Journal of Zhejiang University SCIENCE A
ISSN 1009-3095 (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

A filter algorithm for multi-measurement nonlinear system with parameter perturbation*

GUO Yun-fei†1, WEI Wei[1], XUE An-ke[2], MAO Dong-cai[2]
([1]*School of Electrical Engineering, Zhejiang University, Hangzhou 310027, China*)
([2]*Institute of Intelligence Information and Control Technology, Hangzhou Dianzi University, Hangzhou 310018, China*)
†E-mail: zizhe_yjys@yahoo.com.cn
Received Dec. 20, 2005; revision accepted Apr. 10, 2006

Abstract: An improved interacting multiple models particle filter (IMM-PF) algorithm is proposed for multi-measurement nonlinear system with parameter perturbation. It divides the perturbation region into sub-regions and assigns each of them a particle filter. Hence the perturbation problem is converted into a multi-model filters problem. It combines the multiple measurements into a fusion value according to their likelihood function. In the simulation study, we compared it with the IMM-KF and the *H*-infinite filter; the results testify to its advantage over the other two methods.

Key words: IMM-PF, Parameter perturbation, Multi-measurement
doi:10.1631/jzus.2006.A1717 **Document code:** A **CLC number:** TP13

INTRODUCTION

For the filtering problem of multi-measurement nonlinear system with parameter perturbation, we need to consider the following aspects. First, although the parameter uncertainty does not change the system configuration, it makes the filter designed based on a fixed parameter unsuitable for an alterable parameter. Second, there are false alarms in the raw measurements which are received by multiple sensors. Third, since the optimal nonlinear Bayesian estimation is impossible to implement, an approximate method is needed to get a suboptimal estimation.

An H_∞ filter was presented in (Tsaknakis and Athans, 1994) to track a maneuvering target, which can be modelled as a system with parameter perturbation. The H_∞ filter ensures that the system satisfies some performance criteria if the perturbation is the most serious, yet cannot ensure that the system has better performance than a Kalman filter with an appropriately designed process noise covariance matrix when the perturbation is small. We propose a new method to solve the parameter perturbation problem. It divides the continuous variety region of the parameter into many discrete sub-regions and allocates each sub-region a filter. If the number of the sub-regions is big enough that the parameter can be approximately regarded as constant segment separated by the sub-region, the parameter perturbation problem can then be converted into a multi-model filter problem, the interacting multiple models idea (Bar-Shalom *et al*., 2001) can be applied. The system's inherent nonlinearity can be solved by the extended Kalman filter (EKF) or the particle filter, the latter needs more computer resource. The previous interacting multiple models particle filter (IMM-PF) method (Boers and Driessen, 2003; Blom and Bloem, 2004) is only suitable for single noise-infected measurement. In this paper, an improved IMM-PF algorithm which can deal with multiple noise-corrupted measurements is presented; it combines the multiple measurements into a fusion value according

* Project supported by the National Natural Science Foundation of China (No. 60434020), and the Natural Science Foundation of Zhejiang Province (No. R105341), China

to their likelihood function before the filtering. The results in the simulation study showed its effectiveness and efficiency.

AN IMPROVED IMM-PF ALGORITHM

Problem description

In this paper, we consider a kind of special nonlinear system, with linear state function, nonlinear measurement function and containing an uncertainty parameter.

$$\boldsymbol{x}_{k+1}=\boldsymbol{F}(s)\boldsymbol{x}_k+\boldsymbol{G}(s)\boldsymbol{w}_k,\ \boldsymbol{y}_k=h(\boldsymbol{x}_k)+\boldsymbol{v}_k, \tag{1}$$

where $\boldsymbol{x}_k$ is the system's state, and $\boldsymbol{y}_k$ is the system's measurement, the probability density function of the process noise $\boldsymbol{w}$ and the measurement noise $\boldsymbol{v}$ are denoted respectively as $p_{\boldsymbol{w}}(\cdot)$, $p_{\boldsymbol{v}}(\cdot)$, the parameter s in the transition matrix $\boldsymbol{F}(s)$ and the input matrix $\boldsymbol{G}(s)$ is uncertainty, whose perturbation bound is

