Research Article

Multiple Maneuvering Target Tracking by Improved Particle Filter Based on Multiscan JPDA

Jing Liu,[1] ChongZhao Han,[1] Feng Han,[1] and Yu Hu[2]

[1] *MOE Key Lab for Intelligent and Networked Systems, Institute of Integrated Automation,*
  School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an,
  Shaanxi Province 710049, China
[2] *School of Aeronautics, Northwestern Polytechnical University, Xi'an, Shaanxi Province 710072, China*

Correspondence should be addressed to Jing Liu, elelj20080730@gmail.com

The multiple maneuvering target tracking algorithm based on a particle filter is addressed. The equivalent-noise approach is adopted, which uses a simple dynamic model consisting of target state and equivalent noise which accounts for the combined effects of the process noise and maneuvers. The equivalent-noise approach converts the problem of maneuvering target tracking to that of state estimation in the presence of nonstationary process noise with unknown statistics. A novel method for identifying the nonstationary process noise is proposed in the particle filter framework. Furthermore, a particle filter based multiscan Joint Probability Data Association (JPDA) filter is proposed to deal with the data association problem in a multiple maneuvering target tracking. In the proposed multiscan JPDA algorithm, the distributions of interest are the marginal filtering distributions for each of the targets, and these distributions are approximated with particles. The multiscan JPDA algorithm examines the joint association events in a multiscan sliding window and calculates the marginal posterior probability based on the multiscan joint association events. The proposed algorithm is illustrated via an example involving the tracking of two highly maneuvering, at times closely spaced and crossed, targets, based on resolved measurements.

1. Introduction

The problem of multiple maneuvering target tracking has received a considerable attention in recent years. Multiple model approaches are proposed to model highly maneuvering targets and the interacting multiple model (IMM) algorithm [1–3] is the most popular one among them. Moreover, in the presence of clutter, some of the sensor measurements may not relate to the target of interest. The problem of data association has to be solved. Effective approaches in a Bayesian framework include the probabilistic data association (PDA) approach [4, 5] and the integrated probabilistic data association (IPDA) approach [6–8] for tracking a single

target in clutter, the JPDA approach [4, 9, 10], and the joint integrated probabilistic data association (JIPDA) approach [11] for tracking multiple targets in clutter. The PDA and JPDA approaches address the track maintenance issue, while the IPDA and JIPDA approaches tackle the issues related to track initiation and track termination as well as track maintenance.

Different combinations of the IMM method and the PDA/JPDA/IPDA/JIPDA methods have been used to tackle multiple maneuvering target tracking problems. In [12], the IMM algorithm is combined with the PDA filter in a multiple sensor scenario proposing a combined IMMPDA algorithm. The IMMPDA algorithm has good tracking performance when the targets are sufficiently separated. In [13], an IMMJPDA-coupled filter is developed for situations where the measurements of two targets are unresolved during periods of close encounter. The filters corresponding to the individual targets are coupled through cross-target-covariance terms. In [14], JPDA is combined with a crude approximation of IMM, under the name IMMJPDA. In [15], an IMMJIPDA algorithm is used in situations with a small number of crossing targets and low clutter measurement density. In order to deal with more complex scenarios with a large number of targets, an IMM-LMIPDA algorithm has been proposed by integrating linear multitarget (LM) method with IMM-IPDA. The computational requirements of the proposed IMM-LMIPDA algorithm increase linearly in the number of targets and the number of measurements.

Recently, several methods which significantly outperform IMMJPDA have been proposed to deal with the multiple maneuvering target tracking problem. In [16], an IMMJPDA uncoupled fixed-lag smoothing algorithm is proposed, which performs far better than IMMJPDA filtering. In [17, 18], the descriptor system approach [19] is used to develop a track-coalescence-avoiding version of IMMJPDA. In [20], multiscan information is incorporated in JPDA to tackle the data association problem associated with the multiple maneuvering target tracking. In [21], a new track-coalescence-avoiding IMMCPDA method is proposed to improve the situation of tracking closely spaced targets by characterizing the exact Bayesian filter, and by developing novel target tracking combinations of IMM and PDA.

In the above methods, the data models are assumed Gaussian and weakly nonlinear, and the Kalman filter/extended Kalman filter (KF/EKF) is used to perform target state estimation. More recently, nonlinear filtering techniques have been gaining more attention. The most common one among them is the particle filter [22], a state estimate pdf sampling based algorithm.

There are not many reported works concerning the particle filter based multiple maneuvering target tracking methods. In [23], the problem of maintaining tracks of multiple maneuvering targets from unassociated measurements is formulated as a problem of estimating the hybrid state of a Markov jump linear system from measurements made by a descriptor system with independent, identically distributed (i.i.d.) stochastic coefficients. The possible multiple motion models and transition probability matrices are assumed to be known in the proposed method. In [24], the marginal filtering distributions for each of the targets are approximated with particles. The posterior marginal probability is calculated based on the joint association hypotheses, which are examined in a single scan. However, single scan information may not be enough for tracking maneuvering targets, especially in the condition of tracking two closely spaced targets.

In this work, a novel method is proposed for a multiple maneuvering target tracking, which is a combination of the particle filter based process noise identification algorithm and the particle filter based multiscan JPDA algorithm. The process noise identification process is effective in estimating both the maneuvering movement and the random acceleration of the target, using one general model for each of the maneuvering targets. The multiscan JPDA

is effective in maintaining the tracks of multiple targets using multiple scan information. Compared with the single scan JPDA method, the multiscan JPDA method uses richer information, which results in better estimated probabilities.

The rest of the paper is organized as follows. Section 2 provides a brief introduction to particle filter, and the multiple maneuvering target tracking problem is defined in Section 3. The particle filter based process noise identification method for tracking highly maneuvering target is introduced in Section 4. The data association method based on multiscan JPDA is discussed in Section 5. The proposed multiple maneuvering target tracking algorithm, which is a combination of the above two algorithms, is introduced in Section 6. The simulation results are shown in Section 7 and the paper is summarized in Section 8.

2. Basic Theory of Particle Filter

To define the problem of tracking, consider a dynamic system represented by the state equation:

$$x_k = f\left(x_{k-1}, v_{k-1}^*\right), \tag{2.1}$$

where x is the state, f is a possibly nonlinear function, and v^* is the known process noise with a zero mean Gaussian distribution. The objective of tracking is to recursively estimate x_k from a sequence of measurements up to time step k, $z_{1:k} = \{z_1, z_2, \ldots, z_k\}$. The observation model is described as follows:

$$z_k = h(x_k, n_k), \tag{2.2}$$

where h is a possibly nonlinear function. n is the observation noise with a zero mean Gaussian distribution. From the Bayesian perspective, the tracking problem is to recursively calculate the posterior distribution $p(x_k \mid z_{1:k})$.

In this paper, a particle filter is considered to solve the state estimation problem due to its ability to tackle the nonlinear and non-Gaussian systems. The posterior distribution $p(x_k \mid z_{1:k})$ is approximated by a set of particles with associated weights. The detailed introduction about particle filter algorithm can be found in [22]. The procedures associated with the standard particle filter are listed in the following.

Algorithm 2.1 (particle filter algorithm). (1) *Initialization*. Sample initial particles $\{x_0^i, i = 1, \ldots, H\}$ from the initial posterior distribution $p(x_0)$ and set the weights w_0^i to $1/H$, $i = 1, \ldots, H$. H is the number of particles.

(2) *Prediction*. Particles at time step $k - 1$, $\{x_{k-1}^i, i = 1, \ldots, H\}$, are passed through the system model (2.3) to obtain the predicted particles at time step k, $\{\hat{x}_k^i, i = 1, \ldots, H\}$:

$$\hat{x}_k^i = f\left(x_{k-1}^i, v_{k-1}^{*,i}\right), \tag{2.3}$$

where $v_{k-1}^{*,i}$ is a sample drawn from the known zero mean Gaussian distribution.

(3) *Update*. Once the observation data, z_k, is measured, evaluate the importance weight of each predicted particle and obtain the normalized weight for each particle:

$$\widehat{w}_k^i = p\left(z_k \mid \widehat{x}_k^i\right), \qquad w_k^i = \frac{\widehat{w}_k^i}{\sum_{i=1}^H \widehat{w}_k^i}. \tag{2.4}$$

Thus at time step k we can obtain the estimate of the state, $\widetilde{\widetilde{x}}_k = \sum_{i=1}^H w_k^i \widehat{x}_k^i$.

(4) *Resample*. Resample the discrete distribution $\{w_k^i : i = 1, \ldots, H\}$, H times to generate particles $\{x_k^j : j = 1, \ldots, H\}$, so that for any j, $\Pr\{x_k^j = \widehat{x}_k^i\} = w_k^i$. Set the weights w_k^i to $1/H$, $i = 1, \ldots, H$, and move to Stage 2.

3. Multiple Maneuvering Target Tracking Problem Formulation

The number of maneuvering targets (M) to be tracked is assumed as fixed and known, where each target track has been initiated, and our objective is to maintain the tracks. Each target is parameterized by a state $x_{m,k}$, where m denotes the mth target and k denotes time step k. The combined state, $x_k = (x_{1,k}, \ldots, x_{M,k})$, is the concatenation of the individual target states. The individual targets are assumed to evolve independently according to a general motion model,

$$x_k = f\left(x_{k-1}, u_{k-1}, v_{k-1}^*\right), \tag{3.1}$$

where u is the maneuver acceleration and v^* is the process noise.

