A New Gaussian Mixture Algorithm for GMTI Tracking Under a Minimum Detectable Velocity Constraint

John M. C. Clark, Panagiotis-Aristidis Kountouriotis, and Richard B. Vinter, Fellow, IEEE

Abstract—This paper introduces a new methodology to account for Doppler blind zone constraints, arising, for example, in ground moving target indicator (GMTI) tracking applications. In such problems, target measurements are suppressed when the range rate (Doppler) of the target drops below a specified threshold in magnitude (the minimum detectable velocity). The proposed method, employing Gaussian mixture approximations to the filtering density, differs from earlier Gaussian mixture approaches in the way missed measurements are modelled. The distinctive feature of the algorithm, as compared with other Gaussian mixture filters, is that it is based on an exact calculation of the filtering density when a measurement is not recorded. Algorithms that result from applying this methodology are simple to implement and computationally undemanding. Simulation results indicate a uniform improvement in estimation accuracy over that of earlier proposed analytic techniques, and a tracking performance comparable to that of state-of-the-art particle filters.

Index Terms—Bayesian methods, blind Doppler, ground moving target indicator (GMTI) radar, minimum detectable velocity, target tracking.

I. INTRODUCTION

I	In this paper we continue our study, early results of which were reported in [5] and [6], into a class of ground moving target indicator (GMTI) tracking problems. Here, the sensor provides noisy measurements of target range, bearing and range rate.

A distinctive feature of GMTI trackers, as commonly implemented, is the introduction of a sensor data pre-processing stage, in which measurements are deliberately suppressed, whenever the magnitude of the range rate drops below a specified threshold (the Minimum Detectable Velocity mDV). The purpose of artificially introducing the ‘Doppler blind zone’ (the region of the state space in which the range rate magnitude is small) is to separate out moving objects of interest from heavy, static clutter.

For such a set-up, the occurrence, or non-occurrence, of a measurement in itself provides information about target motion. A key question in GMTI tracker design is how to exploit this information.

In this paper we introduce a new Gaussian mixture filter for GMTI tracking that takes account of the Doppler blind zone in a particularly effective way, and give full details of the analysis underlying its construction. (We also allow for a non-unit probability of detection, unrelated to the location of the measurement relative to the Doppler blind zone).

The proposed filter, which we refer to as the noise related doppler blind mixture filter (NRDB), propagates a Gaussian mixture approximation of the conditional density of the state given measurements up to the present time. The algorithm is based on an exact calculation of the updated density, given a Gaussian mixture prior. The updated density, which is calculated by conditioning on the events that the measured range rate lies in (or fails to lie in) the Doppler blind region, has the form of a weighted sum of densities that are easily calculated. The component densities are then approximated by Gaussian densities with matched first and second moments.

The NRDB filter is constructed according to the same philosophy—performing exact calculations of densities as far as possible before introducing approximations—as the blind doppler mixture filter (BDMF) announced in [5] and elaborated (to take account of multiple models and the presence of clutter) in [6]. But the new filter differs from these predecessors, because it is based on a different model of the mechanism for suppressing measurements in the Doppler blind zone; according to the new model a measurement is returned, depending on whether the observed value of the noise-corrupted range rate is located in the Doppler blind zone, the latter being modeled as a binary region.1 For the measurement model employed in the construction of the algorithm of [5], by contrast, the suppression of a measurement is based on the location of the exact range rate relative to the Doppler blind zone.

We note that the new model is better matched to the practical data gathering process, since the decision to suppress a measurement is made on the basis of the noise-corrupted, not the exact, range rate observations. Surprisingly, even though the new filter is based on a more realistic noise process model, it is both simpler to implement and less computationally demanding than its predecessor [5].

Other filtering schemes based on matching first and second moments, preceeding [5] and [6], have been proposed for GMTI

1There is either none, or there is a complete measurement attenuation, depending on the location of the noisy range rate relative to the Doppler blind zone.
tracking. One example is the algorithm of [14] and [17], in which lost measurements are modelled indirectly, within a multiple model framework to describe the state process. A ‘stopped’ model accounts for the suppression of a measurement when the state enters the Doppler blind zone.

Another Gaussian mixture tracking algorithm, which is closer in spirit to ours because the loss of a measurement is accommodated within the model of the measurement process, is due to Koch et al. [15], [16]. Their algorithm, in common with ours, propagates Gaussian mixture approximations to the conditional density of the target state, but it does so in a different way: account is taken of the Doppler blind zone, by constructing a suitable state-dependent detection probability, which takes low values inside the zone. The tracker takes the form of a Gaussian mixture Kalman filter, in which negative weights may possibly arise. The algorithm in effect softens the Doppler blind zone constraint through the introduction of a fictitious measurement, as compared with our approach which takes account of the constraint directly.

A more recent approach to the problem of state dependent detection and tracking, generalizing the ideas of [16] to arbitrary Gaussian mixture approximations of the detection probability, is presented in [20]. The algorithm of [20] also combines the problem of state dependent detection probability with target existence techniques, so as to discriminate between false tracks in multitarget cluttered environments. In contrast to [16], an extra approximation stage is introduced in order to replace the resulting ‘negative’ Gaussian mixture with one of strictly positive mixture weights, thus improving algorithmic stability. However, when specifically applied to GMTI problems, this algorithm does not significantly depart from the modeling ideas presented in [16].

We mention also that the construction of a recently proposed filter [10] for tracking problems involving quantized measurements, while not explicitly addressed at GMTI tracking problems, employs some similar probability calculations.

The simulations reported in Section V demonstrate a uniform reduction in estimation error achieved by the new filter as compared with earlier proposed methods based on Gaussian or Gaussian mixture approximation (the EKF and the state dependent probability of detection approach of Koch et al. [16]). The new algorithm achieves, furthermore, a high order SIR particle filter ([1], [12], [21]) at greatly reduced computational cost. The improvements in performance of our method over the “fictitious measurement” approach of Koch et al. is a consequence of a more accurate approximation of the conditional density when a measurement is suppressed, employed in our algorithm. Multiple model versions of the new algorithm match, and in some challenging scenarios outperform, the ‘stopped’ model algorithms of Kirubarajan et al. [14].

The paper is organized as follows. The tracking problem is defined in Section II. Section III summarizes calculations required for the derivation of the tracking algorithm in Section IV. In Section IV, we describe the proposed methodology, and outline an algorithm based on its implementation. Section V reports on simulations involving the proposed algorithm and earlier trackers. Concluding remarks appear in Section VI.

Notation: in the following, $N(x; \mu, \Sigma)$ denotes the density of a normally distributed random vector $x$, with mean $\mu$ and covariance $\Sigma$. $\chi_A(y)$ is used to denote the indicator function: $\chi_A(y) = 0$, if $y \not\in A$, and $\chi_A(y) = 1$, if $y \in A$. Given a sequence of variables $\{x_t\}$ and integers $j, k$ with $j \leq k$, we write $x_{j:k}$ for the collection of variables $\{x_j, x_{j+1}, \ldots, x_k\}$.

