Distributed Spatio-Temporal Association and Tracking of Multiple Targets Using Multiple Sensors

Guohua Ren, Vasileios Maroulas, and Ioannis D. Schizas

Abstract

The problem of tracking multiple targets using nonlinear observations acquired at multiple sensors is addressed by combining particle filtering with sparse matrix decomposition techniques. Sensors are spatially scattered, while the unknown number of targets may be time-varying. A framework is put forth where norm-one regularized factorization is employed to decompose the sensor data covariance matrix into sparse factors whose support facilitates recovery of sensors that acquire informative measurements about the targets. This novel sensors-to-targets association scheme is integrated with particle filtering mechanisms to perform accurate tracking. Precisely, distributed optimization techniques are employed to associate targets with sensors, and particle filtering is integrated to perform target tracking using only the sensors selected by the sparse decomposition scheme. Different from existing alternatives, the novel algorithm can efficiently track and associate targets with sensors even in noisy settings. Extensive numerical tests are provided to demonstrate the tracking superiority of the proposed algorithm over existing approaches.

Index Terms

Particle filtering, sensor-to-targets association, distributed processing, multi-target tracking.

Work in this paper is supported by the NSF grant CCF 1218079, UTA and by Grant #279870 from the Simons Foundation and Grant #VF-2012-006 from the Leverhulme Trust to Vasileios Maroulas.

G. Ren and I. D. Schizas are with the Dept. of Electrical Engineering, University of Texas at Arlington, 416 Yates Street, Arlington, TX 76010. Tel/fax: (817) 272-3467/272-2253; Emails: guohua.ren@mavs.uta.edu, schizas@uta.edu.

V. Maroulas is with the Dept. of Mathematics, University of Tennessee at Knoxville, 202 Ayres Hall, 1403 Circle Drive Knoxville, TN 37996. Tel: (865) 974-4302; Email: maroulas@math.utk.edu.

I. INTRODUCTION

The deployment of sensor networks and development of pertinent information processing techniques can facilitate the requirement of situational awareness present in many surveillance and defense systems. Sensor networks allow the collection and distributed processing of information in challenging environments whose structure is not known and is dynamically changing with time, e.g. battlefields. In such harsh environments both equipment and infrastructure, as well as humans, are prone to threats that may be generated due to malicious attacks, functional failures and even human errors. Threats can be quite unpredictable both spatially and temporally, since they could happen anywhere anytime within a setting that consists of heterogeneous units, e.g,. communication units, sensing units and humans. Effective and fast target detection and tracking is really essential to avoid any potential negative effects.

A necessary step towards multi-target tracking is to associate sensors with targets across space and time. Targets present in the sensed field affect only a small portion of the deployed sensor networks (SNs). Thus, given the limited resources, it is pertinent to identify the sensors that acquire informative observations about the targets and use only those which provide this information. We characterize such sensors as 'target-informative' sensors in this manuscript. Many existing tracking techniques require all sensors to be active [1], [8], [39], [40], [57] which may be resource-consuming given the locality of the targets and the fact that only a few sensors bear information about the field targets. To this end, a decentralized algorithmic framework is developed here that does not require a central fusion center and it can associate sensors with targets combined with tracking.

Single-target tracking techniques have been developed for SNs using consensus-averaging techniques [9], [31] combined with the skeleton of particle filtering, e.g., see [10]. Further, extended Kalman filtering (EKF) for tracking a single-target is combined with a probabilistic framework for selecting sensors in [29]; an EKF for distributed multi-target tracking is considered in [41]. Data association and particle filtering have been applied in multi-target tracking applications where the measurements from a *single* sensor are used, while association takes place in time to determine which measurements contain information about a target [12], [16], [20]. Probabilistic models on the number of targets and the target-measurement assignments are also employed in [36] to perform multi-target tracking in single-sensor settings. Improved particle sampling techniques for single sensor settings are considered in [55], where particles corresponding to closely spaced targets are sampled jointly. The latter approaches require the availability of a probabilistic data model which is utilized to associate measurements acquired across time with the targets present. A centralized algorithm, that relies on Markov chain Monte Carlo (MCMC) tools,

performs data association on measurements acquired at a single-sensor across time in polynomial time [38]. The previous framework is extended to a network of sensors in [37]. Again the data association performs matching among temporal measurements and targets. Other centralized approaches that perform data association in time utilize Monte Carlo filtering, see e.g., [52] and [16].

A distributed algorithm that combines joint probabilistic data association with Kalman filtering has been developed in [42]. Though, some limitations are that linear Gaussian measurements models are assumed which are not always suitable for tracking applications, e.g. in low SNR environments and/or when the sensor observations are bearing and range (see e.g. [35]), while the consensus-averaging methods [54] employed, force all sensors to be active and be used in the tracking process despite the fact that some of them may have low quality observations. A different approach is followed in [48] where multiple fusion centers are present in the sensor network and evaluate the posterior Cramer-Rao lower bound that requires knowledge of the underlying data model. Then, as long as the fusion centers know which targets they track, then they can select the sensors which result the smallest Cramer-Rao lower bound. The novel algorithm proposed here does not require linear data models to operate and furthermore does not assume that sensors are aware of which targets they track. A related distributed approach for tracking a *single* target is also proposed in [29]. The latter approach utilizes extended Kalman filtering, while assuming a probabilistic model to determine the sensors that are closely located to the target. Further, the scheme in [29] relies on the target position estimates and leads to instability in noisy environments as will be demonstrated via numerical tests. Finally, work has been done in sensor scheduling and tracking in [13], [26] where the focus is to determine at which time-intervals a sensor operates and when it should be idle. Further, the approach in [56] assumes the availability of the target position to activate sensors using tree-based structures in the network topology.

An algorithmic framework is proposed here that associates targets with the sensors which acquire informative measurements about these targets, and subsequently performs tracking using only these informative sensors. Note that existing data association schemes [12], [16], [20], [36], [37], [52] match measurements with targets across time and rely on probabilistic models. Differently, the sensors-targets association task here is relying only on the acquired sensor data and no probabilistic models are adopted. Specifically, sensors which are positioned close to the same target, acquire data measurements that tend to be correlated, *no matter* what the underlying physical model is. Such correlations induce a sparse structure (presence of many zeros) in the sensor data covariance matrix. Sparsity is an attribute found in many natural and man-made signals, and it has been exploited in a wide range of applications including sparse regression, sub-Nyquist sampling and statistical inference, e.g., see [7], [49].

To facilitate association of sensor measurements with targets a pertinent framework is derived to analyze the sensor data covariance into sparse factors whose support (position of the nonzero entries) will indicate subsets of sensors sensing the same target. Different from [17], [19], [27], [28], [51], [58], the matrix factorization scheme developed here does not require a central fusion center and does not impose structural requirements to the unknown factors such as orthogonality and/or positivity of the factor entries. The idea of covariance sparse factorization was also discussed in [44]. However, the work in [44] is dealing with stationary settings where the targets/sources present in the field are static and immobile, while linear data models are considered not pertinent for tracking applications. Here the framework in [42] is generalized in *nonlinear* highly *dynamic* and time-varying settings where sensors acquire information about multiple *moving* targets whose number may also be changing in time.

There is a plethora of strategies which address the multi-target-tracking problem, for example see the partial list [4], [5], [24], [30], [32]–[35], [53] and references therein. The tracking process here is carried out via particle filtering (PF) [2], [10], [15] due to its flexibility to handle nonlinear and/or nonGaussian scenarios. PF will be combined here with the aforementioned sparse factorization scheme to cope with the time-varying settings and perform real-time association of sensors and targets. A common assumption present in existing multi-target schemes, e.g., [12], [16], [20], [36], [37], [52], is that sensor measurements contain information about *one* target. Here a more relaxed assumption is introduced where each sensor may be sensing multiple targets among which at most one of them will have the strongest signal contribution in the measurement. The latter property holds true as long as the targets are positioned such that no more than one targets are inside the sensing region of a sensor which further allows to distinguish the targets as separate objects during sensor-to-target association. Effectively each sensor can associate with at most one target.

After formulating the problem and giving the necessary background in Sec. II, the task of associating targets with sensors is derived and generalized in the nonlinear and time-varying setting in III-A. Norm-one regularization is combined with pertinent least-squares matrix decomposition which leads to a novel minimization formulation that is tackled via coordinate descent techniques in Sec. III-B. A distributed sensors-targets association algorithm is derived that requires communication only between neighboring sensing units. The latter association scheme is combined with particle filtering (Sec. IV) to enable tracking of multiple targets by utilizing only the measurements of target-informative sensors (Sec. V). After studying the communication cost associated with the proposed algorithmic framework (Sec. VI), extensive numerical tests in Sec. VII show the advantages of the novel scheme here over existing alternatives. Table I summarizes the symbols and their functionality in the paper.

II. Problem Formulation and Preliminaries

Consider an ad-hoc multi-sensor network with a total number of m sensors. Each sensor is able to communicate with its single-hop neighboring sensors which are within its range. The single-hop neighborhood for sensor j will be denoted by $\mathcal{N}_j$, while the sensor network (SN) is modeled as an undirected graph and the inter-sensor links are assumed to be symmetric [see dashed lines (single hop) in Fig. 1]. The connectivity information of the SN is summarized by the $m \times m$ adjacency matrix $\mathbf{E}$ whose (j, j')th entry will be 1 if sensors j and j' are connected and zero otherwise. Sensors monitor a field on which an unknown and time-varying number of multiple moving targets is present. The targets on the field are sensed via measurements $x_j(t)$ acquired at sensor j and time instant t. For instance, in Fig. 1, there are two targets whose location is denoted by the red and green stars. The targets are moving at spatially different locations in the field affecting different parts of the SN. A general setting is considered where new targets are sensed at a given time instance, while other targets maybe becoming inactive (e.g., they are eliminated in a tactical environment). This leads to a setting where the number of targets is time-varying.

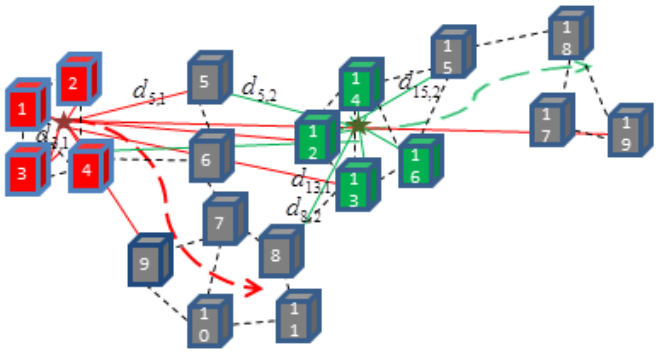

Fig. 1. Tracking multiple targets in a sensor network.

Sensors measure the intensity of signals received from the different moving targets on the field. Sensor j acquires a *scalar* measurement at time instant t that adheres to the following model

$$\mathbf{x}_j(t) = \sum_{\rho=1}^{R} b_\rho(t) d_{j,\rho}^{-2}(t) + w_j(t), \; j = 1, \ldots, m \tag{1}$$

where $b_\rho(t)$ denotes the intensity of a signal *emitted* by the ρth target, while $d_{j,\rho}(t)$ denotes the distance between the ρth target and sensor j at time t. The number R corresponds to the total number of different

targets that move through the field over the lifetime of the SN, while $w_j(t)$ denotes the zero-mean temporally white sensing noise with variance equal to σ_w^2. Note that (1) is formulated assuming that the targets act as transmitters. The signal $b_\rho(t)$ emitted from target ρ propagates via free-space to arrive at sensor j attenuated as $b_\rho(t)/d_{j,\rho}^2(t)$. From wireless transmission (see [14, Ch. 2]), it is known that signals emitted from different targets and propagating via free-space are superimposed in the way described in (1), while the additive noise corresponds to random perturbations generated by the sensors' electronic components. Each of the $b_\rho(t)$ signals emitted by a moving target can be the result of, e.g., a radar signal impinging on the ρ target surface and then bouncing back. Thus, $b_\rho(t)$ could be viewed as the signal resulting after the radar signal has bounced back from target ρ surface. If the radar signal has intensity $\beta(t)$, then the intensity of the signal emitted by the target would be proportional to $b_\rho(t) \sim \frac{\beta(t)}{d_\rho^2(t)}$, where $d_\rho(t)$ denotes the distance of the radar from target ρ. This would give rise to fourth-order distance terms in (1), however here since $b_\rho(t)$ corresponds to the signal emitted by target ρ second-order distance terms appear in (1). Assuming that each sensor will receive one reflection of the bounced radar signal,then the superimposing argument given earlier can be used here to derive (1).

Assuming that targets are independently moving in space, the intensity of the signals bouncing back from the target surfaces are considered to be uncorrelated. Note from (1) that among the summands $b_\rho(t)d_{j,\rho}^{-2}(t)$ some have very small amplitude when sensor j is far from target ρ, whereas others have large amplitude when sensor j is close to target ρ [$d_{j,\rho}(t)$ is small]. Here it is assumed that among these summands in (1) only one has strong amplitude whereas the rest are negligible. This pertains to a setting where only one target, say the ρth target, is close to sensor j whereas the rest are sufficiently far thus their contribution in $x_j(t)$ is very small. This can be realized when targets are sufficiently separated in space such that no more than one targets are positioned inside the sensing region of a sensor, i.e., how far the sensors can 'see'. As the sensing range (and sensing region) of the sensors is reduced (cannot 'see' very far), targets can be located closer while ensuring the previous assumption of one dominant target. Of course it is understood that the density of sensing units should increase in order to better cover the sensed field. Thus, targets can be placed arbitrarily close by reducing the sensors' sensing range. This is a more 'relaxed' version of the common assumption that sensor measurements in multi-target tracking contain information about just *one* target [12], [16], [20], [36], [37], [52]. The intensity $b_\rho(t)$ will be nonzero only for the interval for which a target is sensed by the sensors, otherwise will be zero deactivating target ρ in (1). For instance, if a target is sensed moving within interval $[t_1, t_2]$, then $b_\rho(t) = 0$ for $t < t_1$ and $t > t_2$. At a given time t a subset of the targets, say of cardinality $r(t)$, will be active ($b_\rho(t) \neq 0$) in (1). The distance term $d_{j,\rho}(t)$ is equal to $\|\mathbf{p}_j - \mathbf{p}_\rho(t)\|$, where $\|\cdot\|$ denotes the Euclidean norm, $\mathbf{p}_j \in \mathbb{R}^{K\times 1}$

is the fixed and available position of sensor j, while $\mathbf{p}_\rho(t) := [p_{\rho,x_1}(t), p_{\rho,x_2}(t), \ldots, p_{\rho,x_K}(t)]^T \in \mathbb{R}^{K\times 1}$ denotes the unknown ρth target position in a K-dimensional manifold.

Each target, say the ρth is characterized by a $2K \times 1$ state vector $\mathbf{s}_\rho(t)$ that contains at a given time t its location $\mathbf{p}_\rho(t)$ and the velocity $\mathbf{v}_\rho(t) := [v_{\rho,x_1}(t), \ldots, v_{\rho,x_K}(t)]^T$ at the K different spatial directions, i.e., $\mathbf{s}_\rho(t) := [\mathbf{p}_\rho^T(t), \mathbf{v}_\rho^T(t)]^T$. The target states evolve according to a general Markov model:

$$\mathbf{s}_\rho(t+1) = \mathbf{g}(\mathbf{s}_\rho(t), \mathbf{u}_\rho(t)), \tag{2}$$

where $\mathbf{g}(\cdot,\cdot) : \mathbb{R}^{2K\times 1} \times \mathbb{R}^{2K\times 1} \to \mathbb{R}^{2K\times 1}$ is family of nonlinear, nonsingular functions, while $\mathbf{u}_\rho(t)$ denotes the state noise. Details on the state model used for the numerical tests will be given in Sec. VII.

