Abstract—The problem is joint detection and tracking of possibly several objects moving through a region of interest. A wireless sensor network (WSN), deployed in the region, collects the acoustic energy measurements and sends them to the fusion center for processing. The problem is cast in the sequential Bayesian estimation framework and solved using a particle filter. The number of objects is unknown and can vary over time. The paper presents the algorithm and demonstrates its performance by computer simulations. The particle filter error performance is compared to the theoretical Cramer-Rao bound (CRB) for multiple target tracking with the described WSN.

Keywords: Wireless sensor network, multiple-target tracking, detection, acoustic sensors, nonlinear filters, Cramer-Rao bound.

I. INTRODUCTION

Wireless networks of tiny sensing and computing devices are becoming increasingly attractive for various monitoring applications (traffic, habitat, battlefield, structural integrity, etc) [1], [2]. In this paper we consider the problem of simultaneous detection and tracking of possibly several objects moving through a region of interest, in the ground surveillance context. The number of objects is unknown and can vary over time. The assumption is that the region is populated by a collection of calibrated acoustic sensor nodes placed arbitrarily at locations known to the fusion centre. The energy measurements collected by the sensor nodes are sent through the network to the fusion centre for processing.

The simultaneous multi-object detection/tracking problem is cast in the sequential Bayesian estimation framework and solved in this paper using a particle filter. Particle filters have already been proposed for target tracking in wireless sensor networks by several authors. Tracking a single target using a binary and tertiary network was considered in [3] and [4], respectively. Tracking multiple targets using quantised measurements, studied in [5], assumes the number of objects is fixed and known. We carry out simultaneous multi-object detection/tracking via a hybrid state vector of a variable size, where a discrete-valued random variable, which indicates the number of objects present in the region, is used to perform the detection. The continuous-valued part of the state vector, in the usual manner, is used for estimation (tracking) of target locations, velocities, intensities, etc. Conceptually, the proposed algorithm is similar to the recursive track-before-detect algorithm and multi-object visual tracker, described in [6], [7, Ch.11] and [8].

The paper is organised as follows. Section II describes the problem by defining the measurement model of the WSN of acoustic sensors, as well as the target motion models. Section III presents the sequential Bayesian algorithm and its implementation in the from of the particle filter. The theoretical Cramér-Rao bound for multiple target tracking with a WSN of acoustic sensors is derived in Section IV. The simulations results are shown in Section V and the conclusions of this study drawn in Section VI.

II. PROBLEM FORMULATION

A. Measurement model

A wireless network composed of $S$ acoustic sensor nodes is deployed in a two-dimensional region. The locations of sensor nodes, in Cartesian coordinates denoted by $(x_i, y_i)$, $i = 1, \ldots, S$, are arbitrary but known to the fusion centre. Each sensor node provides (with a certain sampling interval) its measurements of acoustic energy. Let $z_{k,i_k}$ denote such a measurement, with subscript $k$ referring to its time-stamp $t_k$, and subscript $i_k \in \{1, \ldots, S\}$ to the sensor node. Each measurement $z_{k,i_k}$ is represented by $8$ bits and sent in a data packet through the network to the fusion centre. A data packet typically contains several header bytes (protocol type, node identifier, time stamp, etc.), and hence representing the energy measurement by a single bit in order to reduce the communication load (as for example in [3], [4]) is not of interest here. Assuming there are $M \geq 0$ objects emitting the sound in the region of interest, we adopt the following measurement model [9]:

$$z_{ki} = \gamma_i \sum_{m=1}^{M} \frac{\alpha_{km}}{d_{kim}^{\alpha}} e_{i} + w_{ki}$$  \hspace{1cm} (1)

where $\gamma_i$ is the gain factor of sensor $i$; $\alpha_{km}$ is acoustic energy (intensity) of object $m$; $e_{i}$ is the propagation loss factor which in general depends on the environment [9] but here will be
fixed to 2 (inverse-square law); $d_{km}$ is the distance between object $m$ and sensor $i$,
\[ d_{km} = \sqrt{(x_i - x_km)^2 + (y_i - y_km)^2}, \] (2)
with $(x_km, y_km)$ being the location of object $m$ at time $k$ in the Cartesian coordinates. Finally, $w_{ki}$, in (1) is additive noise in sensor $i$, which can be modelled (after the mean removal) as being zero-mean white Gaussian [9] with variance $\sigma^2_w$. We assume that prior to the deployment of the sensor network, sensor nodes have been calibrated and thus $\gamma_i$ and $\sigma^2_w$ are known. Note that the measurements can be acquired asynchronously.