II. THE TRACKING PROBLEM

The time-dependent $n$-dimensional state vector $x_t$, describing motion of a target, evolves according to the equation

$$x_t = Fx_{t-1} + u_{t-1} + w_{t-1} \quad t = 1, 2, \ldots$$

in which $F$ is an $n \times n$ matrix and, for each $t$, $u_t^s$ is an $n$-vector. $\{w_t\}$, the ‘system noise’ process, is a sequence of independent Gaussian random variables ($w_t \sim N(0, Q_t)$).

The components of the state vector $x_t$ will, in some cases, comprise the Cartesian coordinates and the velocity components of the target. But, in other cases, the state might incorporate additional components to take account of coloured noise effects, motion of the sensor platform, etc. The vector $u_t^p$ represents a deterministic (control) input. There are no restrictions on the dimension $n$ of the state vector.

Let $p$ be the ‘dimensionality’ of the Cartesian position of the target (for measurements in the plane and 3-D space, $p$ takes the values 2 and 3, respectively). To develop a model for the observations, we introduce a $p$-dimensional time-varying displacement vector $d_t$ according to

$$d_t = H_x x_t + u_t^m.$$

Here, for each $t$, $u_t^m$ is a deterministic $p$-vector, and $H_x$ is a $p \times n$ matrix. $d_t$ has the interpretation of the Cartesian coordinates of the position of the target relative to the sensor platform position (the location of the sensor platform is assumed perfectly known). We also introduce the $p$-dimensional target velocity vector $v_t$. We assume that $v_t$ can be expressed as a linear function of the state $x_t$

$$v_t = H_v x_t$$

for some $p \times n$ matrix $H_v$.

With the previous definitions, the range $r_t$, azimuth angle $\theta_t$, elevation angle $\phi_t$ and range rate $r_t$ of the target are expressible in terms of the displacement and the state vectors, $d_t$ and $x_t$, as follows (in the case $p = 3$):

$$r_t = \|d_t\|$$

$$b_t(x_t) = (\cos \theta_t \cos \phi_t, \sin \theta_t \cos \phi_t, \sin \phi_t)^T = \|d_t\|^{-1}d_t$$

$$r_t = b_t^T(x_t)v_t$$

where we note that $b_t(x_t)$ is the $p$-dimensional bearing vector (vector of direction cosines), and is a nonlinear function of the state vector $x_t$. The azimuth and elevation angles $\theta_t$ and $\phi_t$ are associated with the bearing vector as shown above. Note also that, in the above definitions, we have assumed that the range rate is compensated for platform motion.

For a possible choice of the vectors $u_t^s$, $u_t^m$ and matrices $H_d$ and $H_v$, see the example application in Section V describing simulation results.
Now introduce the signal (the ‘enhanced’ measurement) $y_t$, obtained by assembling range, azimuth, elevation and range rate as a $p + 1$-vector and adding Gaussian noise

$$y_t = (y_{t,1}, y_{t,2}, y_{t,3}, y_{t,4})^T = (r_t, \theta_t, \phi_t, \dot{r}_t)^T + \mathbf{n}_t. \quad (1)$$

Here, $\{\mathbf{n}_t\}$ is a sequence of independent Gaussian random variables $\mathbf{n}_t \sim N\left(0, Q^m\right)$. $Q^m$ is a diagonal matrix, $Q^m = \text{diag}(\sigma^2_r, \sigma^2_\theta, \sigma^2_\phi, \sigma^2_{\dot{r}})$ whose diagonal elements comprise the measurement noise variance in range, azimuth, elevation and range rate.

Construction of the observation process model also requires introduction of the sequence $\{\xi_t\}$ of independent discrete random variables taking values 0 or 1 (the ‘detection variables’) for which

$$P[\xi_t = 1] = P_d \quad \text{and} \quad P[\xi_t = 0] = 1 - P_d.$$ 

Here, $P_d$ the probability of detection, is some constant, $0 \leq P_d \leq 1$. It is assumed that $\{\mathbf{w}_t\}$, $\{\mathbf{n}_t\}$, $\{\xi_t\}$ and $\mathbf{x}_0$ are independent. Note that we assume the probability of detection $P_d$ to be constant outside the blind zone. In some applications (not covered here) it is more appropriate to assume that $P_d$ depends on the state $x_t$.

The observation process $z_t$ takes values in $R^{p+1} \cup O$. Here, $O$ denotes a ‘missing’ measurement. If a measurement is recorded, $z_t$ coincides with the enhanced-measurement $y_t$. If a measurement is not recorded, $z_t = O$. The precise circumstances under which a measurement is not recorded are: either the target is not detected $3$ ($\xi_t = 0$) or the target is detected $3$ ($\xi_t = 1$) but lies in the Doppler blind zone, i.e. $y_{t,4} \in A$ where $A$ is the interval

$$A \equiv (-mdv_r, +mdv_r).$$

The constant $mdv$ is termed the minimum detectable velocity threshold. Accordingly

$$z_t = \begin{cases} O & \text{if } \xi_t = 0 \text{ or } (\xi_t = 1 \text{ and } y_{t,4} \in A) \\ y_t & \text{if } \xi_t = 1 \text{ and } y_{t,4} \notin A. \end{cases} \quad (2)$$

The tracking problem is to obtain recursive formulae, providing accurate approximations to the conditional mean and covariance of the state $x_t$ at time $t$, given the observation history up to time $t$, $z_{1:t}$.

### III. Preliminary Analysis

The distinctive feature of the proposed filter is the manner in which it takes account of the information that a measurement has been suppressed, when the measured range rate lies in the Doppler blind zone. It makes use of formulae for the conditional mean and covariance of a random variable $x_t$ given that a scalar measurement $m$ lies in a specified interval $A$. In constructing the algorithm we will of course identify $x_t$ with the current state and interpret ‘$m \in A$’ as ‘a measurement is suppressed’. The relevant analysis is summarized as the following proposition, which is the result of routine calculations (see e.g. [9] for a proof):

**Proposition 1**: Take independent random variables $x \sim N(\mathbf{x}_0, P_0)$ and $w \sim N\left(0, \sigma^2\right)$ and an interval $A = [a, b]$. Define the scalar random variable $m$ to be

$$m = q^T x + w.$$ 

(Here $\mathbf{x}_0$, $q$ are given $n$-vectors, $P_0$ is a given $n \times n$ covariance matrix, and $\sigma^2 > 0$). Write

$$K = P_0 q (q^T P_0 q + \sigma^2)^{-1}$$

$$P = P_0 - K q^T P_0,$$

Then

$$E[x|m \in A] = \mathbf{x}_0 + K [\hat{m}_A - \mu] \quad (3)$$

$$\text{cov}[x|m \in A] = P + KV_A K^T \quad (4)$$

where

$$\mu = q^T \mathbf{x}_0 \quad (5)$$

$$\hat{m}_A = E[m|m \in A] \quad (6)$$

$$V_A = \text{cov}[m|m \in A]. \quad (7)$$

Furthermore

$$\hat{m}_A = c^{-1} \sigma^2 \left[ N(\alpha; \mu, \sigma^2) - N(b; \mu, \sigma^2) \right] + \mu \quad (8)$$

$$V_A = c^{-1} \sigma^2 \left[ (a + \mu) N(\alpha; \mu, \sigma^2) - (b + \mu) N(b; \mu, \sigma^2) \right] + (\sigma^2 - \hat{m}_A^2) \quad (9)$$

where

$$\sigma^2 = q^T P_0 q + \sigma^2.$$ 

Here, and below, $c$ denotes a normalizing constant ensuring that a probability density integrates to unity. In this case

$$c = \int_a^b N(m; \mu, \sigma^2) dm.$$ 

We emphasize that the formulae in this proposition, expressing the first two conditional moments of $x$ given ($m \in A$) in terms of simple evaluations of a scalar normal density and the complementary error function, are exact.