Fusing all sensor measurements in (1) on an $m\times 1$ vector we obtain the following measurement model

$$\mathbf{x}_t = \mathbf{D}_t\mathbf{b}_t + \mathbf{w}_t, \text{ where } \mathbf{b}_t := [b_1(t)\ b_2(t)\ldots b_R(t)]^T, \tag{3}$$

while $\mathbf{D}_t$ is a $m \times R$ matrix with entries $\mathbf{D}_t(j,\rho) = d_{j,\rho}^{-2}(t)$ with $j = 1,\ldots,m$ and $\rho = 1,\ldots,R$. The noise $\mathbf{w}_t$ has covariance $\mathbf{\Sigma}_w = \sigma_w^2\mathbf{I}_m$, where $\mathbf{I}_m$ denotes the $m \times m$ identity matrix. Note that vector $\mathbf{x}_t$ is not stored somewhere and it is introduced here for notational purposes. Given that the entries of $\mathbf{b}_t$ are uncorrelated, it follows that the data covariance matrix is

$$\mathbf{\Sigma}_{x,t} = \mathbf{D}_t\mathbf{\Sigma}_b\mathbf{D}_t^T + \sigma_w^2\mathbf{I}_m = \bar{\mathbf{D}}_t\bar{\mathbf{D}}_t^T + \sigma_w^2\mathbf{I}_m, \tag{4}$$

where $\mathbf{\Sigma}_b$ is a diagonal matrix whose diagonal entries correspond to the variance of the entries in $\mathbf{b}_t$, while $\bar{\mathbf{D}}_t := \mathbf{D}_t\mathbf{\Sigma}_b^{1/2}$. Note that the matrix $\mathbf{D}_t$ is time-varying since the distance of the sensors from the targets is changing with time. Further, among the R entries in $\mathbf{b}_t$, there will be $r(t)$ nonzero entries corresponding to the active targets moving at the sensed field at t. Inactive targets at time instant t (either far away from sensors) will be represented by zero $b_\rho(t)$'s that will further zero out the corresponding columns in $\mathbf{D}_t$. Here it is assumed that once a target becomes inactive (i.e. $b_\rho(t) = 0$) it remains inactive.

The ρth column of $\mathbf{D}_t$ contains the distances of all sensors from target ρ at time t. For sensors close to target ρth, the corresponding distances, $d_{j,\rho}(t)$, will be relatively small, thus leading to relatively large entries $\mathbf{D}_t(j,\rho) = d_{j,\rho}^{-2}(t)$, compared to sensors that are further away. For example in Fig. 1, where the number of targets $R = 2$ and $K = 2$, target 1 (green star) will be close to sensors $\{1,2,3,4\}$, while target 2 will be close to sensors $\{12,13,14,16\}$. The measurements of the aforementioned sensors are expected to be strongly affected by the target intensity signals and have a richer information content about the present targets compared to the remaining sensors that are further away. Since targets at a given instant t are very localized and close to a small percentage of sensors, many entries of any column, say the ρth, in $\mathbf{D}_t$ are expected to be close to zero giving rise to an approximately sparse matrix $\mathbf{D}_t$.

Matrices $\mathbf{D}_t$ here are constantly changing, due to the presence of mobile targets. This is to be contrasted with the stationary setting in [44] where sources are immobile and the corresponding covariance matrix time-invariant.

Notice that the matrices $\mathbf{D}_t$ are not available since the targets' locations are not available. Further, someone may approximate $\mathbf{D}_t$ by first applying tracking techniques to estimate the targets' locations and subsequently the entries of $\mathbf{D}_t$. However, the presence of a time-varying number of multiple-targets and the fact that it is unknown which target corresponds to each measurement make our framework more challenging and different than the one in [44]. Nonetheless, if there was a way to locate where the strong-amplitude and small-amplitude entries are located in the ρth column $\mathbf{D}_{t,\rho:}$, then we can identify which sensors are close and acquire informative observations about a specific target, say the ρth. This step of associating sensor measurements with targets will be of paramount importance before applying any tracking techniques. A spatio-temporal data association framework will be designed here that allows sensors to collaborate and determine which subsets of sensors acquire informative measurements about the $r(t)$ active targets at time instant t. This will be executed by employing sparsity-regularization techniques to estimate $\mathbf{D}_t$ and decompose it into sparse factors. Note that existing data association schemes [12], [16], [20], [36], [37], [52] match measurements with targets across time and rely on probabilistic models. The sensor-target association framework proposed here will then be integrated with particle filtering techniques that will be encountered across the different sets of informative sensors to track accurately the targets' positions. The goal here is twofold: i) determine the active targets and identify the sensors that acquire informative measurements about them; and ii) perform tracking among the different subsets of target-informative sensors.

III. Spatio-Temporal Target-to-Sensor Data Association

A. Sparsity-Cognizant Minimization Framework

As mentioned earlier, at a given time instant, the number of targets present on the monitored field, as well as which sensors have informative measurements about a specific target are unknown. Let $\mathcal{T}_{\rho,t}$ denote the subset of sensors that are closely located to target ρ and whose measurements [see eq. (1)] are dominated by the ρth target component at time t. This implies that for all sensors $j \in \mathcal{T}_{\rho,t}$ the corresponding measurements will be approximately distributed as

$$\mathbf{x}_j(t) \sim \mathcal{N}(b_\rho(t) d_{j,\rho}^{-2}(t), \sigma_w^2), \tag{5}$$

since sensors $j \in \mathcal{T}_{\rho,t}$ are much closer to target ρ than the rest targets, resulting $b_\rho(t)d_{j,\rho}^{-2}(t)$ to be the dominant summand in (1). This stems directly from the assumption introduced in Sec. II that targets are sufficiently separated in space, which further implies that $d_{j,\rho} \ll d_{j,\rho'}$ for $\rho' \neq \rho$.

Next, we derive a technique to track the sensor subsets $\mathcal{T}_{\rho,t}$. In this way we will manage to associate sensor measurements with targets. Note that the data association here is spatio-temporal and will be done in a distributed manner and does not rely on specific probabilistic models. Different from [44], the subsets $\mathcal{T}_{\rho,t}$ are time-varying due to the moving targets. Thus, the stationary framework developed in [44] is generalized here to dynamic and time-varying settings. Note that the rank of the information component in the data covariance in (4), which is equal to $r(t)$, reveals the number of active sensed targets at time instant t, whereas the relatively strong-amplitude entries in each column $\{\mathbf{D}_{t,:\rho}\}_{\rho=1}^{R}$ reveal the members of a target-informative subset $\mathcal{T}_{\rho',t}$. With these properties in mind, it is of interest to decompose the sensor data covariance matrix $\mathbf{\Sigma}_{x,t}$ into sparse factors whose nonzero entries will indicate where the strong-amplitude entries are in $\bar{\mathbf{D}}_t$.

Note that the sensor data covariance $\mathbf{\Sigma}_{x,t}$ is time-varying due to the changing number of targets and their movements, while in practical situations the ensemble covariance is not available. This is to be contrasted with the setting in [44], where the sensor data covariance matrix is time-invariant. The covariance matrix should be updated in a way that gives more emphasis to the more recent sensing data while forgets the old data gradually. This process is achieved by utilizing exponential weighing, and is a common technique in adaptive signal processing to estimate efficiently time-varying covariance matrices, see e.g., [43], [46]. Specifically, the covariance entries are estimated by

$$\hat{\mathbf{\Sigma}}_{x,t} = \frac{1-\gamma}{1-\gamma^{t+1}} \sum_{\tau=0}^{t} \gamma^{t-\tau} (\mathbf{x}_\tau - \bar{\mathbf{x}}_t)(\mathbf{x}_\tau - \bar{\mathbf{x}}_t)^T, \tag{6}$$

where $\gamma \in (0,1)$ denotes the forgetting factor that controls the 'memory' process and

$$\bar{\mathbf{x}}_t = \frac{1-\gamma}{1-\gamma^{t+1}} \sum_{\tau=0}^{t} \gamma^{t-\tau} x_\tau, \tag{7}$$

corresponds to an adaptive estimate for the data ensemble mean which is also time-varying. Note that $\gamma^{t-\tau}$ decreases as τ decreases (corresponding to past data), while for $\tau = t$ (present datum) the coefficient multiplying $\mathbf{x}_t$ is equal to one. Thus, the present datum is multiplied by the maximum possible value that $\gamma^{t-\tau}$ can reach. The scaling $(1-\gamma)(1-\gamma^{t+1})^{-1}$ in (6) and (7) is introduced here to ensure that the time-varying covariance and mean estimates $\hat{\mathbf{\Sigma}}_{x,t}$ and $\bar{\mathbf{x}}_t$ will be unbiased estimates of the ensemble quantities $\mathbf{\Sigma}_{x,t}$ and $\mathbb{E}[\mathbf{x}_t]$ respectively, in a time-invariant (stationary) setting, i.e.

$$\mathbb{E}[(1-\gamma)(1-\gamma^{t+1})^{-1}\hat{\mathbf{\Sigma}}_{x,t}] = \mathbf{\Sigma}_{x,t}, \text{ and } \mathbb{E}[\bar{\mathbf{x}}_t] = \mathbb{E}[\mathbf{x}_t].$$

Thus, the scaling introduced in (6) and (7) ensures that the obtained adaptive estimates are properly normalized to give unbiased estimates in a stationary setting which further implies good estimation in non-stationary settings too, see e.g., [43].

In order to adhere to the single-hop connectivity constraints summarized in the adjacency matrix $\mathbf{E}$, each sensor j is responsible for evaluating the 'single-hop' covariance entries $\hat{\mathbf{\Sigma}}_{x,t}(j,j')$ where $j' \in \mathcal{N}_j$. For example, sensor 5 in Fig. 1 will be able to evaluate only the single-hop covariance entries $\hat{\mathbf{\Sigma}}_{x,t}(5,5), \hat{\mathbf{\Sigma}}_{x,t}(5,6)$. The latter tasks involve the exchange of scalar measurements $x_j(t)$ between single-hop neighbors during time t. Thus, covariance entries that correspond to sensors more than one hop away will not be evaluated in the SN.

A standard least-squares based matrix factorization scheme would minimize the Frobenius norm-based cost $\|\hat{\mathbf{\Sigma}}_{x,t} - \mathbf{M}_t\mathbf{M}_t^T - \sigma^2\mathbf{I}_{m\times m}\|_F^2$ with respect to (wrt) the factor estimates in $\mathbf{M}_t \in \mathbb{R}^{m\times r}$. However, such a formulation does not account for the nearly sparse structure of $\bar{\mathbf{D}}_t$. In fact it assumes that the number r of factors (sensed targets) is available, while all covariance entries are available. The need for a framework that accounts for sparsity, unknown number of targets and single-hop connectivity is apparent. To this end, the following framework is put forth

$$\left(\hat{\mathbf{M}}_t, \{\hat{\sigma}_j\}_{j=1}^m\right) := \arg\min_{\mathbf{M}_t, \{\sigma_j\}_{j=1}^m} \|\mathbf{E} \odot \left(\hat{\mathbf{\Sigma}}_{x,t} - \mathbf{M}_t\mathbf{M}_t^T - \text{diag}(\sigma_{1,t}^2, \ldots, \sigma_{m,t}^2)\right)\|_F^2 + \sum_{\ell=1}^{L} \lambda_\ell \|\mathbf{M}_{t,:\ell}\|_1 + \phi \sum_{\ell=1}^{L} \|\mathbf{M}_{t,:\ell}\|_2^2, \quad (8)$$

where $\odot$ denotes the Hadamard operator (entry-wise matrix product), σ_j^2 is the local noise variance estimate at sensor j, while L is an upper bound for the number of active sensed targets $r(t)$ ($L \geq r(t)$) and $\mathbf{M}_{t,:\ell}$ denotes the ℓth column of $\mathbf{M}_t$. Although the sensing noise variance σ_w^2 is common across all sensors we introduce different noise variance estimates $\sigma_{j,t}^2$ to facilitate the development of a decentralized iterative minimization technique for (8). Matrix $\mathbf{M}_t \in \mathbb{R}^{m\times L}$ contains L columns that will estimate the sparse matrix columns of $\bar{\mathbf{D}}_t$, while L is selected sufficiently large to ensure that is an upper bound for the number of present targets $r(t)$.

Sensor j will be responsible for updating the jth row in $\mathbf{M}_t$, namely $\mathbf{M}_{t,j:}$ for $j = 1, \ldots, m$. The adjacency matrix $\mathbf{E}$ in (8) along with the nature of the Hadamard operator ensure that only the available single-hop covariance entries will participate in the minimization formulation, while the updating recursions that will be obtained later for each sensor j to find $\hat{\mathbf{M}}_{t,j:}$ will require message exchanges only between single-hop neighbors. The first term in (8) accounts for the fact that the covariance assumes the structure in (4). The second (norm-one) term in (8) induces sparsity in the columns of $\mathbf{M}_t$ to account

for the approximately sparse structure of $\bar{\mathbf{D}}_t$. Norm-one regularization is well known to affect sparsity in several estimation and regression problems [49], [58]. The larger the nonnegative sparsity-controlling coefficient λ_ρ is, the more zeros the estimated factor $\hat{\mathbf{M}}_{t,:\rho}$ will contain. The third term in (8), where $\phi \geq 0$, is present to adjust the number of nonzero columns of $\hat{\mathbf{M}}_t$ needed to accurately represent $\hat{\mathbf{\Sigma}}_{x,t}$. The number of nonzero columns in $\hat{\mathbf{M}}_t$ will be smaller than L and can be used as an estimate for the number of sensed targets $r(t)$ at time t.

Notice that the optimization formulation in (8) is also different from the one given in [44]. The difference is in the last two terms which are there to control the number of nonzero rows in matrix $\mathbf{M}_t$. In fact the number of nonzero rows in $\mathbf{M}_t$ will correspond to an estimate of the number of targets present in the field. The scheme in [44] works under the assumption that the number of sources is known. Another feature of the minimization formulation in (8), not present in [44], is the estimation of the sensing noise variances σ_j^2, which in general are unknown, and not available as is the case in [44].

B. Decentralized Algorithm

An iterative algorithm is proposed here to minimize numerically the cost in (8) derived using coordinate descent techniques [6], [50]. The approach followed here is to minimize the cost in (8) recursively wrt an entry of $\mathbf{M}_t$ or $\text{diag}(\sigma_1^2, \ldots, \sigma_m^2)$, while keeping the remaining entries fixed. During one coordinate descent cycle all the entries of matrix $\mathbf{M}_t$ and $\text{diag}(\sigma_{1,t}^2, \ldots, \sigma_{m,t}^2)$ are updated. Sensor j is responsible for updating the entries $\{\mathbf{M}_t(j,\ell)\}_{\ell=1}^L$ and $\sigma_{j,t}^2$. Given the most recent updates $\hat{\mathbf{M}}_t^{k-1}$ and $\{\sigma_{j,t,k-1}^2\}$ at the end of coordinate cycle $k-1$, updates $\hat{\mathbf{M}}_t^k(j,\ell)$ at sensor j can be formed by differentiating (8) wrt $\mathbf{M}_t(j,\ell)$ while fixing the rest of the minimization variables to their most up-to-date values from cycle $k-1$. It turns out that (see Appendix A) during coordinate cycle k, the update $\hat{\mathbf{M}}_t^k(j,\ell)$ can be obtained as the value that achieves the minimum possible cost in (8) (while fixing the rest of the variables) among the candidate values: i) $y = 0$; ii) the real positive roots of the third-degree polynomial

$$4y^3 + 4\left[\sum_{i\in\mathcal{N}_j}[\hat{\mathbf{M}}_t^{k-1}(i,\ell)]^2 - \zeta_{t,\Sigma}^k(j,j,\ell) + 0.5\phi\right]y - \left[4\sum_{i\in\mathcal{N}_j}\zeta_{t,\Sigma}^k(j,\mu,\ell)\hat{\mathbf{M}}_t^{k-1}(i,\ell)\right] + \lambda_\ell = 0 \quad (9)$$

and iii) the real negative roots of the third-degree polynomial

$$4y^3 + 4\left[\sum_{i\in\mathcal{N}_j}[\hat{\mathbf{M}}_t^{k-1}(\mu,\ell)]^2 - \zeta_{t,\Sigma}^k(j,j,\ell) + 0.5\phi\right]y - \left[4\sum_{i\in\mathcal{N}_j}\zeta_{t,\Sigma}^k(j,i,\ell)\hat{\mathbf{M}}_t^{k-1}(i,\ell)\right] - \lambda_\ell = 0 \quad (10)$$

where

$$\zeta_{t,\Sigma}^k(j,i,\ell) := \hat{\mathbf{\Sigma}}_{x,t}(j,i) - \delta_{j,i}\hat{\sigma}_{j,t,k-1}^2 - \sum_{\ell'=1,\ell'\neq\ell}^{L}\hat{\mathbf{M}}_t^{k-1}(j,\ell')\hat{\mathbf{M}}_t^{k-1}(i,\ell') \quad (11)$$

while $\delta_{j,i}$ denotes the Kronecker delta, i.e., $\delta_{j,i} = 1$ if $j = i$, and $\delta_{j,i} = 0$ if $j \neq i$.

