Process Noise Identification Based Particle Filter: an Efficient Method to Track Highly Maneuvering Target

Liu Jing
School of Electronics
and Information Engineering,
Xi'an JiaoTong University,
Xi'an, P.R.China.
elelj20080730@mail.xjtu.edu.cn

Han ChongZhao
School of Electronics
and Information Engineering,
Xi'an JiaoTong University,
Xi'an, P.R.China.
czhan@mail.xjtu.edu.cn

Prahlad Vadakkepat
Department of Electrical
and Computer Engineering,
National University of Singapore,
Singapore.
elepv@nus.edu.sg

Abstract – *In this paper, a novel method, process noise identification based particle filter is proposed for tracking highly maneuvering target. In the proposed method, the equivalent-noise approach [1], [2], [3] is adopted, which converts the problem of maneuvering target tracking to that of state estimation in the presence of non-stationary process noise with unknown statistics. A novel method for identifying the non-stationary process noise is proposed in the particle filter framework. Compared with the multiple model approaches for maneuvering target tracking, the proposed method needs to know neither the possible multiple models nor the transition probability matrices. One simple dynamic model is adopted during the whole tracking process. The proposed method is especially suitable for tracking highly maneuvering target due to its capability of dealing with sample impoverishment, which is a common problem in particle filter and becomes serious when tracking large uncertain dynamics.*

Keywords: Particle filter, process noise identification, maneuvering target tracking, sample impoverishment

1 Introduction

Particle filter, which uses sequential Monte Carlo methods for on-line learning within a Bayesian framework, can be applied to any state-space models. Particle filter is more suitable than Kalman filter and extended Kalman filter when dealing with non-linear and non-Gaussian estimation problems. The application of particle filter in maneuvering target tracking has been paid attention only in recent years [4, 5, 6, 7, 8, 9, 10, 11].

Recently, several approaches, which use multiple models to describe the changing maneuvering model, have been proposed in the particle filter framework. One of the methods is based on the auxiliary particle filter. In [6], Karlsson used an auxiliary particle filter to track a highly maneuvering target. In this method, each particle is split deterministically into a number of possible maneuver hypotheses with each hypothesis corresponding to a specific model.

Other methods focus on how to switch between different motion models. In [7], Bayesian switching structure is chosen as the principle which determines switching between different models. A set of models are utilized to cope with the unknown maneuver. Moreover, to deal with non-Gaussian noise, Cauchy distribution is used as the system noise distribution. In [8] and [9], the maneuvering target tracking system is treated as a jump Markov linear system. MCMC process is used as the selection scheme to choose the motion model from a set of candidate models at some specific time step.

However, in the above approaches [6, 7, 8, 9], the possible motion models and transition probability matrices are assumed as known. In practice, the dynamics is hard to break up into several different motion models and the model transition probabilities are difficult to obtain. A general model is needed to cope with the wide variety of motions exhibited by the maneuvering target.

For the single model based methods, Karlsson [16] and Ikoma [17] applied optimal recursive Bayesian filters directly to the nonlinear target model. Both the algorithms are based on the assumption of small amplitude of maneuvers and could not cope with large maneuvers. That reflects that the standard particle filter should be improved to deal with the wide variation of the maneuvering movement. The algorithm may fail due to sample impoverishment when tracking wide variation in maneuvering movements.

In the generic particle filter algorithm, the optimal proposal distribution is $p(x_k|x_{k-1}^i, z_k)$, which is based on both the previous state and the current measurements [12]. However, in the standard particle filter, the prior distribution $p(x_k|x_{k-1}^i)$ is chosen as the proposal distribution for simplicity. In the standard particle filter algorithm, at each time step the predicted particles are generated from the prior distribution and could not catch the latest variation of the state variables, especially when the target is carrying out large maneuvers induced by fast changing motion models. This might be due to the reason that the current measurements are not considered in the prior distribution. As a result, most of the particles are assigned with low weights and eliminated via the resampling process. This leads to serious sample impoverishment and then the tracking process fails.