### IV. Estimation Methodology Incorporating the Doppler Blind Zone

#### A. Conditional Density Calculations

Consider the GMTI tracking problem formulated in Section II. The proposed filter propagates an approximation to the conditional density of the state $x_t$ given the measurement history $z_{1:t}$, in the form of Gaussian mixtures. The construction of the filter at time $t$ is based on the assumption that the conditional density at time $t - 1$ takes the form

$$p(x_{t-1}|z_{1:t-1}) = \sum_{i=1}^N \alpha_i^{t-1} N\left(x_{t-1}; \bar{x}_{t-1}^i, \bar{P}_{t-1}^i\right) \quad (10)$$

A detection event, in this context, is identified with the thresholding problem ($P_d < 1$) of the sensor, and is not to be confused with the Doppler blind zone criterion.

---

3A detection event, in this context, is identified with the thresholding problem ($P_d < 1$) of the sensor, and is not to be confused with the Doppler blind zone criterion.
where the integer $N$, the ‘weights’ $\{\alpha^j_{t-1}\}$, the vectors $\hat{x}^j_{t-1|t-1}$ and the covariance matrices $P^j_{t-1}$ are given parameters. The filter generates an approximation

$$p(x_t|z_{1:t}) = \sum_{j=1}^{N} \alpha^j_t N(x_t; \hat{x}^j_{t|t}, P^j_t)$$

(11)

to the conditional density $p(x_t|z_{1:t})$, taking account of the new measurement (or its absence), which will be used as data for the next iteration at time $t$.

Denote by $\hat{y}_t$ the enhanced-measurement at time $t$, when the $(r_t, \theta_t, \phi_t)$ components are expressed in Cartesian coordinates $\hat{y}_t = (y_{h,1}\cos y_{h,2}\cos y_{h,3}, y_{h,1}\sin y_{h,2}\cos y_{h,3}, y_{h,1}\sin y_{h,3}, y_{r,1})^T$. (12)

We treat the 3-D space problem here: $p = 3$). The range-rate component is not transformed, i.e. $\hat{y}_{r,t} = y_{r,t}$. For purposes of filter construction, we assume that $\hat{y}_t$ is modelled by

$$\hat{y}_t = H^t x_t + u^r_t$$

(13)

where

$$b_t(x_t) = \|H^t x_t + u^r_t\|^{-1} (H^t x_t + u^r_t)$$

(14)

and $u^r_t$ is a zero mean, conditionally independent of $x_t$, given $x_{t-1}$ and with covariance matrix $R_t$ calculated from the measurements

$$R_t = \begin{bmatrix} \sigma^2_{x} & \sigma^2_{y} & \sigma^2_{z} \\ \sigma^2_{xy} & \sigma^2_{y} & \sigma^2_{z} \\ \sigma^2_{xz} & \sigma^2_{yz} & \sigma^2_{z} \\ 0 & 0 & 0 \end{bmatrix}$$

(15)

Formulae for the elements of this matrix are given in the Appendix. Note that (12) only approximately describes the transformed enhanced-measurement $\hat{y}_t$ when $y_t$ is governed by (1). The use of more sophisticated models (based on “unbiased transformations” [3], [18]) may alternatively be employed.

In order to compute the parameters defining $p(x_t|z_{1:t})$ (see (11)), we need to consider two cases separately:

**Case A:** ($z_t = \emptyset$, i.e., no measurement is recorded)

An unrecorded measurement may have two distinct causes: either the target is not detected ($\xi_t = 0$), or the target is detected but the target measurement is suppressed due to Doppler attenuation ($\xi_t = 1$ and $y_{r,t} \in A$). It follows that:

$$p(x_t|z_{1:t}) = p(x_t|z_{1:t-1}, \xi_t = 0)p(\xi_t = 0)$$

$$+ p(x_t|z_{1:t-1}, \xi_t = 1, y_{r,t} \in A)$$

$$\times p(\xi_t = 1)$$

$$\times p(y_{r,t} \in A|z_{1:t-1})$$

(10)

Bearing in mind our assumption that the prior density is (10), we deduce that

$$p(x_t|z_{1:t}) \approx c^{-1} \sum_{j=1}^{N} \alpha^j_t [p(x_t)(1 - P_d)$$

$$+ p_t(x_t|y_{r,t} \in A)P_d p_t(y_{r,t} \in A)]$$

(16)

where, on the RHS of the above expression (and below), we drop the conditioning on $z_{1:t-1}$ for notational simplicity (it is implied throughout). In (16), $c$ is a normalizing constant and the densities $p_t(x_t|y_{r,t} \in A)$, $p_t(x_t)$, etc. are computed from the joint density

$$p_t(x_t, y_{r,t} \in A) = \int p_t(x_t, y_{r,t} \in A|z_{1:t})$$

$$\times N(x_{r,t}; \hat{x}_{r,t|t-1}, P_{r,t|t-1})$$

(17)

and i.e. $p_t(x_t, y_{r,t} \in A)$ is computed from the state and measurement equation, under the assumption that $x_{r,t} \sim N(x_{r,t}; \hat{x}_{r,t|t-1}, P_{r,t|t-1})$. It follows immediately that

$$p_t(x_t) = N(x_t; \hat{x}_{r,t|t-1}, P_{r,t|t-1})$$

and

$$\hat{x}_{r,t|t-1} = F^r \hat{x}_{r,t|t-1} + u^r_{t-1}$$

(18)

$$P_{r,t|t-1} = F^r P_{r,t|t-1} F^r + Q_r^t.$$
and
\[ \tilde{\eta}_t^\mu = (\gamma_t^\mu)^{-1} \mu^2 \left[ N(\mu; \mu, \sigma^2) - N(b_\mu, \sigma^2) \right] + \mu \]  
(29)
\[ V_t^\mu = (\gamma_t^\mu)^{-1} \mu^2 \left[ (a + \mu) N(\mu; \mu, \sigma^2) + (b + \mu) N(b_\mu, \sigma^2) \right] + (\mu^2 + \sigma^2) \]  
(30)
\[ \sigma_t^2 = \mathbf{Q}_t^s \mathbf{P}_t^s_{t-1} \mathbf{Q}_t^s + \sigma_t^2 \]  
(31)
\[ \gamma_t^\mu = \int_{-\infty}^{\infty} N(m_{t-1}, \sigma^2) \mu \, dm_{t-1} \]  
(32)

To complete the approximation of (16), it remains to consider \( p_i(y_{t, A} \in A) \). We approximate this probability by \( p_i(y_{t, A} \in A) \), the modified probability obtained, once again, by the linearized measurement of equation (21). This yields \( \gamma_t^i \) (defined in (32)) as an approximation to \( p_i(y_{t, A} \in A) \).