Further, the noise variance estimates across sensors can be updated during cycle k at time instant t as

$$\hat{\sigma}^2_{j,t,k} = \hat{\mathbf{\Sigma}}_{x,t}(j,j) - \hat{\mathbf{M}}^k_{t,j:}(\hat{\mathbf{M}}^k_{t,j:})^T, \;\; j = 1, \ldots, m. \tag{12}$$

The roots of (9) and (10) can be obtained using, companion matrices [18]. Sensor j can evaluate the coefficients of the polynomials in (9) and (10) by communicating only with its neighbors in $\mathcal{N}_j$. In detail, sensor j receives $\{\hat{\mathbf{M}}^{k-1}_t(i,1), \ldots, \hat{\mathbf{M}}^{k-1}_t(i,L)\}$ and the latest measurements $\{x_i(t)\}$ from sensors $i \in \mathcal{N}_j$ to form the single-hop covariance updates $\hat{\mathbf{\Sigma}}_{x,t}(j,i)$ and subsequently evaluate $\zeta^k_{t,\Sigma}(j,i,\ell)$. Similarly, it sends to its neighbors the L scalar updates for the jth row of $\mathbf{M}_t$, namely $\{\hat{\mathbf{M}}^{k-1}_t(j,1), \ldots, \hat{\mathbf{M}}^{k-1}_t(j,L)\}$ and its current measurement $x_j(t)$. Further, each sensor j can update the noise variance estimates $\hat{\sigma}^2_{j,t,k}$ using only locally available information as can be seen in (12). To facilitate a real-time implementation a small fixed number, say κ, of coordinate cycles is applied per time t. Note that the proposed scheme also involves constant updating of the single-hop covariance entries $\hat{\mathbf{\Sigma}}_{x,t}(j,i)$ needed in $\zeta^k_{t,\Sigma}(j,i,\ell)$ to account for the constantly changing statistical properties of the sensed field. Such online updating is not present in [44].

The task of forming the updates $\{\hat{\mathbf{M}}_t(j,\ell)\}^L_{\ell=1}$ at sensor j at time instant t boils down to determining the roots of the third-degree polynomials given in (9) and (10). The latter task involves: i) evaluating the quantities $\{\zeta^k_{t,\Sigma}(j,\mu,\ell)\}_{\mu\in\mathcal{N}_j\cup\{j\},\ell=1,\ldots,r}$, with a computational complexity of the order of $\mathcal{O}(|\mathcal{N}_j|r^2)$, i.e., $L|\mathcal{N}_j|$ coefficients each evaluated in (11) with a complexity of $\mathcal{O}(L)$; ii) evaluating the $4|\mathcal{N}_j|$ coefficients of the polynomials at (9) and (10) with a complexity of $\mathcal{O}(4|\mathcal{N}_j|L)$; and iii) determining the roots of the third-order polynomials in (9) and (10) that involves evaluation of the corresponding $2L$ companion matrices of size 3×3 at a computational complexity of $\mathcal{O}(L)$. Note that the cost at sensor j per coordinate cycle is linearly dependent on the number of single-hop neighbors $|\mathcal{N}_j|$, while the dependency is quadratic when it comes to the upper bound of the number of targets L. Nevertheless, the number of sources r (and thus L) in practical scenarios is much smaller than the number of sensors m.

Once the sparse factors $\{\hat{\mathbf{M}}_{t,:\ell}\}^{\hat{r}(t)}_{\ell=1}$ are estimated, where $\hat{r}(t) < L$ corresponds to the number of nonzero columns of $\hat{\mathbf{M}}_t := \hat{\mathbf{M}}^\kappa_t$ at t, their support (nonzero entries) can be used to identify the sensors that sense a specific target at time instant t. In that way sensor subsets $\mathcal{T}_{\ell_t,t}$ for $\ell_t = 1, \ldots, \hat{r}(t)$ can be identified and used to track $\hat{r}(t)$ different targets. One challenge that will be addressed in Sec. V is how to determine whether two subsets $\mathcal{T}_{\ell_t,t}$ and $\mathcal{T}_{\ell'_t,t'}$ evaluated at different time instances correspond to the same target or not. This time-association step is necessary to make sure that estimated trajectories corresponding to different targets are updated using newly acquired sensor measurements that correspond to the correct dominant target. At a given time instant t the steps followed across sensors, which form a

connected network, to perform decentralized data association is tabulated as Algorithm 1. During time instant t one coordinate cycle k involves updating the $m \times L$ entries $\hat{\mathbf{M}}_t^k$ and the m variance estimates $\sigma^2_{j,t,k}$ via (9), (10) and (12). In Apdx. B it is demonstrated that Alg. 1 converges at least to a stationary point of (8). The parameters $\{\lambda_\ell\}_{\ell=1}^L$ can be set using the strategy proposed in [44].

To end the iterative process involved in Alg. 1, each sensor j proceeds to evaluate the Euclidean norm of the difference between two consecutive estimates, namely $\|\hat{\mathbf{M}}_{t,j:}^{k-1} - \hat{\mathbf{M}}_{t,j:}^{k}\|_2$, found during iteration steps k and $k-1$. Using a max consensus scheme, e.g., [21], the maximum of these m norm quantities can be found across sensors which then they compare this maximum with a desired threshold of accuracy. Once the maximum norm $\|\hat{\mathbf{M}}_{t,j:}^{k-1} - \hat{\mathbf{M}}_{t,j:}^{k}\|_2$ (sensor with largest updating difference) is less than a threshold ϵ which could be set as a adjustable small positive value (in our tests is set as $5 \cdot 10^{-3}$), then the updating process involved in Alg. 1 will stop across sensors.

Algorithm 1 Distributed Target-Sensor Association

1: At time instant t:
2: Sensor j updates $\hat{\mathbf{\Sigma}}_{x,t}(j,j')$ for $j' \in \mathcal{N}_j \cup \{j\}$ using (6) after receiving the most recent data $\{x_{j'}(t)\}_{j' \in \mathcal{N}_j}$ from its neighbors.
3: Sensor j initializes the jth row of $\mathbf{M}_t$ as $\hat{\mathbf{M}}_{t,j:}^0 = \mathbf{0}_{1\times L}$, while it sets $\hat{\sigma}^2_{j,t,0} = 0$.
4: **for** $k = 1, 2, \ldots, \kappa$ **do**
5: Each sensor j for $j = 1, \ldots, m$:
6: Transmits $\{\hat{\mathbf{M}}_t^{k-1}(j,\ell')\}_{\ell'=1}^L$ to its neighbors in $\mathcal{N}_j$, and receives $\{\hat{\mathbf{M}}_t^{k-1}(j',\ell')\}_{\ell'=1}^L$ from $j' \in \mathcal{N}_j$.
7: Evaluates $\zeta_{t,\Sigma}^k(j,i,\ell)$ for $i \in \mathcal{N}_j \cup \{j\}$ via (11).
8: Determine the updates $\{\hat{\mathbf{M}}_t^k(j,\ell)\}_{\ell=1}^L$ after determining the positive roots of (9) and the negative roots of (10).
9: If $\max_{j=1,\ldots,m}(\|\hat{\mathbf{M}}_{t,j:}^{k-1} - \hat{\mathbf{M}}_{t,j:}^{k}\|_2) \leq \epsilon$ then stop.
10: **end for**

IV. Tracking via Particle Filtering

Next we will take into our advantage the target-informative sensor subsets $\mathcal{T}_{\rho,t}$ which have been retrieved using the decentralized framework in Sec. III in order to perform multi-target tracking. We focus on executing the tracking process via particle filtering (PF) [2], [15] due to its flexibility to handle nonlinear and/or nonGaussian scenarios as in our observation model in (1). For each subset $\mathcal{T}_{\rho,t}$ a different PF will be constructed to track the corresponding target ρ.

Tracking objects consists of computing a conditional expectation

$$\mathbb{E}(f(\mathbf{s}_{\rho,t})|\mathbf{x}_{\mathcal{T}_{\rho,0:t}}) = \int f(\mathbf{s}_{\rho,t}) p(\mathbf{s}_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}}) d\mathbf{s}_{\rho,t}$$

of a function of the state, $\mathbf{s}_{\rho,t}$, of a target ρ, using the measurements of the sensors within the informative subset $\mathcal{T}_{\rho,t}$. Equivalently, the conditional density $p(\mathbf{s}_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})$ given the measurements needs to be computed instead. These measurements are denoted herein by the $|\mathcal{T}_{\rho,t}| \times 1$ vector $\mathbf{x}_{\mathcal{T}_{\rho,t}} := \{x_j(t)\}_{j\in\mathcal{T}_{\rho,t}}$, where $|\mathcal{T}_{\rho,t}|$ denotes the cardinality of sensor subset $\mathcal{T}_{\rho,t}$. The measurements are affiliated with a pertinent likelihood function which depends on the underlying observation model. We generally denote this likelihood function by $p(\mathbf{x}_{\mathcal{T}_{\rho,t}}|\mathbf{s}_{\rho,t})$ given the state $\mathbf{s}_{\rho,t}$ of the ρth target at time t. The reader may refer to Sec. VII for specific details on the likelihood and the associated observation model. All the available data from time 0 up to the current time instant t will be used. Let $\mathbf{x}_{\mathcal{T}_{\rho,0:t}}$ denote the sensor measurements associated with the informative sensor subsets $\mathcal{T}_{\rho,0}, \mathcal{T}_{\rho,1}, \ldots, \mathcal{T}_{\rho,t}$ within the time horizon $[0, t]$.

However, in many instances, it is a rather formidable task to compute or approximate the conditional density, $p(\mathbf{s}_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})$. Therefore, employing importance sampling techniques, one may consider a different distribution, say $q(\mathbf{s}_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})$ and the aforementioned conditional expectation is derived

$$\mathbb{E}(f(\mathbf{s}_{\rho,t})|\mathbf{x}_{\mathcal{T}_{\rho,0:t}}) = \int f(\mathbf{s}_{\rho,t}) \frac{p(\mathbf{s}_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})}{q(\mathbf{s}_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})} q(\mathbf{s}_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}}) d\mathbf{s}_{\rho,t}.$$

Consequently, if one draws Q samples, $\mathbf{s}^i_{\rho,t},\ i = 1, \cdots, Q$, from the proposal distribution $q(\mathbf{s}_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})$, the conditional expectation is in turn approximated by

$$\mathbb{E}(f(\mathbf{s}_{\rho,t})|\mathbf{x}_{\mathcal{T}_{\rho,0:t}}) \approx \frac{1}{Q} \sum_{i=1}^{Q} f(\mathbf{s}^i_{\rho,t}) \frac{p(\mathbf{s}^i_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})}{q(\mathbf{s}^i_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})}, \tag{13}$$

and by further approximating $Q \approx \sum_{i=1}^{Q} \frac{p(\mathbf{s}^i_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})}{q(\mathbf{s}^i_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})}$ we have that

$$\mathbb{E}(f(\mathbf{s}_{\rho,t})|\mathbf{x}_{\mathcal{T}_{\rho,0:t}}) \approx \frac{\sum_{i=1}^{Q} f(\mathbf{s}^i_{\rho,t}) \frac{p(\mathbf{s}^i_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})}{q(\mathbf{s}^i_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})}}{\sum_{i=1}^{Q} \frac{p(\mathbf{s}^i_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})}{q(\mathbf{s}^i_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})}}. \tag{14}$$

Defining

$$w^i_{\rho,t} \propto \frac{p(\mathbf{s}^i_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})}{q(\mathbf{s}^i_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})} \tag{15}$$

to be the ith weight which corresponds to the ith particle $\mathbf{s}^i_{\rho,t}$, the conditional expectation is approximated by

$$\mathbb{E}(f(\mathbf{s}_{\rho,t})|\mathbf{x}_{\mathcal{T}_{\rho,0:t}}) \approx \sum_{i=1}^{Q} w^i_{\rho,t} f(\mathbf{s}^i_{\rho,t}).$$

The particle filter is an importance sampling with a special importance density $q(\cdot)$. To identify this density $q(\cdot)$, one takes into account that the conditional distribution, $p(\mathbf{s}_{\rho,0:t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})$, can be written

$$p(\mathbf{s}_{\rho,0:t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}}) \propto p(\mathbf{x}_{\mathcal{T}_{\rho,t}}|\mathbf{s}_{\rho,t})p(\mathbf{s}_{\rho,t}|\mathbf{s}_{\rho,t-1})p(\mathbf{s}_{\rho,0:t-1}|\mathbf{x}_{\mathcal{T}_{\rho,0:t-1}}), \tag{16}$$

where $\propto$ denotes that the two probability density functions (pdfs) on the right and left hand sides will be equal after appropriate scaling with a constant not dependent on the unknown state. Furthermore, one may consider that the importance density is factorized [2] such that

$$q(\mathbf{s}_{\rho,0:t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}}) = q(\mathbf{s}_{\rho,t}|\mathbf{s}_{\rho,0:t-1}, \mathbf{x}_{\mathcal{T}_{\rho,0:t}})q(\mathbf{s}_{\rho,0:t-1}|\mathbf{x}_{\mathcal{T}_{\rho,0:t-1}}). \tag{17}$$

However, only a filtering estimate is propagated at each time step. Therefore, the importance density $q(\mathbf{s}_{\rho,t}|\mathbf{s}_{\rho,0:t-1}, \mathbf{x}_{\mathcal{T}_{\rho,t}})$ depends only on $\mathbf{s}_{\rho,t-1}$ and $\mathbf{x}_{\mathcal{T}_{\rho,t}}$ which yields that $q(\mathbf{s}_{\rho,t}|\mathbf{s}_{\rho,0:t-1}, \mathbf{x}_{\mathcal{T}_{\rho,t}}) = q(\mathbf{s}_{\rho,t}|\mathbf{s}_{\rho,t-1}, \mathbf{x}_{\mathcal{T}_t})$. Employing the framework of eqs. (16) and (17) into eq. (15) at each time step, we have that the weight corresponding to the ith particle can be updated by

$$w^i_{\rho,t} \propto w^i_{\rho,t-1} \frac{p(\mathbf{x}_{\mathcal{T}_{\rho,t}}|\mathbf{s}^i_{\rho,t})p(\mathbf{s}^i_{\rho,t}|\mathbf{s}^i_{\rho,t-1})}{q(\mathbf{s}_{\rho,t}|\mathbf{s}_{\rho,t-1}, \mathbf{x}_{\mathcal{T}_{\rho,t}})}, \tag{18}$$

where $w^i_{\rho,t-1} = p(\mathbf{s}^i_{\rho,0:t-1}|\mathbf{x}_{\mathcal{T}_{\rho,0:t-1}})/q(\mathbf{s}^i_{\rho,0:t-1}|\mathbf{x}_{\mathcal{T}_{\rho,0:t-1}})$. A popular choice for the density $q(\mathbf{s}^i_{\rho,t}|\mathbf{s}^i_{\rho,t-1}, \mathbf{x}_{\mathcal{T}_{\rho,t}}) = p(\mathbf{s}^i_{\rho,t}|\mathbf{s}^i_{\rho,t-1})$ such that the weights are given by

$$w^i_{\rho,t} \propto w^i_{\rho,t-1} p(\mathbf{x}_{\mathcal{T}_{\rho,t}}|\mathbf{s}^i_{\rho,t}) \; ; i = 1, \ldots, Q \tag{19}$$

where $\mathbf{s}^i_{\rho,t}$ is the ith sample from the Markov transition density $p(\mathbf{s}^i_{\rho,t}|\mathbf{s}^i_{\rho,t-1})$ and $p(\mathbf{x}_{\mathcal{T}_{\rho,t}}|\mathbf{s}^i_{\rho,t})$ the corresponding likelihood function associated with the measurements $\mathbf{x}_{\mathcal{T}_{\rho,t}}$. Consequently, the posterior filtering density is approximated by

$$p(\mathbf{s}_{\rho,t}|\mathbf{x}_{\mathcal{T}_{\rho,0:t}}) \approx \sum_{i=1}^{Q} w^i_{\rho,t} \delta(\mathbf{s}_\rho(t) - \mathbf{s}^i_{\rho,t}),$$

where the weights are defined in (19) and δ is the Dirac delta function.

