Multitarget particle filter track before detect application

Y. Boers and J.N. Driessen

Abstract: The paper deals with a radar 'track before detect' application in a multitarget setting. 'Track before detect' is a method to track weak objects on the basis of raw radar measurements, e.g. the reflected power of the target plus noise. In classical target tracking, the tracking process is performed on the basis of pre-processed measurements that are constructed from the original measurement data every time step. In this way no integration over time takes place and information is lost. The authors give details of a modelling setup and a particle filter based algorithm to deal with a 'multiple target track before detect' situation. Using simulations it is shown that with this method, it is possible to track multiple, closely spaced, weak targets.

1 Introduction

Classical radar tracking methods take as input thresholded measurements, so called plots, that typically consist of range measurements, bearing measurements, elevation measurements and range rate (Doppler) measurements, [1, 2]. In this classical tracking setting tracking consists of estimating kinematic state properties, e.g. position, velocity and acceleration, on the basis of these measurements.

In the classical setup the measurements are the output of the extraction (Fig. 1). In this setup there is a processing chain before the tracking, this processing chain can consist, e.g. in the case of radar, of a detection stage, a clustering stage and an extraction stage (Fig. 1).

In the method considered here, we use as measurements the raw measurement data, e.g. reflected power (see Fig. 1). We consider a recursive Bayesian version of 'track before detect' (TBD) [3–6]. Non-recursive versions of TBD have been around longer and come in all kinds of varieties, see chapter 17 of [1]. For a good and recent overview of TBD, see chapter 11 of [7].

If we look at Fig. 1 we see that in classical tracking (i.e. the separate blocks) threshold based decisions are already made directly on the basis of the raw measurement from one single scan. This means that already at the beginning of the processing chain a hard decision is made with respect to the relevance of the information. Note that this decision is made instantaneously, i.e without using information from the near past.

In TBD the decision is made at the end of the processing chain, i.e. when all information has been used and integrated over time. Note that information in a track is obtained by integrating over time. This method is especially suitable for tracking weak targets, i.e. targets that in the classical setting often will not lead to a detection. The gain that can be obtained by integrating the 'raw' measurements versus measurements after thresholding, i.e. plots, has been investigated for a typical search radar setup in [8] for the single target case, see also chapter 17 of [1].

Furthermore, the recursive TBD setup avoids the classical data association problem, i.e. the problem of measurement to track association. The reason for this is simple and elegant, as no intermediate thresholding takes place, there is no need for an explicit mechanism to attach these thresholded measurements to a track, i.e. no explicit data association algorithm is needed.

In this paper a particle filter is used to perform the recursive TBD. In [9] it has been shown that, at least for a single target, optimal detection can be based on the output of this filter.

Related to the work here, is the work in [10], where a multiple target approach for TBD on the basis of camera observations for well separated targets has been presented. Also in [11] a track-before-detect-like approach is proposed; this work is a first attempt to describe a general multitarget 'track before detect' particle filter setup. The authors, however, remain vague about certain modelling issues, i.e. the birth and death of targets, and about the implementation.

In this paper we present a recursive Bayesian TBD algorithm that can deal with the situation of a limited number of closely spaced targets. Furthermore, targets are assumed to be able to pop up and disappear, according to a birth–death process that is modelled as a jump Markov process. Thus during the scenario the number of targets may vary over time. It is assumed, however, that the maximum number of possible targets is known.

In this paper we restrict ourselves to the case of two closely spaced targets. This case is very important from a practical point of view, because it covers the situation of a platform launching a small secondary target, e.g. a fighter, helicopter, or ship firing a missile. Obviously, it is crucial to track and detect both the platform and the small secondary target from an early stage.

Furthermore, we restrict ourselves to the case where target strengths are known; this assumption could be relaxed. In [4] a single target recursive TBD algorithm is presented, where the target strength is unknown and is estimated along with the other parameters.

IEE Proceedings online no. 20040841

doi: 10.1049/ip-rsn:20040841

Paper first received 2nd February and in revised form 15th June 2004. Originally published online 2nd November 2004

The authors are with Thales Nederland, Zuidelijke Havenweg 40, 7554 RR Hengelo, The Netherlands

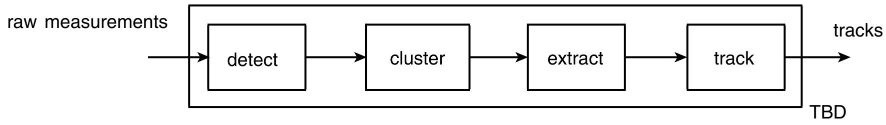

Fig. 1 *Classical data and signal processing (separate boxes) and TBD (large box)*

2 System setup

Consider the general nonlinear dynamic system:

$$\boldsymbol{s}_{k+1} = f(t_k, \boldsymbol{s}_k, m_k, \boldsymbol{w}_k), \;\; k \in \mathbb{N} \tag{1}$$

$$\mathrm{Prob}\{m_{k+1} = i \mid m_k = j\} = [\boldsymbol{\Pi}(t_k)]_{ij} \tag{2}$$

$$z_k = h(t_k, \boldsymbol{s}_k, m_k, \boldsymbol{v}_k), \;\; k \in \mathbb{N} \tag{3}$$

where

- $\boldsymbol{s}_k \in \mathcal{S} \subset \mathbb{R}^n$ is the base state of the system
- $m_k \in \mathcal{M} \subset \mathbb{N}$ is the modal state of the system
- $z_k \in \mathbb{R}^p$ is the measurement
- $t_k \in \mathbb{R}$ is time
- $\boldsymbol{w}_k$ is the process noise and $p_{\boldsymbol{w}}(\boldsymbol{w})$ is the probability distribution of the process noise
- $\boldsymbol{v}_k$ is the measurement noise and $p_{\boldsymbol{v}}(\boldsymbol{v})$ is the probability distribution of the measurement noise
- f is the system dynamics function
- h is the measurement function
- $\boldsymbol{\Pi}(t_k)$ is the Markov transition matrix.

