Simultaneous distributed estimation and classification in sensor networks ⋆

A. Chiuso * **F. Fagnani** ** **L. Schenato** *** **S. Zampieri** ***

* *Dipartimento di Tecnica e Gestione dei Sistemi Industriali, University of Padova, Vicenza, Italy (*chiuso@dei.unipd.it*).*
** *Dipartimento di Matematica, Politecnico di Torino, Torino, Italy (*fabio.fagnani@polito.it*).*
*** *Dipartimento di Ingegneria dell'Informazione, University of Padova, Padova, Italy (*{schenato,zampi}@dei.unipd.it*).*

Abstract: In this work we consider the problem of simultaneously classifying sensor types and estimating hidden parameters in a network of sensors subject to gossip-like communication limitations. In particular, we consider a network of scalar noisy sensors which measure a common unknown parameter. We assume that a fraction of the nodes is subject to the same (but possibly unknown) offset. The goal for each node is to simultaneously identify the class the node belongs to and to estimate the common unknown parameter, only through local communication and computation. We propose a distributed estimator based on the maximum likelihood (ML) approach and we show that, in case the offset is known, this estimator converges to the centralized ML as the number N of sensor nodes goes to infinity. We also compare this strategy with a distributed implementation of estimation-maximization (EM) algorithm and a distributed naive strategy; we show tradeoffs via numerical simulations in terms of robustness, speed of convergence and implementation simplicity.

1. INTRODUCTION

In recent years, we have witnessed an increasing interest in the design of control, estimation algorithms which can operate in a distributed manner over a network of locally communicating units. A prototype of such problems is the average consensus algorithm Olfati-Saber and Murray (2004); Olfati-Saber et al. (2007), which can be used as a distributed procedure providing the average of real numbers, each of them belonging to a unit. Since the average is the building block for many estimation methods, the average consensus has been proposed as a possible way to obtain distributed estimation algorithms and, in particular, to obtain distributed Kalman filtering Olfati-Saber (2005); Carli et al. (2008). However, while averaging is suitable for the estimation of real valued parameters, it is typically of no help when the quantities to be estimated belong to a finite alphabet. Moreover, the average is by definition an operation which fuses information loosing in this way the possible information that is specific of each unit. The model we consider in the present paper has two characteristics, namely we consider the case in which the information of each unit contains both a common parameter and a unit specific parameter. Moreover we assume the unit specific parameter belong to a finite alphabet.

More precisely we assume that we have N units and that each unit i has a number y_i that can be decomposed as follows

$$y_i = \theta + T_i + v_i\,. \tag{1}$$

where $\theta \in \mathbb{R}$ is a continuous parameter influencing all the units, $T_i \in \mathscr{A}$, with $\mathscr{A}$ being a finite set, is a discrete parameter influencing each unit independently and v_i is a noise term. The goal of each unit is to estimate the common parameter θ and its specific one T_i. Notice that the presence of the common parameter θ impose that any efficient estimation technique will require cooperation between units and therefore will require communication. We will assume that communication between the units can occur only according to a graph as discussed in Section 3, which is devoted to the distributed algorithm description.

There are various examples of applications in which the previous estimation problem could be of interest. One application is related to fault detection. Indeed, the units could represent in this case some sensors that, when working properly, measure a noisy version the parameter θ and that, when faulty, add a bias to the measurement. Similar application could consist of heterogeneous sensors belonging to classes which differ by the bias they add. In both cases the parameter of primary interest is θ. Another example could consists in different units belonging to different classes, the objective being to classify them based on the y_i's while also estimating the common parameter θ.

For example we can imagine a network for environmental monitoring; the different values of T_i could model for instance a constant external field only active in certain areas where the sensor is located, such as for instance being on the sunshine or on the shade or being inside or outside of a fire.

More in general these problems fit in the general class of the unsupervised clustering problems, which are quite standard in statistics Titterington et al. (1985); Duda et al. (2001). Algorithms for clustering have been widely proposed in the computer science literature both for the standard centralized case Berkhin (2006) and for the distributed case Rabbat and Nowak (2004); Nowak (2003); Safarinejadian et al. (2010); Bandyopadhyay et al. (2006). Indeed, the technique proposed in this paper can be seen as a distributed algorithm for a specific clustering problem.

⋆ This research has been partially supported by EU FP7-ICT-223866-FeedNetBack project, by CaRiPaRo Foundation "WISE-WAI" project and by the Progetto di Ateneo CPDA090135/09 funded by the University of Padova.

The structure of the paper is as follows: Section 2 introduces the model we consider; the decentralized estimator is studied in Section 3 while its limit behavior is characterized in Section 4. In section 5 an alternative approach based on a Bayesian model is presented and the some generalization are discussed in Section 6. Some simulations are presented in Section 7. Some of the proofs are omitted for reasons of space and can be found in Chiuso et al. (2010).

2. THE MODEL

In this section we give a more precise description of the model we consider and of the estimation cost we will try to minimize by the proposed estimation algorithm. Assume that the numbers y_i are given by (1), where we assume that $\theta \in \mathbb{R}$, $T_i \in \{0,1\}$ and that v_i are independent Gaussian random variables. The goal of each unit i is to estimate θ and T_i. For simplicity, with respect to what mentioned in the introduction, we will restrict to the case in which T_i can take only two values, that are supposed to be known and which, with no loss of generality, can be supposed to be 0 and 1. Extension to the case in which the difference between the two symbols is unknown are discussed in Section 6. The algorithm we propose does not need to know the variance σ which therefore can be assumed unknown.

2.1 The maximum likelihood estimator

When the bias term T_i is not present, the centralized maximum likelihood estimator of θ (assuming that all measurements y_i are available) is given by

$$\hat{\theta} = N^{-1} \sum_i y_i \,. \tag{2}$$

This arithmetic average can be asymptotically evaluated by the agents in the graph through standard consensus algorithms as long as the graph is strongly connected.

The presence of the bias terms makes the problem quite harder. In this paper we propose a decentralized version of the centralized maximum likelihood estimator for this problem. We set some useful notation. We consider the vectors $y = (y_1, \dots y_N)$ and $T = (T_1, T_2, \dots T_N)$ and the following weights $w(T) = \sum T_i$, $w(y) = \sum y_i$. The maximum likelihood estimator is given by

$$(\hat{\theta}^{ML}, \hat{T}^{ML}) \;=\; \underset{(\theta,T)}{\operatorname{argmax}} P(y|\theta,T) = \underset{(\theta,T)}{\operatorname{argmin}} \left[\sum_i \frac{(y_i - \theta - T_i)^2}{2\sigma^2} \right] \tag{3}$$

Remark 1. The choice of the maximum likelihood estimator is motivated by the simplicity of the solution we obtain from it. Of course, it would be natural to seek for "optimal" estimators which minimize, e.g., the variance of $\hat{\theta}$, $\mathbb{E}[(\hat{\theta} - \theta)^2]$ and/or the average classification error $\mathbb{E}[\sum_{i=1}^N |\hat{T}_i - T_i|]$. Unfortunately these optimal estimators are in general difficult (if not impossible) to find even in the centralized case. We will show instead that the maximum likelihood estimator is not only computationally simple, but also prone to a decentralized implementation.