B. State vector and dynamic model

The state vector of a single object consists of position and velocity, in a two-dimensional (2D) Cartesian coordinate system, and of its unknown acoustic energy. Thus at discrete-time $k$ the state vector of object $m = 1, \ldots, M$ is defined as
\[ x_km = [x_km, \dot{x}_km, y_km, \dot{y}_km, \alpha_km]^T \]
where $(\dot{x}_km, \dot{y}_km)$ represents the velocity of object $m$. In order to handle more accurately the non-uniform measurement sampling intervals $T_k = t_{k+1} - t_k$, we adopt the continuous white noise acceleration constant velocity (CV) model for object motion [10, Ch.6] and the random walk model for acoustic energy $\alpha_km$. The dynamic model is then:
\[ x_{k+1,m} = F(T_k)x_{k,m} + u_{k,m}, \] (3)
where
\[ F(T_k) = \begin{pmatrix} 1 & T_k & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & T_k & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \]
is the transition matrix. Vector $u_{k,m} \sim \mathcal{N}(0, Q(T_k))$ is process noise which accounts for deviations from the CV motion and the fluctuations of the acoustic energy level. Process noise is assumed white and uncorrelated to measurement noise. Covariance matrix $Q(T_k)$ is given by [10]:
\[ Q(T_k) = \frac{\sigma_q^2 T_k^3}{2} \begin{pmatrix} 1 & \frac{\sigma_q^2 T_k^2}{2} & 0 & 0 & 0 \\ \frac{\sigma_q^2 T_k^2}{2} & \sigma_q^2 T_k & 0 & 0 & 0 \\ 0 & 0 & 1 & \frac{\sigma_q^2 T_k}{2} & 0 \\ 0 & 0 & \frac{\sigma_q^2 T_k}{2} & 1 & 0 \end{pmatrix} \]
where process noise intensities $\sigma_q^x, \sigma_q^y$ and $\sigma_q^I$ are the tuning parameters.

In order to sequentially estimate the number of objects in the surveillance region, we introduce a discrete-valued random variable $E \in \mathbb{E} = \{0, 1, \ldots, M^*\}$, where $M^*$ is the maximum expected number of objects. The dynamics of random variable $E$ is modeled by an $(M^* + 1)$-state Markov chain, whose transitions are specified by an $(M^* + 1) \times (M^* + 1)$ transitional probability matrix (TPM) $\Pi = [\pi_{ij}]$, where
\[ \pi_{ij} = Pr\{E_k = j | E_{k-1} = i\}, \quad (i, j \in \mathbb{E}) \] (4)
is the probability of a transition from $i$ objects existing at time $k - 1$ to $j$ objects at time $k$. The elements of the TPM satisfy $\sum_{i,j=0}^{M^*} \pi_{ij} = 1$ for each $i \in \mathbb{E}$. The dynamics of variable $E$ is fully specified by the TPM and its initial probabilities at time $k = 0$, i.e., $\mu_i = Pr\{E_0 = i\}$, for $i = 0, 1, \ldots, M^*$.

The overall detection/estimation problem is to provide sequentially a joint estimate of:
- the number of targets present in the surveillance area $\hat{E}_k$, and
- the estimates of target state vectors $\hat{x}_{k,m}$, for $m = 1, \ldots, \hat{E}_k$,
using prior knowledge and a cumulative set of sensor network measurements up to time $k$: $z_{1:k} = \{z_{1,i} : j = 1, \ldots, k\}$.

