Chapter 75
Track-Before-Detect Algorithm Based on Optimization Particle Filter

Huajian Wang

Abstract An improved track-before-detect (TBD) algorithm based on the Cubature particle filter is an efficient approach for weak target detection and track under low signal to noise radio environment. Under the framework of particle filter, the algorithm which combines the particle filter with cubature kalman filter (CKF) algorithm is presented to generate the important density function of particle filter. The simulation results demonstrate that the improved algorithm can provide stable and reliable detection as well as accurate tracking.

Keywords Weak target · Track-before-detect · Cubature particle filter

75.1 Introduction

The Track-Before-Detect (TBD) [1] is good for weak target detecting and tracking in low signal to noise radio (SNR) environment. The method uses the original measurement data of the sensor directly to promote SNR by accumulating target information for a period of time. At the same time the target is being joint detected and estimated, the detected results and target track are announcing.

In recent years, beasuce Particle filter (PF) [2] is not the limit of the nonlinear non-Gaussian problem. The Track-Before-Detect algorithm based on PF has been a lot of attention [3–8]. This is a crucial problem, which affect real-time and tracking accuracy of PF-TBD. In the Paper [5, 6], these described that the

H. Wang (✉)
Department of Information Engineering, Engineering University of China Armed Police Force, No. 710086 WuJing Road, Xi'An Shaanxi, China
e-mail: whj-winner@163.com

W. Lu et al. (eds.), *Proceedings of the 2012 International Conference on Information Technology and Software Engineering*, Lecture Notes in Electrical Engineering 210,
DOI: 10.1007/978-3-642-34528-9_75,

algorithm, which Rao-Blackwellised particle filter-based track-before-detect algorithm for Over-the-horizon radar target. But The algorithm requires a state-space model for condition of linear Gauss model. In the Paper [7], it could avoid the sample impoverishment problem, but convergence problem can not be well guaranteed and operation time is long time. In the Paper [8], it described that the TBD algorithm based on gauss particle filtering avoided resampling and achieved relatively easy. But It required the posterior probability density for gauss distribution. So it's application was restricted.

In order to improve the accuracy and the algorithm running time relationship, this article proposed TBD method based on cubature particle filter. In the tracking phase, the algorithm uses CKF (cubature kalman filter) to construct of the particle filter importance proposal function. The simulation results demonstrate that the algorithm can be obtained under higher filtering accuracy in a small number of particles, meet the requirements of real-time control, and strong robustness.

75.2 Problem Description

75.2.1 State Model

Assuming that the target is in uniform motion in X–Y plane. System state equation is as follows.

$$\mathrm{X}_k = \mathrm{F}\mathrm{X}_{k-1} + \mathrm{w}_{k-1} \tag{75.1}$$

Where $\mathrm{X}_k = [x_k, \dot{x}_k, y_k, \dot{y}_k, I_k]^T$, (x_k, y_k), $(\dot{x}_k, \dot{y}_k)$ and I_k expresses position, speed and strength of the target. w_k is a i.i.d process noise vector. Q_k is the process noise w_k covariance matrix.

State transition matrix is as follows.

$$\mathrm{F} = \begin{bmatrix} \mathrm{F}_x & & \\ & \mathrm{F}_x & \\ & & 1 \end{bmatrix}, \ \mathrm{F}_x = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \tag{75.2}$$

The process noise covariance is as follows.

$$Q_x = \begin{bmatrix} Q_x & & \\ & Q_x & \\ & & q_2 T \end{bmatrix}, \ Q_x = \begin{bmatrix} \frac{q_1}{3}T^3 & \frac{q_1}{2}T^2 \\ \frac{q_1}{2}T^2 & q_1 T \end{bmatrix} \tag{75.3}$$

Where T is a sampling period. q_1 is Process noise power spectral density and q_2 is the target intensity noise power spectral density.

Definiting $E_k \in \{0, 1\}$, 1 is that the target is presence, 0 is that the target is not exist [4]. $\{E_k, k = 0, 1, 2\ldots\}$ is submitted to Markov procedure.

