2005 7th International Conference on Information Fusion (FUSION)

# A Comparison of Particle Filters for Recursive Track-before-detect

Mark G. Rutten, Branko Ristic and Neil J. Gordon
Intelligence, Surveillance and Reconnaissance Division,
Defence Science and Technology Organisation, PO Box 1500, Edinburgh, SA 5111, Australia.

*Abstract*— **Track-before-detect is a powerful technique for detection and tracking of targets with low signal-to-noise ratio. This paper presents a performance comparison of two particle filters recently proposed for this application using several different particle proposal densities designed for track initiation. The first particle filter is a standard SIR particle filter, which treats the track-before-detect problem as a hybrid estimation problem by incorporating a discrete random variable, "target existence," into the state vector. The second particle filter formulates the probability of existence calculation in a different way, avoiding the need for hybrid estimation. Three different particle proposal densities are considered, which are designed to compare performance when the data is used to aid in particle proposal.**

## I. Introduction

A traditional tracking filter uses detections from the data as measurements for target state estimation. These detections are typically made by performing a thresholding process on the output of a sensor's signal processing stage, keeping that portion of the data which exceeds the threshold and disregarding the remainder. There has been some recent work in the literature on particle-based track-before-detect filters based on the work of Salmond and Birch [1] and Boers and Driessen [2]. These filters use the entire output of the signal processing stage as the measurement for the filter, which retains as much of the sensor information as possible. Maximising the data available to the tracker has advantages for tracking target returns having a low signal-to-noise ratio (SNR).

This paper describes and compares two particle filters designed for recursive track-before-detect. The filters perform detection by incorporating a binary target existence variable into the state estimation process. The probability of the target existing in the data can be calculated from this variable, indicating the presence or absence of a target in the data. The first of the filters considered is an orthodox SIR filter, as developed in [3, Ch. 11]. While the first filter augments the target state space with an existence variable, the second filter is derived such that the probability of existence is calculated as a separate process, using the weights of the particles describing the target state [4].

Three different particle proposal densities are considered for particle initiation. At each time step both algorithms employ some particles to search the data for a target (which will be called *birth* particles). The way in which these particles are placed is controlled by the proposal density and the effect of this density on the overall performance of the algorithms is investigated. One of the proposals simply places particles uniformly around the surveillance region. Another uses the data to aid in proposal, only placing the particles in the 200 highest intensity bins of the data. The third proposal is a combination of the two, placing half of the particles uniformly and half only in the highest intensity bins.

Section II introduces the assumed measurement and target dynamic processes and, in particular, describes the likelihood function which is used by both of the TBD algorithms. Section III gives details of the two algorithms compared in this paper, while Section IV describes the three different proposal densities used to initiate particles. Section V describes the simulations made and shows Monte-Carlo results of the six algorithm variations, comparing the performance of each in terms of detection performance and RMS error. Section VI draws some conclusions on the relative performance of the algorithms and proposals based on the results of the simulations.



## II. Target and Signal Models

It is assumed that the target state, $\mathbf{x}_k$, evolves according to a discrete-time linear Gaussian target motion model

$$\mathbf{x}_k = F\mathbf{x}_{k-1} + v_k, \tag{1}$$

where $v_k$ is a zero mean Gaussian noise process with covariance $Q$. The target state vector is composed of position and velocity in the $x$ and $y$ dimensions and it is assumed that the return intensity from the target, $I_k$ is unknown and thus forms part of the target state. Hence the target state vector can be written as

$$\mathbf{x}_k = \begin{bmatrix} x_k & \dot{x}_k & y_k & \dot{y}_k & I_k \end{bmatrix}^T. \tag{2}$$

A constant-velocity process model is used, which is defined by the transition matrix

$$F = \begin{bmatrix} 1 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & T & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

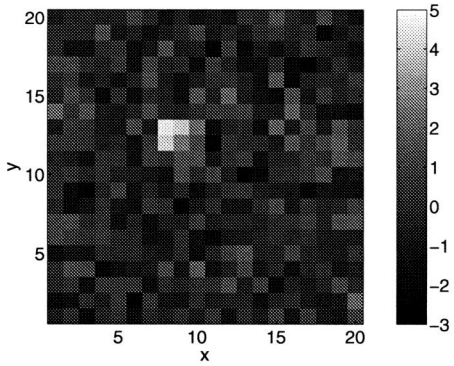



Fig. 1. Simulated measurement data, with a 17dB ($I_k \approx 22$) target return at $x = 8.5$ and $y = 12.5$. $\Delta_x = \Delta_y = 1$, $\Sigma = 0.7$ and $\sigma = 1$.

and the process noise covariance

$$Q = \begin{bmatrix} q_s T^3/3 & q_s T^2/2 & 0 & 0 & 0 \\ q_s T^2/2 & q_s T & 0 & 0 & 0 \\ 0 & 0 & q_s T^3/3 & q_s T^2/2 & 0 \\ 0 & 0 & q_s T^2/2 & q_s T & 0 \\ 0 & 0 & 0 & 0 & q_i T \end{bmatrix}, \quad (4)$$

where $T$ is the sampling period, $q_s$ is the power spectral density of the acceleration noise in the spatial dimensions and $q_i$ is the power spectral density of the noise in the rate of change of target return intensity.