From (3) we immediately obtain that

$$\hat{\theta}(T) = \frac{1}{N} \sum_i (y_i - T_i) = \frac{w(y) - w(T)}{N} \tag{4}$$

So, to estimate θ, what is needed is the average measure $N^{-1} w(y)$ which can be obtained by a standard consensus algorithm, and the average bias $N^{-1} w(T)$. This second term however is not directly available, so that (4) is not an implementable solution. Rather, we can substitute (4) inside (3) and we obtain:

$$\hat{T} = \underset{T}{\operatorname{argmin}} \left[\sum_i \frac{\left(y_i - \frac{w(y)}{N} + \frac{w(T)}{N} - T_i \right)^2}{2\sigma^2} \right] \tag{5}$$

This minimization can be solved in a two-step way by considering

$$\min_{w=0,\dots,N} \left[\min_{T\,:\,w(T)=w} \sum_i \frac{\left(y_i - \frac{w(y)}{N} + \frac{w}{N} - T_i \right)^2}{2\sigma^2} \right] \tag{6}$$

For every $w = 0, \dots, N$, put

$$\hat{T}_w = \underset{T:w(T)=w}{\operatorname{argmin}} \sum_i \frac{\left(y_i - \frac{w(y)}{N} + \frac{w}{N} - T_i \right)^2}{2\sigma^2} \tag{7}$$

Let us define

$$\eta_i = y_i - \frac{w(y)}{N}$$

and consider its ordered permutation $\eta_{[1]} \le \eta_{[2]} \le \dots \le \eta_{[N]}$. Clearly, the above minimization is solved by the vector $\hat{T}_w$ such that

$$(\hat{T}_w)_{[j]} = \begin{cases} 0 \text{ if } j \le N - w \\ 1 \text{ otherwise} \end{cases} \tag{8}$$

Substituting in (6) and performing simple algebraic transformations, we obtain that the solution of the outer minimization problem becomes $\hat{w} = \min F(w)$ where

$$F(w) := -\frac{w^2}{N} + w - 2 \sum_{j=N-w+1}^{N} \eta_{[j]}$$

Clearly, from Eqn. (8),

$$\hat{T}^{ML}_{[j]} = (\hat{T}_{\hat{w}})_{[j]} = \begin{cases} 0 \text{ if } j \le N - \hat{w} \\ 1 \text{ otherwise} \end{cases} \tag{9}$$

and from Eqn. (4) we get:

$$\hat{\theta}^{ML} = \frac{w(y) - \hat{w}}{N} = \frac{w(y) - w(\hat{T}^{AML})}{N} \tag{10}$$

3. A DECENTRALIZED ESTIMATOR

Notice that each agent i can compute η_i by a consensus algorithm. Moreover, as will be discussed later, there exists an efficient decentralized algorithm capable of ordering the η_i, so that each agent i knows its ordering index j_i: $\eta_i = \eta_{[j_i]}$. For each value w, the agent i is thus capable of computing $(\hat{T}_w)_i$ through (8). In order to compute $(\hat{T}_w)_i$ using (9) we need to know the ordered position j_i of agent i with respect to $N - \hat{w}$. This would follow if we could compute $\hat{w}$ in a decentralized fashion, but this is not at all evident, because of the presence of the aggregation term $\sum_{j=N-w+1}^{N} \eta_{[j]}$.

Consider

$$\Delta(w) := F(w+1) - F(w) = -\frac{2w+1}{N} + 1 - 2\eta_{[N-w]} \tag{11}$$

Notice that $\Delta(w)$ can be computed by the agent in ordered position $N - w$.

Define the set of local minima:

$$\mathscr{S} := \{ w \in [1, N-1] \mid \Delta(w-1) < 0 \,,\, \Delta(w) > 0 \}$$

If we knew that $|\mathscr{S}| = 1$ then our computational problem could be solved in the following way. Notice that in this case

we would have that $G(w)$ decreases till the point $\hat{w}$ and then starts to increase. Consider a generic agent i in position j_i. He computes $\Delta(N-j_i)$. If $\Delta(N-j_i)<0$ it means that $N-j_i<\hat{w}$, namely $j_i>N-\hat{w}$ which implies, by (9) that $(\hat{T}_{\hat{w}})_{[j_i]}=1$. If instead $\Delta(N-j_i)>0$, then $(\hat{T}_{\hat{w}})_{[j_i]}=0$. So, in this way, each agent could compute its ML estimated bias $\hat{T}_i$. Again, using consensus all agents can then compute $N^{-1}\hat{w}=N^{-1}w(\hat{T})$ and can therefore also compute θ using formula (4).

Of course the decentralized algorithm proposed above can always be implemented by the agents. In the following part of the paper we will show that, typically, for N large, F possesses just one local minimum in $[0,1/2]$ which happens to be the global minimum on $[0,1]$ while it can show other local minima on $]1/2,1]$. In this way, applying previous algorithm but for all agents whose position j is above $N/2$ and forcing all agents whose position j is below $N/2$ to estimate $\hat{T}_{[j]}=0$, with high probability we will obtain the maximum likelihood estimator. We can summarize the previous reasoning in the following conditions:

$$\hat{T}_i^{AML}=\begin{cases} 1 \text{ if } 2\left(y_i-\dfrac{w(y)}{N}\right)>1-\dfrac{2(N-j_i)+1}{N} \ \wedge \ j_i>\dfrac{N}{2} \\ \\ 0 \text{ otherwise} \end{cases} \tag{12}$$

where the superscript *AML* stands for *approximate maximum likelihood*. This approximate maximum likelihood estimator converges (as $N\to\infty$) to the maximum likelihood estimator in (3) as stated in corollary 12.

Before describing the algorithm to compute $(\hat{\theta}^{AML},\hat{T}^{AML})$ in a distributed fashion, we need to introduce some useful general distributed algorithms that will be used in our algorithm.

3.1 Decentralized average and ranking computation

We model the network of distributed agents with a graph $\mathscr{G}=(\mathscr{N},\mathscr{E})$ where $\mathscr{N}=\{1,2,\dots,N\}$ is the set of nodes and $\mathscr{E}$ is the set of edges corresponding to the communication links. We indicate with $V(i)$, the set of neighbors of node i, i.e. $V(i)=\{j \mid (i,j)\in\mathscr{E}\}$. We assume that the graph is connected, i.e. there is a path between any two nodes, and it is undirected, i.e. nodes are capable of bidirectional communications. We also assume that each sensor node i knows its label i, i.e. nodes are numbered from 1 to N.