The observation vector z_k is composed of multiple sensor measurements $\{z_{j,k}, j = 1, \ldots, N_k\}$, where N_k is the total number of measurements. It is assumed that there are no unresolved measurements (i.e., measurements associated with two or more targets simultaneously); any measurement is either associated with a single target or caused by clutter. Clutter is modeled as independently and identically distributed (IID) with an uniform spatial distribution over the surveillance area.

In this paper, the multiple maneuvering target tracking problem is divided into two parts: single maneuvering target tracking and multiple target tracking, which are introduced in Sections 4 and 5, respectively. The particle filter based process noise identification method for tracking a highly maneuvering target is introduced in Section 4. The data association method based on multiscan JPDA is discussed in Section 5. Finally, in Section 6, the combination of the two methods is combined to deal with the multiple maneuvering target tracking problem.

4. Maneuvering Target Tracking Based on Process Noise Identification

In this section, the particle filter based process noise identification method is proposed for tracking highly maneuvering target. The general motion model of a maneuvering target is

described by (3.1). In the equivalent-noise approach [25–27], it is assumed that the general motion model (3.1) that describes target motions can be simplified to

$$x_k = f(x_{k-1}, v_{k-1}), \tag{4.1}$$

with an adequate accuracy, where v is the equivalent noise that quantifies the error of this model in describing the target motions, in particular, maneuvers. The statistics of this noise v, nonstationary in general, are not known.

In this section, a novel method is proposed for process noise identification. The process noise is modeled as a dynamic system. The noise vector v_{k-1} is chosen as the state of the noise system. The observation vector is z_k, which is the same as in (2.2). The observation equation is defined in

$$z_k = h(x_k, n_k) = h\big[f(x_{k-1}, v_{k-1}), n_k\big]. \tag{4.2}$$

The aim of the proposed method is to estimate the posterior distribution of the process noise, $p(v_{k-1} \mid z_{1:k})$. According to the Bayesian inference theory,

$$p(v_{k-1} \mid z_{1:k}) = \frac{p(z_k \mid v_{k-1})p(v_{k-1} \mid z_{1:k-1})}{p(z_k \mid z_{1:k-1})}, \tag{4.3}$$

where $p(z_k \mid z_{1:k-1})$ is a normalizing constant and $1/p(z_k \mid z_{1:k-1})$ is defined as Υ. So we can obtain

$$p(v_{k-1} \mid z_{1:k}) = \Upsilon \cdot p(z_k \mid v_{k-1})p(v_{k-1} \mid z_{1:k-1}). \tag{4.4}$$

The noise v_{k-1} may be from random accelerations, sudden maneuvers, or both, and there is no information about the distribution of v_{k-1}. v_{k-1} is not dependent on the previous measurements $z_{1:k-1}$, which results in

$$p(v_{k-1} \mid z_{1:k}) = \Upsilon \cdot p(z_k \mid v_{k-1})p(v_{k-1}). \tag{4.5}$$

According to the Bayesian theory, the prior distribution of parameters, on which there is no information, could be considered as a uniform distribution. In this paper, the prior distribution of the process noise is considered as a uniform distribution, and the posterior distribution of the process noise is then obtained via a Monte Carlo deviation process. The general procedure of the particle filter based process noise identification algorithm is as follows.

Initially a number of process noise samples are generated from a uniform distribution since there is no information on the process noise distribution due to the uncertain dynamics. Each of the process noise samples is assigned with a weight based on the current measurement and the particles from the previous time step. According to Monte Carlo theory, the process noise samples and their associated weights approximate the true posterior distribution of the process noise. The process noise samples are then resampled according to their associate weights. The samples with large weights are duplicated, while

the samples with small weights are removed. The resampled process noise samples distribute approximately according to the true posterior distribution of the process noise.

Since there is no information about v_{k-1}, it is reasonable to assume that v_{k-1} is uniformly distributed in the range of $U(-d, d)$, where U denotes a uniform distribution and d is the known process noise bound accounting for the maximin uncertain dynamics. According to the Monte Carlo theory, $p(v_{k-1})$ can be represented by H samples, $\{\hat{v}_{k-1}^j, \ j = 1, \ldots, H\}$, from $U(-d, d)$. Consider

$$p(v_{k-1}) = \frac{1}{H} \sum_{j=1}^{H} \delta\left(v_{k-1} - \hat{v}_{k-1}^j\right). \tag{4.6}$$

The number of process noise samples (H) is proportional to the magnitude of the maximum uncertain dynamics ($\|d\|$). The posterior distribution of v_{k-1} can be represented as

$$p(v_{k-1} \mid z_{1:k}) = \frac{\Upsilon}{H} \sum_{j=1}^{H} p\left(z_k \mid \hat{v}_{k-1}^j\right) \delta\left(v_{k-1} - \hat{v}_{k-1}^j\right)$$

$$= \frac{\Upsilon}{H} \sum_{j=1}^{H} \xi_k^j \cdot \delta\left(v_{k-1} - \hat{v}_{k-1}^j\right), \tag{4.7}$$

where $\xi_k^j = p(z_k \mid \hat{v}_{k-1}^j)$ is defined as the weight assigned to the jth process noise sample $\hat{v}_{k-1}^j$. The process noise samples $\{\hat{v}_{k-1}^j, \ j = 1, \ldots, H\}$ are then resampled according to $\{\xi_k^j, \ j = 1, \ldots, H\}$ based on the principle that $\text{Pr}\{v_{k-1}^i = \hat{v}_{k-1}^j\} = \xi_k^j$, where $\{v_{k-1}^i, \ i = 1, \ldots, H\}$ are the process noise samples obtained from resampling. The obtained resampled process noise samples are approximately distributed according to the posterior distribution $p(v_{k-1} \mid z_{1:k})$.

In order to calculate ξ_k^j, the likelihood function $p(z_k \mid \hat{v}_{k-1}^j)$ is expanded based on the resampled state particles at time step $k - 1$, $\{x_{k-1}^i, \ i = 1, \cdots, H\}$. Consider

$$p\left(z_k \mid \hat{v}_{k-1}^j\right) = \sum_{i=1}^{H} p\left(z_k \mid x_{k-1}^i, \hat{v}_{k-1}^j\right) p\left(x_{k-1}^i \mid \hat{v}_{k-1}^j\right). \tag{4.8}$$

Since x_{k-1}^i and $\hat{v}_{k-1}^j$ are independent, $p(x_{k-1}^i \mid \hat{v}_{k-1}^j) = p(x_{k-1}^i)$.

The resmapled particles at time step $k - 1$, $\{x_{k-1}^i, \ i = 1, \ldots, H\}$, should be assigned with the same and equal weights, $1/H$. We can then obtain

$$p\left(x_{k-1}^i\right) = \frac{1}{H}. \tag{4.9}$$

To calculate $p(z_k \mid x_{k-1}^i, \hat{v}_{k-1}^j)$, define $\mu_k^{i,j}$ as the intermediate particle,

$$\mu_k^{i,j} = f\left(x_{k-1}^i, \hat{v}_{k-1}^j\right), \quad i = 1, \ldots H, \ j = 1, \ldots, H \tag{4.10}$$

and expand $p(z_k \mid x_{k-1}^i, \widehat{v}_{k-1}^j)$ based on $\mu_k^{i,j}$,

$$p\left(z_k \mid x_{k-1}^i, \widehat{v}_{k-1}^j\right) = \sum_{p=1}^{H} \sum_{q=1}^{H} \left[p\left(z_k \mid \mu_k^{p,q}, x_{k-1}^i, \widehat{v}_{k-1}^j\right) p\left(\mu_k^{p,q} \mid x_{k-1}^i, \widehat{v}_{k-1}^j\right)\right]. \qquad (4.11)$$

Since x_{k-1}^i and $\widehat{v}_{k-1}^j$ are known and $\mu_k^{p,q}$ is obtained from a purely deterministic relationship in (4.10), we obtain

$$p\left(\mu_k^{p,q} \mid x_{k-1}^i, \widehat{v}_{k-1}^j\right) = \begin{cases} 1, & \text{for } p = i \text{ and } q = j, \\ 0, & \text{for } p \neq i \text{ or } q \neq j, \end{cases} \qquad (4.12)$$

$$p\left(z_k \mid x_{k-1}^i, \widehat{v}_{k-1}^j\right) = p\left(z_k \mid \mu_k^{i,j}\right). \qquad (4.13)$$

Combining (4.9) and (4.13) with (4.8) results in

$$p\left(z_k \mid \widehat{v}_{k-1}^j\right) = \sum_{i=1}^{H} p\left(z_k \mid \mu_k^{i,j}\right) \cdot \frac{1}{H}. \qquad (4.14)$$

At each time step, the process noise samples are drawn from a uniform distribution $U(-d, d)$. Each process noise sample $\widehat{v}_{k-1}^j$ is evaluated and assigned its corresponding weight ξ_k^j. A resampling procedure is then used to redistribute the process noise samples, from which the process noise samples with large weights are replicated while the samples with small weights are eliminated.

The standard particle filter procedure for state estimation follows next. The predicted particles $\{\widehat{x}_k^i : i = 1, \ldots, H\}$ are then obtained based on the resampled process noise samples $\{v_{k-1}^i : i = 1, \ldots, H\}$ through the dynamic model (4.1). The predicted particles are updated and resampled as in the conventional particle filter algorithm.

