A Sequential Monte Carlo Method for Target Tracking in an Asynchronous Wireless Sensor Network

Mahesh Vemula Joaquin Miguez Antonio Artes-Rodriguez

4th Workshop on Positioning, Navigation and Communication , March 2007, Volume 4, July 2006, pp. V – V.

Stable URL:
http://ieeexplore.ieee.org/iel5/4167798/4167799/04167818.pdf?tp=&isnumber=4167799&arnumber=4167818

A Sequential Monte Carlo Method for Target Tracking in an Asynchronous Wireless Sensor Network

Mahesh Vemula*, Joaquín Míguez† and Antonio Artés-Rodríguez†

*Department of Electrical and Computer Engineering, State University of New York at Stony Brook, USA
Email: vema@ece.sunysb.edu

†Departamento de Teoría de la Señal y Communicaciones, Universidad Carlos III de Madrid, Madrid, Spain,
Email: {jmiguez,antonio}@tsc.uc3m.es

***Abstract*— Target tracking in a wireless sensor network (WSN) has become a relatively standard problem. The WSN typically consists of a collection of sensor nodes, which acquire physical data related to the target dynamics, and a fusion center (FC) where the available data are processed together to sequentially estimate the target state (its instantaneous location and velocity). Very often, tracking algorithms are designed under the assumption that the network is synchronous, i.e., that the local clocks of the sensor nodes and the FC are perfectly aligned or, at least, that their offsets are known. In this paper, we consider an asynchronous WSN, in which the local clocks of the sensors are misaligned and the corresponding offsets are unknown, and aim at designing recursive algorithms for optimal (Bayesian) tracking. In particular, we propose sequential Monte Carlo (SMC) techniques that enable the approximation of the joint posterior probability distribution of the target state and the set of local clock offsets by means of a discrete probability measure with a random support. From this approximation, estimates of the target position and velocity, as well as of the clock offsets, can be readily derived. We illustrate the validity of the proposed approach and assess the performance of the resulting algorithms by means of computer simulations.**

I. Introduction

Wireless sensor networks (WSN) provide us with many interesting opportunities and challenges for sensing and monitoring. Due to recent advances in micro-fabrication, small and cheap sensors can be deployed and provide us with large amounts of physical data related to objects of interest in the environment surrounding them. In particular, many important applications of WSNs, both civilian and military, include the online tracking of randomly maneuvering targets.

A. Motivation

Wireless micro-sensors are transducers which “measure” a physical phenomenon and convert it into an information-bearing electrical signal through sampling, filtering and calibration. The latter, as well as other important steps of signal processing, depend on the internal clocks of the sensors. In the application of WSNs for target tracking, it is often assumed that the clocks of the sensors and the fusion center (FC) are synchronous, i.e., that all sensors sample the phenomenon of interest at the same instants or, alternatively, that the sampling times are possibly different but perfectly known at the FC. However, a misalignment in the clocks of the sensors and the FC due to the inherent drift in the clock frequencies occurs in practice [1]. Estimating and compensating the timing offsets at the sensors using time synchronization protocols results in a significant increase of the communication overhead in the WSN and, as a consequence, in an undesired reduction of the life of all battery-supported nodes [2]. In this paper, we tackle the problem of tracking a target when the sensors are asynchronous *and* their offsets are unknown.

B. Brief Literature Survey

The issue of handling the asynchronous activity of sensors has been addressed in [3], in the context of a multisensor-multitarget bias estimation problem, but assuming that all timing offsets are known. Similarly, in [4] sensor registration is performed using Kalman filtering for asynchronous sensors, but the misalignments of the sensor clocks are assumed known, too. In [5], the authors consider a localization system with asynchronous sensors and an object that periodically transmits a known signal. The inter-arrival time between the received signals is approximated and modeled as being statistically independent of the clock offsets, in order to enable localization using standard maximum-likelihood estimation (MLE) techniques.

