PARTICLE FILTER FOR JOINT ESTIMATION OF MULTI-OBJECT DYNAMIC STATE AND MULTI-SENSOR BIAS

Branko Ristic

ISR Division
DSTO
Melbourne
Australia
branko.ristic@dsto.defence.gov.au

Daniel Clark

EECE EPS
Heriot Watt University
Edinburgh
United Kingdom
d.e.clark@hw.ac.uk

ABSTRACT

The paper formulates the problem of sequential Bayesian estimation of a compound state consisting of a multi-object dynamic state and a multi-sensor bias. The compound state is modelled by a doubly stochastic point process, where the multi-object bias is a parent, whereas the multi-object state is the offspring point process. The prediction and the update steps for the first-order moment of the posterior density of the doubly-stochastic point process can be expressed analytically. The implementation, however, in general has to be done numerically. The paper presents a particle filter implementation illustrated in the context of multi-target tracking using range-azimuth measuring sensors with unknown biases.

Index Terms— Bayesian estimation, random sets, sensor bias, sensor registration, multi-target tracking

1. INTRODUCTION

Calibration of sensors for cooperative networked surveillance is a crucial prerequisite for successful network deployment. Consequently numerous techniques for self-localisation and self-calibration of sensor network have been proposed, e.g. [1, 2]. These techniques, however, are never exact, meaning that the residual sensor biases remain. The problem of sensor bias estimation is particularly important in target tracking, because excessive registration errors can lead to the formation of multiple "ghost" tracks on the same target. A review of multi-sensor multi-target bias estimation approaches is presented in [3]. Most of the sensor registration methods rely on assumptions such as: sensors are synchronous, detection is perfect (no missed or false detections) and the association of multi-sensor detections has been carried out as a pre-processing step.

In order to relax these assumptions it is necessary to formulate the problem as a sequential Bayesian estimation problem on the joint space of *multi-target state* and *multi-sensor bias*. Both the multi-target state and the multi-sensor bias need to be modelled as stochastic dynamic systems, where the multi-target state is characterized by a time varying number of objects (targets). The sensor measurements are typically affected by measurement noise in addition to sensor biases, but also suffer from imperfect detection. An early attempt to formulate a general solution in this framework was [4]. However, the analytic formulation of [4] applies to a single permanently existing target. In order to deal with multiple targets, the method in [4] requires sensor level trackers whose output is ultimately fused via track association and fusion. More recently the Bayesian unified registration and tracking (BURT) was formulated in its most general form using the random set framework [5]. Furthermore the BURT-PHD filter has been proposed as a computationally tractable solution, although it has not been implemented or tested.

In this paper we adopt the same framework as in [5]. In order to formulate a tractable solution, however, we model the joint *multi-target–multi-sensor bias* state by a doubly-stochastic (or cluster) point process [6, Ch.6], where the bias vector is the parent whereas the multi-target state is the offspring (daughter) point process. Moreover, since the multi-sensor bias is of a constant size (i.e. it is a random vector), a special case of the *single-cluster point process* [7] is applicable in this case. The paper formulates the theoretical Bayesian solution, proposes a particle filter based implementation and demonstrates its performance by a numerical example.

2. PROBLEM FORMULATION

Let the target state space be denoted by $\mathcal{X} \subseteq \mathbb{R}^{n_x}$. Targets can appear or disappear anywhere and anytime in $\mathcal{X}$. Suppose that the number of objects that exist in the state space $\mathcal{X}$ at time t_k is denoted by ν_k. The multi-object target state can conveniently be represented by a finite random set (RFS) [8]

$$\mathbf{X}_k = \{\mathbf{x}_{k,1}, \dots, \mathbf{x}_{k,\nu_k}\} \in \mathcal{F}(\mathcal{X}) \tag{1}$$

