Bearings-Only Tracking of Manoeuvring Targets Using Particle Filters

M. Sanjeev Arulampalam

Maritime Operations Division, Defence Science and Technology Organisation (DSTO), Edinburgh, South Australia 5111, Australia
Email: sanjeev.arulampalam@dsto.defence.gov.au

B. Ristic

Intelligence, Surveillance & Reconnaissance Division, Defence Science and Technology Organisation (DSTO), Edinburgh,
South Australia 5111, Australia
Email: branko.ristic@dsto.defence.gov.au

N. Gordon

Intelligence, Surveillance & Reconnaissance Division, Defence Science and Technology Organisation (DSTO), Edinburgh,
South Australia 5111, Australia
Email: neil.gordon@dsto.defence.gov.au

T. Mansell

Maritime Operations Division, Defence Science and Technology Organisation (DSTO), Edinburgh, South Australia 5111, Australia
Email: todd.mansell@dsto.defence.gov.au

We investigate the problem of bearings-only tracking of manoeuvring targets using particle filters (PFs). Three different (PFs) are proposed for this problem which is formulated as a multiple model tracking problem in a jump Markov system (JMS) framework. The proposed filters are (i) multiple model PF (MMPF), (ii) auxiliary MMPF (AUX-MMPF), and (iii) jump Markov system PF (JMS-PF). The performance of these filters is compared with that of standard interacting multiple model (IMM)-based trackers such as IMM-EKF and IMM-UKF for three separate cases: (i) single-sensor case, (ii) multisensor case, and (iii) tracking with hard constraints. A conservative CRLB applicable for this problem is also derived and compared with the RMS error performance of the filters. The results confirm the superiority of the PFs for this difficult nonlinear tracking problem.

Keywords and phrases: bearings-only tracking, manoeuvring target tracking, particle filter.

1. INTRODUCTION

The problem of bearings-only tracking arises in a variety of important practical applications. Typical examples are submarine tracking (using a passive sonar) or aircraft surveillance (using a radar in a passive mode or an electronic warfare device) [1, 2, 3]. The problem is sometimes referred to as target motion analysis (TMA), and its objective is to track the kinematics (position and velocity) of a moving target using noise-corrupted bearing measurements. In the case of autonomous TMA (single observer only), which is the focus of a major part of this paper, the observation platform needs to manoeuvre in order to estimate the target range [1, 3]. This need for ownship manoeuvre and its impact on target state observability have been explored extensively in [4, 5].

Most researchers in the field of bearings-only tracking have concentrated on tracking a nonmanoeuvring target. Due to inherent nonlinearity and observability issues, it is difficult to construct a finite-dimensional optimal Bayesian filter even for this relatively simple problem. As for the bearings-only tracking of a manoeuvring target, the problem is much more difficult and so far, very limited research has been published in the open literature. For example, interacting multiple model (IMM)-based trackers were proposed in [6, 7] for this problem. These algorithms employ a constant velocity (CV) model along with manoeuvre models to capture the dynamic behaviour of a manoeuvring target scenario. Le Cadre and Tremois [8] modelled the manoeuvring target using the CV model with process noise and developed a tracking filter in the hidden Markov model framework.

This paper presents the application of particle filters (PFs) [9, 10, 11] for bearings-only tracking of manoeuvring targets and compares its performance with traditional IMM-based filters. This work builds on the investigation carried out by the authors in [12] for the same problem. The additional features considered in this paper include (a) use of different manoeuvre models, (b) two additional PFs, and (c) tracking with hard constraints. The error performance of the developed filters is analysed by Monte Carlo (MC) simulations and compared to the theoretical Cramér-Rao lower bounds (CRLBs). Essentially, the manoeuvring target problem is formulated in a jump Markov system (JMS) framework and these filters provide suboptimal solutions to the target state, given a sequence of bearing measurements and the particular JMS framework. In the JMS framework considered in this paper, the target motion is modelled by three switching dynamics models whose evolution follows a Markov chain. One of these models is the standard CV model while the other two correspond to coordinated turn (CT) models that capture the manoeuvre dynamics.

Three different PFs are proposed for this problem: (i) multiple model PF (MMPF), (ii) auxiliary MMPF (AUX-MMPF), and (iii) JMS-PF. The MMPF [12, 13] and AUX-MMPF [14] represent the target state and the mode at every time by a set of paired particles and construct the joint posterior density of the target state and mode, given all measurements. The JMS-PF, on the other hand, involves a hybrid scheme where it uses particles to represent only the distribution of the modes, while mode-conditioned state estimation is carried out using extended Kalman filters (EKFs).

The performance of the above algorithms is compared with two conventional schemes: (i) IMM-EKF and (ii) IMM-UKF. These filters represent the posterior density at each time epoch by a finite Gaussian mixture, and they merge and mix these Gaussian mixture components at every step to avoid the exponential growth in the number of mixture components. The IMM-EKF uses EKFs while the IMM-UKF utilises unscented Kalman filters (UKFs) [15] to compute the mode-conditioned state estimates.

In addition to the autonomous bearings-only tracking problem, two further cases are investigated in the paper: multisensor bearings-only tracking, and tracking with hard constraints. The multisensor bearings-only problem involves a slight modification to the original problem, where a second static sensor sends its target bearing measurements to the original platform. The problem of tracking with hard constraints involves the use of prior knowledge, such as speed constraints, to improve tracker performance.

The organisation of the paper is as follows. Section 2 presents the mathematical formulation for the bearings-only tracking problem for each of the three different cases investigated: (i) single-sensor case, (ii) multisensor case, and (iii) tracking with hard constraints. In Section 3 the relevant CRLBs are derived for all but case (iii) for which the analytic derivation is difficult (due to the non-Gaussian prior and process noise vectors). The tracking algorithms for each case are then presented in Section 4 followed by simulation results in Section 5.

2. PROBLEM FORMULATION

2.1. Single-sensor case

Conceptually, the basic problem in bearings-only tracking is to estimate the trajectory of a target (i.e., position and velocity) from noise-corrupted sensor bearing data. For the case of a single-sensor problem, these bearing data are obtained from a single-moving observer (ownship). The target state vector is

$$\mathbf{x}^t = \begin{bmatrix} x^t & y^t & \dot{x}^t & \dot{y}^t \end{bmatrix}^T, \tag{1}$$

where (x, y) and $(\dot{x}, \dot{y})$ are the position and velocity components, respectively. The ownship state vector $\mathbf{x}^o$ is similarly defined. We now introduce the relative state vector defined by

$$\mathbf{x} \triangleq \mathbf{x}^t - \mathbf{x}^o = \begin{bmatrix} x & y & \dot{x} & \dot{y} \end{bmatrix}^T \tag{2}$$

for which the discrete-time state equation will be written. The dynamics of a manoeuvring target is modelled by multiple switching regimes, also known as a JMS. We make the assumption that at any time in the observation period, the target motion obeys one of $s = 3$ dynamic behaviour models: (a) CV motion model, (b) clockwise CT model, and (c) anticlockwise CT model. Let $S \triangleq \{1, 2, 3\}$ denote the set of three models for the dynamic motion, and let r_k be the regime variable in effect in the interval $(k - 1, k]$, where k is the discrete-time index. Then, the target dynamics can be mathematically written as

$$\mathbf{x}_{k+1} = \mathbf{f}^{(r_{k+1})}(\mathbf{x}_k, \mathbf{x}_k^o, \mathbf{x}_{k+1}^o) + \mathbf{G}\mathbf{v}_k \quad (r_{k+1} \in S), \tag{3}$$

where

$$\mathbf{G} = \begin{bmatrix} \dfrac{T^2}{2} & 0 \\ 0 & \dfrac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix}, \tag{4}$$

T is the sampling time, and $\mathbf{v}_k$ is a 2×1 i.i.d. process noise vector with $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$. The process noise covariance matrix is chosen to be $\mathbf{Q} = \sigma_a^2 \mathbf{I}$, where $\mathbf{I}$ is the 2×2 identity matrix and σ_a is a process noise parameter. Note that $\mathbf{G}\mathbf{v}_k$ in (3) corresponds to a piecewise constant white acceleration noise model [16] which is adequate for the large sampling time chosen in our paper. The mode-conditioned transition function $\mathbf{f}^{(r_{k+1})}(\cdot, \cdot, \cdot)$ in (3) is given by

$$\mathbf{f}^{(r_{k+1})}(\mathbf{x}_k, \mathbf{x}_k^o, \mathbf{x}_{k+1}^o) = \mathbf{F}^{(r_{k+1})}(\mathbf{x}_k) \cdot (\mathbf{x}_k + \mathbf{x}_k^o) - \mathbf{x}_{k+1}^o. \tag{5}$$

Here $\mathbf{F}^{(r_{k+1})}(\cdot)$ is the transition matrix corresponding to mode r_{k+1}, which, for the particular problem of interest, can be specified as follows. $\mathbf{F}^{(1)}(\cdot)$ corresponds to CV motion and is thus given by the standard CV transition matrix:

$$\mathbf{F}^{(1)}(\mathbf{x}_k) = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{6}$$

The next two transition matrices correspond to CT transitions (clockwise and anticlockwise, respectively). These are given by

$$\mathbf{F}^{(j)}(\mathbf{x}_k) = \begin{bmatrix} 1 & 0 & \dfrac{\sin{(\Omega_k^{(j)}T)}}{\Omega_k^{(j)}} & -\dfrac{(1-\cos{(\Omega_k^{(j)}T)})}{\Omega_k^{(j)}} \\[2ex] 0 & 1 & \dfrac{(1-\cos{(\Omega_k^{(j)}T)})}{\Omega_k^{(j)}} & \dfrac{\sin{(\Omega_k^{(j)}T)}}{\Omega_k^{(j)}} \\[2ex] 0 & 0 & \cos{(\Omega_k^{(j)}T)} & -\sin{(\Omega_k^{(j)}T)} \\[2ex] 0 & 0 & \sin{(\Omega_k^{(j)}T)} & \cos{(\Omega_k^{(j)}T)} \end{bmatrix}, \\ j = 2,3, \tag{7}$$

where the mode-conditioned turning rates are

$$\Omega_k^{(2)} = \frac{a_m}{\sqrt{(\dot{x}_k + \dot{x}_k^o)^2 + (\dot{y}_k + \dot{y}_k^o)^2}}, \\ \Omega_k^{(3)} = \frac{-a_m}{\sqrt{(\dot{x}_k + \dot{x}_k^o)^2 + (\dot{y}_k + \dot{y}_k^o)^2}}. \tag{8}$$

Here $a_m > 0$ is a typical manoeuvre acceleration. Note that the turning rate is expressed as a function of target speed (a nonlinear function of the state vector $\mathbf{x}_k$) and thus models 2 and 3 are clearly nonlinear transitions.

