Online Multiple Target Tracking and Sensor Registration Using Sequential Monte Carlo Methods

Junfeng Li, William Ng and Simon Godsill
Department of Engineering
University of Cambridge
(jfl28, kfn20, sjg)@eng.cam.ac.uk

Abstract—In tracking applications, the target state (e.g, position, velocity) can be estimated by processing the measurements collected from all deployed sensors at a central node. The estimation performance significantly relies on the accuracy of the sensor positions/rotations when data fusion is conducted. Since in practice precise knowledge of this sensor information may not be available, in this paper two Sequential Monte Carlo (SMC) approaches are proposed to jointly estimate the target state and resolve the sensor position uncertainty. The first one uses the Particle filter combined with the Gibbs sampling method to deal with the general sensor registration problem. The second one uses the Rao-Blackwellised Particle filter for a special case where the uncertainty of the sensor is a nearly constant measurement bias.

TABLE OF CONTENTS

1. INTRODUCTION

In tracking applications, the target state (e.g, position, velocity) can be estimated by processing the measurements collected from all deployed sensors at a central node. Usually the characteristics (e.g. position, rotation) of sensors are assumed precisely known, but in real-world systems this assumption may not hold, thereby degrading the performance of target tracking.

In this paper we consider a tracking system with distributed sensors whose characteristics are unknown. These sensors operate independently, and collect measurements and send them to a central reference node whose position is assumed known. The aims of our algorithm are to jointly track targets states and resolve the uncertainty of the deployed sensors with respect to the reference sensor. Also known as sensor registration problem, this sensor uncertainty estimation problem has been studied in the existing literature, but the contribution of our approach is clear. The off-line batch approaches, e.g. [4], [10], [15] usually require paired sets of target measurements and the clutter free surveillance region. It is difficult to achieve such a condition in the practical tracking applications. Some approaches [11], [12], [14] deal with the problem in the on-line fashion. However in [11], [12], the registration errors are solved based on the Extended Kalman filter (EKF) which may not be able to solve non-linear/non-Gaussian problems perfectly. The Sequential Monte Carlo (SMC) framework developed in [14] assumes that the posterior distributions for the target and sensor states are independent, but this assumption may not hold since the measurement is dependent on both target and sensor states. In this paper we propose SMC approaches to jointly estimate the target state and the sensor uncertainty in a surveillance environment exhibiting low detection probability and high clutter spatial density.

This paper is organised as follows. Section 2 presents the state-space model and the derivation of the distributions required. In section 3 we divide the problem into two. Firstly, we propose a general algorithm which uses particle filters combined with Gibbs sampling [13] to jointly track the posterior distribution for both target and sensor states. Secondly, we propose to use the Rao-Blackwellised Particle filter (RBPF) [5] to deal with a special case in which the uncertain characteristic of sensors is a nearly constant measurement bias due to the sensor hardware flaw. The simulation results are given in Section 4 followed by the conclusion in Section 5.

2. DATA MODEL

Let $\boldsymbol{x}_t = [\boldsymbol{x}_{1,t}^T, \boldsymbol{x}_{2,t}^T, ..., \boldsymbol{x}_{K,t}^T]^T$ denote a combined target state vector for K number of targets. The state evolution equation for modelling the manoeuvring of the kth target is given as:

$$\boldsymbol{x}_{k,t} = \boldsymbol{f}_k(\boldsymbol{x}_{k,t-1}, \boldsymbol{\nu}_{k,t-1}) \tag{1}$$

where $\boldsymbol{\nu}_{k,t}$ denotes mutually uncorrelated zero-mean white Gaussian driving noise of the kth target with a fixed and known covariance matrix $\boldsymbol{\Sigma}_\nu^k$. By modelling all targets as moving independently according to Markovian dynamics [1], [2], and sharing the same dynamical model, the joint transition density for the target state can be factorised as:

$$p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \prod_{k=1}^{K} p(\boldsymbol{x}_{k,t}|\boldsymbol{x}_{k,t-1}) \tag{2}$$

1-4244-0525-4/07/$20.00/©2007 IEEE
IEEEAC paper # 1406

Denote by $\boldsymbol{\varphi}_t = [\boldsymbol{\varphi}_t^0, \boldsymbol{\varphi}_t^1, ..., \boldsymbol{\varphi}_t^{N_o-1}]$ a combined state vector for N_o sensors at time t, with $\boldsymbol{\varphi}_t^0$ being a reference sensor with known characteristics (e.g., position and orientation). For other sensors, their states are unknown and modelled independently as:

$$\boldsymbol{\varphi}_t^p = \boldsymbol{g}^p(\boldsymbol{\varphi}_{t-1}^p, \boldsymbol{u}_{t-1}^p),\ p \neq 0, \tag{3}$$

where $\boldsymbol{u}_t^p$ denotes zero-mean white Gaussian noise of the pth sensor with a fixed and known covariance matrix $\boldsymbol{\Sigma}_u^p$. The transition density for the sensor states can be written as

$$p(\boldsymbol{\varphi}_t|\boldsymbol{\varphi}_{t-1}) = \prod_{p=1}^{N_o-1} p(\boldsymbol{\varphi}_t^p|\boldsymbol{\varphi}_{t-1}^p). \tag{4}$$

Suppose that a set of total M_t measurements collected by N_o sensors is denoted as $\boldsymbol{y}_t = \{\boldsymbol{y}_t^p\}_{p=0}^{N_o-1}$, where $\boldsymbol{y}_t^p = \{\boldsymbol{y}_{m,t}^p\}_{m=1}^{M_t^p}$ denotes M_t^p measurements collected by the pth sensor at time t with $M_t = \sum_{p=0}^{N_o-1} M_t^p$. Since we assume that the sensors operate independently, the likelihood function can be factorised as

$$p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{\varphi}_t) = \prod_{p=0}^{N_o-1} p(\boldsymbol{y}_t^p|\boldsymbol{x}_t, \boldsymbol{\varphi}_t^p). \tag{5}$$

Furthermore we assume that measurements originate from either true targets if detected or clutter. Therefore if the mth measurement collected by the pth sensor originates from the kth target, it is modelled as:

$$\boldsymbol{y}_{m,t}^p = \boldsymbol{h}(\boldsymbol{x}_{k,t}, \boldsymbol{\varphi}_t^p) + \boldsymbol{n}_{m,t}^p, \tag{6}$$

where $\boldsymbol{n}_{m,t}^p$ is a zero-mean white Gaussian observation noise with a known covariance $\boldsymbol{\Sigma}_n^p$. On the contrary, if the measurement originates from clutter, it is modelled to be uniformly distributed within the entire surveillance region.

Generally there are two possible methods to evaluate the likelihood function (5). One is known as the data association free approach which uses Poisson likelihood model [6], [7], [8], and the other is the well-known Joint Probabilistic Data Association (JPDA) filter [2], [3], where the measurement-to-track association probabilities are computed jointly across all targets and measurements. Then targets are tracked independently using *combined* measurements weighted by those probabilities.

