Information Fusion 12 (2011) 194-201

Contents lists available at ScienceDirect

**Information Fusion** 

journal homepage: www.elsevier.com/locate/inffus

# Localized sensor management for multi-target tracking in wireless sensor networks

## Qing Ling<sup>a,\*</sup>, Yinfei Fu<sup>b,1</sup>, Zhi Tian<sup>b,1</sup>

<sup>a</sup> Department of Automation, University of Science and Technology of China, Hefei, Anhui 230027, China <sup>b</sup> Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI 49931, USA

## A R T I C L E I N F O

## ABSTRACT

Article history: Received 21 February 2009 Received in revised form 22 July 2010 Accepted 17 January 2011 Available online 26 January 2011

Keywords: Wireless sensor network Sensor management Multi-target tracking Multi-sensor data fusion distinct challenges to the task of tracking mobile targets, which is traditionally treated primarily from the data fusion perspective. This paper investigates the impact of sensor management on data fusion in a resource-limited network. A localized multi-sensor multi-target tracking framework is presented, consisting of four intertwined modules: data acquisition, data fusion, information propagation, and sensor management. The sensor management module, which boils down to a constrained binary optimization problem, is emphasized for efficient sensing resource allocation. Given limited bandwidth and power in the network, a localized greedy-selection sensor management (GSSM) algorithm is proposed to dynamically select a subset of sensors that contributes most effectively to the tracking accuracy. Using only localized information propagation among one-hop neighbors, the proposed framework obviates the need for a fusion center or multi-hop relays, and thus improves network robustness and scalability. © 2011 Elsevier B.V. All rights reserved.

For a multi-hop wireless sensor network, the limited sensing and communication resources give rise to

## 1. Introduction

Tracking of mobile targets is an important application of wireless sensor networks, e.g., in intrusion detection and habitat monitoring. The target tracking task in a traditional multi-sensor system primarily concerns the data fusion module, which processes the sensed data and provides the trajectories of targets. From a Bayesian perspective, the goal of data fusion is to recursively calculate belief of the state vector based on the measurements up to current observation time. The Kalman filter, extended Kalman filter and particle filter have been widely used herein [5,15,16]. An extensive survey of various filtering techniques can be found in [8].

In wireless sensor networks, it has become essential for a target tracking task to incorporate a sensor management module, in order to cope with the limited bandwidth and power [8,14,20]. Sensor management aims to save energy in sensing and communication by assigning a proper subset of sensors to track a target in the subsequent time slot, while guaranteeing the tracking quality.

For sensor-target assignment, centralized algorithms are discussed in [3,9,19,21], where a single target is tracked by several assigned sensors, and the sensor selection is based on the predicted tracking quality at the next time slot. Tracking quality over a future horizon is discussed in [1,6,11]. However, centralized target tracking and sensor management suffer from two main disadvantages. Firstly, they critically rely on a fusion center that is subject to failure. Secondly, when the network size is large, both communication overhead and computational load of the fusion center can be excessive [22]. For network robustness and scalability, localized decision making and information processing are expected [10,14]. Meanwhile, proper information propagation mechanisms need to be designed to ensure sensor collaboration at affordable communication costs.

This paper presents a localized multi-target tracking framework, and emphasizes the localized sensor management problem in order to efficiently allocate the limited network resources for sensing, computation and communication. The main contributions are:

- (1) The framework is developed for multi-sensor multi-target scenarios. A tracking task is vertically decomposed into four modules: data acquisition, data fusion, information propagation, and sensor management. In each module, the task is horizontally decomposed and assigned to sensors, other than to a fusion center. Decision making and information processing only require local data exchange, i.e., one-hop communication. This localized structure offers improved robustness and scalability over a centralized one.
- (2) A constrained binary optimization problem is formulated for optimal sensor-target assignment, which takes into account of the limited sensing and fusion capacities, as well as the





<sup>\*</sup> Corresponding author. Tel.: +86 0551 360 2482.

*E-mail addresses*: qingling@mail.ustc.edu.cn (Q. Ling), yfu@mtu.edu (Y. Fu), ztian@mtu.edu (Z. Tian).

<sup>&</sup>lt;sup>1</sup> Tel.: +1 906 487 2515.

target coverage requirement. A greedy-selection sensor management (GSSM) algorithm is proposed to efficiently solve the binary optimization problem in a localized manner.

(3) For localized implementation of the GSSM algorithm, two information propagation mechanisms, predicted subset mechanism and nearest subset mechanism, are discussed. The predicted subset mechanism entails multi-hop communication, and the resulting localized GSSM algorithm attains the same performance as its centralized counterpart under certain mild condition. The nearest subset mechanism requires one-hop local communication only, and the corresponding localized GSSM algorithm offers the same asymptotic performance as the centralized one, in the limit of time.

The rest of this paper is organized as follows. In Section 2, a literature survey is provided. Section 3 describes the basic assumptions and models. The localized multi-target tracking framework is discussed in Section 4. Section 5 formulates the sensor management problem and proposes the localized GSSM algorithm. Simulation results are provided in Section 6 to verify the effectiveness of the proposed framework. Section 7 summarizes the paper along with future work.

## 2. Related works

Basic structures of sensor management can be broadly categorized into centralized, hierarchical and localized, as discussed in [8,14,20]. A centralized sensor management structure is introduced in [8]. Sensors collect target measurements and transmit them to a fusion center. At each time slot, the fusion center processes the data, decides which sensor to track which target at the next time slot based on a global objective function, and feeds the decisions back to the sensors. A hierarchical sensor management structure is considered in [20]. Several local sensor managers, controlled by a fusion center, are responsible for managing several subsets of sensors. Given the distributed nature of wireless sensor networks, localized sensor management is most appropriate, but research on this structure is still lagging [14].