III. BAYESIAN RECURSIVE ALGORITHM

This section briefly describes the conceptual solution, followed by its implementation in the form of a particle filter.

A. Conceptual solution

First we define an augmented state vector:
\[ y_k = (E_k \quad x_{1,k} \quad \ldots \quad x_{k,\hat{E}_k})^T. \] (5)
Given the posterior density $p(y_{k-1} | z_{1:k-1})$, and the latest available measurement $\hat{z}_{k,i}$, the goal is to construct the posterior density at time $k$, that is $p(y_k | z_{1:k})$. Then the probability $P_M = Pr\{E_k = M | z_{1:k}\}$ that there are $M$ objects in the surveillance region at time $k$ is computed as the marginal of $p(y_k | z_{1:k})$, i.e.:
\[ P_M = \int p(x_k,1, \ldots, x_{k,M}, E_k = M | z_{1:k}) \, dx_k,1 \ldots dx_k,M \] for $M = 0, 1, \ldots, M^*$. The MAP estimate of the number of objects at time $k$ is then determined as:
\[ \hat{M}_k = \arg \max_{M=0,1,\ldots,M^*} P_M. \] (7)
This estimate provides the means for sequential detection of new object appearance and the existing object disappearance. The posterior pdfs of state components corresponding to individual objects in the scene are then computed as the marginals of pdf $p(x_{k,1}, \ldots, x_{k,M}, E_k = \hat{M}_k | z_{1:k})$.

The formal Bayesian recursive solution to the sequential hybrid estimation consists of two steps: prediction and update. For the prediction step, one must distinguish between the case of independent objects and the case of objects moving in coordination. The coordinated objects case requires to specify both the bulk and individual motion models [7, Ch.12], and will not be considered here. For independent objects, the prediction pdf for $M$ objects, $p(x_{k,1}, \ldots, x_{k,M}, E_k = M | z_{1:k-1})$, can be expressed as a function of:
- the product of transitional densities $p(x_{k,m}|x_{k-1,m})$, for $m = 1, \ldots, M$, defined by dynamic model (3);
- the elements of the TPM $\Pi$, and
- the initial object pdf on its appearance $p_{0i}(x_{k,m})$ (a.k.a. birth density) which is assumed known.
In relation to the birth density, we can assume for example the objects to appear or disappear only in certain segments of the surveillance region (e.g. edges, roads entering the region), or alternatively in the entire region.

The update step results from the application of the Bayes rule and allows us to write the posterior pdf \( p(y_k|x_{1:k}) \) as a function of the prediction pdf and the measurement likelihood \( p(z_{k,i}, x_{k-1}, \ldots, x_{k,M}, E_k = M) \) which, according to (1), is Gaussian with mean \( \gamma_k \sum_{i=1}^{M} \sigma_{z_{k,i}} = [d_{k,i}^{-1}]^2 \) and variance \( \sigma_{z_{k,i}}^{2} \), i.e.

\[
p(z_{k,i}|x_{k-1}, \ldots, x_{k,M}, E_k = M) = N\left( \gamma_k \sum_{i=1}^{M} \sigma_{z_{k,i}}^{-1} \right), \quad \sigma_{z_{k,i}}^{2}
\]

\( B. \) Particle filter

We have implemented the described conceptual solution in the form of a particle filter (PF). Particle filters approximate the posterior density \( p(y_k|x_{1:k}) \) by a weighted set of random samples or particles. In our case, a particle of index \( n \) is characterised by a certain value of \( E^n_k \) and the corresponding number of state vectors \( x^n_{k,m} \), with \( m = 1, \ldots, E^n_k \), i.e.

\[
y^n_k = \left[ x^n_{k,1}, \ldots, x^n_{k,M}, E^n_k \right] \quad (n = 1, \ldots, N)
\]

where \( N \) is the number of particles. The pseudo-code of a single cycle of the PF is presented in Table I.