We model the mode r_k in effect at $(k-1, k]$ by a time-homogeneous 3-state first-order Markov chain with known transition probability matrix $\mathbf{\Pi}$, whose elements are

$$\pi_{ij} \triangleq \mathbb{P}(r_k = j | r_{k-1} = i), \quad i, j \in S, \tag{9}$$

such that

$$\pi_{ij} \geq 0, \quad \sum_j \pi_{ij} = 1. \tag{10}$$

The initial probabilities are denoted by $\pi_i \triangleq \mathbb{P}(r_1 = i)$ for $i \in S$ and they satisfy

$$\pi_i \geq 0, \quad \sum_i \pi_i = 1. \tag{11}$$

The available measurement at time k is the angle from the observer's platform to the target, referenced (clockwise positive) to the y-axis and is given by

$$z_k = h(\mathbf{x}_k) + w_k, \tag{12}$$

where w_k is a zero-mean independent Gaussian noise with variance σ_θ^2 and

$$h(\mathbf{x}_k) = \arctan\left(\frac{x_k}{y_k}\right) \tag{13}$$

is the true bearing angle. The state variable of interest for estimation is the hybrid state vector $\mathbf{y}_k = (\mathbf{x}_k^T, r_k)^T$. Thus, given a set of measurements $\mathbf{Z}_k = \{z_1, \ldots, z_k\}$ and the jump-Markov model (3), the problem is to obtain estimates of the hybrid state vector $\mathbf{y}_k$. In particular, we are interested in computing the kinematic state estimate $\hat{\mathbf{x}}_{k|k} = \mathbb{E}[\mathbf{x}_k | \mathbf{Z}_k]$ and mode probabilities $\mathbb{P}(r_k = j | \mathbf{Z}_k)$, for every $j \in S$.

2.2. Multisensor case

Suppose there is a possibility of the ownship receiving additional (secondary) bearing measurements from a sensor located at (x_k^s, y_k^s) whose measurement errors are independent to that of the ownship sensor. For simplicity, we assume that (a) additional measurements are synchronous to the primary sensor measurements that (b) there is a zero transmission delay from the sensor at (x_k^s, y_k^s) to the ownship at (x_k^o, y_k^o). The secondary measurement can be modelled as

$$z_k' = h'(\mathbf{x}_k) + w_k', \tag{14}$$

where

$$h'(\mathbf{x}_k) = \arctan\left(\frac{x_k + x_k^o - x_k^s}{y_k + y_k^o - y_k^s}\right) \tag{15}$$

and w_k' is a zero-mean white Gaussian noise sequence with variance $\sigma_{\theta'}^2$. If the additional bearing measurement is not received at time k, we set $z_k' = \varnothing$. The bearings-only tracking problem for this multisensor case is then to estimate the state vector $\mathbf{x}_k$ given a sequence of measurements $\mathbf{Z}_k = \{z_1, z_1', \ldots, z_k, z_k'\}$.

2.3. Tracking with constraints

In many tracking problems, one has some hard constraints on the state vector which can be a valuable source of information in the estimation process. For example, we may know the minimum and maximum speeds of the target given by the constraint

$$s_{\min} \leq \sqrt{(\dot{x}_k + \dot{x}_k^o)^2 + (\dot{y}_k + \dot{y}_k^o)^2} \leq s_{\max}. \tag{16}$$

Suppose some constraint (such as the speed constraint) is imposed on the state vector, and denote the set of constrained state vectors by Ψ. Let the initial distribution of the state vector in the absence of constraints be $\mathbf{x}_0 \sim p(\mathbf{x}_0)$. With constraints, this initial distribution becomes a truncated density $\tilde{p}(\mathbf{x}_0)$, that is,

$$\tilde{p}(\mathbf{x}_0) = \begin{cases} \dfrac{p(\mathbf{x}_0)}{\int_{\mathbf{x}_0 \in \Psi} p(\mathbf{x}_0) d\mathbf{x}_0}, & \mathbf{x}_0 \in \Psi, \\ 0 & \text{otherwise.} \end{cases} \tag{17}$$

Likewise, the dynamics model should be modified in such a way that $\mathbf{x}_k$ is always constrained to Ψ. In the absence of hard constraints, suppose that the process noise $\mathbf{v}_k \sim g(\mathbf{v})$ is used in the filter. With constraints, the pdf of $\mathbf{v}_k$ becomes a state-dependent truncated density given by

$$\tilde{g}(\mathbf{v}; \mathbf{x}_k) = \begin{cases} \dfrac{g(\mathbf{v})}{\int_{\mathbf{v} \in G(\mathbf{x}_k)} g(\mathbf{v}) d\mathbf{v}}, & \mathbf{v} \in G(\mathbf{x}_k), \\ 0 & \text{otherwise,} \end{cases} \tag{18}$$

where $G(\mathbf{x}_k) = \{\mathbf{v} : \mathbf{x}_k \in \Psi\}$.

For the bearings-only tracking problem, we will consider hard constraints in target dynamics only. The measurement model remains the same as that for the unconstrained problem. Given a sequence of measurements $\mathbf{Z}_k$ and some constraint Ψ on the state vector, the aim is to obtain estimates of the state vector $\mathbf{x}_k$, that is,

$$\begin{aligned} \hat{\mathbf{x}}_{k|k} &= \mathbb{E}[\mathbf{x}_k \mid \mathbf{Z}_k, \Psi] \\ &= \int \mathbf{x}_k p(\mathbf{x}_k \mid \mathbf{Z}_k, \Psi) d\mathbf{x}_k, \end{aligned} \tag{19}$$

where $p(\mathbf{x}_k | \mathbf{Z}_k, \Psi)$ is the posterior density of the state, given the measurements and hard constraints.

3. CRAMÉR-RAO LOWER BOUNDS

We follow the approach taken in [12] for the development of a conservative CRLB for the manoeuvring target tracking problem. This bound assumes that the true model history of the target trajectory

$$\mathcal{H}_k^* = \{r_1^*, r_2^*, \dots, r_k^*\} \tag{20}$$

is known a priori. Then, a bound on the covariance of $\hat{\mathbf{x}}_k$ was shown to be

$$\begin{aligned} \mathbb{E}\Big[(\hat{\mathbf{x}}_k - \mathbf{x}_k)(\hat{\mathbf{x}}_k - \mathbf{x}_k)^T \Big] \\ \geq \mathbb{E}\Big[(\hat{\mathbf{x}}_k - \mathbf{x}_k)(\hat{\mathbf{x}}_k - \mathbf{x}_k)^T \mid \mathcal{H}_k^* \Big] \\ \geq [\mathbf{J}_k^*]^{-1}, \end{aligned} \tag{21}$$

where the mode-history-conditioned information matrix $\mathbf{J}_k^*$ is

$$\mathbf{J}_k^* = \mathbb{E}\Big[(\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_k, \mathbf{Z}_k)) (\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_k, \mathbf{Z}_k))^T \mid \mathcal{H}_k^* \Big]. \tag{22}$$

Following [17], a recursion for $\mathbf{J}_k^*$ can be written as

$$\mathbf{J}_{k+1}^* = \mathbf{D}_k^{22} - \mathbf{D}_k^{21} (\mathbf{J}_k^* + \mathbf{D}_k^{11})^{-1} \mathbf{D}_k^{12}, \tag{23}$$

where, in the case of additive Gaussian noise models applicable to our problem, matrices $\mathbf{D}_k^{ij}$ are given by

$$\begin{aligned} \mathbf{D}_k^{11} &= \mathbb{E}\Big\{ \big(\tilde{\mathbf{F}}_k^{(r_{k+1}^*)}\big)^T \mathbf{Q}_k^{-1} \tilde{\mathbf{F}}_k^{(r_{k+1}^*)} \Big\}, \\ \mathbf{D}_k^{12} &= -\mathbb{E}\Big\{ \big(\tilde{\mathbf{F}}_k^{(r_{k+1}^*)}\big)^T \Big\} \mathbf{Q}_k^{-1} = (\mathbf{D}_k^{21})^T, \\ \mathbf{D}_k^{22} &= \mathbf{Q}_k^{-1} + \mathbb{E}\{ \tilde{\mathbf{H}}_{k+1}^T \mathbf{R}_{k+1}^{-1} \tilde{\mathbf{H}}_{k+1} \}, \end{aligned} \tag{24}$$

where

$$\begin{aligned} \tilde{\mathbf{F}}_k^{(r_{k+1}^*)} &= \Big[\nabla_{\mathbf{x}_k} \big(\mathbf{f}^{(r_{k+1}^*)}(\mathbf{x}_k)\big)^T \Big]^T, \\ \tilde{\mathbf{H}}_{k+1} &= \Big[\nabla_{\mathbf{x}_{k+1}} h_{k+1}^T(\mathbf{x}_{k+1}) \Big]^T, \end{aligned} \tag{25}$$

$\mathbf{R}_{k+1} = \sigma_\beta^2 \mapsto \mathbf{R}_{k+1} = \sigma_\theta^2$ is the variance of the bearing measurements, and $\mathbf{Q}_k$ is the process noise covariance matrix. The Jacobian $\tilde{\mathbf{F}}_k^{(1)}$ for the case of $r_{k+1}^* = 1$ is simply the transition matrix given in (6). For $r_{k+1}^* \in \{2, 3\}$, the Jacobian $\tilde{\mathbf{F}}_k^{(r_{k+1}^*)}$ can be computed as

$$\tilde{\mathbf{F}}_k^{(j)} = \begin{bmatrix} 1 & 0 & \dfrac{\partial f_1^{(j)}}{\partial \dot{x}_k} & \dfrac{\partial f_1^{(j)}}{\partial \dot{y}_k} \\ 0 & 1 & \dfrac{\partial f_2^{(j)}}{\partial \dot{x}_k} & \dfrac{\partial f_2^{(j)}}{\partial \dot{y}_k} \\ 0 & 0 & \dfrac{\partial f_3^{(j)}}{\partial \dot{x}_k} & \dfrac{\partial f_3^{(j)}}{\partial \dot{y}_k} \\ 0 & 0 & \dfrac{\partial f_4^{(j)}}{\partial \dot{x}_k} & \dfrac{\partial f_4^{(j)}}{\partial \dot{y}_k} \end{bmatrix}, \quad j = 2, 3, \tag{26}$$

where $f_i^{(j)}(\cdot)$ denotes the ith element of the dynamics model function $\mathbf{f}^{(j)}(\cdot)$. The detailed evaluation of $\tilde{\mathbf{F}}_k^{(j)}$ is given in the appendix.