With respect to the Poisson likelihood model, when sensor p is considered, the number of measurements originating from target k is modelled as a Poisson distribution whose mean is $\mu_{k,T}^p$, whereas the number of clutter points has a mean of μ_C^p. Accordingly the likelihood function of all observations $\boldsymbol{y}_t$ can be written as [6], [7], [8]:

$$p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{\varphi}_t) \propto \prod_{p=0}^{N_o-1} \prod_{m=1}^{M_t^p} \Lambda(\boldsymbol{y}_{m,t}^p|\boldsymbol{x}_t, \boldsymbol{\varphi}_t) \tag{7}$$

where

$$\Lambda(\boldsymbol{y}_{m,t}^p|\boldsymbol{x}_t, \boldsymbol{\varphi}_t) = \mu_C^p p_c(\boldsymbol{y}_{m,t}^p) + \sum_{k=1}^{K} \mu_{k,T}^p p_T(\boldsymbol{y}_{m,t}^p|\boldsymbol{x}_{k,t}, \boldsymbol{\varphi}_t^p). \tag{8}$$

The $p_c(\cdot)$ is the likelihood function of clutter points and $p_T(\cdot)$ is the likelihood function of target originated measurements which can be evaluated from (6). Therefore we can write

$$p_c(\boldsymbol{y}_{m,t}^p) = \mathcal{U}_{\Re_p}, \tag{9}$$

$$p_T(\boldsymbol{y}_{m,t}^p|\boldsymbol{x}_{k,t}, \boldsymbol{\varphi}_t^p) = \mathcal{N}(\boldsymbol{y}_{m,t}^p|\boldsymbol{h}(\boldsymbol{x}_{k,t}, \boldsymbol{\varphi}_t^p), \boldsymbol{\Sigma}_n^p) \tag{10}$$

where $\mathcal{U}_{\Re_p}$ denotes the uniform distribution over the observation region $\Re_p$ for the pth sensor and $\mathcal{N}(.|.)$ is the Gaussian distribution. Note that $p_c(\boldsymbol{y}_{m,t}^p)$ and $p_T(\boldsymbol{y}_{m,t}^p|\boldsymbol{x}_{k,t}, \boldsymbol{\varphi}_t^p)$ can be more complex likelihood functions depending on applications.

By using the JPDA, it assumed that a target can generate at most one measurement and a measurement could have originated from at most one target with respect to each sensor. Therefore the function (5) is rewritten as

$$\begin{aligned} p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{\varphi}_t) &= \prod_{k=1}^{K_t} \prod_{p=0}^{N_o-1} p(\boldsymbol{y}_t^p|\boldsymbol{x}_{k,t}, \boldsymbol{\varphi}_t^p) \\ &= \prod_{k=1}^{K_t} \prod_{p=0}^{N_o-1} \left[\beta_{0,k}^p + \sum_{m=1}^{M_t^p} \beta_{m,k}^p p(\boldsymbol{y}_{m,t}^p|\boldsymbol{x}_{k,t}, \boldsymbol{\varphi}_t^p) \right] \end{aligned} \tag{11}$$

where $\beta_{0,k}^p$ is the posterior probability that the kth target is undetected by the pth sensor, and $\beta_{m,k}^p$ is the probability that the mth measurement at the pth sensor originates from the kth target. These joint association probabilities are evaluated from the joint association hypothesis probability, defined as

$$p(\gamma^j|\boldsymbol{y}_{1:t}) \propto (V^{-1})^{N_c^j} \left(\frac{P_d}{1-P_d}\right)^{N_d^j} \prod_{m,k\in\mathcal{I}^j} p_k(\boldsymbol{y}_{m,t}|\boldsymbol{y}_{1:t-1}) \tag{12}$$

with

$$\sum_{\forall j} p(\gamma^j|\boldsymbol{y}_{1:t}) = 1 \tag{13}$$

where γ^j refers to the jth association hypothesis, N_c^j and N_d^j are the number of clutter points and detected targets respectively, V is the volume of the observation space, P_d is the target detection probability, and $\mathcal{I}^j$ is a sub-space of γ^j in which any mth measurement can be associated with any kth target. For simplification, we drop the sensor index in (12) and the following equations unless specified. The predictive likelihood function $p_k(\boldsymbol{y}_{m,t}|\boldsymbol{y}_{1:t-1})$ is defined as:

$$\begin{aligned} &p_k(\boldsymbol{y}_{m,t}|\boldsymbol{y}_{1:t-1}) \\ &= \int p(\boldsymbol{y}_{m,t}|\boldsymbol{x}_{k,t}, \boldsymbol{\varphi}_t) p(\boldsymbol{x}_{k,t}, \boldsymbol{\varphi}_t|\boldsymbol{y}_{1:t-1}) d\boldsymbol{x}_{k,t}\, d\boldsymbol{\varphi}_t \end{aligned} \tag{14}$$

Since the number of hypotheses exponentially grows with respect to the number of targets and measurements, a gating method is normally adopted for pruning. In general, the gating method considers those measurements whose have higher predictive likelihood value as feasible measurements, and omit others. The threshold is normally defined from the χ^2 distribution if the function (14) is Gaussian. Accordingly the association probability can be evaluated as:

$$\beta_{m,k} = \sum_{j \in \psi(m,k)} p(\gamma^j | \boldsymbol{y}_{1:t}), \tag{15}$$

where $\psi(m,k)$ is the subset of all valid association hypothesis where the kth target is associated with the mth measurement. For more details of the JPDAF refer to [2], [3].

3. Algorithms

Here we present a SMC-based method to jointly track targets and sensors states. Two scenarios will be considered. The first one introduces a method that combines SMC with Gibbs sampling method to deal with the general sensor registration problem. The second proposes the RBPF to deal with a special case where all measurements are corrupted with bias.

General case

Let N uniformly weighted particles $\{\boldsymbol{x}_{t-1}^{(i)}, \boldsymbol{\varphi}_{t-1}^{(i)}\}_{i=1}^{N}$ approximate the posterior distribution $p(\boldsymbol{x}_{t-1}, \boldsymbol{\varphi}_{t-1} | \boldsymbol{y}_{1:t-1})$ at time $t-1$. Given the measurement $\boldsymbol{y}_t$, we use particle filtering combined with Gibbs sampling to draw particles from the conditional posterior distribution $p(\boldsymbol{\varphi}_t | \boldsymbol{x}_t, \boldsymbol{y}_{1:t})$ and $p(\boldsymbol{x}_t | \boldsymbol{\varphi}_t, \boldsymbol{y}_{1:t})$ iteratively, as follows

$$\boldsymbol{x}_t^{(j)} \sim p(\boldsymbol{x}_t | \boldsymbol{\varphi}_t^{(j-1)}, \boldsymbol{y}_{1:t}), \tag{16}$$

$$\boldsymbol{\varphi}_t^{(j)} \sim p(\boldsymbol{\varphi}_t | \boldsymbol{x}_t^{(j)}, \boldsymbol{y}_{1:t}), \tag{17}$$

for $j = 1, ..., L$ and L is the total number of iterations with $L \geq N$. With the consideration of an appropriate burn-in period, only the last N particles will be retained, i.e., $\{\boldsymbol{x}_t^{(j)}, \boldsymbol{\varphi}_t^{(j)}\}_{j=L-N+1}^{L}$, to approximate the posterior distribution $p(\boldsymbol{x}_t, \boldsymbol{\varphi}_t | \boldsymbol{y}_{1:t})$.