The complete algorithm including the process noise estimation and state estimation parts is summarized below.

Algorithm 4.1 (particle filter based process noise identification). (1) At time step $k - 1$, draw process noise samples $\{\widehat{v}_{k-1}^j : j = 1, \ldots, H\}$ from a uniform distribution $U(-d, d)$.

(2) Calculate the intermediate particles $\{\mu_k^{i,j} : i = 1, \ldots, H; \ j = 1, \ldots, H\}$ according to (4.10).

(3) Calculate the process noise sample weights $\{\xi_k^j : j = 1, \ldots, H\}$ via (4.14).

(4) Resample the discrete distribution $\{\xi_k^j : j = 1, \ldots, H\}$, H times to generate the resampled process noise samples $\{v_{k-1}^i : i = 1, \ldots, H\}$, so that for any i, $\Pr\{v_{k-1}^i = \widehat{v}_{k-1}^j\} = \xi_k^j$. Set the weights ξ_k^j to $1/H$, $i = 1, \ldots, H$.

(5) Calculate the predicted particles at time step k, $\{\widehat{x}_k^i, \ i = 1, \ldots, H\}$,

$$\widehat{x}_k^i = f\left(x_{k-1}^i, v_{k-1}^i\right). \qquad (4.15)$$

(6) and (7) are the same with the stages (3) and (4) in Algorithm 2.1.

Algorithm 4.2 (simplification of the proposed algorithm). In the proposed algorithm, at each iteration, $H * H$ intermediate particles are calculated through the permutation of particles and process noise samples in (4.10) and evaluated via (4.13). This increases the computation burden and the algorithm runs slowly compared to the conventional particle filter, which are based on H particles (Algorithm 2.1). It is observed that at each time step, after resampling the particles focus in some smaller area, a large portion of particles are with the same value (Algorithm 2.1, Step (4)). To simplify the algorithm, it is assumed that the particles (Algorithm 2.1, Step (4)) are less variable compared with the process noise samples (4.10). In (4.10), particles $\{x_{k-1}^i, \ i = 1, \ldots, H\}$ are replaced by $\widetilde{\widehat{x}}_{k-1}$, the estimate of the state at time step $k - 1$, which results in a simplified version of (4.10),

$$\mu_k^j = f\left(\widetilde{\widehat{x}}_{k-1}, \widehat{v}_{k-1}^j\right), \tag{4.16}$$

and $p(z_k \mid \widehat{v}_{k-1}^j)$ is expanded directly on μ_k^j,

$$p\left(z_k \mid \widehat{v}_{k-1}^j\right) = \sum_{\tau=1}^{H} p(z_k \mid \mu_k^\tau) p\left(\mu_k^\tau \mid \widehat{v}_{k-1}^j\right). \tag{4.17}$$

Using the similar derivation process with (4.12), we can obtain

$$p\left(z_k \mid \widehat{v}_{k-1}^j\right) = p\left(z_k \mid \mu_k^j\right). \tag{4.18}$$

In the simplified version of the proposed algorithm, the number of intermediate particles is reduced to H, which reduces the computation burden and increases the computing speed. More importantly, the performance of the algorithm with the simplification procedure is verified through simulation study. In the following sections, the particle filter based process noise identification algorithm refers to the simplified version.

Discussion. It is required to set the process noise bound d preliminarily, which could be estimated based on the maximin uncertain dynamics. For most of the manned air vehicles, the process noise bound d could be determined according to the flight envelop of the aircraft. For the unmanned air vehicles with unexpected maneuvers, d is set large enough to cover the unexpected maneuvers. If our knowledge about d is imprecise, tracking process may fail when coping with the maneuvers that are not covered by the process noise bound d.

5. Multiple Target Tracking Using Particle Filter Based Multiscan JPDAF

The number of targets (M) to be tracked is assumed as fixed and known, where each target track has been initiated, and our objective is to maintain the tracks. Each target is parameterized by a state $x_{m,k}$, where m denotes the mth target and k denotes time step k. The combined state, $x_k = (x_{1,k}, \ldots, x_{M,k})$, is the concatenation of the individual target states. The individual targets are assumed to evolve independently according to the Markovian dynamic models $p_m(x_{m,k} \mid x_{m,k-1})$. The observation vector z_k is composed of multiple sensor

measurements $\{z_{j,k}, \ j = 1, \ldots, N_k\}$, where N_k is the total number of measurements. It is assumed that there are no unresolved measurements (i.e., measurements associated with two or more targets simultaneously); any measurement is either associated with a single target or caused by clutter. Clutter is modeled as independently and identically distributed (IID) with uniform spatial distribution over the surveillance area.

In the particle filter based JPDA algorithm, the distributions of interest are the marginal filtering distributions for each of the targets $p_m(x_{m,k} \mid z_{1:k})$, $m = 1 \cdots M$, and these distributions are approximated with particles, $\{\hat{x}^i_{m,k}, \ i = 1 \cdots H, \ m = 1 \cdots M\}$, and their associated weights $\{w^i_{m,k}, \ i = 1 \cdots H, \ m = 1 \cdots M\}$, as in

$$p_m(x_{m,k} \mid z_{1:k}) = \sum_{i=1}^{H} w^i_{m,k} \delta\left(x_{m,k} - \hat{x}^i_{m,k}\right). \tag{5.1}$$

At each time step, when the new observation vector arrives, the marginal filtering distributions for each of the targets are updated through the Bayesian sequential estimation recursions [24].

In the standard single scan JPDA framework, a track is updated with a weighted sum of the measurements which could have reasonably originated from the target in track. The only information the standard JPDA algorithm uses is the measurements on the present scan and the state vectors. If more scans of measurements are used, additional information is available resulting in better computed probabilities [4]. Since the tracking systems are unable to store all of the measurements from all the scans, a Bayesian tracking system can at best rely on a sliding window of scans. In [28], the single scan JPDA filter is extended to the multiple scan JPDA filter for tracking multiple targets. And in this work the multiple scan JPDA filter is developed in a particle filter framework.

The multiple scan JPDA calculation examines multiple scan joint association events [28]. The measurement to target the association event of a multiple scan is defined as $\lambda_{k-L+1:k}$, where L denotes the length of the multiple scan sliding window. The multiple scan joint association events are mutually exclusive, and they form a complete set $\Lambda_{k-L+1:k}$ [11]. $\lambda_{k-L+1:k}$ is composed by the association vectors at each scan in the sliding window, $\lambda_{k-L+1:k} = (\theta_{k-L+1}, \theta_{k-L+2}, \ldots, \theta_k)$. The elements of the association vector at time step k, $\theta_k = (\zeta_{1,k}, \ldots, \zeta_{j,k}, \ldots, \zeta_{N_k,k})$ are given by

$$\zeta_{j,k} = \begin{cases} 0, & \text{if } z_{j,k} \text{ is due to clutter,} \\ m \in \{1 \cdots M\}, & \text{if } z_{j,k} \text{ is from target } m. \end{cases} \tag{5.2}$$

The heart of the new algorithm is to find the posterior probability for the joint association event of multiple scans. That is to calculate $p(\lambda_{k-L+1:k} \mid z_{1:k})$ and it can be written as

$$p(\lambda_{k-L+1:k} \mid z_{1:k}) \propto p(z_k \cdots z_{k-L+1} \mid \lambda_{k-L+1:k}, z_{1:k-L}) p(\lambda_{k-L+1:k} \mid z_{1:k-L})$$

$$\propto p(z_k \cdots z_{k-L+1} \mid \lambda_{k-L+1:k}, z_{1:k-L}) p(\lambda_{k-L+1:k}), \tag{5.3}$$

where the conditioning of $\lambda_{k-L+1:k}$ on the history of measurements before the sliding window has been eliminated.

The distribution of the measurements in the sliding window based on a specific association event is given by

$$p(z_k \cdots z_{k-L+1} \mid \lambda_{k-L+1:k}, z_{1:k-L}) = \prod_{s=1}^{L} \left[\prod_{j=1}^{N_{k-L+s}} p(z_{j,k-L+s} \mid \lambda_{k-L+1:k}, z_{1:k-L}) \right]. \tag{5.4}$$

To reduce the notation, the index of the scan s in the sliding window is denoted by $k_s = k - L + s$. Then we can obtain

$$\begin{aligned} p(z_k \cdots z_{k-L+1} \mid \lambda_{k-L+1:k}, z_{1:k-L}) &= \prod_{s=1}^{L} \left[\prod_{j=1}^{N_{k_s}} p(z_{j,k_s} \mid \lambda_{k-L+1:k}, z_{1:k-L}) \right] \\ &= \prod_{s=1}^{L} \left[\prod_{j \in I_{0,k_s}} p_c(z_{j,k_s}) \cdot \prod_{j \in I_{k_s}} p\left(z_{j,k_s} \mid x_{\zeta_{j,k_s},k_s}\right) \right] \\ &= \prod_{s=1}^{L} \left[(V)^{-C_{k_s}} \cdot \prod_{j \in I_{k_s}} p\left(z_{j,k_s} \mid x_{\zeta_{j,k_s},k_s}\right) \right], \end{aligned} \tag{5.5}$$

where $I_{0,k_s} = \{j \in \{1,\dots,N_{k_s}\} : \zeta_{j,k_s} = 0\}$ and $I_{k_s} = \{j \in \{1,\dots,N_{k_s}\} : \zeta_{j,k_s} \neq 0\}$ are, respectively, the subsets of measurement indices corresponding to clutter measurements and measurements from the targets being tracked, on scan k_s. p_c denotes the clutter likelihood model, which is assumed to be uniform over the volume of the surveillance area V. The volume of the surveillance area could be calculated as per $V = 2\pi R_{\max}$, where $R_{\max}$ is the maximum range of the sensor. C_{k_s} is defined as the number of clutter measurements.