Likewise, the Jacobian of the measurement function is given by

$$\tilde{\mathbf{H}}_{k+1} = \begin{bmatrix} \dfrac{\partial h}{\partial x_{k+1}} & \dfrac{\partial h}{\partial y_{k+1}} & \dfrac{\partial h}{\partial \dot{x}_{k+1}} & \dfrac{\partial h}{\partial \dot{y}_{k+1}} \end{bmatrix}, \tag{27}$$

where

$$\begin{aligned} \frac{\partial h}{\partial x_{k+1}} &= \frac{y_{k+1}}{x_{k+1}^2 + y_{k+1}^2}, & \frac{\partial h}{\partial y_{k+1}} &= \frac{-x_{k+1}}{x_{k+1}^2 + y_{k+1}^2}, \\ \frac{\partial h}{\partial \dot{x}_{k+1}} &= \frac{\partial h}{\partial \dot{y}_{k+1}} = 0. \end{aligned} \tag{28}$$

For the case of additional measurements from a secondary sensor, the only change required will be in the computation of $\mathbf{D}_k^{22}$. In particular, $\mathbf{R}_{k+1}$ and $\tilde{\mathbf{H}}_{k+1}$ will be replaced by $\mathbf{R}_{k+1}'$ and $\tilde{\mathbf{H}}_{k+1}'$, corresponding to the augmented measurement vector (z_{k+1}, z_{k+1}'). These are given by

$$\mathbf{R}_{k+1}' = \begin{bmatrix} \sigma_\theta^2 & 0 \\ 0 & \sigma_{\theta'}^2 \end{bmatrix},$$

$$\tilde{\mathbf{H}}_{k+1}' = \begin{bmatrix} \dfrac{\partial h}{\partial x_{k+1}} & \dfrac{\partial h}{\partial y_{k+1}} & \dfrac{\partial h}{\partial \dot{x}_{k+1}} & \dfrac{\partial h}{\partial \dot{y}_{k+1}} \\ \dfrac{\partial h'}{\partial x_{k+1}} & \dfrac{\partial h'}{\partial y_{k+1}} & \dfrac{\partial h'}{\partial \dot{x}_{k+1}} & \dfrac{\partial h'}{\partial \dot{y}_{k+1}} \end{bmatrix}, \quad (29)$$

where $\sigma_{\beta'}^2 \mapsto \sigma_{\theta'}^2$ is the noise variance of the secondary sensor, and the first row of $\tilde{\mathbf{H}}_{k+1}'$ is identical to (27). The elements of the second row of $\tilde{\mathbf{H}}_{k+1}'$ are given by

$$\frac{\partial h'}{\partial x_{k+1}} = \frac{y_{k+1} + y_{k+1}^{01} - y_{k+1}^{02}}{\left(x_{k+1} + x_{k+1}^{01} - x_{k+1}^{02}\right)^2 + \left(y_{k+1} + y_{k+1}^{01} - y_{k+1}^{02}\right)^2},$$

$$\frac{\partial h'}{\partial y_{k+1}} = \frac{-\left(x_{k+1} + x_{k+1}^{01} - x_{k+1}^{01}\right)}{\left(x_{k+1} + x_{k+1}^{01} - x_{k+1}^{02}\right)^2 + \left(y_{k+1} + y_{k+1}^{01} - y_{k+1}^{02}\right)^2},$$

$$\frac{\partial h'}{\partial \dot{x}_{k+1}} = \frac{\partial h'}{\partial \dot{y}_{k+1}} = 0. \quad (30)$$

The simulation experiments for this problem will be carried out on fixed trajectories. This means that for the corresponding CRLBs, the expectation operators in (24) vanish and the required Jacobians will be computed at the true trajectories. The recursion (23) is initialised by

$$\mathbf{J}_1^* = \mathbf{P}_1^{-1}, \quad (31)$$

where $\mathbf{P}_1$ is the initial covariance matrix of the state estimate. This can be computed using the expression (38), where we replace the measurement θ_1 by the true initial bearing.

4. TRACKING ALGORITHMS

This section describes five recursive algorithms designed for tracking a manoeuvring target using bearings-only measurements. Two of the algorithms are IMM-based algorithms and the other three are PF-based schemes. The algorithms considered are (i) IMM-EKF, (ii) IMM-UKF, (iii) MMPF, (iv) AUX-MMPF, and (v) JMS-PF. All five algorithms are applicable to both single-sensor and multisensor tracking problems, formulated in Section 2.

Sections 4.1, 4.2, 4.3, 4.4, and 4.5 will present the elements of the five tracking algorithms to be investigated. The IMM-based trackers will not be presented in detail; the interested reader is referred to [7, 12, 16] for a detailed exposition of these trackers. Section 4.6 presents the required methodology for the multisensor case while Section 4.7 discusses the modifications required in the PF-based trackers for tracking with hard constraints.

4.1. IMM-EKF algorithm

The IMM-EKF algorithm is an EKF-based routine that has been utilised for manoeuvring target tracking problems formulated in a JMS framework [7, 12]. The basic idea is that, for each dynamic model of the JMS, a separate EKF is used, and the filter outputs are weighted according to the mode probabilities to give the state estimate and covariance. At each time index, the target state pdf is characterised by a finite Gaussian mixture which is then propagated to the next time index. Ideally, this propagation results in an s-fold increase in the number of mixture components, where s is the number of modes in the JMS. However, the IMM-EKF algorithm avoids this growth by merging groups of components using mixture probabilities. The details of the IMM-EKF algorithm can be found in [7], where slightly different motion models to the one used here were proposed.

The sources of approximation in the IMM-EKF algorithm are twofold. First, the EKF approximates nonlinear transformations by linear transformations at some operating point. If the nonlinearity is severe or if the operating point is not chosen properly, the resultant approximation can be poor, leading to filter divergence. Second, the IMM approximates the exponentially growing Gaussian mixture with a finite Gaussian mixture. The above two approximations can cause filter instability in certain scenarios.

Next, we provide details of the filter initialisation for the EKF routines used in this algorithm.

4.1.1. Filter initialisation

Suppose the initial prior range is $r \sim \mathcal{N}(\bar{r}, \sigma_r^2)$, where $\bar{r}$ and σ_r^2 are the mean and variance of the initial range. Then, given the first bearing measurement θ_1, the position components of the relative target state vector is initialised according to standard procedure [12], that is,

$$x_1 = \bar{r} \sin \theta_1, \qquad y_1 = \bar{r} \cos \theta_1, \quad (32)$$

with covariance

$$\mathbf{P}^{xy} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{yx} & \sigma_y^2 \end{bmatrix}, \quad (33)$$

$$\sigma_x^2 = \bar{r}^2 \sigma_\theta^2 \cos^2 \theta_1 + \sigma_r^2 \sin^2 \theta_1, \quad (34)$$

$$\sigma_y^2 = \bar{r}^2 \sigma_\theta^2 \sin^2 \theta_1 + \sigma_r^2 \cos^2 \theta_1, \quad (35)$$

$$\sigma_{xy} = \sigma_{yx} = (\sigma_r^2 - \bar{r}^2 \sigma_\theta^2) \sin \theta_1 \cos \theta_1, \quad (36)$$

where σ_θ is the bearing-measurement standard deviation. We adopt a similar procedure to initialise the velocity components. The overall relative target state vector can thus be initialised as follows. Suppose we have some prior knowledge of the target speed and course given by $s \sim \mathcal{N}(\bar{s}, \sigma_s^2)$ and $c \sim \mathcal{N}(\bar{c}, \sigma_c^2)$, respectively. Then, the overall relative target state vector is initialised as

$$\hat{\mathbf{x}}_1 = \begin{bmatrix} \bar{r} \sin \theta_1 \\ \bar{r} \cos \theta_1 \\ \bar{s} \sin(\bar{c}) - \dot{x}_1^o \\ \bar{s} \cos(\bar{c}) - \dot{y}_1^o \end{bmatrix}, \quad (37)$$

where $(\dot{x}_1^o, \dot{y}_1^o)$ is the velocity of the ownship at time index 1. The corresponding initial covariance matrix is given by

$$\mathbf{P}_1 = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & 0 & 0 \\ \sigma_{yx} & \sigma_y^2 & 0 & 0 \\ 0 & 0 & \sigma_{\dot{x}}^2 & \sigma_{\dot{x}\dot{y}} \\ 0 & 0 & \sigma_{\dot{y}\dot{x}} & \sigma_{\dot{y}}^2 \end{bmatrix}, \tag{38}$$

where $\sigma_x^2, \sigma_{xy}, \sigma_{yx}, \sigma_y^2$ are given by (34)–(36), and

$$\sigma_{\dot{x}}^2 = \bar{s}^2 \sigma_c^2 \cos^2(\bar{c}) + \sigma_s^2 \sin^2(\bar{c}),$$
$$\sigma_{\dot{y}}^2 = \bar{s}^2 \sigma_c^2 \sin^2(\bar{c}) + \sigma_s^2 \cos^2(\bar{c}), \tag{39}$$
$$\sigma_{\dot{x}\dot{y}} = \sigma_{\dot{y}\dot{x}} = (\sigma_s^2 - \bar{s}^2 \sigma_c^2) \sin(\bar{c}) \cos(\bar{c}).$$

4.2. IMM-UKF algorithm

This algorithm is similar to the IMM-EKF with the main difference being that the model-matched EKFs are replaced by model-matched UKFs [15]. The UKF for model 1 uses the unscented transform (UT) only for the filter update (because only the measurement equation is non-linear). The UKFs for models 2 and 3 use the UT for both the prediction and the update stage of the filter. The IMM-UKF is initialised in a similar manner to that of the IMM-EKF.

4.3. MMPF

The MMPF [12, 13] has been used to solve various manoeuvring target tracking problems. Here we briefly review the basics of this filter.

The MMPF estimates the posterior density $p(\mathbf{y}_k|\mathbf{Z}_k)$, where $\mathbf{y}_k = [\mathbf{x}_k^T, r_k]^T$ is the augmented (hybrid) state vector. In order to recursively compute the PF estimates, the MC representation of $p(\mathbf{y}_k|\mathbf{Z}_k)$ has to be propagated in time. Let $\{\mathbf{y}_{k-1}^i, w_{k-1}^i\}_{i=1}^N$ denote a random measure that characterises the posterior pdf $p(\mathbf{y}_{k-1}|\mathbf{Z}_{k-1})$, where $\mathbf{y}_{k-1}^i$, $i = 1, \ldots, N$, is a set of support points with associated weights w_{k-1}^i, $i = 1, \ldots, N$. Then, the posterior density of the augmented state at $k-1$ can be approximated as

$$p(\mathbf{y}_{k-1}|\mathbf{Z}_{k-1}) \approx \sum_{i=1}^N w_{k-1}^i \delta(\mathbf{y}_{k-1} - \mathbf{y}_{k-1}^i), \tag{40}$$

where $\delta(\cdot)$ is the Dirac delta measure. Next, the posterior pdf at k can be written as

$$p(\mathbf{y}_k|\mathbf{Z}_k) \propto p(z_k|\mathbf{y}_k) \int p(\mathbf{y}_k|\mathbf{y}_{k-1}) p(\mathbf{y}_{k-1}|\mathbf{Z}_{k-1}) d\mathbf{y}_{k-1}$$
$$\approx p(z_k|\mathbf{y}_k) \sum_{i=1}^N w_{k-1}^i p(\mathbf{y}_k|\mathbf{y}_{k-1}^i), \tag{41}$$