Proposition 2. (Symmetric gossip consensus). Let us assume that each node i has a sensor measurement $y_i\in\mathbb{R}$, and initialize a local variable to $x_i(0)=y_i$. At each time step $k=1,2,\dots$, one edge $(i,j)\in\mathscr{E}$ is selected with probability $p_{ij}>0$ such that $\sum_{(i,j)\in\mathscr{E}}p_{ij}=1$. The nodes i and j exchange their local variables $x_i^{(k)}$ and $x_j^{(k)}$ and updates them as follows

$$x_i^{(k+1)}=\frac{x_i^{(k)}+x_j^{(k)}}{2}$$
$$x_j^{(k+1)}=\frac{x_i^{(k)}+x_j^{(k)}}{2}$$

while all other nodes do no perform any operation. Then we have

$$\lim_{k\to\infty}x_i^{(k)}=\frac{1}{N}\sum_{i=1}^{N}y_i \quad a.s.$$

Proposition 3. (Distributed ranking for complete graphs). Let us assume that each node i knows its own label i and has a sensor measurement $y_i\in\mathbb{R}$. Let us define $y_{[j]}$ the sorted measurements in increasing order, i.e. $y_{[1]}\le y_{[2]}\le\cdots\le y_{[N]}$ and let us indicate with j_i, the index in the ordered measurements of sensor i, i.e.

$$y_i=y_{[j_i]}$$

Let us consider the following algorithm: each sensor sets a local variable to its label, i.e.

$$x_i^{(0)}=i, \ \forall i$$

Then, at each time step $k=1,2,\dots$, one edge $(i,j)\in\mathscr{E}$ is selected with probability $p_{ij}>0$ such that $\sum_{(i,j)\in\mathscr{E}}p_{ij}=1$. The nodes i and j exchange their measurements y_i,y_j and their current indexes $x_i^{(k)},x_j^{(k)}$, and updates them as follows

$$x_i^{(k+1)}=\begin{cases} x_i^{(k)} \ if \ (y_i-y_j)(x_i^{(k)}-x_j^{(k)})\ge 0 \\ \\ x_j^{(k)} \ otherwise \end{cases}$$
$$x_j^{(k+1)}=\begin{cases} x_j^{(k)} \ if \ (y_i-y_j)(x_i^{(k)}-x_j^{(k)})\ge 0 \\ \\ x_i^{(k)} \ otherwise \end{cases}$$

while all other nodes do no perform any operation.

If the graph is *complete*, i.e. $(i,j)\in\mathscr{E},\forall i$, then there exists $T>0$ such that

$$x_i^{(k)}=j_i \ \ \forall k\ge T,\forall i \quad a.s.$$

Proof The proposed algorithm can be interpreted as a Markov chain defined on the indexes of the nodes and this chain has a unique absorbing state defined by the sorted list. Let us first define $\ell_{[j]}$ the node ℓ such that

$$y_{\ell_{[j]}}=y_{[j]}$$

i.e. $\ell_{[j]}$ is the label of the node that it is in position j in the sorted measurement list. We start by observing that if there exists T such that $x_{\ell_{[j]}}(T)=j,\forall j$, then also $x_{\ell_{[j]}}(T+1)=j,\forall j$, therefore $x_{\ell_{[j]}}=j,\forall j$ is an absorbing state.

Let us know compute the probability that after time T the list is ordered, i.e. $\mathbb{P}[x_{\ell_{[j]}}(T)=j, \ \forall j]$. To do so we compute the probability of a specific sequence that leads to the absorbing state. Let us consider the node $\overline{j}_i^{(k)}$ defined as

$$x_{\overline{j}_i^{(k)}}^{(k)}=i$$

i.e. the node j for which $x_j^{(k)}$ is equal to i at time k. Let us now consider the following sequence of edges

$$e_k=(\overline{j}_{N-k}^{(k)},\ell_{[N-k]}), \quad k=0,\dots,N-1$$

and consider the update of $x_i^{(k)}$ as specified in the algorithm. Then this sequence is designed so that $x_{\ell_{[N-k]}}^{(k+1)}=N-k$, i.e. the index $N-k$ is set in the right position. Since the ranking is done starting from the largest, it also follows that $x_{\ell_{[N-k]}}(t)=N-k$ for $t=k+1,\dots,N$, and therefore $x_{\ell_{[j]}}^{(k)}=j, \ \forall j$ for all $k\ge N$. Since this is only one specific sequence that leads to the absorbing state, it follows that

$$\mathbb{P}[x_{\ell_{[j]}}(T)=j, \ \forall j,T=N]\ge\mathbb{P}[e_0,e_1,\dots e_{N-1}]=\prod_{k=0}^{N-1}p_{e_k}\ge\varepsilon^N$$

where $\varepsilon = \min_{i,j} p_{ij} > 0$ since $e_k \in \mathscr{E}$ being the graph complete, and the events e_k are all independent by hypothesis. From the independence of the events e_k also follows that

$$\mathbb{P}[\exists (j,t)\ s.t.\ x^{(t)}_{\ell_{[j]}} \neq j, t \geq T = kN] \leq (1-\varepsilon^N)^k \overset{k\to\infty}{\longrightarrow} 0$$

which concludes the proof.

3.2 Decentralized estimation and classification algorithm

We are now ready to present the algorithm that allow each sensor i to compute the maximum likelihood estimate for the unknown parameter θ and for its unknown class T_i.

Proposition 4. Let us consider the following algorithm based on the measurements y_i available to each node i. We defined and initialize the following local variables:

$$\xi_i^{(0)} = \eta_i^{(0)} = \hat{\theta}_i^{(0)} = y_i,\ \ w_i^{(0)} = 0,\ \ \hat{T}_i^{(0)} = 0,\ \ \ell_i^{(0)} = i$$

At each time step $k = 1,2,\ldots$, one edge $(i,j) \in \mathscr{E}$ is selected with probability $p_{ij} > 0$ such that $\sum_{(i,j)\in\mathscr{E}} p_{ij} = 1$. The nodes i and j exchange their local variables $y, x^{(k)}, \ell^{(k)}, w^{(k)}$, and perform the following update for node i:

$$\begin{aligned}
\xi_i^{(k)} &= \frac{\xi_i^{(k-1)} + \xi_j^{(k-1)}}{2} \\
\eta_i^{(k)} &= y_i - \xi_i^{(k-1)} \\
\ell_i^{(k)} &= \begin{cases} \ell_i^{(k-1)} \ \text{ if } (y_i - y_j)(\ell_i(k-1) - \ell_j(k-1)) \geq 0 \\ \ell_j^{(k-1)} \ \text{ otherwise} \end{cases} \\
\hat{T}_i^{(k)} &= \begin{cases} 1 \ \text{ if } 2\eta_i^{(k)} > 1 - \dfrac{2(N-\ell_i^{(k)})+1}{N} \wedge \ell_i^{(k)} > \dfrac{N}{2} \\ 0 \ \text{ otherwise} \end{cases} \\
w_i^{(k)} &= \frac{w_i^{(k-1)} + w_j^{(k-1)}}{2} + \left(\hat{T}_i^{(k)} - \hat{T}_i^{(k-1)}\right) \\
\hat{\theta}_i^{(k)} &= \xi_i^{(k-1)} - w_i^{(k-1)}
\end{aligned}$$

and likewise for node j by simply replacing the index j with i and i with j in the previous equations. All other nodes do not perform any update.

