An MCMC-based Particle Filter for Tracking Target in Glint Noise Environment

Hongtao Hu   Zhongliang Jing   Anping Li   Shiqiang Hu   Hongwei Tian

Institute of Aerospace Information and Control School of Electrical and Information Engineering
Shanghai Jiao Tong University
Shanghai
20030, P.R.China
hht@sjtu.edu.cn   zljing@sjtu.edu.cn

Abstract - *In radar tracking application, the observation noise is highly non-Gaussian, which is also referred as glint noise. The performance of extended Kalman filter degrades severely in the presence of glint noise. In this paper, an improved particle filter, Markov chain Monte Carlo particle filter (MCMC-PF), is introduced to cope with radar target tracking in glint noise environment. The Monte Carlo simulation results show that MCMC-PF can efficiently track target both in Gaussian noise and glint noise environments.*

Keywords: Particle filter, Markov chain Monte Carlo, glint noise, extended Kalman filter, tracking.

1 Introduction

In radar tracking systems the measurements of the position of a target is reported in polar coordinates with respect to the sensor location. So radar target tracking is a nonlinear problem. Since 1970s many methods have been proposed to solve the problem of nonlinear filtering, such as extended Kalman filter, debiased consistent converted measurements Kalman filter, unscented Kalman filter and so on [1,2-3]. But all these algorithms rely on Gaussian assumption. Their performance degrades severely when the plant or observation disturbances are not Gaussian, particularly when the observation is heavy-tailed. In real radar target tracking systems, changes in the target aspect with respect to the radar can cause the apparent center of radar reflections (direction “seen” by the antenna) to wander significantly. The random wandering of the apparent radar reflecting center gives rise to noisy or jittered angle tracking. This form of measurement noise is called angle fluctuations or target glint [4]. Glint affects the measurement components (mostly the angles) by producing heavy-tailed, non-Gaussian disturbances [5], which may severely affect the tracking accuracy.

Recently particle filter has captured the attention of many researchers in various communities including those of signal processing, statistics, and econometrics, and this interest stems form its potential for coping with difficult nonlinear and/or non-Gaussian problems [6]. In generic particle filter, depletion of samples may appear. To overcome this problem, MCMC-based particle filter (MCMC-PF) has been proposed in the literature [7]. This paper introduces the MCMC-PF to solve the tracking problem with glint noise. The results show MCMC-PF can efficiency track target both in Gaussian and glint noise environment.

This paper is organized as follows. Section 2 describes the MCMC-PF algorithm. Section 3 describes the glint noise and target tracking problem. Section 4 presents the simulation results and analysis. Finally, conclusions of the study are given in Section 5.

2 MCMC-based Particle Filter

2.1 Recursive Bayesian Estimation

The time evolution of the state $x_k \in \mathrm{R}^n$ of a discrete-time dynamic system is described by

$$x_k = f_k(x_{k-1}, \omega_k) \tag{1}$$

where f_k is the system transition function and $\omega_k \in \mathrm{R}^p$ is a dynamic noise which has a known probability density function (PDF). At discrete times, measurements (or observations) $y_k \in \mathrm{R}^m$ of the state x_k become available and are related to the state through the observation equation

$$y_k = h_k(x_k, \upsilon_k) \tag{2}$$

where h_k is the measurement function and $v_k \in \mathrm{R}^m$ is an observation noise of known PDF. At time k the set of measurements $D_k = \{y_1, \dots, y_k\}$ is available and the PDF $p(x_0)$ of the initial state is assumed to be known.

The optimal estimator $\hat{x}_k$ (in the minimum variance sense) is provided by the conditional expectation $E[x_k \mid D_k]$. The problem of assessing this conditional expectation is obviously related to the computation of the conditional PDF $p(x_k \mid D_k)$, which is classically obtained in two stages: prediction and update.