In order to sample from (16), the conditional posterior distribution is rewritten as:

$$\begin{aligned} & p(\boldsymbol{x}_t | \boldsymbol{\varphi}_t^{(j-1)}, \boldsymbol{y}_{1:t}) \\ & \propto p(\boldsymbol{y}_t | \boldsymbol{x}_t, \boldsymbol{\varphi}_t^{(j-1)}) p(\boldsymbol{x}_t, \boldsymbol{\varphi}_t^{(j-1)} | \boldsymbol{y}_{1:t-1}) \\ & \approx p(\boldsymbol{y}_t | \boldsymbol{x}_t, \boldsymbol{\varphi}_t^{(j-1)}) \times \frac{1}{N} \sum_{i=1}^{N} p(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}^{(i)}) p(\boldsymbol{\varphi}_t^{(j-1)} | \boldsymbol{\varphi}_{t-1}^{(i)}), \end{aligned} \tag{18}$$

given the assumption that the dynamic model for $\boldsymbol{x}_t$ is independent of $\boldsymbol{\varphi}_t$. The likelihood and dynamic prior functions in (18) can be evaluated from (5) and (2). The Metropolis-Hastings (MH) algorithm [9] and Importance sampling (IS) [5] are proposed as two possible approaches to generate samples $\{\boldsymbol{x}_t^{(j)}\}$ approximately distributed according to $p(\boldsymbol{x}_t | \boldsymbol{\varphi}_t^{(j-1)}, \boldsymbol{y}_{1:t})$ in (18).

With respect to the MH algorithm, the probability of accepting a target state candidate $\boldsymbol{x}_t^c$ over the current one $\boldsymbol{x}_t^{(j-1)}$ is defined as:

$$\alpha = \min \left(1, \frac{p(\boldsymbol{x}_t^c | \boldsymbol{\varphi}_t^{(j-1)}, \boldsymbol{y}_{1:t}) \, \pi(\boldsymbol{x}_t^{(j-1)} | \boldsymbol{x}_t^c)}{p(\boldsymbol{x}_t^{(j-1)} | \boldsymbol{\varphi}_t^{(j-1)}, \boldsymbol{y}_{1:t}) \, \pi(\boldsymbol{x}_t^c | \boldsymbol{x}_t^{(j-1)})} \right) \tag{19}$$

where $\pi(\cdot|\cdot)$ is a proposal transition density function. Therefore $\boldsymbol{x}_t^{(j)}$ is assigned as $\boldsymbol{x}_t^c$ if $\alpha > u$ where $u \sim \mathcal{U}_{[0,1]}$. Otherwise the state remains the same, i.e. $\boldsymbol{x}_t^{(j)} = \boldsymbol{x}_t^{(j-1)}$.

As an alternative, the Importance sampling method can be used. Let's introduce the variable I_{t-1} which is the index of particles at time $t-1$, then the posterior of interest becomes

$$\begin{aligned} & p(\boldsymbol{x}_t, I_{t-1} | \boldsymbol{\varphi}_t^{(j-1)}, \boldsymbol{y}_{1:t}) \\ & \propto p(\boldsymbol{y}_t | \boldsymbol{x}_t, \boldsymbol{\varphi}_t^{(j-1)}) p(\boldsymbol{x}_t, \boldsymbol{\varphi}_t^{(j-1)} | I_{t-1}, \boldsymbol{y}_{1:t-1}) p(I_{t-1} | \boldsymbol{y}_{1:t-1}) \end{aligned} \tag{20}$$

where

$$\begin{aligned} & p(\boldsymbol{x}_t, \boldsymbol{\varphi}_t^{(j-1)} | I_{t-1}, \boldsymbol{y}_{1:t-1}) \\ & = \int p(\boldsymbol{x}_t, \boldsymbol{\varphi}_t^{(j-1)} | \boldsymbol{x}_{t-1}, \boldsymbol{\varphi}_{t-1}) \\ & \quad \times p(\boldsymbol{x}_{t-1}, \boldsymbol{\varphi}_{t-1} | I_{t-1}, \boldsymbol{y}_{1:t-1}) d\boldsymbol{x}_{t-1}, \boldsymbol{\varphi}_{t-1} \\ & = p(\boldsymbol{x}_t, \boldsymbol{\varphi}_t^{(j-1)} | \boldsymbol{x}_{t-1}^{(I_{t-1})}, \boldsymbol{\varphi}_{t-1}^{(I_{t-1})}). \end{aligned} \tag{21}$$

Furthermore let the proposal density be

$$q(\boldsymbol{x}_t, I_{t-1} | \boldsymbol{y}_{1:t}) = q(\boldsymbol{x}_t | \boldsymbol{x}_t^{(I_{t-1})}, \boldsymbol{y}_t) p(I_{t-1} | \boldsymbol{y}_{1:t-1}), \tag{22}$$

and the importance weight is given as

$$w(\boldsymbol{x}_t, I_{t-1}) \propto \frac{p(\boldsymbol{x}_t, I_{t-1} | \boldsymbol{\varphi}_t^{(j-1)}, \boldsymbol{y}_{1:t})}{q(\boldsymbol{x}_t, I_{t-1} | \boldsymbol{y}_{1:t})}. \tag{23}$$

Since the posterior $p(I_{t-1} | \boldsymbol{y}_{1:t-1})$ is the uniform distribution over 1 to N, the particles form $q(\boldsymbol{x}_t, I_{t-1} | \boldsymbol{y}_{1:t})$ can be obtained by sampling $\boldsymbol{x}_t^{(i)} \sim q(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}^{(i)}, \boldsymbol{y}_t)$ for $i = 1...N$. By substituting (20), (21) and (22) into (23), we can evaluate the importance weight as

$$w_t^{(i)} \propto \frac{p(\boldsymbol{y}_t | \boldsymbol{x}_t^{(i)}, \boldsymbol{\varphi}_t^{(j-1)}) p(\boldsymbol{x}_t^{(i)} | \boldsymbol{x}_{t-1}^{(i)}) p(\boldsymbol{\varphi}_t^{(j-1)} | \boldsymbol{\varphi}_t^{(i)})}{q(\boldsymbol{x}_t^{(i)} | \boldsymbol{x}_{t-1}^{(i)}, \boldsymbol{y}_t)}, \tag{24}$$

with

$$\sum_{i=1}^{N} w_t^{(i)} = 1.$$

Therefore the particle $\boldsymbol{x}_t^{(j)}$ can be obtained by sampling from $\{\boldsymbol{x}_t^{(i)}\}_{i=1}^{N}$ according to the weights $\{w_t^{(i)}\}_{i=1}^{N}$. Accordingly the distribution function (17) can be rewritten as:

$$\begin{aligned} & p(\boldsymbol{\varphi}_t | \boldsymbol{x}_t^{(j)}, \boldsymbol{y}_{1:t}) \\ & \propto p(\boldsymbol{y}_t | \boldsymbol{\varphi}_t, \boldsymbol{x}_t^{(j)}) p(\boldsymbol{\varphi}_t, \boldsymbol{x}_t^{(j)} | \boldsymbol{y}_{1:t-1}) \\ & \approx p(\boldsymbol{y}_t | \boldsymbol{\varphi}_t, \boldsymbol{x}_t^{(j)}) \times \frac{1}{N} \sum_{i=1}^{N} p(\boldsymbol{\varphi}_t | \boldsymbol{\varphi}_{t-1}^{(i)}) p(\boldsymbol{x}_t^{(j)} | \boldsymbol{x}_{t-1}^{(i)}). \end{aligned} \tag{25}$$

As similar procedure that utilises the MH and Importance sampling methods can be used to sample the sensors state. Our proposed algorithm is summarised as below:

---

1. At time 0, initialise N particles $\{\boldsymbol{x}_0^{(i)}, \boldsymbol{\varphi}_0^{(i)}\}_{i=1}^N$.
2. For $t = 1, 2, ...$
   (a) Initialise $\boldsymbol{\varphi}_t^{(0)}$ and $\boldsymbol{x}_t^{(0)}$ appropriately.
   (b) For $j = 1, ..., L$, sample $\boldsymbol{\varphi}_t^{(j)}$ and $\boldsymbol{x}_t^{(j)}$ from (25) and (18) iteratively by using either the MH or Importance sampling methods.
   (c) Among the L particles, retain only the last N particles, i.e., $\{\boldsymbol{x}_t^{(i)}, \boldsymbol{\varphi}_t^{(i)}\}_{i=1}^N = \{\boldsymbol{x}_t^{(j)}, \boldsymbol{\varphi}_t^{(j)}\}_{j=L-N+1}^L$
3. Go to the next time step.