If the graph is *complete*, then *almost surely* we have

$$\lim_{k\to\infty} \xi_i^{(k)} = \bar{y} = \frac{w(y)}{N} \tag{13}$$

$$\lim_{k\to\infty} \eta_i^{(k)} = y_i - \bar{y} \tag{14}$$

$$\lim_{k\to\infty} \ell_i^{(k)} = j_i \tag{15}$$

$$\lim_{k\to\infty} \hat{T}_i^{(k)} = \hat{T}_i^{AML} \tag{16}$$

$$\lim_{k\to\infty} w_i^{(k)} = \frac{\hat{w}}{N} \tag{17}$$

$$\lim_{k\to\infty} \hat{\theta}_i^{(k)} = \bar{y} - \frac{\hat{w}}{N} = \hat{\theta}^{AML} \tag{18}$$

Proof Eqns. (13) and (13) follow directly from Proposition 2, and Eqns. (15) from Proposition 3.

Let us now assume that all measurements are different, i.e. $y_{[1]} < y_{[2]} < \ldots < y_{[N]}$ and define

$$\delta = \min_{k=1,\ldots,N} \left| 2(y_i - \bar{y}) - 1 + \frac{2(N-j_i)+1}{N} \right|$$

From Proposition 2 it also follows that there exists T_1 such that $|\eta_i^{(k)} - (y_i - \bar{y})| < \delta$ for all $k \geq T_1$ and forall i almost surely. This fact and Proposition 3 imply that there exists T such that

$$2\eta_i^{(k)} - 1 + \frac{2(N-\ell_i^{(k)})+1}{N} > 0 \ \wedge\ \ell_i^{(k)} > \frac{N}{2},\ \ k \geq T \ \overset{a.s.}{\Longleftrightarrow}$$

$$\overset{a.s.}{\Longleftrightarrow}\ 2(y_i - \bar{y}) - 1 + \frac{2(N-j_i)+1}{N} > 0 \wedge j_i > \frac{N}{2}$$

Therefore, according to Eqn. (12), this implies that Eqn. (16) holds almost surely for all $k \geq T$.

Note now that

$$\begin{aligned}
\sum_{i=1}^N w_i^{(k)} &= \sum_{i=1}^N w_i^{(k-1)} + \sum_{i=1}^N \left(\hat{T}_i^{(k)} - \hat{T}_i^{(k-1)}\right) \\
&= \sum_{i=1}^N w_i^{(0)} + \sum_{i=1}^N (\hat{T}_i^{(k)} - \hat{T}_i^{(0)}) \\
&= \sum_{i=1}^N \hat{T}_i^{(k)} = \sum_{i=1}^N \hat{T}_i^{AML} = \hat{w}, \quad k \geq T
\end{aligned}$$

where we used the fact that $w_i(0) = \hat{T}_i(0) = 0, \forall i$ and the last equality follows from Equation (16) almost surely for some T. Then Eqns. (17) and (18) follows from Proposition 2.

4. THE LIMIT BEHAVIOR

In what follows we study the behavior (in particular the monotonicity) of the objective random function F when $N \to +\infty$. To emphasize dependence on N, from now on we will use the notation F_N.

We recall that, in our approach, the bias values T_i are fixed, even if unknown to the agents. We put

$$I^1 = \{i = 1,\ldots,N \,|\, T_i = 1\}, \quad I^0 = \{i = 1,\ldots,N \,|\, T_i = 0\}$$

and we assume that

$$\lim_{N\to+\infty} \frac{|I^1|}{N} = \lim_{N\to+\infty} \frac{w(T)}{N} = p \in [0, 1/2[\tag{19}$$

We start with some preliminary considerations on the ordered variables $\eta_{[w]}$. We can write $\eta_i = \xi_i + \Omega$ where

$$\xi_i = T_i + v_i, \quad \text{and } \Omega = \frac{w(v)}{N} - \frac{w(T)}{N}. \tag{20}$$

The variables ξ_i are thus independent and have two possible distribution functions:

$$\begin{aligned}
\mathbb{P}(\xi_i < t) &= F_\sigma(t-1) \ \text{ if } i \in I^1 \\
\mathbb{P}(\xi_i < t) &= F_\sigma(t) \qquad \text{if } i \in I^0
\end{aligned} \tag{21}$$

where

$$F_\sigma(a) := \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^a e^{-\frac{x^2}{2\sigma^2}}\, \mathrm{d}x$$

Notice now that

$$\xi_{[w]} < t \ \Leftrightarrow\ \Lambda_t := |\{i \,|\, \xi_i < t\}| \geq w \tag{22}$$

Put $\Lambda_t^q := |\{i \in I^q \,|\, \xi_i < t\}|$ for $q = 0,1$. Λ_t^1 and Λ_t^0 are two Binomial r.v. of type, respectively, $B(|I^1|, F_\sigma(t-1))$ and $B(|I^0|, F_\sigma(t))$. Since, $\Lambda_t = \Lambda_t^1 + \Lambda_t^0$, we have that $\mathbb{E}(\Lambda_t) = |I_1| F_\sigma(t-1) + |I_0| F_\sigma(t)$ and

$$\lim_{N\to+\infty} \frac{\mathbb{E}(\Lambda_t)}{N} = F_\xi(t) := pF_\sigma(t-1) + (1-p)F_\sigma(t) \tag{23}$$

Let us now consider the function $\overline{F}_N(\omega)$ defined as follows:

$$\overline{F}_N(\omega) = \frac{1}{N}F_N(N\omega) = -\omega^2+\omega-\frac{2}{N}\sum_{k=\lfloor N(1-\omega)\rfloor+1}^{N}\eta_{[k]}$$

$$= -\omega^2+\omega-2\omega\Omega-\frac{2}{N}\sum_{k=\lfloor N(1-\omega)\rfloor+1}^{N}\xi_{[k]}, \qquad \omega\in[N^{-1},1]$$

which is a normalized, scaled and interpolated version of the function $F_N(w)$.