Centralized sensor management algorithms are discussed in [3,9,19,21]. Most of the work confines to the single-target case, in which the goal is to assign several sensors to track the target, and the sensor selection is based on the predicted tracking quality at the next time slot. Specifically, [3] considers the constraints of maximum allowable cost and maximum number of assigned sensors. An outer approximation algorithm is used to solve the formulated binary convex programming problem. Similarly, [9] limits the target to be tracked by a given number of sensors. Detection probability is emphasized in [19] as the objective function of sensor management. In [21], the sensors are mobile, thus the motion control parameters of sensors are also decision variables.

Tracking quality over a future time horizon is discussed in [1,6,11]. The centralized sensor management problem is modelled as a partially observable Markov decision process, and solved with an adaptive dynamic strategy in [1] and a Monte Carlo sampling approach in [6]. The hidden Markov model is considered in [11], where a stochastic dynamic programming algorithm is applied to select an optimal measurement scheduling, aiming at minimizing estimate errors and measurement costs.

There has been little work on localized sensor management. Limited to the single-target case, [10] discusses the localized algorithm as an extension of [9]. In the autonomous node selection algorithm, a candidate subset  $\mathscr{C}_j$  collaboratively selects two sensors, which optimize the overall predicted utility function. Then these two nodes communicate with their neighbors and elect  $N_{smax}$ 

sensors to track the target *j* at the next time slot. However, the proposed autonomous node selection algorithm is not directly applicable to the multi-target case, where sensors are facing the competition among multiple targets. Furthermore, connectivity of the elected  $N_{smax}$  sensors cannot be guaranteed. This fact indicates the significance of a proper information propagation mechanism, which is an important element in our proposed multi-target tracking framework.

This paper considers sensor management for the more complicated multi-target tracking case, under the assumption that sensors exactly know the tags of targets in tracking. When target tags are not available in tracking, sensors need to associate the measurements with different targets, which is known as the data association problem [13]. The multiple hypotheses tracking algorithm [2] and the joint probabilistic data association algorithm [17] enumerate all possibilities of the targets, known as hypotheses. The best hypothesis is accepted based on the associated probability. In the Monte Carlo Markov Chain data association algorithm, a hypothesis is randomly selected, and compared with random samples generated from a proposal function. The hypothesis is then accepted, discarded or modified according to the comparisons [18]. In this work we assume the knowledge of target tags in order to highlight the design issues in sensor management for multiple targets independent of the impact of data association. With the aid of the data association techniques, we can extend our current work to the multi-target tracking case without the assumption of target tags.

Another relevant line of work is information-driven sensor querying, which focuses on scalable information querying and data routing in a distributed network [4,22]. A user query is initially routed to sensor a, which performs initial estimate of the target position. Sensor *a* then selects the next sensor *b*, which is believed to offer the best tracking accuracy at the next time slot, and provides the current estimate to b. Periodically, the state estimate is sent back to the user using a shortest path routing algorithm. Information-driven sensor querying is essentially a cross-layer information acquisition framework, where sensor management amounts to the single-target handoff and does not concern the management of multiple targets. Similar idea appears in [12], where a tree structure is proposed for in-network information querying. Our proposed localized multi-target tracking framework and localized sensor management algorithm are also compatible to the information querying and data routing framework.

## 3. System models

In this section, we discuss the basic assumptions and models of the mobile target tracking task in wireless sensor networks. An information filter is introduced for multi-sensor data fusion.

## 3.1. Basic assumptions

We make the following assumptions on sensors and targets throughout this paper:

- (1) There are  $N_s$  sensors and  $N_t$  targets with tags in a twodimensional sensing field. Each sensor has the knowledge of its own location information.
- (2) Each sensor has an adjustable communication range  $r_c$ , with a maximum value  $r_{cmax}$ . Two sensors can communicate in one hop, if and only if they are within the communication range of each other.
- (3) Data association has been fulfilled. Namely, a sensor exactly knows the tag of a target in tracking.
- (4) The system time is slotted and the network is synchronized.

(5) The sensing capacity of each sensor is set as  $N_{tmax}$ , i.e., each sensor can track no more than  $N_{tmax}$  targets per time slot. Meanwhile, each target should be tracked by no more than  $N_{smax}$  sensors to save network resources, but must be tracked by at least one sensor to ensure target coverage.

## 3.2. Target model and sensor model

We model the motion of each target *j* as a linear discrete-time Markov process:

$$\mathbf{x}_j(t) = \mathbf{F}_j(t)\mathbf{x}_j(t-1) + \mathbf{w}_j(t).$$
(1)

Here  $\mathbf{x}_j(t)$  is the state vector of target j,  $\mathbf{F}_j(t)$  is the state transition matrix, and  $\mathbf{w}_j(t)$  is the process noise which is assumed to be zero mean Gaussian noise with covariance  $\mathbf{Q}_j(t)$ , i.e.,  $\mathbf{w}_j(t) \sim N(0, \mathbf{Q}_j(t))$ .