Thus, the approximation \( \tilde{p}(x_t | z_{1:t}) \) to the posterior density, when a measurement is unrecorded, can be written as
\[
\tilde{p}(x_t | z_{1:t}) = \frac{1}{\gamma_t^i} \sum_{i=1}^{N} \alpha_{i-1}^t \left[ P_{t-1} \mathbf{P}_t^i + N \left( \tilde{x}_t | \tilde{x}_t_{t-1}^i, P_{t-1}^i \right) \right]
+ (1 - P_d) N \left( x_t | \tilde{x}_t_{t-1}^i, P_{t-1}^i \right)
\]
(33)

Notice that the approximation to \( \tilde{p}(x_t | z_{1:t}) \) is a 2-fold Gaussian mixture. A mixture reduction step is required to propagate an \( N \)-fold mixture—number of effective techniques available for this purpose [11, 22, 23].

Case B: \( z_t = \tilde{y}_t \) (i.e., a measurement is recorded)

In the case when a measurement is recorded, we can use any of a number of analytic algorithms currently available to estimate the Gaussian mixture approximation to \( \tilde{p}(x_t | z_{1:t}) \) to \( p(x_t | z_{1:t}) \). (See (11).) For concreteness, we describe an algorithm in the spirit of the parameterized extended Kalman filter (c.f. [2]): such algorithms are essentially weighted sums of the outputs of a bank of EKFs. The procedure is summarized as follows.

Convert the polar measurements to Cartesian coordinates and linearize the rate measurement, i.e., replace the measurement equation by (13)–(15). For each \( i \), obtain \( \tilde{x}_t^i_{t-1} \) and \( P_t^i_{t-1} \) in (11), by applying the extended Kalman filter to the state equation and converted measurement equation with prior state density \( N(\tilde{x}_t^i_{t-1}, P_t^i_{t-1}) \). This involves linearizing the measurement nonlinearity (14) about the predicted value \( \tilde{x}_t^i_{t-1} \) as in Case A (see (20)–(23)).

Finally, the weight \( \alpha_t^i \) is taken to be \( \alpha_t^{i-1} \) scaled by a normalizing constant and by a likelihood function associated with the measurement \( z_t = \tilde{y}_t \) and the prior state density \( N(\tilde{x}_t^i_{t-1}, P_t^i_{t-1}) \). This procedure leads to the following formulae:
\[
\tilde{x}_t^i_{t-1} = F \tilde{x}_t^i_{t-1} + \mathbf{u}_t^i
\]
(34)
\[
P_t^i_{t-1} = F P_t^i_{t-1} F^T + \mathbf{Q}_t^i
\]
(35)
\[
\dot{\mathbf{y}}_t^i = \dot{\mathbf{b}}_t (\tilde{x}_t^i_{t-1})
\]
(36)
\[
H_t^i = \begin{bmatrix}
(b_t^T) & H_t^i \nabla \tilde{x}_t^i_{t-1} (b_t(x_t))
\end{bmatrix}
\]
(37)
\[
\hat{y}_t^i_{t-1} = \begin{bmatrix}
H_t^i \mathbf{g}_t^i_{t-1} + \mathbf{u}_t^i
\end{bmatrix}
\]
(38)
\[
S_t^i = H_t^i P_t^i_{t-1} H_t^i + \mathbf{R}_t
\]
(39)
\[
K_t^i = P_t^i_{t-1} (H_t^i)^T S_t^i
\]
(40)
\[
\pi_t^i = N \left( \hat{y}_t^i; \hat{y}_t^i_{t-1}, S_t^i \right)
\]
(41)
\[
\tilde{x}_t^i_{t-1} = \tilde{x}_t^i_{t-1} + K_t^i (\hat{y}_t^i - \hat{y}_t^i_{t-1})
\]
(42)
\[
P_t^i_{t} = (I - K_t^i H_t^i) P_t^i_{t-1}
\]
(43)

The approximation \( \tilde{p}(x_t | z_{1:t}) \) to the posterior density can now be written as
\[
\tilde{p}(x_t | z_{1:t}) = \frac{1}{\gamma_t^i} \sum_{i=1}^{N} \alpha_{i-1}^t \pi_t^i N \left( x_t | \tilde{x}_t^i_{t-1}, P_t^i_{t-1} \right)
\]
(44)

B. Algorithm Outline

The main steps of the NRDB algorithm are now summarized:
- For each \( i = 1, \ldots, N \)
  - Calculate \( \tilde{x}_t^i_{t-1} \) and \( P_t^i_{t-1} \) from (18) and (19).
  - Case A (\( z_t \in \mathcal{C} \)):
    1. Use the ‘EKF-like’ equations (22)–(32) to evaluate \( \tilde{x}_t^i_{t-1}, P_t^i_{t-1} \) and \( \gamma_t^i \).
    2. Define the updated weights \( \alpha_t^{i-1} = \alpha_t^{i-1} \gamma_t^i P_d \) and \( \alpha_t^i = \alpha_t^{i-1} (1 - P_d) \).
  - Case B (\( z_t \notin \mathcal{C} \)):
    1. Calculate \( \tilde{x}_t^i_{t-1} \) and \( P_t^i_{t-1} \) from (12), (15) and (36)–(43).
    2. Evaluate the updated weights \( \alpha_t^i = \alpha_t^{i-1} \pi_t^i \).
- Replace the weights \( \alpha_t^{ij} \) (for \( j = 1, 2 \)) in Case A, or \( \alpha_t^i \) in Case B, by the normalized weights \( \alpha_t^{ij} \) or \( \hat{\alpha}_t^i \), respectively:
  - Case A: \( \alpha_t^{ij} = \alpha_t^{ij} / \sum_{j=1}^{N} \alpha_t^{ij} \)
  - Case B: \( \hat{\alpha}_t^i = \alpha_t^i / \sum_{i=1}^{N} \alpha_t^i \)
- Obtain Gaussian mixture approximations to the posterior density:
  - Case A:
    \[
    \tilde{p}(x_t | z_{1:t}) = \sum_{i=1}^{N} \alpha_{i-1}^t \pi_t^i N \left( x_t | \tilde{x}_t^i_{t-1}, P_t^i_{t-1} \right)
    \]
  - Case B: \( \tilde{p}(x_t | z_{1:t}) = \sum_{i=1}^{N} \hat{\alpha}_t^i N \left( x_t | \tilde{x}_t^i_{t-1}, P_t^i_{t-1} \right) \)
- Replace the high order Gaussian mixture density \( \tilde{p}(x_t | z_{1:t}) \) by a lower order Gaussian mixture, using a suitable reduction algorithm (e.g. [11], [22], [23]).