<table>
<thead>
<tr>
<th>TABLE I</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARTICLE FILTER PSEUDO-CODE (SINGLE CYCLE)</td>
</tr>
</tbody>
</table>

\[ \left[ \{ y^n_k \}_{n=1}^{N} \right] = \text{PF} \left[ \{ y^n_{k-1} \}_{n=1}^{N}, z_{k,i} \right] \]

1) Random transitions of \( E_{k-1} \) variable:

\[ \left[ \{ E^n_k \}_{n=1}^{N} \right] = \text{ETrans} \left[ \{ E^n_{k-1} \}_{n=1}^{N}, \Pi \right] \]  

2) FOR \( n = 1 : N \)  

a. Based on \( (E^n_{k-1}, E^n_k) \) pair, draw at random \( x^n_{k,1}, \ldots, x^n_{k,M} \);  

b. Compute unnormalised importance weight \( q^n_k \) using \( z_{k,i} \), and (10) and (11).

3) END FOR  
4) Normalise importance weights:  

From \( \{ q^n_k \}_{n=1}^{N} \) to \( \{ \tilde{q}^n_k \}_{n=1}^{N} \)  
5) Resampling:  

\[ \left[ \{ y^n_k \}_{n=1}^{N} \right] = \text{RESAMPLE} \left[ \{ y^n_k, \tilde{q}^n_k \}_{n=1}^{N} \right] \]

6) Regularisation  
7) Compute the output of the PF (for reporting)

The first step in the algorithm represents random transition of \( E^n_{k-1} \) to \( E^n_k \) based on the TPM \( \Pi \). The pseudo-code of this step is given in Table 3.9 of [7].

Step 2.a of Table I follows from the prediction step in the previous section. If \( E^n_{k-1} = E^n_k \) and \( E^n_{k-1} > 0 \), then we draw \( x^n_{k,m} \) from the transitional prior \( p(x_k|x^n_{k-1,m}) \) for \( m = 1, \ldots, E^n_k \). If the number of objects is increased, i.e. \( E^n_{k-1} < E^n_k \), then for the objects that continue to exist we draw \( x^n_{k,m} \) using the transitional prior (as above), while for the newborn objects we draw particles from \( p_0(x_k) \). Finally if \( E^n_{k-1} > E^n_k \), we select at random with equal probability \( (E^n_k - E^n_{k-1}) \) objects to be removed from \( y^n_k \). For the particles that continue to exist we draw \( x^n_{k,m} \) using the transitional prior (as above), while the remaining particles are deleted.

In Step 2.b we compute unnormalised weight of each particle \( n \) using the measurement likelihoods given by (8):

\[
\tilde{q}^n_k = \prod_{i \in I_k} p(z_{k,i}|x^n_{k,1}, \ldots, x^n_{k,E^n_k}, E^n_k) \propto \prod_{i \in I_k} \frac{1}{\sqrt{2\pi\sigma_i}} \exp \left\{ -\frac{(z_{k,i} - h^n_{k,i})^2}{2\sigma^2_i} \right\}
\]

where

\[
h^n_{k,i} = \frac{\alpha_{z_{k,i}}}{\sqrt{\sum_{m=1}^{M} (z^n_{k,m} - h^n_{k,i})^2 + (y^n_{k,m} - y_i)^2}}
\]

and \( I_k \subseteq \{1, \ldots, S\} \) is the set of all sensor indices that report a measurement at time \( k \).

For the resampling step (Step 5), standard \( O(N) \) algorithms exist, see for example Table 3.2 in [7].

Step 6, particle regularisation, is carried out for continuous-valued state components using a Gaussian kernel, and taking into account that particle dimension depends on \( E^n_k \) [11].