Prediction: Assume that the required PDF $p(x_{k-1} | D_{k-1})$ is known at time $k-1$. The prediction step yields $p(x_k | D_{k-1})$ by a simple application of the Chapman-Kolgomorov equation

$$p(x_k | D_{k-1}) = \int p(x_k | x_{k-1}) p(x_{k-1} | D_{k-1})\, dx_{k-1} \quad (3)$$

Update: At time k, when y_k becomes available, the prior PDF $p(x_k | D_{k-1})$ is updated via Bayes' rule

$$p(x_k | D_k) = \frac{p(y_k | x_k) p(x_k | D_{k-1})}{p(y_k | D_{k-1})} \quad (4)$$

where $p(y_k | D_{k-1})$ is the normalizing denominator. Equations (3) and (4) represent a general recursive solution for the Bayesian recursive estimation. Unfortunately, this recursive propagation is only a conceptual solution in that in general, it cannot be determined analytically except for some special cases. For example, when f_k and h_k are linear, and ω_k and v_k are additive Gaussian noises with known PDF, (3) and (4) give rise to the well-known Kalman filter algorithm. For solving this problem, many sub-optimal algorithms have been proposed such as extended Kalman filter (EKF), approximate grid-based methods and particle filter [8].

2.2 Basic particle filter

The particle filter is a new filtering method based on Bayesian estimation and Monte Carlo method and can effectively cope with complicated nonlinear and/or non-Gaussian problems. The basic idea of this method is Monte Carlo simulation, in which the posterior density is approximated by a set of particles with associated weights $\left\{\left(x_{k-1}^i, w_{k-1}^i\right) \quad i = 1,2\cdots,N\right\}$. At every time step we sample form the proposal distribution $q(x_k | x_{k-1}, D_k)$ to achieve new particles and compute new weights according to the particle likelihoods. After normalisation of weights, the posterior density can be represented by $\left\{\left(x_k^i, w_k^i\right) \quad i = 1,2\cdots,N\right\}$.

The basic particle filter implementation is as follows:

The Basic Particle Filter

1 Initialization

Draw a set of particles from the prior $p(x_0)$:

$x_0^i \sim p(x_0)$, $i = 1,2\cdots,N$, let $k = 1$

2 Sampling step

a) For $i = 1,2\cdots,N$

Sample x_k^i from the proposal distribution $q(x_k | x_{k-1}^i, D_k)$: $x_k^i \sim q(x_k | x_{k-1}^i, D_k)$

b) Evaluate the importance weights

$w_k^i = w_{k-1}^i \dfrac{p(y_k | x_k^i) p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{k-1}^i, D_k)}$, $i = 1,2\cdots,N$

c) Normalise the weights

$\tilde{w}_k^i = w_k^i / \sum_{j=1}^{N} w_k^j$, $i = 1,2\cdots,N$

3 Selection (resampling) step

Resample particles x_k^i with probability $\tilde{w}_k^i$ to obtain N random particles x_k^j, approximately distributed according to $p(x_k | D_k)$.

Set $w_k^i = \tilde{w}_k^i = 1/N$, $i = 1,2\cdots,N$

4 Output step

Output a set of particles $\left\{\left(x_k^i, w_k^i\right) \quad i = 1,2\cdots,N\right\}$ that can be used to approximate the posterior distribution.

Expectation: $\hat{x}_k = \sum_{i=1}^{N} w_k^i x_k^i$

Covariance: $P_k = \sum_{i=1}^{N} w_k^i (x_k^i - \hat{x}_k)(x_k^i - \hat{x}_k)^T$

5 $k = k+1$, go to step 2.

2.3 Markov Chain Monte Carlo

The selection procedure based on the importance weights described earlier greatly diminishes the diversity of the particles. Therefore, we require a procedure to introduce sample variety after the selection step without affecting the validity of the approximation. In this paper Markov chain Monte Carlo (MCMC) step has been used to solve this problem.

The MCMC step, as described in [7] has an invariant distribution $\prod_{i=1}^{N} p(x_{0:k}^i | D_k)$, which is applied to each of the N particles, one at the time. The Metropolis-Hastings (MH) algorithm is a way to simulate from such a chain. In a nutshell, the idea of MH is to sample states form a Markov chain with the posterior as invariant distribution. Such a Markov chain is constructed by choosing a candidate for the next state $x_k^{*(i)}$ given the current state x_{k-1}^i according to a proposal distribution $p(x_k | x_{k-1})$. This state transition is accepted with probability

$$\alpha(x_{k-1}^i, x_k^{*(i)}) = \min\left\{1, \frac{p(y_k | x_k^{*(i)})}{p(y_k | x_k^{(i)})}\right\} \quad (5)$$