In this paper, we propose to use the sequential Monte Carlo (SMC) methodology [6] to jointly estimate the sensor offsets and the target trajectory and velocity. The problem is formally modeled as the joint Bayesian estimation of a set of static parameters and the time-varying state of a discrete-time random dynamical system. The SMC approach consists in approximating the posterior probability distribution of the random signals of interest using a discrete probability measure with random support, which enables the straightforward com-

This work was supported by *Comunidad Autónoma de Madrid* (project PRO-MULTIDIS-CM, ref. S0505/TIC/0223), *Ministerio de Educación y Ciencia* of Spain (projects DOIRAS, ref. TIC2003-02602, and MONIN, ref. TEC2006-13514-C02-01) and the EC (Network of Excellence ‘CRUISE’, ref. IST-4-027738).

putation of estimates. This is not a simple task for our problem, though, since it is hard to guarantee convergence (to optimal solutions) of conventional SMC algorithms when static and dynamic random magnitudes must be handled together. The joint (static) parameter and (dynamic) state estimation problem has been previously addressed in [7], [8], [9], [10], [11]. In [7], an artificial evolution of the static parameters is proposed. In [8], the evolution of the parameters using kernel methods is described. A sampling scheme for fixed parameters that imposes restrictive assumptions on the probabilistic model is proposed in [9], while in [10] point-optimization methods are proposed. Finally, in [11], "density assisted" methods, which approximate the posterior distribution of static parameters using a model probability density function (pdf), are discussed. In this paper, we propose two novel techniques, based on the general methodologies of [11] and [8].

C. Organization of Paper

In Section II, we present the signal model and describe the estimation problem that we tackle. The proposed SMC algorithms are introduced in Section III. Computer simulation results are shown in Section IV and, finally, Section V is devoted to a brief discussion of the obtained results and some concluding remarks.

II. Problem statement

We aim at recursively estimating the time-varying position and velocity of a target tracking that moves along a 2-dimensional region. The state of the target at continuous time t is $\boldsymbol{x}(t) = [x_1(t), x_2(t), \dot{x}_1(t), \dot{x}_2(t)]^\top \in \mathbb{R}^4$, where $[x_1(t), x_2(t)]^\top \in \mathbb{R}^2$ denotes the target location, $\dot{x}_i(t)$ is the time derivative of $x_i(t)$ and, therefore, $[\dot{x}_1(t), \dot{x}_2(t)]^\top \in \mathbb{R}^2$ is the target velocity vector at time t.

If the system is converted into discrete-time by sampling every T seconds (s), we obtain the dynamic state space (DSS) model,

$$\boldsymbol{x}_k = \mathrm{A}_T \boldsymbol{x}_{k-1} + \boldsymbol{u}_k, \quad k \in \mathbb{N}, \tag{1}$$

where $\mathbf{A}_T$ is a 4×4 transition matrix,

$$\boldsymbol{x}_k = [x_{1,k}, \dot{x}_{1,k}, x_{2,k}, \dot{x}_{2,k}]^\top = \boldsymbol{x}(kT)$$

is the target state at time kT (i.e., $[x_{1,k}, x_{2,k}] = [x_1(kT), x_2(kT)]^\top$ is the sampled position and $[\dot{x}_{1,k}, \dot{x}_{2,k}] = [\dot{x}_1(kT), \dot{x}_2(kT)]$ is the sampled velocity, both at time kT) and $\boldsymbol{u}_t \sim \mathcal{N}(0, \mathbf{C}_T)$ is zero-mean Gaussian noise with covariance matrix $\mathbf{C}_T$. Both the transition matrix, $\mathbf{A}_T$, and the noise covariance matrix, $\mathbf{C}_T$, are parameterized by T, specifically

$$\mathrm{A}_T = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{C}_T = \sigma_u^2 \begin{bmatrix} \frac{T^3}{3} & \frac{T^2}{2} & 0 & 0 \\ \frac{T^2}{2} & T & 0 & 0 \\ 0 & 0 & \frac{T^3}{3} & \frac{T^2}{2} \\ 0 & 0 & \frac{T^2}{2} & T \end{bmatrix}.$$

Figure 1 depicts the timing diagram for the observations collected by two sensors, whose clocks are misaligned with the FC. Specifically, the knots in the line labeled "FC" denote the time instants at which the FC collects the data, while the knots in the lines labeled "S1" and "S2" indicate the sampling instants for sensors 1 and 2, respectively. We observe that the sensors have fixed, but random and unknown, offsets $\tau_1, \tau_2 > 0$, $\tau_1 \neq \tau_2$, hence they do not sample the function of the target trajectory in the same points.