where approximation (40) was used in (41). Now, to represent the pdf $p(\mathbf{y}_k|\mathbf{Z}_k)$ using particles, we employ the *importance sampling* method [9]. By choosing the importance density to be $p(\mathbf{y}_k|\mathbf{y}_{k-1})$, one can draw samples $\mathbf{y}_k^{*i} \sim p(\mathbf{y}_k|\mathbf{y}_{k-1}^i)$, $i = 1, \ldots, N$. To draw a sample from $p(\mathbf{y}_k|\mathbf{y}_{k-1}^i)$, we first draw a sample from $p(r_k|r_{k-1}^i)$ which is a discrete probability mass function given by the ith row of the Markov chain transition probability matrix. That is, we choose $r_k^{*i} \sim p(r_k|r_{k-1}^i)$ such that

$$\mathbb{P}(r_k^{*i} = j) = \pi_{ij}. \tag{42}$$

Next, given the mode r_k^{*i}, one can easily sample $\mathbf{x}_k^{*i} \sim p(\mathbf{x}_k|\mathbf{x}_{k-1}^i, r_k^i)$ by generating process noise sample $\mathbf{v}_{k-1}^i \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ and propagating $\mathbf{x}_{k-1}^i, r_k^{*i}$, and $\mathbf{v}_{k-1}^i$ through the dynamics model (3). This gives us the sample $\{\mathbf{y}_k^{*i} = [(\mathbf{x}_k^{*i})^T, r_k^{*i}]^T\}_{i=1}^N$ which can be used to approximate the posterior pdf $p(\mathbf{y}_k|\mathbf{Z}_k)$ as

$$p(\mathbf{y}_k|\mathbf{Z}_k) \approx \sum_{i=1}^N w_k^i \delta(\mathbf{y}_k - \mathbf{y}_k^{*i}), \tag{43}$$

where

$$w_k^i \propto w_{k-1}^i p(z_k|\mathbf{y}_k^{*i}). \tag{44}$$

Note that $p(z_k|\mathbf{y}_k^{*i}) = p(z_k|\mathbf{x}_k^{*i})$ which can be computed using the measurement equation (12). This completes the description of a single recursion of the MMPF. The filter is initialised by generating N samples $\{\mathbf{x}_1^i\}_{i=1}^N$ from the initial density $\mathcal{N}(\hat{\mathbf{x}}_1, \mathbf{P}_1)$, where $\hat{\mathbf{x}}_1$ and $\mathbf{P}_1$ were specified in (37) and (38), respectively.

A common problem with PFs is the degeneracy phenomenon, where, after a few iterations, all but one particle will have negligible weight. A measure of degeneracy is the effective sample size N_{eff} which can be empirically evaluated as

$$\hat{N}_{\text{eff}} = \frac{1}{\sum_{i=1}^N w_k^{i\,2}}. \tag{45}$$

The usual approach to overcoming the degeneracy problem is to introduce resampling whenever $\hat{N}_{\text{eff}} < N_{\text{thr}}$ for some threshold N_{thr}. The resampling step involves generating a new set $\{\mathbf{y}_k^i\}_{i=1}^N$ by sampling with replacement N times from the set $\{\mathbf{y}_k^{*i}\}_{i=1}^N$ such that

$$\mathbb{P}(\mathbf{y}_k^i = \mathbf{y}_k^{*j}) = w_k^j. \tag{46}$$

4.4. AUX-MMPF

The AUX-MMPF [14] focuses on the characterisation of pdf $p(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$, where i refers to the ith particle at $k-1$. This density is marginalised to obtain a representation of $p(\mathbf{x}_k|\mathbf{Z}_k)$.

A proportionality for the joint probability density $p(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$ can be written using Bayes' rule as

$$p(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$$
$$\propto p(z_k|\mathbf{x}_k) p(\mathbf{x}_k, i, r_k|\mathbf{Z}_{k-1})$$
$$= p(z_k|\mathbf{x}_k) p(\mathbf{x}_k|r_k, i, \mathbf{Z}_{k-1}) p(r_k|i, \mathbf{Z}_{k-1}) p(i|\mathbf{Z}_{k-1})$$
$$= p(z_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{x}_{k-1}^i, r_k) p(r_k|r_{k-1}^i) w_{k-1}^i, \tag{47}$$

where $p(r_k|r_{k-1})$ is an element of the transition probability matrix $\mathbf{\Pi}$ defined by (9). To sample directly from $p(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$ as given by (47) is not practical. Hence, we again use importance sampling [9, 14] to first obtain a sample from a density which closely resembles (47), and then weight the samples appropriately to produce an MC representation of $p(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$. This can be done by introducing the function $q(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$ with proportionality

$$q(\mathbf{x}_k, i, r_k|\mathbf{Z}_k) \propto p(z_k|\mu_k^i(r_k)) p(\mathbf{x}_k|\mathbf{x}_{k-1}^i, r_k) p(r_k|r_{k-1}^i) w_{k-1}^i, \tag{48}$$

where

$$\begin{aligned} \mu_k^i(r_k) &= \mathbb{E}\{\mathbf{x}_k|\mathbf{x}_{k-1}^i, r_k\} \\ &= \mathbf{f}^{(r_k)}(\mathbf{x}_{k-1}^i, \mathbf{x}_{k-1}^o, \mathbf{x}_k^o). \end{aligned} \tag{49}$$

Importance density $q(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$ differs from (47) only in the first factor. Now, we can write $q(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$ as

$$q(\mathbf{x}_k, i, r_k|\mathbf{Z}_k) = q(i, r_k|\mathbf{Z}_k) g(\mathbf{x}_k|i, r_k, \mathbf{Z}_k) \tag{50}$$

and define

$$g(\mathbf{x}_k|i, r_k, \mathbf{Z}_k) \triangleq p(\mathbf{x}_k|\mathbf{x}_{k-1}^i, r_k). \tag{51}$$

In order to obtain a sample from the density $q(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$, we first integrate (48) with respect to $\mathbf{x}_k$ to get an expression for $q(i, r_k|\mathbf{Z}_k)$,

$$q(i, r_k|\mathbf{Z}_k) \propto p(z_k|\mu_k^i(r_k)) p(r_k|r_{k-1}^i) w_{k-1}^i. \tag{52}$$

A random sample can now be obtained from the density $q(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$ as follows. First, a sample $\{i^j, r_k^j\}_{j=1}^N$ is drawn from the discrete distribution $q(i, r_k|\mathbf{Z}_k)$ given by (52). This can be done by splitting each of the N particles at $k-1$ into s groups (s is the number of possible modes), each corresponding to a particular mode. Each of the sN particles is assigned a weight proportional to (52), and N points $\{i^j, r_k^j\}_{j=1}^N$ are then sampled from this discrete distribution. From (50) and (51), it is seen that the samples $\{\mathbf{x}_k^j\}_{j=1}^N$ from the joint density $q(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$ can now be generated from $p(\mathbf{x}_k|\mathbf{x}_{k-1}^{i^j}, r_k^j)$. The resultant triplet sample $\{\mathbf{x}_k^j, i^j, r_k^j\}_{j=1}^N$ is a random sample from the density $q(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$. To use these samples to characterise the density $p(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$, we attach the weights w_k^j to each particle, where w_k^j is a ratio of (48) and (47), evaluated at $\{\mathbf{x}_k^j, i^j, r_k^j\}$, that is,

$$\begin{aligned} w_k^j &= \frac{p(z_k|\mathbf{x}_k^j) p(\mathbf{x}_k^j|\mathbf{x}_{k-1}^{i^j}, r_k^j) p(r_k^j|r_{k-1}^{i^j}) w_{k-1}^{i^j}}{p(z_k|\mu_k^{i^j}(r_k)) p(\mathbf{x}_k^j|\mathbf{x}_{k-1}^{i^j}, r_k^j) p(r_k^j|r_{k-1}^{i^j}) w_{k-1}^{i^j}} \\ &= \frac{p(z_k|\mathbf{x}_k^j)}{p(z_k|\mu_k^{i^j}(r_k))}. \end{aligned} \tag{53}$$

By defining the augmented vector $\mathbf{y}_k \triangleq (\mathbf{x}_k^T, i, r_k)^T$, we can write down an MC representation of the pdf $p(\mathbf{x}_k, i, r_k|\mathbf{Z}_k)$ as

$$p(\mathbf{x}_k, i, r_k|\mathbf{Z}_k) = p(\mathbf{y}_k) \approx \sum_{j=1}^N w_k^j \delta(\mathbf{y}_k - \mathbf{y}_k^j). \tag{54}$$

Observe that by omitting the $\{i^j, r_k^j\}$ components in the triplet sample, we have a representation of the marginalised density $p(\mathbf{x}_k|\mathbf{Z}_k)$, that is,

$$p(\mathbf{x}_k|\mathbf{Z}_k) \approx \sum_{j=1}^N w_k^j \delta(\mathbf{x}_k - \mathbf{x}_k^j). \tag{55}$$

The AUX-MMPF is initialised according to the same procedure as for MMPF.

4.5. JMS-PF

The JMS-PF is based on the jump Markov linear system (JMLS) PF proposed in [18, 19] for a JMLS. Let

$$\begin{aligned} \mathbf{X}_k &= \{\mathbf{x}_1, \dots, \mathbf{x}_k\}, \\ R_k &= \{r_1, \dots, r_k\} \end{aligned} \tag{56}$$

denote the sequences of states and modes up to time index k. Standard particle filtering techniques focused on the estimation of the pdf of the state vector $\mathbf{x}_k$. However, in the JMS-PF, we place emphasis on the estimation of the pdf of the mode sequence R_k, given measurements $\mathbf{Z}_k = \{z_1, \dots, z_k\}$. The density $p(\mathbf{X}_k, R_k|\mathbf{Z}_k)$ can be factorised into

$$p(\mathbf{X}_k, R_k|\mathbf{Z}_k) = p(\mathbf{X}_k|R_k, \mathbf{Z}_k) p(R_k|\mathbf{Z}_k). \tag{57}$$

Given a specific mode sequence R_k and measurements $\mathbf{Z}_k$, the first term on the right-hand side of (57), $p(\mathbf{X}_k|R_k, \mathbf{Z}_k)$, can easily be estimated using an EKF or some other nonlinear filter. Therefore, we focus our attention on $p(R_k|\mathbf{Z}_k)$; for estimation of this density, we propose to use a PF.