Definiting $P_b = P\{E_k = 1 | E_{k-1} = 0\}$, $P_d = P\{E_k = 0 | E_{k-1} = 1\}$

Transfer probability matrix is as follows.

$$\Pi = \begin{pmatrix} 1 - P_b & P_b \\ P_d & 1 - P_d \end{pmatrix} \tag{75.4}$$

75.2.2 Measurement Model

Assuming that the sensor is scanning for $n \times m$ resolution cell in X–Y plane. Each resolution cell (i, j) corresponds to each the matrix region $\Delta_x \times \Delta_y$, $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, m$. The measured value of each resolution cell (i, j) in every moment is as follows.

$$z_k^{(i,j)} = \begin{cases} h_k^{(i,j)}(x_k) + v_k^{(i,j)} & \text{Target exist} \\ v_k^{(i,j)} & \text{Target not exist} \end{cases} \tag{75.5}$$

For the remote target, $h_k^{(i,j)}$ is the point spread function.

$$h_k^{(i,j)}(\mathrm{x}_k) = \frac{\Delta_x \Delta_y I_k}{2\pi \sum^2} \exp\left\{ -\frac{(i\Delta_x - x_k)^2 + (j\Delta_y - y_k)^2}{2\sum^2} \right\} \tag{75.6}$$

Where $\sum$ is sensor's fuzzy degree, as the known quantity.

The measured value of k is $z_k = \{z_k^{(i,j)}, i = 1, 2, \ldots n, j = 1, 2, \ldots m\}$.

The all measured value in front of all time of k is as follows. $z_{1:k} = \{z_n | n = 1, 2, \ldots k\}$

75.2.3 Likelihood Function

Assuming that the every resolution cell is i.i.d. The likelihood function is as follows.

$$p(\mathrm{z}_k | \mathrm{x}_k, E_k) = \begin{cases} \prod_{i=1}^{n} \prod_{j=1}^{m} p_{s+N}(z_k^{(i,j)} | \mathrm{x}_k) & E_k = 1 \\ \prod_{i=1}^{n} \prod_{j=1}^{m} p_N(z_k^{(i,j)}) & E_k = 0 \end{cases} \tag{75.7}$$

Where $p_N(z_k^{(i,j)})$ is the likelihood function of the resolution cell (i, j) noise. $p_{s+N}(z_k^{(i,j)} | \mathrm{x}_k)$ is the likelihood function of the resolution cell (i, j) signal and noise. Because of each resolution element of the noises are independent, and the effects of target to measure the signal intensity is limit. Likelihood ratio is as follows.

$$L(Z_k|X_k,E_k)=\frac{p(Z_k|X_k,E_k)}{p(Z_k|X_k,E_k=0)}\approx\begin{cases}\frac{1}{M}\sum\limits_{m=1}^{M}\prod\limits_{i\in C_i(x_k^m)}\prod\limits_{j\in C_j(x_k^m)}l(z_k^{(i,j)}|x_k^m) & E_k=1\\ 1 & E_k=0\end{cases}\tag{75.8}$$

Where

$$l(z_k^{(i,j)}|x_k^m)=\frac{p_{s+N}(z_k^{(i,j)}|\mathrm{x}_k)}{p_N(z_k^{(i,j)})}=\exp\left\{-\frac{h_k^{(i,j)}(x_k^m)(h_k^{(i,j)}(x_k^m)-2z_k^{(i,j)})}{2\sigma^2}\right\}\tag{75.9}$$

75.3 Track-Before-Detect Algorithm Based on Cubature Particle Filter

The problem of TBD is a mixed problem. What is the solution to the TBD problem is that the target state x_k and the joint posterior probability density $p(x_k,E_k=1|Z_k)$ are estimated in k time. There is that the frame of Salmond is used.

The new algorithm that incorporates the latest observation into a prior updating phase develops the importance density function by CKF that is more close to posterior density in Nonlinear and non-Gauss [9, 10]. So a iteration steps used CKF to update particle is as follows

Step1 Calculation Cubature point

$$P_{k-1/k-1}=S_{k-1/k-1}S_{k-1/k-1}^{T}\tag{75.10}$$

$$X_{i,k-1/k-1}=S_{k-1/k-1}\xi_i+\hat{x}_{k-1/k-1}\tag{75.11}$$

Step2 One step prediction of state

$$X_{i,k/k-1}^{*}=f(X_{i,k-1/k-1})\tag{75.12}$$

$$\hat{x}_{k/k-1}=\frac{1}{2n}\sum_{i=1}^{2n}X_{i,k/k-1}^{*}\tag{75.13}$$

$$P_{k/k-1} = \frac{1}{2n}\sum_{i=1}^{2n} X^{*}_{i,k/k-1}X^{*T}_{i,k/k-1} - \hat{x}_{k/k-1}\hat{x}^{T}_{k/k-1} + Q_{k-1} \tag{75.14}$$