Fig. 1. Timing diagram of the clocks of two sensors and the FC.

In most target tracking applications, perfect timing is assumed (i.e., $\tau_1 = \tau_2 = 0$), which is hard to achieve in a practical situation. In this paper, we study the effect of such offsets on the tracking problem. Let us denote time intervals as $t_k = [(k-1)T, kT)$, $k \in \mathbb{N}$. The set of observations that the FC receives from the n-th sensor during interval t_k is $\mathbf{y}_{n,t_k} = [y_{n,a,t_k}, y_{n,d,t_k}, y_{n,v,t_k}]^\top$, where

$$y_{n,a,t_k} = \tan^{-1}\left(\frac{x_{1,t_k+\tau_n} - s_{n,1}}{x_{3,t_k+\tau_n} - s_{n,2}}\right) + v_{n,a,t_k} \tag{2a}$$

$$y_{n,d,t_k} = \sqrt{(x_{1,t_k+\tau_n} - s_{n,1})^2 + (x_{3,t_k+\tau_n} - s_{n,2})^2} + v_{n,d,t_k} \tag{2b}$$

$$y_{n,v,t_k} = \sqrt{(x_{2,t_k+\tau_n})^2 + (x_{4,t_k+\tau_n})^2} + v_{n,d,t_k}. \tag{2c}$$

The angle, distance and velocity observations given by (2a) (2b) and (2c) , respectively, depend on the sensor offset, τ_n, the target state at time $(k-1)T + \tau_n$, denoted as

$$\boldsymbol{x}_{t_k+\tau_n} = \mathbf{A}_{\tau_n} \boldsymbol{x}_{k-1} + \mathbf{u}_{t_k+\tau_n},$$

where $\mathbf{u}_{t_k+\tau_n} \sim N(\mathbf{0}, \mathbf{C}_{\tau_n})$, and the sensor positions, $\{s_{n,1}, s_{n,2}\}$. The measurement noise processes at the n-th sensor are denoted as v_{n,a,t_k}, v_{n,d,t_k} and v_{n,v,t_k} for angle, distance and velocity, respectively, and their pdf's are known. The complete set of observations during t_k is $\mathbf{y}_{t_k} = \{\mathbf{y}_{1,t_k}, \ldots, \mathbf{y}_{N_s,t_k}\}$, where N_s is the number of sensors in the WSN. Our objective is to estimate the sequence of target states $\boldsymbol{x}_{0:k} = \{\boldsymbol{x}_0, \ldots, \boldsymbol{x}_k\}$ and clock offsets $\tau_{1:N_s} = \{\tau_1, \ldots, \tau_{N_s}\}$ using the measurements $\mathbf{y}_{t_1:t_k} = \{\mathbf{y}_{t_1}, \ldots, \mathbf{y}_{t_k}\}$.

III. Algorithms

All the statistical information needed to optimally solve the proposed estimation problem is contained in the *a posteriori* pdf $p(\boldsymbol{x}_{0:k}, \tau_{1:N_s} | \mathbf{y}_{t_1:t_k})$. Since the observations (2a) and (2b) are nonlinear, there is no feasible (optimal) analytical solution, and we propose to resort to SMC methodology [6]. SMC methods aim at approximating the *a posteriori* pdf by means of a discrete random measure that consists of a set of weighted samples in the state space, usually termed *particles* [6]. In the problem at hand, the approximation with M particles takes the form

$$p_M(\boldsymbol{x}_{0:k}, \tau_{1:N_s}|\mathbf{y}_{t_1:t_k}) = \sum_{m=1}^{M} w_k^{(m)} \delta(\boldsymbol{x}_{0:k} - \boldsymbol{x}_{0:k}^{(m)}) \times \times \delta(\tau_{1:N_s} - \tau_{1:N_s}^{(m)}), \quad (3)$$

where $w_k^{(m)}$ is the *importance weight* associated to the sample $(\boldsymbol{x}_{0:k}^{(m)}, \tau_{1:N_s}^{(m)})$ and $\delta(\cdot)$ denotes the Dirac delta function. A SMC method is a procedure to recursively update the random measure approximation (3) when a new set of observations, $\mathbf{y}_{t_{k+1}}$, is received. It is difficult, in general, to guarantee the convergence of conventional SMC methods when there exist random (unknown) fixed parameters, because the dynamic system becomes non-ergodic [12]. In the sequel, we propose two algorithms that specifically take into account the fixed offsets, $\tau_{1:N_s}$, based on the density-assisted particle filtering (DAPF) methodology [11] and the sequential kernel approximation proposed by Liu and West [8].