where $\mathcal{F}(\mathcal{X})$ is the set of finite subsets of $\mathcal{X}$. A RFS $\mathbf{X}_k$ is a random variable that take values as unordered finite sets, with both the number of elements in $\mathbf{X}_k$ and their spatial position in $\mathcal{X}$ being random. Individual target dynamics is modelled by a Markov transition density from t_{k-1} to t_k, denoted $\pi_{k|k-1}(\mathbf{x}|\mathbf{x}')$. The probability that a target with state $\mathbf{x}'$ at t_{k-1} will survive until time t_k is denoted $p_S(\mathbf{x}') \stackrel{\text{abbr}}{=} p_{S,k|k-1}(\mathbf{x}')$. Finally, target birth between t_{k-1} and t_k is modelled by a Poisson RFS, characterised by intensity function $\gamma_{k|k-1}(\mathbf{x})$.

Suppose R sensors with overlapping coverage provide measurements of the targets in $\mathcal{X}$ by taking values in the measurement space $\mathcal{Z} \subseteq \mathbb{R}^{n_z}$. Sensor measurements are affected by two sources of error: the systematic errors (or biases) and the stochastic zero-mean additive noise. Furthermore, due to the imperfections in the detector, objects are typically detected with the probability of detection less than 1, whereas the spurious (false) detections are also reported. The set of measurements provided at time t_k by sensor $r_k \in \{1, \dots, R\}$ can also be conveniently represented by a RFS:

$$\mathbf{Z}_k^{(r_k)} = \{\mathbf{z}_{k,1}^{(r_k)}, \dots, \mathbf{z}_{k,m_k^{(r_k)}}^{(r_k)}\} \in \mathcal{F}(\mathcal{Z}) \tag{2}$$

978-1-4673-0046-9/12/$26.00

where $\mathcal{F}(\mathcal{Z})$ is the set of finite subsets of $\mathcal{Z}$ and in general $\nu_k \neq m_k^{(r_k)}$. For measurement $\mathbf{z}_k^{(r_k)} \in \mathbf{Z}_k^{(r_k)}$, which originates from target $\mathbf{x}_k \in \mathbf{X}_k$, the measurement equation can be expressed as:

$$\mathbf{z}_k^{(r_k)} = h_k^{(r_k)}(\mathbf{x}_k) + \beta_k^{(r_k)} + \mathbf{v}_k^{(r_k)} \tag{3}$$

where $h_k^{(r)} : \mathcal{X} \to \mathcal{Z}$ is the (possibly) nonlinear measurement function of sensor r; $\beta_k^{(r_k)} \in \mathcal{B} \subseteq \mathbb{R}^{n_z}$ is the bias vector of sensor r_k at t_k and $\mathbf{v}_k^{(r_k)}$ is zero-mean additive white noise with probability density function $p_\mathbf{v}$. The likelihood function of target-originated measurement $\mathbf{z}$ can then be written as $g_k^{(r)}(\mathbf{z}|\mathbf{x}, \beta_k^{(r)}) = p_\mathbf{v}(\mathbf{z} - h_k^{(r)}(\mathbf{x}) - \beta_k^{(r)})$. False detections at k are modelled by a Poisson RFS, characterised by intensity function $\kappa_k^{(r)}(\mathbf{z}) = \lambda^{(r)}\, c^{(r)}(\mathbf{z})$. Finally we point out that sensors operate asynchronously with a non-uniform sampling interval $T_k = t_k - t_{k-1}$.

The problem is sequential Bayesian estimation of the posterior PDF of the joint multi-target state $\mathbf{X}_k$ and the R-sensor bias vector $\boldsymbol{\beta}_k = \left[\left(\beta_k^{(1)}\right)^\intercal, \ldots, \left(\beta_k^{(R)}\right)^\intercal\right]^\intercal$. This posterior at time k is denoted by $p_k(\mathbf{X}_k, \boldsymbol{\beta}_k|\mathbf{Z}_{1:k})$ where $\mathbf{Z}_{1:k}$ is a shortened notation for the sequence of measurement sets received up to time k, i.e. $\mathbf{Z}_{1:k} \equiv (\mathbf{Z}_1^{(r_1)}, \ldots, \mathbf{Z}_k^{(r_k)})$.