There have been some systematic techniques proposed recently to solve the problem of sample impoverishment. One such technique is regularized particle filter [18], which resamples from a continuous approximation of the posterior density of the target state, whereas the standard particle lter resamples from the discrete approximation of the posterior density. This approach is frequently found to improve performance, despite a less rigorous derivation. An alternative solution to the same problem is the resample-move algorithm [19]. This technique uses periodic MCMC steps to diversify particles in an importance sampling-based particle lter. It does so in a rigorous manner that ensures the particles asymtotically approximate samples from the posterior. However, the resample-move algorithm can not avoid sample impoverishment due to the existence of ressampling procedure at each time step. Moreover, the traditional MCMC sampling needs a lot of iterations to converge to the target posterior distribution, which is very slow and not suitable for real-time tracking.

In this paper, a new method, named process noise identification based particle filter, is proposed to tackle the sample impoverishment problem. The proposed method is based on the assumption that the random effect (including maneuvers and random noises) can be modeled by (part of) a white or colored noise process sufficiently well. This fundamental assumption converts the problem of maneuvering target tracking to that of state estimation in presence of non-stationary process noise with unknown statistics. In order to identify the distribution of the process noise, a dynamic system is modeled with the process noise and measurement vectors as its state and measurement vectors respectively. A sampling based algorithm is then proposed in the particle filter framework to estimate the state vector (process noise) based on the current measurements. Using the newly obtained process noise samples, the predicted particles are generated easily from a single general dynamic model. In the proposed method, the predicted particles are generated indirectly from the current measurements through the process noise samples. The predicted particles are more close to the current state of the target, which can reduce the sample impoverishment effectively.

The rest of the paper is organized as follows. In Section 2 the basic theory and procedure of particle filter for state estimation are provided. The process noise identification based particle filter algorithm for tracking highly maneuvering target is introduced in Section 3. The simulation results are shown in Section 4 and the paper is summarized in Section 5.

2 Basic Theory of Particle Filter

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

$$x_k = f(x_{k-1}, v^*_{k-1}), \tag{1}$$

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, $z_{1:k} = \{z_1, z_2, \cdots, z_k\}$. The observation model is described as follows,

$$z_k = h(x_k, n_k), \tag{2}$$

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 $p(x_k|z_{1:k})$.

In this paper, a particle filter is considered to solve the state estimation problem due to its ability to tackle the non-linear and non-Gaussian systems. The posterior distribution $p(x_k|z_{1:k})$ is approximated by a set of particles with associated weights. The detailed introduction about particle filter algorithm can be found in [12]. The procedures associated with the standard particle filter are listed in the following:

Algorithm 1: Particle Filter Algorithm

(1) **Initialization**: Sample initial particles $\{x^i_0, i = 1, ..., H\}$ from the initial posterior distribution $p(x_0)$ and set the weights w^i_0 to $\frac{1}{H}$, $i = 1, ..., H$. H is the number of particles.

(2) **Prediction**: Particles at time step $k-1$, $\{x^i_{k-1}, i = 1, ..., H\}$, are passed through the system model (3) to obtain the predicted particles at time step k, $\{\hat{x}^i_k, i = 1, ..., H\}$:

$$\hat{x}^i_k = f(x^i_{k-1}, v^{*,i}_{k-1}), \tag{3}$$

where $v^{*,i}_{k-1}$ 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 (4).

$$\hat{w}^i_k = p(z_k|\hat{x}^i_k), \qquad w^i_k = \frac{\hat{w}^i_k}{\sum_{i=1}^{H} \hat{w}^i_k} \tag{4}$$

Thus at time step k we can obtain the estimate of the state, $\tilde{\hat{x}}_k = \sum_{i=1}^{H} w^i_k \hat{x}^i_k$.

(4) **Resample** : Resample the discrete distribution $\{w^i_k : i = 1, \cdots, H\}$ H times to generate particles $\{x^j_k :$

$j = 1, \cdots, H\}$, so that for any j, $Pr\{x_k^j = \hat{x}_k^i\} = w_k^i$. Set the weights w_k^i to $\frac{1}{H}$, $i = 1, ..., H$, and move to Stage 2.