A. Density Assisted Particle Filtering

The proposed SMC algorithm for the joint estimation of the target state and the sensor timing offsets is based upon the parametric approximation of the marginal posterior pdf of the n-th offset at time k by means of a beta pdf with properly chosen parameters[1], i.e., $p(\tau_n|\mathbf{y}_{t_1:t_k}) \approx \mathcal{B}(\tau_n; \pi_{n,k}, \phi_{n,k})$, which is updated, together with the importance weights, as new observations are collected. The algorithm steps are outlined below.

(i) *Intialization ($k = 0$)*: Target-state samples are drawn from the *a priori* pdf $p(\boldsymbol{x}_0)$. Offset samples are drawn from the single beta pdf $\mathcal{B}(\tau_n, 1, 1)$, i.e., we assume that, *a priori*, $p(\tau_n) = \mathcal{B}(\tau_n, 1, 1)$ for all n.

At time k, and given $\{\boldsymbol{x}_{0:k-1}^{(m)}, w_{k-1}^{(m)}\}_{m=1}^{M}$, the following recursive steps are taken.

(ii) *Particle propagation*: For each $n = \{1, 2, \cdots N_s\}$, timing offset samples are drawn from the beta-approximation of their marginal posterior pdf's at time $k-1$,

$$\tau_n^{(m)} \sim \mathcal{B}(\tau_n; \pi_{n,k-1}, \phi_{n,k-1}).$$

For each sample $m \in \{1, 2, \cdots M\}$, the offsets $\tau_{1:N_s}^{(m)}$ are sorted in the ascending order, i.e., we find a sequence $i_1, \ldots, i_{N_s}$ of distinct indices such that $i_k \in \{1, \ldots, N_s\}$ for all k and $\tau_{i_1}^{(m)} < \tau_{i_2}^{(m)} < \ldots < \tau_{i_{N_s}}^{(m)}$. State samples

$$\boldsymbol{x}_{t_k+\tau_{i_1}^{(m)}}^{(m)}, \ldots, \boldsymbol{x}_{t_k+\tau_{i_{N_s}}^{(m)}}^{(m)}$$

are drawn using the (adequately parameterized) Markov state equation (1), and we denote the resulting set of particles as $\boldsymbol{\chi}_{t_k}^{(m)} = \{\boldsymbol{x}_{t_k+\tau_{1:N_s}^{(m)}}^{(m)}, \tau_{1:N_s}^{(m)}\}$. As an example, assume $N_s = 2$ and the offset samples $\tau_1^{(m)} < \tau_2^{(m)}$. We draw the state samples at the time instants $(k-1)T + \tau_1^{(m)}$ and $(k-1)T + \tau_2^{(m)}$ as follows

$$\begin{aligned} \boldsymbol{x}_{t_k+\tau_1^{(m)}}^{(m)} &= A_{\tau_1^{(m)}} \boldsymbol{x}_{k-1}^{(m)} + \boldsymbol{u}_{t_k,1}^{(m)} \\ \boldsymbol{x}_{t_k+\tau_2^{(m)}}^{(m)} &= A_{\tau_2^{(m)}-\tau_1^{(m)}} \boldsymbol{x}_{t_k+\tau_1^{(m)}}^{(m)} + \boldsymbol{u}_{t_k,2}^{(m)} \end{aligned}$$

where $\boldsymbol{u}_{t_k,1}^{(m)} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_{\tau_1^{(m)}})$ and $\boldsymbol{u}_{t_k,2}^{(m)} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_{\tau_2^{(m)}-\tau_1^{(m)}})$.

