Distributed Data Association for Multi–Target Tracking in Sensor Networks

Lei Chen, Müjdat Çetin, and Alan S. Willsky

Stochastic Systems Group

Laboratory for Information and Decision Systems

Massachusetts Institute of Technology, Cambridge, MA 02139

Email: {lchen, mcetin, willsky}@mit.edu

Abstract— Associating sensor measurements with target tracks is a fundamental and challenging problem in multi–target tracking. The problem is even more challenging in the context of sensor networks, since association is coupled across the network, yet centralized data processing is in general infeasible due to power and bandwidth limitations. Hence efficient, distributed solutions are needed. We propose techniques based on graphical models to efficiently solve such data association problems in sensor networks. Our approach takes advantage of the sparsity inherent in the problem structure resulting from the fact that each target can be observed by only a small number of sensors and makes use of efficient message–passing algorithms for graphical models to infer the maximum *a posteriori* association configuration. We illustrate our approach for several typical scenarios in multi–target tracking. Our approach scales well with the number of sensor nodes in the network, and it is well–suited for distributed implementation. Distributed inference is realized by a message–passing algorithm which requires iterative, parallel exchange of information among neighboring nodes on the graph. So as to address trade–offs between inference performance and communication costs, we also propose a communication–sensitive form of message–passing that is capable of achieving near–optimal performance using far less communication. We demonstrate the effectiveness of our approach with experiments on simulated data.

I. Introduction

Recent years have witnessed the emergence of a new approach to sensing applications, involving the deployment of a large number of small, myopic and relatively inexpensive sensors. Such *ad hoc* sensor networks (SN) have the potential to provide enhanced spatio–temporal sensing coverage in ways that are either prohibitively expensive or even impossible using more conventional approaches to sensing [1]. However, realizing the potential of these large and distributed sensor networks requires the development of techniques for *distributed inference* using sensing and wireless communication nodes with constrained capacities for both computation and communication. This paper is devoted to developing distributed techniques for a particular inference problem in SN, namely the *data association* problem. Our primary motivation comes from problems that arise in multi–target tracking (MTT) with SN.

In the MTT context, data association is a fundamental problem and it involves finding the correct correspondence between measurements and target tracks [2]. The multiple hypothesis tracking (MHT) approach [3] is the most successful data association algorithm in the target–dense and clutter–dense environment [4]. A practical MHT algorithm employs the so–called N–scan technique [5], which enumerates every possible association for the measurements in the temporal window from $t = k$ to $t = k+N$ to build a hypothesis tree, and determines the most likely association at $t = k$ by evaluating the measurement likelihood of each hypothesis branch. Even with a single sensor, the problem is challenging due to the exponential growth of the number of hypotheses. In SN–based tracking, each target may be observed by multiple sensors, which couples the association decisions across the network, hence makes the problem even more challenging. With a centralized MHT approach, the zero–scan (using measurements at $t = k$ only) data association in an M–sensor network has the same complexity as an M–scan data association at one node. If each sensor has n possible association configurations, there could be as many as n^M configurations, which would make the problem intractable for large M even in the zero–scan case. Considering multiple scans of data increases the complexity even further. An alternative distributed MHT algorithm has been proposed in [6], which focuses on identifying redundant information in the fusion process, but this is not a trivial task in a large network with many loops. As a result, progress in MTT with SN requires the development of efficient, distributed data association algorithms.

In this paper, we propose a new approach to solving the data association problem in SN, using the framework of *graphical models* [7]. A typical scenario we are considering is shown in Fig. 1, where a large number of sensors are deployed to surveil a vast area. Each sensor has a limited detection range and its surveillance region partially overlaps with those of its neighbors. The sensors are synchronized and each sensor receives noise–corrupted measurements (e.g., bearing) from targets in its coverage area simultaneously at each scan. One contribution of this paper is that we propose a graphical model–based data association approach to solve the above problem efficiently and in a distributed fashion. The efficiency of our approach is mainly due to the observation that although the overall problem involves a large number of sensors and targets, each target is only observable by a small number of sensors at a time, thus the association at two distant sensors is conditionally independent of each other. Such a sparse structure inherent in the problem can be exploited by graphical
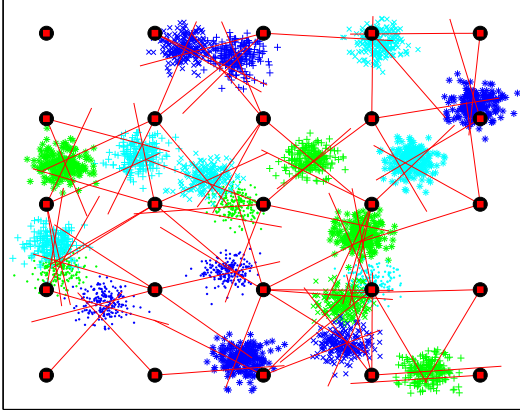
Fig. 1. A snapshot of a typical data association scenario in SN. 25 sensors (circle nodes) and the bearing-only measurements (line segments) are shown. Each cluster of samples represents the prior position distribution of one target.

models, which are well–suited to represent the structure of statistical dependencies of a collection of random variables. With a carefully designed modeling approach to transform the data association problem into the framework of graphical models, we obtain an inference problem that can be solved efficiently by the max–product algorithm [8]. Furthermore, the resulting algorithm can be implemented in a distributed fashion through *parallel message–passing*, where each sensor performs some local processing of the measured data and communicates with its neighbors. Such iterative exchange of information leads to a distributed solution of the data association problem.