If the prior density is an \( N \)-fold Gaussian mixture, the proposed algorithm has complexity that of \( N \) parallel extended Kalman filters. This holds true whether or not a measurement is recorded. Note that, in the ‘no measurement’ case, the resulting ‘EKF-like’ filter (that is, (26)–(28)) operates on a scalar range.
rate variable, so that no matrix inversion is required (see (27)). The departure from the standard EKF approach is the calculation of the mean and variance of a scalar truncated Gaussian random variable ((29), (30)). Note that the evaluation of these moments merely involves the execution of two ‘error function’ calls and two evaluations of a scalar Gaussian density function.

C. Extensions for Clutter and Multiple Models

We note that the basic algorithm can be straightforwardly modified to allow for clutter and manoeuvring target motion described by a jump Markov linear model. There are several ways to perform such extensions; the construction of filters taking account of such extended problem formulations, however, is along similar lines to that of the basic NRDB algorithm in all cases.

We will refer to the resulting algorithm as NRDB-MM in the rest of this paper. Space constraints do not permit us to provide full details of the NRDB-MM algorithm (as the notation developed in Section II has to be redefined). In what follows, we provide some extra guidance on how to construct the extended algorithm—full details of the NRDB-MM algorithm can be found in the technical report [7].

In summary, manoeuvring target motion is modeled within a multiple motion model framework, see eg. [6], [8]. Presence of clutter (assumed uniformly distributed in space, and Poisson in population) is taken into account using probabilistic data association (PDA) techniques, as in [13], suitably modified to accommodate the extra hypothesis that a non-association event occurs due to the mdw constraint.

In this case, an $N$-fold Gaussian mixture prior gives rise to a posterior density consisting of $NM(n_t + 2)$ Gaussian mixture components, where $M$ is the number of (motion) modes, and $n_t$ is the number of (validated) radar returns. Again, we mention that there is a variety of techniques to perform mixture condensation—see the next Section reporting on simulation results where one possible choice is outlined.

Finally, note that the basic NRDB algorithm is a special case of NRDB-MM, in which $M = 1$ and the clutter spatial density, $\rho$, is equal to 0 (i.e. no clutter).

V. SIMULATION RESULTS

A. Scenario I—Single-Target Single-Sensor

In this section, we report on simulations of the proposed algorithm in a scenario of military relevance in which a GMTI sensor, mounted on an airborne platform, is employed to track the motion of a moving, ground based vehicle. The algorithms that are considered are the proposed NRDB algorithm, the state-dependent probability of detection filter (SDPD) from [16], a standard first-order Extended Kalman Filter (EKF), as well as an SIR particle filter, which is treated as a benchmark. In addition, a multiple-model implementation of the EKF (EKF-MM) will be separately compared with the NRDB-MM algorithm. This latter comparison is carried out separately because multiple-model implementations are, in general, expected to exhibit improved performance.

We choose a simple scenario, which nonetheless reveals the benefits of algorithms that take account of the Doppler blind zone, and which permits us to compare their performance. Details of the scenario are as follows:

The target, moving eastbound, starts at a constant speed of 10 m/s, which it maintains for 180 s, before it accelerates at a rate of 1 m/s$^2$ up to a speed of 25 m/s. After traveling at this constant speed for 180 s, the target starts to decelerate for 25 s until it comes to a standstill. No measurements are received while the target is stopped. The target remains stationary for 60 s, before accelerating again to a speed of 15 m/s. Target speed as a function of time is plotted in Fig. 1(a). Notice that a scenario has been chosen in which the target is stationary over a significant time period, to assess the performance of trackers that exploit blind Doppler zone information.

The target state vector comprises the target position and velocities on the ground plane, $x_t = (x_t, \dot{x}_t, y_t, \dot{y}_t)^T$. To model target motion we adopt a direct discrete-time constant velocity model [3], with additive white noise of intensity $\sigma_v^2 = (0.5)^2 (m/s^2)^2$. In this case, the vector $\mathbf{u}_t$ is simply the zero vector.

We also consider a sensor platform, travelling northbound at a constant speed of 120 m/s and at an altitude of approximately 10 km, which takes noisy measurements of the ground-moving target every $h = 5$ seconds. We use the method described in [19] to transform the 3-D range and range rate measurements to their planar equivalents (thus compensating for platform height) and set $p = 2$. Sensor position is described by the vector $\mathbf{v}_t^m = (-x_t^p, -y_t^p)^T$, where $x_t^p$ and $y_t^p$ denote the $x$ and $y$ coordinates of the sensor platform. The matrices $H_d$ and $H_v$ of Section II then become

$$ H_d = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad H_v = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. $$

![Scramble](Fig. 1. Scenario I configuration. (a) Ground target speed; (b) sensor and target trajectories.)
TABLE I
BLIND ZONE PERFORMANCE ASSESSMENT (SCENARIO I): TIME-AVERAGED RMSE, IN METERS (x-COORDINATE)

<table>
<thead>
<tr>
<th>Scenario</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
<th>VI</th>
<th>VII</th>
<th>VIII</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P_d ) (m/sec)</td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
</tr>
<tr>
<td>( \sigma_v ) (m/sec)</td>
<td>1.5</td>
<td>1.0</td>
<td>0.75</td>
<td>0.6</td>
<td>1.5</td>
<td>1.0</td>
<td>0.75</td>
<td>0.6</td>
</tr>
<tr>
<td>( MDV/\sigma_v )</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Time-Averaged RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>EKF</td>
<td>420.89 418.08 462.27</td>
</tr>
<tr>
<td>SDPD</td>
<td>241.84 220.54 280.94</td>
</tr>
<tr>
<td>NRDB</td>
<td>197.41 168.31 219.18</td>
</tr>
<tr>
<td>SIR-PF</td>
<td>198.78 163.47 221.05</td>
</tr>
<tr>
<td>EKF-MM</td>
<td>56.29 55.64 59.761</td>
</tr>
<tr>
<td>NRDB-MM</td>
<td>56.32 56.771 61.915</td>
</tr>
</tbody>
</table>

TABLE II
BLIND ZONE PERFORMANCE ASSESSMENT (SCENARIO I): TIME-AVERAGED RMSE, IN METERS (y-COORDINATE)

<table>
<thead>
<tr>
<th>Scenario</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
<th>VI</th>
<th>VII</th>
<th>VIII</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P_d ) (m/sec)</td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
</tr>
<tr>
<td>( \sigma_v ) (m/sec)</td>
<td>1.5</td>
<td>1.0</td>
<td>0.75</td>
<td>0.6</td>
<td>1.5</td>
<td>1.0</td>
<td>0.75</td>
<td>0.6</td>
</tr>
<tr>
<td>( MDV/\sigma_v )</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Time-Averaged RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>EKF</td>
<td>105.57 118.52 126.08</td>
</tr>
<tr>
<td>SDPD</td>
<td>101.11 113.63 123.61</td>
</tr>
<tr>
<td>NRDB</td>
<td>101.82 114.8 123.31</td>
</tr>
<tr>
<td>SIR-PF</td>
<td>121.64 122.12 133.55</td>
</tr>
<tr>
<td>EKF-MM</td>
<td>34.019 36.73 39.487</td>
</tr>
<tr>
<td>NRDB-MM</td>
<td>40.167 49.014 53.35</td>
</tr>
</tbody>
</table>

The sensor inaccuracies are characterized by standard deviations of \( \sigma_v = 20 \) m and \( \sigma_v = 1 \) m rad. The range rate noise standard deviation, \( \sigma_v \), and the probability of detection \( P_d \) vary as summarized in Table I. The minimum detectable velocity threshold, \( MDV \), is set to 3 m/s, while the spatial density of the clutter, \( \rho \), is set so low as to play no role in the simulations. Plots of the target and sensor platform paths are given in Fig. 1(b).