(iii) *Weight update*: Since the noise pdf's in (1), (2a), (2b) and (2c) are assumed known, the likelihood function, $p(\mathbf{y}_{t_k}|\boldsymbol{\chi}_{t_k}^{(m)})$, can be easily evaluated and the importance weights are recursively updated as

$$\tilde{w}_k^{(m)} = w_{k-1} p(\mathbf{y}_{t_k}|\boldsymbol{\chi}_{t_k}^{(m)}), \quad (4)$$

with

$$p(\mathbf{y}_{t_k}|\boldsymbol{\chi}_{t_k}^{(m)}) = \prod_{n=1}^{N_s} p(y_{n,a,t_k} \,|\, \boldsymbol{\chi}_{t_k}^{(m)}) p(y_{n,s,t_k} \,|\, \boldsymbol{\chi}_{t_k}^{(m)}) \times p(y_{n,v,t_k} \,|\, \boldsymbol{\chi}_{t_k}^{(m)}).$$

The weights in (4) need to be normalized, $w_k^{(m)} = \tilde{w}_k^{(m)} / \sum_{i=1}^{M} \tilde{w}_k^{(i)}$.

(iv) *Update of the posterior pdf's of the sensor offsets*: Using the set of weighted sensor offset samples, $\left\{w_k^{(m)}, \tau_n^{(m)}\right\}_{m=1}^{M}$, the sample mean and variance of τ_n are obtained,

$$\begin{aligned} \mu_{n,k} &= \sum_m w_k^{(m)} \tau_n^{(m)} \\ \sigma_{n,k}^2 &= \sum_m w_k^{(m)} (\tau_n^{(m)} - \mu_{n,k})^2. \end{aligned}$$

Then, the parameters of the beta-approximation to the posterior pdf of τ_n, $\pi_{n,k}, \phi_{n,k}$, can be calculated as

$$\begin{aligned} \pi_{n,k} &= \mu_{n,k} \left\{ \frac{\mu_{n,k}(1-\mu_{n,k})}{\sigma_{n,k}^2} - 1 \right\} \\ \phi_{n,k} &= (1-\mu_{n,k}) \frac{\pi_{n,k}}{\mu_{n,k}}. \end{aligned} \quad (5)$$

(v) *Estimation of parameter and states* : It is straightforward to approximate any moment of the posterior distribution of $\boldsymbol{x}_{t_k+\tau_{1:N_s}}$ and $\tau_{1:N_s}$ using the discrete probability measure given by the weighted set of particles $\{\boldsymbol{\chi}_{t_k}^{(m)}, w_k^{(m)}\}$. In particular, the target position and velocity at time $(k-1)T + \tau_n$ can be estimated as

$$\hat{\boldsymbol{x}}_{t_k+\tau_n} = \sum_{m=1}^{M} w_k^{(m)} \boldsymbol{x}_{t_k+\tau_n}^{(m)},$$

which is an approximation of the minimum mean square error (MMSE) estimate of $\boldsymbol{x}_{t_k+\tau_n}$ given $\mathbf{y}_{t_1:t_k}$. Similarly, $\mu_{n,k}$ is the (approximate) MMSE estimate of the sensor offset τ_n.

[1] $\mathcal{B}(\tau; \pi, \phi) = \frac{(\tau-a)^{\pi-1}(b-\tau)^{\phi-1}}{\beta(\pi,\phi)(b-a)^{\pi+\phi-1}}$ where $\beta(\pi, \phi) = \int_0^1 s^{\pi-1}(1-s)^{\phi-1} ds$ is the beta function, and a and b are the lower and upper bounds of τ.

Fig. 2. Series of operations during each time interval

(vi) *Resample and Move*: Occasional resampling steps are needed in order to avoid the well-known phenomenon of weight degeneracy [6]. In our simulations, we perform a multinomial resampling step for every k. Following resampling of the sample $\{\boldsymbol{\chi}_{t_k}^{(m)}\}_{m=1}^{M}$, we obtain a new stream of particles $\{\tilde{\boldsymbol{\chi}}_{t_k}^{(m)}\}_{m=1}^{M}$. For each of this stream of particles, we first find the state particle corresponding to the maximum timing offset and then propagate the particle to time kT, i.e.,

$$\mathbf{x}_k^{(m)} = \mathbf{A}_{kT-\tau_{i_{N_s}}^{(m)}} \mathbf{x}_{t_k+\tau_{i_{N_s}}^{(m)}} + \mathbf{u}_k^{(m)},$$

where $\mathbf{u}_k^{(m)}$ is drawn from $\mathcal{N}(\mathbf{0}, \mathbf{C}_{kT-\tau_{i_{N_s}}^{(m)}})$.