Equations (22) and (23) suggest that $\xi_{[w]}$ and $F_\xi^{-1}(w/N)$ should be close to each other for large N. We can thus guess (formal proofs are omitted for reasons of space) that:

$$\lim_{N\to\infty}\overline{F}_N(\omega)\overset{a.s.}{=}\mathscr{F}(\omega):=-\omega^2+\omega+2p\omega-2\int_{1-\omega}^{1}F_\xi^{-1}(t)dt,\ \omega\in(0,1]$$

Likely enough, local extrema of F_N will converge, almost surely, to the local extrema of $\mathscr{F}$ so that if $\mathscr{F}$ possess just one local minimum on $[0,1/2]$ which is the global minimum, then this will also happen for F_N almost surely when $N\to+\infty$. This would mean that our decentralized algorithm will almost surely coincide with the centralized ML algorithm. Next section will make precise all these considerations.

4.1 The analysis of the function $\mathscr{F}(\omega)$

We start with some preliminary remarks on the function $\mathscr{F}$. It is immediate to verify that $\mathscr{F}$ is continuous. The other important fact is that can have one or two local minima depending on the particular values for σ and p, i.e. the derivative of $\mathscr{F}$ is equal to zero once or three times. However, the derivative of $\mathscr{F}$ seems to be equal to zero in only one point in $\omega\in(0,1/2)$ which corresponds to the global minimum.

The "small noise" case, i.e. the limit $\sigma\to 0$, deserves to be studied; this is done in the following proposition:

Proposition 5. Under the assumption of model given by Eqn. (3) we have that

$$\lim_{\sigma\to 0}\mathscr{F}(\omega) = -\omega^2+\omega+2p\omega-2p-2(\omega-p)\delta_{-1}(p-\omega)$$

$$\lim_{\sigma\to 0}\hat{\omega} := \operatorname*{argmin}_{\omega}\mathscr{F}(\omega) = p$$

where $\delta_{-1}(x)$ is equal to one for positive x and zero otherwise. We also have

$$\lim_{\sigma\to+\infty}\hat{\omega} := \operatorname*{argmin}_{\omega}\mathscr{F}(\omega) = \frac{1}{2}$$

The previous theorem states that if the the two distribution do not overlap, then the proposed algorithm exactly compute the proportions measurements generated by each of the two Gaussian distributions. However, when there is substantial overlap, the estimation has a bias toward the midpoint $1/2$, and in the limit of very large variance estimate $\hat{\omega}$ is completely uninformative.

The value of the minimum $\hat{\omega}$ of the asymptotic function $\mathscr{F}(\omega)$ as a function of the noise variance σ for $p=0.3$ is reported in Figure 1 (dotted line). As stated in the previous proposition, $\hat{\omega}=p$ for small σ and $\hat{\omega}=1/2$ for large σ. As mentioned above, the graph shows that this minimum monotonically increases from p to $1/2$, thus confirming the hypothesis that the global minimum is always in the interval $(0,1/2)$ for all values of p and σ. Figure 1 also shows the mean and standard deviation of the minimum of $F_N(\omega)$ over 10 Monte Carlo runs for $N=100$ sensor nodes.

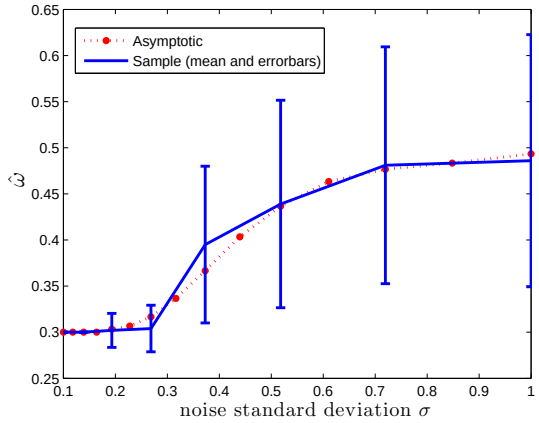

Fig. 1. Minima of $\mathscr{F}(\omega)$ (asymptotic) and of $\overline{F}_N(\omega)$ (sample, 10 Monte Carlo runs) vs. noise standard deviation σ. Data are always generated with $p=\frac{\sum_i T_i}{N}=0.3$.

4.2 The concentration results

In the sequel we present some concentration results which make rigorous the considerations done above. In the interest of space, all technical proofs are omitted.

We recall a standard result on the concentration of binomial r.v. which will be our main technical tool.

Theorem 6. Let Z be a binomial r.v of type $B(N,p)$. Put, for $x>0$, $\gamma(x)=x\log x-x+1$. Then, for any $x<1<y$, it holds

$$\mathbb{P}(Z\le Npx)\le e^{-Np\gamma(x)}, \quad \mathbb{P}(Z\ge Npy)\le e^{-Np\gamma(y)}$$

Remark: Notice that, for any $y_0>1$, there exists a constant $C>0$, such that $\gamma(y)\ge Cy\log y$

The following result is standard but we will give an elementary proof for the sake of making the paper self-contained.

Lemma 7. For any $0<a<b<1$ and for every $\delta>0$, there exists $l_\delta>0$ such that, for N sufficiently large,

$$\mathbb{P}(|\xi_{[j]}-F_\xi^{-1}(j/N)|\ge\delta)\le e^{-Nl_\delta}, \quad \forall j\in[aN,bN]$$

With the following bound we take care of the behavior of $\xi_{[j]}$ for j close to 0 and to N.

Lemma 8. There exist $0<a<1$ and $l>0$ such that, for N sufficiently large and for $j\in[1,aN]$, it holds

$$\begin{aligned}\mathbb{P}\left(\xi_{[j]}\le-(N/j)^{1/2}\right) &\le e^{-lN}\\ \mathbb{P}\left(\xi_{[N-j]}\ge(N/j)^{1/2}\right) &\le e^{-lN}\\ \mathbb{P}(\xi_{[N]}\ge N^{1/2}) &\le e^{-LN}\end{aligned} \tag{24}$$

Theorem 9. For every $\delta>0$ there exists $L_\delta>0$ such that, for N sufficiently large,

$$\mathbb{P}\left(\exists w:\left|\frac{F_N(w)}{N}-\mathscr{F}\left(\frac{w}{N}\right)\right|\ge\delta\right)\le e^{-NL_\delta}$$

Since our decentralized algorithm is influenced by the position of the local minima of F_N in $[0,1/2]$, the result above is not sufficient to study the performance. Indeed, we need to study the asymptotic behavior of the variation function $\Delta(w)$.

Theorem 10. For every $\delta>0$, there exists $\tilde{L}_\delta>0$ such that

$$\mathbb{P}\left(\exists w:\left|\Delta(w)-\mathscr{F}'\left(\frac{w}{N}\right)\right|\ge\delta\right)\le e^{-N\tilde{L}_\delta}$$

for N sufficiently large.