$$s_{\min}\le s\le s_{\max}. \tag{2}$$

When the system is observed by multiple sensors, the raw measurement set contains not only the valid data (from the system) but also the false data (from the environment), which is called clutter or false alarm. We denote the original measurement set at time step k as $\boldsymbol{Y}_k=\{\boldsymbol{y}_k^j\}_{j=1}^m$ (m is the number of the valid sensors or the number of raw measurements), the valid measurement set as $\hat{\boldsymbol{Y}}_k$ ($\hat{\boldsymbol{Y}}_k \subseteq \boldsymbol{Y}_k$), and the total measurement set up to and including time step k as $\boldsymbol{Y}^k=\{\boldsymbol{Y}_i\}_{i=1}^k$. The objective of filtering the system is to get the state estimation $\hat{\boldsymbol{x}}_k$ and its error covariance matrix $\boldsymbol{P}_k$ based on $\boldsymbol{Y}^k$.

Parameter perturbation conversion

If the parameter perturbation region is divided into N sub-regions and N is big enough so that the parameter can be approximately regarded as constant in each sub-region (it is similar to the numerical integration method), then the parameter perturbation problem can be converted into a multiple models filter problem. The only difference among the models is the value of the parameter s in each model.

If no more a priori information on the parameter s is available, we can divide the region $[s_{\min}, s_{\max}]$ into N equal sub-regions, and take the middle value of the qth sub-region as the parameter value in the qth model.

$$s_q = s_{\min} + \frac{2q-1}{2N}(s_{\max}-s_{\min}). \tag{3}$$

The state function of the qth model is

$$\boldsymbol{x}_{q,k+1}=\boldsymbol{F}(s_q)\boldsymbol{x}_{q,k}+\boldsymbol{G}(s_q)\boldsymbol{w}_k,\ \boldsymbol{y}_{q,k}=h(\boldsymbol{x}_{q,k})+\boldsymbol{v}_k. \tag{4}$$

Considering that $\boldsymbol{w}$, $\boldsymbol{v}$ are unnecessary Gaussian distribution noises and the measurement function is nonlinear, we allocate each model a particle filter.

Improved IMM-PF algorithm flow

The IMM-PF algorithm is a combination of the interacting multiple models and the particle filters. The conventional approach is using hybrid particles and is based on the Gaussian sum theorem (Boers amd Driessen, 2003; McGinnity and Irwin, 2000; Kotecha and Djuric, 2003). A novel method using the exact Bayesian equation for the conditional mode probabilities is proposed (Blom and Bloem, 2004). They are all only suitable for single noise-infected measurement but not multiple measurements. In this paper, an improved IMM-PF algorithm is proposed to deal with the latter problem. It will combine the raw multiple data in one scan period into a fusion value according to their likelihood function before the filtering procedure.

For convenience, some main variables, vectors and matrices in the algorithm are given as follows:

N, n, m: the number of the models, the particles in each model, the measurements respectively;

$\boldsymbol{\Pi}=\{\pi_{i,q}\}$: the Markovian switching matrix. $\pi_{i,q}$ is the Markov transition probability from the ith model to the qth model;

$\mu_{q,k}$: the qth model probability at time k. $\{\mu_{q,0}\}_{q=1}^N$ is the designed initial value;

$\hat{p}_{q,k}(\boldsymbol{x}_{q,k}\,|\,\boldsymbol{Y}^k)$: the estimated probability density function of the qth model's state at time k. $\{\hat{p}_{q,0}(\boldsymbol{x}_{q,0}\,|\,\boldsymbol{Y}^0)\}_{q=1}^N$ is a priori known initial density;

$p_{\boldsymbol{w}}$, $p_{\boldsymbol{v}}$: the probability density functions of the process noise and measurement noise.

The improved IMM-PF algorithm first combines the raw multiple data in each scan period into a fusion value according to their likelihood function; second, it mixes the models to get each model's mixing initial probability and probability density function; third, it draws particles, computes their weights, and outputs each model's state estimation; last, it gets a fusion state estimation. The algorithm flow is as follows.

Step 1: Computing the valid measurement fusion value of the qth model.