One may easily conclude that the particle filter's implementation is straightforward and can be adapted for different problems as long as the algorithm is tuned according to the specific dynamics. Precisely this has led to the particle filter algorithm's increased popularity [11]. However, it has been shown, in [15] for example, that particle filter suffers from degeneracy and that it needs a lot of samples in order to describe accurately the conditional density $p(\mathbf{s}_\rho(t)|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})$. Therefore several resampling schemes have been incorporated in order to alleviate this problem, e.g. see [15], [23], [24], [35], [47] by replicating samples with significant weight and disregarding the rest. In our paper, we used the popular multinomial resampling scheme, e.g. see [15]. After the resampling stage, the particles $\mathbf{s}^i_{\rho,t}$, are used to estimate the

state for target ρ at time instant t. Of course there are multiple targets present in the field, thus in the next section it is described how the sensor-target distributed association algorithm in Sec. III can be combined with PF to track effectively multiple-targets.

V. Joint Sensor-Target Association and Particle Filtering

Here it is described in detail how the PF unit in Sec. IV and the sensor-target association scheme in Sec. III interact to enable tracking using only target-informative sensors in the SN.

Specifically, during a start-up stage each sensor acquires T_s measurements, namely $\{x_j(\tau)\}_{\tau=-(T_s-1)}^{0}$. It is assumed that the sampling rate is fast enough such that the present targets, say $r(0)$ in number, are essentially stationary/immobile. The T_s acquired data are then used by the distributed sensor-target association framework in Sec. III to determine the sets of informative sensors $\{\mathcal{T}_{\rho_\ell^0,0}\}_{\ell=1}^{\hat{r}(0)}$ where each $\rho_\ell^0 \in \{1,\ldots,R\}$ for $\ell = 1,\ldots,\hat{r}(0)$, and $\hat{r}(0)$ is the estimated number of $r(0)$ sensed targets at time $t=0$. One sensor in each set $\mathcal{T}_{\rho_\ell^0,0}$ is designated as a leading sensor $C_{\rho_\ell^0,0}$ which collects from all sensors $j \in \mathcal{T}_{\rho_\ell^0,0}$ their corresponding measurements $x_j(0)$ and their position $\mathbf{p}_j$ for $j \in \mathcal{T}_{\rho_\ell^0,0}$ and $\ell = 1,\ldots,\hat{r}(0)$. During initialization the leading sensor $C_{\rho_\ell^0,0}$ can be selected randomly among the sensors in $\mathcal{T}_{\rho_\ell^0,0}$. Then, for time $t > 0$ it will be described later on how the leading sensors are selected. Each leading sensor $C_{\rho_\ell^0,0}$, for $\ell = 1,\ldots,\hat{r}(0)$, then calculates the 'average' informative sensors' position as

$$\hat{\mathbf{p}}_{\rho_\ell^0}(0) = \sum_{j \in \mathcal{T}_{\rho_\ell^0,0}} \mathbf{p}_j, \quad \ell = 1,\ldots,\hat{r}(0). \tag{20}$$

Then, each leading sensor $C_{\rho_\ell^0,0}$ uses the corresponding average location in (20) to initialize the PF recursions in Sec. IV, and find a state estimate $\hat{\mathbf{s}}_{\rho_\ell^0}(0)$ for target ρ_ℓ^0 using the informative measurements $x_j(0)$, for $j \in \mathcal{T}_{\rho_\ell^0,0}$ and $\ell = 1,\ldots,\hat{r}(0)$.

Suppose that at time t each leading sensor $\{C_{\rho_\ell,t}\}$ has available state estimates $\hat{\mathbf{s}}_{\rho_\ell}(t)$ for $\ell = 1,\ldots,\hat{r}(t)$. From $\hat{\mathbf{s}}_{\rho_\ell}(t)$ the estimated target position $\hat{\mathbf{p}}_{\rho_\ell}(t)$ can be extracted and it is utilized to select a set of 'candidate' target-informative sensors, namely $\mathcal{J}_{\rho_\ell,t+1}$, for target ρ_ℓ. Specifically, the leading sensor $C_{\rho_\ell,t}$ transmits $\hat{\mathbf{s}}_{\rho_\ell}(t)$ to its single-hop neighbors, which will subsequently transmit to their own neighbors and the estimate propagates in time. A sensor j that receives $\hat{\mathbf{s}}_{\rho_\ell}(t)$ will forward this estimate only to those neighbors in $j' \in \mathcal{N}_j$ that are located within a radius R_s from the estimated target location, i.e., $\|\mathbf{p}_{j'} - \hat{\mathbf{s}}_{\rho_\ell}(t)\|_2 \leq R_s$. Note that through the aforementioned process the set of sensors $\mathcal{J}_{\rho_\ell,t+1}$ selected at time $t+1$ has the following two properties: i) each sensor $j \in \mathcal{J}_{\rho_\ell,t}$ is located within a radius R_s of the estimated position $\hat{\mathbf{p}}_{\rho_\ell}(t)$, i.e., $\|\mathbf{p}_j - \hat{\mathbf{p}}_{\rho_\ell}(t)\| \leq R_s$; and ii) sensors in $j \in \mathcal{J}_{\rho_\ell,t+1}$ form a connected

communication subgraph characterized by the adjacency matrix $\mathbf{E}_{\rho_\ell,t+1}$ which is going to be a submatrix of $\mathbf{E}$ after keeping the rows and columns with indices in $\mathcal{J}_{\rho_\ell,t+1}$.

In each of the subsets $\mathcal{J}_{\rho_\ell,t+1}$ the distributed targets-to-sensors data association algorithm (Alg. 1) is employed to determine the target-informative sensor subsets $\mathcal{T}_{\rho_\ell,t+1} \subseteq \mathcal{J}_{\rho_\ell,t+1}$ for each of the targets ρ_ℓ at time instant $t+1$. The radius R_s through which $\mathcal{J}_{\rho_\ell,t+1}$ are constructed is up to our control, and the faster the target moves the larger R_s should be set to guarantee that all target-informative sensors are included in $\mathcal{J}_{\rho_\ell,t+1}$. Performing the sensor-target association algorithm in different sensor subsets $\mathcal{J}_{\rho_\ell,t+1}$ of the SN facilitates tracking the present targets, while it requires less computational and communication complexity than when applied in the whole SN. In fact, there will also be instances where the sensors-targets association algorithm (Alg. 1) will be implemented across the whole SN whenever it is detected that the present targets may have changed in number and we have to redetermine the target-informative subsets. Indicators used to determine when to apply the scheme in Sec. III across the whole SN are the following:

C.1 If any of the estimated target-informative sets $\mathcal{T}_{\rho_\ell,t+1}$, returned by Algorithm 1, are empty. This implies that most likely some of targets being tracked at previous time instances are not present in the sensed field anymore.

C.2 If the energy of the measurements of a given sensor, not currently in any set $\mathcal{T}_{\rho_\ell,t}$ exceeds a certain threshold. This implies that most likely a new target has entered the sensed field and this is indicated by an elevated energy level in the measurements of a currently non-informative sensor.

The procedure described earlier for the start-up stage is applied every time it is determined that the distributed sensors-targets association scheme has to be applied in the whole network. This process is necessary due to the fact that the target population has changed since old targets may not be sensed anymore, while new ones may have entered the sensed field.

The leading sensor $C_{\rho_\ell,t+1}$ is chosen as that sensor in $\mathcal{T}_{\rho_\ell,t+1}$, which is closest to the estimated position of the ρ_ℓth target, i.e.,

$$C_{\rho_\ell,t+1} = \arg \min_{j \in \mathcal{T}_{\rho_\ell,t+1}} \|\mathbf{p}_j - \hat{\mathbf{p}}_{\rho_\ell}(t)\|_2. \tag{21}$$

The process of electing a new leading sensor can take place among the sensors in $\mathcal{T}_{\rho_{\ell,t+1}}$ that can determine their distance from $\hat{\mathbf{p}}_{\rho_\ell}(t)$ and find which sensor has the minimum in a distributed fashion, e.g., see [21]. The leading sensor $C_{\rho_\ell,t+1}$ then collects i) the corresponding state particles and weights $\{\mathbf{s}^i_{\rho_\ell,t}, w^i_{\rho_\ell,t}\}_{i=1}^Q$ from $C_{\rho_\ell,t}$; and ii) the sensors measurements $x_j(t+1)$ for $j \in \mathcal{T}_{\rho_\ell,t+1}$, namely the updated informative sensor subset for target ρ_ℓth at time instant $t+1$. Assuming that the sampling rate is fast enough, the target

locations $\mathbf{s}_{\rho_\ell}(t)$ and $\mathbf{s}_{\rho_\ell}(t+1)$ will be close, thus a few communication hops (1 or 2 in simulations) suffice to have the previous leading sensor $C_{\rho_\ell,t}$, as well as the current target-informative sensors in $\mathcal{T}_{\rho_\ell,t+1}$ transmit their information to $C_{\rho_\ell,t+1}$. For instance, in Fig. 2 sensor 4 corresponds to one leading sensor at time instant t, while the informative sensors are $\{1,2,3,4\}$. Then, at time instant $t+1$ and since the target has moved the informative sensor set changes to $\{5,6,8\}$ while sensor 6 is elected as the new leading sensor at $t+1$ being closer to the estimated target position.

The leading sensor $C_{\rho_\ell,t+1}$ proceeds to draw new state particles from the importance sampling pdf $q(\mathbf{s}_\rho(t)|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})$ and update their corresponding weights as in (19). Then, $C_{\rho_\ell,t+1}$ forms the new state estimate $\hat{\mathbf{s}}_{\rho_\ell}(t+1) \approx E[\mathbf{s}_{\rho_\ell}(t+1)|\mathbf{x}_{\mathcal{T}_{\rho,0:t}}]$ using (13), and extract from $\hat{\mathbf{s}}_{\rho_\ell}(t+1)$ the estimated location for target ρ_ℓ at time instant t, namely $\hat{\mathbf{p}}_{\rho_\ell}(t+1)$. The leading sensor $C_{\rho_\ell,t+1}$ transmits $\hat{\mathbf{s}}_{\rho_\ell}(t+1)$ to its single-hop neighbors and the process described earlier is repeated to update the subsets of candidate informative sensors $\mathcal{J}_{\rho_\ell,t+2}$. The joint algorithmic framework for multi-target tracking and distributed sensor-target association is tabulated as Algorithm 2.

Algorithm 2 Joint Target-Sensor Association and Multi-Target Tracking

1: **Start-up stage ($t=0$)/Reconfiguration ($t\neq 0$):** Each sensor j collects T_s measurements $x_j(t)$ and Algorithm 1 is applied in the whole network to determine the target-informative groups $\mathcal{T}_{\rho_\ell,t}$ and $\ell = 1,\ldots,\hat{r}(t)$, where $\hat{r}(t)$ is the estimated number of sensed targets.

2: **for** $\tau = t,\ldots,$ **do**

3: Determine the leading sensor $C_{\rho_\ell,\tau}$ in each $\mathcal{T}_{\rho_\ell,\tau}$ for $\ell = 1,\ldots,\hat{r}(t)$ as specified in (21).

4: Each leading sensor $C_{\rho_\ell,\tau}$ receives particles and weights from $C_{\rho_\ell,\tau-1}$, and $x_j(\tau)$ from $j \in \mathcal{T}_{\rho_\ell,\tau}$ to perform tracking for $\rho_\ell = 1,\ldots,\hat{r}(t)$ target via the PF recursions and obtain $\hat{\mathbf{s}}_{\rho_\ell}(\tau)$ via (13).

5: The state estimates $\hat{\mathbf{s}}_{\rho_\ell}(\tau)$ are propagated from $C_{\rho_\ell,\tau}$ via single-hop transmissions to every sensor j that can be reached from $C_{\rho_\ell,\tau}$ by a multi-hop path and satisfies $\|\mathbf{p}_j - \hat{\mathbf{p}}_{\rho_\ell}(\tau)\|_2 < R_s$. Then, the candidate informative sets $\{\mathcal{J}_{\rho_\ell,\tau+1}\}_{\ell=1}^{\hat{r}(t)}$ are formed.

6: Algorithm 1 is applied in each connected set of sensors $\mathcal{J}_{\rho_\ell,\tau+1}$ to obtain the target-informative sets $\mathcal{T}_{\rho_\ell,\tau+1}$.

7: If either condition C.1, or C.2 is true then go to step 1, otherwise go to step 2.

8: **end for**

VI. Inter-Sensor Communication Costs

The information exchanges occurring during different steps of Algorithm 2 and the associated communication costs are outlined next. Specifically, inter-sensor communications take place when Algorithm 1 is applied to perform sensors-to-targets association and every time a leading sensor has to be updated.

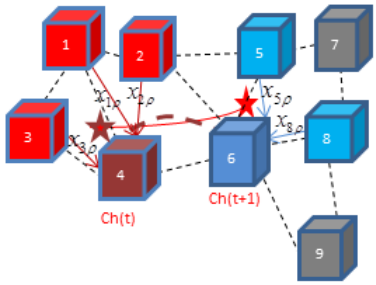

Fig. 2. Update of the target-informative sets and the leading sensors as a target moves in the sensed field.

In detail at time instant $t+1$ sensor j has to receive $|\mathcal{N}_j|$ scalar measurements from its neighbors, namely $\{x_{j'}(t+1)\}_{j'\in\mathcal{N}_j}$, to update $\hat{\mathbf{\Sigma}}_{x,t+1}(j,j')$ (step 1 in Alg. 1). Further, sensor j receives the updates $\{\hat{\mathbf{M}}^{k-1}_{t+1}(j',\ell)\}^{L}_{\ell=1}$, $L|\mathcal{N}_j|$ scalars in total, to form its local updates $\{\hat{\mathbf{M}}^{k}_{t+1}(j,\ell)\}^{L}_{\ell=1}$ (step 8 in Alg. 1). Thus, sensor j receives $(L+1)|\mathcal{N}_j|$ scalars. In the same way sensor j has to transmit $x_j(t+1)$ and $\{\hat{\mathbf{M}}^{k-1}_{t+1}(j,\ell)\}^{L}_{\ell=1}$, a total of $L+1$ scalars, to its neighbors per iteration k in Alg. 1. It is worth emphasizing that the communication complexity for each sensor is linear with respect to its neighborhood size and the number L used to bound the number of targets at a given time instant. Such a complexity can be handled easily by networks of sensors. Note that the previous information exchanges occur during step 1 and step 6 in Alg. 2.

Every time the target-informative sets $\{\mathcal{T}_{\rho,t+1}\}$ are updated, the old leading sensor, say $C_{\rho,t}$, has to send to the new leader $C_{\rho,t+1}$ Q particles, each entailing $2K$ scalars, and Q corresponding scalar weights. Further, $C_{\rho,t}$ has to send out the $2K$ scalars corresponding to the estimate $\hat{\mathbf{s}}_\rho(t)$ (step 5 in Alg. 2). Thus, $C_{\rho,t}$ has to transmit $(2K+1)Q+2K$ scalars for implementing steps $3-5$ in Alg. 2. Every sensor in $\mathcal{T}_{\rho,t+1}$ has to also transmit its scalar measurement $x_j(t+1)$ that will reach $C_{\rho,t+1}$, and will be used to update the target state estimates. Thus, $C_{\rho,t+1}$ receives in total $(2K+1)Q+2|\mathcal{T}_{\rho,t}|$ scalars. Clearly, the communication cost is proportional to the number of particles Q and the dimensionality of the target state vectors $2K$.