The pseudo-code for the MH algorithm is as follows:

MH step

1 Sample $v \sim \mathcal{U}[0,1]$, $\mathcal{U}[0,1]$ is uniformly distribution in the interval [0,1].

2 Sample the proposal candidate $x_k^{*(i)} \sim p(x_k | x_{k-1}^i)$

3 If $v \le \min\left\{1, \dfrac{p(y_k | x_k^{*(i)})}{p(y_k | x_k^i)}\right\}$

then accept move:

$x_k^i = x_k^{*(i)}$

else reject move:

$x_k^i = x_k^i$

end if

3 Radar Target Tracking

3.1 Target Tracking Problem Formulation

We consider the nearly constant velocity (NCV) motion of a target in the XY plane[1]. The discrete-time state equation with sampling interval T is

$$X(k+1) = FX(k) + \Gamma\omega(k) \tag{6}$$

where

$$X(k) = \left[x(k), \dot{x}(k), y(k), \dot{y}(k)\right]^T \tag{7}$$

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

$$\Gamma = \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix} \tag{9}$$

and $\omega(k)$ is a discrete-time white, Gaussian noise:

$$\omega(k) \sim \mathcal{N}(0\,,\,\mathrm{Q_k}) \tag{10}$$

$$Q_k = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} \tag{11}$$

2-D radar measures the range (r), azimuth (α) of a target. The radar measurement equation is described by

$$Z(k) = h(X(k), s(k)) + v(k) \tag{12}$$

where $s(k)$ and $v(k)$ are sensor position and measurement noise at time k :

$$s(k) = [s_x(k), s_y(k)]' \tag{13}$$

$$h(X(k), s(k)) = \begin{bmatrix} r(k) \\ \alpha(k) \end{bmatrix} = \begin{bmatrix} \sqrt{(x(k)-s_x(k))^2 + (y(k)-s_y(k))^2} \\ \arctan\left(\dfrac{y(k)-s_y(k)}{x(k)-s_x(k)}\right) \end{bmatrix} \tag{14}$$

$$v(k) = \left[v_r(k), v_\alpha(k)\right]' \tag{15}$$

In ideal environment we can assume $v(k)$ is a zero-mean independent Gaussian noise with covariance R_k :

$$R_k = \begin{bmatrix} \sigma_r^2(k) & 0 \\ 0 & \sigma_\alpha^2(k) \end{bmatrix} \tag{16}$$

but in real environment, $v(k)$ is a non-Gaussian noise, which is referred as glint noise. Section 3.2 describes the statistical characteristics of the glint noise.

3.2 Glint Noise

In radar target applications, the observation noise is highly non-Gaussian. It is well documented in the literature that the so-called “glint noise” possesses the characteristics of a long-tailed distribution [5]. Conventional minimum mean square estimators can be seriously degraded if non-Gaussian noise is present. Therefore, it is of paramount importance to have accurate modeling of the non-Gaussian noise phenomenon prior to the development of any efficient tracking algorithm. Many different models have been used for the non-Gaussian glint noise present in target tracking applications. Among them is a mixture approach, originally proposed by Hewer et al. [5], which argues that the radar glint noise can be modeled as a mixture of background Gaussian noise with outliers. Their results were based on the analysis of the QQ-plot of glint noise records. Examination of such records reveals that the glint QQ-plot is fairly linear around the origin, an indication that the distribution is Gaussian-like around its mean. However, in the tail region, the plot deviates form linearity and indicates a non-Gaussian, long-tailed character. The data in the tail region are essentially associated with the glint spikes and are considered to be outliers. These outliers have a considerable influence on conventional target tracking filters, such as the Kalman filter, which is quite non-robust. The effect of the glint spikes is even greater on the sample variance used in the derivation of the filter’s gain. It is not difficult to see that variances which are quadratic functions of the data are more sensitive to outliers than the sample means. Therefore, the glint spikes can be modeled as a Gaussian noise with large variance, resulting in an overall glint noise model which can be considered as a Gaussian mixture with the two components used to model the background (thermal) Gaussian noise and the glint spikes, respectively. The weighting coefficients (glint probability) in the mixture (percentage of contamination) can be used to model the non-Gaussian nature of the glint spikes. Therefore, the glint noise model can be generated as the mixture of two Gaussian distributions, each with zero mean and with fixed variance.