Using Bayes' rule, we note that

$$p(R_k|\mathbf{Z}_k) = \frac{p(z_k|\mathbf{Z}_{k-1}, R_k) p(r_k|r_{k-1})}{p(z_k|\mathbf{Z}_{k-1})} p(R_{k-1}|\mathbf{Z}_{k-1}). \tag{58}$$

Equation (58) provides a useful recursion for the estimation of $p(R_k|\mathbf{Z}_k)$ using a PF. We describe a general recursive algorithm which generates N particles $\{R_k^i\}_{i=1}^N$ at time k which characterises the pdf $p(R_k|\mathbf{Z}_k)$. The algorithm requires the introduction of an importance function $q(r_k|\mathbf{Z}_k, R_{k-1})$. Suppose at time $k-1$, one has a set of particles $\{R_{k-1}^i\}_{i=1}^N$ that characterises the pdf $p(R_{k-1}|\mathbf{Z}_{k-1})$. That is,

$$p(R_{k-1}|\mathbf{Z}_{k-1}) \approx \frac{1}{N} \sum_{i=1}^N \delta(R_{k-1} - R_{k-1}^i). \tag{59}$$

Now draw N samples $r_k^i \sim q(r_k | \mathbf{Z}_k, R_{k-1}^i)$. Then, from (58) and the principle of importance sampling, one can write

$$p(R_k | \mathbf{Z}_k) \approx \sum_{i=1}^{N} w_k^i \delta(R_k - R_k^i), \qquad (60)$$

where $R_k^i \equiv \{R_{k-1}^i, r_k^i\}$ and the weight

$$w_k^i \propto \frac{p(z_k | \mathbf{Z}_{k-1}, R_k^i) \, p(r_k^i | r_{k-1}^i)}{q(r_k^i | \mathbf{Z}_k, R_{k-1}^i)}. \qquad (61)$$

From (60), we note that one can perform resampling (if required) to obtain an approximate i.i.d. sample from $p(R_k | \mathbf{Z}_k)$. The recursion can be initialised according to the specified initial state distribution of the Markov chain, $\pi_i = \mathbb{P}(r_1 = i)$.

How do we choose the importance density $q(r_k | \mathbf{Z}_k, R_{k-1})$? A sensible selection criterion is to choose a proposal that minimises the variance of the importance weights at time k, given R_{k-1} and $\mathbf{Z}_k$. According to this strategy, it was shown in [18] that the optimal importance density is $p(r_k | \mathbf{Z}_k, R_{k-1}^i)$. Now, it is easy to see that this density satisfies

$$p(r_k | \mathbf{Z}_k, R_{k-1}^i) = \frac{p(z_k | \mathbf{Z}_{k-1}, R_{k-1}^i, r_k) \, p(r_k | r_{k-1}^i)}{p(z_k | \mathbf{Z}_{k-1}, R_{k-1}^i)}. \qquad (62)$$

Note that $p(r_k | \mathbf{Z}_k, R_{k-1}^i)$ is proportional to the numerator of (62) as the denominator is independent of r_k. Also, the term $p(r_k | r_{k-1})$ is simply the Markov chain transition probability (specified by the transition probability matrix $\mathbf{\Pi}$). The term $p(z_k | \mathbf{Z}_{k-1}, R_k)$, which features in the numerator of (62), can be approximated by one-step-ahead EKF outputs, that is, we can write

$$p(z_k | \mathbf{Z}_{k-1}, R_k) \approx \mathcal{N}(\nu_k(R_k, \mathbf{Z}_{k-1}); 0, \mathbf{S}_k(R_k, \mathbf{Z}_{k-1})), \quad (63)$$

where $\nu_k(\cdot, \cdot)$ and $\mathbf{S}_k(\cdot, \cdot)$ are the mode-history-conditioned innovation and its covariance, respectively. Thus, $p(r_k | r_{k-1})$ and (63) allow the computation of the optimal importance density.

Using (62) as the importance density $q(\cdot | \cdot, \cdot)$ in (61), we find that the weight

$$w_k^i \propto p(z_k | \mathbf{Z}_{k-1}, R_{k-1}^i). \qquad (64)$$

Since $r_k \in \{1, \ldots, s\}$, the importance weights given above can be computed as

$$\begin{aligned} w_k^i &\propto p(z_k | \mathbf{Z}_{k-1}, R_{k-1}^i) \\ &= \sum_{j=1}^{s} p(z_k | \mathbf{Z}_{k-1}, R_{k-1}^i, r_k = j) \, p(r_k = j | r_{k-1}^i). \end{aligned} \qquad (65)$$

Note that the computation of the importance weights in (65) requires s one-step-ahead EKF innovations and their covariances.

This completes the description of the PF for estimation of the Markov chain distribution $p(R_k | \mathbf{Z}_k)$. As mentioned earlier, given a particular mode sequence, the state estimates are easily obtained using a standard EKF.

4.6. Methodology for the multisensor case

The methodology for the multisensor case is similar to that of the single-sensor case. The two main points to note for this case are that (a) the secondary measurement is processed in a sequential manner assuming a zero time delay between the primary and secondary measurements and (b) for the processing of the secondary measurement, the measurement function (15) is used in place of (13) for the computation of the necessary quantities such as Jacobians, predicted measurements, and weights.

4.7. Modifications for tracking with hard constraints

The problem of bearings-only tracking with hard constraints was described in Section 2.3. Recall that for the constraint $\mathbf{x}_k \in \Psi$, the state estimate is given by the mean of the posterior density $p(\mathbf{x}_k | \mathbf{Z}_k, \Psi)$. This density cannot be easily constructed by standard Kalman-filter-based techniques. However, since PFs make no restrictions on the prior density or the distributions of the process and measurement noise vectors, it turns out that $p(\mathbf{x}_k | \mathbf{Z}_k, \Psi)$ can be constructed using PFs. The only modifications required in the PFs for the case of constraint $\mathbf{x}_k \in \Psi$ are as follows:

(i) the prior distribution needs to be $\tilde{p}(\mathbf{x})$ defined in (17) and the filter needs to be able to sample from this density;

(ii) in the prediction step, samples are drawn from the constrained process noise density $\tilde{g}(\mathbf{v}; \mathbf{x}_k)$ instead of the standard process noise pdf.

Both changes require the ability to sample from a truncated density. A simple method to sample from a generic truncated density $\tilde{t}(\mathbf{x})$ defined by

$$\tilde{t}(\mathbf{x}) = \begin{cases} \dfrac{t(\mathbf{x})}{\int_{\mathbf{x} \in \Psi} t(\mathbf{x}) d\mathbf{x}}, & \mathbf{x} \in \Psi, \\ 0 & \text{otherwise} \end{cases} \qquad (66)$$

is as follows. Suppose we can easily sample from $t(\mathbf{x})$. Then, to draw $\mathbf{x} \sim \tilde{t}(\mathbf{x})$, we can use rejection sampling from $t(\mathbf{x})$ until the condition $\mathbf{x} \in \Psi$ is satisfied. The resulting sample is then distributed according to $\tilde{t}(\mathbf{x})$. This simple technique will be adopted in the modifications required in the PF for the constrained problem.[1] With the above modifications, the PF leads to a cloud of particles that characterise the posterior density $p(\mathbf{x}_k | \mathbf{Z}_k, \Psi)$ from which the state estimate $\hat{\mathbf{x}}_{k|k}$ and its covariance $\mathbf{P}_{k|k}$ can be obtained.

---

[1]This rejection sampling scheme can potentially be inefficient. For more efficient schemes to sample from truncated densities, see [20].

5. SIMULATION RESULTS

In this section, we present a performance comparison of the various tracking algorithms described in the previous section. The comparison will be based on a set of 100MC simulations and where possible, the CRLB will be used to indicate the best possible performance that one can expect for a given scenario and a set of parameters. Before proceeding, we give a description of the four performance metrics that will be used in this analysis: (i) RMS position error, (ii) efficiency η, (iii) root time-averaged mean square (RTAMS) error, and (iv) number of divergent tracks.

To define each of the above performance metrics, let (x_k^i, y_k^i) and $(\hat{x}_k^i, \hat{y}_k^i)$ denote the true and estimated target positions at time k at the ith MC run. Suppose M of such MC runs are carried out. Then, the RMS position error at k can be computed as

$$\text{RMS}_k = \sqrt{\frac{1}{M} \sum_{i=1}^{M} (\hat{x}_k^i - x_k^i)^2 + (\hat{y}_k^i - y_k^i)^2}. \quad (67)$$

Now, if $\mathbf{J}_k^{-1}[i, j]$ denotes the ijth element of the inverse information matrix for the problem at hand, then the corresponding CRLB for the metric (67) can be written as

$$\text{CRLB}\left(\text{RMS}_k\right) = \sqrt{\mathbf{J}_k^{-1}[1, 1] + \mathbf{J}_k^{-1}[2, 2]}. \quad (68)$$

The second metric stated above is the efficiency parameter η defined as

$$\eta_k \triangleq \frac{\text{CRLB}\left(\text{RMS}_k\right)}{\text{RMS}_k} \times 100\% \quad (69)$$

which indicates "closeness" to CRLB. Thus, $\eta_k = 100\%$ implies an efficient estimator that achieves the CRLB exactly.

For a particular scenario and parameters, the overall performance of a filter is evaluated using the third metric which is the RTAMS error. This is defined as

$$\text{RTAMS} = \sqrt{\frac{1}{(t_{\max} - \ell)M} \sum_{k=\ell+1}^{t_{\max}} \sum_{i=1}^{M} (\hat{x}_k^i - x_k^i)^2 + (\hat{y}_k^i - y_k^i)^2}, \quad (70)$$

where $t_{\max}$ is the total number of observations (or time epochs) and ℓ is a time index after which the averaging is carried out. Typically ℓ is chosen to coincide with the end of the first ownship manoeuvre.

The final metric stated above is the number of divergent tracks. A track is declared divergent if its estimated position error at any time index exceeds a threshold which is set to be 20 km in our simulations. It must be noted that the first three metrics described above are computed only on nondivergent tracks.
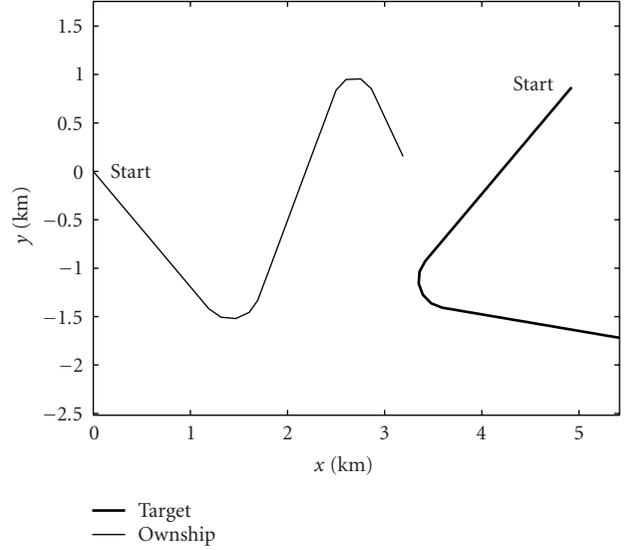
FIGURE 1: A typical bearings-only tracking scenario with a manoeuvring target.