In [29] a Monte Carlo method is employed to select the active sensors when tracking a *single* target. Specifically, at time t there is a cluster head sensor, say sensor j, that selects among its neighbors, say $\mathcal{N}_j$, a number of μ sensors whose measurements will be used to track, via EKF, the target at the next time

instance. A probability of detection is calculated for each sensor in the neighborhood $\mathcal{N}_j$ by the cluster head, and the first active sensor is selected as the one in $\mathcal{N}_j$ having the highest probability of detection. This task is carried out at a computational complexity of $\mathcal{O}(|\mathcal{N}_j|)$. In the same way the second active sensor is selected, as the one among the remaining $|\mathcal{N}_j|-1$, achieving the highest value of a pertinent joint probability of detection metric; the corresponding complexity is $\mathcal{O}(|\mathcal{N}_j|-1)$. Similarly the ith active sensor is determined at a complexity of $\mathcal{O}(|\mathcal{N}_j|-i)$ for $i=1,\ldots,\mu$. Thus, the total complexity at the cluster head for selecting μ sensors is $\mathcal{O}(\mu(|\mathcal{N}_j|-\frac{\mu-1}{2}))$. Depending on the number of active sensors chosen ($1\leq\mu\leq|\mathcal{N}_j|$), if $\mu=1$ then complexity is in the order of $\mathcal{O}(|\mathcal{N}_j|)$ per cluster head which is the same as in our scheme for $r=1$ target. If $\mu=|\mathcal{N}_j|$ (all sensors selected), then the computational complexity of [29] is in the order of $\mathcal{O}(|\mathcal{N}_j|^2/2)$ which leads to a higher complexity compared to the one achieved in our case when $|\mathcal{N}_j|>2$. Further, in [29], all neighboring sensors send their data to the current leading sensor, leading to a communication cost which is proportional to the neighborhood size $|\mathcal{N}_j|$.

The approach for choosing active sensors in [29] relies on the prediction state estimate and MSE covariance obtained through extended Kalman filtering (EKF), see, e.g., [25]. The latter two quantities are used in a Gaussian pdf to evaluate the aforementioned probabilities of detection whose values will determine the active sensors. Different from [29], the proposed sensors-to-targets association scheme does not depend on the state and observation model parameters in (1) and (22). On the contrary, the novel tracking scheme here relies on the sensor measurements to update the target-informative portion of the SN, and it is not affected by the tracking algorithm [cf. (9) and (10)]. Linearization in EKF may result errors in the tracking process which can propagate to the sensor selection process in [29] and deteriorate performance. In the same way, selecting the closest sensors to the estimated target position is prone to error propagation and cannot perform better than [29]. Numerical tests will corroborate the previous claims.

VII. Numerical Tests

A. Target Dynamics and Particle Sampling

As in the majority of methods developed for target tracking, e.g., [29] in the numerical tests we consider a scenario where the targets move according to a near constant velocity model [3]. Specifically, the ρth target's state vector evolves according to the following model

$$\mathbf{s}_\rho(t+1)=\mathbf{A}\mathbf{s}_\rho(t)+\mathbf{u}_\rho(t), \tag{22}$$

where $\mathbf{A}$ is a $2K \times 2K$ transition matrix, while $\mathbf{u}_\rho(t)$ denotes zero-mean Gaussian noise with covariance $\mathbf{\Sigma}_u$. The matrices $\mathbf{A}$ and $\mathbf{\Sigma}_u$ have the following structure (e.g., see [3])

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \dots & \Delta T & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 1 & 0 & 0 & \dots & \Delta T \\ 0 & 0 & 1 & \dots & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \qquad \mathbf{\Sigma}_u = \sigma_u^2 \begin{bmatrix} (\Delta T)^3/3 \cdot \mathbf{I}_K & (\Delta T)^2/2 \cdot \mathbf{I}_K \\ (\Delta T)^2/2 \cdot \mathbf{I}_K & \Delta T \cdot \mathbf{I}_K \end{bmatrix}$$

where ΔT is the sampling period, and σ_u^2 is a nonnegative constant controlling the variance of the noise entries in $\mathbf{u}_\rho(t)$ while $\mathbf{I}_K$ denotes the $K \times K$ identity matrix. The state noise is assumed to be temporally white and uncorrelated with the observation noise across sensors, namely $\mathbf{w}_t := [w_1(t), \dots, w_m(t)]^T$.

Using the state transition model in (22) it follows readily that state transition pdf $p(\mathbf{s}_\rho(t)|\mathbf{s}^i_{\rho,t-1})$ is Gaussian with expectation $\mathbf{A}\mathbf{s}^i_{\rho,t-1}$ and covariance $\mathbf{\Sigma}_u$. Thus, the new Q state particles at time instant t can be generated from the ones obtained at time instant $t-1$ as follows:

$$\mathbf{s}^i_{\rho,t} = \mathbf{A} \times \mathbf{s}^i_{\rho,t-1} + \mathbf{v}_t, \;\; i = 1, \dots, Q \tag{23}$$

where $\mathbf{v}_t \in \mathbb{R}$ is a $2K \times 1$ zero-mean Gaussian vector with covariance $\mathbf{\Sigma}_u$.

From the observation model in (1) it turns out that the likelihood pdf of the informative observations corresponding to the sensors in $\mathcal{T}_{\rho,t}$, given the ith particle for the state of target ρ, i.e., $p(\mathbf{x}_{\mathcal{T}_{\rho,t}}|\mathbf{s}^i_{\rho,t})$ in (19) is Gaussian, i.e.,

$$p(\mathbf{x}_{\mathcal{T}_{\rho,t}}|\mathbf{s}^i_{\rho,t}) = \frac{1}{(2\pi\sigma_w^2)^{|\mathcal{T}_{\rho,t}|/2}} \exp\left(\frac{-\left\|\mathbf{x}_{\mathcal{T}_{\rho,t}} - \mathbf{d}_\rho(\mathbf{s}^i_{\rho,t})\right\|^2}{2\sigma_w^2} \right) \tag{24}$$

where σ_w^2 denotes the observation noise variance, while

$$\mathbf{d}_\rho(\mathbf{s}^i_{\rho,t}) = \left[\|\mathbf{p}_{j_1^t} - \mathbf{p}^i_{\rho,t}\|^{-2}, \|\mathbf{p}_{j_2^t} - \mathbf{p}^i_{\rho,t}\|^{-2}, \dots, \|\mathbf{p}_{j^t_{|\mathcal{I}_{\rho,t}|}} - \mathbf{p}^i_{\rho,t}\|^{-2} \right]^T \tag{25}$$

where $\mathbf{p}_{j_1^t}$ is the known position of sensor j_1 at time t, while $j_1^t, j_2^t, \dots, j^t_{|\mathcal{I}_{\rho,t}|}$ are the indices of the sensors in the informative set $\mathcal{T}_{\rho,t}$. Further, $\mathbf{p}^i_{\rho,t}$ is the estimated position for target ρ extracted from the state particle $\mathbf{s}^i_{\rho,t}$ for $i = 1, \dots, Q$. Intuitively, the vector $\mathbf{d}_\rho(\mathbf{s}^i_{\rho,t})$ in (25) can be viewed as an estimate (prediction) of the informative sensor measurements $\mathbf{x}_{\mathcal{T}_{\rho,t}}$ using the most recent particles. In this way, only the measurements from the informative sensors in $\mathcal{T}_{\rho,t}$ will be used to find the weights for the sampled particles, and thus track the corresponding target ρ.

B. Tracking of a Single-Target

We start by testing the performance of the novel tracking Alg. 2 and compare to existing alternatives in a wireless network setting with $m = 150$ sensors which are randomly placed in a region $[0, 100] \times [0, 100]m^2$. A scenario with a single target is considered first. The target starts at location $[27.00, 72.00]$ and moves with a speed of $1.8m/s$ at the x-axis and the y-axis. The state and observation models introduced in (22) and (1) are utilized here for $K = 2$.

In the following numerical tests we compare the tracking performance, via the localization root mean-square error (RMSE), among i) the novel Alg. 2; ii) the EKF approach with sensor selection in [29]; iii) EKF combined with the sensor-targets association Alg. 1 (EKF+Alg. 1); iv) EKF with all sensor measurements used and there is no sensors-targets association (EKF-All sensors); v) PF with all sensor measurements used and there is no sensors-targets association (PF-All sensors); vi) PF combined with a scheme that selects as target-informative sensors the J-nearest sensors to the current target position estimate (Nearest sensor) with $J = 3$; and vii) unscented Kalman filtering (UKF) (see e.g., [22]) combined with the sensors-to-targets association Alg. 1 (UKF+Alg. 1).

In all these tracking methods, the target position is initialized by applying Alg. 1 and finding (20), ensuring that the initial error is the same for all different tracking approaches. As for the parameters in Alg. 2, the forgetting factor for updating (6) is set to $\gamma = 0.1$, while the radius R_s, used in forming the candidate sets $\mathcal{J}_{\rho,t}$, is set equal to 15. The threshold to decide which entries in $\hat{\mathbf{M}}_t$ are zero and which are nonzero is set to be 10^{-5} in the single-target case. The parameters λ_ℓ are set using the method in [44, Sec. V-A], and $\phi = 1.1$. Fig. 3 displays the root mean-square tracking error (RMSE) achieved by the tracking schemes described earlier versus time t. Four different test cases are considered in Fig. 3 in which the state noise and the measurement noise variances change. On the top left diagram a setting with relatively small state and observations noise variances, namely $\sigma_u^2 = 0.005$ and $\sigma_w^2 = 0.0001$ (corresponding to a sensing signal-to-noise ratio of 30dB), is considered. The bottom left diagram corresponds to a setting with relatively small state variance, and relatively large observation noise variance, namely $\sigma_u^2 = 0.005$ and $\sigma_w^2 = 0.1$ (corresponding to a sensing signal-to-noise ratio of 2dB). The top right diagram deals with a setting with relatively large state variance, and relatively small observation noise variance, namely $\sigma_u^2 = 0.05$ and $\sigma_w^2 = 0.0001$ (corresponding to a sensing signal-to-noise ratio of 30dB). Finally, the bottom right diagram corresponds to a setting where both the state and observation noise variances are relatively large, i.e., $\sigma_u^2 = 0.005$ and $\sigma_w^2 = 0.1$ (corresponding to a sensing signal-to-noise ratio of 2dB which is low).

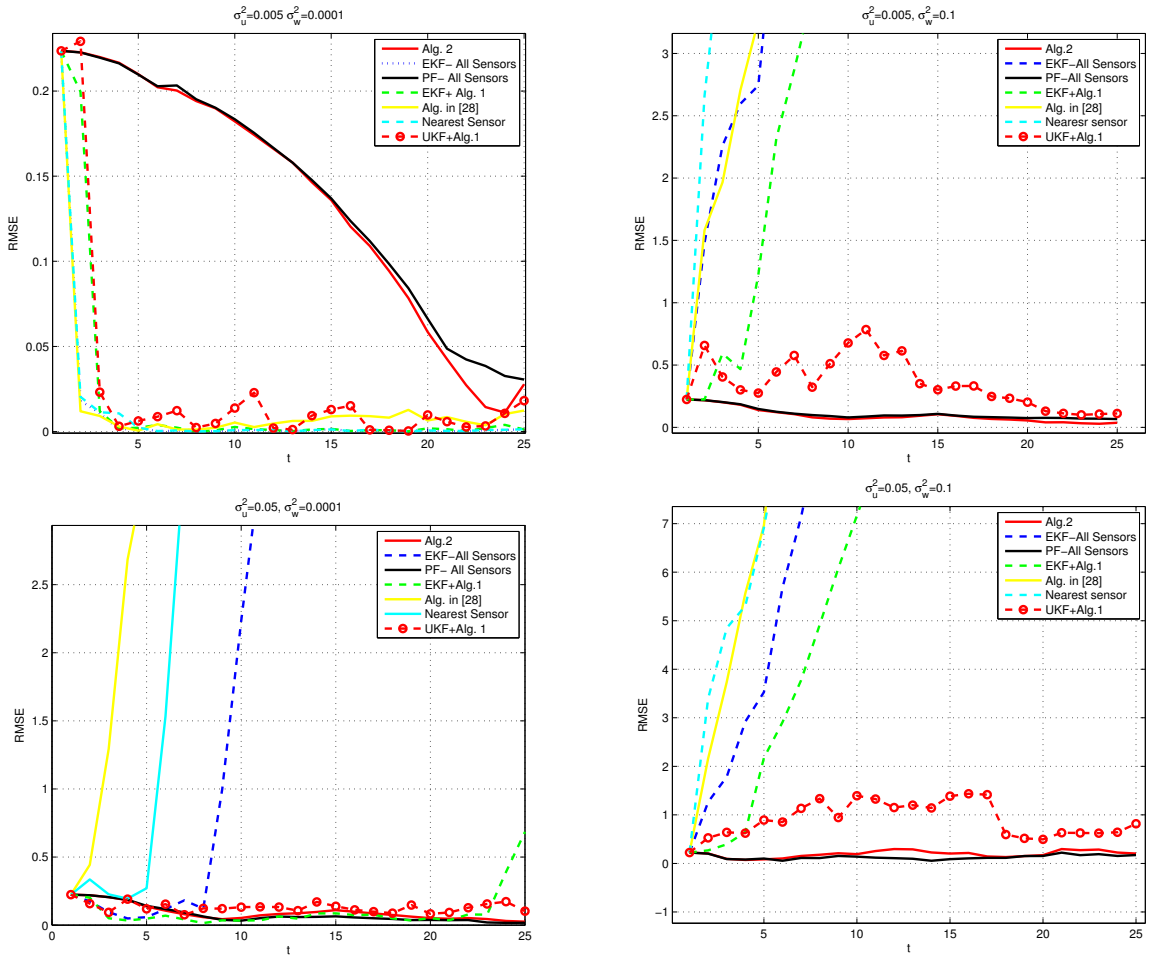

Fig. 3. Tracking root mean-square error (RMSE) vs. time (t) for different tracking schemes and different tracking conditions.

Fig. 3 corroborates that the novel tracking Alg. 2 outperforms in terms of tracking RMSE alternative tracking approaches when the observation and/or state noise variance are relatively high. The EKF-based approaches rely on linearization and therefore the presence of high-variance noise in the state and/or observation models will result such approaches to deviate as time progresses. The method in [29] utilizes the EKF target position estimates to determine the target informative sensors and as it can be seen in Fig. 3 the RMSE will diverge. The reason for the big performance gap between the EKF and Alg. 2 lies on the linearization errors involved in EKF. Such errors are accentuated in the presence of state and measurement noise of high variance. Then, bad trajectory estimates further deteriorate the sensor selection process causing an error propagation behavior. The same behavior is also exhibited when the

nearest to the estimated target position sensor are selected to perform tracking. This method also diverges in the presence of high variance noises. Further, it can be seen that UKF combined with Alg. 1 performs better than EKF+Alg. 1 as expected, however its performance is still worse than Alg. 2. This is expected since UKF aims to approximate the mean and covariance of the state, whereas the particle filters in fact track the posterior pdf which in our setting is not Gaussian despite the presence of Gaussian noise.

Alg. 2 performs reliable tracking even under high-variance noise environments. The reason is that the sensors-to-targets association scheme (Alg. 1) does not rely on the tracking algorithm and employs the data directly to determine target-informative sensors. Another important property of Alg. 2 is that its tracking performance is very close to that of a particle filtering approach that uses all the sensors in the network. This demonstrates the efficiency of Alg. 2 in selecting a few target-informative sensors without compromising the tracking performance. Clearly, Alg. 2 has the potential to prolong the lifetime of the network without losing much tracking accuracy when compared to a setting where all sensors measurements are utilized during tracking. When the state and observation noise variances are small, then all six different methods perform well and reach a small tracking RMSE. In such a setting the PF-based methods appear to reach steady-state at a slower rate than the EKF-based and UKF-based approaches. The reason is the small number of particles used here $Q = 100$ per time instant t that results slower convergence. Nonetheless, what is important is that the novel tracking Alg. 2 is able to accurately track the target even in hostile settings suffering from high variance noises.