Assuming the state estimation output of the qth model at time $k-1$ is $\hat{\boldsymbol{x}}_{q,k-1}$, and the residual covariance matrix is $\boldsymbol{S}_{q,k-1}$, we can get the one-step prediction of the measurement:

$$\hat{\boldsymbol{y}}_{q,k}=h[\boldsymbol{F}(s_q)\hat{\boldsymbol{x}}_{q,k-1}]. \tag{5}$$

If the raw measurement set at time k is $\boldsymbol{Y}_k=\{\boldsymbol{y}_k^j\}_{j=1}^m$, then the qth model's valid measurement should satisfy the following inequality:

$$(\boldsymbol{y}_k^j-\hat{\boldsymbol{y}}_{q,k})\boldsymbol{S}_{q,k-1}(\boldsymbol{y}_k^j-\hat{\boldsymbol{y}}_{q,k})^{\mathrm{T}}\le\gamma, \tag{6}$$

where γ is the valid gate constant related to the measurement dimension (Fortmann *et al*., 1983; Zhou *et al*., 1991). The valid measurement set is denoted as $\hat{\boldsymbol{Y}}_{q,k}=\{\hat{\boldsymbol{y}}_{q,k}^j\}_{j=1}^{\bar{m}(q)}$, $\bar{m}(q)\le m$. If the likelihood function of measurement is approximated to the Gaussian form, the fusion value of the $\bar{m}(q)$ valid measurements of the qth model is

$$\bar{\boldsymbol{y}}_{q,k}=\sum_{j=1}^{\bar{m}(q)}\beta_{q,k}^j\cdot\hat{\boldsymbol{y}}_{q,k}^j, \tag{7}$$

$$\beta_{q,k}^j=\frac{\exp\left[-\frac{1}{2}(\boldsymbol{y}_k^j-\hat{\boldsymbol{y}}_{q,k})^{\mathrm{T}}\boldsymbol{S}_{q,k-1}(\boldsymbol{y}_k^j-\hat{\boldsymbol{y}}_{q,k})\right]}{\sum_{j=1}^{\bar{m}(q)}\exp\left[-\frac{1}{2}(\boldsymbol{y}_k^j-\hat{\boldsymbol{y}}_{q,k})^{\mathrm{T}}\boldsymbol{S}_{q,k-1}(\boldsymbol{y}_k^j-\hat{\boldsymbol{y}}_{q,k})\right]}. \tag{8}$$

If $\bar{m}(q)=0$, the fusion value will be taken as the one-step prediction value, i.e., $\bar{\boldsymbol{y}}_{q,k}=\hat{\boldsymbol{y}}_{q,k}$.

Step 2: Calculating the mixing probability from the ith model to the qth model at time k, and the probability density estimation function of the qth model.

$$\mu_{i|q,k}=\frac{\pi_{iq}\mu_{i,k}}{\sum_{i=1}^{N}\pi_{iq}\mu_{i,k}},\ q=1,\ldots,N, \tag{9}$$

$$\hat{p}_q^0(\boldsymbol{x}_{0q,k}\mid\boldsymbol{Y}^k)=\sum_{i=1}^{N}\hat{p}_i(\boldsymbol{x}_{i,k-1}\mid\boldsymbol{Y}^{k-1})\cdot\mu_{i|q,k}. \tag{10}$$

Step 3: Drawing particles, computing their weights and outputting each model's state estimation.

Sampling particle set $\{\boldsymbol{x}_{0q,k}^{(j)}\}_{j=1}^n$ from the qth model's probability density estimation function $\hat{p}_q^0(\boldsymbol{x}_{0q,k}\mid\boldsymbol{Y}^k)$, we can get the one-step prediction particle set $\{\boldsymbol{x}_{q,k}^{(j)}\}_{j=1}^n$.

$$\boldsymbol{x}_{q,k}^{(j)}=\boldsymbol{F}(s_q)\boldsymbol{x}_{0q,k}^{(j)}+\boldsymbol{G}(s_q)\boldsymbol{w}_{q,k}^{(j)}. \tag{11}$$

The weight of each particle is

$$\omega_{q,k}^{(j)}=p(\bar{\boldsymbol{y}}_{q,k}\mid\boldsymbol{x}_{q,k}^{(j)})=p_v[\bar{\boldsymbol{y}}_{q,k}-h(\boldsymbol{x}_{q,k}^{(j)})]. \tag{12}$$

The normalization of the weight is

$$\tilde{\omega}_{q,k}^{(j)}=\omega_{q,k}^{(j)}\Big/\sum_{j=1}^{n}\omega_{q,k}^{(j)}. \tag{13}$$

The one-step prediction measurement of each particle is $h(\boldsymbol{x}_{q,k}^{(j)})$, and its residual is

$$\tilde{\boldsymbol{y}}_{q,k}^{(j)}=\bar{\boldsymbol{y}}_{q,k}-h(\boldsymbol{x}_{q,k}^{(j)}). \tag{14}$$