At time *t*, sensor *i* measures target *j* according to the following measurement equation:

$$\mathbf{z}_{ij}(t) = \mathbf{H}_{ij}(t)\mathbf{x}_{j}(t) + \mathbf{v}_{ij}(t).$$
(2)

Here  $\mathbf{H}_{ij}(t)$  is the observation model which maps the true state space onto the observed space, and  $\mathbf{v}_{ij}(t)$  is the observation noise which is assumed to be zero mean Gaussian noise with covariance  $\mathbf{R}_{ij}(t)$ , i.e.,  $\mathbf{v}_{ij}(t) \sim N(0, \mathbf{R}_{ij}(t))$ . We also assume that the initial state, and the noise vectors at each step are mutually independent.

Without loss of generality, we assume the observation covariance depends on the distance as  $\mathbf{R}_{ij}(t) = \mathbf{K}_{ij}(t) d_{ij}^2(t)$ , where  $\mathbf{K}_{ij}(t)$  is a distance-independent coefficient, and  $d_{ij}(t)$  is the distance from sensor *i* to target *j*.

## 3.3. Information filter

In this paper, we use an information filter [16], an equivalent form of the Kalman filter, for multi-sensor data fusion. Assume that at time *t*, sensor *i* tracks target *j* and estimates its state  $\mathbf{x}_j(t)$ . Our objective is to obtain the estimated state vector  $\hat{\mathbf{x}}_j(t|t)$  and the predicted state vector  $\hat{\mathbf{x}}_j(t|t) - \hat{\mathbf{x}}_j(t|t)$  and prediction error  $\mathbf{x}_j(t|t) - \hat{\mathbf{x}}_j(t|t)$  and prediction error  $\mathbf{x}_j(t+1|t) - \hat{\mathbf{x}}_j(t+1|t)$  are denoted as  $\mathbf{P}_j(t-t)$  and  $\mathbf{P}_j(t+1-t)$ , respectively. Based on the information filter structure, we define the information state vectors as  $\hat{\mathbf{y}}_j(t|t) = \mathbf{P}_j^{-1}(t|t)\hat{\mathbf{x}}_j(t|t)$  and  $\hat{\mathbf{y}}_j(t+1|t) = \mathbf{P}_j^{-1}(t+1|t)\hat{\mathbf{x}}_j(t+1|t)$ , and the information matrices as  $\mathbf{Y}_j(t|t) = \mathbf{P}_j^{-1}(t|t)$  and  $\mathbf{Y}_j(t+1|t)$ . The process of information filter includes the following two steps, i.e., estimation and prediction:

(1) *Estimation*. Estimate the information state vector  $\hat{\mathbf{y}}_j(t|t)$  from current measurement  $\mathbf{z}_{ij}(t)$ , and estimate the associated information matrix  $\mathbf{Y}_j(t-t)$ , as follows:

$$\hat{\mathbf{y}}_j(t|t) = \hat{\mathbf{y}}_j(t|t-1) + \mathbf{H}_{ij}^T(t)\mathbf{R}_{ij}^{-1}(t)\mathbf{z}_{ij}(t);$$
(3)

$$\mathbf{Y}_{j}(t|t) = \mathbf{Y}_{j}(t|t-1) + \mathbf{H}_{ij}^{\mathrm{T}}(t)\mathbf{R}_{ij}^{-1}(t)\mathbf{H}_{ij}(t).$$

$$(4)$$

(2) *Prediction.* Predict the information vector  $\hat{\mathbf{y}}_j(t+1|t)$  and predict the information matrix  $\mathbf{Y}_i(t+1-t)$ :

$$\hat{\mathbf{y}}_{j}(t+1|t) = \mathbf{Y}_{j}(t+1|t)\mathbf{F}_{j}(t+1)\mathbf{Y}_{j}^{-1}(t+1|t)\hat{\mathbf{y}}_{j}(t|t);$$
(5)

$$\mathbf{Y}_{j}(t+1|t) = \left(\mathbf{F}_{j}(t)\mathbf{Y}_{j}^{-1}(t|t)\mathbf{F}_{j}^{T}(t) + \mathbf{Q}_{j}(t)\right)^{-1}.$$
(6)

For the multi-sensor data fusion case, if a subset  $\mathcal{S}_j$  of sensors track the target *j*, then the estimation rules (3) and (4) are replaced by (7) and (8):

$$\hat{\mathbf{y}}_{j}(t|t) = \hat{\mathbf{y}}_{j}(t|t-1) + \sum_{i \in \mathscr{S}_{j}} \mathbf{H}_{ij}^{\mathrm{T}}(t) \mathbf{R}_{ij}^{-1}(t) \mathbf{z}_{ij}(t);$$
(7)

$$\mathbf{Y}_{j}(t|t) = \mathbf{Y}_{j}(t|t-1) + \sum_{i \in \mathscr{S}_{j}} \mathbf{H}_{ij}^{T}(t) \mathbf{R}_{ij}^{-1}(t) \mathbf{H}_{ij}(t).$$
(8)

One important property of the information filter is, when the model is accurate and the values for any  $\hat{\mathbf{x}}_j(0|0)$  and  $\mathbf{P}_j(0|0) = \mathbf{Y}_j^{-1}(0|0)$  accurately reflect the distribution of the initial state  $x_j(0)$ , the covariance matrix  $\mathbf{P}_j(t|t) = \mathbf{Y}_j^{-1}(t|t)$  can accurately reflect the covariance of estimation error  $\mathbf{e}_j(t|t) = \mathbf{x}_j(t) - \hat{\mathbf{x}}_j(t|t)$ , that is,

$$\mathbf{P}_{j}(t|t) = E\{\mathbf{e}_{j}(t|t)^{T}\mathbf{e}_{j}(t|t)\}.$$
(9)

## 4. Localized multi-target tracking

In this section, we discuss the challenges in designing localized multi-target tracking algorithms under constrained network resources, and propose a practical framework that effectively tackles these challenges.