Step3 Measurement update

$$P_{k/k-1} = S_{k/k-1}S^{T}_{k/k-1} \tag{75.15}$$

$$X_{i,k/k-1} = S_{k/k-1}\xi_i + \hat{x}_{k/k-1} \tag{75.16}$$

$$Z_{i,k/k-1} = h(X_{i,k/k-1}) \tag{75.17}$$

$$\hat{z}_{k/k-1} = \frac{1}{2n}\sum_{i=1}^{2n} Z_{i,k/k-1} \tag{75.18}$$

Step4 Estimated value

$$\hat{x}_{k/k} = \hat{x}_{k/k-1} + K_k(z_k - \hat{z}_{k/k-1}) \tag{75.19}$$

$$P_{kk} = P_{k/k-1} - K_k P_{zz,k/k-1} K_k^T \tag{75.20}$$

$$P_{zz,k/k-1} = \frac{1}{2n}\sum_{i=1}^{2n} Z_{i,k/k-1}Z^{T}_{i,k/k-1} - \hat{z}_{k/k-1}\hat{z}^{T}_{k/k-1} + R_k \tag{75.21}$$

$$P_{xz,k/k-1} = \frac{1}{2n}\sum_{i=1}^{2n} X_{i,k/k-1}Z^{T}_{i,k/k-1} - \hat{x}_{k/k-1}\hat{z}^{T}_{k/k-1} \tag{75.22}$$

$$K_k = P_{xz,k/k-1}P^{-1}_{zz,k/k-1} \tag{75.23}$$

Assuming that the posterior probability density of the joint target state (x_k, E_k) is $p(x_k, E_k|z_{1:k})$. So, the steps of the track-before-detect algorithm based on Cubature particle filter are as follows:

Step 1 Initialization

P_b is as appearance probability of targets. While $E_k = 1$, the prior distribution $p(x_0)$ samples from $\{x_0^{(i)}\}_{i=1}^{N}$.

$$\hat{x}_0^{(i)} = E[x_0^{(i)}] \tag{75.24}$$

$$\hat{P}_0^{(i)} = E[(x_0^{(i)} - \hat{x}_0^{(i)})(x_0^{(i)} - \hat{x}_0^{(i)})^T] \tag{75.25}$$

Step 2 Calculating the predicted particle targets exist variables

According to the transfer probability matrix Π and $\{E_{k-1}^n\}_{n=1}^N$, to calculate $\{E_k^n\}_{n=1}^N$

Step 3 Forecasting $\hat{x}_{k/k}$ and P_{kk} by CKF

For while "Newborn particle" of $E_{k-1} = 0, E_k = 1$, the prior distribution $p(x_0)$ samples from $\{x_0^{(i)}\}_{i=1}^N$.

For while "Survival particle" of $E_{k-1} = 1$, $E_k = 1$, $\hat{x}_k^{(i)}$ and $\hat{P}_k^{(i)}$ are estimated by CKF.

Step 4 The normalized weights

$$\{\omega_k^n = \tilde{\omega}_k^n / \sum_{n=1}^N \tilde{\omega}_k^n\}_{n=1}^N \tag{75.26}$$

Step 5 Resampling

If $\hat{N}_{eff} < \hat{N}_{tthreshold}$, weights is $\{\omega_k^n = \frac{1}{N}\}_{n=1}^N$.

Step 6 Statistics the number of $E_k = 1$, so the appearance probability of targets is as follows

$$\hat{p}_{E_k=1}(E_k = 1|z_{1:k}) = \frac{N_{E_k=1}}{N} \tag{75.27}$$

Step 7 setting threshold p_T, if $\hat{p}_{E_k=1} > p_T$, this describes the target appears The target state estimation value is as follows

$$\hat{x}_k = \sum_{n=1}^N x_k^n \cdot \omega_k^n \tag{28}$$

75.4 Simulation Result

In order to verify the performance of the algorithm, there has 30 frame radar simulation data in 20 × 20 area. Every resolution cell is $\Delta_x \times \Delta_y = 1 \times 1$, $p = 2$. Radar located at the origin of coordinates. Scanning time of a frame is 1 s. Assuming that the target appears from 5 to 25 s for uniform motion. The initial location is (4.2,7.2), The initial velocity is (0.45,0.25). The target signal strength $I = 20$, $\Sigma = 0.7. P_b = 0.1. P_d = 0.1$.