Figure 2 is a graphical representation of the recursive steps of the proposed DAPF algorithm, where each cloud represents the state and sensor offset samples.

B. Liu and West algorithm (LW)

In the joint static-parameter and dynamic-state algorithm proposed in [8], a Gaussian mixture density is used for an artificial evolution of the fixed parameters. Clearly we cannot use a Gaussian mixture density for an artificial evolution of the sensor offset parameter τ_n because it is bounded, i.e., $0 \leq \tau_n \leq T$. Therefore, we propose the following truncated Gaussian mixture density for the evolution of τ_n,

$$p(\tau_{n,k+1} \mid \mathbf{y}_{t_1:t_k})) = \sum_k w_k^{(m)} T\mathcal{N}_{(0,T)}(\tau_{n,k+1}; \hat{\tau}_{n,k}^{(m)}, h^2\sigma_{n,k}^2) \tag{6}$$

where $T\mathcal{N}_{(0,T)}(\cdot)$ is the normal distribution truncated outside of the interval $(0, T)$, with mean $\hat{\tau}_{n,k}^{(m)}$ and variance $h^2\sigma_{n,k}^2$, the k subscript in $\tau_{n,k}^{(m)}$ is due to the artificial time evolution (not to the model dynamics) and $\mu_{n,k}$ and $\sigma_{n,k}^2$ are the sample mean and variance of τ_n at time k. The choices $\hat{\tau}_{n,k}^{(m)} = a\tau_{n,k}^{(m)} + (1-a)\mu_{n,k}$ and $h^2 = 1 - a^2$ ensure that the mixture density preserves the original sample mean and variance (this technique is termed "shrinkage" in [8]).

We now summarize the main steps of the algorithm. At time k, we have available $\{\boldsymbol{x}_{0:k-1}^{(m)}, \tau_{1:N_s}^{(m)}, w_{k-1}^{(m)}\}_{m=1}^{M}$ and receive the new observation $\mathbf{y}_{t_k}$.

(i) *Estimation of prior estimates*: For each $n = 1 \cdots N_s$ compute $\hat{\boldsymbol{x}}_{t_k+\tau_n^{(m)}}$ as

$$\hat{\boldsymbol{x}}_{t_k+\tau_n^{(m)}}^{(m)} = \mathrm{A}_{\tau_n^{(m)}} \boldsymbol{x}_{k-1}^{(m)}$$

(ii) *Sampling of sensor offset parameters*: Consider a discrete random variable V which takes values on the set $\{1, \ldots, M\}$ with probabilities proportional to $p(\mathbf{y}_{t_k} | \hat{\boldsymbol{x}}_{t_k+\tau_{1:N_s}^{(m)}}, \tau_{1:N_s}^{(m)})$. Draw M times from V and denote this set of samples as $\{V^{(1)}, \ldots, V^{(M)}\}$. Corresponding to each of these elements, draw $\tau_n^{(j)}$ from the $V^{(j)}$-th kernel of the truncated gaussian mixture in (6).

(iii) *Sampling target state parameters*: For each $m = 1, \ldots, M$ and each $n = 1, \ldots, N_s$, compute a state particle using the Markov prior (1), as described in step (ii) of the DAPF algorithm, i.e., compute the aggregated particles $\boldsymbol{\chi}_{t_k}^{(m)}$, $m = 1, \ldots, M$.

(iv) *Evaluation of the weights*:

$$w_k^{(m)} \propto \frac{p(\mathbf{y}_{t_k} | \boldsymbol{x}_{t_k+\tau_{1:N_s}^{(m)}}^{(m)})}{p(\mathbf{y}_{t_k} | \hat{\boldsymbol{x}}_{t_k+\tau_{1:N_s}^{(V^{(m)})}}^{(V^{(m)})})} \tag{7}$$

The likelihood factors are calculated as described in step (iii) of the DAPF algorithm. The estimation, resampling and move steps are also performed in the same way as for the latter algorithm.