The system defined by (1), (2) and (3) is also often referred to as a jump Markov system. Jump Markov systems are often used in target tracking applications. Their key feature is that they involve both a continuous valued, kinematic target state and a discrete mode or modal state, whose dynamics are governed by a Markov process (see also chapter 1 of [7]).

Define

$$\boldsymbol{Z}_k = \{z_1, \ldots, z_k\} \tag{4}$$

The optimal hybrid filtering problem [12] can be formulated as follows.

Problem 1 (Optimal filtering problem): Consider the system represented by the equations (1), (2) and (3). Assume that the initial pdf $p(\boldsymbol{s}_0, m_0)$ is available. The hybrid filtering problem is the problem of constructing the *a posteriori* probability density function

$$p(\boldsymbol{s}_k, m_k \mid \boldsymbol{Z}_k) \tag{5}$$

Note that given the solution to problem 1, the mean of the state $(\boldsymbol{s}_k, m_k)$ is obtained as

$$E_{p(\boldsymbol{s}_k, m_k \mid \boldsymbol{Z}_k)}(\boldsymbol{s}_k, m_k) \tag{6}$$

For any function of the state $\phi(\boldsymbol{s}_k, m_k)$, the mean

$$E_{p(\boldsymbol{s}_k, m_k \mid \boldsymbol{Z}_k)}\phi(\boldsymbol{s}_k, m_k) \tag{7}$$

can be calculated on the basis of the filtering solution.

3 Particle filter solution

The following algorithm results in an approximation of the *a posteriori* filtering distribution [12, 13]

$$p(\boldsymbol{s}_k, m_k \mid Z_k)$$

Algorithm 1: Consider the system represented by the equations (1), (2) and (3). Assume that an initial pdf

$$p(\boldsymbol{s}_0, m_0)$$

is given. Choose an integer N, the sample size.

Step 1: Draw N samples according to $p(\boldsymbol{s}_0, m_0)$, to obtain $\{(\tilde{\boldsymbol{s}}_0^i, \tilde{m}_0^i)\}_{i=1,\ldots,N}$.
Step 2: Generate $\{m_k^i\}_{i=1,\ldots,N}$ on the basis of $\{\tilde{m}_{k-1}^i\}_{i=1,\ldots,N}$ and $\boldsymbol{\Pi}(t_{k-1})$.
Step 3: Draw $\{\boldsymbol{w}_{k-1}^i\}_{i=1,\ldots,N}$ according to $p_{\boldsymbol{w}}(\boldsymbol{w})$ and obtain $\{(\boldsymbol{s}_k^i)\}_{i=1,\ldots,N}$ using

$$\boldsymbol{s}_k^i = f(t_{k-1}, \tilde{\boldsymbol{s}}_{k-1}^i, \tilde{m}_{k-1}^i, \boldsymbol{w}_{k-1}^i)$$

Step 4: Given z_k, define

$$\tilde{q}_k^i = p(z_k \mid s_k^i, m_k^i, t_k), \;\; i = 1, \ldots, N$$

Step 5: Normalise

$$q_k^i := \frac{\tilde{q}_k^i}{\sum_{i=1}^N \tilde{q}_k^i}, \;\; i = 1, \ldots, N$$

Step 6: Resample N times from

$$\hat{p}(\boldsymbol{s}, m) := \sum_{j=1}^N q_k^i \delta((\boldsymbol{s}, m) - (\boldsymbol{s}_k^i, m_k^i))$$

and obtain $\{(\tilde{\boldsymbol{s}}_k^i, \tilde{m}_k^i)\}_{i=1,\ldots,N}$ to construct

$$\hat{p}(\boldsymbol{s}_k, m_k \mid Z_k) := \sum_{j=1}^N \frac{1}{N} \delta((\boldsymbol{s}, m) - (\tilde{\boldsymbol{s}}_k^i, \tilde{m}_k^i))$$

goto 2.

The above algorithm is a simple ‘standard’ particle filter implementation. It consists essentially of a prediction step, an updating step and a resampling step; this last step is crucial for the successful application of a particle filter [14]

Different and more efficient algorithms for (multiple model) particle filters exist, see e.g. [13].

It is known that the above algorithm guarantees convergence, in a certain sense, of the empirical distribution

$$\hat{p}(\boldsymbol{s}_k, m_k \mid Z_k) := \sum_{j=1}^N \frac{1}{N} \delta((\boldsymbol{s}, m) - (\tilde{\boldsymbol{s}}_k^i, \tilde{m}_k^i))$$

to the true, but unknown, *a posteriori* distribution of interest, i.e. $p(\boldsymbol{s}_k, m_k \mid Z_k)$, if the number of particles N tends to infinity. See [15], for more detail and a good overview of convergence results.

4 TBD system setup

In this Section we describe the models that will be used in the TBD application, namely the system dynamics model and the measurement model.

4.1 System dynamics

A widely used model is the constant velocity model [1, 2]. This model is used to describe the position and velocity using Cartesian coordinates. Furthermore the model has an additive process noise term. The discrete-time system dynamics of this model is of the form

$$s_{k+1} = f(t_k, s_k, m_k) + g(t_k, s_k, m_k) w_k \tag{8}$$

where

$$f(t_k, s_k, m_k) = \begin{pmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} s_k \tag{9}$$

with the state vector $s_k = [x_k, y_k, \dot{x}_k, \dot{y}_k]^T$ where x_k and y_k are the positions and $\dot{x}_k$ and $\dot{y}_k$ are the velocities. The process noise w_k is assumed to be standard white Gaussian noise. T is the revisit time, which has been assumed to be constant here; this assumption can be relaxed, however.