At the end of each iteration of the proposed algorithm, a standard mixture reduction technique [22] was employed to avoid the exponential growth of the ‘dimensionality’ of the Gaussian mixture approximation to the posterior density. Specifically, the posterior mixture of the NRDB and SDPD algorithms was reduced to two components at each time step—this was appropriate since the densities encountered were typically bimodal.

A standard EKF, which, in the absence of a measurement at time \( t \), produces the predictive estimates \( x_{\text{EKF}}(t|t-1), P_{\text{EKF}}(t|t-1) \) as the time-updated posterior mean and covariance, is also included in the comparison, so as to demonstrate the benefits of incorporating the blind Doppler constraints of the sensors in the tracker.

The particle filter employed in this example is a standard 5000-particle SIR implementation, and is mainly used here as a performance benchmark, as the CRLB calculation for the problem is intractable. The implementation of the particle filter was based on the actual (noise-corrupted) range-rate model to account for suppressed measurements, with the Doppler blinness constraint incorporated in the measurement update stage (likelihood evaluation) of the algorithm—the actual algorithm being a particular case of the more general particle filter for quantized measurements [12], properly modified to account for the \( (P_d < 1) \) thresholding of the sensor.

Finally, both multiple-model algorithms, that is EKF-MM and NRDB-MM, employ \( M = 3 \) motion models to describe target motion as follows: a) a low-intensity constant velocity model, with \( \sigma_v^2 = (0.05)^2(\text{m/s}^2)^2 \), b) a high-intensity constant velocity model, with \( \sigma_v^2 = (0.5)^2(\text{m/s}^2)^2 \), to cope with target acceleration, and c) a stopped model (see for example [14]). The relevant mode transition probabilities were calculated according to the method described in [14]—the resulting transition matrix \( \nu \) is given by (45). The number of retained components was set to a total of 9 for both algorithms; of these, 5 correspond to components in the stopped mode, while two components correspond to each of the modes (a) and (b), respectively.

\[
\nu = \begin{pmatrix}
0.9500 & 0.0495 & 0.0005 \\
0.2182 & 0.7273 & 0.0545 \\
0.0008 & 0.0825 & 0.9167
\end{pmatrix}.
\]

All algorithms were initialized using a single Gaussian prior density \( \mathcal{N}(\mathbf{x}; \mathbf{0}, P_0) \), obtained by two-point differencing [3].

The time-averaged 100-trial Root Mean Squared Errors of the \( x \) and \( y \) coordinates are listed in Tables I and II, for the time interval during which the target is stationary (sample times \( k = 80 \) to \( k = 92 \)). Representative RMS error curves of the \( x \) and \( y \) coordinates for the relevant time intervals are shown in Figs. 2(a) and 2(b) for algorithms employing simple Markovian dynamics, and Figs. 2(c) and 2(d) for multiple-model implementations. Estimation errors associated with the \( x \) coordinate are most relevant here for comparison purposes, as estimates of the \( y \) coordinate are practically indistinguishable for all methods (see Fig. 2(b)).

The proposed NRDB method consistently outperforms the rival SDPD method by 18% to 25% with respect to the \( x \) coordinate RMSE values, most notably as the ratio \( m dh/\sigma_v \) increases. In particular, the estimates produced by the proposed NRDB algorithm are nearly indistinguishable from those of the particle filter which, it can be assumed, closely approximates the optimal filter (they typically differ by less than 1%). All methods that exploit the information on the minimum detectable velocity constraint perform significantly better than the EKF.
With respect to the $y$ coordinate, all the aforementioned algorithms show comparable performance, which is marginally better than that of the standard EKF [see Table II and Fig. 2(b)]. The multiple-model implementations, EKF-MM and NRDB-MM, produce similar estimates for the $x$ coordinate [Fig. 2(c)]; as expected, these are superior to estimates obtained by the simpler algorithms whose performance is illustrated in Fig. 2(a). With respect to the $y$ coordinate, the EKF-MM algorithm outperforms the NRDB-MM, as the former method gives greater weighting to Gaussian mixture components corresponding to the stopped model in that time interval. Again, both algorithms perform significantly better than their single-mode counterparts [see Table II and Figs. 2(b), 2(d)].

Finally, all algorithms exhibit similar performance over the rest of the track (when the target is detected by the sensor).

B. Scenario II—Single-Target Multiple-Sensor

Here, we revisit the previous scenario with one modification: we introduce a second airborne sensor, travelling eastbound, on a path parallel to the target. The two sensors are characterized by the same measurement uncertainties as in the previous section. In this case, however, both sensors take measurements every $h = 10$ sec, with a 5 s offset between the two; thus, a target measurement is effectively taken every $h_{eff} = 5$ sec, but from interchangeable sensor positions. (Sensor 1 takes measurements at odd sample times, while Sensor 2 takes measurements at even sample times, for an effective sample period of $h_{eff} = 5$ sec.)

The paths of the target and sensor platforms are illustrated in Fig. 3(a). The target moves as in the previous section, with a velocity profile as shown in Fig. 1(a). All the other parameters relevant to the simulation were set to the same values as in Scenario I.

The addition of Sensor 2, moving in a direction perpendicular to that of Sensor 1, adds to the observability of the target as far as the blind Doppler constraint is concerned. However, the problem is set up in such a way so that the target “goes blind” with respect to the second sensor, while it is still in motion (i.e. before the stop manoeuvre is executed). This is indeed the case, as the paths traversed by Sensor 2 and the target are parallel, making the line of sight perpendicular to the direction of target motion between sample times $k = 62$ and $k = 80$, when the target actually stops. This is highlighted in Fig. 3(b), which shows the (noisy) range-rate values collected by the two sensors, along with the blind zone limits. Because of this last complication, the problem is particularly challenging for algorithms that do not take account of the Doppler blind zone in the measurement process model.