3. BAYESIAN SOLUTION

Joint state $\mathbb{X} = (\boldsymbol{\beta}, \mathbf{X})$ represents a single-cluster point process, where $\boldsymbol{\beta}$ is the *parent* state vector (the multi-sensor bias) and $\mathbf{X}$ is its daughter point process. Assume the posterior density of the joint state at time t_{k-1}, denoted $p_{k-1|k-1}(\mathbb{X}|\mathbf{Z}_{1:k-1})$, is known. The sequential Bayesian estimator of $\mathbb{X}$ propagates the posterior density to the next time k in two steps, prediction and update [8]:

$$\begin{aligned} p_{k|k-1}(\mathbb{X}|\mathbf{Z}_{1:k-1}) &= \int \pi_{k|k-1}(\mathbb{X}|\mathbb{X}')\, p_{k-1|k-1}(\mathbb{X}|\mathbf{Z}_{1:k-1})\delta\mathbb{X}' \\ p_{k|k}(\mathbb{X}|\mathbf{Z}_{1:k}) &= \frac{g_k(\mathbf{Z}_k|\mathbb{X})\, p_{k|k-1}(\mathbb{X}|\mathbf{Z}_{1:k-1})}{\int g_k(\mathbf{Z}_k|\mathbb{X})\, p_{k|k-1}(\mathbb{X}|\mathbf{Z}_{1:k-1})\delta\mathbb{X}.} \end{aligned}$$

Here $\pi_{k|k-1}(\mathbb{X}|\mathbb{X}')$ represents the single-cluster Markov transition density and $g_k(\mathbf{Z}|\mathbb{X})$ is the single-cluster likelihood. The integrals above are set-integrals and the closed form solution for the two equations above is in general intractable. Instead we restrict to the solution for *the first order moment* of the posterior density.

Suppose the first order moment of the posterior density at t_{k-1}, $p_{k-1|k-1}(\mathbb{X}|\mathbf{Z}_{1:k-1})$, is available and denoted

$$D_{k-1|k-1}(\boldsymbol{\beta}, \mathbf{x}) = s_{k-1|k-1}(\boldsymbol{\beta}) \cdot D_{k-1|k-1}(\mathbf{x}|\boldsymbol{\beta}) \tag{4}$$

where

- $s_{k-1|k-1}(\boldsymbol{\beta})$ is the posterior density of the parent at t_{k-1} and
- $D_{k-1|k-1}(\mathbf{x}|\boldsymbol{\beta})$ is the posterior *intensity* function of the daughter process conditioned on the parent at t_{k-1}.

The first order moment of the predicted density $p_{k|k-1}(\boldsymbol{\beta}, \mathbf{X}|\mathbf{Z}_{1:k-1})$ can be expressed as [7]:

$$D_{k|k-1}(\boldsymbol{\beta}, \mathbf{x}) = \int s_{k-1|k-1}(\boldsymbol{\beta}')\pi_{k|k-1}(\boldsymbol{\beta}|\boldsymbol{\beta}')\tilde{D}_{k|k-1}(\mathbf{x}|\boldsymbol{\beta}')d\boldsymbol{\beta}' \tag{5}$$

where

$$\tilde{D}_{k|k-1}(\mathbf{x}|\boldsymbol{\beta}') = \gamma_{k|k-1}(\mathbf{x}|\boldsymbol{\beta}') + \int p_S(\mathbf{x}'|\boldsymbol{\beta}')\pi_{k|k-1}(\mathbf{x}|\mathbf{x}';\boldsymbol{\beta}')D_{k-1|k-1}(\mathbf{x}'|\boldsymbol{\beta}')d\mathbf{x}' \tag{6}$$

and $\pi_{k|k-1}(\boldsymbol{\beta}|\boldsymbol{\beta}')$ is the Markov transition density for the parent (bias). Other terms in (6) have been defined in Sec.2, with the only difference that now they are conditioned on the sensor bias vector at t_{k-1}, that is $\boldsymbol{\beta}'$.