## 4.1. Challenges in localized multi-target tracking

A localized multi-target tracking framework generally consists of the following modules:

- Data acquisition. At time t, for each target j, a previously selected subset of sensors, denoted as *S<sub>j</sub>*, measure the target positions.
- (2) Data fusion. At time t, for each target j, sensors in  $\mathcal{S}_j$  process their measurements and provide estimations and predictions of targets using (5)–(8).
- (3) *Information propagation.* At time *t*, for each target *j*, the predictions  $\hat{\mathbf{y}}_j(t+1|t)$  and  $\mathbf{Y}_j(t+1-t)$  are propagated to a subset of candidate sensors  $\mathscr{C}_j$ .
- (4) Sensor management. At time t, for each target j, an updated subset of sensors S<sub>j</sub> is selected from C<sub>j</sub> to track target j at time t + 1.

Assuming reliable multi-hop communication and data routing to a fusion center, centralized decision making and information processing have been discussed in wireless sensor network scenarios [3,9]. However, to achieve better scalability, higher robustness and lower latency, we prefer a localized multi-target tracking framework which features in autonomous in-network decision making and information processing. The framework is localized in the sense that only one-hop local communication among neighboring sensors is required, while the goal is to attain network-wide cooperation. The main challenges in the localized multi-target tracking framework are as follows:

- (1) Challenge in data fusion. As shown in (7) and (8), the information filter can be easily implemented in a sequential way. However, for each target j, sensors in  $S_j$  should be organized to exchange information one-by-one, using only one-hop local communication.
- (2) Challenge in information propagation. For each target j, the predictions should be propagated efficiently to the subset *C<sub>j</sub>*, i.e., via one-hop local communication.
- (3) Challenge in sensor management. For each target j, sensors in the candidate subset  $\mathscr{C}_j$  should be able to directly communicate with each other, and locally elect a subset of sensors  $\mathscr{S}_j$  to track target j at the next time slot.

## 4.2. Localized multi-target tracking framework

To tackle the aforementioned challenges, we propose a localized multi-target tracking framework as follows:

## Step 1:

Sensor management.

At time t - 1, sensors in  $\mathscr{C}_j$  locally decide  $\mathscr{S}_j$ , i.e., the subset of

sensors to track target j at time t. Sensors in  $\mathcal{S}_j$  coordinate to set a sequence for data fusion. One possible rule of setting the sequence is by sorting the tags of sensors.

Step 2:

Data acquisition.

At time *t*, for each target *j*, sensors in  $\mathcal{S}_j$  measure positions of the targets.

Step 3:

Data fusion.

At time *t*, for each target *j*, sensors in  $\mathcal{S}_j$  sequentially process the measurements using the information filter, according to the sequence decided in Step 1.

Step 4:

Information propagation.

At time *t*, for each target *j*, we propose two mechanisms to decide the current candidate sensor subset  $\mathscr{C}_{j}$ , i.e., the subset of sensors which receive the target information:

*Mechanism A: Nearest subset.* For each target *j*, the last sensor in the data fusion sequence broadcasts the predictions  $\hat{\mathbf{y}}_j(t+1|t)$  and  $\mathbf{Y}_j(t+1-t)$  to its neighbors within the communication range  $r_{cmax}/2$ . These neighbors form the candidate sensor subset  $\mathcal{C}_j$ .

*Mechanism B: Predicted subset.* For each target *j*, the last sensor in the data fusion sequence broadcasts the predictions  $\hat{\mathbf{y}}_j(t+1|t)$  and  $\mathbf{Y}_j(t+1-t)$  to all other sensors in the network. The subset  $\mathscr{C}_j$  is composed of the sensors whose distances to the predicted target position are smaller than or equal to  $r_{cmax}/2$ .

Step 5:

Iteration.

Go to Step 1 until the tracking mission ends.

This localized framework tackles the challenge in data fusion by introducing a sensor coordination mechanism in the sensor management module. In the information propagation module, the nearest subset mechanism is a localized operation. The predicted subset mechanism requires multi-hop communication to flood the network with predictions; nevertheless, it is discussed here to aid performance analysis.

With both of the information propagation mechanisms, any two sensors in the candidate sensor subset  $\mathscr{C}_j$  have a distance no larger than  $r_{cmax}$ . Therefore, sensors in the candidate subset can locally decide the tracking subset  $\mathscr{S}_j$  in the sensor management step. However, in a distributed network setting, the sensor management problem essentially aims at a global objective under global constraints. Hence a non-trivial problem arises: is it possible to decompose the objective and constraints for autonomous and localized decision making? We will answer this question in the subsequent section.

## 5. Localized sensor management

In this section, sensor management is formulated as a constrained binary optimization problem, and a localized greedyselection sensor management (GSSM) algorithm is proposed.

## 5.1. Sensor management problem

In the localized multi-target tracking framework, the remaining issue is the sensor management problem, which aims at preserving the limited sensing resources and prolonging the network lifetime. Intuitively, it is energy-efficient to select only a subset of sensors to track one target, since the measurements collected from some distant sensors do not contribute much to the tracking accuracy. To formulate this sensor selection problem, we construct the decision variables as a binary matrix  $\mathbf{a} = \{a_{ij}\}$ , where  $a_{ij} = 1$  denotes that sensor *i* tracks target *j*, while  $a_{ij} = 0$  denotes that sensor *i* does not track target *j*. Hence we relate  $\mathcal{S}_j$  to  $\mathbf{a}$  with  $\mathcal{S}_j = \{i | a_{ij} = 1\}$ .