In order to evaluate the effect of the target tracking, this algorithm compares to UPF-TBD which is introducted in paper [14]. To definine the root mean square error is as follows:

$$RMSE = \frac{1}{N}\sum_{i=1}^{N}\sqrt{(x_k - \hat{x}_k^i)^2 + (y_k - \hat{y}_k^i)^2} \tag{75.30}$$

From Fig. 75.1, while the same number of particles (N = 4,000) is used, the performance of CPF-TBD is better than UPF-TBD in the latter tracking part in low SNR conditions(SNR = 6 dB). As shown in Table 75.1, position RMSE improves about 20 % and velocity RMSE improves about 23 %. At the same time, execution time is relatively less.

From Fig. 75.2 while the different number of particles is used, the performance of CPF-TBD is nearly as same as UPF-TBD in low SNR conditions(SNR = 6 dB). while particles number of CPF-TBD ($N_{CPF\text{-}TBD}$ = 3,000) is only one third of UPF-TBD $N_{UPF\text{-}TBD}$ = 10,000. As shown in Table 75.1, position RMSE only worsens about 4.3 % and velocity RMSE about 0.86 % while the execution time of CPF-TBD is relatively 38.2 % of UPF-TBD.

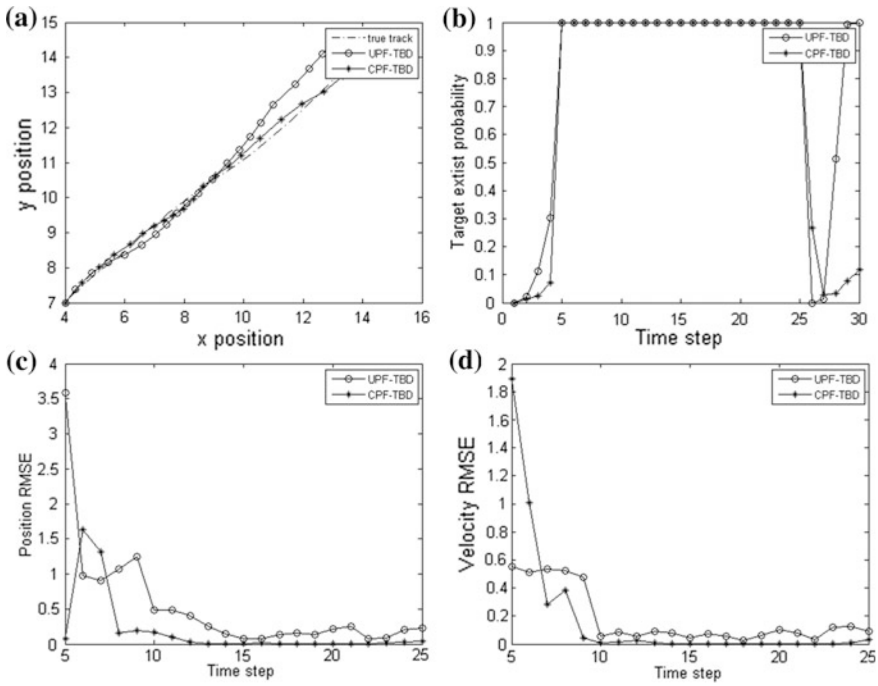

Fig. 75.1 Curve while the same number of particles (SNR = 6 dB). **a** The curve of the real and estimated values. **b** Target appearance probability. **c** Position RMSE. **d** Velocity RMSE

Table 75.1 The performance comparison in PF-TBD and CPF-TBD

Algorithm	Particles number	Position RMSE	Velocity RMSE	Execution time
UPF-TBD	4,000	4.618726	0.242404	2.2926
	10,000	3.892483	0.195826	5.1151
CPF-TBD	3,000	3.925910	0.204278	1.9563
	4,000	3.678081	0.184375	2.0038