The presence of a target in the data is calculated through a target existence variable [5], [6]. Probability of target existence is modelled as a two-state Markov chain, with a target either existing in the data, $E_k = 1$ or absent from the data, $E_k = 0$. The probability transitions are defined by the matrix

$$\Pi = \begin{bmatrix} 1 - P_d & P_b \\ P_d & 1 - P_b \end{bmatrix}, \quad (5)$$

where $P_b$ denotes the probability of target *birth* and $P_d$ denotes the probability of target *death*.

The result of the sensor signal processing is a series of images. As in [3, Ch. 11], the signal is assumed to be a 2-dimensional image consisting of $n$ cells in the $x$-dimension and $m$ cells in the $y$-dimension. An example of the data used for simulation in this paper is shown in Figure 1, although larger images are used in Section V ($64 \times 64$, rather than $20 \times 20$). The measurement at each time $k$ is the intensity of each cell in the image. For each cell, indexed by $i \in \{1,\ldots,n\}$ and $j \in \{1,\ldots,m\}$, the measured intensity is modelled by

$$z_k^{(i,j)} = \begin{cases} h^{(i,j)}(\mathbf{x}_k) + w_k^{(i,j)}, & \text{target present} \\ w_k^{(i,j)}, & \text{no target,} \end{cases} \quad (6)$$

that is the intensity in the cell $(i,j)$ contains a contribution from the target, $h^{(i,j)}(\mathbf{x}_k)$, plus noise, $w_k^{(i,j)}$, if a target is present and noise only if there is no target in the data. The complete measurement at time $k$ is thus given by $z_k = \{z_k^{(i,j)} | i = 1,\ldots,n, j = 1,\ldots,m\}$ and the set of all measurements up to time $k$ is denoted by $Z_k = \{z_l | l = 1,\ldots,k\}$.

In general a target will contribute to more than one cell. This spread of the target in the data is modelled by a 2-dimensional Gaussian with known variance, $\Sigma^2$, giving

$$h^{(i,j)}(\mathbf{x}_k) = \frac{\Delta_x \Delta_y I_k}{2\pi\Sigma^2} \exp\left( -\frac{(\Delta_x i - x_k)^2 + (\Delta_x j - y_k)^2}{2\Sigma^2} \right), \quad (7)$$

where $\Delta_x$ and $\Delta_y$ denote the size of a cell in the $x$ and $y$ dimensions respectively. In this work it is assumed that the measurement noise process, $w_k^{(i,j)}$ in (6), is Gaussian distributed with zero mean, variance $\sigma^2$ and is independent between cells in the image.

Under the assumptions of Gaussian measurement noise, which is independent from cell-to-cell, the likelihood function in the presence and absence of a target, can be written as

$$p(z_k^{(i,j)} | \mathbf{x}_k, E_k = 1) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left( -\frac{\left[ z_k^{(i,j)} - h^{(i,j)}(\mathbf{x}_k) \right]^2}{2\sigma^2} \right) \quad (8)$$

$$p(z_k^{(i,j)} | E_k = 0) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left( -\frac{\left[ z_k^{(i,j)} \right]^2}{2\sigma^2} \right), \quad (9)$$

respectively, and then

$$p(z_k | \mathbf{x}_k, E_k) = \prod_{i=1}^{n} \prod_{j=1}^{m} p(z_k^{(i,j)} | \mathbf{x}_k, E_k). \quad (10)$$

In order to limit the number of cells included in the likelihood it is observed that the target's only significant contribution to the likelihood will be local and so when a target is present (10) can be approximated by

$$p(z_k | \mathbf{x}_k, E_k = 1) \approx \prod_{i \in C_x(\mathbf{x}_k)} \prod_{j \in C_y(\mathbf{x}_k)} p(z_k^{(i,j)} | \mathbf{x}_k, E_k = 1) \times \prod_{i \notin C_x(\mathbf{x}_k)} \prod_{j \notin C_y(\mathbf{x}_k)} p(z_k^{(i,j)} | \mathbf{x}_k, E_k = 0), \quad (11)$$

where $C_x(\mathbf{x}_k)$ and $C_y(\mathbf{x}_k)$ are the set of cell indices containing the target in the $x$ and $y$ dimensions respectively. It is thus convenient to use the *likelihood ratio*, which can be written in terms of (10), with $E_k = 0$ or 1, giving

$$\ell(z_k | \mathbf{x}_k, E_k) = \begin{cases} \frac{p(z_k | \mathbf{x}_k, E_k = 1)}{p(z_k | E_k = 0)}, & E_k = 1 \\ 1 & E_k = 0, \end{cases} \quad (12)$$

where

$$\ell(z_k | \mathbf{x}_k, E_k = 1) \approx \prod_{i \in C_x(\mathbf{x}_k)} \prod_{j \in C_y(\mathbf{x}_k)} \exp\left( -\frac{h^{(i,j)}(\mathbf{x}_k) \left[ h^{(i,j)}(\mathbf{x}_k) - 2 z_k \right]}{2\sigma^2} \right). \quad (13)$$