Suppose that at time t_k sensor r_k reports its measurement set $\mathbf{Z}_k^{(r_k)}$. The update step will then affect only the component $\beta^{(r_k)}$ of the sensor bias vector $\boldsymbol{\beta}$. For this reason only the "reduced" predicted intensity function $D_{k|k-1}(\beta^{(r_k)}, \mathbf{x})$ of (5) needs to be updated. Keeping this in mind, but for the sake of a simplified notation, we will use notation β instead of $\beta^{(r_k)}$ in equations below.

The first order moment of the updated density $p_{k|k}(\beta, \mathbf{X}|\mathbf{Z}_{1:k})$ can be derived under the assumption that the predicted state is a single-cluster with a Poisson daughter. The updated intensity function is then [7]

$$D_{k|k}(\beta, \mathbf{x}) = s_{k|k}(\beta) \cdot D_{k|k}(\mathbf{x}|\beta) \tag{7}$$

where

$$s_{k|k}(\beta) = \frac{s_{k|k-1}(\beta)L_{\mathbf{Z}_k}(\beta)}{\int s_{k|k-1}(\beta)L_{\mathbf{Z}_k}(\beta)d\beta} \tag{8}$$

$$D_{k|k}(\mathbf{x}|\beta) = (1 - p_D(\mathbf{x}|\beta))D_{k|k-1}(\mathbf{x}|\beta) + \sum_{\mathbf{z}\in\mathbf{Z}_k} \frac{p_D(\mathbf{x}|\beta)D_{k|k-1}(\mathbf{x}|\beta)g_k(\mathbf{z}|\mathbf{x};\beta)}{\kappa_k(\mathbf{z}) + \int p_D(\mathbf{x}|\beta)D_{k|k-1}(\mathbf{x}|\beta)g_k(\mathbf{z}|\mathbf{x};\beta)d\mathbf{x}} \tag{9}$$

and $L_{\mathbf{Z}_k}(\beta)$ is the multi-target likelihood:

$$L_{\mathbf{Z}_k}(\boldsymbol{\beta}) = \exp\{-\int p_D(\mathbf{x}|\beta)D_{k|k-1}(\mathbf{x}|\beta)d\mathbf{x}\}\times \prod_{\mathbf{z}\in\mathbf{Z}_k}\left(\kappa_k(\mathbf{z}) + \int p_D(\mathbf{x}|\beta)g_k(\mathbf{z}|\mathbf{x};\beta)D_{k|k-1}(\mathbf{x}|\beta)d\mathbf{x}\right) \tag{10}$$

Note that $g_k(\mathbf{z}|\mathbf{x};\beta)$, $p_D(\mathbf{x}|\beta)$ and $\kappa_k(\mathbf{z})$, which feature in (9) and (10), have been defined in Sec.2. Here, however, we explicitly express their conditioning on the bias of sensor r_k.

Equations (6) and (9) represent the prediction and update equations of the standard PHD filter [9], respectively, for the daughter process (multi-target state) conditioned on the parent (sensor bias).

4. PARTICLE FILTER

The sequential Bayesian estimator described in Sec.3 can be solved only numerically and we adopt the particle filter implementation [10, 11] for this purpose. The particle approximation of the posterior PDF of the parent $\boldsymbol{\beta}$ (multi-sensor bias vector) at $k-1$ can be expressed by:

$$\widehat{s}_{k-1|k-1}(\boldsymbol{\beta}) = \sum_{i=1}^{M} w_{k-1}^i \cdot \delta_{\boldsymbol{\beta}_{k-1}^i}(\beta) \tag{11}$$

where $\boldsymbol{\beta}_{k-1}^i$ is a random sample (particle) from $\mathcal{B}^R$, w_{k-1}^i is the sample weight satisfying $\sum_{i=1}^M w_{k-1}^i = 1$ and $\delta_y(x)$ denotes the Dirac delta function focused at y. Conditioned on each parent particle $\boldsymbol{\beta}_{k-1}^i$, the approximation of intensity function $D_{k-1|k-1}(\mathbf{x}|\boldsymbol{\beta}_{k-1}^i)$ by a particle set $\{(q_{k-1}^{j|i}, \mathbf{x}_{k-1}^{j|i})\}_{j=1}^{N_{k-1}}$ as follows:

$$\widehat{D}_{k-1|k-1}(\mathbf{x}|\boldsymbol{\beta}_{k-1}^i) = \sum_{j=1}^{N_{k-1}} q_{k-1}^{j|i} \cdot \delta_{\mathbf{x}_{k-1}^{j|i}}(\mathbf{x}) \tag{12}$$