The joint association prior $p(\lambda_{k-L+1:k})$, can be calculated as in (5.6) according to [4, 16],

$$p(\lambda_{k-L+1:k}) = \prod_{s=1}^{L} \left[\frac{C_{k_s}! \varepsilon}{N_{k_s}!} \prod_{m=1}^{M} (P_D)^{\delta_m(\theta_{k_s})} (1 - P_D)^{1 - \delta_m(\theta_{k_s})} \right], \tag{5.6}$$

where ε is a "diffuse" prior [16] and P_D is the detection probability. $\delta_m(\theta_{k_s})$ is a binary variable and set to one if the mth target is assigned with a measurement in the event θ_{k_s}.

The posterior probability for the joint association event of multiple scans is obtained as

$$p(\lambda_{k-L+1:k} \mid z_{1:k}) \propto p(\lambda_{k-L+1:k}) \prod_{s=1}^{L} \left[(V)^{-C_{k_s}} \cdot \prod_{j \in I_{k_s}} p\left(z_{j,k_s} \mid x_{\zeta_{j,k_s},k_s}\right) \right]. \tag{5.7}$$

The posterior probability that the jth measurement is associated with the mth target on scan k, β_{jm}, is calculated by summing over the probabilities of the corresponding joint association events via

$$\beta_{jm} = p(\zeta_{j,k} = m \mid z_{1:k}) = \sum_{\{\lambda_{k-L+1:k} \in \Lambda_{k-L+1:k} : \zeta_{j,k} = m\}} p(\lambda_{k-L+1:k} \mid z_{1:k}). \tag{5.8}$$

These approximations can, in turn, be used in (5.9) to approximate the target likelihood according to [24],

$$p_m(z_k \mid x_{m,k}) = \beta_{0m} + \sum_{j=1}^{N_k} \beta_{jm} p(z_{j,k} \mid x_{m,k}), \tag{5.9}$$

where β_{0m} is the posterior probability that the mth target is undetected. Finally, setting the new importance weights to

$$w_{m,k}^i \propto w_{m,k-1}^i \frac{p_m\left(z_k \mid \widehat{x}_{m,k}^i\right) p_m\left(\widehat{x}_{m,k}^i \mid x_{m,k-1}^i\right)}{q_m\left(\widehat{x}_{m,k}^i \mid x_{m,k-1}^i\right)},$$

$$\sum_{i=1}^{H} w_{m,k}^i = 1, \tag{5.10}$$

where $q_m(\widehat{x}_{m,k}^i \mid x_{m,k-1}^i)$ is the proposal distribution, which is used to generate the predicted particles. Equation (5.10) leads to the sample set $\{w_{m,k}^i, \widehat{x}_{m,k}^i\}_{i=1}^{H}$ being approximately distributed according to the marginal filtering distribution $p_m(x_{m,k} \mid z_{1:k})$.

A summary of the particle filter based multiple scan JPDA filter algorithm is presented in what follows. Assuming that the sample sets $\{w_{m,k-1}^i, x_{m,k-1}^i\}_{i=1}^{H}$, $m = 1 \cdots M$, are approximately distributed according to the corresponding marginal filtering distributions at the previous time step $p_m(x_{m,k-1} \mid z_{1:k-1})$, $m = 1 \cdots M$, the algorithm proceeds as follows at the current time step.

Algorithm 5.1 (particle filter based multiscan JPDA filter). (1) For $m = 1 \cdots M$, $i = 1 \cdots H$, generate predicted particles for the target states $\widehat{x}_{m,k}^i \sim q_m(\widehat{x}_{m,k}^i \mid x_{m,k-1}^i)$.

(2) For $m = 1 \cdots M$, calculate $\widetilde{x}_{m,k}$, the preapproximation of $x_{m,k}$, which is to be substituted into (5.7) to calculate the posterior probability of the joint association event in multiple scan, $p(\lambda_{k-L+1:k} \mid z_{1:k})$. Consider

$$\widetilde{x}_{m,k} = \sum_{i=1}^{H} w_{m,k-1}^i \widehat{x}_{m,k}^i. \tag{5.11}$$

(3) Enumerate all the valid joint measurement to target association events in the sliding window $k - L + 1 : k$ to form the set $\Lambda_{k-L+1:k}$.

(4) For each $\lambda_{k-L+1:k} \in \Lambda_{k-L+1:k}$, compute the posterior probability of the joint association event in multiple scan, $p(\lambda_{k-L+1:k} \mid z_{1:k})$, via (5.7).

(5) For $m = 1 \cdots M$, $j = 1 \cdots N_k$, compute the marginal association posterior probability, β_{jm}, via (5.8).

(6) For $m = 1 \cdots M$, $i = 1 \cdots H$, compute the target likelihood, $p_m(z_k \mid \widehat{x}_{m,k}^i)$, via (5.9).

(7) For $m = 1 \cdots M$, $i = 1 \cdots H$, compute and normalize the particle weights, via (5.10).

(8) Resample the discrete distribution $\{w^i_{m,k} : i = 1,\ldots,H\}$, H times to generate particles $\{x^j_{m,k} : j = 1,\ldots,H\}$, so that for any j, $\Pr\{x^j_{m,k} = \hat{x}^i_{m,k}\} = w^i_{m,k}$. Set the weights $w^i_{m,k}$ to $1/H$, $i = 1,\ldots,H$, and move to Stage 1.

6. An Algorithm for Tracking Multiple Maneuvering Targets

The proposed multiple maneuvering target tracking algorithm is a combination of the particle filter based process noise identification algorithm (proposed in Section 4) and the particle filter based multiscan JPDA algorithm (proposed in Section 5). In the proposed algorithm, firstly the particle filter based process noise identification algorithm is used to estimate the maneuvering movement for each target. Then the particles of each target model are propagated to the next time step based on the new distributed process noise samples to obtain the predicted particles. In the update process, each predicated particle of one target model is assigned with a weight based on the multiscan JPDA process. The steps of the proposed multiple maneuvering target tracking algorithm are listed in the following.

Algorithm 6.1 (multiple maneuvering target tracking). (1) At time step $k - 1$, for $m = 1 \cdots M$,

(a) draw process noise samples $\{\hat{v}^j_{m,k-1} : j = 1,\ldots,H\}$ from a uniform distribution $U(-d,d)$,

(b) calculate the intermediate particles $\{\mu^j_{m,k} : j = 1,\ldots,H\}$ according to

$$\mu^j_{m,k} = f\left(\tilde{\tilde{x}}_{m,k-1}, \hat{v}^j_{m,k-1}\right), \tag{6.1}$$

(c) calculate the process noise sample weights $\{\xi^j_{m,k} : j = 1,\ldots,H\}$ via (6.2) and normalize each weight:

$$\xi^j_{m,k} = p\left(z^m_k \mid \mu^j_{m,k}\right), \tag{6.2}$$

where z^m_k are the measurements that are close to $\mu^j_{m,k}$ obtained from the nearest neighbor method,

(d) resample the discrete distribution $\{\xi^j_{m,k} : j = 1,\ldots,H\}$, H times to generate the new process noise samples $\{v^i_{m,k-1} : i = 1,\ldots,H\}$, so that for any i, $\Pr\{v^i_{m,k-1} = \hat{v}^j_{m,k-1}\} = \xi^j_{m,k}$. Set the weights $\xi^j_{m,k}$ to $1/H$, $i = 1,\ldots,H$,

(e) obtain the predicted particles $\{\hat{x}^i_{m,k} : i = 1,\ldots,H\}$ at time step k from the new process noise samples via

$$\hat{x}^i_{m,k} = f\left(x^i_{m,k-1}, v^i_{m,k-1}\right). \tag{6.3}$$

(2) At time step k, for $m = 1 \cdots M$, calculate $\tilde{x}_{m,k}$, the postapproximation of $x_{m,k}$,

$$\tilde{x}_{m,k} = \sum_{i=1}^{H} \xi_{m,k}^{i} \hat{x}_{m,k}^{i}, \tag{6.4}$$

where $\{\xi_{m,k}^{i} : i = 1, \ldots, H\}$ are the process noise samples' weights and are calculated based on the measurements at the current time step (k). However, in (5.11) $\tilde{x}_{m,k}$ is calculated based on $\{w_{m,k-1}^{i} : i = 1, \ldots, H\}$, which rely on the measurements at the previous time step $(k-1)$. As a result, in maneuvering target tracking, $\tilde{x}_{m,k}$ calculated via (6.4) is closer to the true state than that via (5.11) since the information from the current time step is considered in (6.4).

(3)–(8) are the same with the stages (3)–(8) in Algorithm 5.1.

(3) Thus we can obtain $\tilde{\tilde{x}}_{m,k} = \sum_{i=1}^{H} w_{m,k}^{i} \hat{x}_{m,k}^{i}$. $\tilde{\tilde{x}}_{m,k}$ is the estimate of the true state $x_{m,k}$.