From the global network perspective, the goal of sensor management is to minimize the future estimation error. At each time *t*, the collecting tracking error is  $\sum_{j=1}^{N_t} tr\{E\{\mathbf{e}_j(t+1|t+1)\}^T\mathbf{e}_j(t+1|t+1)\}\}$ . According to (9), we have  $E\{\mathbf{e}_j(t+1|t+1)^T\mathbf{e}_j(t+1|t+1)\} = \mathbf{P}_j(t+1|t+1) = \mathbf{Y}_j^{-1}(t+1|t+1)$ . Thus, substituting (8) into (9), the inverse of the tracking error boils down to

$$\sum_{j=1}^{N_t} tr \left\{ \mathbf{P}_j^{-1}(t+1|t) + \sum_{i \in \mathscr{S}_j} \mathbf{H}_{ij}^{T}(t+1) \mathbf{R}_{ij}^{-1}(t+1) \mathbf{H}_{ij}(t+1) \right\}$$

Clearly,  $\mathbf{P}_{j}^{-1}(t+1|t)$  is the prediction term common to all sensors, while  $\mathbf{H}_{ij}^{T}(t+1)\mathbf{R}_{ij}^{-1}(t+1)\mathbf{H}_{ij}(t+1)$  quantifies the contribution of sensor *i* to the tracking error of target *j* at time *t* + 1.

From the analysis above, we omit the common prediction terms and set the global objective function as maximizing the *information contribution* [7,8], as follows:

$$\max_{\mathbf{a}} \quad J(\mathbf{a}) = \sum_{j=1}^{N_t} \sum_{i \in \mathscr{S}_j} tr \Big\{ \mathbf{H}_{ij}^T(t+1) \mathbf{R}_{ij}^{-1}(t+1) \mathbf{H}_{ij}(t+1) \Big\}.$$
(10)

Under this objective function, contributions of all sensor-target pairs are decoupled, and the profit of each sensor-target pair (i, j) is the information contribution  $tr\{\mathbf{H}_{ij}^{T}(t+1)\mathbf{R}_{ij}^{-1}(t+1)\mathbf{H}_{ij}(t+1)\}$ . At each time  $t, \mathbf{H}_{ij}^{T}(t+1)$  is known, while  $\mathbf{R}_{ij}^{-1}(t+1) =$ 

At each time t,  $\mathbf{H}_{ij}^{-1}(t+1)$  is known, while  $\mathbf{R}_{ij}^{-1}(t+1) = \mathbf{K}_{ij}d_{ij}^{2}(t+1)$  is related to the distance from sensor i to target j. Here we use the predicted state vector  $\hat{\mathbf{x}}_{j}(t+1|t)$  to approximate the real state vector  $\mathbf{x}_{j}(t+1)$ , which yields the predicted position of target j and correspondingly  $\mathbf{R}_{ij}(t+1)$ . Both  $H_{ij}(t+1)$  and  $R_{ij}(t+1)$  are known locally to sensor i.

The sensor selection problem is subject to a set of constraints due to limited network resources. The first constraint is the sensing capacity of sensors:

$$\sum_{j=1}^{N_t} a_{ij} \leqslant N_{tmax}, \forall i.$$
(11)

Because the sensing task is time- and energy-consuming, the sensing capacity constraint is necessary for resource conservation. The second set of constraints describes the lower and upper bounds for the number of sensors tracking each target:

$$\sum_{i=1}^{N_{s}} a_{ij} \leqslant N_{smax}, \qquad \forall j; \tag{12}$$

$$\sum_{i=1}^{N_s} a_{ij} \ge 1, \qquad \forall j. \tag{13}$$

When too many sensors track one target, both communication and computation overhead will be too high, whereas the tracking quality only enhances slightly when distant sensors join the tracking task. On the other hand, one target should be tracked by at least one sensor to prevent loss of tracking.

#### 5.2. Greedy-selection sensor management

Let us consider the sensor management problem (10)-(13), which is a constrained binary optimization problem, and can be solved by the branch-and-bound method. However, it is difficult to implement the branch-and-bound method in a localized manner as it requires global information exchange.

In this section, we propose a greedy-selection sensor management (GSSM) algorithm, which reaches a near-optimal solution to the constrained binary optimization problem. Specifically, we firstly introduce a centralized GSSM algorithm, which requires global information exchange too. The centralized GSSM algorithm is then implemented locally. Interestingly, the localized GSSM algorithm achieves the same performance as the centralized GSSM algorithm under certain mild conditions.

The centralized GSSM algorithm adopts an iterative assignment strategy:

- *Step 1: Initialization.* Construct a  $N_t \times N_s$  profit matrix for sensortarget pairs. The profit for a sensor-target pair (i, j) is defined as  $tr\{\mathbf{H}_{ii}^{T}(t+1)\mathbf{R}_{ii}^{-1}(t+1)\}$  $H_{ii}(t+1)$ .
- Step 2: Assignment. Find the best sensor-target pair that corresponds to the highest profit, set the corresponding element in *a* as 1, and set the corresponding profit as minus infinite. Repeat this process for  $\min(N_t, N_s)$  times, but guarantee that one sensor or one target is selected at most for one time.
  - Step 3: Iteration. Go to Step 2 and repeat for  $min(N_{tmax}, N_{smax})$ times.

It should be noted that, when two sensor-target pairs have the same profits, we may simply choose the pair with smaller sensor tag or smaller target tag to avoid ambiguity. It is not difficult to show that the centralized GSSM algorithm, though sub-optimal, satisfies the resource constraints (11)-(13).