5.1. Single-sensor case

The target-observer geometry for this case is shown in Figure 1. The target which is initially 5 km away from the ownship maintains an initial course of $-140°$. It executes a manoeuvre in the interval 20–25 minutes to attain a new course of $100°$, and maintains this new course for the rest of the observation period. The ownship, travelling at a fixed speed of 5 knots and an initial course of $140°$, executes a manoeuvre in the interval 13–17 minutes to attain a new course of $20°$. It maintains this course for a period of 15 minutes at the end of which it executes a second manoeuvre and attains a new course of $155°$. The final target-observer range for this case is 2.91 km. Bearing measurements with accuracy $\sigma_\theta = 1.5°$ are received every $T = 1$ minute for an observation period of 40 minutes.

Unless otherwise mentioned, the following nominal filter parameters were used in the simulations. The initial range and speed prior standard deviations were set to $\sigma_r = 2$ km and $\sigma_s = 2$ knots, respectively. The initial course and its standard deviation were set to $\bar{c} = \theta_1 + \pi$ and $\sigma_c = \pi/\sqrt{12}$, where θ_1 is the initial bearing measurement. The process noise parameter was set to $\sigma_a = 1.6 \times 10^{-6}$ km/s^2. The MMPF and AUX-MMPF used $N = 5000$ particles while the JMS-PF used $N = 100$ particles. Resampling was carried out if $\hat{N}_{\text{eff}} < N_{\text{thr}}$, where the threshold was set to $N_{\text{thr}} = N/3$. The resampling scheme used in the simulations is an algorithm based on order statistics [21].

The transition probability matrix required for the jump Markov process was chosen to be

$$\mathbf{\Pi} = \begin{bmatrix} 0.9 & 0.05 & 0.05 \\ 0.4 & 0.5 & 0.1 \\ 0.4 & 0.1 & 0.5 \end{bmatrix} \quad (71)$$

TABLE 1: Performance comparison for the single-sensor case.

Algorithm/ CRLB	RMS error (final)			RTAMS	Improvement	Divergent
	(km)	(%)	η	(km)	(%)	tracks
IMM-EKF	1.18	40	22	1.07	0	0
IMM-UKF	0.80	28	32	0.72	32	1
MMPF	0.59	20	43	0.44	59	0
AUX-MMPF	0.55	19	46	0.47	56	0
JMS-PF	0.77	27	33	0.64	40	0
CRLB	0.25	9	100	0.21	80	—
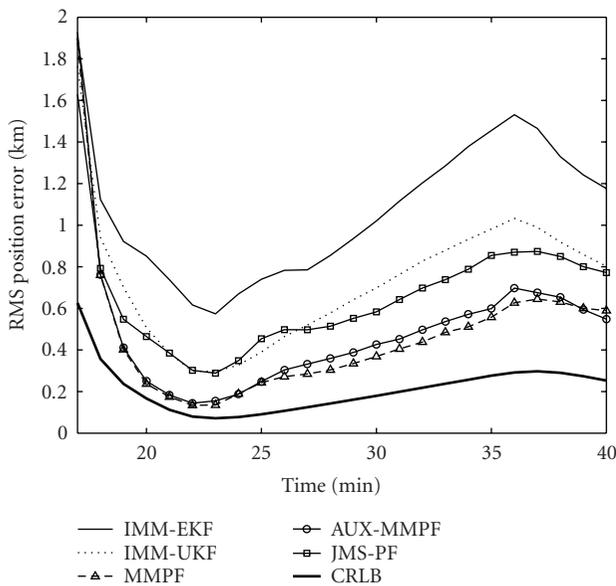
FIGURE 2: RMS position error versus time for a manoeuvring target scenario.

and the typical manoeuvre acceleration parameter for the filters was set to $a_m = 1.08 \times 10^{-5}$ km/s^2.

Figure 2 shows the RMS error curves corresponding to the five filters: IMM-EKF, IMM-UKF, MMPF, AUX-MMPF, and JMS-PF. A detailed comparison is also given in Table 1. Note that the column "improvement" refers to the percentage improvement in RTAMS error compared with a baseline filter which is chosen to be the IMM-EKF. From the graph and the table, it is clear that the performance of the IMM-EKF and IMM-UKF is poor compared to the other three filters. Though the final RMS error performance of the IMM-UKF is comparable to the JMS-PF, since it has one divergent track, its overall performance is considered worse than that of the JMS-PF. It is clear that the best filters for this case were the MMPF and AUX-MMPF which achieved 59% and 56% improvement, respectively, over the IMM-EKF. Also note that the JMS-PF performance is between that of IMM-EKF/IMM-UKF and MMPF/AUX-MMPF. From the simulations, it appears that the relative computational requirements (with respect to the IMM-EKF) for the IMM-UKF, MMPF, AUX-MMPF, and JMS-PF are 2.6, 23, 32, and 30, respectively.

The rationale for the performance differences noted above can be explained as follows. There are two sources of approximations in both IMM-EKF and IMM-UKF. Firstly, the probability of the mode history is approximated by the IMM routine which merges mode histories. Secondly, the mode-conditioned filter estimates are obtained using an EKF and an UKF (for the IMM-EKF and IMM-UKF, respectively), both of which approximate the non-Gaussian posterior density by a Gaussian. In contrast, the MMPF and AUX-MMPF attempt to alleviate both sources of approximations: they estimate the mode probabilities with no merging of histories and they make no linearisation (as in EKF) and characterise the non-Gaussian posterior density in a near-optimal manner. Thus we observe the superior performance of the MMPF and AUX-MMPF. The JMS-PF on the other hand is worse than MMPF/AUX-MMPF but better than IMM-EKF/IMM-UKF as it attempts to alleviate only one of the sources of approximations discussed above. Specifically, while the JMS-PF attempts to compute the mode history probability exactly, it uses an EKF (a local linearisation approximation) to compute the mode-conditioned filtered estimates. Hence, note that even if the number of particles for the JMS-PF is increased, its performance can never reach that of the MMPF/AUX-MMPF. It is interesting to note from the improvement figures for the JMS-PF and MMPF that the first source of approximation is more critical than the second one. In fact, the contributions of the first and second sources of approximation appear to be in the ratio 2 : 1.

It is worth noting that in the above simulations, the performance of the AUX-MMPF is comparable to that of the MMPF. This is expected due to the low process noise used in the simulations as one would expect the performance of the AUX-MMPF to approach that of MMPF as the process noise tends to zero. However, for problems with moderate to high process noise, the AUX-MMPF is likely to outperform the MMPF.

Next, we illustrate a case where the IMM-EKF shows a tendency to diverge while the MMPF tracks the target well for the same set of measurements. Figure 3a shows the estimated track and 95% error ellipses (plotted every 8 minutes) for the IMM-EKF. Note that the IMM-EKF covariance estimate at 8 minutes is poor as it does not encapsulate the true target position. This has resulted in not only subsequent poor track estimates, but also inability to detect the target manoeuvre.
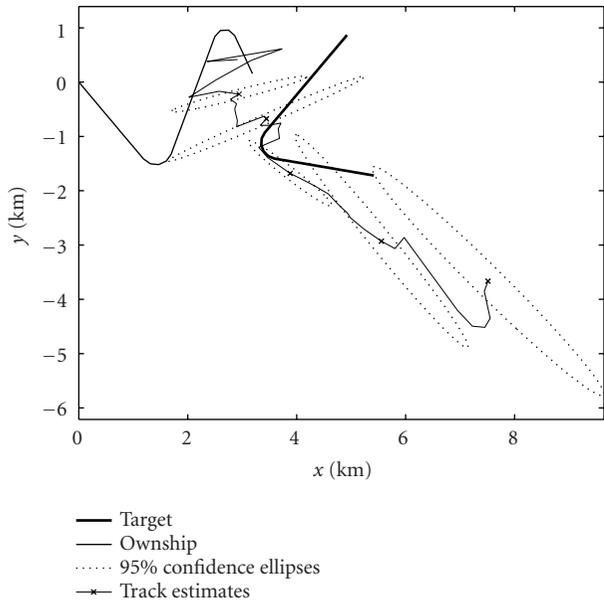
(a)
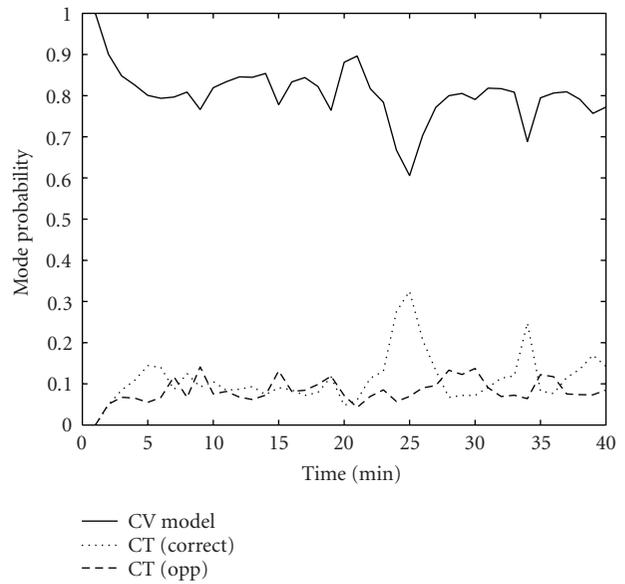
(b)

Figure 3: IMM-EKF tracker results. (a) Track estimates and 95% confidence ellipses. (b) Mode probabilities.
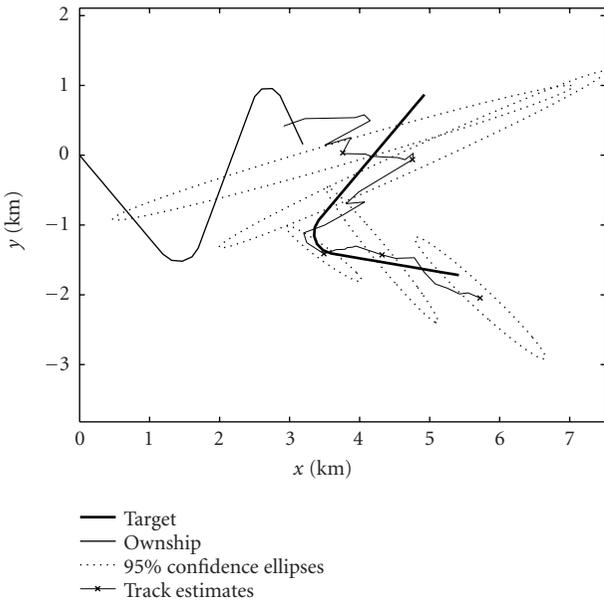
(a)
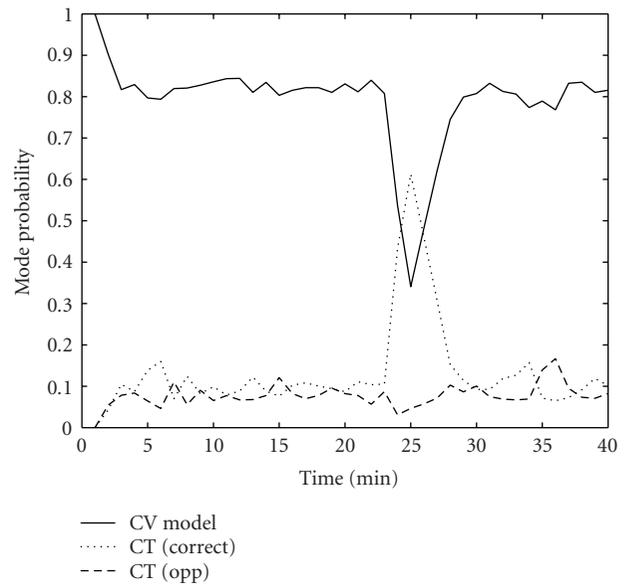
(b)

Figure 4: MMPF tracker results. (a) Track estimates and 95% confidence ellipses. (b) Mode probabilities.