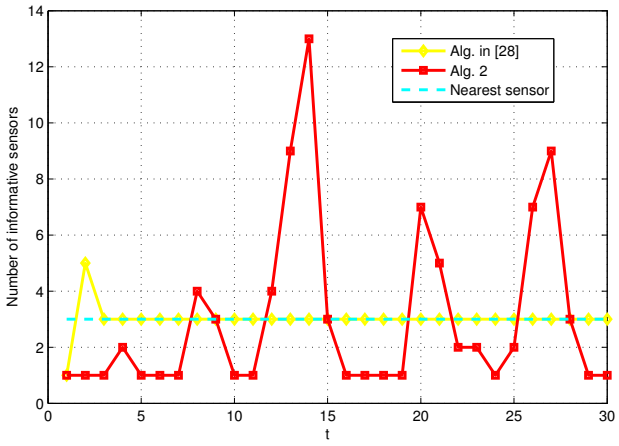

Fig. 4. Number of selected sensors vs. time t in a single-target setting.

Next, we consider the same setting as in the upper diagram of Fig. 3 where the state and observation noise variances, $\sigma_u^2 = 0.005$ and $\sigma_w^2 = 0.0001$, are relatively small. Fig. 4 depicts the number of sensors selected to utilize their measurements for tracking per time instant t. Alg. 2, the scheme in [29] and the nearest sensor selection rule for $J = 3$ are applied. Notice that the average number of selected sensors per method is equal to 3 ensuring a fair comparison in terms of RMSE in Fig. 3 (top). The bottomline is that for the same number of active sensors our approach outperforms existing alternatives for selecting informative sensors, while it performs closely to the benchmark (but demanding) scheme where all sensors are used to perform tracking.

C. Changing the number of particles

Fig. 5 depicts the tracking RMSE achieved by Alg. 2 versus time for different values of the number of particles Q used. Two different settings are considered: Fig. 5 (left) corresponds to a setting with relatively high state and observation noise; and Fig. 5 (right) corresponds to a setting with relatively low-variance state and observation noise. As expected the left diagram in Fig. 5 indicates worse tracking performance than the right diagram. Nonetheless, in all cases an increasing number of particles always leads to considerable improvements of the tracking performance. The reason for observing inflexions has

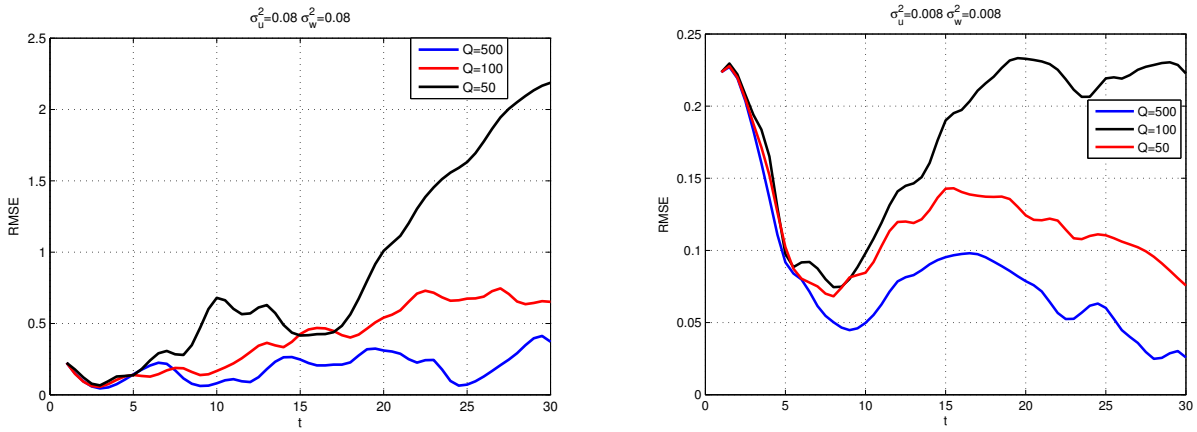

Fig. 5. Tracking RMSE vs. time t for different number of particles Q using Alg. 2.

to do with the number of sensors around the target at a give time instant. At certain time instances there may be only one sensor close to the target whereas the number of sensors may increase or decrease with

time. The more sensors in the proximity of a target, the more active sensors will be selected via Alg. 1, resulting a reduction in the tracking RMSE. However, at times if there are not many sensors close to the target the RMSE may increase in value as observed in Fig. 5.

D. Tracking of multiple targets

A setting where three targets are moving in the sensed field is considered next. Note that the scheme in [29] and nearest sensor method are not capable to track multiple targets. The same applies when using all sensors measurements and there is no sensor-data association embedded in the tracking scheme (EKF or PF). Here we will test the tracking RMSE achieved by Alg. 2, as well as the scheme where EKF is combined with Alg. 1. A setting with $m = 120$ sensors is considered that are randomly deployed in the area of $[0, 100] \times [0, 100] m^2$. The three targets are set to move from initial positions $[30, 80]$, $[35, 25]$ and $[40, 45]$, respectively. The speed of the targets per x-axis and y-axis is set equal to $1.8 m/s$. These trajectories are well separated in space and satisfy the assumption of having one dominant term in (1), as introduced in Sec. II. Step 1 in Alg. 2 is applied to associate sensors with targets and initialize Alg. 2. Fig. 6 depicts the average tracking RMSE (averaged across the three different targets) versus time for i) Alg. 2; and ii) the EKF combined with the sensors-to-targets association Alg. 1. The left diagram corresponds to a relatively low-variance state and observation noise setting, i.e., $\sigma_u^2 = \sigma_w^2 = 8 \cdot 10^{-3}$ (the corresponding sensing SNR is 13dB). The right diagram corresponds to a relatively high-variance state and observation noise setting, i.e., $\sigma_u^2 = \sigma_w^2 = 8 \cdot 10^{-2}$ (the corresponding sensing SNR is 3dB). As in

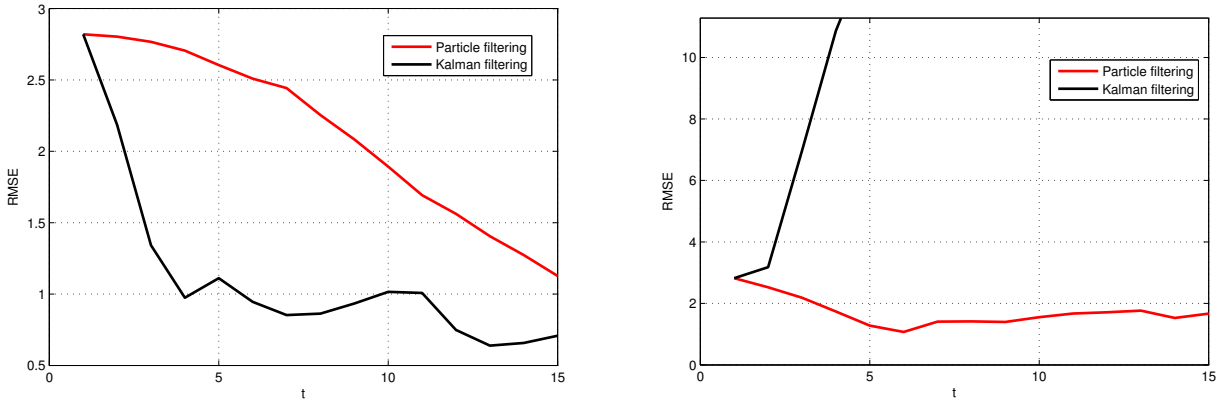

Fig. 6. Average tracking RMSE versus time for a low-variance noise setting (left); and a high-variance noise setting (right).

the single-target case it follows that both Alg. 2 and the EKF-based schemes perform accurate tracking for the low-variance noise setting. Again Alg. 2 converges slower to steady state because of the limited number of particles, $Q = 100$, used per time instant t. In the high-variance noise setting clearly Alg. 2 is still able to track the three targets whereas the EKF based method fails since the linearization process breaks down and results misleading target position estimates that eventually diverge.

Fig. 7 demonstrates how the steady-state tracking RMSE behaves as a function of the state and observation noise variance. Two curves are depicted, for the blue curve the x-axis corresponds to measurement noise variance while the state noise variance is set to $\sigma_u^2 = 0.1$, whereas for the red curve the x-axis corresponds to state noise variance while the measurement noise variance is set to $\sigma_w^2 = 0.1$. Note that although the noise variance increases almost by an order of magnitude on the x-axis, the RMSE increases approximately by 1.6 meters which in the $[0, 100] \times [0, 100]$ area corresponds to a relatively small tracking performance degradation. This advocates the robustness and 'graceful' degradation of Alg. 2 in the presence of state/measurement noise.

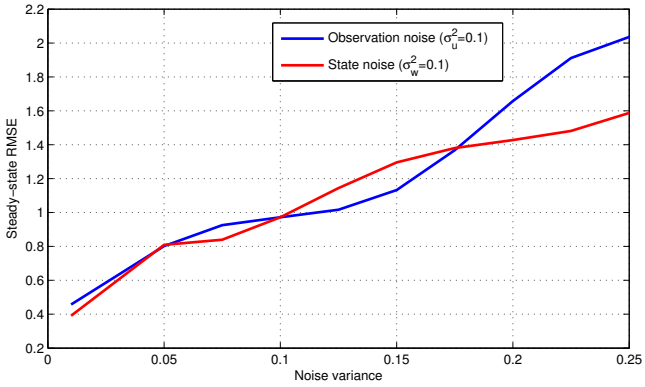

Fig. 7. Steady-state RMSE versus state/ measurement noise variance.

E. Time-varying number of targets

Next, we test the tracking performance of our proposed method in a setting where the number of targets can change in time. Again a number of $m = 120$ sensors are placed randomly in the region of $[0, 100] \times [0, 100]m^2$. The total number of targets appearing and disappearing across time is $R = 12$. In that region, five groups of different targets appear and disappear orderly. The target configuration in the

test is set as follows: Targets $\rho = 1, 2, 3$ start moving at positions $[35, 25]$, $[40, 45]$, $[20, 55]$ and follow the dynamics in (22), with a speed of $2m/s$ across the x-axis. Targets $\rho = 1, 2, 3$ move in the field for the time interval $[1, 15]$s and then are not sensed anymore. In the interval $[15, 17]$s no targets are present in the field. Then, targets $\rho = 4, 5$ start at positions $[12, 25]$, $[30, 80]$ and move according to same state model followed by the first three for the time interval $[17, 30]$s but with speed $1.5m/s$ across the x-axis. Again no targets are present during $[30, 32]$s. Then, targets $\rho = 6, 7$ show up at initial positions $[75, 35]$, $[10, 30]$ and start moving, according to (22), for the time interval $[32, 45]$s and speed $1.5m/s$ per axis. Two new targets, namely $\rho = 8, 9$, appear in initial positions $[40, 10]$, $[40, 70]$ and move in the field for the time interval $[47, 60]$s with speed $1.7m/s$ on the y-axis, and speed $1.5m/s$ across the x-axis. Finally, the last three targets $\rho = 10, 11, 12$ start at positions $[60, 20]$, $[60, 70]$ and $[70, 50]$ and move within the field for the time interval $[62, 72]$s. Target $\rho = 10$ follows the same state model as targets $\rho = 6, 7$, while targets $\rho = 11, 12$ follow the same state model as targets $\rho = 1, 2, 3$. Again, the targets are placed in the field such that at every time instant t every sensor senses one dominant target in (1).

Here we have to emphasize that when testing the novel Alg. 2 we do not know the number of targets present in the field at a given time instant, and we do not know when the target configuration changes, with old targets vanishing and new targets showing up. Alg. 2 entail steps 1 and 7 that detect when a change in the targets' configuration may have happened and estimate the number of targets present as well as the sensors acquiring informative observations for the different targets present in the field. Thus, Alg. 2 does not really have available the time period for which each target is active; this is something that it estimates. This is to be contrasted with the sensor selection framework in [44], where the number of sources/targets is known and fixed. The parameter L in Alg. 1 is set to 4, which indicates that at every time instant the number of targets present in the field will not exceed 4. The radius R_s for determining the candidate informative sensors subsets $\mathcal{J}_{\rho,t}$ is set equal to $R_s = 10$. The forgetting factor is set $\gamma = 0.1$. The state noise variance is set as $\sigma_u^2 = 0.1$, while the measurement noise variance is also set to $\sigma_w^2 = 0.1$ (which amounts to an observation SNR of roughly 10dB).

Fig. 8 depicts the true target trajectories (blue dashed curves), along with the estimated trajectories from Alg. 2 (light green curves). The blue stars correspond to the starting position of the targets and the red stars denote the ending position. Clearly, Alg. 2 is able to carry out accurate tracking of all $R = 12$ targets. Another interesting property shown in Fig. 8 is the small number of sensors selected by Alg. 1 to be utilized in the tracking process. The red circles in Fig. 8 depict the target informative sensors at time instances 15s, 30s, 45s, 60s and 72s, thus all the red circles correspond to the cumulative number of informative sensors throughout the simulation. As it can be seen the informative sensors are

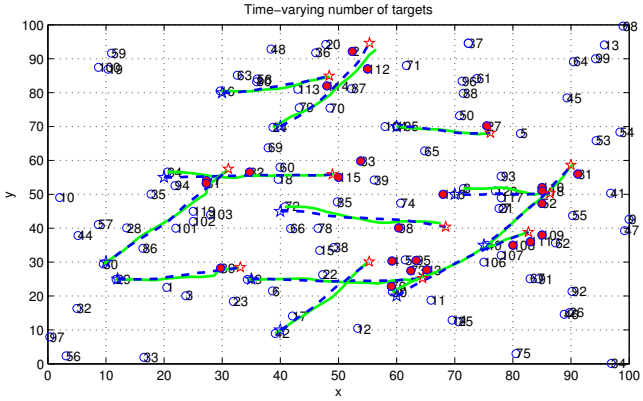

Fig. 8. Tracking of multiple targets in a setting with time-varying number of targets. Blue dashed curves indicate the true target trajectories, while the light green curves correspond to the estimated trajectories. The red circles represent the active sensors.

selected such that they are closely located to a corresponding target. This further implies that Alg. 2 performs efficient tracking by utilizing only a small portion of the sensors available in the network. It is worth mentioning that only a small portion of the network is used to gather data. In Fig. 9 it is depicted where the target is (red star) and what sensors are active during time-instances 45s and 72s. Clearly the active sensors are in the vicinity of the targets' location corroborating the capability of Alg. 1 to utilize the sparse covariance sensing data structure and select those sensors acquiring informative measurements about the present targets. Another tracking scenario where there are many trajectory crossings, the targets

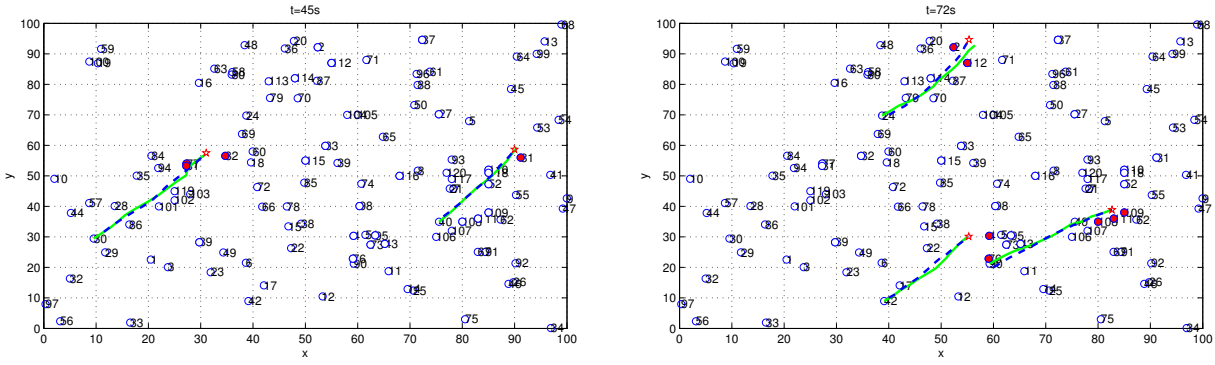

Fig. 9. Active sensors and position of targets at time instances (left) $t = 45$s; and (right) $t = 72$s. The red circles represent the active sensors.

are moving at different directions and different speeds is depicted in Fig. 10. The noise remains the same

as before and there are $R = 12$ targets in total. Specifically, targets $\rho = 1, 2, 3$ start moving at positions $[35, 25]$, $[40, 45]$, $[20, 55]$ and follow the dynamics in (22). While targets $\rho = 1, 3$ move at a speed of $2m/s$ across the x-axis, target $\rho = 2$ moves with a speed of $2m/s$ across the y-axis. Targets $\rho = 1, 2, 3$ move in the field for the time interval $[1, 15]$s and then are not sensed anymore. In the interval $[15, 17]$s no targets are present in the field. Then, targets $\rho = 4, 5$ start at positions $[23, 40]$, $[50, 75]$ and move according to same state model followed by the first three for the time interval $[17, 30]$s but with speeds $-1.3m/s$ and $-1.7m$ respectively across the x-axis. Again no targets are present during $[30, 32]$s. Then, targets $\rho = 6, 7$ show up at initial positions $[75, 35]$, $[10, 30]$ and start moving, according to (22), for the time interval $[32, 45]$s and with speed $1.5m/s$ on x-axis and $1.7m/s$ across y-axis. Three new targets, namely $\rho = 8, 9, 10$, appear in initial positions $[40, 70]$, $[40, 10]$, $[60, 70]$ and move in the field for the time interval $[47, 60]$s with different speed $1.4, 1.2, 1.6m/s$ on both the y and x-axis. Finally, the last two targets $\rho = 11, 12$ start at positions $[85, 25]$ and $[48, 48]$ and move within the field for the time interval $[62, 72]$s. Target $\rho = 11$ moves with $-1.0m/s$ and $2.6m/s$ across x and y-axis, while target $\rho = 12$ with corresponding x-axis and y-axis speeds of $-0.7m/s$ and $-2.5m/s$, respectively. Clearly, it can be seen that the configuration of the targets does not really affect the tracking performance of Alg. 2 corroborating its flexibility to track under different geometric configurations of the targets.