Another contribution of this paper is that we propose a communication–sensitive version of the message–passing algorithm, which can achieve near–optimal solution using much less communication (thus power) than standard message–passing. In the standard algorithm, the nodes continue transmitting messages until the overall algorithm converges, which may consume a large amount of communication and power. This is not ideal for SN applications because the communication and power resource for each sensor is usually limited and should be sparingly used. In contrast, the communication–sensitive message–passing algorithm provides the sensor nodes with the authority to decide whether or not messages should be sent at each iteration, based on statistical rules related to the information content of messages. We show that our new algorithm can provide considerable savings in communication without much sacrifice in overall decision–making performance. Furthermore, this approach provides insights into performance–communication trade–offs, as well as into information flow dynamics inside SN.

The remainder of this paper is organized as follows. Section II introduces the essential background on graphical models and message–passing algorithms. We propose our graphical model–based data association approach in Section III. The communication–sensitive message–passing algorithm is proposed in Section IV. Experimental results are presented in

Section V. We summarize our work and discuss directions for future research in Section VI.

II. BACKGROUND ON GRAPHICAL MODELS AND MESSAGE–PASSING ALGORITHMS

A graphical model consists of a collection of random variables $\mathbf{x} := \{\mathbf{x}_s\}$ that are associated with the nodes of a graph $\mathcal{G} = (V, E)$, where V is the set of nodes, and E is the set of edges. The conditional dependency among the random variables is represented by the graph structure. Here we consider undirected graphical models with discrete variables. When $\mathbf{x}$ is *Markov* with respect to the graph, the distribution $p(\mathbf{x})$ can be factorized as the product of functions defined on the cliques of the graph [9]. If the random variables have at most pairwise interactions, which is the case in our work, then the factorization of $p(\mathbf{x})$ takes the form

$$p(\mathbf{x}) = \kappa \prod_{s \in V} \psi_s(\mathbf{x}_s) \prod_{(s,t) \in E} \psi_{st}(\mathbf{x}_s, \mathbf{x}_t), \qquad (1)$$

where κ is a normalization constant, $\psi_s(\mathbf{x}_s)$ is the *node compatibility function* that depends only on the individual variable $\mathbf{x}_s$, and $\psi_{st}(\mathbf{x}_s, \mathbf{x}_t)$ is the *edge compatibility function* that depends only on the variables $\mathbf{x}_s$ and $\mathbf{x}_t$ joined by edge (s,t). In many applications, the random vector $\mathbf{x}$ is not observed; given instead are independent noisy observations $\mathbf{y} = \{y_s \mid s \in V\}$ based on which $\mathbf{x}$ needs to be inferred. The effect of including these measurements — i.e., the transformation from the prior distribution $p(\mathbf{x})$ to the conditional distribution $p(\mathbf{x} \mid \mathbf{y})$ — is simply to modify the factors in (1). As a result, we suppress explicit mention of measurements in this section, since the problems involving either $p(\mathbf{x} \mid \mathbf{y})$ or $p(\mathbf{x})$ are of identical structure and complexity.

Of interest in this paper is the problem of finding the maximum *a posteriori* (MAP) configuration

$$\hat{\mathbf{x}} = \arg \max_{\mathbf{x} \in \mathcal{X}^N} p(\mathbf{x}). \qquad (2)$$

The max–product algorithm [8], a generalization of the Viterbi algorithm on Markov chains to arbitrary tree–structured graphs, can be used to solve this problem efficiently. A distributed implementation of the algorithm entails an iterative procedure called *parallel message–passing*, where each iteration involves each node t passing a message to each of its neighbors $s \in \mathcal{N}(t)$ simultaneously and in parallel, as shown in Fig. 2. The message in the k-th iteration, which we denote by $M_{ts}^k(\mathbf{x}_s)$, is a function of the possible states $\mathbf{x}_s \in \mathcal{X}_s$. In the max–product algorithm, the messages are updated according to the recursion

$$M_{ts}^k(\mathbf{x}_s) = \kappa \max_{\mathbf{x}_t'} \left\{ \psi_{st}(\mathbf{x}_s, \mathbf{x}_t') \psi_t(\mathbf{x}_t') \prod_{u \in \mathcal{N}(t) \backslash s} M_{ut}^{k-1}(\mathbf{x}_t') \right\}, \qquad (3)$$

where $\mathcal{N}(t) \backslash s$ is the set of neighbors of node t in the graph $\mathcal{G}$ excluding node s. For any tree–structured graph, the message update equation (3) converges to a unique fixed point $\mathbf{M}^* = \{M_{ts}^*\}$ after a finite number of iterations. The converged values
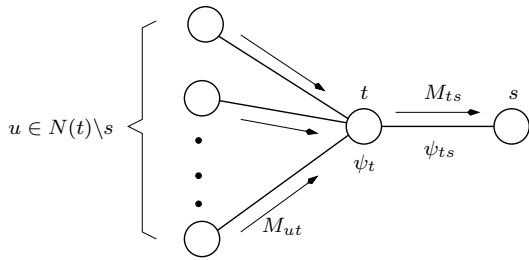
Fig. 2. Parallel message passing for graphical models. Only messages relevant to the computation of the message from node t to node s are shown.

of the messages $\mathbf{M}^*$ can be used to compute the so–called max–marginal $P_s(\mathbf{x}_s) = \kappa p(\mathbf{x}')$ at each node

$$P(x_s) = \kappa \max_{\{\mathbf{x}|\mathbf{x}_s = x_s\}} \psi_s(\mathbf{x}_s) \prod_{u \in \mathcal{N}(s)} M^*_{us}(\mathbf{x}_s). \qquad (4)$$

The MAP configuration $\hat{\mathbf{x}}$ is given by $\hat{\mathbf{x}}_s = \arg\max_{\mathbf{x}_s} P_s(\mathbf{x}_s)$.