IV. SIMULATIONS

Consider a network with $N_s = 3$ sensors and observation period $T = 1$ s. The initial pdf of the target state is $p(\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{\mu}_0, \mathbf{K}_0)$, with $\boldsymbol{\mu}_0 = [0; 0.5; 0.0; 0.05]^\top$ and $\mathbf{K}_0 = diag\{1, \ 1, \ 0.01, \ 0.01\}$. The measurement noise processes are Gaussian, with zero-mean and standard deviations $\sigma_{v_a} = 0.01$, $\sigma_{v_d} = 0.5$ and $\sigma_{v_v} = 0.1$ for angle, distance and velocity, respectively. We have set the number of particles to $M = 3000$ for all SMC algorithms.

We study the proposed methods for the scenario when the three sensors have different offsets, $\tau_1 = 0.2T$, $\tau_2 = 0.5T$, and $\tau_3 = 0.8T$. The nth sensor timing offset particles are all initially drawn from $\beta(\tau_n, 1, 1)$.

As an illustrative example, Fig 3 shows the estimation of a complete target trajectory for a single simulation run, using DAPF and LW algorithms. For comparison, we also depict the trajectory estimate obtained with a standard SMC algorithm with perfect knowledge of the timing offsets (labeled "SPF-Knw"). It can be seen that all three algorithms can track the target along a highly nonlinear trajectory. From this single trial, the LW algorithm seems to be the weakest technique.

In order to statistically asses the performance of the algorithms, we have considered the root mean square error (RMSE) as a figure of merit. In particular, for $L = 100$ independent

Fig. 3. Trajectory and its estimates using the DAPF algorithm

simulation runs we have evaluated

$$
\begin{aligned}
RMSE_{\tau_{n,k}} &= \frac{1}{L}\sum_{l=1}^{L}(\hat{\tau}_{n,k,l} - \tau_{n,l})^2 \\
RMSE_{x_{i,k}} &= \frac{1}{L}\sum_{l=1}^{L}(\hat{x}_{i,k,l} - x_{i,k,l})^2 \\
RMSE_{\dot{x}_{i,k}} &= \frac{1}{L}\sum_{l=1}^{L}(\hat{\dot{x}}_{i,k,l} - \dot{x}_{k,l})^2,
\end{aligned}
$$

where $\tau_{n,l}$, $x_{i,k,l}$ and $\dot{x}_{i,k,l}$ are the nth offset, the ith state dynamic variable ($i \in \{1,2\}$) at time kT and its derivative, respectively, all of them for the lth simulation run. Their corresponding estimates are denoted as $\hat{\tau}_{n,k,l}$ (at time kT), $\hat{x}_{i,k,l}$ and $\hat{\dot{x}}_{i,k,l}$, respectively.

Figure 4(a) shows the RMSEs obtained for the estimation of the three offsets, τ_1, τ_2, and τ_3, using the DAPF algorithm. The corresponding RMSEs resulting from the application of the LW technique are shown in Figure 4(b). The RMSEs obtained with the DAPF algorithm are similar for the three sensors, and clearly smaller than the RMSEs obtained with the LW method. This is a consequence of the misconvergence of the LW algorithm for some scenarios.

Figure 5 plots the RMSEs obtained for the estimation of the targets dynamics for the DAPF and SMC-Knw algorithms. We have found that the DAPF technique is consistently better than the conventional SMC-Knw algorithm, with known offsets, *for this particular scenario*. We conjecture that this is due to the smoothing effect of using a set of different offsets for each particle, which means that we assess the quality of the particle (through the computation of the corresponding likelihoods and weights) using a longer segment of the target state realization.

Finally, we show an example of how the estimation of the offsets with the DAPF algorithm evolves with time in a single simulation trial. In particular, Figure 6 shows the marginal posteriors of the sensor offsets, approximated by beta distributions, at time instants $t = 25, 50, 75$ and 100

(a) DAPF

(b) LW

Fig. 4. The RMSE of $\tau_{1:3}$ with the DAPF and LW algorithm

Fig. 5. The RMSE of $\boldsymbol{x}_t$ with the DAPF and SMC-Knw algorithm. Upper left: RMSE of $x_{1,k}$. Upper right: RMSE of $x_{2,k}$. Lower left: RMSE of $\dot{x}_{1,k}$. Lower right: RMSE of $\dot{x}_{2,k}$.