The time-averaged 100-trial RMSE for the $x$ and $y$ coordinates, for the the interval of target stop (sample times $k = 80$ to
TABLE III
BLIND ZONE PERFORMANCE ASSESSMENT (SCENARIO II): TIME-AVERAGED RMSE, IN METERS (x-COORDINATE)

<table>
<thead>
<tr>
<th>Scenario 2</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
<th>VI</th>
<th>VII</th>
<th>VIII</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_d$ (m/sec)</td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
</tr>
<tr>
<td>$\sigma_f$ (m/sec)</td>
<td>1.5</td>
<td>1.0</td>
<td>0.75</td>
<td>0.6</td>
<td>1.5</td>
<td>1.5</td>
<td>1.0</td>
<td>0.75</td>
</tr>
<tr>
<td>$M/DV/\sigma_f$</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>EKF</td>
<td>520.6</td>
<td>575.9</td>
<td>638.88</td>
<td>518.85</td>
<td>355.21</td>
<td>410.63</td>
<td>375.37</td>
<td>346.08</td>
</tr>
<tr>
<td>SDPD</td>
<td>449.58</td>
<td>498.51</td>
<td>562.56</td>
<td>440.8</td>
<td>210.3</td>
<td>274.65</td>
<td>242.65</td>
<td>217.89</td>
</tr>
<tr>
<td>NRDB</td>
<td>427.6</td>
<td>477.04</td>
<td>534.86</td>
<td>417.38</td>
<td>158.89</td>
<td>212.36</td>
<td>189.67</td>
<td>180.2</td>
</tr>
<tr>
<td>SIR-PF</td>
<td>425.84</td>
<td>475.26</td>
<td>524.18</td>
<td>419.33</td>
<td>158.02</td>
<td>212.86</td>
<td>189.04</td>
<td>182.92</td>
</tr>
<tr>
<td>EKF-MM</td>
<td>123.01</td>
<td>90.083</td>
<td>193.81</td>
<td>115.39</td>
<td>63.238</td>
<td>92.338</td>
<td>71.858</td>
<td>56.819</td>
</tr>
<tr>
<td>NRDB-MM</td>
<td>105.74</td>
<td>77.693</td>
<td>157.59</td>
<td>102.05</td>
<td>49.322</td>
<td>65.529</td>
<td>49.623</td>
<td>38.32</td>
</tr>
</tbody>
</table>

TABLE IV
BLIND ZONE PERFORMANCE ASSESSMENT (SCENARIO II): TIME-AVERAGED RMSE, IN METERS (y-COORDINATE)

<table>
<thead>
<tr>
<th>Scenario 2</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
<th>VI</th>
<th>VII</th>
<th>VIII</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_d$ (m/sec)</td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
</tr>
<tr>
<td>$\sigma_f$ (m/sec)</td>
<td>1.5</td>
<td>1.0</td>
<td>0.75</td>
<td>0.6</td>
<td>1.5</td>
<td>1.5</td>
<td>0.75</td>
<td>0.6</td>
</tr>
<tr>
<td>$M/DV/\sigma_f$</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>EKF</td>
<td>125.53</td>
<td>147.1</td>
<td>137.27</td>
<td>124.79</td>
<td>117.26</td>
<td>123.53</td>
<td>117.59</td>
<td>120.99</td>
</tr>
<tr>
<td>SDPD</td>
<td>85.916</td>
<td>87.979</td>
<td>93.119</td>
<td>81.679</td>
<td>50.095</td>
<td>55.371</td>
<td>51.48</td>
<td>50.611</td>
</tr>
<tr>
<td>NRDB</td>
<td>81.349</td>
<td>85.009</td>
<td>89.606</td>
<td>77.792</td>
<td>43.794</td>
<td>46.705</td>
<td>42.656</td>
<td>44.212</td>
</tr>
<tr>
<td>SIR-PF</td>
<td>83.802</td>
<td>96.183</td>
<td>104.76</td>
<td>101.44</td>
<td>46.019</td>
<td>50.085</td>
<td>50.391</td>
<td>57.361</td>
</tr>
<tr>
<td>EKF-MM</td>
<td>36.373</td>
<td>41.895</td>
<td>37.291</td>
<td>38.937</td>
<td>33.009</td>
<td>33.948</td>
<td>30.842</td>
<td>31.437</td>
</tr>
<tr>
<td>NRDB-MM</td>
<td>34.154</td>
<td>35.598</td>
<td>32.772</td>
<td>33.425</td>
<td>27.117</td>
<td>27.279</td>
<td>24.18</td>
<td>24.436</td>
</tr>
</tbody>
</table>

Fig. 3. Scenario II configuration. (a) Sensor and target trajectories; (b) range-rate sequence.

$k = 92$), are tabulated in Tables III and IV respectively, along with the simulation parameters that are varied in the scenario (namely $P_d$ and $\sigma_f$). Fig. 4(a)–(d) show the 100-trial RMS errors for one set of these parameters.

The results demonstrate that the proposed NRDB algorithm consistently outperforms the SDPD method in both the $x$ and $y$ coordinates. In particular, the NRDB produces estimates that are approximately 5% more accurate (in both coordinates) than those produced by SDPD (for $P_d = 0.6$), while the increased accuracy is more pronounced when $P_d$ is set to 0.8, in which case the improvement lies between 18% and 25% in the $x$ direction, and 12% and up to 17% in the $y$ coordinate respectively.

The particle filter gives estimates of the $x$ coordinate of the target which are nearly indistinguishable from those obtained by the NRDB algorithm; Table IV and Fig. 4(b) show that the NRDB estimates of the $y$ coordinate are more accurate than those of the particle filter. Increasing the number of particles could lead to smaller discrepancies in this case.

Finally, in the multiple-model setting, the NRDB-MM algorithm exhibits an improvement in the order of 14% to 19% with respect to EKF-MM for a value of $P_d$ equal to 0.6, while, for $P_d = 0.8$, the improvement in estimation accuracy is in the order of 22% to 32% for both the $x$ and $y$ coordinates. What is more, these latter figures do not take into account the time interval before the target stops (sample times $k$ to $k = 80$), during which the EKF-MM algorithm estimates exhibit erratic behavior [see Fig. 4(c)].

More specifically, Fig. 4(c) reveals that, between sample times 62 and 80, the estimation error curve of EKF-MM follows a ‘sawtooth-like’ shape, jumping between high and low values at consecutive sample times. This behavior can be explained by looking at Figs. 3(b) and 5. These plots indicate that the EKF-MM algorithm gives a very large weight to the stopped model at every second (even) time step in the time interval $k = 62$ to $k = 80$, as a result of the (still-moving) target...
lying in the blind zone of Sensor 2, and the inability of the
algorithm to account for such a possibility. The NRDB-MM
algorithm, on the other hand, does not give significant weight
to the "target stop" mode until the target has indeed come to a
halt ($k = 80$). It therefore exhibits smaller tracking error and
results in smoother estimated tracks.

Finally, Table V lists the average CPU time required by all al-
gorithms, compared with the EKF. Among single-mode imple-
mentations, the NRDB algorithm requires the equivalent time of
three standard EKFs, thus reducing the computational de-
mand of the SDPD method by approximately 16%. The par-
ticle filter is fairly expensive compared with the other filters;
its computational demand is the equivalent of approximately
94 EKFs. In addition, we see that use of the multiple-model ver-
sion of the new algorithm (NRDB-MM) requires approximately
do double the CPU time of the EKF-MM method. Note that these
are still considerably less than the aforementioned particle filter
requirements.

For completeness, we mention that the proposed NRDB
and NRDB-MM techniques, compared to their BDMF and
BDMF-MM predecessors, have been found to exhibit increased
robustness in scenario variations while, at the same time, they
reduce the corresponding computational demands of these
algorithms by a factor of two.