The mean of the prediction measurement is

$$\boldsymbol{y}'_{q,k}=\frac{1}{n}\sum_{j=1}^{n}h(\boldsymbol{x}_{q,k}^{(j)}). \tag{15}$$

The residual's covariance matrix of the qth model is

$$\boldsymbol{S}_{q,k}=\sum_{j=1}^{n}\tilde{\omega}_{q,k}^{(j)}\left[h(\boldsymbol{x}_{q,k}^{(j)})-\boldsymbol{y}'_{q,k}\right]\left[h(\boldsymbol{x}_{q,k}^{(j)})-\boldsymbol{y}'_{q,k}\right]^{\mathrm{T}}. \tag{16}$$

The likelihood function of the qth model is

$$\varLambda_{q,k}=\sum_{j=1}^{n}\tilde{\omega}_{q,k}^{(j)}N(\tilde{\boldsymbol{y}}_{q,k}^{(j)};\boldsymbol{0},\boldsymbol{S}_{q,k}). \tag{17}$$

The updating of the model probability at time k is

$$\bar{c}_q=\sum_{i=1}^{N}\pi_{iq}\mu_{i,k-1},\ \ \mu_{q,k}=\frac{\varLambda_{q,k}\cdot\bar{c}_q}{\sum_{q=1}^{N}\varLambda_{q,k}\cdot\bar{c}_q}. \tag{18}$$

The state estimation and the state error covariance matrix estimation of the qth model according to the minimum mean square error (MMSE) criterion are

$$\left.\begin{aligned}\hat{\boldsymbol{x}}_{q,k} &\approx \sum_{j=1}^{n} \tilde{\omega}_{q,k}^{(j)} \cdot \boldsymbol{x}_{q,k}^{(j)},\\ \boldsymbol{P}_{q,k} &\approx \sum_{j=1}^{n} \tilde{\omega}_{q,k}^{(j)} \cdot [\boldsymbol{x}_{q,k}^{(j)} - \hat{\boldsymbol{x}}_{q,k}][\boldsymbol{x}_{q,k}^{(j)} - \hat{\boldsymbol{x}}_{q,k}]^{\mathrm{T}}.\end{aligned}\right\} \quad (19)$$

The qth model's state probability density function can be estimated as the sum of finite Gaussian density function

$$\hat{p}_q(\boldsymbol{x}_{q,k} \mid \boldsymbol{Y}^k) \approx \sum_{j=1}^{n} \tilde{\omega}_{q,k}^{(j)} N(\boldsymbol{x}_{q,k}^{(j)}; \hat{\boldsymbol{x}}_{q,k}, \boldsymbol{P}_{q,k}). \quad (20)$$

Step 4: Outputting the fusion state estimation and its covariance matrix.

The system's state probability density estimated function is

$$\hat{p}(\boldsymbol{x}_k \mid \boldsymbol{Y}^k) = \sum_{q=1}^{N} \mu_{q,k} \hat{p}_q(\boldsymbol{x}_{q,k} \mid \boldsymbol{Y}^k). \quad (21)$$

The system's estimated state is

$$\hat{\boldsymbol{x}}_k \approx \sum_{q=1}^{N} \mu_{q,k} \hat{\boldsymbol{x}}_{q,k}. \quad (22)$$

The system's estimated state covariance matrix is

$$\boldsymbol{P}_k = \sum_{q=1}^{N} \mu_{q,k} [\boldsymbol{P}_{q,k} + (\hat{\boldsymbol{x}}_{q,k} - \hat{\boldsymbol{x}}_k)(\hat{\boldsymbol{x}}_{q,k} - \hat{\boldsymbol{x}}_k)^{\mathrm{T}}]. \quad (23)$$

The algorithm's flow chart is illustrated in Fig.1.