The process noise input model is given by

$$g(t_k, s_k, m_k) = \begin{pmatrix} \frac{1}{2}(\frac{1}{3} a_{x,max})T^2 & 0 \\ 0 & \frac{1}{2}(\frac{1}{3} a_{y,max})T^2 \\ \frac{1}{3} a_{x,max} T & 0 \\ 0 & \frac{1}{3} a_{y,max} T \end{pmatrix} \tag{10}$$

with maximum accelerations $a_{x,max}$ and $a_{y,max}$.

4.2 Measurement model

The measurements are measurements of reflected power. One measurement z_k consists of $N_r \times N_d \times N_b$ power measurements z_k^{ijl}, where N_r, N_d and N_b are the number of range, Doppler and bearing cells.

Figure 2 shows power measurements for a fixed bearing angle, but as a function of different range and Doppler. The power measurements in this figure correspond to a target that has a SNR of 13 dB.

The power measurements per range–Doppler–bearing cell are defined by

$$z_k^{ijl} = |z_{A,k}^{ijl}|^2 \quad k \in \mathbb{N} \tag{11}$$

where $z_{A,k}^{ijl}$ represents the complex amplitude data of the target, which is

$$z_{A,k} = A_k h_A(s_k, t_k) + n(t_k), \quad k \in \mathbb{N} \tag{12}$$

where

$$A_k = \tilde{A}_k e^{\imath \phi_k}, \qquad \phi_k \in (0, 2\pi) \tag{13}$$

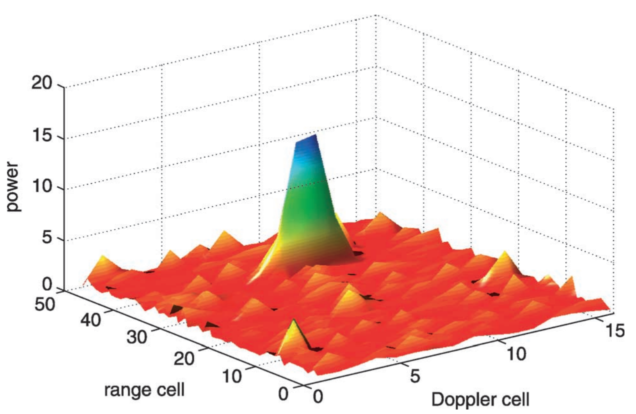

Fig. 2 *Power of a target in noise (SNR 13 dB) as a function of range and Doppler cells*

is the complex amplitude of the target and $h_A(s_k, t_k)$ is the reflection form that is defined for every range–Doppler–bearing cell by

$$h_A^{ijl}(s_k, t_k) = \exp\left\{ -\frac{(r_i - r_k)^2}{2R} L_r - \frac{(d_j - d_k)^2}{2D} L_d - \frac{(b_l - b_k)^2}{2B} L_b \right\} \tag{14}$$

$i = 1, \ldots, N_r,\ j = 1, \ldots, N_d,\ l = 1, \ldots, N_b$ and $k \in \mathbb{N}$

with

$$r_k = \sqrt{x_k^2 + y_k^2} \tag{15}$$

$$d_k = \dot{r}_k = \frac{1}{\sqrt{x_k^2 + y_k^2}} (x_k \dot{x}_k + y_k \dot{y}_k) \tag{16}$$

$$b_k = \arctan\left(\frac{y_k}{x_k}\right) \tag{17}$$

which are the range, Doppler and bearing respectively of the target. R, D and B are constants related to the size of a range, Doppler and bearing cell. L_r, L_d and L_b represent constants of losses.

The noise is defined by

$$n(t_k) = n_I(t_k) + \imath n_Q(t_k) \tag{18}$$

which is complex Gaussian, where $n_I(t_k)$ and $n_Q(t_k)$ are independent, zero-mean white Gaussian with variance σ_n^2. In this way the power measurements in a range–Doppler–bearing cell are defined by

$$z_k^{ijl} = |z_{A,k}^{ijl}|^2 \tag{19}$$

$$= |A_k h_A^{ijl}(s_k, t_k) + n_I(t_k) + \imath n_Q(t_k)|^2, \quad k \in \mathbb{N}$$

These measurements, conditioned on the state, s_k, are now assumed to be exponentially distributed, [16].

$$p^{(1)}(z_k^{ijl} | s_k) = \frac{1}{\mu_0^{ijl}} \exp\left\{ -\frac{1}{\mu_0^{ijl}} z_k^{ijl} \right\} \tag{20}$$

where

$$\begin{aligned} \mu_0^{ijl} &= E_{n_I, n_Q}[z_k^{ijl}] \\ &= E_{n_I, n_Q}[|\tilde{A}_k e^{i\phi_k} h_A^{ijl}(s_k, t_k) + n_I(t_k) + \imath n_Q(t_k)|^2] \\ &= E_{n_I, n_Q}[(\tilde{A}_k h_A^{ijl}(s_k, t_k) \cos(\phi_k) + n_I(t_k))^2 \\ &\quad + (\tilde{A} h_A^{ijl}(s_k, t_k) \sin(\phi_k) + n_Q(t_k))^2] \\ &= \tilde{A}^2 (h_A^{ijl}(s_k, t_k))^2 + 2\sigma_n^2 \\ &= P h_P^{ijl}(s_k, t_k) + 2\sigma_n^2 \end{aligned} \tag{21}$$

with

$$h_P^{ijl}(s_k, t_k) = (h_A^{ijl}(s_k, t_k))^2$$

$$= \exp\left\{ -\frac{(r_i - r_k)^2}{R} L_r - \frac{(d_j - d_k)^2}{D} L_d - \frac{(b_l - b_k)^2}{B} L_b \right\} \tag{22}$$

which describes the power contribution of a target in every range–Doppler–bearing cell.

5 Multiple target setting

In this Section we consider the (possible) presence of two targets, where one target can originate (spawn) from the other; e.g. a missile being fired from a fighter airplane.