In Step 2 the individual assignment process is repeated for  $\min(N_t, N_s)$  times, since we need to guarantee that one sensor or one target is selected for no more than one time in each round. In Step 3 we have  $min(N_{tmax}, N_{smax})$  iterations, such that one sensor or one target is selected for at most  $min(N_{tmax}, N_{smax})$ . Therefore constraints Eqs. (11) and (12) can be satisfied. Furthermore, if  $N_s \ge N_t$ , Eq. (13) also hold according to Step 2. Apparently, the centralized GSSM algorithm requires global information exchange for the tracking profits. For the purpose of localized implementation, we limit information exchange within each candidate sensor subset  $\mathscr{C}_i$ . As discussed in the previous section, each sensor exactly knows which candidate subsets it belongs to after the information propagation step. Furthermore, sensors in one candidate subset can communicate with each other within one-hop. Therefore we have a localized GSSM algorithm, represented as a localized iterative greedy selection process, which contains  $min(N_{tmax}, N_{smax})$ rounds:

Step 1: Initialization. Each sensor i calculates its profit to track target j, if  $i \in \mathscr{C}_j$ . The profit is represented as  $tr\{\mathbf{H}_{ij}^{T}(t+1)\mathbf{R}_{ij}^{-1}(t+1)\mathbf{H}_{ij}(t+1)\}.$ 

Step 2: Assignment. This step is similar to a cooperative game, and contains at most N<sub>t</sub> rounds:v>

- (2a) Each sensor *i* selects the best target, for example, j, to track, and broadcasts its sensor tag, target tag and profit to all other sensors, with communication range  $r_{cmax}$ . Upon receiving a packet, sensor *i* checks whether it is in the candidate subset of the corresponding target. If it is true, sensor *i* accepts the packet; otherwise drops the packet.
- (2b) Sensor *i* compares its current profit with the accepted profits of other sensors. If the profit of *i* is the best one among the profits, then *i* tracks *j* in this round, and does not attend the successive rounds in this game. Otherwise *i* finds the best sensor-target pair, and remembers not to track the corresponding target in the successive rounds in this game.

- (2c) Each sensor *i* repeats Steps (2a) and (2b), until it tracks one target in this game, or all targets which look *i* as the candidate sensor, namely  $\{j | i \in \mathscr{C}_i\}$ , have been tracked by other sensors in this game.
- Step 3: Iteration. Each sensor saves its current assignment, and set the corresponding profit as minus infinite. Then go to Step 2 and repeat for  $\min(N_{smax}, N_{tmax})$  times. The localized GSSM algorithm features in the localized decision making and information exchange. Furthermore, the communication load of a single sensor is at most  $N_t \times \min(N_{s-1})$  $_{max}$ ,  $N_{tmax}$ ), thus the network is energy-efficient from the communication perspective.

## 5.3. Discussions

Here we analyze the impact of information propagation on the localized algorithm. Specifically, we prove that, if the information propagation adopts the predicted subset mechanism, the localized GSSM algorithm achieves the same performance as the centralized version. Then we discuss the case in which the information propagation adopts the nearest subset mechanism. Intuitively the localized GSSM algorithm asymptotically reaches the performance of the centralized algorithm in this case.

Property 1. With the predicted subset mechanism, the localized GSSM algorithm reaches the same solution as the centralized version, if any circle of radius  $r_{cmax}/2$  in the sensing area contains at least  $N_t + min(N_{smax}, N_{tmax}) - 1$  sensors.

**Proof.** We firstly prove that for each target *i*, the assigned  $\min(N_{smax}, N_{tmax})$  sensors are among the  $N_t + \min(N_{smax}, N_{tmax}) - 1$ sensors nearest to the predicted position of *j* in the centralized GSSM algorithm. The nearest  $N_t + \min(N_{smax}, N_{tmax}) - 1$  sensors means the best  $N_t + \min(N_{smax}, N_{tmax}) - 1$  profits in tracking target *j*. After the first round of the centralized GSSM algorithm, the selected sensor is at least among the nearest  $N_t$  sensors. Thus after  $\min(N_{smax}, N_{tmax})$  rounds, the selected sensors are among the nearest  $N_t$  + min( $N_{smax}$ ,  $N_{tmax}$ ) – 1 sensors to the predicted position of *j*.

If any circle of radius  $r_{cmax}/2$  in the sensing area contains at least  $N_t + \min(N_{smax}, N_{tmax}) - 1$  sensors, then for any target *j*, the nearest  $N_t + \min(N_{smax}, N_{tmax}) - 1$  sensors are included in  $\mathscr{C}_i$  under the predicted subset mechanism.

Now both the localized and centralized GSSM algorithms select sensor-target pairs in the same sets. Since the two algorithms follow the same greedy selection rule, they both reach the same solution. 

In the information propagation step, the predicted subset mechanism requires multi-hop data transmission to the whole network in order to select a candidate subset of sensors local to the predicted target position, and thus results in extra communication overload and information delay. Therefore, although data acquisition, data fusion and sensor management are done locally, information propagation is actually not a localized operation. However, we have discussed that the localized framework based on this mechanism achieves the same performance as the centralized method under certain mild condition.

Alternatively, the nearest subset mechanism requires only onehop broadcast to neighboring sensors, therefore the communication load and information delay are minimum. The disadvantage of this mechanism is the loss of optimality. Namely, when the optimal sensor selection  $\mathcal{G}_i$  is not in the candidate subset  $\mathcal{G}_i$ , system performance will be degraded. But intuitively, each local decision



Fig. 1. The true trajectory (a) and the estimated trajectory (b) of the target.