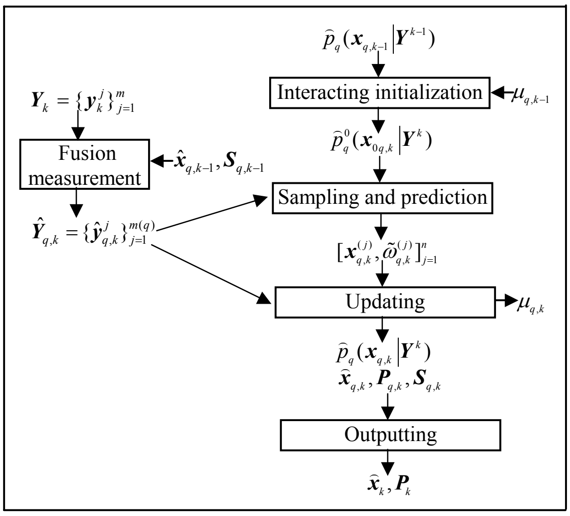

Fig.1 IMM-PF algorithm flow chart

SIMULATION

In the simulation study, we take a target tracking system with signal propagation delay as an example. For simplicity, we consider the problem in a two dimensional plane. The target is observed by six acoustic sensors with the target speed v_t (v_t=90 m/s) being less than the sound speed in air v_c (v_c=340 m/s). v_t, v_c are both time invariable scalar quantities. For the signal propagation delay effect, when the signal emitted by the target at position A arrives at the sensor, the target has moved to a different position A'. Even if the sensor scan period T is a constant (T=1 s), the corresponding signal emitted at interval ΔT is time dependent. If the next received signal is emitted at position B, see Fig.2, the following time delay equation holds:

$$\frac{l_{AO}}{v_c} + T = \Delta T + \frac{l_{BO}}{v_c}, \quad (24)$$

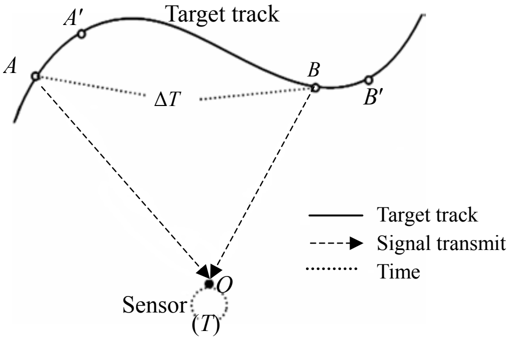

Fig.2 Signal propagation delay illustration

where l_{AO}, l_{BO} denote the distances between the signal emitted positions A, B and the sensor position O respectively. Because they are both time dependent, the signal emitted interval ΔT is also time dependent and its perturbation bound [$T_{\min}$, $T_{\max}$] can be approximately calculated as follows (under the assumption that the scan period is so small that the target trajectory $\widetilde{AB}$ in ΔT can be approximated as a line $\overrightarrow{AB}$

and $\Delta T \approx |\overline{AB}| / v_t$ is satisfied):

$$T_{\max} \approx \frac{v_c}{v_c - v_t} \cdot T, \quad T_{\min} \approx \frac{v_c}{v_c + v_t} \cdot T. \tag{25}$$

Taking the target position and velocity as the system state $\boldsymbol{X}_k=[x_k, v_{x,k}, y_k, v_{y,k}]^{\mathrm{T}}$, the bearing β and range ρ as measurements; we can get the following constant velocity (CV) model:

$$\left.\begin{aligned} &\boldsymbol{X}_{k+1} = \boldsymbol{F}(\Delta T)\boldsymbol{X}_k + \boldsymbol{G}(\Delta T)\boldsymbol{w}_k, \\ &\begin{bmatrix} \rho_k \\ \beta_k \end{bmatrix} = \begin{bmatrix} \sqrt{\boldsymbol{x}_k^2 + \boldsymbol{y}_k^2} \\ \arctan(\boldsymbol{x}_k / \boldsymbol{y}_k) \end{bmatrix} + \boldsymbol{v}_k, \\ &\boldsymbol{F}(\Delta T) = \begin{bmatrix} 1 & \Delta T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\ &\boldsymbol{G}(\Delta T) = \begin{bmatrix} \Delta T & 0 \\ 0.5(\Delta T)^2 & 0 \\ 0 & \Delta T \\ 0 & 0.5(\Delta T)^2 \end{bmatrix}, \end{aligned}\right\} \tag{26}$$

where $\boldsymbol{w}$, $\boldsymbol{v}$ are independent white noises with "student distribution" probability density functions $p_{\boldsymbol{w}}$, $p_{\boldsymbol{v}}$ (freedom is one). ΔT is the perturbation parameter in the model. We divide its region $[T_{\min}, T_{\max}]$ into N sub-regions (N=3) and allocate each sub-region an initial model weight ($\mu_{i,0}=1/N$, i=1, 2, 3). The Markovian switching probability matrix is given as $\boldsymbol{\Pi} = \begin{bmatrix} 0.90 & 0.05 & 0.05 \\ 0.05 & 0.90 & 0.05 \\ 0.05 & 0.05 & 0.90 \end{bmatrix}$; the number of particles in each model is n=50.