Assuming that the Gaussian terms are denoted as $\mathcal{N}_1(\mu_1, P_1)$ and $\mathcal{N}_2(\mu_2, P_2)$, the mixture distribution has the following form:

$$f(V) = (1-\varepsilon)N(\mu_1, P_1) + \varepsilon N(\mu_2, P_2) \tag{17}$$

with $0 < \varepsilon < 1$. A random variable V of this distribution can be generated by first selecting uniformly a sample U from the interval (0,1). If $U > \varepsilon$, then V is generated by an independent sample form $\mathcal{N}_1(\mu_1, P_1)$. Otherwise, the requested variable V is a sample from $\mathcal{N}_2(\mu_2, P_2)$.

3.3 Tracking Algorithms

Here we use two algorithms (EKF and MCMC-PF) tracking a target considering two scenarios described above.

In the first scenario, assuming $v(k) \sim \mathcal{N}(0, R)$, we use standard EKF equations in [1] with the equivalent measurement matrix given by

$$H(k) = \left.\frac{\partial h(X(k))}{\partial X}\right|_{\hat{X}(k|k-1)}$$

$$= \begin{bmatrix} \frac{\hat{x}-x_0}{\sqrt{(\hat{x}-x_0)^2+(\hat{y}-y_0)^2}} & 0 & \frac{\hat{y}-y_0}{\sqrt{(\hat{x}-x_0)^2+(\hat{y}-y_0)^2}} & 0 \\ \frac{-(\hat{y}-y_0)}{(\hat{x}-x_0)^2+(\hat{y}-y_0)^2} & 0 & \frac{\hat{x}-x_0}{(\hat{x}-x_0)^2+(\hat{y}-y_0)^2} & 0 \end{bmatrix}_{k|k-1} \tag{18}$$

where $\hat{X}$ is one step prediction value. When we use MCMC-PF algorithm mentioned above, the importance step evaluating the importance weights given by:

$$w_k^i = w_{k-1}^i p(Z(k) \mid X^i(k \mid k-1)) = w_{k-1}^i \mathcal{N}(u(k);0,R) \tag{19}$$

where u_k is innovation:

$$u(k) = Z(k) - h(\hat{X}(k \mid k-1)) \tag{20}$$

$\mathcal{N}(u(k);0,R)$ denotes a Gaussian PDF at $u(k)$ with mean 0 and variance R.

In other scenario, the measure noise is glint noise whose PDF described by Eq (17). A mixture PDF can be approximated by a single Gaussian PDF with moments equal to those of the mixture. The first and second moment is:

$$\mu = E[V] = (1-\varepsilon)\mu_1 + \varepsilon\mu_2 \tag{21}$$

$$P = E[(w-\mu)(w-\mu)^T] = (1-\varepsilon)P_1 + \varepsilon P_2 + \tilde{P} \tag{22}$$

where $\tilde{P} = (1-\varepsilon)\mu_1\mu_1^T + \varepsilon\mu_2\mu_2^T - \mu\mu^T$.

Then we use standard EKF algorithm to track target. When we use MCMC-PF algorithm, the evaluating the importance weights procedure given by:

$$\begin{aligned} w_k^i &= w_{k-1}^i p(Z(k) \mid X^i(k \mid k-1)) \\ &= w_{k-1}^i((1-\varepsilon)\mathcal{N}(u(k);\mu_1,P_1) + \varepsilon\mathcal{N}(u(k);\mu_2,P_2)) \end{aligned} \tag{23}$$

4 Simulation Analysis

We use 100 Monte-Carlo runs. The performance of the two algorithms was compared based on RMS errors in target position and velocity along the x- and y-axis. The numerical values of the parameters of the dynamic system are as follows.

Sample interval $T = 1$ s

Process noise covariance $Q = diag(5m/s^2, 5m/s^2)$

Initial state $X(0) = [50km, 0.3km/s, 50km, -0.1km/s]'$

Radar position $s = [0,0]'$ km

Particle number $N = 1000$

4.1 Gaussian Noise environment

Covariance matrix for the Gaussian measurement noise $R = diag(250m^2, 1\deg^2)$. Figure 1-4 shows the RMS error of position and velocity.

