed of N nodes where the communications for each node are restricted to a small neighborhood. The information flow among the nodes is described by means of an undirected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ is the set of vertices (nodes) and $\mathcal{E}$ is the set of edges (links) e_{ij} $\forall \{i, j\} \in \mathcal{V}$ with $e_{ij} = e_{ji}$ [20]. The set of neighbors of node i is denoted as $\mathcal{N}_i = \{j \in \mathcal{V} : e_{ij} \in \mathcal{E}\}$ for all $i = 1, \cdots, N$. We further assume that the network is connected such that there is a path between any pair of nodes $\{i, j\} \in \mathcal{V}$. Consider then a *weight matrix* $\boldsymbol{W} \in \mathbb{R}^{N \times N}$ with a nonzero $\{ij\}^{th}$ entry W_{ij} only if $j \in \mathcal{N}_i$ and $i \in \mathcal{N}_j$. $\boldsymbol{W}$ is assumed symmetric and satisfies

$$i) \; \mathbf{1}^T \boldsymbol{W} = \mathbf{1}^T, \; \boldsymbol{W}\mathbf{1} = \mathbf{1}, \; \rho(\boldsymbol{W} - \tfrac{\mathbf{1}\mathbf{1}^T}{N}) < 1,$$

where $\mathbf{1}$ is an all-ones vector and $\rho(\cdot)$ denotes spectral radius [9].

Further, for any node i and any time instant $k \geqslant 0$ define the following vector of length $L = M(1 + D + D^2) + 1$

$$\boldsymbol{f}(i, k) = [1, \hat{\zeta}_{1,i}^k, \cdots, \hat{\zeta}_{M,i}^k, \tag{7a}$$
$$\hat{\zeta}_{1,i}^k \boldsymbol{y}_i^T, \cdots, \hat{\zeta}_{M,i}^k \boldsymbol{y}_i^T, \tag{7b}$$
$$\hat{\zeta}_{1,i}^k \text{vec}(\boldsymbol{y}_i \boldsymbol{y}_i^T), \cdots, \hat{\zeta}_{M,i}^k \text{vec}(\boldsymbol{y}_i \boldsymbol{y}_i^T)]^T \tag{7c}$$

where vec$(\cdot)$ denotes the operation of vectorizing a matrix. Note that the entries from 2 to $M + 1$ of $\boldsymbol{f}(i, k)$ are the estimates at time k of the a posteriori probabilities, the following $D \times M$ entries correspond to the product of the a posteriori probabilities and the observation vector, whereas the last $D^2 \times M$ entries correspond to the product of the a posteriori probabilities and the entries of $\boldsymbol{y}_i \boldsymbol{y}_i^T$. The distributed EM estimator based on the DB-DEM algorithm in [18] is described below:

_____

At each node i, for $m = 1, \cdots, M$

   1. Initialize:
For $k \geqslant 1$ $\hat{\pi}_m^0, \quad \hat{\boldsymbol{\mu}}_m^0, \quad \hat{\boldsymbol{\Sigma}}_m^0.$

   2. *E-Step*: given $\hat{\pi}_m^{k-1}, \hat{\boldsymbol{\mu}}_m^{k-1}, \hat{\boldsymbol{\Sigma}}_m^{k-1}$, compute the a posteriori probabilities $\hat{\zeta}_{m,i}^k$ according to (4) and define $\boldsymbol{f}(i, k)$ according to (7).

   3. *M-Step*: For $l = 1, \cdots, L$ compute:

$$\phi_l(i, k) = (1 - \beta_k) \sum_{j=1}^{N} W_{ij} \phi_l(j, k - 1)$$
$$+ \alpha_k \sum_{j=1}^{N} W_{ij} \boldsymbol{f}_l(j, k) \tag{8}$$

and update the parameters as follows:

$$\hat{\pi}_m^k = \frac{\phi_{m+1}(i, k)}{\phi_1(i, k)}, \quad \hat{\boldsymbol{\mu}}_m^k = \left(\frac{\phi_{\Delta_\mu}(i, k)}{\phi_{m+1}(i, k)} \right)^T$$
$$\hat{\boldsymbol{\Sigma}}_m^k = \text{vec}_{D \times D}^{-1} \left(\frac{\phi_{\Delta_\Sigma}(i, k)}{\phi_{m+1}(i, k)} \right) - \hat{\boldsymbol{\mu}}_m^k [\hat{\boldsymbol{\mu}}_m^k]^T$$

where $\Delta_\mu = [M + (m-1)D + 2 : M + mD + 1]$, $\Delta_\Sigma = [M(1+D) + D^2(m-1) + 2 : M(1+D) + D^2 m + 1]$, and

$$\alpha_k = \frac{1}{k}, \quad \beta_k = \frac{1}{k^\delta}, \quad 0 < \delta < 1. \tag{9}$$