## III. Particle Filters for Recursive TBD

### A. Standard SIR Filter

The algorithm outlined here is discussed in detail in [3, Ch. 11], and is based on the work of [1]. The target state is not defined when the target does not exist in the data, so assuming that the target does exist in the data the density describing its state can be expanded using Bayes' rule

$$p(\mathbf{x}_k, E_k = 1|Z_k) = \frac{p(z_k|\mathbf{x}_k, E_k = 1)p(\mathbf{x}_k, E_k = 1|Z_{k-1})}{p(z_k|Z_{k-1})} \quad (14)$$

$$\propto \ell(z_k|\mathbf{x}_k, E_k = 1)p(\mathbf{x}_k, E_k = 1|Z_{k-1}), \quad (15)$$

since the likelihood when the target is absent, (9), is independent of the target state. The predicted target state can be written in terms of the target state and existence at the previous time, giving

$$\begin{aligned} &p(\mathbf{x}_k, E_k = 1|Z_{k-1}) \\ &= \int p(\mathbf{x}_k, E_k = 1|\mathbf{x}_{k-1}, E_{k-1} = 1)\times \\ &\quad p(\mathbf{x}_{k-1}, E_{k-1} = 1|Z_{k-1})d\mathbf{x}_{k-1}+ \\ &\quad p(\mathbf{x}_k, E_k = 1, E_{k-1} = 0|Z_{k-1}) \end{aligned} \quad (16)$$

$$\begin{aligned} &= \int p(\mathbf{x}_k|\mathbf{x}_{k-1}, E_k = 1, E_{k-1} = 1)\times \\ &\quad [1 - P_d]p(\mathbf{x}_{k-1}, E_{k-1} = 1|Z_{k-1})d\mathbf{x}_{k-1}+ \\ &\quad p_b(\mathbf{x}_k)P_b, \end{aligned} \quad (17)$$

where $p_b(\mathbf{x}_k)$ is the prior density of a target when it has appeared in the data between times $k-1$ and $k$. Thus by substituting (17) into (15) the posterior density at time $k$ can be written in terms of the likelihood ratio (12), the transition density defined by (1) and (5) and the posterior density at time $k-1$.

A particle filter can be used to directly approximate the densities required by (15). In this case a hybrid estimation technique is used, where the state is augmented to include the existence variable

$$\mathbf{y}_k = \begin{bmatrix} \mathbf{x}_k & E_k \end{bmatrix}^T. \quad (18)$$

Thus the set of particles is implicitly divided into two classes, those particles describing "alive" particles, with $E_k = 1$, that describe a target state and those particles describing "dead" particles, with $E_k = 0$, where the target state is undefined. Given a set of particles describing the posterior at time $k-1$, $\{\mathbf{y}^i_{k-1}|i = 1,\ldots,N\}$, with uniform weights, and $N_a$ "alive" particles, the algorithm contains the following steps

1) Randomly select $N_a P_d$ of the "alive" particles and assign $E^i_k = 0$ for each particle $i$ in the selected set. These particles have "died".
2) Randomly select $(N - N_a)P_b$ of the "dead" particles and assign $E^i_k = 1$ for each particle $i$ in the selected set. These are the "birth" particles. Set the birth particle state as a sample from the proposal density

$$\mathbf{x}^i_k \sim q(\mathbf{x}_k|E_k = 1, E_{k-1} = 0, z_k). \quad (19)$$

3) The remaining "alive" particles have survived from time $k-1$. These particles are assigned $E^i_k = 1$ and have a state sampled from the proposal

$$\mathbf{x}^i_k \sim q(\mathbf{x}_k|\mathbf{x}_{k-1}, E_k = 1, E_{k-1} = 1, z_k). \quad (20)$$

4) For each particle $i = 1,\ldots,N$, compute the unnormalised weight using the likelihood ratio (12)

$$\tilde{w}^i_k = \ell(z_k|\mathbf{x}^i_k, E^i_k = 1), \quad (21)$$

for particles with $E^i_{k-1} = 1$ and

$$\tilde{w}^i_k = \frac{\ell(z_k|\mathbf{x}^i_k, E^i_k = 1)p(\mathbf{x}^i_k|E^i_k = 1, E^i_{k-1} = 0)}{q(\mathbf{x}^i_k|E^i_k = 1, E^i_{k-1} = 0, z_k)}, \quad (22)$$

for those with $E^i_{k-1} = 0$.
5) Normalise the weights

$$w^i_k = \frac{\tilde{w}^i_k}{\sum_{j=1}^N \tilde{w}^j_k}. \quad (23)$$

6) Resample the particles $\{\mathbf{y}^i_k|i = 1,\ldots,N\}$, with weights $\{w^i_k|i = 1,\ldots,N\}$, where $\mathbf{y}^i_k = [\mathbf{x}^i_k \; E^i_k]^T$.