Special case

Here we consider another type of practical problems facing accurate target tracking. Assuming that we have *precise* knowledge of all sensor states at any instant, due to hardware problems a sensor may introduce bias to the measurements $\boldsymbol{y}_t^p$. Denote $\bar{\boldsymbol{y}}_t^p$ as a random, unknown measurement bias introduced by sensor p. It is assumed that this bias is shared by all measurements collected by sensor p and that it can be modelled as a random walk, given by

$$\bar{\boldsymbol{y}}_t^p = \bar{\boldsymbol{y}}_{t-1}^p + \boldsymbol{u}_{t-1}^p. \tag{26}$$

where $\boldsymbol{u}_t^p$ is the zero mean Gaussian noise with a fixed and known covariance matrix Σ_u^p. With the bias term, the received measurement $\boldsymbol{y}_{m,t}^p$ in (6) becomes

$$\boldsymbol{y}_{m,t}^p = \boldsymbol{h}(\boldsymbol{x}_{k,t}, \boldsymbol{\varphi}_t^p) + \bar{\boldsymbol{y}}_t^p + \boldsymbol{n}_{m,t}^p. \tag{27}$$

From equations (26) and (27), we can see that the bias term $\bar{\boldsymbol{y}}_t^p$ is linear and Gaussian, given the measurement $\boldsymbol{y}_t^p$ and target state $\boldsymbol{x}_t$.

Here we propose to use the RBPF to cope with this special case. The joint posterior function can be written as

$$p(\boldsymbol{x}_{1:t}, \bar{\boldsymbol{y}}_{1:t}|\boldsymbol{y}_{1:t}) = p(\bar{\boldsymbol{y}}_{1:t}|\boldsymbol{x}_{1:t}, \boldsymbol{y}_{1:t})\, p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}). \tag{28}$$

The main idea behind the RBPF is to track the non-linear part, i.e. $p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t})$, using the particle filter, and update the linear part, i.e. $p(\bar{\boldsymbol{y}}_t|\boldsymbol{x}_{1:t}, \boldsymbol{y}_{1:t})$ in an analytical way using Kalman filter, for example. Since samples are only required for a smaller space, the filtering normally becomes efficient. Suppose that we have a set of N particles $\left\{\boldsymbol{x}_{1:t-1}^{(i)}, \bar{\boldsymbol{y}}_{t-1}^{(i)}, \boldsymbol{\Sigma}_{\bar{\boldsymbol{y}}_{t-1}}^{(i)}\right\}_{i=1}^N$ with associated importance weights $\{w_{t-1}^{(i)}\}_{i=1}^N$ to approximate the posterior distribution $p(\boldsymbol{x}_{1:t-1}|\boldsymbol{y}_{1:t-1})$ as

$$p(\boldsymbol{x}_{1:t-1}|\boldsymbol{y}_{1:t-1}) \approx \sum_{i=1}^N w_{t-1}^{(i)} \delta_{\boldsymbol{x}_{1:t-1}^{(i)}} \tag{29}$$

where $\bar{\boldsymbol{y}}_{t-1}^{(i)} = \left\{\bar{\boldsymbol{y}}_{t-1}^{p^{(i)}}\right\}_{p=1}^{N_o-1}$ and $\boldsymbol{\Sigma}_{\bar{\boldsymbol{y}}_{t-1}}^{(i)} = \left\{\boldsymbol{\Sigma}_{\bar{\boldsymbol{y}}_{t-1}^p}^{(i)}\right\}_{p=1}^{N_o-1}$ are the sets of the mean and covariance matrices of the conditional posterior $p(\bar{\boldsymbol{y}}_{t-1}^p|\boldsymbol{x}_{1:t-1}^{(i)}, \boldsymbol{y}_{1:t-1})$, i.e.

$$p\left(\bar{\boldsymbol{y}}_{t-1}^p|\boldsymbol{x}_{1:t-1}^{(i)}, \boldsymbol{y}_{1:t-1}\right) = \mathcal{N}\left(\bar{\boldsymbol{y}}_{t-1}^p|\bar{\boldsymbol{y}}_{t-1}^{p^{(i)}}, \boldsymbol{\Sigma}_{\bar{\boldsymbol{y}}_{t-1}^p}^{(i)}\right). \tag{30}$$

At time t the posterior of the target state can be written as:

$$\begin{aligned} p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) &= \int p(\boldsymbol{x}_{1:t}, \bar{\boldsymbol{y}}_t|\boldsymbol{y}_{1:t})d\bar{\boldsymbol{y}}_t \\ &\propto \int p(\boldsymbol{y}_t|\boldsymbol{x}_t, \bar{\boldsymbol{y}}_t)\, p(\boldsymbol{x}_{1:t}, \bar{\boldsymbol{y}}_t|\boldsymbol{y}_{1:t-1})d\bar{\boldsymbol{y}}_t, \end{aligned} \tag{31}$$

where

$$\begin{aligned} &p(\boldsymbol{x}_{1:t}, \bar{\boldsymbol{y}}_t|\boldsymbol{y}_{1:t-1}) \\ &= p(\boldsymbol{x}_t|\boldsymbol{x}_{1:t-1}, \bar{\boldsymbol{y}}_t, \boldsymbol{y}_{1:t-1}) \\ &\quad \times p(\bar{\boldsymbol{y}}_t|\boldsymbol{x}_{1:t-1}\boldsymbol{y}_{1:t-1})p(\boldsymbol{x}_{1:t-1}|\boldsymbol{y}_{1:t-1}) \\ &= p(\boldsymbol{x}_t|\boldsymbol{x}_{1:t-1})p(\boldsymbol{x}_{1:t-1}|\boldsymbol{y}_{1:t-1}) \\ &\quad \times \int p(\bar{\boldsymbol{y}}_t|\bar{\boldsymbol{y}}_{t-1})p(\bar{\boldsymbol{y}}_{t-1}|\boldsymbol{x}_{1:t-1}, \boldsymbol{y}_{1:t-1})d\bar{\boldsymbol{y}}_{t-1}, \end{aligned} \tag{32}$$

given the assumption that the dynamic model for $\boldsymbol{x}_t$ is independent of $\bar{\boldsymbol{y}}_t$. By substituting (11), (26), (30), (29) and (32) into (31), the posterior function can be rewritten as:

$$\begin{aligned} &p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) \\ &\propto \sum_{i=1}^N \prod_{p=0}^{N_o-1} \prod_{k=1}^K J(\boldsymbol{\beta}_k^p, \boldsymbol{y}_t^p, \boldsymbol{\mu}_{k,t}^{p^{(i)}})p(\boldsymbol{x}_t|\boldsymbol{x}_{1:t-1}^{(i)})w_{t-1}^{(i)}. \end{aligned} \tag{33}$$

where

$$J(\boldsymbol{\beta}_k^p, \boldsymbol{y}_t^p, \boldsymbol{\mu}_{k,t}^{p^{(i)}}) = \beta_{0,k}^p + \sum_{m=1}^{M_t^p} \beta_{m,k}^p \mathcal{N}(\boldsymbol{y}_{m,t}^p|\boldsymbol{\mu}_{k,t}^{p^{(i)}}, \boldsymbol{\Sigma}_t^{p^{(i)}}) \tag{34}$$