3 Maneuvering Target Tracking Based on Process Noise Identification

The general motion model of a maneuvering target can be described by the following state-space model,

$$x_k = f(x_{k-1}, u_{k-1}, v_{k-1}^*), \tag{5}$$

where u is the maneuver acceleration and v^* is the process noise. In the equivalent-noise approach [1], [2], [3], it is assumed that the equation (5) that describes target motions can be simplified to,

$$x_k = f(x_{k-1}, v_{k-1}), \tag{6}$$

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, non-stationary 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 same as in equation (2). The observation equation is defined in (7),

$$z_k = h(x_k, n_k) = h[f(x_{k-1}, v_{k-1}), n_k]. \tag{7}$$

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

$$p(v_{k-1}|z_{1:k}) = \frac{p(z_k|v_{k-1})p(v_{k-1}|z_{1:k-1})}{p(z_k|z_{1:k-1})}, \tag{8}$$

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

$$p(v_{k-1}|z_{1:k}) = \Upsilon \cdot p(z_k|v_{k-1})p(v_{k-1}|z_{1:k-1}). \tag{9}$$

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 (10),

$$p(v_{k-1}|z_{1:k}) = \Upsilon \cdot p(z_k|v_{k-1})p(v_{k-1}). \tag{10}$$

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, $\{\hat{v}_{k-1}^j, j = 1, \cdots, H\}$, from $U(-d, d)$.

$$p(v_{k-1}) = \frac{1}{H}\Sigma_{j=1}^H \delta(v_{k-1} - \hat{v}_{k-1}^j). \tag{11}$$

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,

$$\begin{aligned} p(v_{k-1}|z_{1:k}) &= \frac{\Upsilon}{H}\Sigma_{j=1}^H p(z_k|\hat{v}_{k-1}^j)\delta(v_{k-1} - \hat{v}_{k-1}^j) \\ &= \frac{\Upsilon}{H}\Sigma_{j=1}^H \, \xi_k^j \cdot \delta(v_{k-1} - \hat{v}_{k-1}^j), \end{aligned} \tag{12}$$

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

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

$$p(z_k|\hat{v}_{k-1}^j) = \Sigma_{i=1}^H p(z_k|x_{k-1}^i, \hat{v}_{k-1}^j)p(x_{k-1}^i|\hat{v}_{k-1}^j). \tag{13}$$

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

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

$$p(x_{k-1}^i) = \frac{1}{H}. \tag{14}$$

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

$$\mu_k^{i,j} = f(x_{k-1}^i, \hat{v}_{k-1}^j), \quad i = 1, \cdots H, \; j = 1, \cdots, H \tag{15}$$

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

$$\begin{aligned} p(z_k|x_{k-1}^i, \hat{v}_{k-1}^j) &= \Sigma_{p=1}^H \Sigma_{q=1}^H [p(z_k|\mu_k^{p,q}, x_{k-1}^i, \hat{v}_{k-1}^j) \\ &\quad \times p(\mu_k^{p,q}|x_{k-1}^i, \hat{v}_{k-1}^j)]. \end{aligned} \tag{16}$$

Since x_{k-1}^i and $\hat{v}_{k-1}^j$ are known and $\mu_k^{p,q}$ is obtained from a purely deterministic relationship in (15), we obtain

$$p(\mu_k^{p,q}|x_{k-1}^i, \hat{v}_{k-1}^j) = \left\{ \begin{array}{ll} 1, & for\ p = i\ \ and\ \ q = j \\ 0, & for\ p \neq i\ \ or\ \ q \neq j \end{array} \right., \tag{17}$$

and,

$$p(z_k|x^i_{k-1}, \hat{v}^j_{k-1}) = p(z_k|\mu^{i,j}_k). \quad (18)$$

Combining (14) and (18) with (13) results in,

$$p(z_k|\hat{v}^j_{k-1}) = \sum_{i=1}^{H} p(z_k|\mu^{i,j}_k) \cdot \frac{1}{H}. \quad (19)$$

At each time step, the process noise samples are drawn from a uniform distribution $U(-d, d)$. Each process noise sample $\hat{v}^j_{k-1}$ is evaluated and assigned its corresponding weight ξ^j_k. A resampling procedure is then used to re-distribute 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 $\{\hat{x}^i_k, : i = 1, \ldots, H\}$ are then obtained based on the resampled process noise samples $\{v^i_{k-1} : i = 1, \ldots, H\}$ through the dynamic model (6). 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 2: Process Noise Identification Based Particle Filter

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

(2) Calculate the intermediate particles $\{\mu^{i,j}_k : i = 1, \cdots, H; j = 1, \cdots, H\}$ according to (15).