We consider the general system introduced in Section 2, i.e.

$$\boldsymbol{s}_{k+1} = f(t_k, \boldsymbol{s}_k, m_k, \boldsymbol{w}_k), \quad k \in \mathbb{N} \tag{23}$$

$$\text{Prob}\{m_{k+1} = i \mid m_k = j\} = [\boldsymbol{\Pi}(t_k)]_{ij} \tag{24}$$

$$\boldsymbol{z}_k = h(t_k, \boldsymbol{s}_k, m_k, \boldsymbol{v}_k), \quad k \in \mathbb{N} \tag{25}$$

Furthermore, for this problem it is convenient to define a base state vector that consists of the base states of both targets. Thus,

$$\boldsymbol{S}_k = \begin{pmatrix} \boldsymbol{s}_k^{(1)} \\ \boldsymbol{s}_k^{(2)} \end{pmatrix} \tag{26}$$

The discrete mode m_k represents one of three hypotheses

- $m_k = 0$: there is no target present.
- $m_k = 1$: there is one target present.
- $m_k = 2$: there are two targets present.

5.1 Measurements models

For the multitarget setup the measurement models need to be extended.

The complex amplitude data that are received from two targets can be modelled by

$$z_{A,k} = A_k^{(1)} h_A^{(1)}(\boldsymbol{s}_k, t_k) + A_k^{(2)} h_A^{(2)}(\boldsymbol{s}_k, t_k) + n(t_k) \tag{27}$$

where $A_k^{(1)}$, $h_A^{(1)}$ and $A_k^{(2)}$, $h_A^{(2)}$ are the amplitude and reflection form of the first and second target respectively. Thus, the power measurements are

$$\begin{aligned} z_k^{ijl} &= |z_{A,k}^{ijl}|^2 \\ &= |A^{(1)} h_A^{(1)ijl}(\boldsymbol{s}_k, t_k) + A^{(2)} h_A^{(2)ijl}(\boldsymbol{s}_k, t_k) \\ &\quad + n_I(t_k) + \imath n_Q(t_k)|^2 \end{aligned} \tag{28}$$

These will, again be exponentially distributed

$$p^{(2)}(z_k^{ijl}|\boldsymbol{s}_k) = \frac{1}{\mu_0^{ijl}} \exp\left\{-\frac{1}{\mu_0^{ijl}} z_k^{ijl}\right\} \tag{29}$$

where

$$\begin{aligned} \mu_0^{ijl} &= E_{n_I,n_Q}[z_k^{ijl}] \\ &= P^{(1)} h_P^{(1)ijl}(\boldsymbol{s}_k, t_k) \\ &\quad + P^{(2)} h_P^{(2)ijl}(\boldsymbol{s}_k, t_k) + 2\sigma_n^2 \end{aligned} \tag{30}$$

using this and the assumptions that the noise is independent from cell to cell and that the reflections of the two targets are independent, we obtain for the mode-dependent likelihood

$$p(z_k|\boldsymbol{S}_k, m_k) = \begin{cases} \prod_{ijl} p_v(z_k^{ijl}) & \text{for } m_k = 0 \\ \prod_{ijl} p^{(1)}(z_k^{ijl}|\boldsymbol{S}_k) & \text{for } m_k = 1,\ \boldsymbol{s}_k^{(2)} = \boldsymbol{s}_k^{(1)} \\ \prod_{ijl} p^{(2)}(z_k^{ijl}|\boldsymbol{S}_k) & \text{for } m_k = 2 \end{cases} \tag{31}$$

6 Example

In this Section we give a demonstration of a particle filter TBD algorithm that is capable of tracking two closely spaced targets.

In the scenario, initially there is no target present, the stronger primary target appears after 5 s at a position of 88.6 km from the sensor and flies at a constant velocity of $200\,\text{m}\,\text{s}^{-1}$ directly to the sensor. At $t = 20$ a weaker secondary target spawns from the first, accelerating to a velocity of $300\,\text{m}\,\text{s}^{-1}$ over three scans.

We consider range cells in the interval $[80, 90]$ km, Doppler cells in the interval $[-0.35, -0.10]\,\text{km}\,\text{s}^{-1}$. We consider only one bearing cell in this example. We therefore have $N_r \times N_d \times N_b$ cells, where $N_r = 50$, $N_d = 16$ and $N_b = 1$.

Initially, 1000 particles are uniformly distributed in the state space, in an area between $[85, 90]$ km and $[-0.23, -0.10]\,\text{km}\,\text{s}^{-1}$ in the x-direction and $[-0.1, 0.1]$ km $[-0.10, 0.10]\,\text{km}\,\text{s}^{-1}$ and in the y-direction. Furthermore, initially we uniformly distribute the particles over all modes.

The transition probability matrix is assumed to be

$$\boldsymbol{\Pi}(t_k) = \begin{pmatrix} 0.90 & 0.10 & 0.00 \\ 0.10 & 0.80 & 0.10 \\ 0.00 & 10.10 & 0.90 \end{pmatrix}$$

This choice of a transition matrix implies that we assume that there is no direct transition from the situation of zero targets present to two targets present and *vice versa*. We emphasise that this assumption is not crucial for the correct working of the algorithm.

The update time, T, is set to 1 s.

The dynamics of both targets are captured by a constant velocity model, but with different values for their maximum acceleration. For the first target we set $a_{max,x} = a_{max,y} = 5\,\text{m}\,\text{s}^{-2}$. For the second target we choose $a_{max,x} = 35\,\text{m}\,\text{s}^{-2}$ and $a_{max,y} = 5\,\text{m}\,\text{s}^{-2}$.

We assume that the power of both targets is known, $P^{(1)} = 10$ and $P^{(2)} = 1$. The SNR for the first target is defined by

$$\text{SNR} = 10 \log\left(\frac{P^{(1)}}{2\sigma_n^2}\right) [\text{dB}]$$

Thus we assume a stronger primary target and a weaker, but more agile secondary target. This particular setup would correspond to a moving launch platform, e.g. a fighter aircraft that could possibly fire an agile weak secondary target, e.g. a missile.