Here $\hat{\nu}^i_{k-1} = \sum_{j=1}^{N_{k-1}} q^{j|i}_{k-1}$ represents the conditional estimate of the cardinality of the multi-object state $\mathbf{X}_{k-1}$ (i.e. an estimate of the number of targets at t_{k-1}). The number of particles used in approximation of intensity function (12) varies with time because we use a fixed number of particles per target (and the number of targets is time-varying).

The intensity function on the joint space $(\boldsymbol{\beta}, \mathbf{X})$ at $k-1$ then is represented by an $M \times N_{k-1}$ particle set:

$$\mathcal{P}_{k-1} = \left\{ \left(w^i_{k-1}, \boldsymbol{\beta}^i_{k-1}, \left\{ \left(q^{j|i}_{k-1}, \mathbf{x}^{j|i}_{k-1} \right) \right\}_{j=1}^{N_{k-1}} \right) \right\}_{i=1}^{M} \quad (13)$$

The steps of the particle filter that implements the Bayes estimator of Sec.3 are given in Algorithm 1. The implementation is based on multi-stage tempering [12, p.540], i.e. with the multi-target likelihood raised to $\gamma_s \in (0,1)$ at stage $s = 1, \ldots, S$, such that $\sum_{s=1}^{S} \gamma_s = 1$ [13]. This facilitates the move of bias-particles $\boldsymbol{\beta}^i_k$, $i = 1, \ldots, M$, towards the relevant region of the multi-sensor bias space $\mathcal{B}^R$.

Algorithm 1 The steps of the particle at time t_k

1: Input: $\mathcal{P}_{k-1}$, $\mathbf{Z}^{(r_k)}_k$
2: Copy: $\boldsymbol{\beta}^i_k \leftarrow \boldsymbol{\beta}^i_{k-1}$, for $i = 1, \ldots, M$
3: $\bar{\beta}^{i,0}_k \sim \pi(\beta | \beta^{(r_k),i}_{k-1})$ for $i = 1, \ldots, M$ ▷ Compon. r_k of $\boldsymbol{\beta}^i_{k-1}$
4: **for** $s = 1, \ldots, S$ **do**
5: **for** $i = 1, \ldots, M$ **do**
6: $\beta^{i,s,*}_k \leftarrow \bar{\beta}^{i,s-1}_k$
7: Estimate $\widehat{D}_{k|k}(\mathbf{x}|\beta^{i,s,*}_k)$ of (12) using $\mathbf{Z}^{(r_k)}_k$
8: Compute $\tilde{w}^i_k = L_{\mathbf{Z}_k}(\beta^{i,s,*}_k)^{\gamma_s}$ using (10)
9: **end for**
10: Normalise weights: $w^i_k = \tilde{w}^i_k / \sum_{i=1}^{M} \tilde{w}^i_k$ for $i = 1, \ldots, M$
11: Find $i_{\max} = \max_{i=1,\ldots,M} w^i_k$
12: Resample from $\{w^i_k, \beta^{i,s,*}_k\}_{i=1}^{M}$ to obtain $\{\frac{1}{M}, \boldsymbol{\beta}^{i,s}_k\}_{i=1}^{M}$
13: MCMC move step: $\bar{\beta}^{i,s}_k \leftarrow \beta^{i,s}_k$ for $i = 1, \ldots, M$
14: **end for**
15: $\beta^{(r_k),i}_k \leftarrow \bar{\beta}^{i,S}_k$ for $i = 1, \ldots, M$ ▷ Component r_k of $\boldsymbol{\beta}^i_k$
16: Estimate multi-object state $\widehat{\mathbf{X}}_k$ from $\widehat{D}_{k|k}(\mathbf{x}|\beta^{i_{\max},S,*}_k)$
17: Estimate multi-sensor bias $\hat{\boldsymbol{\beta}}_k = \frac{1}{M}\sum_{i=1}^{M} \boldsymbol{\beta}^i_k$;
18: Report $\hat{\mathbf{X}}_k$, $\hat{\boldsymbol{\beta}}_k$
19: Output for the next time step: $\mathcal{P}_k$