For tree–structured problems, the max–product algorithm produces exact solutions with complexity $\mathcal{O}(n^2 d)$, where n is the number of states per node, and d is the number of nodes on the longest path in the graph. The same algorithm is also applied frequently to graphs with cycles, although in that case it serves as an approximate method. A modified version of the max–product algorithm, tree–reweighted max–product (TRMP), is proposed in [10]. The TRMP algorithm outputs the correct MAP assignment even on loopy graphs under mild conditions. More details on TRMP can be found in [10].

III. GRAPHICAL MODEL–BASED DATA ASSOCIATION

For simplicity, we first discuss how to use measurements of the current scan only to solve the data association problem in SN, which corresponds to setting $N = 0$ in the N–scan technique. Incorporating multi–scan data to solve the association at the current scan will be discussed in Section III-B.

A. Zero–Scan Data Association

We consider a scenario where M sensors are deployed to surveil a 2–D planar field. We assume sensor calibration and localization has been accomplished (for example, using the techniques proposed in [11]) such that each sensor knows its own location in a global coordinate. Each sensor has a limited and well–defined detection region. Due to the overlapping sensor coverage, the whole surveillance area is divided into disjoint subregions $\{r_1, r_2, \ldots\}$, each of which is covered by a distinct subset of sensors. The sensors generate noisy bearing, range, or 2–D position measurements for the targets present in their own surveillance region by a given measurement model. Let $\mathcal{Y}_i$ be the set of measurements generated by sensor s_i. The data association problem is to assign measurements in $\mathcal{Y}_i$ to the targets present in the surveillance region of sensor s_i given some prediction–based prior information on target locations. Let $\mathbf{x}_i$ be the association variable for s_i that takes values in

the set of all valid association configurations of s_i, where an association configuration is valid if and only if in such a configuration the measurements in $\mathcal{Y}_i$ and the targets covered by s_i are in a one–to–one correspondence[1]. Note that if each sensor s_i knows which subset of targets it can observe, then the sample space of variable $\mathbf{x}_i$ can be obtained by enumerating the mapping between the measurements reported by s_i and the targets covered by s_i. Otherwise, when such a sensor–target coverage relation is ambiguous, we need special consideration to obtain the sample space of the association variables, which will be described in Section III-A.2. The maximum likelihood (ML) data association $\hat{\mathbf{x}} = \{\hat{\mathbf{x}}_i\}$ satisfies

$$\hat{\mathbf{x}} = \arg \max_{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M} p(\mathcal{Y} \mid \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M), \qquad (5)$$

where $\mathcal{Y} = \bigcup_{i=1}^M \mathcal{Y}_i$ is the set of all measurements in the network. However, due to the overlapping coverage, measurement noise, and the uncertainty of the target locations, the random variables $\{\mathbf{x}_i\}$ are correlated, and evaluating the measurement likelihood for every possible value of $\mathbf{x}$ is generally infeasible for a large–scale SN.

Our approach to attack this problem is to construct a graphical model such that each $\mathbf{x}_i$ is represented by a graph node and the measurement likelihood in Eq. (5) takes a factorized form on the model as in Eq. (1). Assuming that each valid association configuration is equally likely, finding the ML estimates in Eq. (5) is equivalent to finding the MAP estimates of $\mathbf{x}$, hence the problem can be solved by the max–product algorithm. If each sensor has n possible association configurations, thus each $\mathbf{x}_i$ has n states, then with M variables to be estimated, the complexity of our approach is roughly $\mathcal{O}(n^2 M)$, which is much less than the centralized MHT approach. However, all of this relies on constructing a graphical model and factorizing the measurement likelihood function in Eq. (5) to the pairwise compatibility functions for the above problem. We present our approach to constructing such graphical models for the data association problem in the following two scenarios. For each scenario, we work on a toy example to illustrate our modeling approach and describe how to relate Eq. (5) to Eq. (1) in each case.

1) Completely Organized Sensor Networks: We say a network is *completely organized* if each sensor knows which targets it can observe[2]. In such a case, the subregion in which each target is located is known. A piece of such a completely organized SN is shown in Fig. 3(a), where targets T_1, T_2, and T_3 are known to be present in the subregions r_1, r_2, and r_3, respectively. In this completely organized scenario, the measurement likelihood in Eq. (5) can be factorized into two

---

[1]With possible false alarms and missed detections, the real measurements and the real targets are not in one–to–one correspondence. However, missed detections and false alarms can be handled and incorporated into our framework, as described in [12], by introducing virtual measurements and virtual targets. Thus, for the sake of simplicity, we ignore them in the following discussion.