Since targets are modelled as Markovian dynamics, let $\boldsymbol{x}_t^{(i)}$ be sampled from a proposal density $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}^{(i)}, \boldsymbol{y}_t)$ for $i = \{1, ..., N\}$, the importance weight can be updated from [1]:

$$w_t^{(i)} \propto w_{t-1}^{(i)} \times \frac{\prod_{p=0}^{N_o} \prod_{k=1}^K J(\boldsymbol{\beta}_k^p, \boldsymbol{y}_t^p, \boldsymbol{\mu}_{k,t}^{p^{(i)}})p(\boldsymbol{x}_t^{(i)}|\boldsymbol{x}_{t-1}^{(i)})}{q(\boldsymbol{x}_t^{(i)}|\boldsymbol{x}_{t-1}^{(i)}, \boldsymbol{y}_t)} \tag{35}$$

where

$$\boldsymbol{\mu}_{k,t}^{p^{(i)}} = \boldsymbol{h}(\boldsymbol{x}_{k,t}^{(i)}, \boldsymbol{\varphi}_t^p) + \bar{\boldsymbol{y}}_{t-1}^{p^{(i)}} \tag{36}$$

$$\boldsymbol{\Sigma}_t^{p^{(i)}} = \boldsymbol{\Sigma}_{\bar{\boldsymbol{y}}_{t-1}}^{p^{(i)}} + \boldsymbol{\Sigma}_u^p + \boldsymbol{\Sigma}_n^p, \tag{37}$$

with

$$\sum_{i=1}^N w_t^{(i)} = 1. \tag{38}$$

In this case the predictive likelihood (14) becomes intractable, so the Gaussian approximation is adopted to evaluated the association probability in (35). Therefore the function (14) with respect to the pth sensor and kth target can be approximated as:

$$p_k(\boldsymbol{y}^p_{m,t}|\boldsymbol{y}_{1:t-1}) \approx \mathcal{N}(\boldsymbol{y}^p_{m,t}|\widetilde{\boldsymbol{\mu}}^p_k, \widetilde{\boldsymbol{\Sigma}}^p_k) \tag{39}$$

where

$$\widetilde{\boldsymbol{\mu}}^p_k = \sum_{i=1}^{N} w^{(i)}_{t-1} \left[\boldsymbol{h}(\boldsymbol{x}^{(i)}_{k,t}, \boldsymbol{\varphi}^p_t) + \bar{\boldsymbol{y}}^{p^{(i)}}_{t-1}\right] \tag{40}$$

$$\widetilde{\boldsymbol{\Sigma}}^p_k = \boldsymbol{\Sigma}^p_n + \sum_{i=1}^{N} w^{(i)}_{t-1} \left\{\boldsymbol{\Sigma}^{(i)}_{\bar{\boldsymbol{y}}^p_{t-1}} + \boldsymbol{\Sigma}^p_u + [\Delta \boldsymbol{y}^p_t]\,[\Delta \boldsymbol{y}^p_t]^T\right\} \tag{41}$$

$$\Delta \boldsymbol{y}^p_t = \boldsymbol{g}(\boldsymbol{x}^{(i)}_{k,t}, \boldsymbol{\varphi}^p_t) - \sum_{i=1}^{N} w^{(i)}_{t-1}\, \boldsymbol{g}(\boldsymbol{x}^{(i)}_{k,t}, \boldsymbol{\varphi}^p_t) \tag{42}$$

$$\boldsymbol{x}^{(i)}_{k,t} \sim p(\boldsymbol{x}_{k,t}|\boldsymbol{x}^{(i)}_{k,t-1}). \tag{43}$$

Given the target state particle $\boldsymbol{x}^{(i)}_t$, measurements $\boldsymbol{y}^p_t$, prior $\bar{\boldsymbol{y}}^{p^{(i)}}_{t-1}$, and $\boldsymbol{\Sigma}^{p^{(i)}}_{\bar{\boldsymbol{y}}_{t-1}}$, we can update the sensor bias $\bar{\boldsymbol{y}}^{p^{(i)}}_t$, and $\boldsymbol{\Sigma}^{p^{(i)}}_{\bar{\boldsymbol{y}}_t}$ using the standard JPDA Kalman filter. In short, our proposed algorithm is summarised as below.

---

1. At time 0, initialise N particles $\left\{\boldsymbol{x}^{(i)}_0, \bar{\boldsymbol{y}}^{(i)}_0, \boldsymbol{\Sigma}^{p^{(i)}}_{\bar{\boldsymbol{y}}_0}\right\}_{i=1}^{N}$, with importance weights $\{w^{(i)}_0 = 1/N\}_{i=1}^N$.
2. For $t = 1, 2, ...,$
 (a) Gating and evaluating the association probabilities using Gaussian approximation according to (39), (12) and (15).
 (b) Sample $\boldsymbol{x}^{(i)}_t \sim q(\boldsymbol{x}_t|\boldsymbol{x}^{(i)}_{t-1}, \boldsymbol{y}_t)$ for $i = 1, ..., N$.
 (c) Update the importance weight according to (35).
 (d) Update $\bar{\boldsymbol{y}}^{p^{(i)}}_t$ and $\boldsymbol{\Sigma}^{p^{(i)}}_{\bar{\boldsymbol{y}}_t}$, given $\bar{\boldsymbol{y}}^{p^{(i)}}_{t-1}$, $\boldsymbol{\Sigma}^{p^{(i)}}_{\bar{\boldsymbol{y}}_{t-1}}$, $\boldsymbol{x}^{(i)}_t$ and $\boldsymbol{y}_t$ via the standard Kalman filter.
 (e) Resample when necessary to avoid the particle filter degeneracy.
3. Go to the next time step.