The set of particles $\{\mathbf{y}^i_k|i = 1,\ldots,N\}$, with uniform weights then approximates the density $p(\mathbf{x}_k, E_k|Z_k)$.

The proposal function for particles that exist at both time $k$ and $k-1$, $q(\mathbf{x}_k|\mathbf{x}_{k-1}, E_k = 1, E_{k-1} = 1, z_k)$, is defined by the system dynamics (1), resulting in the simple form of the weight update given by (21). Several options will be explored for the proposal of birth particles, $q(\mathbf{x}_k|E_k = 1, E_{k-1} = 0, z_k)$, which will be described in more detail in Section IV.

An estimate of the probability of existence and the target state can be made from the set of particles resulting from the algorithm above. If $\bar{E}_k = \{i|E^i_k = 1\}$ denotes the set of alive particles, then the probability of existence is given by

$$\hat{P}_k = \frac{|\bar{E}_k|}{N}, \quad (24)$$

and the target state estimate

$$\hat{\mathbf{x}}_k = \frac{\sum_{i\in\bar{E}_k} \mathbf{x}^i_k}{|\bar{E}_k|}. \quad (25)$$

### B. Filter with Efficient Probability of Existence Calculation

In this algorithm the same density is estimated, $p(\mathbf{x}, E_k = 1|Z_k)$, but a different expansion is used, which alleviates the need to include target existence as part of each particle state vector. This results in a more efficient use of particles, since there are no "dead" particles, only "alive" particles. This algorithm is comparable to that presented by Driessen and Boers [7].

As stated in Section III-A, the target state is undefined if a target does not exist in the data, so the target state can be written as

$$p(\mathbf{x}_k, E_k = 1|Z_k) = p(\mathbf{x}_k|E_k = 1, Z_k)P_k, \quad (26)$$

where $P_k = P(E_k = 1|Z_k)$. It can be seen that starting the derivation in this way separates the target state and the probability of existence. The first factor in (26) is the target state, conditioned

on the target existence, which can be expanded over target existence at the previous time

$$p(\mathbf{x}_k|E_k=1,Z_k) = p(\mathbf{x}_k|E_k=1,E_{k-1}=1,Z_k)P(E_{k-1}=1|E_k=1,Z_k)+ p(\mathbf{x}_k|E_k=1,E_{k-1}=0,Z_k)P(E_{k-1}=0|E_k=1,Z_k). \quad (27)$$

This expansion results in the target state as a weighted sum of two densities. The first density describes the case where the target existed at both times $k-1$ and $k$, called the *continuing density*. The second density describes the case where the target did not exist at time $k-1$, but exists at time $k$, called the *birth density*.

The continuing and birth densities can be rewritten using Bayes' rule, giving [4]

$$p(\mathbf{x}_k|E_k=1,E_{k-1}=1,Z_k) \propto \ell(z_k|\mathbf{x}_k,E_k=1)p(\mathbf{x}_k|E_k=1,E_{k-1}=1,Z_{k-1}), \quad (28)$$

and

$$p(\mathbf{x}_k|E_k=1,E_{k-1}=0,Z_k) \propto \ell(z_k|\mathbf{x}_k,E_k=1)p(\mathbf{x}_k|E_k=1,E_{k-1}=0). \quad (29)$$

The predicted density in (28) can be written in terms of the process transition density and posterior target state at time $k-1$

$$p(\mathbf{x}_k|E_k=1,E_{k-1}=1,Z_{k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1},E_k=1,E_{k-1}=1)\times p(\mathbf{x}_{k-1}|E_{k-1}=1,Z_{k-1})d\mathbf{x}_{k-1}, \quad (30)$$

while the density in (29), $p(\mathbf{x}_k|E_k=1,E_{k-1}=0)=p_b(\mathbf{x}_k)$, is defined in Section III-A.

The first mixing probability in (27) can be expanded as

$$P(E_{k-1}=1|E_k=1,Z_k) \propto \ell(z_k|E_k=1,E_{k-1}=1,Z_{k-1})[1-P_d]P_{k-1}, \quad (31)$$

where

$$\ell(z_k|E_k=1,E_{k-1}=1,Z_{k-1}) = \int \ell(z_k|\mathbf{x}_k,E_k=1)p(\mathbf{x}_k|E_k=1,E_{k-1}=1,Z_{k-1})d\mathbf{x}_k. \quad (32)$$

Thus the mixing term can be written in terms of the probability of target death, the probability of target existence at $k-1$, the likelihood ratio (12) and the predicted density (30). The second mixing probability is

$$P(E_{k-1}=0|E_k=1,Z_k) \propto \ell(z_k|E_k=1,E_{k-1}=0)P_b[1-P_{k-1}], \quad (33)$$

where

$$\ell(z_k|E_k=1,E_{k-1}=0) = \int \ell(z_k|\mathbf{x}_k,E_k=1)p(\mathbf{x}_k|E_k=1,E_{k-1}=0)d\mathbf{x}_k. \quad (34)$$

which is written in terms of the probability of target birth, the probability of target existence at $k-1$, the likelihood ratio (12) and the target prior.