[2]In this paper, the meaning of "organized" is not the same as in network self–organization literature, where "organized" means each node in the network has established connection with its neighbors and has received information about their status.
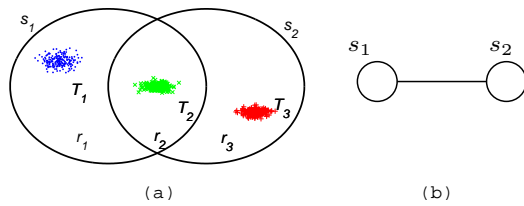
Fig. 3. (a) A piece of a completely organized SN. Two sensors with their surveillance region and non–parametric representations of three target distributions are shown. (b) The graphical model for the scenario in (a).
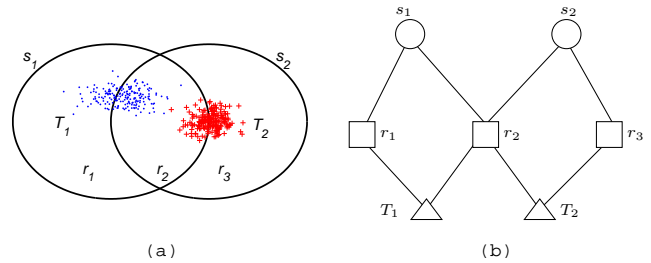

Fig. 4. (a) A piece of a partially organized SN. Two sensors with their surveillance region and non–parametric representations of two target distributions are shown. (b) The graphical model for the scenario in (a).

parts. The first part consists of the product of the likelihoods of the measurements assigned to targets covered by a single sensor. The second part consists of the product of the likelihoods of measurements assigned to targets covered by multiple sensors. Hence a graphical model for data association in an organized SN contains a node representing the association variable $\mathbf{x}_i$ for each sensor. The nodes that correspond to sensors observing common targets are connected, as shown in Fig. 3(b). The node compatibility function is defined as the measurement likelihood for those measurements assigned to the targets covered by only the corresponding sensor. The edge compatibility function is defined as the joint measurement likelihood for the measurements assigned to the targets in the corresponding shared subregion. The following example illustrates how to model the scenario shown in Fig. 3(a).

Example 1 For the scenario shown in Fig. 3(a), the graphical model is shown in Fig. 3(b). Suppose $\mathcal{Y}_1 = \{y_{11}, y_{12}\}$ and $\mathcal{Y}_2 = \{y_{21}, y_{22}\}$, then the node states and corresponding compatibility functions are

states of $\mathbf{x}_1$ for node s_1

y_{11}	y_{12}	ψ_{s_1}
T_1	T_2	$p(y_{11}; T_1)$
T_2	T_1	$p(y_{12}; T_1)$

states of $\mathbf{x}_2$ for node s_2

y_{21}	y_{22}	ψ_{s_2}
T_2	T_3	$p(y_{22}; T_3)$
T_3	T_2	$p(y_{21}; T_3)$

and the edge compatibility function is defined as

$$\psi_{s_1 s_2}: \begin{bmatrix} p(y_{12}, y_{21}; T_2) & p(y_{12}, y_{22}; T_2) \\ p(y_{11}, y_{21}; T_2) & p(y_{11}, y_{22}; T_2) \end{bmatrix}$$

such that $p(\mathcal{Y} \mid \mathbf{x}_1, \mathbf{x}_2) = \psi_{s_1} \psi_{s_2} \psi_{s_1 s_2}$, i.e., $p(\mathcal{Y} \mid \mathbf{x}_1, \mathbf{x}_2)$ is factorized as in Eq. (1).

When a target is covered by three or even more sensors, a target node can be introduced into the graph to avoid the high–order interactions and keep the compatibility functions pairwise, thus yielding a sensor–target hybrid modeling approach. For details please refer to [12].

2) Partially Organized Sensor Networks: In Section III-A.1, we have assumed that we perfectly know which subregion each target is located in. However, in practice that information may contain uncertainties, for example, when the target is moving across a sensor's detection boundary, or when the predicted target location has very large uncertainty. In such a *partially organized* network, several subregions might be postulated to have a particular target located within their own boundaries with certain non–zero probabilities. Fig. 4(a) shows

a piece of a partially organized SN, where target T_1 is possibly in either r_1 or r_2, and target T_2 is possibly in either r_2 or r_3. In such cases, which subset of targets is observed by each sensor remains to be estimated. The association problem thus consists of both the task of associating targets to subregions, as well as the task of associating measurements to targets. Note that in addition to the one–to–one constraints on measurement association, a new set of constraints that each target can only be assigned to one subregion must also be enforced.

We propose a region–based modeling approach to transform such a partially organized scenario into the framework of graphical models. To this end, we define $\mathbf{x}_i$ on the Cartesian product of the measurement set $\mathcal{Y}_i$ and the set of possible target–subregion pairs covered by sensor s_i. Let $\tilde{\mathbf{x}}_j$ be a random variable defined for subregion r_j taking values on the power set of the set consisting of all the targets that are possibly present in region r_j. Note that when two subregion r_j and r_k compete for the same target, the values of $\tilde{\mathbf{x}}_j$ and $\tilde{\mathbf{x}}_k$ should be mutually exclusive regarding the target. A region–based model contains a node for each sensor, for each subregion, and for each target, as shown in Fig. 4(b). The association variables $\mathbf{x}_i$'s and $\tilde{\mathbf{x}}_j$'s are represented by the corresponding sensor nodes and subregion nodes, respectively. Each sensor node has its compatibility function defined as $\psi_{s_i} = 1$, and each subregion node has the compatibility function defined according to the incidence probability for the potential targets in the subregion. The target nodes are auxiliary nodes for introducing into the model the measurement likelihood under various association configurations. The states of a target node enumerate the Cartesian product of the set of possible subregions the target may be in and the set of possible measurements it can be associated with. The measurement likelihood associated with the target defines the compatibility function of each target node. The nodes in the graph are connected in an intuitive way. A subregion node is connected with every sensor node that covers the subregion, and each target is connected with each subregion it could enter. The consistency between the nodes needs to be ensured by the edge compatibility functions. In the special case of complete organization, this modeling approach reduces to the models described in Section III-A.1 (after appropriate node aggregation). The following example shows how to construct

a graphical model for the scenario shown in Fig. 4(a).