---

4. SIMULATIONS

Experiment I

In this subsection we evaluate the performance of the proposed particle filter combined with Gibbs sampling algorithms by jointly tracking the motions of two simulated moving targets with two sensors. The first sensor is a stationary reference one whose location is fixed and known at $(0, 0)$, and the second sensor heads from $(50, 50)$ towards southeast with a nearly constant velocity. The surveillance region within which the sensors operate has a size of $\Re_V = [0, 2\pi] \times [0, 350]$ and is characterised with low target detection probability $P_d = 0.5$ and large mean number of clutter points $\lambda_C = 10$ per sensor in every scan. The targets are synthesised using the nearly constant velocity model [2], where the evolution equation (1) is given by

$$\begin{aligned}\boldsymbol{x}_{k,t} &= [x_{k,t}, \dot{x}_{k,t}, y_{k,t}, \dot{y}_{k,t}]^T \\ &= \boldsymbol{A}\boldsymbol{x}_{k,t-1} + \boldsymbol{B}\boldsymbol{\nu}_t,\end{aligned} \tag{44}$$

where $(x_{k,t}, y_{k,t})$ and $(\dot{x}_{k,t}, \dot{y}_{k,t})$ are the position and velocity of the kth target in the Cartesian coordinates at time t. The coefficient matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are, respectively, given as follows

$$\boldsymbol{A} = \begin{bmatrix} 1 & T_s & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ and } \boldsymbol{B} = \begin{bmatrix} T_s^2/2 & 0 \\ T_s & 0 \\ 0 & T_s^2/2 \\ 0 & T_s \end{bmatrix} \tag{45}$$

where T_s is the sampling instant. The driving noise $\boldsymbol{\nu}_t$ is a zero mean Gaussian random variable whose covariance matrix is $\boldsymbol{\Sigma}_\nu = diag[\sigma^2_x, \sigma^2_y]$. Likewise, the movement of the second sensor is modelled as follows:

$$\boldsymbol{\varphi}^1_t = [x^1_o, \dot{x}^1_o, y^1_o, \dot{y}^1_o]^T_t = \boldsymbol{A}\boldsymbol{\varphi}^1_{t-1} + \boldsymbol{B}\boldsymbol{u}_{t-1} \tag{46}$$

where x^1_o and y^1_o are the (x, y) position of the sensor. $\dot{x}^1_o$ and $\dot{y}^1_o$ denote its velocity. $\boldsymbol{u}_t$ is zero mean Gaussian noise with variance $\Sigma_u = diag[\sigma^2_{x_o}, \sigma^2_{y_o}]$. Note that the state of the reference sensor $\boldsymbol{\varphi}^0_t = [0, 0, 0, 0]^T, \forall t$.

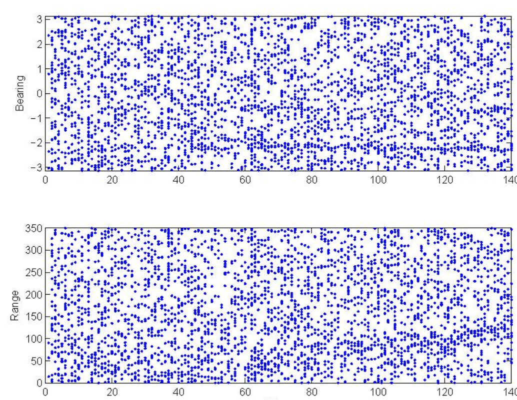


Figure 1. Synthesised observations with $P_d = 0.5$, and $\lambda_C = 10$.

Furthermore the detected target observation model in (6) contains bearings and range measurement components, defined as:

$$\boldsymbol{y}^p_{m,t} = \begin{bmatrix} \tan^{-1}\left(\frac{y_{k,t} - y^p_o}{x_{k,t} - x^p_o}\right) \\ \sqrt{(x_{k,t} - x^p_o)^2 + (y_{k,t} - y^p_o)^2} \end{bmatrix} + \boldsymbol{n}^p_{m,t}, \tag{47}$$

where the observation noise $\boldsymbol{n}^p_{m,t}$ is also a zero mean Gaussian random variable with covariance matrix $\boldsymbol{\Sigma}_n$, defined as:

$$\boldsymbol{\Sigma}_n = \begin{bmatrix} \sigma^2_\theta & 0 \\ 0 & \sigma^2_r \end{bmatrix}. \tag{48}$$

Parameters	Values
σ_x^2, σ_y^2	3×10^{-3}
$\sigma_{x_o}^2, \sigma_{y_o}^2$	3×10^{-4}
$\sigma_\theta^2, \sigma_r^2$	$0.003, 100$
T_s	1

Table 1. Parameters for the scenario generation.

σ_θ^2 and σ_r^2 are the observation noise variances for bearings and range, respectively. The parameters used for synthesising the scenario are listed in Table 1. In this experiment, targets appear at time 1 with initial values $\boldsymbol{x}_{1,0} = [-40, 1.001, 40, -1.055]^T$ and $\boldsymbol{x}_{2,0} = [40, -1.001, 40, -1.055]^T$, and both last $T = 140$ time steps. The initial state of the second sensor is assigned as $\boldsymbol{\varphi}_0^1 = [50, -0.3, 50, -0.3]^T$. An overall view of the synthesised observations is plotted in Figure 1.

At this stage, we run our algorithms with 400 particles which are initialised according to a Gaussian distribution around the true state with covariance matrix $\Sigma_0 = diag[10, 0.1, 10, 0.1]$. The initial target and sensor states for the Gibbs sampling in our algorithms are sampled form their dynamic priors, i.e. $p(\boldsymbol{x}_t|\hat{\boldsymbol{x}}_{t-1})$ and $p(\boldsymbol{\varphi}_t|\hat{\boldsymbol{\varphi}}_{t-1})$, where $\hat{\boldsymbol{x}}_{t-1}$ and $\hat{\boldsymbol{\varphi}}_{t-1}$ are the mean of the target and sensor posterior distributions at $t-1$ respectively. By adopting the MH approach, the proposal transition functions is set as a Gaussian mixture, given as:

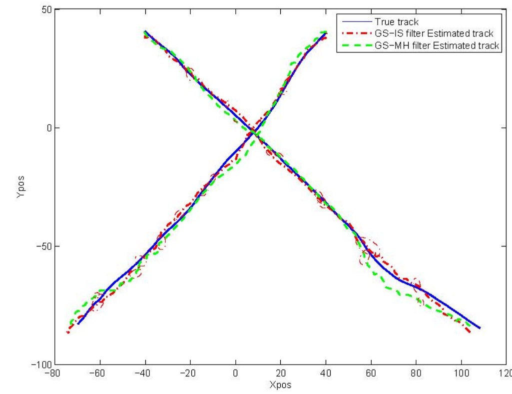
$$\pi(\boldsymbol{x}_t^c|\boldsymbol{x}_t^l) = \beta_{\boldsymbol{x}} \mathcal{N}(\boldsymbol{x}_t^l; \Sigma_{\boldsymbol{x}}^1) + (1-\beta_{\boldsymbol{x}})\mathcal{N}(\boldsymbol{x}_t^l; \Sigma_{\boldsymbol{x}}^2),$$
$$\pi(\boldsymbol{\varphi}_t^c|\boldsymbol{\varphi}_t^l) = \beta_{\boldsymbol{\varphi}} \mathcal{N}(\boldsymbol{\varphi}_t^l; \Sigma_{\boldsymbol{\varphi}}^1) + (1-\beta_{\boldsymbol{\varphi}})\mathcal{N}(\boldsymbol{\varphi}_t^l; \Sigma_{\boldsymbol{\varphi}}^2).$$

On the other hand, the prior distributions in (44) and (46) are chosen as the proposal functions for the target and sensor state when the Importance sampling method is used. A summary of the parameters used in our algorithms is listed in Table 2, and a realisation of the result is shown in Figure 2.