Simulations are performed, where the SNR of the strongest target is 13 dB, this implies that the SNR of

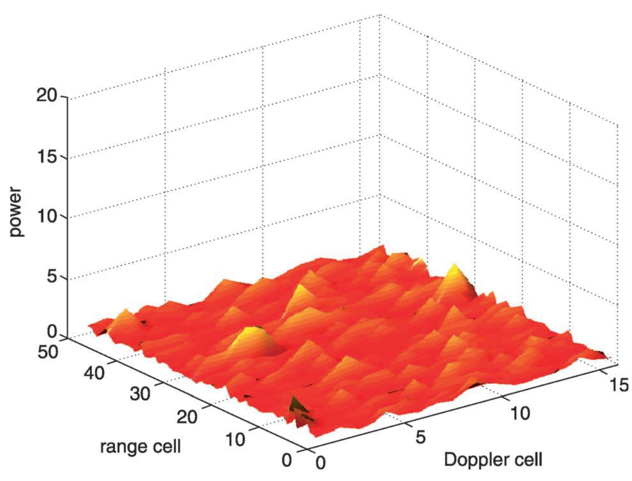

Fig. 3 *Power measurements as a function of range and Doppler cells at time step 2 (noise only)*

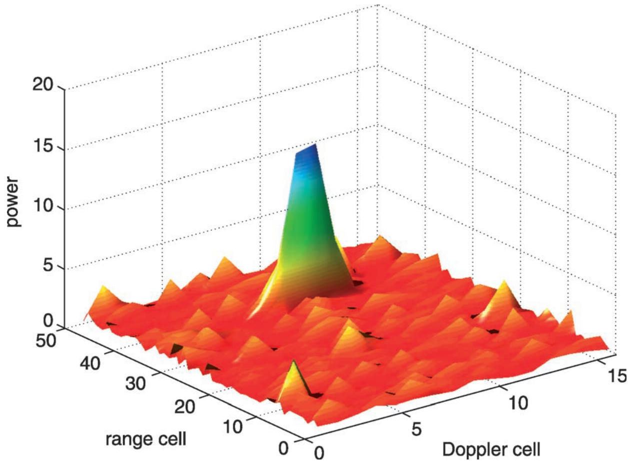

Fig. 4 *Power measurements as function of range and Doppler cells at time step 15 (single 13 dB target in noise)*

the secondary target is 3 dB, i.e. a fairly weak secondary target. In the Figs. 3–8, we see some examples of the power measurements received at several different time steps. Figures 6 and 8 are almost identical to Figs. 5 and 7, the only difference is that we have 'artificially' increased the power

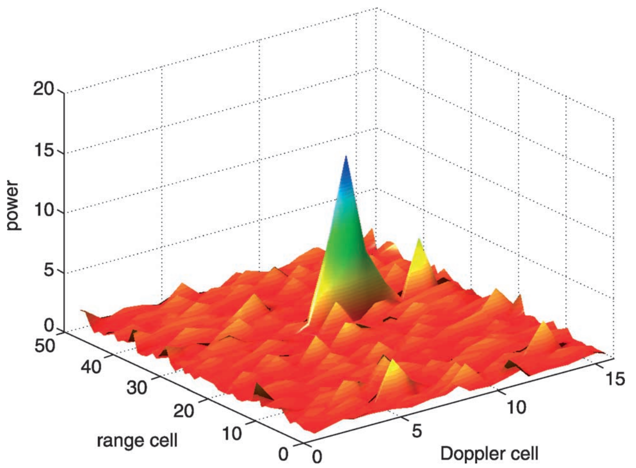

Fig. 5 *Power measurements as a function of range and Doppler cells at time step 22 (primary 13 dB target and secondary 3 dB target in noise)*

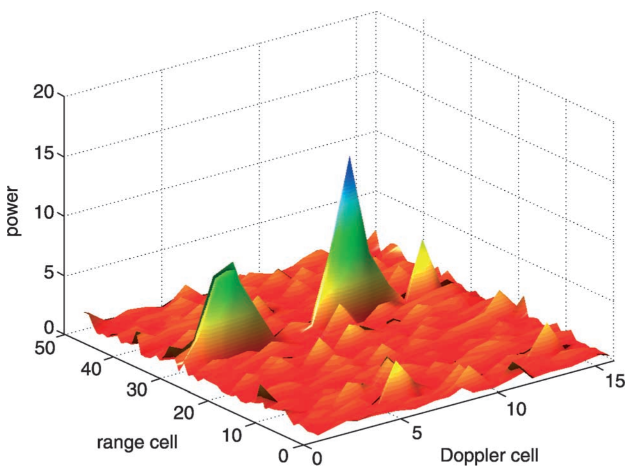

Fig. 6 *Power measurements as a function of range and Doppler cells at time step 22 (primary 13 dB target and secondary 13 dB target in noise)*

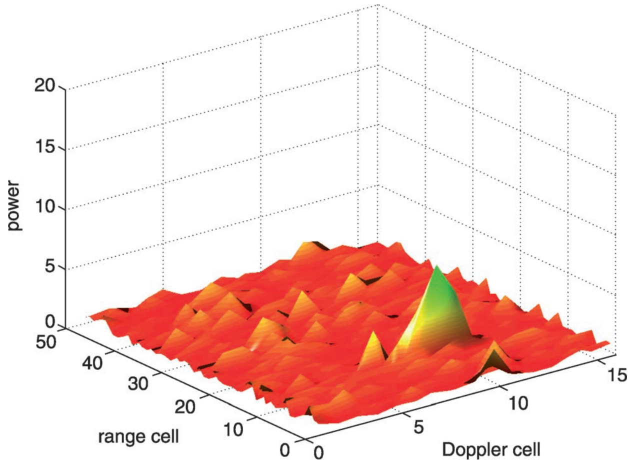

Fig. 7 *Power measurements as a function of range and Doppler cells at time step 40 (primary 13 dB target and secondary 3 dB target in noise)*

of the secondary target, for display purposes only. Thus, the measurement data, shown in Figs. 5 and 7, are data that are provided to the TBD algorithm.