4. Repeat steps 2 and 3 until convergence.
_____

Table I: DB-DEM Estimator

Remark that the computation of $\phi_l(i, k)$ in (8) at each node i, at time k and for all $l = 1, \cdots, L$, entails communication with neighboring nodes only, since $\phi_l(j, k)$ is nonzero for $j \in \mathcal{N}_i$ only. The step-size parameter α_k in (8) controls the rate of information diffusion across the network, whereas the parameter β_k controls the rate of information averaging. Note that Δ_μ and Δ_Σ define the intervals of subindices to choose the entries used for the computations. Finally, note also that for $k = 1$, the first term on the right-hand side of (8) is zero, so the initial condition $\phi_l(i, 0)$ is irrelevant. Based on convergence results not included here due to lack of space, we claim that the DB-DEM converges to a fixed point of the centralized EM solution for a proper initialization of the parameters, provided that assumption $i)$ holds and assuming the step-sizes α_k and β_k in (9). In fact, the convergence to a fixed point of the centralized solution has been studied for different values of the control parameters α_k and β_k, showing that an attractive feature of the DB-DEM is its robustness with respect to this choice, as it has been verified by the numerical results.

4. SIMULATION RESULTS

The analytical results are supported with computer simulations of a WSN composed of $N = 50$ nodes randomly deployed on a unit square with connectivity radius $r_c = 0.2$. The step-size parameter $\beta_k = 1/k^\delta$ is chosen with $\delta = 0.85$, and we assume mixture model of $M = 2$ Gaussians. For simplicity, we consider $D = 1$, i.e., the 1-dimensional case with $\mu_1 = 2$, $\mu_2 = 4$, $\sigma_1 = 0.4$, $\sigma_2 = 0.2$, $\pi_1 = 0.3$, $\pi_1 = 0.7$ where
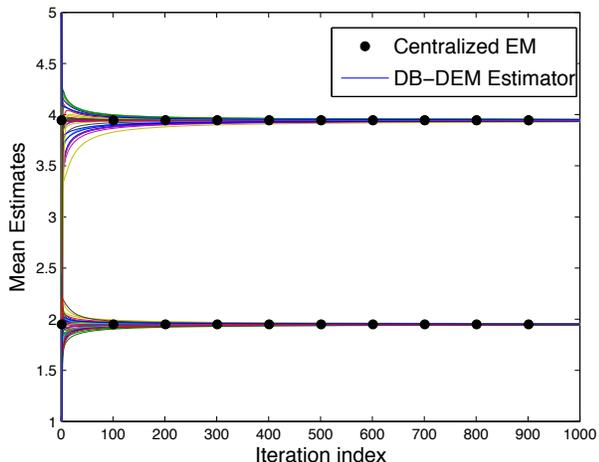
Fig. 1. Evolution of the mean estimates as a function of k for all nodes. The centralized solution is included as a reference.

Fig. 2. Evolution of $\hat{\zeta}_{1,i}$ and $\hat{\zeta}_{2,i}$ vs. iteration index.

the nodes initialize the estimates equally with $\mu_1^0 = 0$, $\mu_2^0 = 5$, $\sigma_1^0 = 0.5$, $\sigma_2^0 = 0.3$ and $\pi_1^0 = \pi_2^0 = 0.5$. The DB-DEM algorithm is run assuming a Metropolis weight matrix with entries defined as

$$W_{ij} = \begin{cases} 1/(1 + \max\{|\mathcal{N}_i|, |\mathcal{N}_j|\}) & j \in \mathcal{N}_i \\ 1 - \sum_{k \in \mathcal{N}_i} W_{ik} & i = j \\ 0 & \text{otherwise} \end{cases}$$

where $|\cdot|$ denotes cardinality [21]. In summary, the weights assigned to the data arriving from neighboring nodes are decided locally. Fig. 1 depicts the evolution of the mean estimates for each node as a function of the iteration index k for one random realization, whereas Fig. 2 depicts the evolution of $\hat{\zeta}_{1,i}$ and $\hat{\zeta}_{2,i}$. Clearly, the nodes are able to estimate the parameters for each density. In particular, for classification applications where the nodes make a decision based on the a posteriori probabilities, we observe from the simulations that a correct classification can be made after approximately 50 iterations. Finally, Fig. 3 depicts the log-likelihood function $Q(\hat{\boldsymbol{\theta}}^{t-1}; \hat{\boldsymbol{\theta}}^{t-1})$ computed as in (5) for the DB-DEM estimator (solid line) along with the curves for the centralized EM solution (dashed-dotted line) and a consensus-based EM (dashed line) [6–8]. In order to make a fair comparison, and since the curves for the DB-DEM and for the centralized EM have converged after approximately 100 and 10 iterations respectively, we run a consensus-based EM with 10 averaging iterations at each M-step. Clearly, the truncation of the averaging computation after only 10 iterations is not enough to estimate the parameters correctly. In summary, a higher number of averaging iterations per M-step is required by the consensus-based scheme to obtain better results.
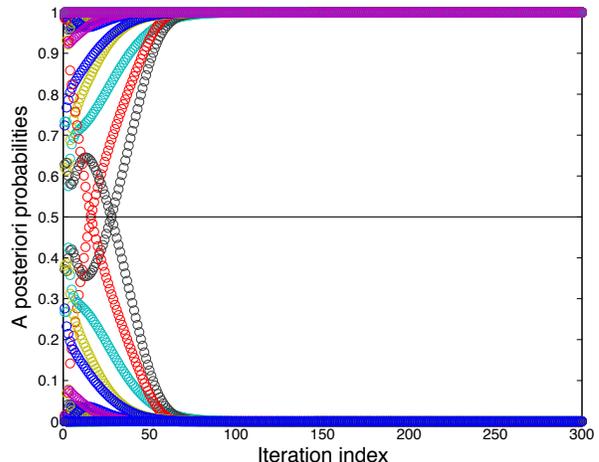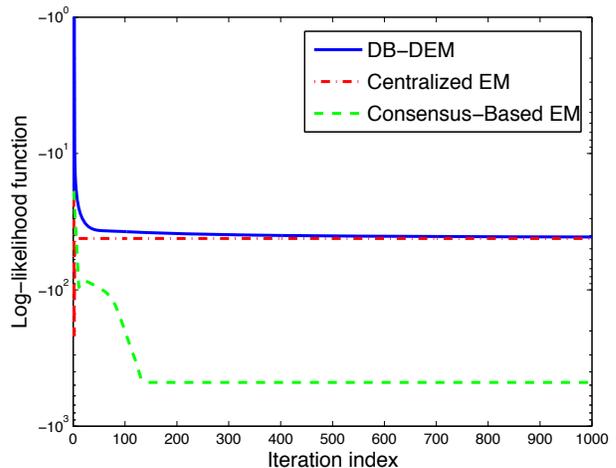