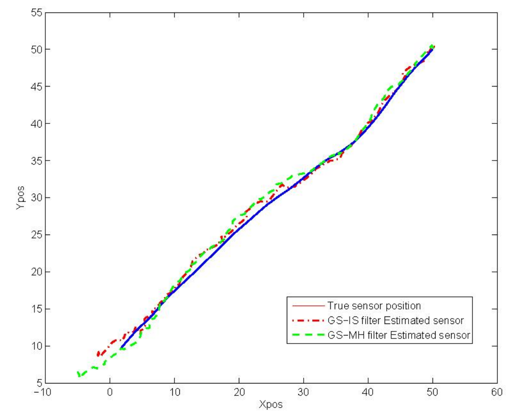
Parameters	Values
N	400
L (burn-in size)	450 (50)
$\beta_{\boldsymbol{x}} = \beta_{\boldsymbol{\varphi}}$	0.5
$\Sigma_{\boldsymbol{x}}^1$	$diag[10, 0.1, 10, 0.1]$
$\Sigma_{\boldsymbol{x}}^2$	$diag[100, 0.1, 100, 0.1]$
$\Sigma_{\boldsymbol{\varphi}}^1$	$diag[1, 0.01, 1, 0.01]$
$\Sigma_{\boldsymbol{\varphi}}^2$	$diag[5, 0.01, 5, 0.01]$
$\mu_{k,T}^j, \mu_C^j$	1, 10

Table 2. Parameters for tracking algorithms setup.

In addition, a total of $N_{RMSE} = 10$ independent runs with the same synthesised tracks but re-generated observations are



(a)Targets position estimation.



(b)Sensor position estimation.

Figure 2. A simulation of target tracking with sensor registration using proposed algorithms.

conducted to compare the performance of the joint target and sensor state estimation of the proposed algorithms. Performance is measured in terms of the Root Mean Square Error (RMSE), defined as:

$$RMSE(\boldsymbol{x}) = \sqrt{\frac{1}{N_{RMSE}T} \sum_{r=1}^{N_{RMSE}} \sum_{t=1}^{T} ||\boldsymbol{x}_t - \hat{\boldsymbol{x}}_t^r||^2}, \quad (49)$$

where $\hat{\boldsymbol{x}}_t^r$ is the estimated target state at time t for rth run. Likewise $RMSE(\boldsymbol{\varphi})$ is defined for the sensor state estimation error.

Furthermore we adopt the standard PF [1] to track targets using measurements from the unresolved sensor position. In this case, we assume the sensor moves in a straight line whose position is interpolated from the true initial sensor state via equation,

$$\boldsymbol{\varphi}_t = \boldsymbol{A}\boldsymbol{\varphi}_{t-1} \quad (50)$$

where $\boldsymbol{A}$ is defined in Eq (45). The comparison results are summarised in Table 3. From Figure 2 and Table 3, we can see that our proposed algorithms can jointly track target and sensor states in a hostile environment. As expected, without taking the uncertainty of sensor position into account the standard PF is outperformed by our algorithms.

Algorithms	GS-IS	GS-MH	Standard PF
$RMSE(\boldsymbol{x})$	4.2356	4.437	10.4308
$RMSE(\boldsymbol{\varphi})$	3.8385	4.2559	-

Table 3. RMSE evaluated for different algorithms: GS-IS stands for the Gibbs sampling using Importance sampling, and GS-MH stands for the MH within the Gibbs sampling.

Experiment II

In this experiment we evaluate the performance of the proposed RBPF algorithm for tracking targets in the presence of constant, unknown measurement bias $\boldsymbol{\Delta}_{\text{bias}} = \{\boldsymbol{\Delta}^p_{\text{bias}}\}_{p=0}^{N_o-1}$, where $\boldsymbol{\Delta}^p_{\text{bias}}$ is the bias of the pth sensor. Accordingly the observation model can be expressed as

$$\boldsymbol{y}^p_{m,t} = \tag{51}$$

$$\begin{bmatrix} \tan^{-1}\left(\frac{y_{k,t}-y^p_o}{x_{k,t}-x^p_o}\right) \\ \sqrt{(x_{k,t}-x^p_o)^2+(y_{k,t}-y^p_o)^2} \end{bmatrix} + \boldsymbol{\Delta}^p_{\text{bias}} + \boldsymbol{n}^p_{m,t}, \tag{52}$$

where $\boldsymbol{\Delta}^p_{\text{bias}} = [\theta^p_{\text{bias}}, r^p_{\text{bias}}]^T$ are the constant bearing and range biases for the sensor p respectively.

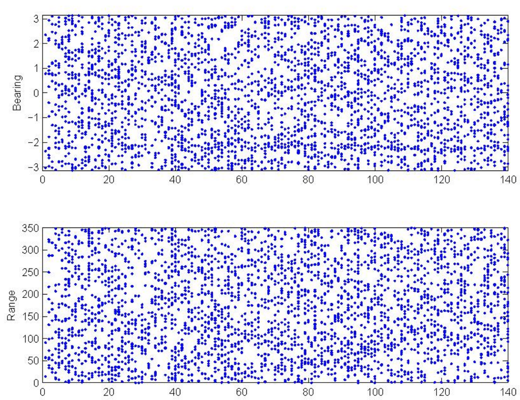


Figure 3. Synthesised observations with $P_d = 0.5$, and $\lambda_C = 10$.

In this experiment, two stationary sensors are deployed. They are located at $(0, 0)$ and $(50, 50)$. The former is the reference sensor without any bias ($\boldsymbol{\Delta}^0_{\text{bias}} = \mathbf{0}$), and the latter has a constant measurement bias $\boldsymbol{\Delta}^1_{\text{bias}} = [+0.1\text{rad}, +10]^T$. Two targets and other parameters that describe the environment conditions, including λ_C, P_d, etc., are identical to those in the Experiment I. An example of synthesised observations is given in Figure 3.

We model the unknown measurement bias as a random walk given by

$$\bar{\boldsymbol{y}}^1_t = [\theta^1_{\text{bias},t}, r^1_{\text{bias},t}]^T = \bar{\boldsymbol{y}}^1_{t-1} + \boldsymbol{u}^1_{t-1}. \tag{53}$$

where $\boldsymbol{u}^1_t$ is the zero mean Gaussian noise with a fix and known covariance matrix Σ_u. The reference sensor is unbiased, that is $\bar{\boldsymbol{y}}^0_t = [0, 0]^T, \forall t$. A total number of 1500 particles are adopted in this experiment. The particles of targets' state are initialised as same as in the Experiment I, whereas the particles of measurement bias are initialised as a Gaussian distribution with mean of $[0, 0]^T$ and covariance matrix of $diag[0.1, 10]$. A summary of the parameters used in the filter setup is given in Table 4 and an instance tracking result is plot in Figure 4.

Parameters	**Values**
N	1500
σ^2_x, σ^2_y	3×10^{-3}
$\sigma^2_\theta, \sigma^2_r$	$0.003, 100$
Σ_u	$diag[0.1 \times 10^{-4}, 0.7 \times 10^{-3}]$

Table 4. Parameters for tracking algorithms setup in Experiment II.

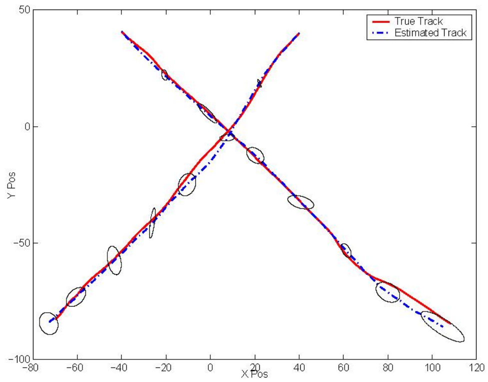
In addition we compare our algorithm with the standard particle filter which does not take the bias into account. A total of $N_{RMSE} = 10$ independent runs with the same synthesised tracks but re-generated observations are adopted to evaluate the RMSE of the target state estimation for both algorithms. A comparison of the results is given in Table 5. From Figure 4 and Table 5, we can see that our proposed algorithm can jointly track the target state and resolve the measurement bias. It is clear that by taking the measurement bias into account, our algorithm outperforms the standard particle filter in terms of the RMSE of the target state estimation.