This is clear from the mode probability curves shown in Figure 3b, where we note that though there is a slight bump in the mode probability for the correct manoeuvre model, the algorithm is unable to establish the occurrence of the manoeuvre. The overall result is a track that is showing a tendency to diverge from the true track.

For the same set of measurements, the MMPF shows excellent performance as can be seen from Figure 4. Here we note that the 95% confidence ellipse of the PF encapsulates the true target position at all times. Notice that the size of the covariance matrix shortly after the target manoeuvre is small compared to other times. The reason for this is that the
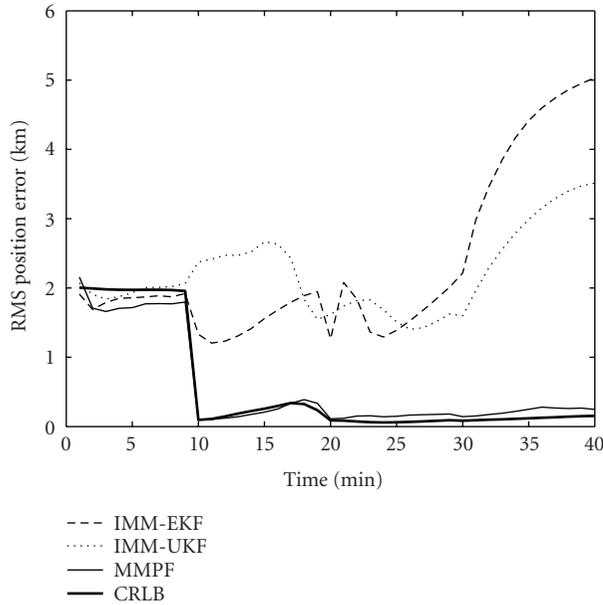
FIGURE 5: RMS position error versus time for a multisensor case.

target observability is best at that instant compared to other times. For the given scenario, both the ownship manoeuvre and the target manoeuvre have resulted in a geometry that is very observable at that instant. After the target manoeuvre, the relative position of the target increases and this leads to a slight decrease in observability and hence slight enlargement of the covariance matrix. The mode probability curves for the MMPF shows that unlike the results of IMM-EKF, the MMPF mode probabilities indicate a higher probability of occurrence of a manoeuvre. The overall result is a much better tracker performance for the same set of measurements.

5.2. Multisensor case

Here we consider the scenario identical to the one considered in Section 5.1, except that an additional static sensor, located at $(5\,\text{km}, -2\,\text{km})$, provides bearing measurements to the ownship at regular time intervals. These measurements, with accuracy $\sigma_{\theta'} = 2°$, arrive at only 3 time epochs, namely, at $k = 10, 20$, and 30. Figure 5 shows a comparison of IMM-EKF, IMM-UKF, and MMPF for this case. It is seen that the MMPF exhibits excellent performance, with RMS error results very close to the CRLB. The detailed comparison given in Table 2 shows that MMPF achieves a final RMS error accuracy that is within 8% of the final range. By comparing with the corresponding results for the single-sensor case, we note that the final RMS error is reduced by a factor of 2.5. Interestingly, the IMM-EKF and IMM-UKF performance is very poor and is worse than their corresponding performance when no additional measurement is received. Though this may seem counterintuitive, it can be explained as follows. For the given geometry, at the time of the first arrival of the bearing measurement from the second sensor, it is possible that due to nonlinearities and low observability in the time interval 0–10 minutes, the track estimates and filter calcu-

lated covariance of the IMM-based filters are in error, leading to a large innovation for the second sensor measurement. The inaccurate covariance estimate results in an incorrect filter gain computation for the second sensor measurement. In the update equations of these filters, the large innovation gets weighted by the computed gain which does not properly reflect the contribution of the new measurement. The consequence of this is filter divergence. It turns out that for the ownship measurements-only case, even if the track and covariance estimates are in error, the errors introduced in the filter gain computation are not as severe as in the multisensor case. Furthermore, as the uncertainty is mainly along the line of bearing, the innovation for this case is not likely to be very large. Thus the severity of track and covariance error for this particular scenario is worse for the multisensor case than for the single-sensor case. Similar results have been observed in the context of an air surveillance scenario [12].

5.3. Tracking with hard constraints

In this section, we present the results for the case of bearings-only tracking with hard constraints. The scenario and parameters used for this case are identical to the ones considered in Section 5.1. This time, however, in addition to the available bearing measurements, we also impose some hard constraints on target speed. Specifically, assume that we have prior knowledge that the target speed is in the range $3.5 \le s \le 4.5$ knots. This type of nonstandard information is difficult to incorporate into the standard EKF-based algorithms (such as the IMM-EKF), and so in the comparison below, the IMM-EKF will not utilise the hard constraints. However, the PF-based algorithms, and, in particular, the MMPF and AUX-MMPF, can easily incorporate such nonstandard information according to the technique described in Section 4.7.

Figure 6 shows the RMS error in estimated position for the MMPF that incorporates prior knowledge of speed constraint (referred to as MMPF-C). The figure also shows the performance curves of the IMM-EKF and the standard MMPF that do not utilise knowledge of hard constraints. A detailed numerical comparison is given in Table 3. It can be seen that the MMPF-C achieves 83% improvement in RTAMS over the IMM-EKF. Also, observe that by incorporating the hard constraints, the MMPF-C achieves a 50% reduction in RTAMS error over the standard MMPF that does not utilise hard constraints (emphasising the significance of this nonstandard information). Incorporating such nonstandard information results in highly non-Gaussian posterior pdfs which the PF is effectively able to characterise.

6. CONCLUSIONS

This paper presented a comparative study of PF-based trackers against conventional IMM-based routines for the problem of bearings-only tracking of a manoeuvring target. Three separate cases have been analysed: single-sensor case; multisensor case, and tracking with speed constraints. The results overwhelmingly confirm the superior performance of PF-based algorithms against the conventional IMM-based

TABLE 2: Performance comparison for the multisensor case.

Algorithm/	RMS error (final)			RTAMS	Improvement	Divergent
CRLB	(km)	(%)	η	(km)	(%)	tracks
IMM-EKF	5.03	173	3	3.16	0	17
IMM-UKF	3.51	121	4	2.32	27	7
MMPF	0.25	8	63	0.22	93	1
CRLB	0.15	5	100	0.13	96	—

TABLE 3: Performance comparison for tracking with hard constraints.

Algorithm	RMS error (final)		RTAMS	Improvement	Divergent
	(km)	(%)	(km)	(%)	tracks
IMM-EKF	1.37	47	1.21	0	0
MMPF	0.53	18	0.44	64	0
MMPF-C	0.12	4	0.20	83	0
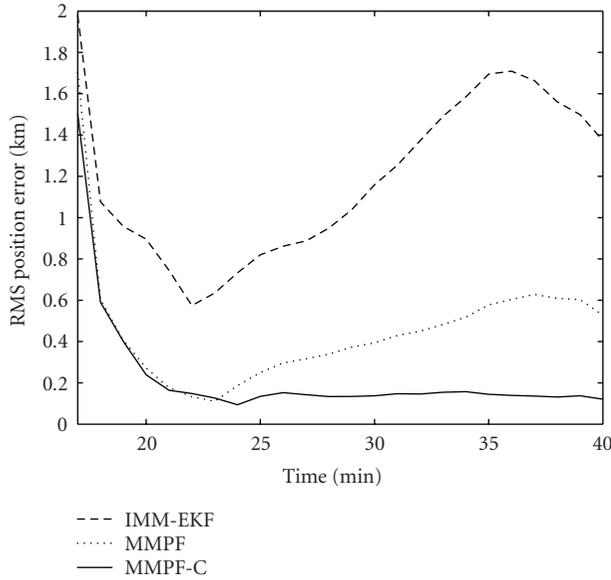
FIGURE 6: RMS position error versus time for the case of tracking with speed constraint $3.5 \leq s \leq 4.5$ knots.

schemes. The key strength of the PF, demonstrated in this application, is its flexibility to handle nonstandard information along with the ability to deal with nonlinear and non-Gaussian models.

APPENDIX

JACOBIANS OF THE MANOEUVRE DYNAMICS

The Jacobians $\tilde{\mathbf{F}}_k^{(r_{k+1}^*)}, r_{k+1}^* \in \{2, 3\}$, of the manoeuvre dynamics can be computed by taking the gradients of the respective transitions. Let $f_i^{(j)}(\cdot)$ denote the ith element of the dynamics model function $\mathbf{f}^{(j)}(\cdot)$ and let $(\dot{x}_k^t, \dot{y}_k^t)$ denote the target velocity vector. Then, by taking the gradients of $\mathbf{f}^{(j)}(\cdot)$ for