Example 2 For the scenario shown in Fig. 4(a), the graphical model is shown in Fig. 4(b). Suppose $\mathcal{Y}_1 = \{y_{11}, y_{12}\}$ and $\mathcal{Y}_2 = \{y_{21}, y_{22}\}$, the states and compatibility functions of nodes s_1, r_2 and T_1 are defined as

states of $\mathbf{x}_1$ for node s_1

y_{11}	y_{12}	ψ_{s_1}
$T_1 \rightarrow r_1$	$T_2 \rightarrow r_2$	1
$T_1 \rightarrow r_2$	$T_2 \rightarrow r_2$	1
$T_2 \rightarrow r_2$	$T_1 \rightarrow r_1$	1
$T_2 \rightarrow r_2$	$T_1 \rightarrow r_2$	1

states of $\tilde{\mathbf{x}}_2$ for node r_2

Targets	ψ_{r_2}
$\{T_1\}$	$p(T_1 \rightarrow r_2)$
$\{T_2\}$	$p(T_2 \rightarrow r_2)$
$\{T_1, T_2\}$	$p(T_1, T_2 \rightarrow r_2)$
$\emptyset$	1

node T_1:

meas.	region	ψ_{T_1}
y_{11}	r_1	$p(y_{11}; T_1, r_1)$
y_{12}	r_1	$p(y_{12}; T_1, r_1)$
y_{11}, y_{21}	r_2	$p(y_{11}, y_{21}; T_1, r_2)$
y_{11}, y_{22}	r_2	$p(y_{11}, y_{22}; T_1, r_2)$
y_{12}, y_{21}	r_2	$p(y_{12}, y_{21}; T_1, r_2)$
y_{12}, y_{22}	r_2	$p(y_{12}, y_{22}; T_1, r_2)$

and the edge compatibility functions $\psi_{s_1 r_2}$ and $\psi_{r_2 T_1}$ are defined as

$$\psi_{s_1 r_2} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \psi_{r_2 T_1} = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Other node and edge compatibility functions can be defined in a similar way. It can be verified that $p(\mathcal{Y} \mid \mathbf{x}, \tilde{\mathbf{x}})$ equals to the product of the node and edge compatibility functions defined above, and thus satisfies Eq. (1). The all–zero columns of $\psi_{s_1 r_2}$ indicate that T_2 has to be in r_2 (so that it can be detected by s_1) since s_1 generates two observations and we assume no missed detections or false alarms exist. In such a scenario, the simple fact that a sensor does see something or see nothing can provide valuable information to substantially reduce the target uncertainty.

B. N–Scan Data Association

In some cases, the ambiguity of data association may not be easily resolved by using the data in the current scan only, and future scans might provide useful information (for example, when two target tracks are crossing each other at the current scan). This requires associating the measurements of the current scan by considering the measurements from the current scan as well as several future scans. Our graphical model–based data association approach can be extended to such an N–scan data association scenario. We discuss such an extension only on the region–based modeling approach as it applies more generally.

To take advantage of the reports in the next N scans, our model contains a copy of the sensor nodes and subregion nodes for each of the $N + 1$ scans in the temporal window, with the nodes affiliated with association variables $\{\mathbf{x}_i^{(t)}\}$ and $\{\tilde{\mathbf{x}}_j^{(t)}\}$, $t \in \{0, 1, \ldots, N\}$, respectively. However, the model maintains only one copy of the target nodes for each target across all scans. Each target node is connected with a subregion in a certain scan if the target might enter the subregion in that scan. The states of each target node correspond to the branches of
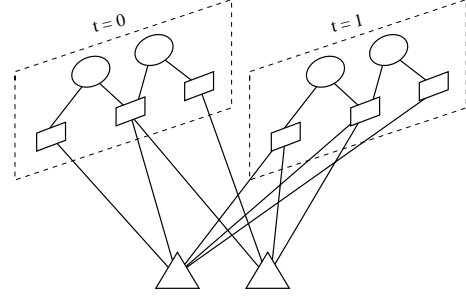

Fig. 5. A graphical model for N–scan ($N = 1$) association.

its hypothesis tree as obtained by an MHT algorithm, and the measurement likelihood evaluated on each branch is used as the target node compatibility function. A model for N–scan data association with $N = 1$ is shown in Fig. 5. After the max–product algorithm is applied on such a graphical model, the MAP estimates $\{\hat{\mathbf{x}}_1^{(0)}, \hat{\mathbf{x}}_2^{(0)}, \ldots, \hat{\mathbf{x}}_M^{(0)}\}$ indicate the association decisions at the current scan $t = 0$.