The probability that a target exists in the data can be written as

$$P(E_k=1|Z_k) = C_E^{-1}\{\ell(z_k|E_k=1,E_{k-1}=1,Z_{k-1})[1-P_d]P_{k-1}+ \ell(z_k|E_k=1,E_{k-1}=0)P_b[1-P_{k-1}]\}, \quad (35)$$

where $C_E$ is a normalisation term and the likelihoods are defined in (32) and (34) respectively. The normalisation constant is given by

$$C_E = \ell(z_k|E_k=1,E_{k-1}=1,Z_{k-1})[1-P_d]P_{k-1}+ \ell(z_k|E_k=1,E_{k-1}=0)P_b[1-P_{k-1}]+ P_dP_{k-1}+[1-P_b][1-P_{k-1}]. \quad (36)$$

The particle filter implementation of the above derivation proceeds by approximating both the birth and continuing densities using separate sets of particles. The mixing probabilities and probability of existence can then be calculated based on these particle sets and finally the two sets of particles are combined to form an approximation of the posterior target state. Starting with a set of $N_c$ particles $\{\mathbf{x}_{k-1}^i|i=1\ldots N_c\}$ describing the posterior target state at time $k-1$ and an estimate of the probability of existence at time $k-1$, $\hat{P}_{k-1}$, the algorithm consists of the following steps

1) Create a set of $N_b$ birth particles by sampling from one of the proposal functions defined in Section IV

$$\mathbf{x}_k^{(b)i} \sim q(\mathbf{x}_k|E_k=1,E_{k-1}=0,z_k). \quad (37)$$

The unnormalised birth particle weights are calculated using the likelihood ratio (12)

$$\tilde{w}_k^{(b)i} = \frac{\ell(z_k|\mathbf{x}_k^{(b)i},E_k^{(b)i}=1)p(\mathbf{x}_k^{(b)i}|E_k^{(b)i}=1,E_{k-1}^{(b)i}=0)}{N_b q(\mathbf{x}_k^{(b)i}|E_k^{(b)i}=1,E_{k-1}^{(b)i}=0,z_k)}, \quad (38)$$

which can be normalised

$$w_k^{(b)i} = \frac{\tilde{w}_k^{(b)i}}{\sum_{j=1}^{N_b}\tilde{w}_k^{(b)j}}. \quad (39)$$

2) Create a set of $N_c$ continuing particles by sampling from the existing proposal defined in Section III-A

$$\mathbf{x}_k^{(c)i} \sim q(\mathbf{x}_k|\mathbf{x}_{k-1},E_k=1,E_{k-1}=1,z_k), \quad (40)$$

The unnormalised weights are calculated and normalised as above

$$\tilde{w}_k^{(c)i} = \frac{1}{N_c}\ell(z_k|\mathbf{x}_k^{(c)i},E_k^{(b)i}=1) \quad (41)$$

$$w_k^{(c)i} = \frac{\tilde{w}_k^{(c)i}}{\sum_{j=1}^{N_c}\tilde{w}_k^{(c)j}}. \quad (42)$$

3) The mixing probabilities are calculated using sums of unnormalised weights

$$\tilde{M}_b = P_b[1-\hat{P}_{k-1}]\sum_{i=1}^{N_b}\tilde{w}_k^{(b)i} \tag{43}$$

$$\tilde{M}_c = [1-P_d]\hat{P}_{k-1}\sum_{i=1}^{N_c}\tilde{w}_k^{(c)i}, \tag{44}$$

which can be normalised

$$M_b = \frac{\tilde{M}_b}{\tilde{M}_b+\tilde{M}_c} \tag{45}$$

$$M_c = \frac{\tilde{M}_c}{\tilde{M}_b+\tilde{M}_c}. \tag{46}$$

4) The probability of existence at time $k$ can also be calculated in terms of unnormalised weights

$$\hat{P}_k = \frac{\tilde{M}_b+\tilde{M}_c}{\tilde{M}_b+\tilde{M}_c+P_d\hat{P}_{k-1}+[1-P_b][1-\hat{P}_{k-1}]}. \tag{47}$$

5) The particle weights describing the target state are scaled according to the mixing probabilities

$$\hat{w}_k^{(b)i} = M_b w_k^{(b)i} \tag{48}$$

$$\hat{w}_k^{(c)i} = M_c w_k^{(c)i}. \tag{49}$$

The two sets of particles can then be combined into one large set

$$\left\{(\mathbf{x}_k^{(t)i}, \hat{w}_k^{(t)i}) | i = 1,\ldots,N_t, t = c,b\right\} \tag{50}$$

6) Resample from $N_b+N_c$ down to $N_c$ particles.