7. Simulation Results and Analysis

This section consists of three parts. In the first and second part, the particle filter based process noise identification algorithm and the particle filter based multiscan JPDA algorithm are, respectively, used to track single maneuvering targets and two slow-maneuvering targets in clutter. In the third part, the combination of the two algorithms are used to track two highly maneuvering, at times closely spaced and crossed, targets.

7.1. Single Maneuvering Target Tracking

The simulation study using nearly coordinate turn model is performed. The maneuvering target tracking is done by setting up a 2D flight path in x-y plane, which is similar to the path considered in [16]. At time step 1, the target starts at location $[-310, 310]$ in Cartesian coordinates in meters with initial velocity (in m/s) $[10, -400]$. The following trajectory is considered: a straight line with constant velocity between 1 and 17 s, a coordinated turn (0.09 rad/s) between 17 and 34 s, a straight line with constant velocity between 34 and 51 s, a coordinated turn (0.09 rad/s) between 51 and 68 s, and a straight line with constant velocity between 68 and 100 s.

In the particle filter with process noise identification, a general model

$$X_k = \Phi X_{k-1} + \Gamma v_{k-1} \tag{7.1}$$

is adopted during the whole tracking process, where in (7.1),

$$\Phi = \begin{bmatrix} \Phi_b & 0 \\ 0 & \Phi_b \end{bmatrix}, \qquad \Phi_b = \begin{bmatrix} 1 & \Delta T & \dfrac{\Delta T^2}{2} \\ 0 & 1 & \Delta T \\ 0 & 0 & 1 \end{bmatrix}, \tag{7.2}$$

$$\Gamma = I_{6 \times 6}.$$

Matrix Φ is the transition matrix and ΔT is the sample interval. $X_k = [p_x, \gamma_x, a_x, p_y, \gamma_y, a_y]_k^T$ is the state vector; $p_x, \gamma_x,$ and a_x denote, respectively, the position, velocity, and acceleration
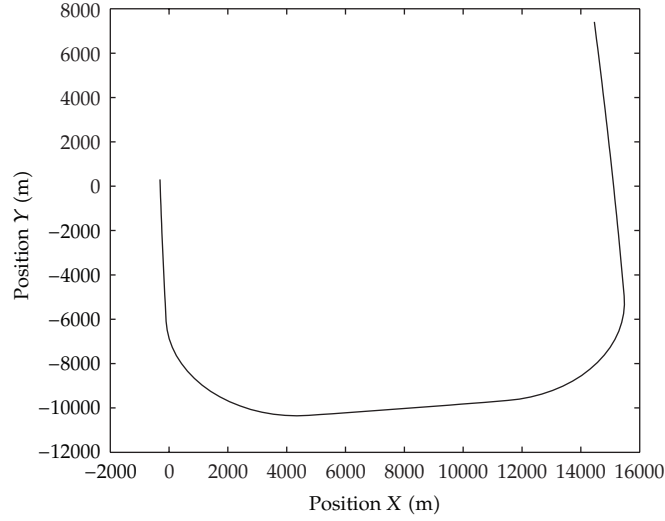
Figure 1: True trajectory of the single maneuvering target.

of the moving object along the x-axis of Cartesian frame and, p_y, γ_y, and a_y along the y-axis. The equivalent process noise, $v_{k-1} = [v_{p_x}, v_{\gamma_x}, v_{a_x}, v_{p_y}, v_{\gamma_y}, v_{a_y}]_{k-1}^T$, with unknown statistics is required to be identified. The bound of the process noise (d), which accounts for the uncertain dynamics, is chosen as $\{20\,\text{m}, 20\,\text{m/s}, 10\,\text{m/s}^2, 20\,\text{m}, 20\,\text{m/s}, 10\,\text{m/s}^2\}$. The number of the process noise samples is equal to the number of particles, which is set as 500. The algorithm is initialized with Gaussian around the initial state of the true target, and the standard deviation of the Gaussian distribution is chosen as $\{10\,\text{m}, 10\,\text{m/s}, 5\,\text{m/s}^2, 10\,\text{m}, 10\,\text{m/s}, 5\,\text{m/s}^2\}$.

A track-while-scan (TWS) radar is positioned at the origin of the plane. The measurement equation is as follows:

$$Z_k = h(X_k) + n_k, \tag{7.3}$$

where $Z_k = [z_1, z_2]_k$ is the observation vector. z_1 is the distance between the radar and the target, and z_2 is the bearing angle. The measurement noise $n_k = [n_{z_1}, n_{z_2}]_k$ is a zero mean Gaussian white noise process with standard deviations of $20\,\text{m}$ (σ_{z_1}) and $0.01\,\text{rad}$ (σ_{z_2}). Resolution of the sensor is selected after from [29] (twice of the standard deviations of the measurement noise). The sampling interval is $\Delta T = 1\,\text{s}$.

The particle filter based process noise identification algorithm is compared to the IMM filter and the regularized particle filter. The IMM filter consists of three extended Kalman filter (EKF) with different motion models. The details regarding these models may be found in [16]. The initial model probabilities and the mode switching probability matrix are set the same values as in [16]. For the regularised particle filter, Epanechnikov kernel is chosen as the rescaled kernel density, which is the same as in [30]. Moreover, the proposed algorithm is compared to its complete version in the same simulation setup.

The simulation results are obtained from 1000 Monte Carlo runs except the results from the complete version of the particle filter based multiscan JPDA algorithm, which is run for 100 times. Figure 1 shows the true trajectory of the maneuvering target. The root mean-square errors (RMSEs) in position at each time step, respectively, using the four methods are shown
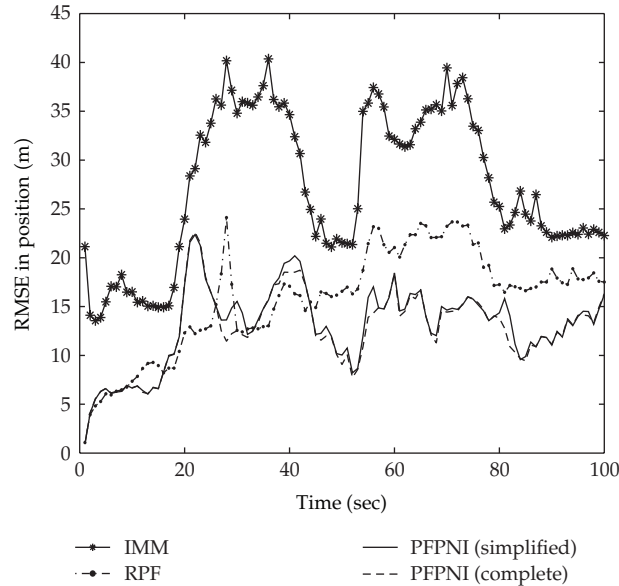
Figure 2: RMSE in position using IMM, RPF, PFPNI (simplified), and PFPNI (complete) algorithms.

Table 1: Performance comparison.

	RMSE (m)	ET (s)	TLR
IMM	28.34	0.0239	3.7%
RPF	16.66	0.8708	0
PFPNI (simplified)	13.67	0.4503	0
PFPNI (complete)	13.29	190.47	0

in Figure 2, where RPF and PFPNI represent the regularised particle filter algorithm and the particle filter based process noise identification algorithm. The performance of the methods is also compared via the global RMSE (in position), the tracking loss rate (TLR), and the executing time (ET), which are listed in Table 1. The tracking loss rate (TLR) is defined as the ratio of the number of simulations, in which the target is lost in track, to the total number of simulations carried out. The target is defined as lost in track when its global RMSE in position is larger than the ten times of the magnitude of the standard deviation in position. The executing time (ET) is the CPU time needed to execute one time step in MATLAB 7.1 on a 3 GHz (Mobile) Pentium IV operating under Windows 2000.

From the simulation results, it can be seen that the simplified version of the proposed particle filter based process noise identification algorithm outperforms the IMM filter and the regularised particle filter algorithm, with computing time within the limits of practically realizable systems. Moreover, the proposed algorithm needs neither the possible multiple motion models nor the transition probability matrices, which makes it a more general algorithm for maneuvering target tracking. From the simulation results, it can also be seen that the complete version of the particle filter based process noise identification algorithm is not suitable for practical application due to the long computing time, though it gains a 2.8% increase in accuracy (RMSE) compared with its simplified version.

Table 2: Performance comparison.

	RMSE (m)	ET (s)	TLR	SR
JPDA	T1: 20.24, T2: 22.87	0.07	24%	0
PFJPDA(1)	T1: 13.21, T2: 11.36	1.28	0	0
PFJPDA(2)	T1: 12.92, T2: 11.01	1.54	0	0
PFJPDA(3)	T1: 12.75, T2: 10.67	1.8	0	0

7.2. Multiple Target Tracking

The simulation is carried out for tracking two slow-maneuvering targets in clutter. The Wiener process acceleration model (7.1) is chosen as the motion model for the two targets. The process noise, $v_{k-1} = [v_{p_x}, v_{\gamma_x}, v_{a_x}, v_{p_y}, v_{\gamma_y}, v_{a_y}]_{k-1}^T$, is a zero mean Gaussian white noise process with standard deviations of $1\,\mathrm{m}(\sigma_{v_{p_x}})$, $1\,\mathrm{m/s}(\sigma_{v_{\gamma_x}})$, $20\,\mathrm{m/s^2}(\sigma_{v_{a_x}})$, $1\,\mathrm{m}(\sigma_{v_{p_y}})$, $1\,\mathrm{m/s}(\sigma_{v_{\gamma_y}})$, and $20\,\mathrm{m/s^2}(\sigma_{v_{a_y}})$.