Fig. 2. The covariances of position error (a) and velocity error (b) of the target.



**Fig. 3.** Ratio of the objective function value of the localized GSSM solution to that of the optimal solution.

tends to select the  $\mathcal{S}_j$  near to the predicted position of j, thus the final sensor selection will converge to the solution of the predicted subset mechanism, as shown in the following simulation results.

## 6. Simulation results

In this section, we provide extensive simulation results to illustrate the effectiveness of the localized multi-target tracking framework and the embedded localized sensor management algorithm. We discuss a single-target tracking case with a small network and a multi-target tracking case with a large network.

## 6.1. General settings

We adopt the same target state model and sensor measurement model throughout the simulations. The initial positions and velocities of the targets are randomly generated in a two-dimensional area. The total simulation time is set as 40, and the length of one time slot as  $\Delta t$  = 1. For each target *j*, its state vector  $\mathbf{x}_j(t)$  contains the positions of *x* and *y* axes, and velocities of *x* and *y* axes. Parameters pertinent to the target state model in (1) are:

$$\mathbf{F}_{j}(t) = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{Q}_{j}(t) = \begin{pmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 0.04 & 0 \\ 0 & 0 & 0 & 0.04 \end{pmatrix}.$$

Measurement of sensor *i* on target *j* is the position of *j*, as modelled by:

$$\mathbf{H}_{ij}(t) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

The measurement noise covariance is  $\mathbf{R}_{ij}(t) = \mathbf{K}_{ij}d_{ij}^2(t)$ , where  $d_{ij}(t)$  is the distance from sensor *i* to target *j*, and

$$\mathbf{K}_{ij} = egin{pmatrix} 1 imes 10^{-6} & 0 \ 0 & 1 imes 10^{-6} \end{pmatrix}.$$

Note that when utilizing the objective function (10) for sensor management, we set the same weights to the position error and the velocity error. It is straightforward to scale the trace operator tr with weighting factors, which allows to treat the position error and the velocity error discriminately.

## 6.2. Small network for single-target tracking

We firstly consider a simple multi-sensor single-target case in a small grid network. Let  $N_s = 4$ ,  $N_t = 1$ ,  $N_{smax} = 2$ , and  $N_{tmax} = 1$ . Sensors are deployed in a two-dimensional area at positions (0, 0),

(0, 100), (100, 0), (100, 100). The maximum communication range of sensors are set to be  $r_{cmax} = \infty$ , i.e., sensors can communicate with each other directly in this small network. Furthermore, the predicted subset mechanism and the nearest subset mechanism both select all sensors into the candidate subset. Therefore, the sensor management solutions are the same under these two mechanisms.

The true trajectory and estimated trajectory of the target are illustrated in Fig. 1. At t = 0, sensors have no prior knowledge about the initial position and velocity of the target, and hence initialize the estimated position as (0, 0). When more information is



Fig. 4. Assignment of sensors 1-4 through the simulation.

acquired, the estimate becomes more accurate. The estimated trajectory coincides with the true trajectory eventually.

The position error and velocity error are shown in Fig. 2. When t = 0, the tracking error is large without any prior knowledge about the initial position and velocity of the target. As the simulation proceeds, the tracking error reaches a steady level. Note that the sharp decrease of the position error in the beginning is caused by overfitting. In summary, Figs. 1 and 2 prove the effectiveness of the localized target tracking framework.

Next we consider the performance bound of the localized GSSM algorithm. The performance ratio, defined as the ratio of the objective function value of the localized GSSM solution to that of the optimal solution, is shown in Fig. 3. The localized GSSM algorithm, though sub-optimal, still achieves acceptable performance.

Fig. 4 shows the sensor assignment through the simulation, with 1 denotes tracking and 0 denotes no tracking. At the first five time slot, the predicted target position is near to sensor 1, thus sensor 1 is assigned to track the target. Gradually the target moves to near sensor 3 or sensor 4, and the network assigns sensor 3 or sensor 4 to track it accordingly. Here the proposed sensor management algorithm reduces to the single-target handoff in [4,22] for this multi-sensor single-target tracking case.

It should be noted that we permit  $N_{smax}$  = 2 sensors to track the target, while the localized GSSM algorithm only assigns 1 sensor at each time. The reason is that the localized GSSM algorithm allows only min( $N_{smax}$ ,  $N_{tmax}$ ) sensors to track the target.

## 6.3. Large network for multi-target tracking

We now consider a multi-sensor multi-target case in a large random network, with  $N_s = 500$ ,  $N_t = 4$ ,  $N_{smax} = 4$  and  $N_{tmax} = 2$ . Sensors have a common maximum communication range  $r_{cmax} = 200$ ,



Fig. 5. The true trajectories (a) and the estimated trajectories (b) of the targets.



Fig. 6. The covariances of position errors (a) and velocity errors (b) of the targets.



**Fig. 7.** Ratio of the objective function value of the localized GSSM solution to that of the optimal solution, under both the predicted subset mechanism and the nearest subset mechanism.

and are randomly deployed in a  $1000 \times 1000$  area. In this case, centralized decision making and information processing are impractical for both vast communication overhead and unaffordable computation load. On the contrary, the proposed localized multi-target tracking framework and the embedded localized sensor management algorithm are scalable for this large network.

In the simulation we adopt the nearest subset mechanism in the information propagation step. We also provide the sensor management results for the localized GSSM algorithm with the predicted subset mechanism and the centralized branch-and-bound algorithm for performance comparison.