Fig. 3. Log-likelihood function vs. iteration index for the DB-DEM, the centralized EM and a consensus-based EM.

5. CONCLUSIONS

The problem of distributed density estimation and classification under the assumption of Gaussian mixtures models has been addressed in this paper. A distributed expectation-maximization estimator has been derived for wireless sensor networks using diffusion strategies, where the information diffusion across the network is embedded in the iterative update of parameter estimates. The advantage with respect to prior art relies in the simplicity of the iterative algorithm including the two time scale M-step, a reduction in the number of iterations with respect to consensus-based schemes, the robustness with respect to the choice of step-sizes, and the existing convergence analysis supporting the model. Numerical examples provided show that the performance of the distributed EM is, in practice, equal to that of the centralized EM solution.

6. REFERENCES

[1] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.

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

[3] J. Wolfe, A. Haghighi, and D. Klein, "Fully distributed EM for very large datasets," in *Proc. of the 25th ICML*, New York, NY, USA. ACM, pp. 1184–1191, 2008.

[4] K. Bhaduri and A. N. Srivastava, "A local scalable distributed expectation maximization algorithm for large peer-to-peer networks," in *Proc. of the 9th IEEE ICDM*, Washington, DC, USA. IEEE Computer Society, pp. 31–40, 2009.

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

[6] W. Kowalczyk and N. Vlassis, "Newscast EM," in *In NIPS 17*. MIT Press, pp. 713–720, 2005.

[7] 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 ICASSP 2008*, pp. 1989–1992.

[8] S. S. Pereira, S. Barbarossa, and A. Pages-Zamora, "Consensus for distributed EM-based clustering in WSNs," in *Proc. of the 6th IEEE SAM*, pp. 45–48, 2010.

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

[10] R. Olfati-Saber, and R.M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[11] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion Recursive Least-Squares for Distributed Estimation Over Adaptive Networks," *IEEE Trans. on Signal Processing*, vol. 56, no. 5, pp. 1865–1877, 2008.

[12] C. G. Lopes and A. H. Sayed, "Diffusion Least-Mean Squares Over Adaptive Networks: Formulation and Performance Analysis" *IEEE Trans. on Signal Processing*, vol. 56, no. 7, pp. 3122–3136, 2008.

[13] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS Strategies for Distributed Estimation," *IEEE Trans. on Signal Processing*, vol. 58, no. 3, pp. 1035–1048, 2010.

[14] Y. Weng, L. Xie, and W. Xiao, "Diffusion scheme of distributed EM algorithm for Gaussian mixtures over random networks," in *Proc. of the IEEE ICCA*, pp. 1529–1534, 2009.

[15] 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.

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

[17] Z. J. Towfic, J. Chen and A. Sayed, "Collaborative learning of mixture models using diffusion adaptation," *IEEE MLSP*, pp. 1–6, Sep. 2011.

[18] S. Silva Pereira, R. López-Valcarce, A. Pagès-Zamora, "A Diffusion-Based EM Algorithm for Distributed Estimation in Unreliable Sensor Networks", *Submitted to IEEE SPLetters*.

[19] C. F. Jeff Wu, "On the Convergence Properties of the EM Algorithm," in *Annals of Statistics*, vol. 11, no. 1, pp. 95–103, 1983.

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

[21] L. Xiao, S. Boyd, and S. Lall, "Distributed Average Consensus with Time-Varying Metropolis Weights", 2006. Online: http://www.stanford.edu/ boyd/papers/avg metropolis.html