Line 3 of Algorithm 1 predicts the sensor bias particle, and since the bias is typically slowly varying, we adopt a random walk model. The for-loop from lines 5 to 9 computes for each bias (parent) particle the corresponding intensity function, which effectively involves running a particle PHD filter. Line 7 is therefore a non-trivial step with full explanation given in [14],[15]. There are two types of particles in the particle PHD filter, the persistent and newborn target particles. The un-normalised weight of each parent particle is computed in line 8, while the weights are normalised in line 10. Line 11 finds the parent particle with the highest weight. The index of this particle will be used in line 16 to find the maximum a posteriori estimate of the intensity function. The parent particles are resampled in line 12 to obtain a new set of equally weighted parent particles, followed by Metropolis-Hastings move step [10, p.55] in line 13. The for-loop from line 4 to 14 is repeated for a sequence of factors $\gamma_s \in (0,1)$, $s = 1, \ldots, S$. Line 16 estimates the multi-object state $\widehat{\mathbf{X}}_k$ from the intensity function approximated by particles as in (12). This step does not involve clustering of particles (as suggested by numerous authors): a statistically and computationally efficient method is presented in [14]. The estimate of the multi-sensor bias vector $\hat{\boldsymbol{\beta}}_k$ is computed as the sample mean in line 17. At the end of each processing cycle, the algorithm reports (line 18) the estimates of multi-object state $\widehat{\mathbf{X}}_k$ and multi-sensor bias $\hat{\boldsymbol{\beta}}_k$.

5. NUMERICAL RESULTS

The algorithm is demonstrated using a 2D scenario shown in Fig.1. The measurements are collected over $k = 1, \ldots, 12$ scans using $R = 2$ (static) sensors, reporting their measurements alternatively (i.e. 6 scans of detections from each sensor). The sensors are located at $(0,0)$m (measurements shown in red) and $(120, 35)$m (measurements in green). There are six targets, of which four appear at $k = 1$ and the remaining two at $k = 3$. All targets are moving in the plane with a nearly constant velocity motion. The state vector of each individual target is $\mathbf{x} = [x \quad \dot{x} \quad y \quad \dot{y}]^\intercal$, where (x, y) denotes target position and $(\dot{x}, \dot{y})$ its velocity. Both sensors measure target range and azimuth, that is

$$h^{(r)}_k(\mathbf{x}) = \begin{bmatrix} \sqrt{(x - x_r)^2 + (y - y_r)^2} \\ \arctan\left(\frac{x - x_r}{y - y_r}\right) \end{bmatrix}, \qquad r = 1, 2, \quad (14)$$

where (x_r, y_r) is the position of sensor r. The sensor bias vectors $\beta^{(r)}_k = [\Delta\rho_r \quad \Delta\theta_r]^\intercal$, for $r = 1, 2$, initially take values $\Delta\rho_1 = 7.2$m, $\Delta\rho_2 = -5$m, $\Delta\theta_1 = -3.5^\circ$, $\Delta\theta_2 = 2^\circ$ and subsequently drift away. Measurement noise $\mathbf{v}^{(r)}_k$ is white zero-mean Gaussian, i.e. $\mathbf{v}^{(r)}_k \sim \mathcal{N}(\mathbf{v}; \mathbf{0}, \Sigma^{(r)}_k)$, where $\Sigma^{(r)}_k$ is a diagonal matrix with square-root diagonal elements $\sigma^{(r)}_\rho = 0.1$m and $\sigma^{(r)}_\theta = 0.2^\circ$, for $r = 1, 2$. The probability of detection is $p^{(1)}_D = p^{(2)}_D = 0.95$; the mean number of false detections per scan per sensor is $\lambda^{(1)} = \lambda^{(2)} = 10$. The false detections are uniformly distributed in range, from 10m to 300m and in azimuth from $-\pi$ to π radians.