IV. COMMUNICATION–SENSITIVE MESSAGE–PASSING

Although the message-passing algorithm we have introduced is inherently distributed and hence appealing for sensor network applications, it may still require a large amount of communication and pose a significant challenge to the sensors. In particular the parallel–message passing operation requires each graph node to send a message to each of its neighbors at every iteration, which corresponds to a certain amount of communication in the sensor network and results in a certain amount of power consumption. It is critical to reduce the amount of communication that the message–passing algorithm requires to make it broadly applicable for SN. However, it is not straight–forward how one should develop a mechanism to achieve such communication savings without severe adverse effects on the inference quality.

We propose an adaptive approach that, while reducing the amount of communication, does *not* lead to serious degradation in performance. In this approach, after a new message is formed at a node, the node has the authority to make a decision about whether it needs to transmit this message or not. A message will be sent only when it contains "significant" new information compared to the message sent by the same node on the same edge in the previous iteration; otherwise the message will not be sent. If the message in the current iteration is not sent, the destination node uses the corresponding message from the previous iteration instead. Such a communication–sensitive message–passing (CSMSG) algorithm requires each node t to compute $d(M_{ts}^k, M_{ts}^{k-1})$ according to a certain distance measure $d(\cdot, \cdot)$ at each iteration k, and to compare it with a message tolerance ϵ. If $d(M_{ts}^k, M_{ts}^{k-1}) < \epsilon$, message M_{ts}^k will not be sent, and node s will use M_{ts}^{k-1} that it already received in the previous iteration to do its own local computation. We use the Kullback-Leibler (KL) divergence [13] to measure the distance between the information content of two messages. The KL divergence is widely used to

measure the similarity of two probability distributions in the information theory literature. In the context of this paper, it is defined as

$$d(M_{ts}^k, M_{ts}^{k-1}) = \sum_{\mathbf{x}_s} M_{ts}^k(\mathbf{x}_s) \log \frac{M_{ts}^k(\mathbf{x}_s)}{M_{ts}^{k-1}(\mathbf{x}_s)}. \qquad (6)$$

With CSMSG, a trade-off arises between the performance the algorithm can achieve and the amount of communication it requires. By using a proper message tolerance ϵ, we can tune the algorithm to achieve a suboptimal solution according to the budget for the communication cost. With smaller ϵ, the algorithm obtains a more accurate approximation to the value computed by standard message-passing, at the expense of more messages exceeding the message tolerance and being transmitted. Yet even with very small ϵ such that the loss of performance is trivial, the communication saving compared with the standard message-passing might be still significant. In Section V, we show the tremendous communication saving of CSMSG compared with standard message-passing. Note that CSMSG can be used both in the zero–scan and the N–scan settings.

The overhead of implementing CSMSG instead of standard message-passing is insignificant considering the potential savings in communication. In CSMSG, every sensor node requires some additional memory for storing messages from the previous iteration, so that it can compare the new and old messages that it generated, and use the old messages that it received when necessary. In addition, we also require a mechanism for letting the sensor know when a new message has not been sent, so that it knows it should use the old message instead. One possibility is to pass one extra bit of information on every link in each iteration to indicate if the nodes have new information or not. Alternatively, we could synchronize the communication and design the protocol in a way such that the sensor will use the old message after some latency period, whether the new message was not sent or simply lost on the way.

V. EXPERIMENTAL RESULTS

A. Data Association for MTT

To test our graphical model–based data association approach, we simulate tracking 20 targets with a 25–sensor network as shown in Fig 1. Each sensor measures the bearing of the targets corrupted by independent Gaussian noise with zero mean and $5°$ standard deviation. The detection rate of each sensor is set to 0.8 and we assume no false alarms. The average surveillance area of each sensor normalized by the area of the overall network surveillance field is set to 0.08. Fig. 6 shows the average association error rate of 50 Monte-Carlo runs at various levels of prior target location uncertainty (normalized by sensor surveillance range), based on the results of the max–product algorithm. The results show that the smallest association error rate is achieved in the complete–organization scenario, because the subregion for each target is known. In a partially organized network, the benefit of using one–scan versus zero–scan data association
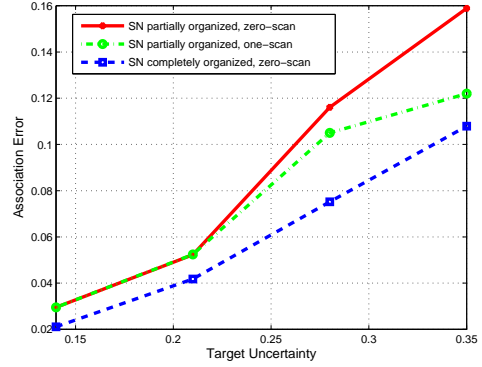

Fig. 6. Data association results in a 25–sensor network. The level of target uncertainty is defined as the standard deviation of the target prior distribution normalized by the radius of the sensor coverage region. Association error is defined as the percentage of incorrectly associated measurements.
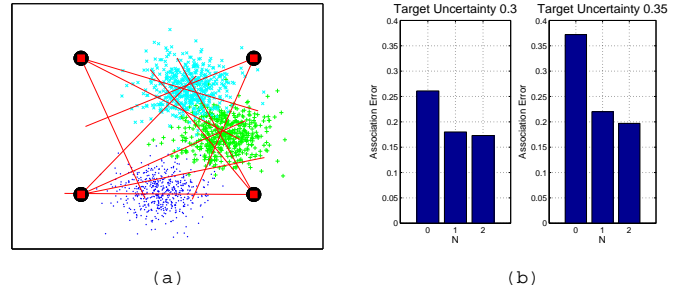

Fig. 7. (a) A snapshot of a target–dense scenario in a four-sensor network. (b) N–scan association error rate.

increases with the target uncertainty level. This reflects the increasing necessity of future data reports as the association problem becomes more challenging. Overall, this experiment demonstrates the applicability of our efficient data association algorithm on a reasonably–sized problem, where the three scenarios we consider in Fig. 6 exhibit intuitive behavior.