In Step 7 we compute the estimate \( \hat{M}_k \) using (7) and

\[
P_M = \frac{1}{N} \sum_{n=1}^{N} \delta(E^n_k, \hat{M}_k),
\]

where \( \delta(i, j) \) is Kroneker delta. Object state estimates follow from:

\[
\hat{x}_{k,m} = \frac{\sum_{n=1}^{N} x^n_{k,m} \delta(E^n_k, \hat{M}_k)}{\sum_{n=1}^{N} \delta(E^n_k, \hat{M}_k)},
\]

for \( m = 1, \ldots, \hat{M}_k \). The performance of the PF is analyzed in Section V. The next section derives the best achievable second-order error performance for multi-target tracking with a WSN.

IV. CRB FOR MULTIPLE TARGET TRACKING WITH A WSN

Derivation of the Cramér-Rao bound for multiple object tracking in general can be very difficult, see for example [12]. The main source of difficulties in the derivation is the assumption that measurements have been thresholded in a signal detector, before being processed by a tracker. If instead tracking is carried out using the unthresholded (raw) intensity measurements (as we do here with acoustic measurements), the derivation of the required CRB becomes surprisingly simple [13], [14].

Let us assume that \( M \) is a fixed and known number of targets present in the surveillance region of interest throughout
the observation interval. Then we can define a compound target state vector as:
\[ X_k = [x^T_{k,1}, \ldots, x^T_{k,M}]^T. \] (14)

Furthermore, let us assume that all \( M \) targets move independently from each other, and for the sake of convenience, let their dynamics be modelled by the CV model given by (3). Then the dynamic model of the compound target can be expressed by:
\[ X_{k+1} = \Phi X_k + U_k \] (15)
where \( \Phi = \text{block-diag} \{ \mathbf{F}, \ldots, \mathbf{F} \} \), and \( U_k \sim \mathcal{N}(0, \Sigma) \), with \( \Sigma = \text{block-diag} \{ \mathbf{Q}, \ldots, \mathbf{Q} \} \). Finally, let us denote the measurement vector at time \( k \) by \( Z_k = \{ z_{ki} : i \in I_k \} \). The measurement equation for the compound target is then given by:
\[ Z_k = h(X_k) + W_k \] (16)
where
\[ h_i(X_k) = \begin{cases} 0, & \text{if } i \notin I_k \\ \gamma_i \sum_{m=1}^{M} \frac{\alpha_m}{(x_i - x_{km})^2 + (y_i - y_{km})^2}, & \text{if } i \in I_k \end{cases} \] (18)

In this way we have reformulated the multi-target tracking problem using intensity measurements as a nonlinear filtering problem. The CRB for nonlinear filtering can be worked out using Riccati-like recursions for the computation of the (Fisher) information matrix \( J_k \) [15]:
\[ J_{k+1} = D_k^{22} - D_k^{21}(J_k + D_k^{11})^{-1}D_k^{12} \quad (k > 0) \] (19)
where, for the additive Gaussian form of process and measurement noise as in (15) and (16), we have:
\[ D_k^{11} = \mathbb{E} \{ \Phi^T \Sigma^{-1} \Phi \} \] (20)
\[ D_k^{12} = -\mathbb{E} \{ \Phi^T \Sigma^{-1} \} = [D_k^{21}]^T \] (21)
\[ D_k^{22} = \Sigma^{-1} + \mathbb{E} \{ H_k^T R^{-1} H_{k+1} \}. \] (22)

The initial information matrix is derived from the prior density \( p(X_0) \) as
\[ J_0 = \mathbb{E} \{ [\nabla_{X_0} \log p(X_0)] [\nabla_{X_0} \log p(X_0)]^T \}. \] (25)

If the prior distribution \( p(X_0) \) is Gaussian with covariance \( \mathbf{P}_0 \), then from (25) we get: \( J_0 = \mathbf{P}_0^{-1} \). For a general non-Gaussian prior, however, equation (25) may not have an analytic solution. Then one approach is to approximate the prior with a Gaussian with covariance \( \mathbf{P}_0 \) and use \( J_0 \approx \mathbf{P}_0^{-1} \). For example, if the prior is a uniform density:

\[ x_{0m} \sim \mathcal{U}[x_{\text{min}}, x_{\text{max}}], \quad x_{0m} \sim \mathcal{U}[-v_{\text{max}}, v_{\text{max}}], \]
\[ y_{0m} \sim \mathcal{U}[y_{\text{min}}, y_{\text{max}}], \quad y_{0m} \sim \mathcal{U}[-v_{\text{max}}, v_{\text{max}}], \] (26)

for \( m = 1, \ldots, M \), then the initial \( J_0 \) can be approximated as a diagonal matrix with:
\[ J_0[s, s] \approx \frac{1}{\text{var}(U[A_s, B_s])} = 12/(B_s - A_s)^2, \] (27)
for all components \( s = 1, \ldots, 5M \), where \( A_s \) and \( B_s \) are the appropriate limits of the uniform density as defined in (26).

The Jacobian \( H_k \) defined in (23) is an \( S \times 5M \) matrix with elements:
\[ H_k[i, j] = \frac{\partial h(X_k)[i]}{\partial X_k[j]} \] (28)

with \( i = 1, \ldots, S \), and \( j = 1, \ldots, 5M \). By differentiation we get for every \( i \in I_k \) and \( m = 1, \ldots, M \):
\[ H_k[i, 5m - 4] = -2\gamma_i X_k[5m] \cdot (X_k[5m - 4] - x_i) \]
\[ H_k[i, 5m - 2] = -2\gamma_i X_k[5m] \cdot (X_k[5m - 2] - y_i) \]
\[ H_k[i, 5m] = \gamma_i d^2_{kim}, \]

with \( d^2_{kim} = (X_k[5m - 4] - x_i)^2 + (X_k[5m - 2] - y_i)^2 \). The remaining elements of \( H_k \) are zero.

In the next section we illustrate the performance of the PF for integrated detection and tracking and we compare its error performance to the derived CRB.

V. NUMERICAL RESULTS

A. Simulation setup and a single run

In the simulation we consider a sensor network with \( S = 64 \) sensors for convenience uniformly placed on a grid (the placement can be arbitrary). We consider two moving objects traversing a surveillance region spanning from \( x_{\text{min}} = 0 \) to \( x_{\text{max}} = 200m \) in \( x \) direction and \( y_{\text{min}} = 0 \) to \( y_{\text{max}} = 200m \) in \( y \) direction. The true initial states (at time \( k = 0 \)) are:
\[ x_{0,1} = [310m, -3.3m/s, 260m, -2.2m/s, 80] \] for object 1 and \( x_{0,2} = [40m, 0.45m/s, 240m, -2.15m/s, 90] \) for object 2 (object 2 is louder). All sensors are assumed to have identical gains \( \gamma_i = 1 \) and standard deviations \( \sigma_i = 0.08 \). Suppose an object of acoustic intensity \( \alpha_0 \) is in the sensor field. Then
we can define the signal-to-noise ratio (SNR) for a sensor receiving the sound from this object as follows:

\[
\text{SNR}_i \text{[dB]} = 20 \log \frac{\left(\frac{\alpha_0}{d^2}\right)}{\sigma_i},
\]

where \(d\) is the distance between them. With adopted values in the simulation setup, the SNR (in dB) as a function of the radial distance for both objects is shown in Fig.1 (i.e. \(\alpha_0 = 80\) for object 1 and \(\alpha_0 = 90\) for object 2). We note that for both objects the SNR drops to about 20 dB at a distance of 10 m and to 0 dB at a distance of 30 m. The plot in Fig.1 provides an indication of the sensing range of each sensor in this example.