Algorithms	RBPF	Standard PF
$RMSE(\boldsymbol{x})$	3.8067	12.7608
$RMSE(\theta_{bias})$	0.0128	-
$RMSE(r_{bias})$	1.5708	-

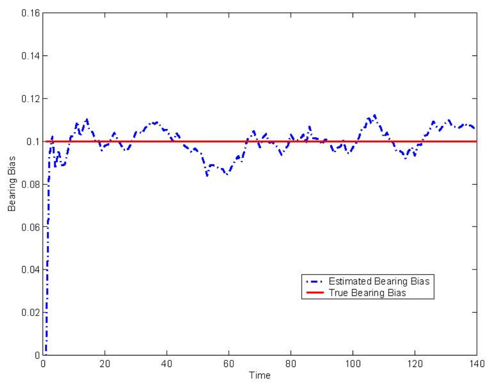
Table 5. RMSE evaluated for different algorithms.

5. CONCLUSIONS

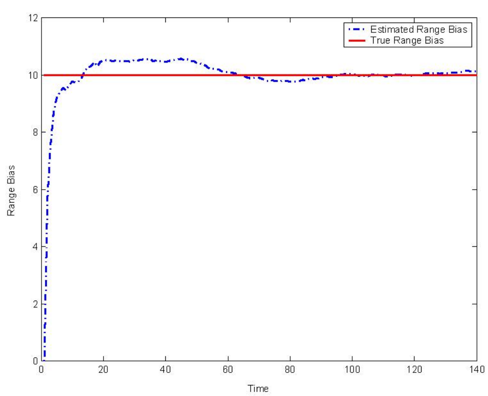
In this paper we have presented two sequential Monte Carlo approaches for joint target tracking and sensor registration. In particular the first one uses Gibbs sampling method to iteratively sample target and sensor states from conditional posterior distributions. Two possible sampling approaches, MH and Importance Sampling, were introduced. Secondly we have also introduced using RBPF to efficiently track targets when the sensor has a nearly constant measurement bias. The demonstration showed that the proposed approaches are very robust in jointly tracking the moving targets and resolving the sensor uncertainty within the environment exhibiting low detection probability and high clutter spatial density.



(a)Targets position estimation.



(b)Sensor bearing measurement bias estimation.



(c)Sensor range measurement bias estimation.

Figure 4. A simulation of target tracking with sensor registration using RBPF.

There are several outstanding research areas remaining for improvement of the proposed algorithms. For instance the proposed algorithms will be extended to account for dealing with more manoeuvring targets by integrating other dynamic models, e.g. variable rate intrinsic model.

REFERENCES

[1] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, February 2002.

[2] Y. Bar-Shalom and T. E. Fortmann. *Tracking and Data Association*. Academic Press, 1988.

[3] Yaakov. Bar-Shalom and W. D. Blair. *Multitarget-Multisensor Tracking: Applications and Advances*, volume III. Archtech House, Norwood, MS, 2000.

[4] S. Blackman and R. Popoli. *Design and Analysis of Modern Tracking Systems*. Artech House, Norwood, MA, 1999.

[5] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo in Practice*. Springer-Verlag, New York, 2001.

[6] K. Gilholm and D.Salmond. Extended object and group tracking. In *In RTO SET Symposium on Target tracking and sensor data fusion for military observation systems, RTO-MP-SET-059, NATO, Budapest, Hungary*, October 2003.

[7] K. Gilholm, S. J. Godsill, S. Maskell, and D. J. Salmond. Poisson models for extended target and group tracking. In *Proceedings of SPIE*, 2005.

[8] K. Gilholm and D. Salmond. A spatial distribution model for tracking extended objects. *IEE Proceedings on Radar, Sonar and Navigation*, 152(5), 2005.

[9] Zia Khan, Tucker Balch, and Frank Dellaert. MCMC-Based Particle Filtering for Tracking a Variable Number of Interacting Targets. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 27:1805–1819, November 2005.

[10] D. McMichael and N. Okello. Maximum likelihood registration of dissimilar sensors. In *Proceedings of the First Australian Data Fusion Symposium*, Adelaide, Australia, 1996.

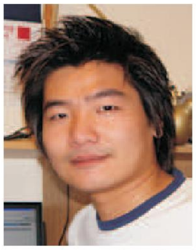
[11] Nassib Nabaa and Robert H. Bishop. A recursive solution to the sensor registration problem in a multiple sensor tracking scenario. In *In SPIE, Signal and Data Processing of Small Targets*, 1996.

[12] N N Okello and S Challa. Joint sensor registration and track-to-track fusion for distributed trackers. *IEEE Transactionson Aerospace and Electronic Systems*, 40:808–823, 2004.

[13] C.P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer-Verlag, New York, 1999.

[14] J. Vermaak, S. Maskell, and M. Briers. Online Sensor Registration. *IEEE Aerospace Big Sky conference*, 2005.

[15] Y. Zhou, H Leung, and M. Yip. An exact maximum lieklihood registration algorithm for data fusion. *IEEE Transactions on Signal Processing*, 45(6):1560–1573, June 1997.



Jack Li *received the B.Eng. Degree from the University of Edinburgh, UK in electronic electrical engineering in 2003. Currently, he is pursuing the Ph.D. Degree in Statistical Signal Processing Group in the Information Engineering Division, Cambridge University Engineering Department. His research focuses on developing Sequential Monte Carlo (SMC) methods for Multitarget tracking problems.*

William Ng *was born in Hong Kong. He received the B.Eng. degree from the University of Western Ontario, London, ON, in electrical engineering in 1994, the M.Eng. and Ph.D. degrees from McMaster University, Hamilton, ON, in 1996 and 2004, respectively, both in electrical engineering, and the M.M.Sc. from the University of Waterloo, Waterloo, ON, in management sciences in 2004. From 1996 to 1999, he was with Forschungszentrum Informatik, Karlsruhe, Germany, developing an expert system using neural networks for nondestructive pipeline evaluation, and from 1999 to 2002 he was with the Pressure Pipe Inspection Company Ltd., Missisauga, ON, where he was the head of Software and IT department. Currently, he is working in the Signal Processing Group at the University of Cambridge, Cambridge, U.K., as a research associate. His research interests include statistical signal processing for sensor arrays and multitarget tracking and multisource information fusion. Dr. Ng is a registered professional engineer in the province of Ontario since 2000. For more information see www-sigproc.eng.cam.ac.uk/~kfn20.*

Simon Godsill *is Reader in Statistical Signal Processing in the Engineering Department of Cambridge University. He is an Associate Editor for IEEE Trans. Signal Processing and the journal Bayesian Analysis, and is a member of IEEE Signal Processing Theory and Methods Committee. He has research interests in Bayesian and statistical methods for signal processing, Monte Carlo algorithms for Bayesian problems, modelling and enhancement of audio and musical signals, source separation, tracking and genomic signal processing. He has published extensively in journals, books and conferences. He has co-edited in 2002 a special issue of IEEE Trans. Signal Processing on Monte Carlo Methods in Signal Processing and a recent special issue of the Journal of Applied Signal Processing, and organised many conference sessions on related themes.*