Thus after completing the above steps the particles, $\{\mathbf{x}_k^i | i = 1\ldots N_c\}$, with uniform weights, approximate the posterior target state density at time $k$ and $\hat{P}_k$ is an estimate of the probability of target existence.

## IV. Proposal Densities

The source being considered in this paper generates $64\times 64 = 4096$ bins of data at each time step. While there will be a large number of particles applied in exploring this space, knowledge of the data can be used to increase the efficiency of the filters. To examine this effect three different proposal densities are considered for the *birth particles*:

1) Particle positions are uniformly distributed around the data. This is a naïve proposal, which does not exploit the data at all. The proposal density in this case is thus the prior density of the particle states and the proposal and prior densities cancel in the weight calculation (21) or (38).
2) A combination of both indiscriminant and directed particle placement. In this case half of the particles are placed uniformly around the data and the other half are placed solely in the top 200 high intensity bins. The proposal function (and thus the weight calculation) is different for both types of particles.
3) Particle positions are uniformly distributed within the 200 highest intensity bins. In this case the data is used to influence the initial position of the particles, while the remaining dimensions are sampled from the prior. The ratio of the prior density to the proposal density required by the weight calculations is then simply the ratio of the number of high intensity bins to the total number of bins, that is

$$\frac{p(\mathbf{x}_k|E_k=1,E_{k-1}=0)}{q(\mathbf{x}_k|E_k=1,E_{k-1}=0,z_k)} = \frac{200}{4096}. \tag{51}$$

The proposal for particles describing the target state (the continuing particles) remains as the system dynamics in each case. The data could conceivably be used to influence the proposal for the continuing particles, but since the target SNRs are expected to be low there is little to gain. In addition, the measurement function is highly non-linear, making a local linearisation scheme impractical and the system noise is large enough to ensure that an MCMC move step will not offer any significant advantage.

## V. Simulations and Results

To compare the three TBD filters, each filter processed exactly the same set of data using the same filter parameters. The number of particles used by each filter was set so that the processing times were closely matched. In this case the SIR filter used 21000 particles and the filter with separate probability of existence calculation (named $\text{SIR}_{P_e}$ for brevity in the remainder of the section) used 10000 particles. The filter performance is gauged on an average over 100 Monte-Carlo runs.

The simulated data consists of a total of 30 frames, with the target appearing in the data at frame 7 and disappearing at frame 23. The target process equation (1), is defined by the transition matrix $F$, given by (3), and $Q$, given by (4), with $q_s = 0.001$, $q_i = 0.01$ and $T = 1$. The probability of birth and death required by the Markov existence model (5) are defined as $P_b = P_d = 0.05$. The simulated target has initial position

$$\mathbf{x}_0 = \begin{bmatrix}54.2 & 0.45 & 47.2 & 0.25 & I_0\end{bmatrix}^T, \tag{52}$$

where $I_0$ corresponds to an intensity resulting in a 12, 6 or 3dB SNR peak in the data. The target trajectory at subsequent times is generated randomly according to the process equation. The target spread function has standard deviation $\Sigma = 0.7$. Each frame of data has a total of 64 cells in the $x$ and $y$ directions, $n = m = 64$, each cell has unit dimensions, $\Delta_x = \Delta_y = 1$ and the noise variance is set to unity, $\sigma^2 = 1$.

As discussed in the previous sections, the proposal for "alive" particles, $q(\mathbf{x}_k|\mathbf{x}_{k-1},E_k=1,E_{k-1}=1,z_k)$ for both the SIR and $\text{SIR}_{P_e}$ filters is simply the state dynamics. Both filters have been run using the three proposals for birth particles as described in Section IV, with the velocities and target intensities drawn from the following uniform distributions

$$\dot{x}_0 \sim \mathcal{U}(-1,1) \tag{53}$$

$$\dot{y}_0 \sim \mathcal{U}(-1,1) \tag{54}$$

$$I_0 \sim \mathcal{U}(4,14). \tag{55}$$

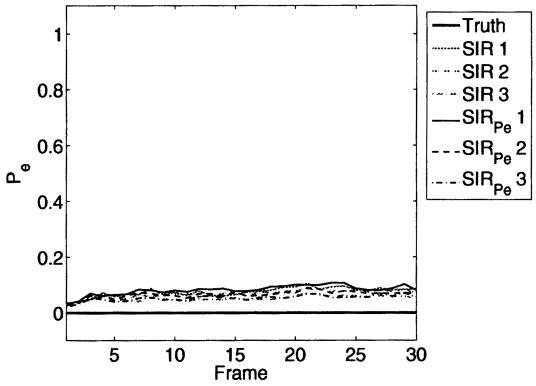



Fig. 2. Probability of existence performance with no target in the data.

The range of intensities covers peak target strengths of between approximately 2.3 and 13dB in SNR.