Notice that, we choose N=3 here just for simplicity; a bigger N helps to improve the proposed algorithm performance in estimation precision but requires more computer resource. We apply three algorithms (IMM-PF, H_∞ filter, IMM-KF) respectively to track the target. The track scenarios by the three methods are illustrated in Figs.3a~3c. The corresponding estimation root mean square errors (RMSE) in position (x axis) are shown in Figs.4a~4c. Table 1 shows the elapsed total time in 100 Monte Carlo simulations, the correct track probability and the maximum position error RMSE by the three algorithms.

The results showed that the estimated performance of the proposed algorithm is much better than the H_∞ filter and the IMM-KF—the maximum RMSE in position of the IMM-PF is only 62.5% of that of the

Table 1 Performance comparison

Parameter	IMM-KF	*H*-infinite	IMM-PF
Elapsed total time (s)	2.593	1.012	127.219
Correct probability (%)	89	86	87
Maximum RMS (m)	250	160	100

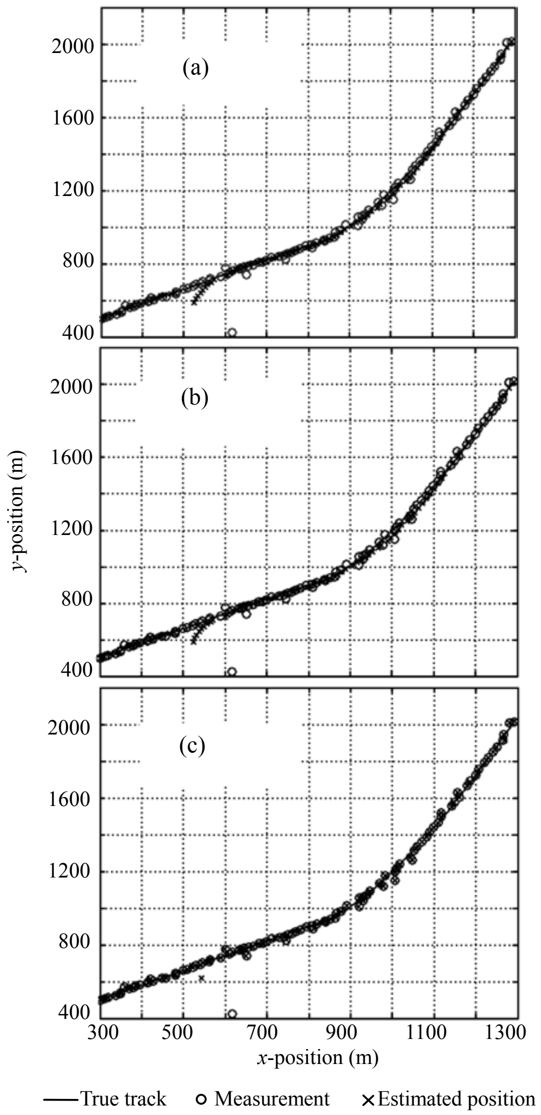

Fig.3 Target track. (a) IMM-KF; (b) *H*-inf.Filter; (c) IMM-PF

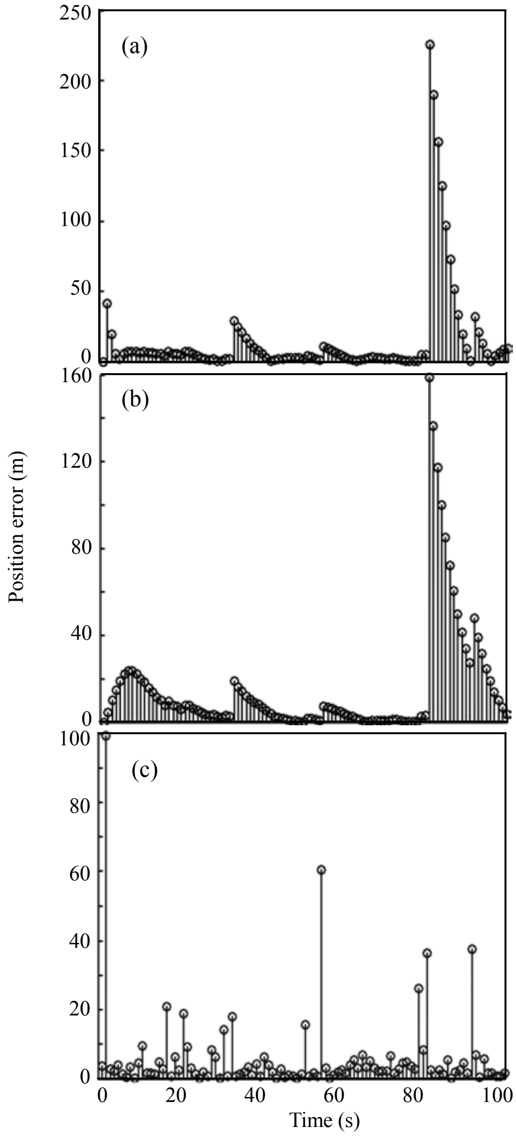

Fig.4 Position error RMS. (a) IMM-KF; (b) *H*-inf.Filter; (c) IMM-PF

H_∞ filter and 40% of that of the IMM-KF. The correct probability of the three methods is similar. The elapsed time in one recursion of the IMM-PF is acceptable compared with the scan period. The particle filters represent the probability densities with weighted particles but not approximate it with the first two order moment (mean and covariance). With the number of the particles increasing, the particle filters can approach the optimal Bayesian estimator (Ristic *et al*., 2004). In fact, the elapsed time of the IMM-PF is proportional to the number of the models and the number of the particles in each model. And, the elapsed time of the IMM-KF is approximately 3 times that of the H_∞ filter and about 1/50 that of the IMM-PF (Ababsa *et al*., 2004). If we have a higher real-time processing requirement for the tracking system, we can decrease the number of the particles in each model.