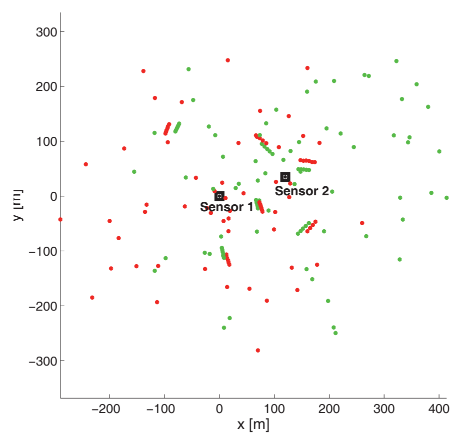

Fig. 1. Testing scenario involving two sensors and six targets

The particle filter was implemented using 75 particles for each newborn target (each new measurement is potentially a newborn target) and 300 particles for each persistent target. The number of bias vector particles was $M = 625$. Fig. 2 shows the estimates $\widehat{\boldsymbol{\beta}}_k$ over time on one particular run of the algorithm. The black solid lines are the true values, the red and green lines represent the estimates. The particle filter approximation of the posterior density of the multi-sensor bias, $\widehat{s}_{k|k}(\boldsymbol{\beta})$ of (11), is shown in Fig.3 at $k = 6$ and $k = 12$. The true values for sensor 1 and 2 are indicated by asterisks. Figs. 2 and 3 confirm that the particle filter estimates the multi-sensor bias correctly.

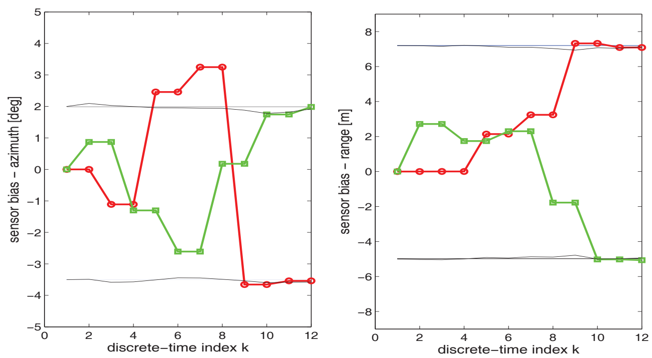

Fig. 2. True and estimated multi-sensor bias vector $\widehat{\boldsymbol{\beta}}_k$ over time

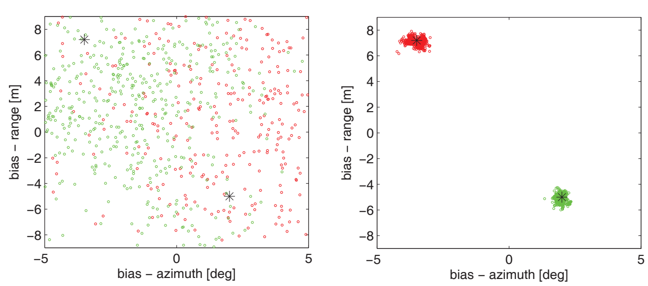

Fig. 3. Particle approximation of the posterior density of the multi-sensor bias, $\widehat{s}_{k|k}(\boldsymbol{\beta})$, at $k = 6$ (left) and $k = 12$ (right).

The accuracy of the multi-object state estimate $\widehat{\mathbf{X}}_k$ was measured using the OSPA error metric [16], with parameters $p = 1$ and $c = 5$m. By choosing $p = 1$, the OSPA error can be interpreted as a sum of the cardinality estimate error and the object localisation error. The cut-off parameter c was chosen as a measure of penalty (expressed in meters) for cardinality error. Fig.4 shows the OSPA error over time: initially it is dominated by cardinality error, but as the sensor bias estimates become more accurate, both the cardinality and localisation errors become smaller.