We also note that no significant improvement was achieved
through the use of higher order mixture approximations in the
NRDB algorithm in the two scenarios considered.

VI. CONCLUSION

We have proposed a new methodology for incorporating the
Doppler blind zone constraints in target tracking, based on a new
measurement model for describing the observation process and
Gaussian mixture approximations to conditional densities. It is
suitable for use in tracking scenarios where the state-space is of
high dimension. A simple extension covers problems involving
multiple models and the presence of clutter.

Algorithms that result from implementing the approach are
very versatile. The user is free to choose any nonlinear filtering
scheme matching first and second moments (e.g., EKF, modified
KF, UKF) to process the nonlinear sensor measurements. The
modest computational demands of the new approach make it
suitable for incorporation into larger-scale algorithms that deal
with problems involving multiple targets, terrain obscuration,
road constraints, etc. In particular, the derived algorithm can be
used for the extension of the assignment procedure described
in [17], so as to simultaneously perform data association and
filtering by taking account of all target-sensor geometries in a
structured manner.
TABLE V  
AVERAGE CPU TIME COMPARISON

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>EKF</th>
<th>SDPD</th>
<th>NRDB</th>
<th>SIR-PF</th>
<th>EKF-MM</th>
<th>NRDB-MM</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU time (relative)</td>
<td>1</td>
<td>3.94</td>
<td>3.39</td>
<td>93.92</td>
<td>14.86</td>
<td>32.36</td>
</tr>
</tbody>
</table>

In this expression, \( r_0, \theta_0, \phi_0 \) denote the noiseless range, azimuth and elevation. The measurement noise, \( \mathbf{n} \), is a zero mean independent Gaussian random variable with covariance matrix \( Q = \text{diag}(\sigma^2_x, \sigma^2_y, \sigma^2_z) \).

Conversion formulae from polar measurements \((r, \theta, \phi)\) to Cartesian positions \((x, y, z)\), based on linearization, appear, for example, in [4]. They approximate the transformed Cartesian vector \( \mathbf{y} \) by a Gaussian random vector, whose mean, \( \mathbf{m} \), is given by

\[
\mathbf{m} = (r \cos \theta \cos \phi, r \sin \theta \cos \phi, r \sin \phi)^T
\]

and whose covariance matrix \( \mathbf{R} \) depends on the values of the (noisy) measurements \( r, \theta \) and \( \phi \), and is given by

\[
\mathbf{R} = \begin{bmatrix}
\sigma^2_x & \sigma^2_{xy} & \sigma^2_{xz} \\
\sigma^2_{yx} & \sigma^2_y & \sigma^2_{yz} \\
\sigma^2_{zx} & \sigma^2_{zy} & \sigma^2_z
\end{bmatrix}.
\]

In the above

\[
\sigma^2_x = \left( r \cos \theta \cos \phi \right)^2 \sigma^2_x + \left( r \sin \theta \cos \phi \right)^2 \sigma^2_\theta + \left( r \sin \phi \right)^2 \sigma^2_\phi
\]

(47)

\[
\sigma^2_y = \left( \sin \theta \cos \phi \right)^2 \sigma^2_x + \left( \cos \theta \cos \phi \right)^2 \sigma^2_\theta + \left( \sin \phi \right)^2 \sigma^2_\phi
\]

(48)

\[
\sigma^2_z = \sin^2 \phi \sigma^2_r + r^2 \cos^2 \phi \sigma^2_\phi
\]

(49)

\[
\sigma^2_{xy} = \frac{1}{2} \sin 2 \theta \left( \cos^2 \phi \sigma^2_r - r^2 \cos^2 \phi \sigma^2_\phi + r^2 \sin^2 \phi \sigma^2_\phi \right)
\]

(50)

\[
\sigma^2_{yx} = \frac{1}{2} \sin \theta \sin 2 \phi \left( \sigma^2_r - r^2 \sigma^2_{\phi} \right)
\]

(51)

\[
\sigma^2_{xz} = \frac{1}{2} \cos \theta \sin 2 \phi \left( \sigma^2_r - r^2 \sigma^2_{\phi} \right)
\]

(52)

REFERENCES


Fig. 5. Mode probabilities as functions of time. (a) Mode probability—EKF-MM; (b) mode probability—NRDB-MM.

Simulation results indicate that algorithms based on the new methodology exhibit a uniform improvement in estimation accuracy, as compared with earlier Gaussian mixture techniques, while they compare favourably to particle filters. Furthermore, the resulting Gaussian mixture representations have strictly positive weights, in contrast to earlier approaches, which makes the process of mixture reduction numerically robust.

To conclude, we note that as the filtering method presented in this paper essentially proposes the use of a new model to describe the observation process of GMTI sensors, application to real datasets is essential in order to have an objective assessment of the applicability of the various algorithms considered in this comparative study.

APPENDIX

Let \( \mathbf{y} = (r, \theta, \phi)^T \) be a Gaussian random vector of noise-corrupted polar measurements, such that

\[
\mathbf{y} = (r, \theta, \phi)^T = (r_0, \theta_0, \phi_0)^T + \mathbf{n}.
\]


John M. C. Clark received the B.A. degree in mathematics and engineering from Cambridge University, Cambridge, U.K., in 1961 and the Ph.D. degree from Imperial College London, London, U.K., in 1966. He was a Postdoctoral Fellow at Stanford University, Stanford, CA, from 1966 to 1967. He was employed in the Mathematics Division, Bell Laboratories, Murray Hill, NJ, from 1967 to 1968. Since 1969, he has worked at Imperial College, where he is now Emeritus Reader in Stochastic Systems. He has held visiting positions at numerous institutions, including the Universities of California at Berkeley and Davis, Harvard, MIT, and the Australian National University. His main fields of interest are stochastic modeling and estimation, and he has made contributions in nonlinear filtering, the robustness analysis of nonlinear filters, the computational aspects of stochastic processes, and other areas. Recently he has also worked in the field of target tracking, with emphasis on new, computationally efficient filters for difficult tracking problems.

Panagiotis-Aristidis Kountouriotis received the M.Eng. (with first class honors) in electrical engineering from Imperial College London, London, U.K., in 2005 where he is currently pursuing the Ph.D. degree in the Control and Power Group, Department of Electrical and Electronic Engineering. His research interests focus on analytically and computationally efficient techniques for nonlinear filtering and tracking problems.

Richard B. Vinter (F’05) received the B.Sc. degree (with first class honors) in engineering science from Oxford University, Oxford, U.K., the Ph.D. degree in control engineering and the Sc.D. degree in mathematics from Cambridge University, Cambridge, U.K. As a recipient of a Harkness Fellowship, he was a Postdoctoral Researcher at the Electronic Systems Laboratory, Massachusetts Institute of Technology, Cambridge, for two years. Since 1974 he has been on the academic staff of Imperial College, London, where he is now Professor of Control Theory. He is Head of the Control and Power Group in the Department of Electrical and Electronic Engineering. He has published over 140 scientific papers and two text books. His research has centered on nonlinear control, optimization, differential games, tracking, and estimation.