Some of the filtering results are shown through the particle clouds in Figs. 9–16. The darker particles

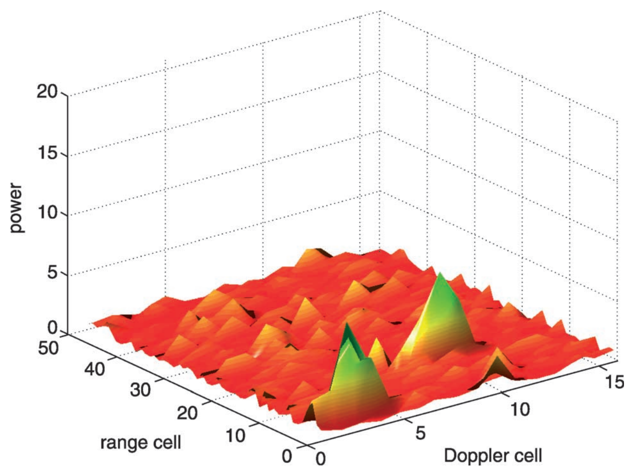

Fig. 8 *Power measurements as a function of range and Doppler cells at time step 40 (primary 13 dB target and secondary 13 dB target in noise)*

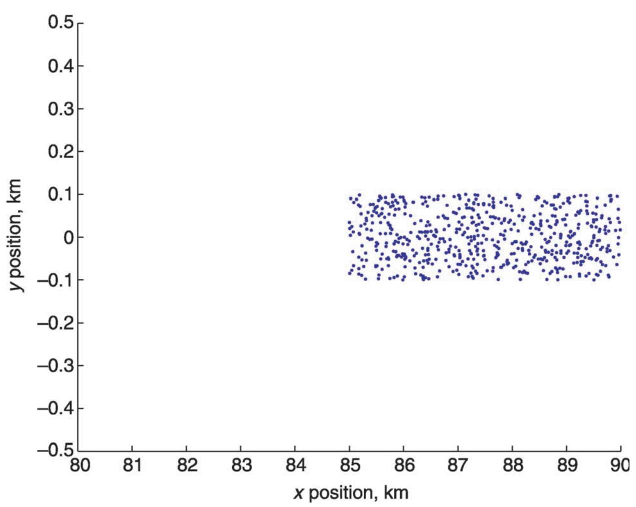

Fig. 9 *Target position, particles at time step 5*

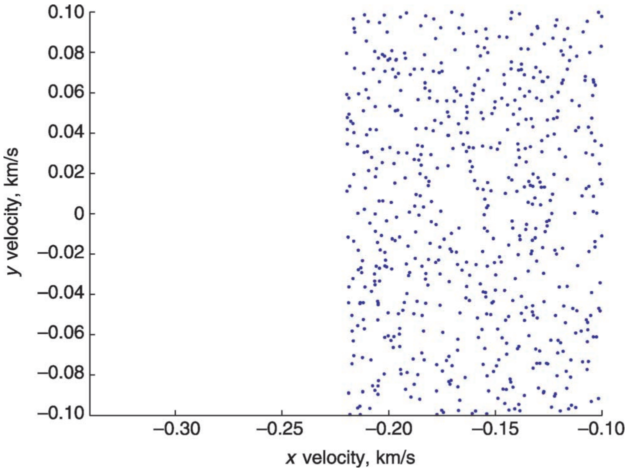

Fig. 10 *Target velocity, particles at time step 5*

correspond to the primary target and the lighter ones to the secondary target. The true target positions and velocities are marked by a star, if the target is present.

Also the true modes and filtered modes are shown in Fig. 17, i.e. these give information on the presence of target(s) and the evidence that the filter has of their presence.