The true trajectories and estimated trajectories of the targets are illustrated in Fig. 5. The estimated trajectories accurately coincides with the true trajectories while filtering out the noise. The summations of position errors and velocity errors are shown in Fig. 6. It proves that the localized target tracking framework can achieve satisfactory tracking accuracy for the multi-target case.

Fig. 7 shows the performance gap between the localized GSSM algorithm and the optimal solution. For the nearest subset mechanism, the performance ratio converges to that of the predicted subset mechanism case after a short transient state. It verifies our intuitive assessment in the previous section.

## 7. Conclusions

In a wireless sensor network setup, this paper develops a localized multi-sensor multi-target tracking framework. Conservation of sensing, computation and communication resources features in all four modules of this framework: data acquisition, data fusion, information propagation, and sensor management. With a fully localized nearest subset mechanism for information propagation, a localized greedy-selection sensor management algorithm is proposed to allocate sensing resources using only local information. Compared to the centralized optimal solution, the localized solution outperforms in robustness and scalability, at the cost of acceptable performance degradation.

By combining the advanced filtering techniques, the current framework can be easily extended to the cases of non-linear models and non-Gaussian noises [8]. Another interesting future direction is to introduce mobile sensors to enhance the flexibility of the network, and improve the tracking accuracy [21].

## Acknowledgement

Qing Ling is supported in part by the NSFC grant #61004137. Zhi Tian is supported in part by the NSF grant #ECS-0925881.

## References

- S. Aeron, V. Saligrama. D. Castanon, Energy efficient policies for distributed target tracking in multihop sensor networks, in: Proceedings of IEEE Conference on Decision and Control, 2006, pp. 380–385.
- [2] S. Blackman, Multiple hypothesis tracking for multiple target tracking, IEEE Aerospace and Electronic Systems Magazine. 19 (2004) 5–18.
- [3] A. Chhetri, D. Morrell, A. Papandreou-Suppappola, Sensor resource allocation for tracking using outer approximation, IEEE Signal Processing Letters 14 (2007) 216–321.
- [4] M. Chu, H. Haussecker, F. Zhao, Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks, Journal of High Performance Computing Applications. 16 (2002) 293–313.
- [5] D. Guo, X. Wang, Dynamic sensor collaboration via sequential Monte Carlo, IEEE Journal on Selected Areas in Communications 22 (2004) 1037–1047.
- [6] Y. He, E. Chong, Sensor scheduling for target tracking: a Monte Carlo sampling approach, Digital Signal Processing 16 (2006) 533–545.
- [7] M. Kalandros, L. Pao, Covariance control for multisensor systems, IEEE Transactions on Aerospace and Electronic Systems 38 (2002) 1138–1157.
- [8] M. Kalandros, L. Trailovic, L. Pao, Y. Bar-Shalom, Tutorial on multisensor management and fusion algorithms for target tracking, in: Proceedings of the American Control Conference, 2004, pp. 4734–4748.
- [9] L. Kaplan, Global node selection for localization in a distributed sensor network, IEEE Transactions on Aerospace and Electronic Systems 42 (2006) 113–135.
- [10] L. Kaplan, Local node selection for localization in a distributed sensor network, IEEE Transactions on Aerospace and Electronic Systems 42 (2006) 136–146.
- [11] V. Krishnamurthy, Algorithm for optimal scheduling and management of hidden Markov model sensors, IEEE Transactions on Signal Processing 50 (2002) 1382–1397.
- [12] C. Lin, W. Peng, Y. Tseng, Efficient in-network moving object tracking in wireless sensor networks, IEEE Transactions on Mobile Computing 5 (2006) 1044–1056.
- [13] J. Liu, M. Chu, J. Reich, Multitarget tracking in distributed sensor networks, IEEE Signal Processing Magazine 24 (2007) 36–46.
- [14] G. Ng, K. Ng, Sensor management: what, why and how, Information Fusion 1 (2000) 67–75.
- [15] T. Vercauteren, D. Guo, X. Wang, Joint multiple target tracking and classification in collaborative sensor networks, IEEE Journal on Selected Areas in Communications 23 (2005) 714–723.
- [16] T. Vercauteren, X. Wang, Decentralized sigma-point information filters for target tracking in collaborative sensor networks, IEEE Transactions on Signal Processing 53 (2005) 2997–3009.
- [17] J. Vermaak, N. Ikoma, S. Godsill, Extended object tracking using particle techniques, in: Proceedings of IEEE Aerospace Conference, 2004, pp. 1876– 1885.
- [18] J. Vermaak, S. Godsill, P. Perez, Monte Carlo filtering for multi-target tracking and data association, IEEE Transactions on Aerospace and Electronic Systems 41 (2005) 309–332.
- [19] W. Xiao, L. Xie, J. Lin, J. Li, Multi-sensor scheduling for reliable target tracking in wireless sensor networks, in: Proceedings of the International Conference on ITS Telecommunications, 2006, pp. 996–1000.
- [20] N. Xiong, P. Svensson, Multi-sensor management for information fusion: issues and approaches, Information Fusion 3 (2002) 163–186.
- [21] P. Zhan, D. Casbeer, A. Swindlehurst, A centralized control algorithm for target tracking with UAVs, in: Proceedings of the Asilomar Conference on Signals, Systems and Computers, 2005, pp. 1148–1152.
- [22] F. Zhao, J. Shin, J. Reich, Information-driven dynamic sensor collaboration, IEEE Signal Processing Magazine 19 (2002) 61–72.