To highlight the benefits of using multi–scan data, we also present the performance of N–scan association in an illustrative target–dense scenario. Fig. 7(a) shows a four-sensor network with three targets intentionally placed in the middle area such that all the targets are covered by all four sensors. This is a contentious scenario by design, and it is difficult to make correct association decisions based on the bearing measurements in one scan only. However, the targets move in random directions at random speed, and data from future scans can help to resolve part of the association ambiguity. Fig. 7(b) shows the average association error rate of 50 Monte–Carlo runs, with $N = 0, 1$, and 2 respectively, where one–scan association has much lower error rate than the zero–scan as expected. Two-scan association has even lower error rate than one–scan, although the improvement is not that dramatic as most of the resolvable ambiguity has been solved by one–scan data association.

| Alg. | TRMP | max–prod. | CSMSG | |
			$\epsilon = 0.1$	$\epsilon = 1$
Error	25.84%	25.89%	27.88%	28.95%
Comm.	1029.3	55.25	10.41	8.79

TABLE I

COMPARISON OF TRMP, MAX-PRODUCT, AND CSMSG

Fig. 8. Performance–communication trade-off of CSMSG. Each error bar shows two times of std. deviation.

B. Communication–Sensitive Message–Passing

The performance–communication trade-off is investigated by applying the CSMSG algorithm on data association in the same SN as shown in Fig 1, but this time 33 targets are present. For brevity, we present CSMSG results only on zero–scan data association in the complete–organization scenario. Similar experiments can be conducted for N–scan data association or for partially organized networks. Table I compares the performance achieved and the amount of communication needed by TRMP, the max–product, and CSMSG algorithms. The amount of communication is evaluated as the average number of messages sent by each sensor node. The error rate generated by the TRMP algorithm is optimal in the MAP sense. The results show that the CSMSG algorithm with reasonable message thresholds has only slightly higher error rates than TRMP and the max–product algorithm. However, the communication cost associated with the CSMSG algorithm is significantly less than the other two algorithms. Therefore, when communication is costly, CSMSG is a preferable algorithm in that it can achieve a near–optimal performance with far less communication (with the appropriate choice of tolerance parameter). The trade-off curves for CSMSG at various message tolerances are shown in Fig. 8, where we observe an interesting threshold around $\epsilon = 1.7$. With smaller message tolerances than this threshold, CSMSG can achieve a similar error rate to max–product but using much less communication. However, when the message tolerance exceeds this threshold, the error rate increases sharply as some messages that are crucial to obtain a reliable estimation are ignored. The existence of such a threshold also suggests that the message tolerance corresponding to it might be an ideal parameter when we want to pursue the best performance-communication cost ratio. However, how to identify this message tolerance in advance remains an open question.

The information flow dynamics in the network is revealed by displaying the message transmission in each iteration of CSMSG, as shown in Fig. 9. In the first iteration, every node sends messages to its neighbors. As the iteration goes on, fewer and fewer nodes need to transmit messages. Finally only one node sends a message in the last iteration. This observation suggests that some sensors can be shut off earlier for saving power, for example, the five nodes at the upper left corner. There are also other nodes which can be shut off temporarily but need to start communication again later. For example, the node s_{12} stops sending messages after the second iteration, but as new information keeps coming in, it resumes sending messages in iteratio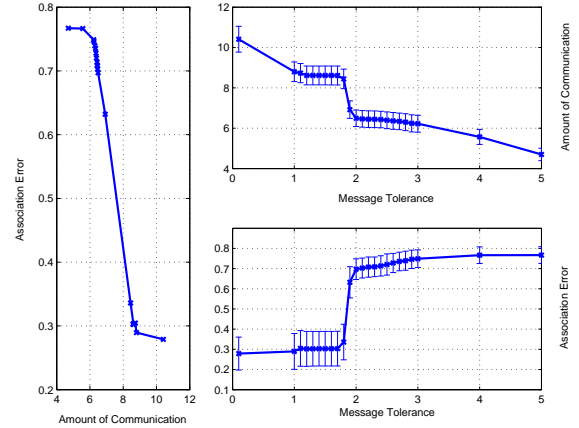n five, probably because it finds its previous estimate is inaccurate or the incoming information is important to its neighbors.

VI. CONCLUSION

In this paper, we introduced techniques using the framework of graphical models to solve data association problems arising in distributed sensing scenarios. We proposed several different approaches to modeling, in which nodes in the underlying graphical model were associated with different quantities (such as sensors, subregions, and targets) in the sensor network. The proposed graphical model–based approach well captures the sparse structure inherent in the SN, and scales well with the number of sensors in the network, thereby rendering optimal data association feasible in applications involving large–scale SN. We also proposed a communication–sensitive message–passing algorithm, and found that it is capable of achieving near–optimal performance with substantial savings in communication. This is very attractive when communication and power are limited resources for sensors. Moreover, we found that applying CSMSG on the distributed data association problem yielded insights into the dynamics of the message–passing during information fusion. Experimental results based on simulated data show the effectiveness of our approach.