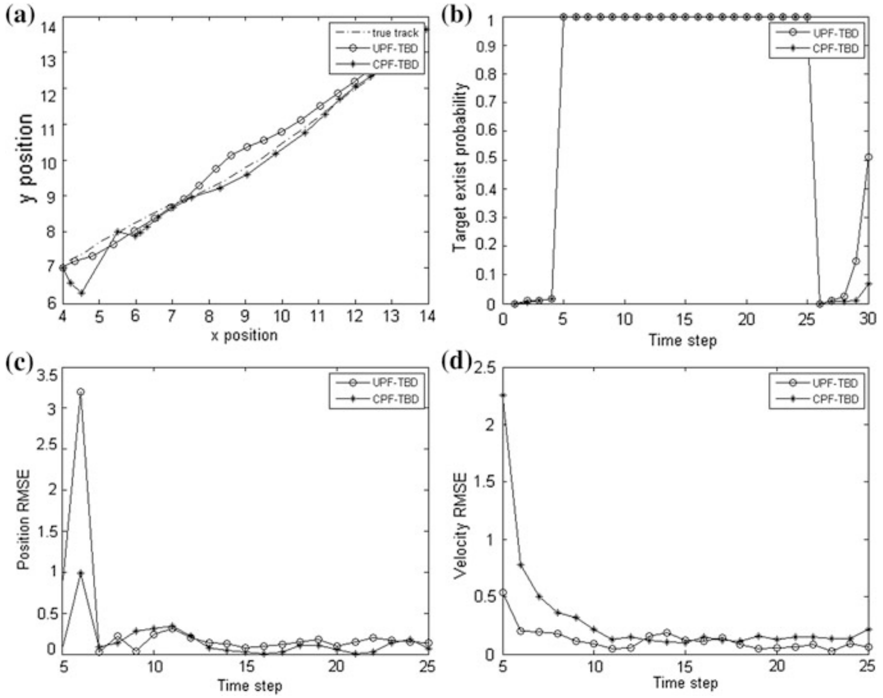

Fig. 75.2 Curve while the different number of particles (SNR = 6 dB). **a** The curve of the real and estimated values. **b** Target appearance probability. **c** Position RMSE. **d** Velocity RMSE

75.5 Conclusion

The track-before-detect algorithm based on Cubature particle filter is proposed. The new algorithm that incorporates the latest observation into a prior updating phase develops the importance density function by CKF that is more close to posterior density in nonlinear and non-Gauss. The simulation results demonstrate that the improved algorithm can provide stable and reliable detection as well as accurate tracking, and balances the relationship between the precision of the filter and the algorithm running time.

Acknowledgments This research work was supported by the foundation fund of China Armed Police Force No. WJY201114.

References

1. Kramer JDR, Reid WS (1990) Track-before- detect processing for an airbrone type radar. IEEE international conference on radar
2. Gordon NJ, Salmond DJ, Smith AFM (1993) Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEEE Proc-F 140(2):107–113
3. Salmond DJ, Birch H (2001) A particle filter for track-before-detect. In: Proceedings of the American control conference, pp 3755–3760
4. Rutten MG, Gordon NJ, Maskell S (2005) Recursive track-before-detect with target amplitude fluctuations. IEE Proc Radar, Sonar and Navig 152(5):345–352
5. Su HT, Wu TP, Liu HW et al (2008) Rao- Blackwellised particle filter based track- before-detect algorithm. IET Sig Process 2(2):169–176
6. Cuiyun LI, Hongbing JI (2009) Track-before- detection for multi weak targets based on Rao-Blackwellized particle filter. Opt Precis Eng 17(9):2342–2349
7. Ling F (2011) Fission reproduction particle filter-based track-before-detect algorithm. J Comput Appl 31(9):2581–2583
8. Ling F (2011) Gaussian particle filter-based track-before-detect method. Comput Eng Appl 47(23):121–123
9. Arasaratnam I, Haykin S (2009) Cubature kalman filter. IEEE Trans Autom Control 54(6):1254–1269
10. Zhang H (2006) Research for track-before-detect algorithm based on Bayesian filtering. Master degree thesis of Northwestern Polytechnical University, p 3
11. Wang X, Pan Q, Huang H, Gao A (2012) Overview of deterministic sampling filtering algorithms for nonlinear system. Control Decis 27(6):801–812
12. Sun F, Tang L (2011) Cubature particle filter. Syst Eng Electron 33(11):2554–2557