Figure 2 shows the average probability of existence calculated by the six filter variations when there is only noise present for the full 30 frames of data. It can be seen that the SIR and SIR$_{P_e}$ filters perform well regardless of the proposal density, with the probability of existence staying below approximately 0.1, on average, for the duration of the simulation. The stationary value of the Markov existence probability is 0.5, due to the symmetry of the birth and death probabilities, although the data drives down the value that the filters converge to. If a threshold is used on the probability of existence to indicate target detection, then 0.3 would be a sensible conservative value.

The results of the filters in terms of detection performance, summarised by the probability of existence, is shown in Figure 3 for 12, 6 and 3dB targets. As expected each of the algorithms follows the true existence of the target much more closely for the higher intensity target, with performance degrading as intensity reduces. The SIR$_{P_e}$ filter performs better for each of the proposal densities over each of the target intensities, despite the equivalence in processing time. In terms of the proposal functions it can be seen in the 6dB plot, for instance, that the third proposal density (that which places particles only around high intensity bins) performs better, in terms of probability of existence, compared with the uniform proposal. The combination of proposals performs between the two. The initiation and termination delays are similar for each proposal, but the proposals which use the data to place particles result in higher probability of existence while the target is present in the data.

The RMS position errors for each filter for each of the 12, 6 and 3dB targets is shown in Figure 4, where position error is defined as

$$\text{position error} = \sqrt{\text{error}_x^2 + \text{error}_y^2}. \tag{56}$$

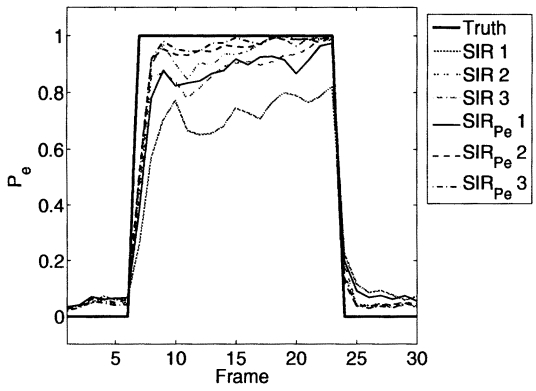
The performance of the filters in each case are compared to the posterior Cramér-Rao bound (PCRLB) [8], as derived in [3, Ch. 11], where the bound on position is defined in a similar way to (56). As can be expected there is a correlation between the target intensity, the detection performance and the RMS error, with both the detection performance and RMS error degrading with lower target intensities. The RMS error plots must be interpreted in conjunction with the corresponding probability of existence plot. For instance the target initiation delay for the 12dB target, which can be seen in the probability of existence plot, is also obvious in the RMS error plot. The filters are not confident in the target existence until the 8th or 9th frame, which is where the RMS error begins to approach the bound. Before that time the filters may not have been tracking the target and hence the RMS error is meaningless. The plots show that the performance of the SIR$_{P_e}$ is slightly better in each case, on average. The birth proposal functions resulting in better detection performance also give lower RMS error.

## VI. Conclusions

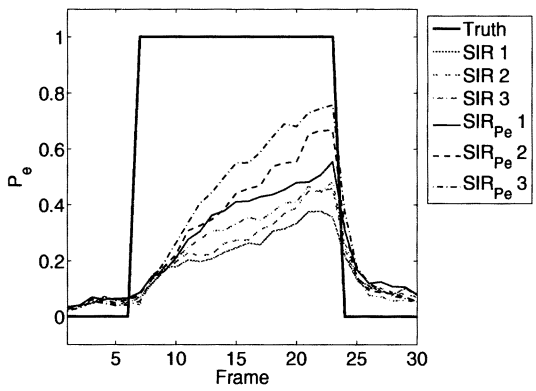
This paper has summarised and compared two particle filters designed for recursive track-before-detect in combination with several particle proposal densities. The first filter uses a hybrid estimation technique to calculate a probability of existence, which indicates the presence of a target in the data, as described in [3, Ch. 11]. The second filter separates the probability of existence and target state estimation, using the particle weights to calculate the probability of existence analytically, rather than adding a variable to the target state. The three particle proposal densities were designed to examine the effect of using the data to aid in proposal. The first density did not use the data, simply placing particles uniformly around the possible target positions. The third density placed particles only in the 200 highest intensity bins of data. The second density was a combination of the two.

Both filters performed well in simulated data, with each technique reliably detecting targets at 6dB, with occasional detections at 3dB. The SIR$_{P_e}$ algorithm results in the best combination of detection performance and positional accuracy. As one would expect, the RMS error degrades as the average detection performance drops. The filters perform at close to the PCRLB when they are confident of target presence. The proposal placing particles in only the highest intensity bins gave the best performance in detection and RMS error. The uniform proposal was the poorest performer, with the combination proposal between the two. The simulation results show that a proposal density which exploits knowledge about the form of the data gives superior performance for the type of data considered here.