Proposition 11. Consider an interval $[a,b] \subseteq [0,1]$ and $\varepsilon > 0$. Then,

$$\begin{array}{l} \mathscr{F}'(x) \geq \varepsilon \ \forall x \in [a,b] \ \Rightarrow \mathbb{P}(\Delta(w) \geq 0 \, \forall w \in [Na,Nb]) \geq p_\varepsilon(N) \\ \mathscr{F}'(x) \leq -\varepsilon \ \forall x \in [a,b] \ \Rightarrow \mathbb{P}(\Delta(w) \leq 0 \, \forall w \in [Na,Nb]) \geq p_\varepsilon(N) \end{array} \tag{25}$$

where $p_\varepsilon(N) := 1 - Ce^{-\tilde{L}_\varepsilon N}$.

We are now ready to state and prove the main theoretical result of our work. Denote by S_N the set of local minima of F_N in $[0,1/2]$ and by S_N^{glob} the subset of S_N consisting of the global minima of F_N living in $[0,1/2]$ (of course a priori this set could as well be empty).

Corollary 12. Assume that

(a) $\min_{\omega \in [0,1/2]} \mathscr{F}(\omega) < \min_{\omega \in [1/2,1]} \mathscr{F}(\omega)$.

(b) $\mathscr{F}$ admits just one local minimum point $\bar{\omega}$ in $[0,1/2]$ (which is thus the only global minimum for (a)).

Then, for every $\delta > 0$, there exists $J_\delta > 0$ such that

$$\begin{array}{l} \mathbb{P}(S_N/N \subseteq]\bar{\omega} - \delta, \bar{\omega} + \delta[) \geq 1 - Ce^{-J_\delta N} \\ \mathbb{P}(S_N^{\text{glob}} \neq \emptyset) \geq 1 - Ce^{-J_\delta N} \end{array} \tag{26}$$

5. BAYESIAN MODELING AND EM

An alternative approach to this estimation and detection problem is possible if one postulates that T_i, $i = 1,..,N$ are independent and identically distributed (i.i.d.) binary random variables, taking values in $\{0,1\}$. This implies that the T_i's are Bernoulli random variables with parameter p

$$T_i \sim \mathscr{B}(p) \quad p := P[T_i = 1]. \tag{27}$$

so that

$$\mathbb{P}(T|p) = \prod_{i=1}^{N} p^{T_i}(1-p)^{1-T_i}$$

Hence, one can formulate the problem of estimating p, θ and σ from measurements $y_1,...,y_N$. The maximum likelihood estimator is defined by

$$(\hat{p}, \hat{\theta}, \hat{\sigma}) := \arg\max_{p,\theta,\sigma} P(y|\theta,T)P(T|p) \tag{28}$$

Note that in the estimation problem (3) the number of unknowns grows with the number of data; instead the i.i.d. assumption on the T_i's allows to keep the parameter space in (28) of fixed dimension. As a result, the asymptotic properties of the estimators in (28), such as consistency and asymptotic efficiency, follow straightforwardly from standard asymptotic theory of maximum likelihood estimators, see Zacks (1971).

An estimator of the variables $T_1,..,T_N$ can then be obtained by maximizing the posterior probability

$$(\hat{T}_1,..\hat{T}_N) := \arg\max_{T \in \{0,1\}^N} \hat{p}(T|y,\theta,p,\sigma).$$

The maximum likelihood estimator $\hat{p}(T|y,\theta,p,\sigma)$ of the posterior probability $p(T|y,\theta,p,\sigma)$ is given, from the invariance principle (see e.g. Zacks (1971)), by

$$\begin{array}{rcl} \hat{p}(T|y,\theta,p,\sigma) & = & p(T|y,\hat{\theta},\hat{p},\hat{\sigma}) \\ & = & ce^{-\frac{1}{2}\sum_{i=1}^{N}\left(\frac{y_i-\hat{\theta}-T_i}{\hat{\sigma}}\right)^2 + ln\left(\frac{\hat{p}}{1-\hat{p}}\right)\sum_{i=1}^{N} T_i} \end{array} \tag{29}$$

where c is a suitable normalization constant.

The maximum likelihood problem (28) is a typical estimation problem for a finite mixture distribution (see Titterington et al. (1985)) and does not have a closed form solution. One possible approach is to resort to the well known Expectation-Maximization (EM) algorithm in Dempster et al. (1977). This is an iterative algorithm which is known to converge to a local maxima of the likelihood. For reasons of space we shall only report the final equations for EM iterations; we refer the reader to the book by Titterington et al. (1985) for a derivation of the EM algorithm which can be easily adapted to this specific problem.

Let $\hat{\theta}^{(k)}, \hat{\sigma}^{(k)}$ and $\hat{p}^{(k)}$ the estimators at the $k-th$ iteration of the EM algorithm; the estimators for the $(k+1)$-th iteration are given by:

(1) **Expectation Step**: compute the posterior probabilities

$$\begin{array}{rcl} \hat{\mu}_j^{(k+1)} & := & p(T_j|y,\hat{\theta}^{(k)},\hat{p}^{(k)},\hat{\sigma}^{(k)}) \\ & = & \dfrac{\hat{p}^{(k)} e^{-\frac{1}{2}\left(\frac{y_j-\hat{\theta}^{(k)}-1}{\hat{\sigma}^{(k)}}\right)^2}}{\hat{p}^{(k)} e^{-\frac{1}{2}\left(\frac{y_j-\hat{\theta}^{(k)}-1}{\hat{\sigma}^{(k)}}\right)^2} + (1-\hat{p}^{(k)}) e^{-\frac{1}{2}\left(\frac{y_j-\hat{\theta}^{(k)}}{\hat{\sigma}^{(k)}}\right)^2}} \end{array} \tag{30}$$

(2) **Maximization Step**:

$$\begin{array}{rcl} \hat{p}^{(k+1)} & = & \dfrac{1}{N}\sum_{j=1}^{N} \hat{\mu}_j^{(k+1)} \\ \hat{\theta}^{(k+1)} & = & \dfrac{1}{N}\sum_{j=1}^{N} y_j - \hat{p}^{(k+1)} \\ \hat{\sigma}^{(k+1)} & = & \sqrt{\dfrac{1}{N}\sum_{j=1}^{N}\left(\underbrace{(y_j-\theta)^2 + \mu_j - 2\mu_j(y_j-\theta)}_{\theta=\hat{\theta}^{(k+1)} \quad \mu_j=\hat{\mu}_j^{(k+1)}}\right)} \end{array} \tag{31}$$

The EM algorithm (30),(31) has a "centralized" nature; however it can be easily decentralized (i.e. computed by each node only using local information) since it is essentially based upon computing averages. It is well known that this can be done resorting to consensus algorithms; for instance an algorithm based on gossip has been proposed by Kowalczyk and Vlassis (2005). In this paper we have implemented the averages in (31) using a symmetric gossip algorithm assuming the nodes are connected via a complete graph. We shall refer to this algorithm as *distributed-EM*.