At time step 1, target one starts at location $[-310, 310]$ in x-y Cartesian coordinates in meters with the initial velocity (in m/s) $[10, -400]$. Target two starts at location (in m) $[-310, -19000]$ with the initial velocity (in m/s) $[10, 400]$. The length of each simulation run is 50 seconds. A TWS radar is positioned at the origin of the plane, whose details are provided in Section 7.1.

The sampling interval is $\Delta T = 1\,\mathrm{s}$ and it is assumed that the probability of detection $P_D = 0.9$ for the radar. For generating measurements in simulations, the clutter is assumed uniformly distributed with density $1 \times 10^{-6}/\mathrm{m^2}$.

In the particle filter based multiscan JPDA methods, each target model is assigned with 500 particles. The algorithm is initialized with Gaussians around the initial states of the true targets, and the standard deviations of the two Gaussian distributions are chosen equally as $\{10\,\mathrm{m}, 10\,\mathrm{m/s}, 5\,\mathrm{m/s^2}, 10\,\mathrm{m}, 10\,\mathrm{m/s}, 5\,\mathrm{m/s^2}\}$.

In the simulations carried out, the length of the multiple scan sliding window (L) varies from 1 to 3. The corresponding particle filter based multiscan JPDA methods with $L = 1, 2, 3$ are utilized to track two slow-maneuvering targets individually. A comparison to the standard JPDA filter is also studied on the same simulation setup. The simulation results are obtained from 1000 Monte Carlo runs. Figure 3 shows the true trajectories of the two targets and Figure 4 shows the distance between the two targets along time, through which we can see that the two targets reach the smallest distance at time step 25. The RMSEs in position for the two targets are, respectively, shown in Figures 5 and 6, where PFJPDA represents the particle filter based JPDA filter with its following number in the bracket (e.g., (2.1)) denoting the scan number (L). The performance of the four methods is also compared in Table 2. The swap rate (SR) is defined as the ratio of the number of simulations, in which the two targets swap, to the total number of simulations.

Compared with the standard JPDA method (based on extended Kalman filter), the particle filter based JPDA methods (single scan and multiple scan) are much more accurate and robust, at the cost of longer computing time. This verifies that when dealing with nonlinear problem (nonlinear observation equation) and large random acceleration (large process noise), the performance of particle filter is better than extended Kalman filter using local linearization.

In the comparison between the three particle filter based JPDA methods ($L = 1, 2, 3$, resp.), from Figures 5 and 6 it can be seen that there is no significant deference between the three methods except around time step 25, when the two targets are very close to each
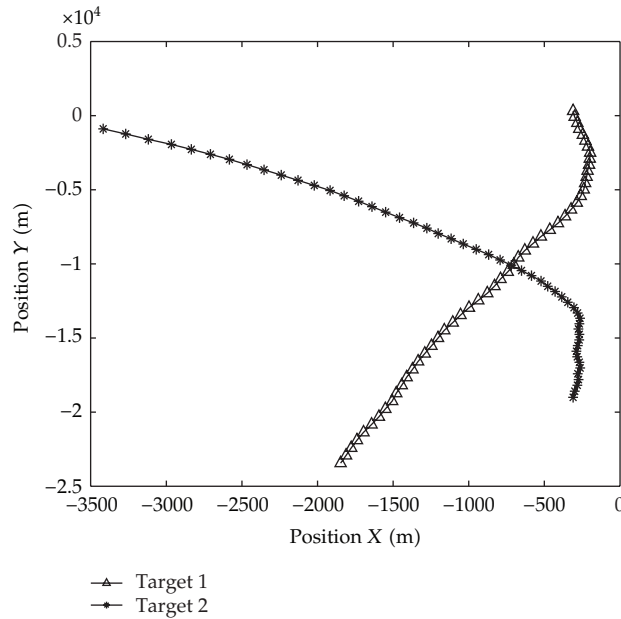
Figure 3: True trajectories of the two slow-maneuvering targets.
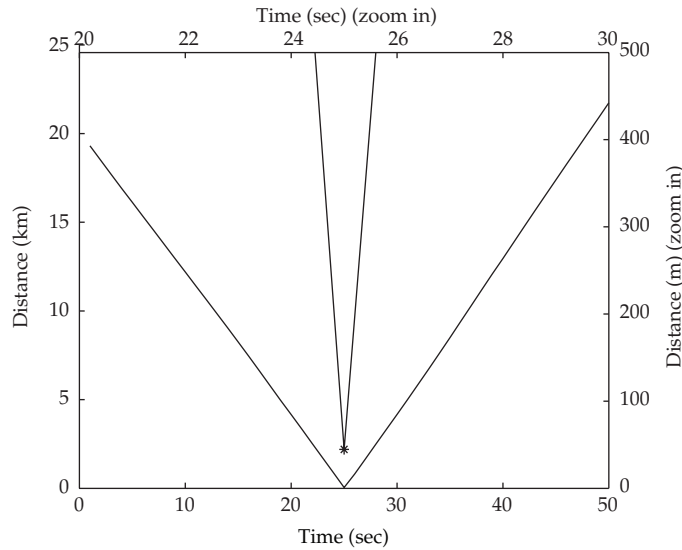

Figure 4: Distance between the two targets (- distance; -* zoom in distance).

other (Figure 4). As the scan number increases, the RMSE (in position) of the corresponding algorithm decreases significantly around time step 25 (Figures 5 and 6). This verified that the particle filter based multiscan JPDA method provides a better performance especially when the targets are very close. The additional information of more scans improve the association probabilities in such critical situation, resulting in lower estimation errors (RMSE) and larger robustness (tracking loss rate).
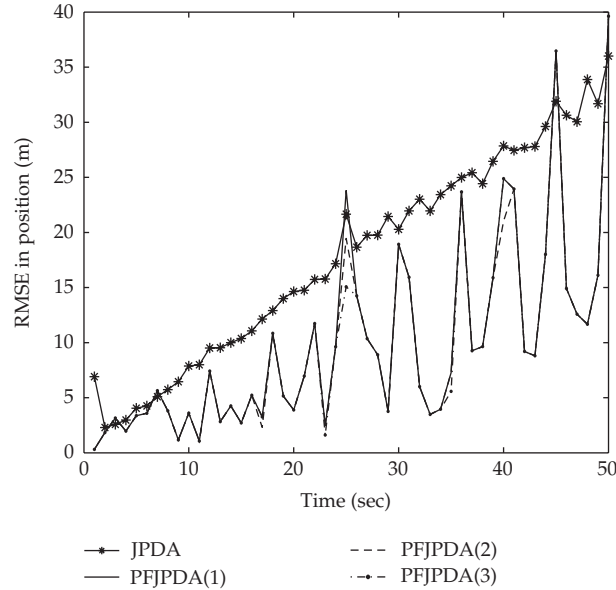
Figure 5: Target 1: RMSE in position using standard JPDA filter, particle filter based multiscan JPDA algorithms with a scan number equals to 1, 2, and 3, respectively.
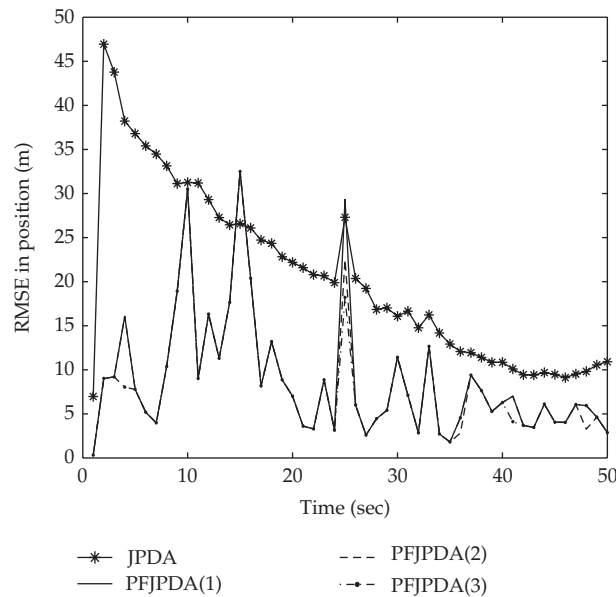
Figure 6: Target 2: RMSE in position using standard JPDA filter, particle filter based multiscan JPDA algorithms with a scan number equals to 1, 2, and 3, respectively.