6. CONCLUSIONS

The paper presented a particle filter implementation of the sequential Bayesian estimator of a doubly-stochastic point process, where the parent process is the sensor bias vector and the offspring processes is the multi-target state. Essentially for each bias-particle it is required to compute a particle PHD (intensity function) whose likelihood represented by (10) is evaluated for the purpose of importance weights update. Numerical results demonstrate reliable performance. The proposed technique is applicable to appearing/disappearing targets, asynchronous sensors with possibly imperfect detection and without a need for prior multi-sensor measurement association.

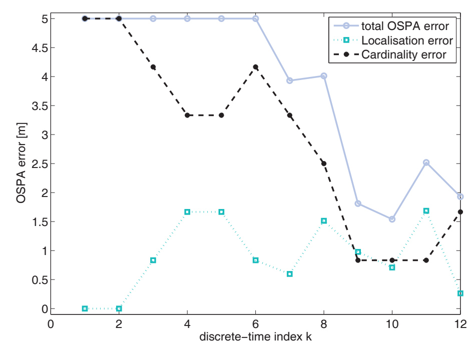

Fig. 4. OSPA error over time:

7. REFERENCES

[1] A. Ihler, J. Fisher, J. W. Fisher, A. Willsky, and R. Moses, “Nonparametric belief propagation for self-calibration in sensor networks,” in *In Proc. 3rd Int. Symp. Information Processing in Sensor Networks*, 2004, pp. 225–233.

[2] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal, “Locating the nodes,” *IEEE Signal Processing Magazine*, pp. 54–68, July 2005.

[3] X. Lin, Y. Bar-Shalom, and T. Kiribarajan, “Multisensor-multitarget bias estimation for general asynchronous sensors,” *IEEE Trans Aerospace and Electronic Syst.*, , no. 3, pp. 899–921, 2005.

[4] N. Okello and S. Challa, “Joint sensor registration and track-to-track fusion,” *IEEE Trans Aerospace and Electronic Systems*, vol. 40, no. 3, 2004.

[5] R. Mahler and A. El-Fallah, “Bayesian unified registration and tracking,” in *Proc. SPIE*, 2011, vol. 8050.

[6] D. J. Daley and D. Vere-Jones, *An Introduction to the Theory of point processes*, Springer, 2nd edition, 2003.

[7] A. Swain and D. Clark, “The single-group PHD filter: an analytic solution,” in *Proc. Int. Conf. Information Fusion*, July 2011.

[8] R. Mahler, *Statistical Multisource Multitarget Information Fusion*, Artech House, 2007.

[9] R. P. S. Mahler, “Multi-target Bayes filtering via first-order multi-target moments,” *IEEE Trans. Aerospace & Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, 2003.

[10] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman filter: Particle filters for tracking applications*, Artech House, 2004.

[11] A. Doucet and A. M. Johansen, “A tutorial on particle filtering and smoothing: Fifteen years later,” Tech. Rep., Department of Statistics, University of British Columbia, Dec. 2008.

[12] C. P. Robert and G. Casella, *Monte Carlo statistical methods*, Springer, 2nd edition, 2004.

[13] N. Oudjane and C. Musso, “Progressive correction for regularized particle filters,” in *Proc. 3rd Int. Conf. Information Fusion*, Paris, France, 2000.

[14] B. Ristic, D. Clark, and B.-N. Vo, “Improved SMC implementation of the PHD filter,” in *Proc. 13th Int. Conf. Information Fusion*, Edinburgh, UK, July 2010.

[15] B. N. Vo, S. Singh, and A. Doucet, “Sequential Monte Carlo methods for multi-target filtering with random finite sets,” *IEEE Trans. Aerospace & Electronic Systems*, vol. 41, no. 4, pp. 1224–1245, Oct. 2005.

[16] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, “A consistent metric for performance evaluation of multi-object filters,” *IEEE Trans. Signal Processing*, vol. 56, no. 8, pp. 3447–3457, Aug. 2008.