(3) Calculate the process noise sample weights $\{\xi^j_k : j = 1, \cdots, H\}$ via (19).

(4) Resample the discrete distribution $\{\xi^j_k : j = 1, \cdots, H\}$, H times to generate the resampled process noise samples $\{v^i_{k-1} : i = 1, \cdots, H\}$, so that for any i, $Pr\{v^i_{k-1} = \hat{v}^j_{k-1}\} = \xi^j_k$. Set the weights ξ^j_k to $\frac{1}{H}$, $i = 1, ..., H$.

(5) Calculate the predicted particles at time step k, $\{\hat{x}^i_k, i = 1, ..., H\}$, via (20),

$$\hat{x}^i_k = f(x^i_{k-1}, v^i_{k-1}). \quad (20)$$

(6)∼(7) are same with the stages (3)∼(4) in Algorithm 1.

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 (15) and evaluated via (18). This increases the computation burden and the algorithm runs slowly compared to the conventional particle filter, which are based on H particles (Algorithm 1). It is observed that at each time step, after resampling the particles focus in some smaller area and a large portion of particles are with the same value (Algorithm 1, Step (4)). To simplify the algorithm, it is assumed that the particles (Algorithm 1, Step (4)) are less variable compared with the process noise samples (15). In (15), particles $\{x^i_{k-1}, i = 1, \cdots, H\}$ are replaced by $\tilde{\tilde{x}}_{k-1}$, the estimate of the state at time step $k-1$, which results in a simplified version of (15),

$$\mu^j_k = f(\tilde{\tilde{x}}_{k-1}, \hat{v}^j_{k-1}), \quad (21)$$

and $p(z_k|\hat{v}^j_{k-1})$ is expanded directly on μ^j_k,

$$p(z_k|\hat{v}^j_{k-1}) = \Sigma^H_{\tau=1} p(z_k|\mu^\tau_k) p(\mu^\tau_k|\hat{v}^j_{k-1}). \quad (22)$$

Using the similar derivation process with (17), we can obtain,

$$p(z_k|\hat{v}^j_{k-1}) = p(z_k|\mu^j_k). \quad (23)$$

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.

4 Simulation Results and Analysis

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 [13]. 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

$$X_k = \Phi X_{k-1} + \Gamma v_{k-1} \quad (24)$$

is adopted during the whole tracking process, where in (24),

$$\Phi = \begin{bmatrix} \Phi_b & 0 \\ 0 & \Phi_b \end{bmatrix}, \Phi_b = \begin{bmatrix} 1 & \Delta T & \Delta T^2/2 \\ 0 & 1 & \Delta T \\ 0 & 0 & 1 \end{bmatrix}, \quad (25)$$

and,

$$\Gamma = I_{6\times 6}. \quad (26)$$

Matrix Φ is the transition matrix and ΔT is the sample interval. $X_k = [p_x, \gamma_x, a_x, p_y, \gamma_y, a_y]^T_k$ is the state vector; p_x, γ_x and a_x denote respectively the position, velocity and acceleration 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, $v_{k-1} = [v_{p_x}, v_{\gamma_x}, v_{a_x}, v_{p_y}, v_{\gamma_y}, v_{a_y}]^T_{k-1}$, with unknown statistics is required to be identified. The bound of the process noise (d), which accounts for the uncertain dynamics, is chosen as $\{20\ m, 20\ m/s, 10\ m/s^2, 20\ m, 20\ m/s, 10\ m/s^2\}$. 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 $\{10\ m, 10\ m/s, 5\ m/s^2, 10\ m, 10\ m/s, 5\ m/s^2\}$.

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

$$Z_k = h(X_k) + n_k, \tag{27}$$

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 m (σ_{z_1}) and 0.01 rad (σ_{z_2}). Resolution of the sensor is selected after from [14] (twice of the standard deviations of the measurement noise). The sampling interval is $\Delta T = 1s$.

The process noise identification based particle filter 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 [13]. The initial model probabilities and the mode switching probability matrix are set the same values as in [13]. For the regularised particle filter, Epanechnikov kernel is chosen as the rescaled kernel density, which is same as in [15]. 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. Fig. 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 in Fig. 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 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 process noise identification based particle filter 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.