7.3. Multiple Maneuvering Target Tracking

We now consider tracking two highly maneuvering targets in clutter. The true trajectories are shown in Figure 7 in x-y plane. The velocities of the targets are shown in Figure 8, where VX and VY represent the velocity, respectively, in x and y coordinate. The distance between
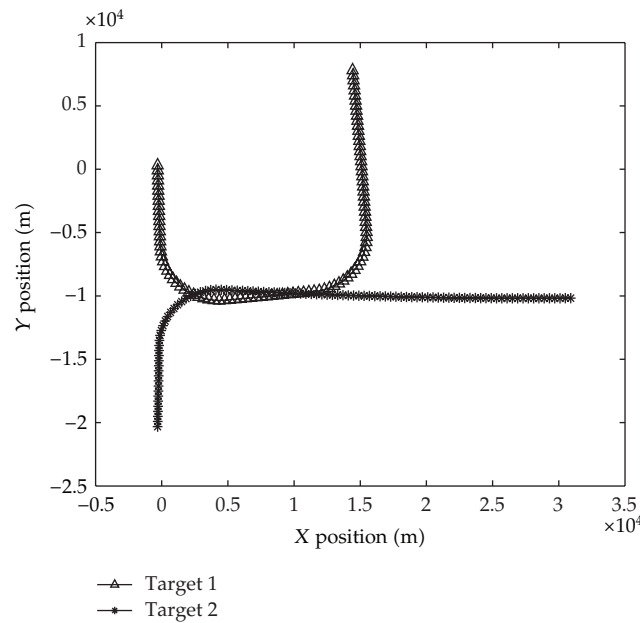
Figure 7: True trajectories of the two maneuvering targets.
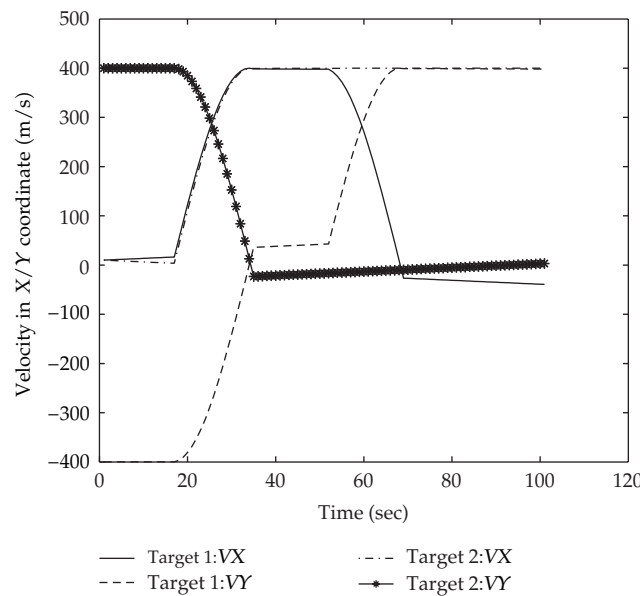
Figure 8: True velocities of the two maneuvering targets.

the two targets is shown in Figure 9. Target one is with the same setup as in the example proposed in Section 7.1. Target two starts at location $[-310, -20310]$ in Cartesian coordinates in meters with the initial velocity (in m/s) $[10, 400]$. Its trajectory is a straight line with constant velocity between 1 and 17 s, a coordinated turn (0.09 rad/s) between 17 and 34 s, and a straight line constant velocity between 34 and 100 s.
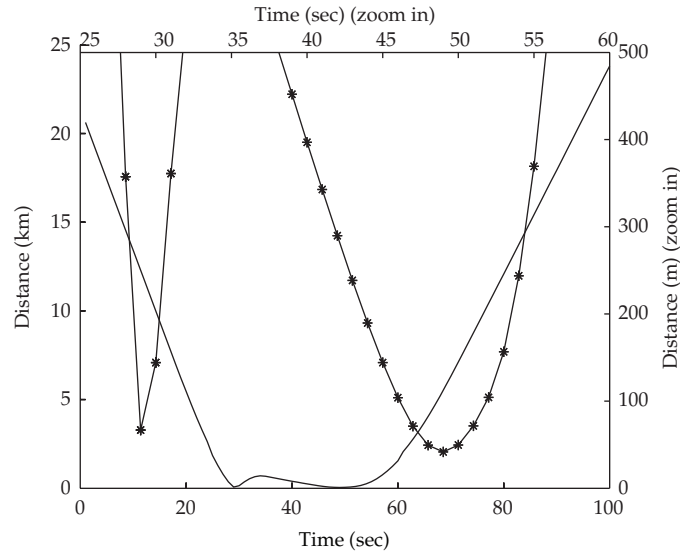
Figure 9: Distance between the two targets (- distance; -* zoom in distance).

A TWS radar is positioned at the origin of the plane (refer to Section 7.1). The sampling interval (ΔT), the probability of detection (P_D), and the clutter density are the same as in Section 7.2.

In the proposed method, each target model is assigned with 1000 particles. The length of the multiple scan sliding window (L) is chosen equal to 3. The bound of the process noise (d) is the same as in Section 7.1. The algorithm is initialized with Gaussians around the initial states of the true targets, and the standard deviations of the two Gaussian distributions are chosen equally as $\{10\,\mathrm{m}, 10\,\mathrm{m/s}, 5\,\mathrm{m/s^2}, 10\,\mathrm{m}, 10\,\mathrm{m/s}, 5\,\mathrm{m/s^2}\}$.

In this work, the proposed method is compared to an IMMJPDA method and a multiscan IMMJPDA method, whose details could be found in [16, 20], respectively.

The simulation results are obtained from 1000 Monte Carlo runs. The RMSEs (resp. in position and velocity) at each time step for the two targets are shown in Figures 10, 11, 12, and 13, where MS-IMMJPDA and PFPNI-PFMSJPDA represent, respectively, the multiscan IMMJPDA algorithm and the proposed algorithm. The performance comparison is listed in Table 3. Moreover, for the proposed algorithm, the influence of the particle number in its performance is studied and simulations are carried out based on different sample sizes. The results are listed in Table 4.

The simulation results show that the proposed algorithm is more accurate than the multiple scan IMMJPDA filter and IMMJPDA filter, though it takes longer computing time. However, the three algorithms are implemented using Matlab. The computing time is considerably reduced when coded in C++. For the proposed method, it takes about 0.8 second for one time step in C++.

In the scene of simulation, at the period of 46 s ~ 52 s, the two targets are very near to each other, which easily results in track swap. In the IMMJPDA method, the judgement of which measurements belong to which target depends on the information from the current scan. If the measurements from two targets are very close, it is hard to distinguish the different targets based on the measurements from a single scan, which leads to a track swap. In the proposed method and the multiple scan JPDA method, the information from
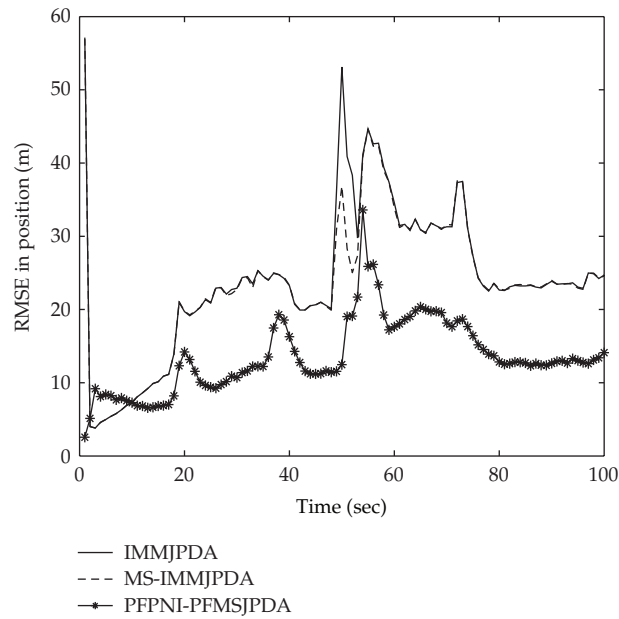
Figure 10: Target 1: RMSE in position using IMMJPDA filter, multiple scan IMMJPDA filter, and the proposed method.
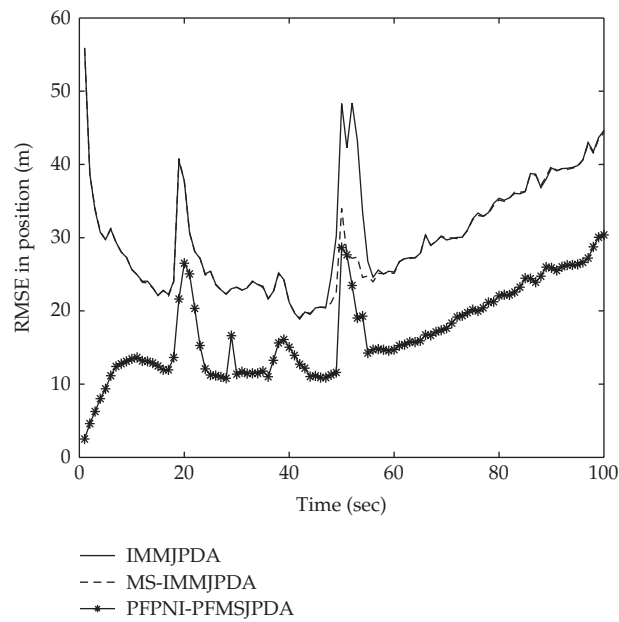

Figure 11: Target 2: RMSE in position using IMMJPDA filter, multiple scan IMMJPDA filter, and the proposed method.
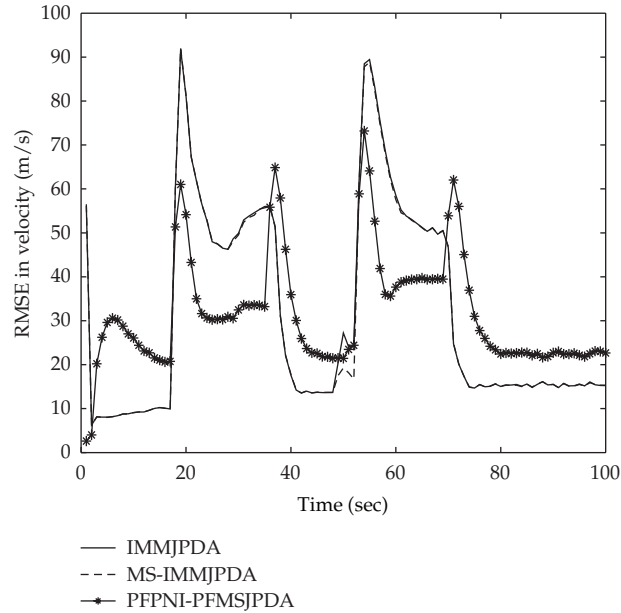
Figure 12: Target 1: RMSE in velocity using IMMJPDA filter, multiple scan IMMJPDA filter, and the proposed method.