As expected, if the number of gossip iterations is sufficient to reach consensus, the distributed-EM algorithm converges to the maximum likelihood estimator (28). However, as soon as the number of iterations is not sufficient to reach consensus, the distributed-EM algorithm either oscillates or even diverge, failing to provide sensible estimates. This simple simulation experiments suggest that distributed-EM is not robust against errors in computing the averages in (31) which may result from an insufficient number of consensus iterations.

6. GENERALIZATION

One drawback of the model in (1) is that the T_i's are assumed to belong to a known alphabet $\mathscr{A}$. In particular in this paper we have considered the case $T_i \in \{0,1\}$. A simple yet important generalization is to allow that the alphabet is partially unknown.

For instance one can assume that only the cardinality of $\mathscr{A}$ is known. In the binary case considered in this paper this is equivalent to assume that

$$y_i = \theta + \alpha T_i + v_i \tag{32}$$

with $T_i \in \{0,1\}$ and $\alpha \in \mathbb{R}^+$ unknown [1].

In this more general scenario the maximum likelihood estimator (3) becomes:

$$\begin{aligned}(\hat{\theta}^{ML}, \hat{T}^{ML}, \hat{\alpha}^{ML}) &= \underset{(\theta,T,\alpha)}{\operatorname{argmax}} P(y|\theta,T,\alpha) \\ &= \underset{(\theta,T,\alpha)}{\operatorname{argmin}} \left[\sum_i \frac{(y_i-\theta-\alpha T_i)^2}{2\sigma^2}\right]\end{aligned} \tag{33}$$

Solving (33) is considerably more difficult than (3); one possible approach is to utilize and alternating minimization algorithm as follows:

(i) Fix $\alpha := \hat{\alpha}^{(k-1)}$ and solve

$$\hat{T}^{(k)}(\alpha) = \underset{T}{\operatorname{argmin}} \min_{\theta} \left[\sum_i \frac{(y_i-\theta-\alpha T_i)^2}{2\sigma^2}\right] \tag{34}$$

(ii) Fix $T := \hat{T}^{(k)}$ and solve

$$(\hat{\theta}^{(k)}(T), \hat{\alpha}^{(k)}(T)) = \underset{(\theta,\alpha)}{\operatorname{argmin}} \left[\sum_i \frac{(y_i-\theta-\alpha T_i))^2}{2\sigma^2}\right] \tag{35}$$

Problem (34) is analogous to (3) with the only difference that in (3) we assume $\alpha = 1$. Hence this can be solved as described in Section 2.1.

Instead, problem (35) admits a closed form solution as:

$$\hat{\theta}^{(k)} = \frac{\sum_i y_i}{N} - \hat{\alpha}^{(k)} \frac{\sum_i \hat{T}_i^{(k)}}{N} \qquad \hat{\alpha}^{(k)} = \frac{\frac{\sum_i \hat{T}_i^{(k)} y_i}{N} - \frac{\sum_i \hat{T}_i^{(k)}}{N}\frac{\sum_i y_i}{N}}{\frac{\sum_i \hat{T}_i^{(k)}}{N}\left(1 - \frac{\sum_i \hat{T}_i^{(k)}}{N}\right)} \tag{36}$$

In Section (7) we shall also report simulation experiments, in which α is not assumed to be known, using the alternating minimization approach above; experimental evidence shows that this alternating minimization algorithm converges in few steps (2 or 3) in all the examples considered. Of course, in the distributed scenario the averages in (36) will have to be computed resorting to consensus algorithms.

As an alternative one could also consider the Bayesian formulation in Section 5 for the measurement model (32). This is standard estimation problem for a mixture of two Gaussian distributions with unknown means and unknown (but common) variance. An EM algorithm similar to (30), (31) in Section 5 can be derived (see Titterington et al. (1985)). Of course, in the distributed setting, averages will have to be computed using consensus algorithm, with the same limitations discussed in Section 5.

7. SIMULATIONS

In order to compare the algorithm introduced in this paper with more standard EM algorithms (based on gossip iterations, as proposed in Section 5, 6), we consider the following setup. In Example 1 (see Fig. 2) we assume $N = 50$ sensors are deployed which measure data according to the model (1) or equivalently according to the model (32) with $\alpha = 1$. We generate data with $\theta = 0$, $\sigma = 0.3$ and assume that T_i are i.i.d. Bernoulli random variables with mean $p = 0.3$. In order to test the robustness of the algorithms against outliers, in Example 2 we consider a second setup in which data are generated as in Example 1, except for an outlier $y_0 = -2$ which is artificially added.

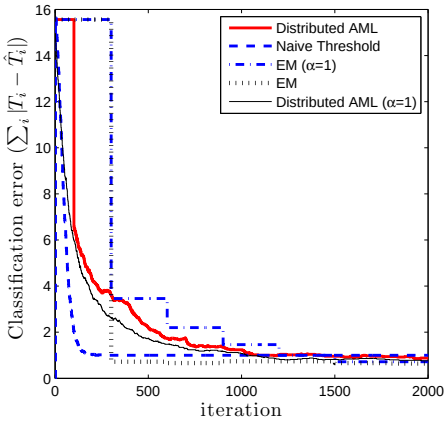

Fig. 2. Example 1: Average (over 50 Monte Carlo runs) of the classification error $\sum_{i=1}^N |T_i - \hat{T}_i|$ as a function of the number of gossip iterations. Data are generated as follows: $\theta = 0$, $T_i \sim \mathscr{B}(0.3)$, $\sigma = 0.3$.

We compare the following algorithms:

(1) **Distributed AML ($\alpha = 1$)**: this is the distributed approximate Maximum Likelihood described in Section 3 which is based on the model (1) with $T_i \in \{0,1\}$ as in Section 2.
(2) **Distributed AML**: the distributed approximate Maximum Likelihood based on model (32), which also estimates α using the alternating maximization approach described in Section 6.
(3) **EM ($\alpha = 1$)**: this is the distributed implementation of the EM algorithm introduced in Section 5, based on the measurement model (1) with $T_i \in \{0,1\}$ as in Section 2.
(4) **EM**: this is the distributed implementation of the EM algorithm for the estimation of a mixture of two Gaussian distributions with different and unknown means discussed at the end of Section 6.
(5) **Naive threshold**. This is the most naive algorithm one can come up with: classify measurements based on the following rule:

$$T_i = \begin{cases} 1 & y_i > \dfrac{\min\{y_i\} + \max\{y_i\}}{2} \\ 0 & \text{otherwise} \end{cases}$$

In its distributed version the maximum and the minimum can be calculated using a distributed ranking algorithm as in Section 3.1.