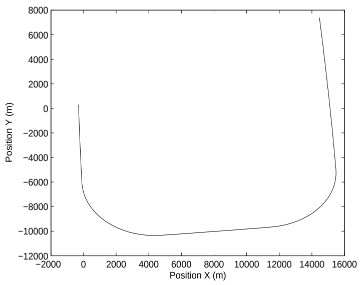

Figure 1: True trajectory of the single maneuvering target

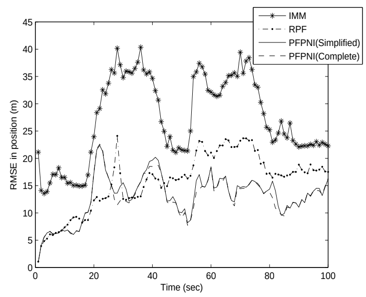

Figure 2: RMSE in position using IMM, RPF, PFPNI (simplified) and PFPNI (complete) algorithms

5 Conclusions

In this paper, a novel method, process noise identification based particle filter is proposed for tracking highly maneuvering target. The proposed method is based on the assumption that the random effect (including maneuvers and random noises) can be modeled by (part of) a white or colored noise process sufficiently

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

well. This fundamental assumption converts the problem of maneuvering target tracking to that of state estimation in presence of non-stationary process noise with unknown statistics. In order to identify the distribution of the process noise, the process noise is modeled as an individual dynamic system. A sampling based algorithm is then proposed in the particle filter framework to estimate the state vector (process noise) based on the current measurements. The predicted particles are then generated from the newly obtained process noise samples and are more close to the current state of the target, which can reduce the sample impoverishment effectively. The proposed algorithm is illustrated via an example involving tracking a highly maneuvering target.

References

[1] A. H. Jazwinski, *Stochastic processes and filtering theory.* New York: Academic Press, 1970.

[2] X. R. Li and Y. Bar-Shalom, "A recursive multiple model approach to noise identification," *IEEE Trans. Aerospace and Electronic Systems*, vol. 30, no. 3, pp. 671–684, July 1994.

[3] ——, "A recursive hybrid system approach to noise identification," *Proceedings of the first IEEE Conference on Control Applications*, pp. 847–852, 1992.

[4] R. Karlsson and F. Gustafsson, "Range estimation using angle-only target tracking with particle filters," *Proceedings of American Control Conference*, vol. **5**, pp. 3743–3748, 2001.

[5] N. Ikoma, N. Ichimura, T. Higuchi, and H. Maeda, "Maneuvering target tracking by using particle filter," *IFSA World Congress and 20th NAFIPS International Conference*, vol. **4**, pp. 2223–2228, 2001.

[6] R. Karlsson and N. Bergman, "Auxiliary particle filters for tracking a maneuvering target," *Proceedings of the 39th IEEE Conference on Decision and Control*, vol. **4**, pp. 3891–3895, 2000.

[7] N. Ikoma, T. Higuchi, and H. Maeda, "Tracking of maneuvering target by using switching structure and heavy-tailed distribution with particle filter method," *Proceedings of the 2002 International Conference on Control Applications*, vol. **2**, pp. 1282–1287, 2002.

[8] W. Malcolm, A. Doucet, and S. Zollo, "Sequential Monte Carlo tracking schemes for maneuvering targets with passive ranging," *Proceedings of the Fifth International Conference on Information Fusion*, vol. **1**, pp. 482–488, 2002.

[9] A. Doucet, J. F. G. de Freitas, and N. J. Gordon, *Sequential Monte Carlo Methods in Practice.* New York: Springer-Verlag, 2001.

[10] M. Morelande and S. Challa, "Manoeuvring target tracking in clutter using particle filters," *IEEE Trans. Aerospace and Electronic Systems*, vol. 41, no. 1, pp. 252–270, Jan. 2005.

[11] S. J. Godsill, J. Vermaak, W. Ng, and J. F. Li, "Models and algorithms for tracking of maneuvering objects using variable rate particle filters," *Proceedings of the IEEE*, vol. 95, no. 5, pp. 925–952, May 2007.

[12] 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.

[13] B. Chen and J. K. Tugnait, "Tracking of multiple maneuvering targets in clutter using IMM/JPDA filtering and fixed-lag smoothing," *Automatica*, vol. 37, pp. 239–249, Feb. 2001.

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

[15] 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. New York: Springer, 2001, pp. 247–271.