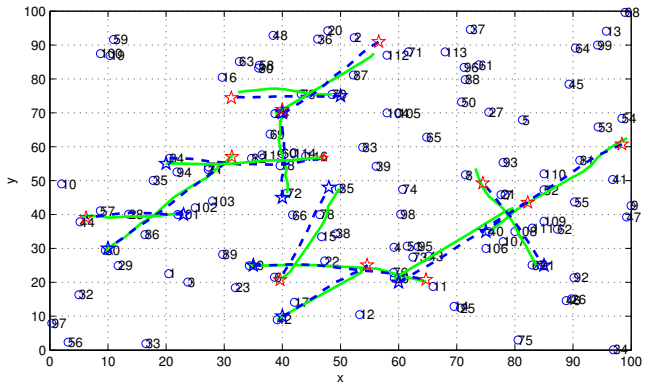

Fig. 10. Tracking of multiple targets in a setting with time-varying number of targets. Trajectories are crossing with each other and targets are moving at different speeds and directions.

In fact. Fig. 11 depicts the number of informative sensors versus time throughout the simulation. Clearly, the number of informative sensors does not exceed 20 (16% of the network), while the average number of informative sensors is equal to approximately 8. Fig. 11 depicts the capability of Alg. 1 to

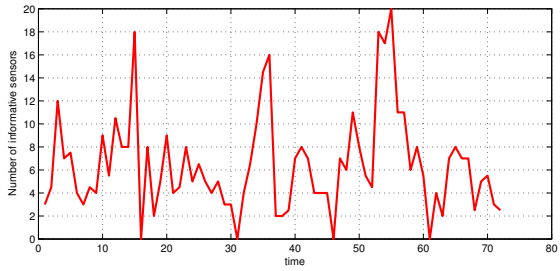

Fig. 11. Number of target-informative sensors versus time t for the setting in Fig. 8.

select only the target-informative sensors to perform tracking. Despite the fact that only a small percentage of the sensors is utilized, still it can track the trajectories quite accurately as seen by Fig. 8.

Fig. 12 depicts the average tracking RMSE corresponding to the tracking of the different targets present in the field at every time instant. Alg. 2, the EKF-based scheme combined with Alg. 1 and the UKF scheme combined with Alg. 1 are compared. Note that at the time intervals $15, 30, 45, 60$s the average

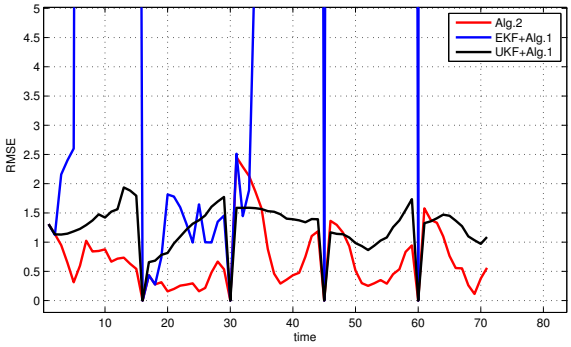

Fig. 12. Average tracking RMSE versus time in a setting with $R = 12$ targets for the setting in Fig. 8.

tracking RMSE is zero. It is initialized there because during these time intervals no targets are detected in the field and thus there is nothing to track and no corresponding tracking RMSE. However, when targets are present, the superiority of Alg. 2 over the EKF based approach combined with Alg. 1 is apparent in Fig. 12. Clearly, the EKF-based approach cannot perform efficient tracking which is further challenged by the varying number of targets and high-variance state and observation noise. The linearization error is

the reason for the big performance gap between the two aforementioned schemes. Further, it can be seen that UKF combined with Alg. 1 performs better than EKF combined with Alg. 1 as expected, however its performance is still worse than Alg. 2. Again EKF suffers from linearization errors, such errors are resolved by UKF which still is worse than Alg. 2 since it just estimates the posterior mean and covariance instead of tracking the posterior pdf (see also details for Fig. 3).

Last but not least, we demonstrate in Fig. 13 the performance of the novel tracking scheme in a setting where some targets are moving in close distance. Specifically, $m = 120$ sensors are randomly employed in the region $[0, 100] \times [0, 100]m^2$. As before a total number of $R = 12$ targets show up and vanish orderly in the sensed field. The configuration is set as follows: Targets $\rho = 1, 2, 3$ start moving at positions $[35, 25]$, $[40, 45]$, $[20, 55]$ and follow the dynamics in (22), with a speed of $2m/s$ across the x-axis. The three targets move in the field for the time interval $[1, 15]$s and then are not sensed anymore. In the interval $[15, 17]$s, no targets are present in the field. Then, targets $\rho = 4, 5$ start moving *closely* at positions $[12, 25]$, $[10, 29]$ and move in parallel according to the same state model followed by the first three targets but with speed $1.5m/s$ across the x-axis. The latter takes place during the time interval $[17, 30]$s. Again no targets are present during $[30, 32]$s. Then, targets $\rho = 6, 7$ show up at close initial positions $[76, 30]$, $[76, 34]$ and move within the time interval $[32, 45]$s. Target $\rho = 6$ moves with $1.0m/s$ and $1.0m/s$ across x and y-axis, while target $\rho = 7$ with corresponding x-axis and y-axis speeds of $-1.0m/s$ and $-1.0m/s$. The two targets are crossing and they move away from each other (far right of Fig. 13). The remaining targets follow the same configuration as the one used in generating Fig. 8.

Targets can be placed as close as possible, as long as no more than one targets are positioned inside the sensing region of a sensor, i.e., the region specifying how far the sensor can 'see'. As the sensing region of the sensors is reduced, targets can be located closer while ensuring that all targets will be distinguished and tracked. One may wonder how is it possible to track both targets at the far right in Fig. 13 when they cross. A crucial step is that the two targets should be sufficiently apart when the sparsity-aware decomposition scheme is utilized to detect the targets and associate them with the sensor measurements. Once both targets have been associated successfully then they can be tracked accurately no matter how close they appear.

VIII. Concluding Remarks

A novel method performing distributed sensor-target association and multi-target tracking was designed and tested in multi-sensor networks. Our approach is based on a novel blending of particle filtering and sparsity-aware matrix decomposition techniques. Target-informative sensors are selected online and

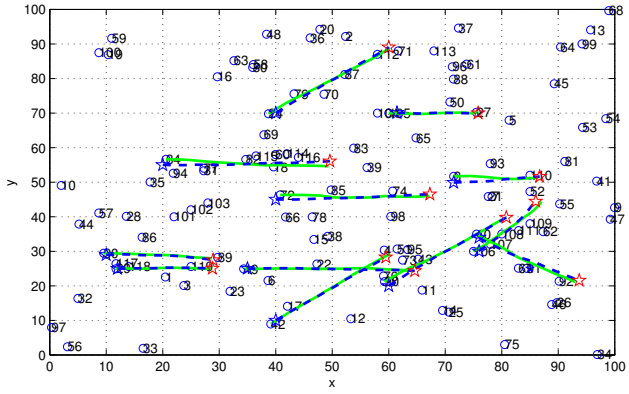

Fig. 13. Tracking of multiple targets in a setting with time-varying number of targets and some targets's trajectory closely placed. Blue dashed curves indicate the true target trajectories, while the light green curves correspond to the estimated trajectories.

their measurements are used for tracking. The proposed approach is capable to detect changes in the configuration and population of the targets present in the sensed field. Extensive numerical tests show that the proposed tracking framework outperforms related approaches in tracking multiple targets. The novel tracking methodology is robust even in high-variance state and observations noises, and provides accurate estimates of the targets' position by utilizing only a small number of the available sensors.

APPENDIX A
PROOF OF EQS. (9), (10) AND (12)

Let $\mathbf{M}_t(j,\ell) = y$, while setting the rest minimization variables in (8) to their most up-to-date values at the end of cycle $k-1$. It follows that $\hat{\mathbf{M}}_t^k(j,\ell)$ is the minimizer of

$$\arg\min_y y^4 + c_1 \cdot y^2 + c_2 \cdot y + \lambda_\rho t, \quad \text{s. to } |y| \leq t, \tag{26}$$

where

$$c_1 = 2\sum_{i\in\mathcal{N}_j}[\hat{\mathbf{M}}_t^{k-1}(i,\ell)]^2 - 2\zeta_{t,\Sigma}^k(j,j,\ell) + \phi, \text{ and } \quad c_2 = -4\sum_{i\in\mathcal{N}_j}\zeta_{t,\Sigma}^k(j,i,\ell)\hat{\mathbf{M}}_t^{k-1}(i,\ell). \tag{27}$$

After evaluating the derivatives of the cost in (26) wrt y and t and applying the Karush-Kuhn-Tucker optimality conditions [6] it follows that $y^* := \hat{\mathbf{M}}_t^k(j,\ell)$ should satisfy $4(y^*)^3 + 2c_1y^* + c_2 + \mu_1^* - \mu_2^* = 0$ and $-\mu_1^* - \mu_2^* + \lambda_\ell = 0$, where μ_1^* and μ_2^* are the optimal multipliers corresponding to the inequality constraints of (26). Note that $\mu_1^* \geq 0, \mu_2^* \geq 0$, while the complementary slackness conditions impose that $\mu_1^*(y^* - t^*) = \mu_2^*(-t^* - y^*) = 0$. If $y^* > 0$ the slackness conditions imply that $\mu_2^* = 0$ from which

it follows that $\mu_1^* = \lambda_\ell$. Substituting the latter values in $4(y^*)^3 + 2c_1y^* + c_2 + \mu_1^* - \mu_2^* = 0$ gives (9). Similarly, the negative candidate minimizers of (26) can be obtained by the roots of (10). Differentiating the cost in (8) with respect to $\sigma_{j,t}$ and setting the derivative equal to zero we can obtain (12). □

APPENDIX B
CONVERGENCE OF ALG. 1:

Let $h(\{\mathbf{M}_t(j,\rho)\}_{j=1,\rho=1}^{m,L}, \{\sigma_{j,t}\}_{j=1}^m)$ denote the cost given in (8) which is defined over $\mathbb{R}^{m(L+1)\times 1}$, and let's define

$$h_0(\{\mathbf{M}_t(j,\rho)\}_{j=1,\rho=1}^{m,L}, \{\sigma_{j,t}\}_{j=1}^m) := \sum_{j=1}^m \sum_{j'\in\mathcal{N}_j} [\hat{\mathbf{\Sigma}}_{x,t}(j,j') - \sum_{l=1}^L \mathbf{M}_t(j,l)\mathbf{M}_t(j',l)]^2 + \phi \sum_{\ell=1}^L \|\mathbf{M}_{t,:\ell}\|_2^2.$$

Further, consider the level set

$$\mathcal{H}_t^0 := \{\{\mathbf{M}_t(j,\rho)\}_{j,\rho=1}^{m,L} : h(\{\mathbf{M}_t(j,\rho)\}_{j=1,\rho=1}^{m,L}, \{\sigma_{j,t}\}_{j=1}^m) \le h(\hat{\mathbf{M}}_t^0)\}, \tag{28}$$

where $\hat{\mathbf{M}}_t^0$ is the $m \times L$ matrix used to initialize Alg. 1 and selected such that $\|\hat{\mathbf{M}}_t^0\|_1 < \infty$ from which it follows that $h(\hat{\mathbf{H}}^0) < \infty$, while the noise variances $\sigma_{j,t,0} = 0$ for $j = 1, \ldots, m$. Then, from (28) and the form of $h(\cdot)$ it follows that the member matrices $\mathbf{M}_t$ of $\mathcal{H}_t^0$ satisfy

$$\sum_{\ell=1}^L \sum_{j=1}^m \lambda_\ell |\mathbf{M}_t(j,\ell)| \le h(\hat{\mathbf{M}}_t^0) < \infty.$$

Thus, the set $\mathcal{H}^0$ is closed and bounded (compact). Also, $h(\cdot)$ is continuous on $\mathcal{H}^0$.

Recall that the cost involved in updating $\hat{\mathbf{M}}_t^k(j,\ell)$ can be written as $J_t^k(j,\ell) := y^4 + c_1y^2 + c_2y + \lambda_\rho|y|$, [cf. (26)]. If $c_2 \neq 0$ then after determining the monotonicity of $J_t^k(j,\ell)$ it follows that it has a unique minimizer. If $c_2 = 0$, then $J^k(j,\rho)$ is symmetric around zero. In that case if $c_1 > 0$ then the unique minimizer of $J_t^k(j,\rho)$ is 0. Though, if $c_1 < 0$ then $J_t^k(j,\rho)$ has two minimizers with the same magnitude but different sign. In that case we can consistently select the positive (or negative) minimizer ensuring a unique minimizer per iteration. Function $h(\cdot)$ satisfies the regularization conditions outlined in [50, (A1)]. In detail, the domain of $h_0(\cdot)$ is formed by matrices whose entries satisfy $\mathbf{M}_t(j,\ell) \in (-\infty, +\infty)$. Then, domain$(h_0) = (-\infty, \infty)^{m(L+1)\times 1}$ is an open set. Further, $h_0(\cdot)$ is Gâteaux differentiable over domain(h_0). The Gâteaux derivative is

$$h_0'(\mathbf{M}; \mathbf{\Delta}_M) := \lim_{\epsilon\to 0} [h_0(\mathbf{M} + \epsilon\mathbf{\Delta}_M) - h_0(\mathbf{M})]/\epsilon. \tag{29}$$

After carrying out the necessary algebraic operations it follows readily that $h_0'(\mathbf{M}; \boldsymbol{\Delta}_M)$ exists for all $\boldsymbol{\Delta}_M \in \text{domain}(h_0)$, and it is equal to

$$-2\text{tr}[(\mathbf{E} \odot (\hat{\boldsymbol{\Sigma}}_{x,t} - \mathbf{M}_t\mathbf{M}_t^T))(\mathbf{E} \odot (\mathbf{M}_t\boldsymbol{\Delta}_M^T + \boldsymbol{\Delta}_M\mathbf{M}_t^T)] + \mathbf{1}^T(\mathbf{M}_t \odot \boldsymbol{\Delta}_M)\mathbf{1}.$$

The aforementioned properties ensure convergence of the Alg. 1 iterates to a stationary point of $h(\cdot)$ [50, Thm. 4.1 (c)]. □

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their insightful comments helping to improve the presentation of this paper.