$j = 2, 3$, the required Jacobians can be computed to give

$$\tilde{\mathbf{F}}_k^{(j)} = \begin{bmatrix} 1 & 0 & \dfrac{\partial f_1^{(j)}}{\partial \dot{x}_k} & \dfrac{\partial f_1^{(j)}}{\partial \dot{y}_k} \\[2mm] 0 & 1 & \dfrac{\partial f_2^{(j)}}{\partial \dot{x}_k} & \dfrac{\partial f_2^{(j)}}{\partial \dot{y}_k} \\[2mm] 0 & 0 & \dfrac{\partial f_3^{(j)}}{\partial \dot{x}_k} & \dfrac{\partial f_3^{(j)}}{\partial \dot{y}_k} \\[2mm] 0 & 0 & \dfrac{\partial f_4^{(j)}}{\partial \dot{x}_k} & \dfrac{\partial f_4^{(j)}}{\partial \dot{y}_k} \end{bmatrix}, \quad j = 2, 3, \qquad (A.1)$$

where

$$\frac{\partial f_1^{(j)}}{\partial \dot{x}_k} = \frac{\sin(\Omega_k^{(j)} T)}{\Omega_k^{(j)}} + g_1^{(j)}(k) \frac{\partial \Omega_k^{(j)}}{\partial \dot{x}_k},$$

$$\frac{\partial f_1^{(j)}}{\partial \dot{y}_k} = \frac{-(1 - \cos(\Omega_k^{(j)} T))}{\Omega_k^{(j)}} + g_1^{(j)}(k) \frac{\partial \Omega_k^{(j)}}{\partial \dot{y}_k},$$

$$\frac{\partial f_2^{(j)}}{\partial \dot{x}_k} = \frac{(1 - \cos(\Omega_k^{(j)} T))}{\Omega_k^{(j)}} + g_2^{(j)}(k) \frac{\partial \Omega_k^{(j)}}{\partial \dot{x}_k},$$

$$\frac{\partial f_2^{(j)}}{\partial \dot{y}_k} = \frac{\sin(\Omega_k^{(j)} T)}{\Omega_k^{(j)}} + g_2^{(j)}(k) \frac{\partial \Omega_k^{(j)}}{\partial \dot{y}_k},$$

$$\frac{\partial f_3^{(j)}}{\partial \dot{x}_k} = \cos(\Omega_k^{(j)} T) + g_3^{(j)}(k) \frac{\partial \Omega_k^{(j)}}{\partial \dot{x}_k},$$

$$\frac{\partial f_3^{(j)}}{\partial \dot{y}_k} = -\sin(\Omega_k^{(j)} T) + g_3^{(j)}(k) \frac{\partial \Omega_k^{(j)}}{\partial \dot{y}_k},$$

$$\frac{\partial f_4^{(j)}}{\partial \dot{x}_k} = \sin(\Omega_k^{(j)} T) + g_4^{(j)}(k) \frac{\partial \Omega_k^{(j)}}{\partial \dot{x}_k},$$

$$\frac{\partial f_4^{(j)}}{\partial \dot{y}_k} = \cos(\Omega_k^{(j)} T) + g_4^{(j)}(k) \frac{\partial \Omega_k^{(j)}}{\partial \dot{y}_k},$$

(A.2)

with

$$g_1^{(j)}(k) = \frac{T \cos(\Omega_k^{(j)} T) \dot{x}_k^t}{\Omega_k^{(j)}} - \frac{\sin(\Omega_k^{(j)} T) \dot{x}_k^t}{(\Omega_k^{(j)})^2}$$

$$- \frac{T \sin(\Omega_k^{(j)} T) \dot{y}_k^t}{\Omega_k^{(j)}}$$

$$+ \frac{(-1 + \cos(\Omega_k^{(j)} T)) \dot{y}_k^t}{(\Omega_k^{(j)})^2},$$

$$g_2^{(j)}(k) = \frac{T \sin(\Omega_k^{(j)} T) \dot{x}_k^t}{\Omega_k^{(j)}} - \frac{(1 - \cos(\Omega_k^{(j)} T)) \dot{x}_k^t}{(\Omega_k^{(j)})^2}$$

$$+ \frac{T \cos(\Omega_k^{(j)} T) \dot{y}_k^t}{\Omega_k^{(j)}} - \frac{\sin(\Omega_k^{(j)} T) \dot{y}_k^t}{(\Omega_k^{(j)})^2}, \qquad \text{(A.3)}$$

$$g_3^{(j)}(k) = -\sin(\Omega_k^{(j)} T) T \dot{x}_k^t - \cos(\Omega_k^{(j)} T) T \dot{y}_k^t,$$

$$g_4^{(j)}(k) = \cos(\Omega_k^{(j)} T) T \dot{x}_k^t - \sin(\Omega_k^{(j)} T) T \dot{y}_k^t,$$

$$\frac{\partial \Omega_k^{(j)}}{\partial \dot{x}_k} = \frac{(-1)^{j+1} a_m \dot{x}_k^t}{\left[(\dot{x}_k^t)^2 + (\dot{y}_k^t)^2 \right]^{3/2}},$$

$$\frac{\partial \Omega_k^{(j)}}{\partial \dot{y}_k} = \frac{(-1)^{j+1} a_m \dot{y}_k^t}{\left[(\dot{x}_k^t)^2 + (\dot{y}_k^t)^2 \right]^{3/2}}$$

for $j = 2, 3$.

REFERENCES

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

[2] J. C. Hassab, *Underwater Signal and Data Processing*, CRC Press, Boca Raton, Fla, USA, 1989.

[3] S. Nardone, A. Lindgren, and K. Gong, "Fundamental properties and performance of conventional bearings-only target motion analysis," *IEEE Trans. Automatic Control*, vol. 29, no. 9, pp. 775–787, 1984.

[4] E. Fogel and M. Gavish, "Nth-order dynamics target observability from angle measurements," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 24, no. 3, pp. 305–308, 1988.

[5] T. L. Song, "Observability of target tracking with bearings-only measurements," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 4, pp. 1468–1472, 1996.

[6] S. S. Blackman and S. H. Roszkowski, "Application of IMM filtering to passive ranging," in *Proc. SPIE Signal and Data Processing of Small Targets*, vol. 3809 of *Proceedings of SPIE*, pp. 270–281, Denver, Colo, USA, July 1999.

[7] T. Kirubarajan, Y. Bar-Shalom, and D. Lerro, "Bearings-only tracking of maneuvering targets using a batch-recursive estimator," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 3, pp. 770–780, 2001.

[8] J.-P. Le Cadre and O. Tremois, "Bearings-only tracking for maneuvering sources," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 1, pp. 179–193, 1998.

[9] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings Part F: Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, 1993.

[10] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*, Springer, New York, NY, USA, 2001.

[11] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/ non-Gaussian Bayesian tracking," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[12] B. Ristic and M. S. Arulampalam, "Tracking a manoeuvring target using angle-only measurements: algorithms and performance," *Signal Processing*, vol. 83, no. 6, pp. 1223–1238, 2003.

[13] S. McGinnity and G. W. Irwin, "Multiple model bootstrap filter for maneuvering target tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, no. 3, pp. 1006–1012, 2000.

[14] R. Karlsson and N. Bergman, "Auxiliary particle filters for tracking a maneuvering target," in *Proc. 39th IEEE Conference on Decision and Control*, vol. 4, pp. 3891–3895, Sydney, Australia, December 2000.

[15] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Trans. Automatic Control*, vol. 45, no. 3, pp. 477–482, 2000.

[16] Y. Bar-Shalom and X. R. Li, *Estimation and Tracking: Principles, Techniques and Software*, Artech House, Norwood, Mass, USA, 1993.

[17] P. Tichavsky, 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, 1998.

[18] A. Doucet, N. J. Gordon, and V. Krishnamurthy, "Particle filters for state estimation of jump Markov linear systems," *IEEE Trans. Signal Processing*, vol. 49, no. 3, pp. 613–624, 2001.

[19] N. Bergman, A. Doucet, and N. Gordon, "Optimal estimation and Cramér-Rao bounds for partial non-Gaussian state space models," *Annals of the Institute of Statistical Mathematics*, vol. 53, no. 1, pp. 97–112, 2001.

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

[21] J. Carpenter, P. Clifford, and P. Fearnhead, "Improved particle filter for nonlinear problems," *IEE Proceedings—Radar, Sonar and Navigation*, vol. 146, no. 1, pp. 2–7, 1999.

M. Sanjeev Arulampalam received the B.S. degree in mathematics and the B.E. degree with first-class honours in electrical and electronic engineering from the University of Adelaide in 1991 and 1992, respectively. In 1993, he won a Telstra Research Labs Postgraduate Fellowship award to work toward a Ph.D. degree in electrical and electronic engineering at the University of Melbourne, which he completed in 1997. Since 1998, Dr. Arulampalam has been with the Defence Science and Technology Organisation (DSTO), Australia, working on many aspects of target tracking with a particular emphasis on nonlinear/non-Gaussian tracking problems. In March 2000, he won the Anglo-Australian Postdoctoral Research Fellowship, awarded by the Royal Academy of Engineering, London. This postdoctoral research was carried out in the UK, both at the Defence Evaluation and Research Agency (DERA) and at Cambridge University, where he worked on particle filters for nonlinear tracking problems. Currently, he is a Senior Research Scientist in the Submarine Combat Systems Group of Maritime Operations

Division, DSTO, Australia. His research interests include estimation theory, target tracking, and sequential Monte Carlo methods. Dr. Arulampalam coauthored a recent book, *Beyond the Kalman Filter: Particle Filters for Tracking Applications,* Artech House, 2004.

B. Ristic received all his degrees in electrical engineering: a Ph.D. degree from Queensland University of Technology (QUT), Australia, in 1995, an M.S. degree from Belgrade University, Serbia, in 1991, and a B.E. degree from the University of Novi Sad, Serbia, in 1984. He began his career in 1984 at the Vinča Institute, Serbia. From 1989 to 1994, he was with the University of Queensland, Brisbane, and QUT, Australia, doing research related to the automatic placement and routing of integrated circuits and the design and analysis of time-frequency and time-scale distributions. In 1995, he was with GEC Marconi Systems in Sydney, Australia, developing a concept demonstrator for noise cancellation in towed arrays. Since 1996, he has been with the Defense Science and Technology Organisation (DSTO), Australia, where he has been involved in the design and analysis of target tracking and data fusion systems. During 2003/2004, he has been on a study leave with Université Libre de Bruxelles (ULB), Belgium, doing research on reasoning under uncertainty. Dr. Ristic coauthored a recent book, *Beyond the Kalman Filter: Particle Filters for Tracking Applications,* Artech House, 2004. During his career, he has published more than 80 technical papers.

N. Gordon obtained a B.S. in mathematics and physics from Nottingham University in 1988 and a Ph.D. degree in statistics from Imperial College, University of London in 1993. He was with the Defence Evaluation and Research Agency (DERA) and QinetiQ in the UK from 1988 to 2002. In 2002, he joined the Tracking and Sensor Fusion Research Group at Defence Science and Technology Organisation (DSTO) in Australia. Neil has written approximately 65 articles in peer-reviewed journals and international conferences on tracking and other dynamic state estimation problems. He is the coauthor/coeditor of two books on particle filtering.

T. Mansell received a B.S. degree with first-class honours in physics and electronics from Deakin University in 1989. Following graduation, he joined the Defence Science and Technology Organisation (DSTO) in Melbourne as a Cadet Research Scientist, and in 1990 began a Ph.D. in artificial intelligence with The University of Melbourne. On completion of his Ph.D. in 1994, Dr. Mansell began working on the application of information fusion techniques to naval problems (including mine warfare and combat systems). In 1996, Dr. Mansell undertook a 15-month posting with Canada's Defence Research Establishment Atlantic in the sonar information management group looking at tactical decision-making and naval combat systems. On return to Australia, Dr. Mansell relocated to DSTO, Edinburgh, South Australia, to lead the submarine combat systems task. In 2000, he became Head of the Submarine Combat Systems Group (SMCS) within Maritime Operations Division. In 2003, Dr. Mansell was appointed Head of Maritime Combat Systems and the DSTO lead for the Air Warfare Destroyer, Combat System Studies. Dr. Mansell is currently the Counsellor for Defence Science in the Australian High Commission, London. Dr. Mansell's main research interests are in combat system architectures, open system architectures, information fusion, artificial intelligence, human-machine interaction, and tactical decision support systems.