In simulations we assume that all sensors acquire measurements synchronously (although this is not necessary), i.e. \(T_k = \{1, \ldots, S\}\), for \(k = 1, 2, \ldots\). The acquired measurements are sent to the central processor with the sampling interval of \(T_k = 1.5\) s. There are in total 90 observation time steps. Fig.2 shows the tracking performance of the PF: (a) displays the established tracks at time \(k = 29\), with the placement of sensors indicated by the green dots; (b) illustrates the true object trajectories (blue line for object 1 and red line for object 2), with overlayed estimated tracks (the result of the particle filter). The implemented PF was designed to track up to \(M^* = 2\) targets, with the TPM given by:

\[
\Pi = \begin{bmatrix}
(1 - P_b) & P_b & 0 \\
P_d & (1 - P_b - P_d) & P_b \\
0 & P_d & (1 - P_d)
\end{bmatrix}
\]

Here \(P_b\) and \(P_d\) are the tuning parameters which represent the probability of object “birth” and “death”, respectively. In simulations we set \(P_b = P_d = 0.1\). Process noise parameters were set to \(q_x = q_y = 0.2\) and \(q_I = 0.3\). The birth density \(p_b(x)\) is a uniform density across the entire surveillance region, with \(x_{\text{min}} = y_{\text{min}} = 0\) to \(x_{\text{max}} = y_{\text{max}} = 200\) m, \(v_{\text{max}} = 5\) m/s, \(\alpha_{\text{min}} = 50\) and \(\alpha_{\text{max}} = 120\). Fig.3 shows the probabilities \(P_M\) of eq.(12) and the resulting estimate of the number of objects in the region, \(\hat{M}_k\), over time. Object 1 first enters the region at about \(k = 14\), followed by object 2 at \(k = 24\). Object 2 is moving faster, hence leaves the region first at \(k = 63\); object one exits the region at \(k = 78\). At one point of time the two objects are very close to each other, but this does not effect the performance.

We have demonstrated here that the proposed concept of integrated detection and tracking of multiple targets works. However, we point out, that for its reliable performance with only up to \(M^* = 2\) targets in the surveillance region, the PF requires in excess of 25000 particles. For larger values of \(M^*\) it would be necessary to develop a more computationally efficient implementation of the PF.
B. CRB and tracking error assessment

Next we run Monte Carlo simulations to estimate the root-mean-squared (RMS) errors of the tracks generated by the PF. These tracking RMS errors are compared with the theoretical CRB derived in Section IV. The scenario and all parameters were the same as before. The results are shown in Fig.4, (a) for target 1, and (b) for target 2. The square-root of the theoretical CRB for positional error is indicated by solid red lines. This CRB was computed using the process noise parameters of the PF, as described in Sec.V-A, while the actual target trajectories had practically zero-process noise.

The theoretical CRB curves clearly indicate the regions at the start and towards the end of the scenario, where the object parameters are difficult to estimate; this is the region where the objects are undetectable. In the middle of the observation interval, while objects are inside the surveillance region, the positional error bounds indicate that it is possible to estimate the position of each target with an error of less than about 5m. Interestingly, the positional error bound is not constant in the middle of the observation interval, because the distances between a moving target and the static sensors are time-varying; smaller distances result in more accurate estimation. The positional RMS errors of the PF were obtained by averaging over 10 Monte Carlo runs. They show a remarkable agreement with the theoretical bound.

VI. CONCLUSIONS

The paper presented a conceptual solution for integrated detection and tracking of multiple objects in the context of a sensor network for ground surveillance. The sensors are providing possibly asynchronous acoustic energy measurements, which are transmitted to the central processor for detection and tracking. The proposed algorithm has been implemented in the form of a particle filter and conceptually verified by computer simulations for up to two targets. The theoretical Cramer-Rao lower bound for multiple-target tracking in the context of this acoustic sensor network has been derived and verified by Monte Carlo simulations.

In the future we plan to focus on theoretical refinements of the proposed algorithm as well as on the experimental work, using a prototype wireless network of acoustics sensors. The theoretical part will be devoted to: (i) tracking in the presence of uncertain propagation factors $\epsilon_i$ in (1); (ii) computationally more efficient implementations of the particle filter capable of detection and tracking of more than two objects; (3) the replacement of centralised processing with a suitable distributed scheme.

REFERENCES

Fig. 4. Tracking RMS errors (position): (a) target 1; (b) target 2. Red solid lines are the theoretical lower bound; blue thin line is obtained by Monte Carlo runs of the PF.