REFERENCES

[1] N. Ahmed, M. Rutten, T. Bessell, S. S. Kanhere, N. Gordon, and S. Jha, "Detection and Tracking Using Particle-Filter-Based Wireless Sensor Networks," *IEEE Trans. on Mobile Computing,* vol. 9, no. 9, pp. 1332–1345, Sep. 2010.

[2] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Trans. Signal Proc.*, vol. 50, no. 2, 174–188, 2002.

[3] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation With Applications to Tracking and Navigation*. New York: Wiley, 2001.

[4] M. Baum and U. D. Hanebeck, "The Kernel-SME Filter for Multiple Target Tracking," *Proc. of the 16th International Conference on Information Fusion (Fusion 2013)*, Istanbul, Turkey, July 2013, pp. 288–295.

[5] M. Baum and U. D. Hanebeck,"Extended Object Tracking Based on Set-Theoretic and Stochastic Fusion," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 3103–3115, Oct. 2012.

[6] D. P. Bertsekas, *Nonlinear Programming*, Second Edition, Athena Scientific, 2003.

[7] E. Candès, J. Romberg, and T. Tao, "Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information," *IEEE Trans. on Info. Theory*, pp. 489–509, Feb. 2006.

[8] M. Coates, "Distributed Particle Filters for Sensor Networks," *Proc. of the 3rd Intl. Symposium on Information Proc. in Sensor Networks (IPSN 04)*, pp. 99-107, April 2004.

[9] S. S. Dias and M. G. S. Bruno, "Cooperative Target Tracking using Decentralized Particle Filtering and RSS Sensors," *IEEE Trans. on Sig. Proc.,* vol. 61, no. 14, pp. 3632–3646, July 2013.

[10] P. Djuric, J. H. Kotecha, J. Zhang, Y. Hang, T. Ghirmai, M. F. Bugallo, and J. Miguez, "Particle Filtering," *IEEE Signal Processing Magazine,* vol. 20, no. 5, pp. 19–38, Sep. 2003.

[11] A. Doucet, N. Freitas, and N. Gordon (eds.), *Sequential Monte Carlo Methods in Practice*, Springer, NY, 2001.

[12] A. Doucet, B.-N. Vo, C. Andrieu, and M. Davy, "Particle Filtering for Multi-Target Tracking and Sensor Management," *Proc. of the Fifth Intl. Conf. on Information Fusion*, Annapolis, Maryland, July 2002, pp. 474–481.

[13] J. A. Fuemmeler and V. V. Veeravalli, "Energy Efficient Multi-Object Tracking in Sensor Networks," *IEEE Trans. on Sig. Proc.*, vol. 58, no. 7, pp. 3742–3750, 2010.

[14] A. Goldsmith, "Wireless Communications," *Cambridge University Press,* 2005.

[15] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation," *IEE Proc. for Radar and Sig. Processing,* vol 140, no. 2, pp. 107–113, 1993.

[16] A. Gorji and M. B. Menhaj, "Multiple Target Tracking for Mobile Robots Using the JPDAF Algorithm," *Proc. of IEEE Intl. Conf. on Tools with Artificial Intelligence,* Patras, Greece, Oct. 2007, pp. 137–145.

[17] N. Guan, D. Tao, Z. Luo, and B. Yuan, "NeNMF: An Optimal Gradient Method for Nonnegative Matrix Factorization," *IEEE Trans. on Sig. Proc.*, vol. 60, no. 6, pp. 2882–2898, June 2012.

[18] R. A. Horn and C. R. Johnson, Matrix Analysis. Cambridge, U.K.: Cambridge Univ. Press, 1985.

[19] P. Hoyer, "Non-negative Matrix Factorization with Sparseness Constraints," *Journal of Machine Learn. Res.*, vol. 5, pp. 1457–1469, 2004.

[20] C. Hue, J.-P. Le Cadre, and P. Perez, "Tracking Multiple Objects with Particle Filtering," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 3, pp. 791–812, 2002.

[21] F. Iutzeler, P. Ciblat, and J. Jakubowicz, "Analysis of Max-Consensus Algorithms in Wireless Channels," *IEEE Trans. on Signal Processing,* vol. 60, no. 11, pp. 6103–6107, 2012.

[22] S. J. Julier and J. K. Uhlmann, "Unscented Filtering and Nonlinear Estimation,"*Proc. of the IEEE,* vol. 92, no.3, pp. 401–422, 2004.

[23] K. Kang and V. Maroulas, "Drift Homotopy Methods for a NonGaussian Filter," *Proc. of the 16th Intl. Conf. on Information Fusion (FUSION)*, pp. 1088–1094, 2013.

[24] K. Kang and V. Maroulas, "Drift homotopy particle filter for non-Gaussian multi-target tracking," *Information Fusion (FUSION), 2014 17th International Conference*, pp.1,7, 7-10 July 2014

[25] S. M. Kay, *Fundamental of Statistical Signal Processing: Estimation Theory*, Prentice Hall, 1993.

[26] V. Krishnamurthy, M. Maskery, and G. Yin, "Decentralized Adaptive Filtering Algorithms for Sensor Activation in an Unattended Ground Sensor Network," *IEEE Trans. on Sig. Proc.*, vol. 56, pp. 6086–6101, 2008.

[27] D. D. Lee and H. S. Seung, "Algorithms for Non-negative Matrix Factorization," *Proc. of 13th Conference on Advanced Neural Information Processing Systems (NIPS)*, Denver, CO, 2000, pp. 556–562.

[28] C. J. Lin, "Projected Gradient Methods for Nonnegative Matrix Factorization," *Neural Computation*, vol. 19, no. 10, pp. 2756–2779, 2007.

[29] J. Lin, W. Xiao, F. L. Lewis, and L. Xie, "Energy-Efficient Distributed Adaptive Multisensor Scheduling for Target Tracking in Wireless Sensor Networks," *IEEE Tr. on Instr. and Meas.*, vol. 58, pp. 1886–1896, 2009.

[30] J. S. Liu and R. Chen, "Sequential Monte Carlo Methods for Dynamic Systems," *Journal of the American Stat. Association*, vol. 93, no. 443, pp. 1032-1044, 1993.

[31] H. Q. Liu, H. C. So, F. K. W. Chan, and K. W. K. Lui, "Distributed Particle Filter for Target Tracking in Sensor Networks," *Progress In Electromagnetics Research C*, vol. 11, pp. 171-182, 2009.

[32] R. P. S. Mahler, *Statistical Multisource-Multitarget Information Fusion*, Artech House Publishers MA, 2007.

[33] R. P. S. Mahler, "Multitarget Bayes Filtering via First-order Multitarget Moments," *IEEE Trans. Aero. Elect. Sys.*, vol. 39, no. 4, pp. 1152–1178, 2003.

[34] R. P. S. Mahler and V. Maroulas, "Tracking Spawning Objects", *IET Radar, Sonar & Navigation*, vol. 7, no. 3, pp. 321–331, 2013.

[35] V. Maroulas and P. Stinis, "Improved Particle Filters for Multi-Target Tracking," *Journal of Computational Physics*, vol. 231, no. 2, pp. 602–611, 2012.

[36] W. Ng, J. Li, S. Godsill, and J. Vermaak, "A Hybrid Approach for Online Joint Detection and Tracking for Multiple Targets," *Proc. of the IEEE Aerospace Conference*, Big Sky, MT, March 2005, pp. 2126–2141.

[37] S. Oh, "A Scalable Multi-Target Tracking Algorithm for Wireless Sensor Networks," *International Journal of Distributed Sensor Networks*, 2012.

[38] S. Oh, S. Russell, and Shankar Sastry, "Markov Chain Monte Carlo Data Association for Multi-Target Tracking," *IEEE Transactions on Automatic Control*, vol. 54, pp. 481–497, 2009.

[39] R. Olfati-Saber, "Distributed Kalman Filter with Embedded Consensus Filters," *Proc. 44th Conf. Dec., the Eur. Control Conf.*, Seville, Spain, Dec. 2005, pp. 8179–8184.

[40] O. Ozdemir, R. Niu, and P. K. Varshney, "Tracking in Wireless Sensor Networks Using Particle Filtering: Physical Layer Considerations," *IEEE Trans. on Signal Processing*, vol. 57, no. 5, pp. 1987–1999, May 2009.

[41] G. Ren and I. D. Schizas, "Distributed Sensor-Informative Tracking of Targets," *Proc. of the IEEE Intl. Workshop on Computational Advances in Multi-Sensor Adaptive Processing,* Saint Martin, Dec. 15-18, 2013.

[42] N. F. Sandell and R. Olfati-Saber, "Distributed Data Association for Multi-Target Tracking in Sensor Networks," *IEEE Conference on Decision and Control*, Cancun, Mexico, pp. 1085–1090, 2008.

[43] A. H. Sayed, *Fundamentals of Adaptive Filtering*. New York: Wiley, 2003.

[44] I. D. Schizas, "Distributed Informative-Sensor Identification via Sparsity-Aware Matrix Factorization," *IEEE Trans. on Sig. Proc.*, vol. 61, no. 18, pp. 46104624, Sep. 2013.

[45] C. Snyder, T. Bengtsson, P. Bickel, and J. Anderson, "Obstacles to High-dimensional Particle Filtering," *Mon. Wea. Rev.*, vol. 136, pp. 4629–4640, 2008.

[46] V. Solo and X. Kong, *Adaptive Signal Processing Algorithms: Stability and Performance.* Prentice Hall, 1995.

[47] P. Stinis, "Conditional Path Sampling for Stochastic Differential Equations by Drift Relaxation," *Comm. Appl. Math. Comput. Sci.* vol. 6, no. 1, 63–78, 2011.

[48] R. Tharmarasa, T. Kirubarajan, A. Sinha, and T. Lang, "Decentralized Sensor Selection for Large-Scale Multisensor-Multitarget Tracking," *IEEE Trans. on Aerospace and Electronic Systems,* vol. 47, no. 2, pp. 1307–1324, April 2011.

[49] R. Tibshirani, "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society, Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[50] P. Tseng, "Convergence of a Block Coordinate Descent Method for Nondifferentiable Minimization," *Journal of Opt. Theory and Applications*, vol. 109, no. 3, pp. 475–494, Jun. 2001.

[51] M. O. Ulfarsson and V. Solo, "Sparse Variable PCA Using Geodesic Steepest Descent," *IEEE Transactions on Signal Processing*, vol. 10, no. 12, pp. 5823–5832, 2008.

[52] J. Vermaak, S. J. Godsill, and P. Perez, "Monte Carlo Filtering for Multi Target Tracking and Data Association," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no, 1, pp. 309–332, 2005.

[53] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo Methods for Multi-Target Filtering with Random Finite Sets," *IEEE Trans. Aero. Elect. Sys.*, vol. 41, no. 4, pp. 1224–1245, 2005.

[54] L. Xiao and S. Boyd, "Fast Linear Iterations for Distributed Averaging," *System and Control Letters,* vol. 53, pp. 65–78, Sept. 2004.

[55] W. Yi, M. Morelande, L. Kong, and J. Yang, "A Computationally Efficient Particle Filter for Multi-Target Tracking Using an Independence Approximation," *IEEE Tran. on Signal Proc.*, vol. 61, no. 4, pp. 843–856, 2013.

[56] W. Zhang and G. Cao, "DCTC: Dynamic Convoy Tree-Based Collaboration for Target Tracking in Sensor Networks," *IEEE Trans.on Wir. Com.*, vol. 3, no. 5, pp. 1689–1701, Sep. 2004.

[57] H. Zhu, I. D. Schizas, and G. B. Giannakis, "Power-Efficient Dimensionality Reduction for Distributed Channel-Aware Kalman Tracking Using Wireless Sensor Networks," *IEEE Tran. on Sig. Proc.,* vol. 57, no. 8, pp. 3193–3207, 2009.

[58] H. Zou, T. Hastie, and R. Tibshirani, "Sparse Principal Component Analysis," *Journal of Computational and Graphical Statistics*, vol. 15, no. 2, 2006.

Guohua Ren received the Diploma in electrical engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2012. Since August 2012, he has been working toward the Ph.D. degree with the Department of Electrical Engineering, University of Texas at Arlington, Texas. His research interests lie in the areas of signal processing and networking. His current research focuses on distributed sources and target tracking with wireless sensor networks.

Vasileios Maroulas Maroulas received his PhD in Statistics from the University of North Carolina at Chapel Hill in 2008. He held an industrial postdoctoral fellowship in the Institute for Mathematics and its Applications (IMA) at the University of Minnesota and Lockheed Martin. He joined the University of Tennessee as an Assistant Professor of Mathematics at the University of Tennessee in 2010. He also holds a courtesy appointment in the Business Analytics and Statistics Department at the Haslam School of Business and he is a member of the Center of Intelligence and Machine Learning at the College of Engineering at the University of Tennessee. His main research interests are Dynamic Data Learning, Bayesian Filtering and Data Assimilation; Rare Events Estimation with applications in Biology, Defense, and Engineering.

Ioannis D. Schizas (M12) received the diploma in Computer Engineering and Informatics (with honors) from the University of Patras, Greece, in 2004, the M. Sc. And the Ph. D. degrees, both in Electrical and Computer Engineering from the University of Minnesota, Minneapolis, in 2007 and June 2011, respectively. Since August 2011, he has been an Assistant Professor at the Electrical and Computer Engineering Department, University of Texas at Arlington. His general research interests lie in the areas of statistical signal processing, wireless sensor networks and data dimensionality reduction. Specifically, he is working on the design and analysis of distributed processing algorithms, dynamic learning techniques and tracking.

TABLE I

TABLE OF VARIABLES.

Variables	Denotes
$\mathbf{A}$	State transition matrix
$b_\rho(t)$	Intensity of signal emitted by the ρth target
$C_{\rho_\ell,t}$	Leading sensor at t for the ρ_ℓth target
$d_{j,\rho}(t)$	Distance between sensor j and target ρ at t
$\mathbf{D}_t$	Distance matrix
$\mathbf{E}$	Adjacency Matrix of the SN
$\mathcal{J}_{\rho,t}$	Candidate target-informative sensors for target ρ at t
K	State vector dimensionality
λ_ρ	Sparsity controlling coefficient
L	Upper bound on number of active targets
m	Number of sensors
$\mathbf{M}_t$	Sparse factors matrix
$\mathcal{N}_j$	Single-hop neighborhood of sensor j
$\mathbf{P}_j$	Sensor j's position
$\mathbf{p}_\rho(t)$	Unknown position of target ρ at t
$p(\mathbf{s}_\rho(t)\|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})$	Posterior pdf for target ρ
Q	Total number of particles
$q(\mathbf{s}_\rho(t)\|\mathbf{x}_{\mathcal{T}_{\rho,0:t}})$	Importance sampling pdf
R	Total number of targets
$r(t)$	Number of active targets at t
R_s	Radius for constructing candidate sets $\mathcal{J}_{\rho,t}$
$\mathbf{s}_\rho(t)$	State of target ρ
$\mathbf{s}^i_{\rho,t}$	i-th particle for target ρ at t
$\mathbf{\Sigma}_u$	State noise covariance matrix
σ^2_w	Sensing noise variance
$\mathbf{\Sigma}_w$	Measuring noise covariance matrix
$\mathbf{\Sigma}_{x,t}$	Sensor data covariance matrix at t
$\hat{\mathbf{\Sigma}}_{x,t}$	Covariance matrix estimate
$\mathcal{T}_{\rho,t}$	Informative-sensors selected for target ρ at t
T_s	Number of measurements acquired at the start-up phase
$\mathbf{u}_\rho(t)$	State noise associated with target ρ
$\mathbf{v}_\rho(t)$	Velocity for target ρ
$\mathbf{w}_t$	Sensing (Measurement) noise
$w^i_{\rho,t}$	Weight corresponds to the i-th particle for target ρ at t
$\bar{\mathbf{x}}_t$	Estimate for the data ensemble mean
$x_j(t)$	Measurement of sensor j at time t
γ	Forgetting factor
ΔT	Sampling period