The simulation results show that there is not a clear-cut distinction between different algorithms. The EM algorithm is not robust if the number of gossip iterations between successive M-steps and E-steps is not sufficient to reach "almost" consensus (i.e. compute reliably enough the averages in (31)). The number of these gossip iterations has been fixed to 300 in our simulation experiments; this number seemed large enough to reach essen-

[1] It is immediate to show that, for identifiability reasons, only the difference between the two symbols have to be parameterized; in addition this difference can be assumed to be positive modulo permutations.

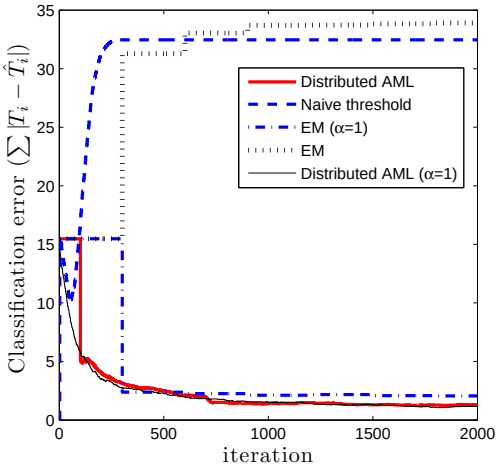

Fig. 3. Example 2 (with outlier): Average (over 50 Monte Carlo runs) of the classification error $\sum_{i=1}^{N} |T_i - \hat{T}_i|$ as a function of the number of gossip iterations. Data are generated as follows: $\theta = 0$, $T_i \sim \mathscr{B}(0.3)$, $\sigma = 0.3$. An outlier is added to each Monte Carlo realization by setting $y_1 = -2$.

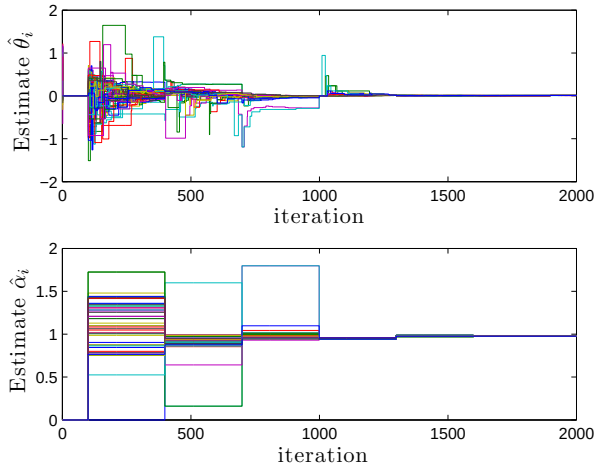

Fig. 4. Distributed AML: estimates $\hat{\theta}_i$ and $\hat{\alpha}_i$ for each node i as a function of the number of gossip iterations.

tially consensus in the setup we consider while smaller values gave sometimes rise to unstable results. On the other hand, the "naive" thresholding approach is not robust against the possible presence of outliers. Also the EM algorithm which does not assume α to be known seems to get trapped in a local minima (see Figure 3). The algorithm introduced in this paper seems, overall, a bit slower than its competitors, but more robust in the examples considered. The simulation results suggest that, if α is not assumed to be known (and hence it has to be estimated as described in Section 6), then the algorithm essentially has the same performance. One typical realization of the estimators $\hat{\theta}_i$ and $\hat{\alpha}_i$ (estimators for θ and α at the i-th node) obtained by the distributed AML algorithm are reported in Figure 4.

8. CONCLUSIONS

In this work we studied the problem of distributively computing simultaneous binary classification and noisy parameter estimation in network of distributed sensors subject to topological communication constraints. The proposed ML strategy has shown different trade-offs as compared to an EM approach in terms of speed of convergence and robustness in particular when the offset of the "misbehaving" sensors is not known. Different research avenues are possible, such as the generalization of the distributed ranking to simply connected graphs, the generalization to multiple class, and the development of more robust strategies when the offset is unknown.

REFERENCES

Bandyopadhyay, S., Giannella, C., Maulik, U., Kargupta, H., Liu, K., and Datta, S. (2006). Clustering distributed data streams in peer-to-peer environments. *Information Sciences*, 176(14), 1952 – 1985.

Berkhin, P. (2006). *Grouping Multidimensional Data: Recent Advances in Clustering*, volume 2466/2002 of *Lecture Notes in Computer Science*, chapter A Survey of Clustering Data Mining Techniques, 25–71. Springer.

Carli, R., Chiuso, A., Schenato, L., and Zampieri, S. (2008). Distributed Kalman filtering based on consensus strategies. *IEEE Journal on Selected Areas in Communications*, 26, 622–633.

Chiuso, A., Fagnani, F., Schenato, L., and Zampieri, S. (2010). Simultaneous distributed estimation and classification in sensor networks. Technical report, Univ. of Padova. Available at www.dei.unipd.it/∼chiuso/DOWNLOAD/NECSYS.pdf.

Dempster, A.P., Laird, N.M., and Rubin, D.B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1), 1–38.

Duda, R., Hart, P., and Stork, D. (2001). *Pattern classification*. Wiley Interscience.

Kowalczyk, W. and Vlassis, N. (2005). Newcast EM. In *Advances in Neural Information Processing*.

Nowak, R. (2003). Distributed EM algorithms for density estimation and clustering in sensor networks. *IEEE Trans. on Signal Processing*, 51(8), 2245–1208.

Olfati-Saber, R. (2005). Distributed Kalman filter with embedded consensus filters. *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on*, 8179–8184.

Olfati-Saber, R., Fax, J.A., and Murray, R.M. (2007). Consensus and cooperation in networked multi-agent systems. *Proocedings of IEEE*, 95(1), 215–233.

Olfati-Saber, R. and Murray, R.M. (2004). Consensus problems in networks of agents with switching topology and time-delays. *Automatic Control, IEEE Transactions on*, 49(9), 1520–1533. doi:10.1109/TAC.2004.834113.

Rabbat, M. and Nowak, R. (2004). Distributed optimization in sensor networks. In *IPSN '04: Proceedings of the 3rd international symposium on Information processing in sensor networks*, 20–27. ACM.

Safarinejadian, B., Menhaj, M.B., and Karrari, M. (2010). Distributed variational Bayesian algorithms for Gaussian mixtures in sensor networks. *Signal Process.*, 90(4), 1197–1208.

Titterington, D., Smith, A., and Makov, U. (1985). *Statistical Analysis of Finite Mixture Distributions*. John Wiley and Sons.

Zacks, S. (1971). *The Theory of Statistical Inference*. Wiley Series in Probability and Mathematical Statistics. Wiley.