There are number of research directions that remain to be explored. First, a model thinning or hypothesis pruning technique to reduce the model size is of interest for N–scan association in large–scale networks. We are currently exploring a complexity reduction method by introducing hypothesis sampling into our graphical model-based approach. Second, the graphical model structure where message–passing algorithms are applied is not the same as the communication–layer structure of SN. Consequently, a protocol to implement message–passing in real SN architectures is an interesting topic for further research. Third, it is of interest to provide more theoretical analysis of the CSMSG algorithm, on which some preliminary work already exists [14]. An interesting open problem is how to identify in advance the performance–communication trade–off (i.e., the message tolerance). In addition, the CSMSG
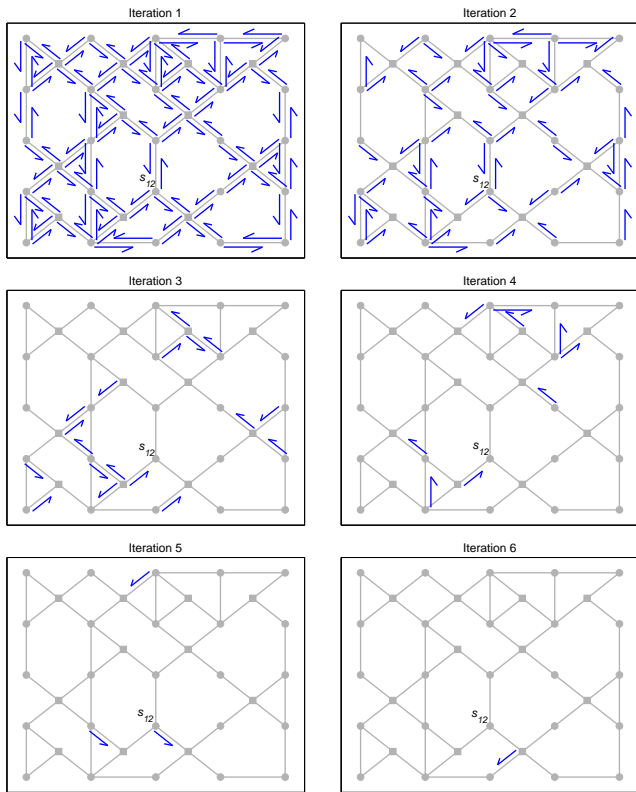
Fig. 9. Information flow dynamics revealed by CSMSG ($\epsilon = 0.1$). An arrow indicates a message being sent.

algorithm is only a simple way to address the communication challenge for SN applications; more advanced algorithms for doing distributed inference under communication constraints will be of interest.

REFERENCES

[1] C. Y. Chong and S. Kumar, "Sensor networks: evolution, opportunities, and challenges," *Proc. IEEE*, vol. 91, pp. 1247–1256, August, 2003.

[2] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*. Orlando: Academic Press, 1988.

[3] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. on Automatic Control*, vol. AC-24, pp. 843–854, 1979.

[4] S. S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Norwood, MA: Artech House, 1999.

[5] T. Kurien, "Issues in the design of practical multitarget tracking algorithms," in *Multitarget-Multisensor Tracking: Advanced Applications*, Y. Bar-Shalom, Ed. Norwood, MA: Artech House, 1990, vol. 1, pp. 43–83.

[6] C. Y. Chong, S. Mori, and K. C. Chang, "Distributed multitarget multisensor tracking," in *Multitarget-Multisensor Tracking: Advanced Applications*, Y. Bar-Shalom, Ed. Norwood, MA: Artech House, 1990, vol. 1, pp. 247–295.

[7] M. I. Jordan, Ed., *Learning in Graphical Models*. Cambridge MA: MIT Press, 1999.

[8] Y. Weiss and W. T. Freeman, "On the optimality of solutions of the max–product belief–propagation algorithm in arbitrary graphs," *IEEE Trans. on Information Theory*, vol. 47, pp. 736–744, Feb. 2001.

[9] R. G. Cowell, A. P. Dawid, S. L. Lauritzen, and D. J. Spiegelhalter, *Probabilistic Networks and Expert Systems*. Springer-Verlag, 1999.

[10] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky, "MAP estimation via agreement on (hyper) trees: message-passing and linear programming approaches," in *Proc. Allerton Conf. on Comm., Control and Computing*, Monticello, IL, Oct. 2002.

[11] A. T. Ihler, J. W. Fisher, R. L. Moses, and A. S. Willsky, "Nonparametric belief propagation for self–calibration in sensor networks," in *Third International Symposium on Information Processing in Sensor Networks*, Berkeley, CA, 2004, pp. 225–233.

[12] L. Chen, M. Wainwright, M. Çetin, and A. S. Willsky, "Multitarget-multisensor data association using the tree-reweighted max-product algorithm," in *SPIE AeroSense Symposium, Signal Processing, Sensor Fusion, and Target Recognition XII*, vol. 5096, Orlando, FL, 2003, pp. 127–138.

[13] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: John Wiley, 1991.

[14] A. T. Ihler, J. W. Fisher, and A. S. Willsky, "Message errors in belief propagation," in *Neural Inforamtion Processing Systems (NIPS)*, 2004.