Fig. 6. The evolution of the beta distribution for each of the sensor offsets at time instants t=25,50,75,100s

Fig. 7. An estimate of the sensor offsets. The dashed horizontal lines are the true sensor offsets

s. It can be seen how the modes of the beta density get closer to the true offset values and then become narrower with time. Alternatively, Figure 7 plots the time evolution of the offset estimators ($\hat{\tau}_n$, $n = 1, 2, 3$). We can see how the three estimators get close to the true values.

V. Conclusion

In this paper we have addressed the problem of target tracking in an asynchronous sensor network. We have proposed sequential Monte Carlo algorithms for the joint estimation of the sensor offsets and target state dynamics. In the DAPF algorithm, the marginal distributions of the sensors offsets are approximated by a generalized beta distribution. Using these distributions, offset samples are generated at each time interval, which are, in turn, used to propagate the target state samples. In the LW algorithm , the marginal distributions of the sensors offsets are approximated by a mixture of truncated Gaussian kernels. Through computer simulations we have compared these two methods and observed that the DAPF algorithm has a clearly better performance than the LW algorithm for the joint tracking of sensor offsets and target dynamics.

Some potential limitations of these techniques should also be remarked, though. It has been observed that, in a few cases, due to the apparent non-linearity of the offset parameters through the covariance matrix of the process noise in the DSS model, the marginal posterior of the sensor offsets is multimodal and the DAPF method can get stuck at local maxima. Good initialization or informative prior knowledge of the sensor offsets may be helpful. Also when very few offset samples have non-negligible weights and are concentrated in a particular region, approximation of the posterior using parametric densities does not seem to contribute much to sample diversity. This may be particularly harmful in scenarios where the filter gets stuck at a local maxima. A potential problem with the LW algorithm is that when the sensor offset estimates or target dynamic estimates are poor, the denominator in (7) may get close to zero and cause the particle filter to diverge.

References

[1] F. Sivrikaya and B. Yener, "Time synchronization in sensor networks," *IEEE Network*, vol. 18, no. 4, pp. 45– 50, 2004.

[2] L. Doherty, B. A. Warneke, B. Boser, and K. S. J. Pister, "Energy and performance considerations for smart dust," in *International Journal of Parallel and Distributed Sensor Networks*, Dec 2001.

[3] X.Lin, Y. Bar-Shalom, and T. Kirubarajan, "Multisensor-multitarget bias estimation for general asynchronous sensors," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 3, pp. 899– 921, 2005.

[4] Y. Zhou, "A kalman filter based registration approach for asynchronous sensors in multiple sensor fusion applications," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04)*, vol. 2, may 2004, pp. 293–6.

[5] T. Li, A. Ekpenyong, and Y. Huang, "A location system using asynchronous distributed sensors," in *INFOCOM*, 2004.

[6] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 2001.

[7] G. Kitagawa, "A self-organizing state space model," *Journal of the American Statistical Association*, vol. 93, no. 443, pp. 1203–1215, Sept. 1998.

[8] J. Liu and M. West, "Combined parameter and state estimation in simulation-based filtering," in *Sequential Monte Carlo Methods in Practice. New York*, J. F. G. D. F. A. Doucet and N. J. Gordon, Eds. Springer-Verlag, New York, 2000.

[9] G. Storvik, "Particle filters for state-space models with the presence of unknown static parameters," *IEEE Transactions Signal Processing*, vol. 50, no. 2, pp. 281–289, February 2002.

[10] A. Doucet and V. B. Tadic, "Parameter estimation in general state-space models using particle methods," *Annals of the Institute of Statistical Mathematics*, vol. 55, no. 2, pp. 409–422, June 2003.

[11] P.M.Djurić, M.F.Bugallo, and J. Míguez, "Density assisted particle filters for state and parameter estimation," *Proceedings of the IEEE 29th International Conference on Acoustics, Speech and Signal Processing*, May 2004.

[12] D. Crisan and A. Doucet, "A survey of convergence results on particle filtering," *IEEE Transactions Signal Processing*, vol. 50, no. 3, pp. 736–746, March 2002.