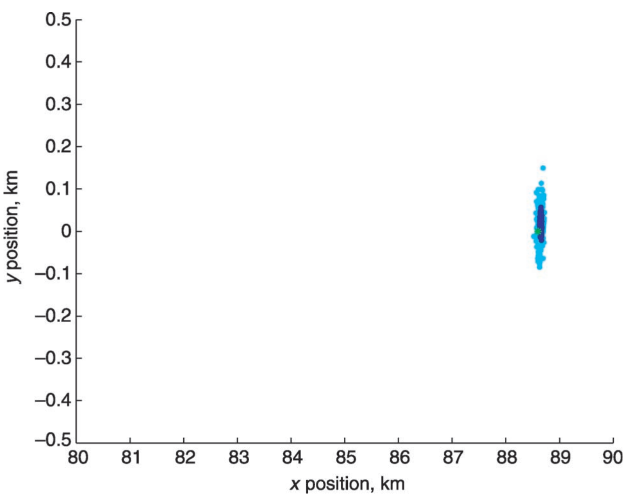

Fig. 11 *Target position, particles at time step 10*

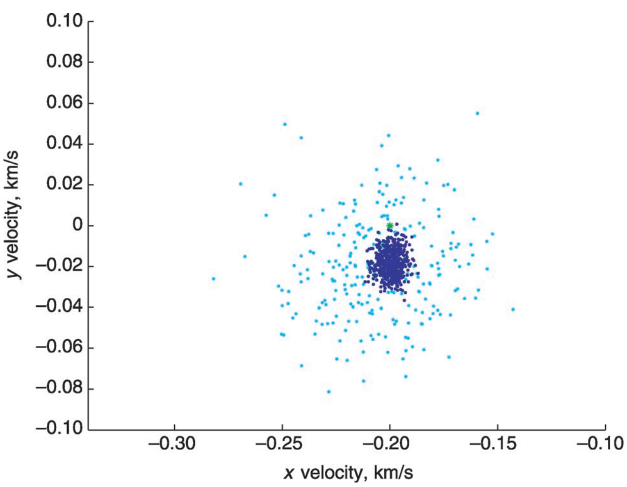

Fig. 12 *Target velocity, particles at time step 10*

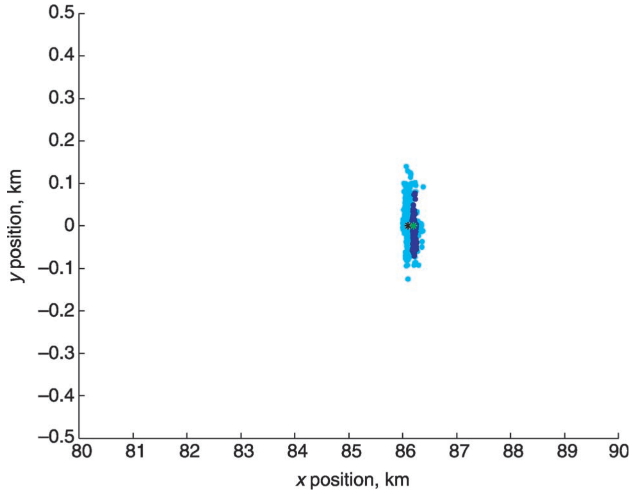

Fig. 13 *Target position, particles at time step 22*

The particle cloud in Figs. 9 and 10 is still more or less uniformly distributed over the area of possible target positions and velocities, reflecting the fact that no target is present at this moment; see also Fig. 17.

The same clouds at time step 10 (Figs. 11 and 12) are concentrated on the target location and velocity. Looking