Table 3: Performance comparison.

	RMSE in position (m)	RMSE in velocity (m/s)	ET (s)	TLR	SR
IMMJPDA	T1: 26.04, T2: 31.28	T1: 39.48, T2: 31.98	0.75	8.7%	13.9%
MS-IMMJPDA	T1: 25.30, T2: 30.26	T1: 39.22, T2: 31.76	3.78	0	4.5%
PFPNI-PFMSJPDA	T1: 14.42, T2: 18.14	T1: 34.44, T2: 30.58	4.0	0	3.7%

Table 4: Influence of particle number in the performance of the proposed algorithm for tracking multiple maneuvering target.

Particle number	RMSE in position (m)	ET (s)	TLR
200	NA	NA	100%
500	T1: 17.37, T2: 21.22	2.9	35.9%
1000	T1: 14.42, T2: 18.14	4.0	0
2000	T1: 13.75, T2: 17.93	20.2	0

several previous scans is combined with the information from current scan to calculate the association probabilities. The decision of which measurements belong to which target based on the information from multiple scans will be more accurate than single scan method, which reduces the swap rate effectively.

From Table 4, it can be seen that when the number of particles is increased, the performance of the proposed algorithm also increases. But when the number of particles exceeds some threshold (1000), the increase in the performance slows down.
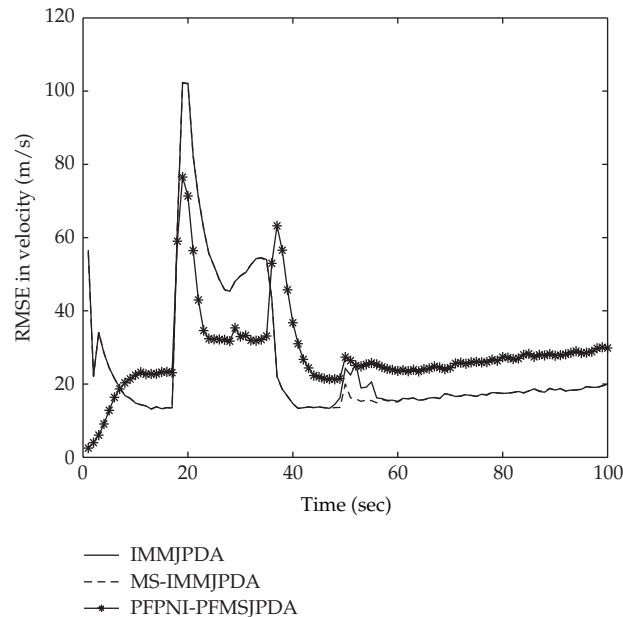
Figure 13: Target 2: RMSE in velocity using IMMJPDA filter, multiple scan IMMJPDA filter, and the proposed method.

8. Conclusions

A new algorithm is proposed for multiple maneuvering target tracking in the particle filter framework. In order to track the highly maneuvering target, the particle filter based process noise identification method is proposed to estimate the equivalent process noise induced by both maneuvering and random acceleration. Compared with the multiple model based methods for maneuvering target tracking, only one general model is adopted in the proposed method. In order to tackle the data association problem in multiple maneuvering target tracking, the particle filter based multiscan JPDA filter is adopted. Compared with the single scan JPDA method, the multiscan JPDA method uses richer information, which results in better estimated probabilities.

Acknowledgments

References

[1] H. A. P. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with a jump-linear smoothing application," *IEEE Transaction on Automatic Control*, vol. 33, no. 8, pp. 780–783, 1988.

[2] Y. Bar-Shalom, K. Chang, and H. Blom, "Tracking a maneuvering target using input estimation versus the interacting multiple model algorithm," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 25, no. 2, pp. 296–300, 1989.

[3] A. Munir and P. Artherton, "Adaptive interacting multiple model algorithm for tracking a maneuvering target," *IEE Proceedings Radar, Sonar and Navigation*, vol. 142, no. 1, pp. 11–17, 1995.

[4] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*, vol. 179, Academic Press, Orlando, Fla, USA, 1988.

[5] Y. Bar-Shalom and X.-R. Li, *Estimation and Tracking*, Artech House, Norwood, NJ, USA, 1993.

[6] D. Musicki, R. Evans, and S. Stanković, "Integrated probabilistic data association," *IEEE Transactions on Automatic Control*, vol. 39, no. 6, pp. 1237–1241, 1994.

[7] D. Mušicki and R. Evans, "Integrated probabilistic data association—finite resolution," *Automatica*, vol. 31, no. 4, pp. 559–570, 1995.

[8] D. Mušicki and R. Evans, "Clutter map information for data association and track initialization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 2, pp. 387–398, 2004.

[9] Y. Bar-Shalom and X. Li, *Multitarget-Multisensor Tracking: Principlesand Techniques*, YBS Publishing, Storrs, Conn, USA, 1995.

[10] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE Journal of Oceanic Engineering*, vol. 8, pp. 173–184, 1983.

[11] D. Mušicki and R. Evans, "Joint integrated probabilistic data association: JIPDA," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 3, pp. 1093–1099, 2004.

[12] A. Houles and Y. Bar-Shalom, "Multisensor tracking of a maneuvering target in clutter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 25, pp. 176–188, 1989.

[13] Y. Bar-Shalom, K. Chang, and B. H. A. Blom, "Tracking splitting targets in clutter by using an interacting multiple model joint probabilistic data association filter," in *Multitarget Multisensor Tracking: Applications and Advances*, Y. Bar-Shalom, Ed., vol. 2, pp. 93–110, Artech House, 1992.

[14] M. de Feo, A. Graziano, R. Miglioli, and A. Farina, "IMMJPDA versus MHT and Kalman filter with NN correlation: performance comparision," *IEE Proceedings—Radar Sonar and Navigation*, vol. 144, pp. 49–56, 1997.

[15] D. Musicki and S. Suvorova, "Tracking in clutter using IMM-IPDA Based algorithms," *IEEE Transaction on Aerospace and Electronic Systems*, vol. 44, pp. 111–126, 2008.

[16] B. Chen and J. K. Tugnait, "Tracking of multiple maneuvering targets in clutter using IMM/JPDA filtering and fixed-lag smoothing," *Automatica*, vol. 37, no. 2, pp. 239–249, 2001.

[17] H. A. P. Blom and E. A. Bloem, "Combining IMM and JPDA for tracking multiple maneuvering targets in clutter," in *Proceedings of the 5th International Conference on Information Fusion*, vol. 1, pp. 705–712, July 2002.

[18] H. A. P. Blom and E. A. Bloem, "Interacting multiple model joint probabilistic data association avoiding track coalescence," in *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 3, pp. 3408–3415, December 2002.

[19] H. A. P. Blom and E. A. Bloem, "Probabilistic data association avoiding track coalescence," *IEEE Transactions on Automatic Control*, vol. 45, no. 2, pp. 247–259, 2000.

[20] S. Puranik and J. K. Tugnait, "Tracking of multiple maneuvering targets using multiscan JPDA and IMM filtering," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 1, pp. 23–35, 2007.

[21] H. A. P. Blom and E. A. Bloem, "Exact Bayesian filter and joint IMM coupled PDA tracking of maneuvering targets from possibly missing and false measurements," *Automatica*, vol. 42, no. 1, pp. 127–135, 2006.

[22] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, pp. 174–188, 2002.

[23] H. A. P. Blom and E. A. Bloem, "Joint particle filtering of multiple maneuvering targets from unassociated measurements," *Journal of Advances in Information Fusion*, vol. 1, no. 1, pp. 15–34, 2006.

[24] J. Vermaak, S. J. Godsill, and P. Pérez, "Monte Carlo filtering for multitarget tracking and data association," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, pp. 309–332, 2005.

[25] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*, Academic Press, New York, NY, USA, 1970.

[26] X. R. Li and Y. Bar-Shalom, "A recursive multiple model approach tonoise identification," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 30, no. 3, pp. 671–684, 1994.

[27] X. R. Li and Y. Bar-Shalom, "A recursive hybrid system approach to noise identification," in *Proceedingsof the 1st IEEE Conference on Control Applications*, pp. 847–852, 1992.

[28] J. Roecker, "Multiple scan joint probabilistic data association," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 31, pp. 1204–1209, 1995.

[29] W. Koch and G. Keuk, "Multiple hypothesis track maintenance with possibly unresolved measurements," *IEEE Transaction on Aerospace and Electronic Systems*, vol. 33, pp. 883–892, 1997.

[30] C. Musso, N. Oudjane, and F. LeGland, "Improving regularized particle filters," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. DeFreitas, and N. Gordon, Eds., pp. 247–271, Springer, New York, NY, USA, 2001.