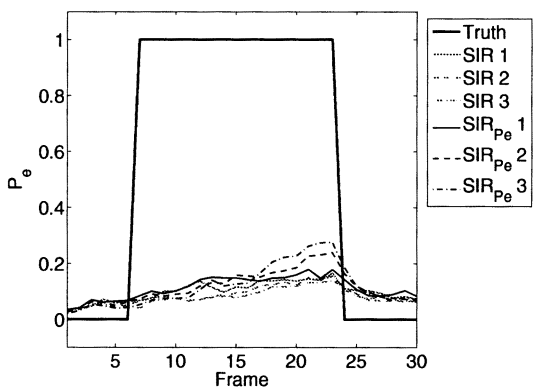
The SIR$_{P_e}$ algorithm gives improved performance over the SIR algorithm because the particle state vector describes only the target state, so the particles explore the target state space and not the existence of the target. Formulating the filter in this way ensures that there is always an equal number of particles searching for a new target (the birth particles) and tracking the target (the continuing particles). In the standard SIR algorithm the ratio of the number of birth and continuing particles changes significantly depending on the presence of a target in the data.




(a) 12dB target.

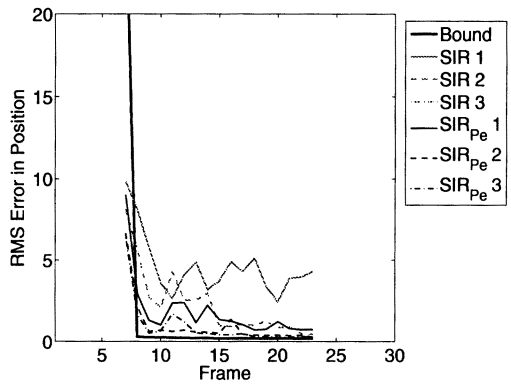



(b) 6dB target.

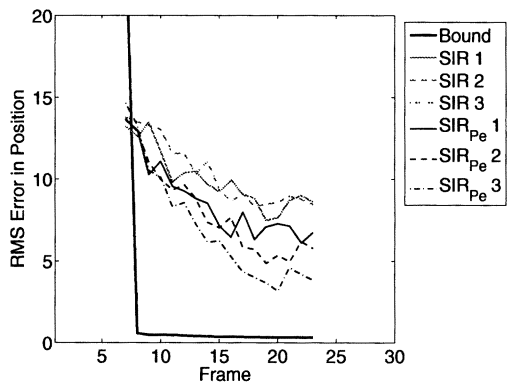



(c) 3dB target.

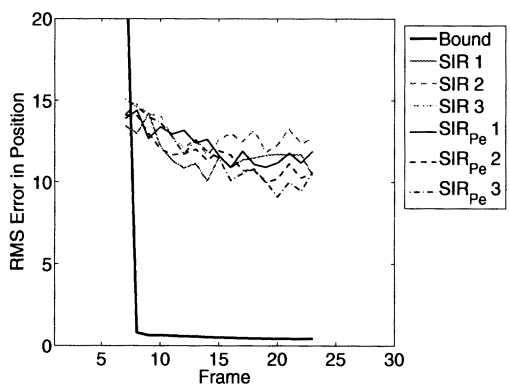
Fig. 3. Probability of existence.




(a) 12dB target.




(b) 6dB target.




(c) 3dB target.

Fig. 4. RMS error in position.

## REFERENCES


[1] D. J. Salmond and H. Birch, "A particle filter for track-before-detect," in *Proc. American Control Conf.*, Arlington, VA, USA, June 2001, pp. 3755–3760.

[2] Y. Boers and J. N. Driessen, "Particle filter based detection for tracking," in *Proceedings of the American Control Conference*, Arlington, VA, USA, June 2001, pp. 4393–4397.

[3] B. Ristic, S. Arulampalam, and N. J. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, 2004.

[4] M. G. Rutten, N. J. Gordon, and S. Maskell, "Efficient particle-based track-before-detect in Rayleigh noise," in *Fusion 2004: Proc. 7th Int. Conf. on Information Fusion*, Stockholm, Sweden, June 2004.

[5] S. B. Colegrove, A. W. Davis, and J. K. Ayliffe, "Track initiation and nearest neighbours incorporated into probabilistic data association," *Journal of Electrical and Electronics Engineers, Australia*, vol. 6, no. 3, pp. 191–198, Sept. 1986.

[6] D. Mušicki, R. Evans, and S. Stankovic, "Integrated probabilistic data association," *IEEE Trans. Automat. Contr.*, vol. 39, no. 6, pp. 1237–1240, June 1994.

[7] H. Driessen and Y. Boers, "An efficient particle filter for nonlinear jump Markov systems," in *Proceedings of the IEE Seminar on Target Tracking: Algorithms and Applications*, Sussex, UK, Mar. 2004.

[8] P. Tichavský, C. H. Muravchik, and A. Nehorai, "Posterior Cramér-Rao bounds for discrete-time nonlinear filtering," *IEEE Trans. Signal Processing*, vol. 46, no. 5, pp. 1386–1396, May 1998.