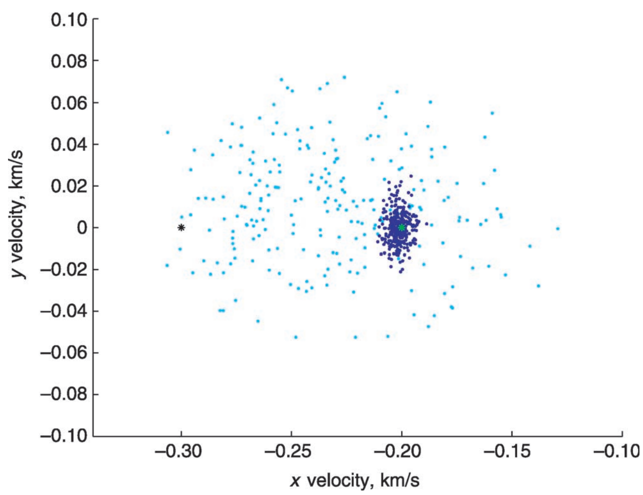

Fig. 14 *Target velocity, particles at time step 22*

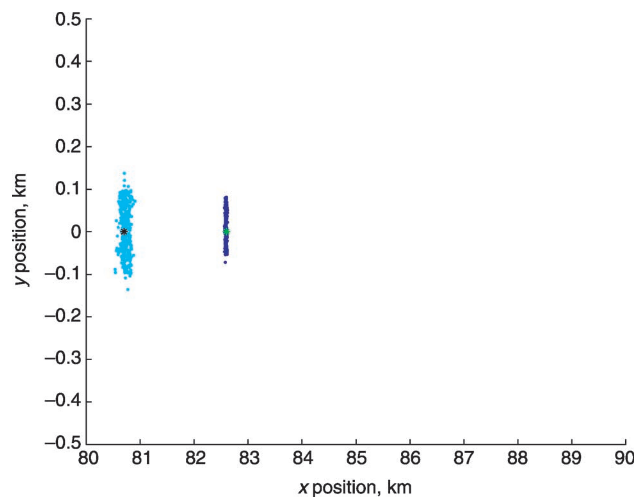

Fig. 15 *Target position, particles at time step 40*

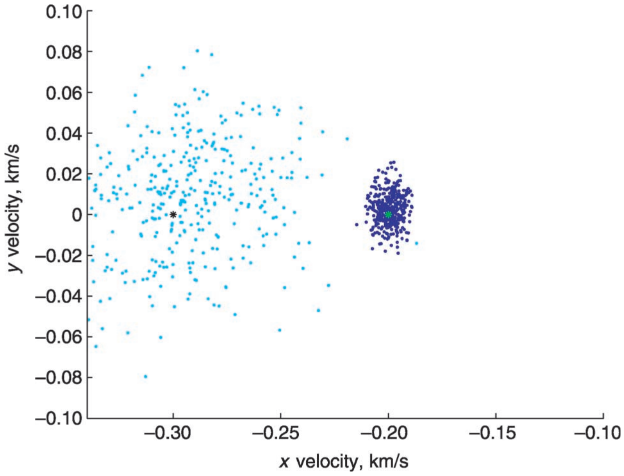

Fig. 16 *Target velocity, particles at time step 40*

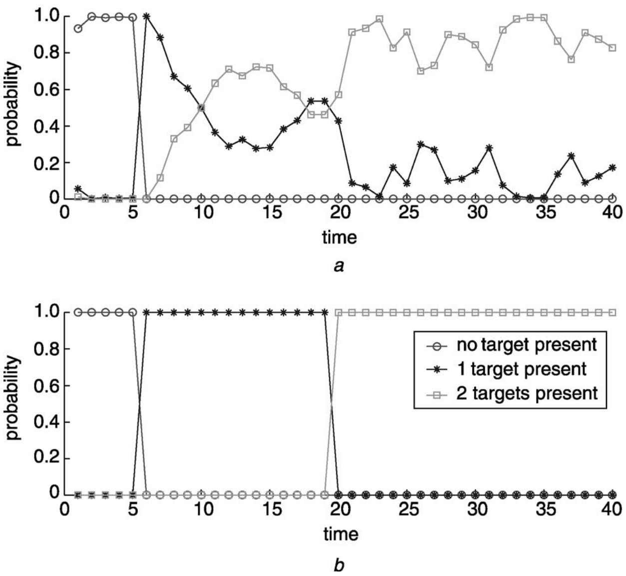

Fig. 17 *Mode probabilities*

a Estimated mode
b True mode

at Fig. 17, for time step 10, we see that the filter is certain about the presence of a target, but it is uncertain whether there is a single target or whether there a two targets, the probability for either two of the options is about a half, this is also seen from the fact that the particle clouds in Figs. 11 and 12 are a mixture of light and dark. The explanation for this is that the filter is unable to rigorously conclude whether the data have been generated by one strong target moving at a certain speed, or by the superposition of one strong and one weak target, moving exactly one on top of the another.

As soon as the two targets separate in position and speed, caused by the secondary target accelerating away from the primary one (see Figs. 13–16 for the respective particle clouds), the filter is fairly confident that the secondary target has split from the primary one, see also Fig. 17.

Furthermore, we stress the fact that only 1000 particles have been used and that a plain vanilla particle filter implementation has been used. Even with these ‘limitations’, the algorithm runs much faster than real time on an 800 MHz machine under MATLAB™.

Also in order to add a true detection stage to the algorithm, the filter output could be used the to construct different types of detectors, e.g. a maximum *a posteriori* detector [17], or a likelihood ratio based detector, [9, 17].

7 Conclusions

We have presented a modelling setup and an algorithm for a multiple target recursive Bayesian TBD application in a radar context. The problem has been solved by using a multiple model particle filter. Simulations show that the algorithm works well and that the method can deal with fairly weak targets that are closely spaced.

It has been shown that with a fairly straightforward particle filter implementation a good and relatively fast algorithm can be constructed and it has also been pointed out how the filtering output could be used in a detection stage.

8 Acknowledgments

A preliminary version of this paper has been presented at the Workshop on Multi Object Tracking (WOMOT) during the 2003 IEEE Conference on Vision and Pattern Recognition (CVPR) [18].

9 References

1 Blackman, S., and Popoli, R.: ‘Design and analysis of modern tracking systems’ (Artech House, Norwood, MA, USA, 1999)
2 Bar-Shalom, Y., and Li, X.R.: ‘Estimation and tracking principles, techniques and software’ (Artech House, Norwood, MA, USA, 1993)
3 Salmond, D.J., and Birch, H.: ‘A particle filter for track-before-detect’. Proc. American Control Conf., Arlington, VA, USA, 25–27 June 2001
4 Rollason, M., and Salmond, D.: ‘A particle filter for track-before-detect of a target with unknown amplitude’. Proc. IEE Colloquium on Target Tracking: Algorithms and Applications, Enschede, The Netherlands, Oct. 2001
5 Boers, Y., and Driessen, J.N.: ‘Particle filter based detection for tracking’. Proc. American Control Conf., Arlington, VA, USA, 25–27 June 2001
6 Boers, Y., Driessen, J.N., and Grimmerink, K.: ‘Particle filter based detection schemes’. *Proc. SPIE–Int. Soc. Opt. Eng.*, 2002, **4728**, pp. 128–137
7 Ristic, B., Arulampalam, S., and Gordon, N.: ‘Beyond the Kalman filter-particle filters for tracking applications’ (Artech House, Boston, MA, USA, 2004)
8 Zwaga, J.H., Driessen, J.N., and Meijer, W.J.H.: ‘Track-before-detect for surveillance radar: a recursive filter based approach’, *Proc. SPIE–Int. Soc. Opt. Eng.*, 2002, **4728**, pp. 103–115
9 Boers, Y., and Driessen, J.N.: ‘A particle filter based detection scheme’, *IEEE Signal Process. Lett.*, 2003, **10**, (10), pp. 300–302
10 Ballantyne, D.J., Chan, H.Y., and Kouritzin, M.A.: ‘A branching particle-based nonlinear filter for multi-target tracking’. Proc. FUSION Conf., Montreal, Canada, 7–10 August 2001
11 Kreucher, C., Kastella, K., and Hero, A.O., III: ‘Tracking multiple targets using a particle filter representation of the joint multitarget probabiltiy density’, *Proc. SPIE–Int. Soc. Opt. Eng.*, 2003, **5204**, pp. 258–269
12 Boers, Y., and Driessen, J.N.: ‘Hybrid state estimation: a target tracking application’, *Automatica*, 2002, **38**, (12), pp. 2153–2158
13 Doucet, A., De Freitas, J.F.G., and Gordon, N.J. (Eds): ‘Sequential Monte Carlo methods in practice’ (Springer-Verlag, New York, NY, USA, 2001)
14 Gordon, N.J., Salmond, D.J., and Smith, A.F.M.: ‘Novel approach to nonlinear/non-Gaussian Bayesian state estimation’, *IEE Proc. F, Radar Signal Process.*, 1993, **1400**, (2), pp. 107–113
15 Crisan, D., and Doucet, A.: ‘A survey of convergence results on particle filtering methods’, *IEEE Trans. Signal Process.*, 2002, **50**, (3), pp. 736–746
16 Skolnik, M.I.: ‘Introduction to radar systems’ (McGraw-Hill, Inc, Singapore, 1981, 2nd edn.)
17 Melsa, J.L., and Cohn, D.L.: ‘Decision and estimation theory’ (McGraw-Hill, New York, NY, USA, 1978)
18 Boers, Y., Driessen, J.N., Verschure, F., Heemels, W.P.M.H., and Juloski, A.: ‘A multi target track before detect application’. IEEE Conf. on Computer Vision and Pattern Recognition - Workshop on Multi Object Tracking, Madison, WI, USA, 16–22 June 2003 (on CDROM)