CONCLUSION

In this paper, we propose an improved interacting multiple models particle filter algorithm for the multi-measurement nonlinear system with parameter perturbation. The parameter perturbation problem is converted into a multiple models filter problem by a region division method. The raw multiple measurements are combined to a fusion value before the filtering procedure. In the simulation study, we compared the proposed approach with the H_∞ filter and the IMM-KF; the results testify to its advantage over the other two methods.

References

Ababsa, F., Mallem, M., Roussel, D., 2004. Comparison between particle filter approach and Kalman filter-based technique for head tracking in augmented reality systems. *IEEE ICRA*, **1**:1021-1026. [doi:10.1109/ROBOT.2004.1307284]

Bar-Shalom, Y., Li, X.R., Kirubarajan, T., 2001. Estimation with Applications to Tracking and Navigation. John Wiley & Sons.

Blom, H.A.P., Bloem, E.A., 2004. Particle filtering for stochastic hybrid systems. *IEEE Conf. Decision and Control*, **3**:3221-3226.

Boers, Y., Driessen, J.N., 2003. Interacting multiple model particle filter. *IEEE Proc. RSN*, **150**(5):344-349. [doi:10.1049/ip-rsn:20030741]

Fortmann, T., Bar-Shalom, Y., Scheffe, M., 1983. Sonar tracking of multiple targets using joint probabilistic data association. *IEEE Journal of Oceanic Engineering*, **8**(3):173-184. [doi:10.1109/JOE.1983.1145560]

Kotecha, J.H., Djuric, P.M., 2003. Gaussian sum particle filtering. *IEEE Trans. SP*, **51**(10):2602-2612. [doi:10.1109/TSP.2003.816754]

McGinnity, S., Irwin, G.W., 2000. Multiple model bootstrap filter for maneuvering target tracking. *IEEE Trans. AES*, **36**(3):1006-1012. [doi:10.1109/7.869522]

Ristic, B., Arulampalam, S., Gordon, N., 2004. Beyond the Kalman Filter. Artech House.

Tsaknakis, H., Athans, M., 1994. Tracking maneuvering targets using H_∞ filters. *ACC*, **2**:1796-1803. [doi:10.1109/ACC.1994.752382]

Zhou, H.R., Jing, Z.L., Wang, P.D., 1991. Tracking of Maneuvering Targets. National Defense Industry Press, Beijing (in Chinese).