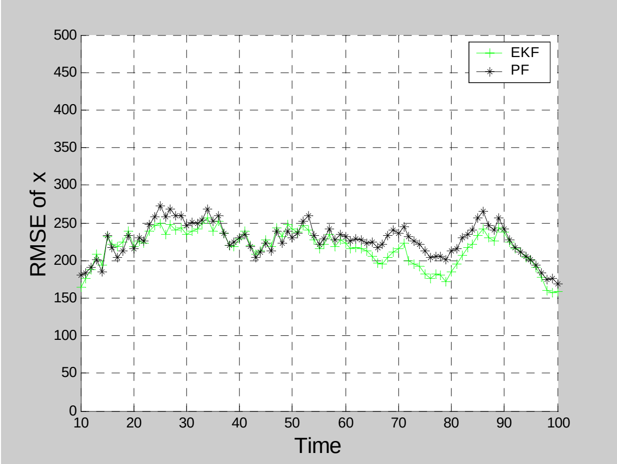

Fig.1 RMS position error of x-axis

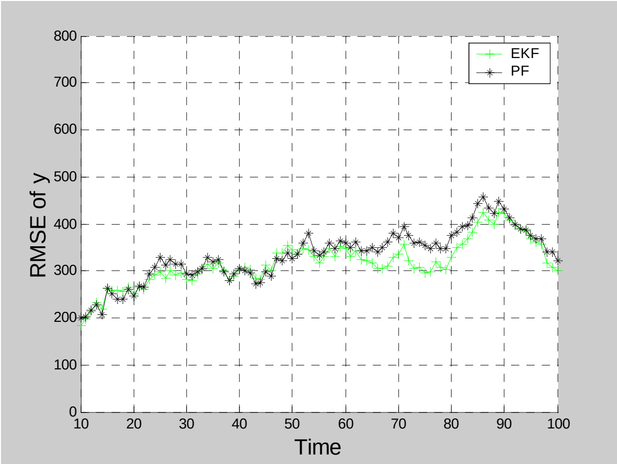

Fig.2 RMS position error of y-axis

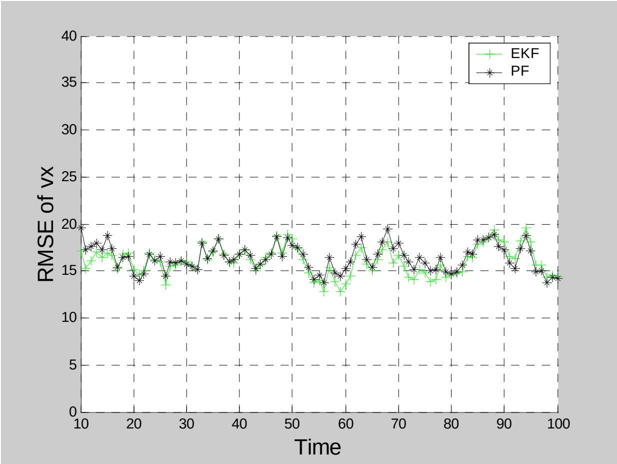

Fig.3 RMS velocity error of x-axis

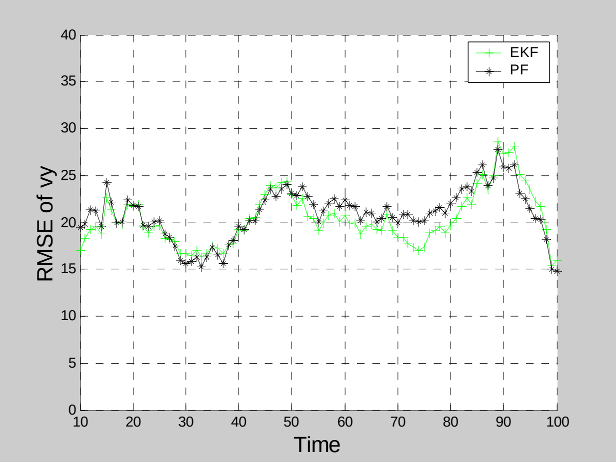

Fig.4 RMS velocity error of y-axis

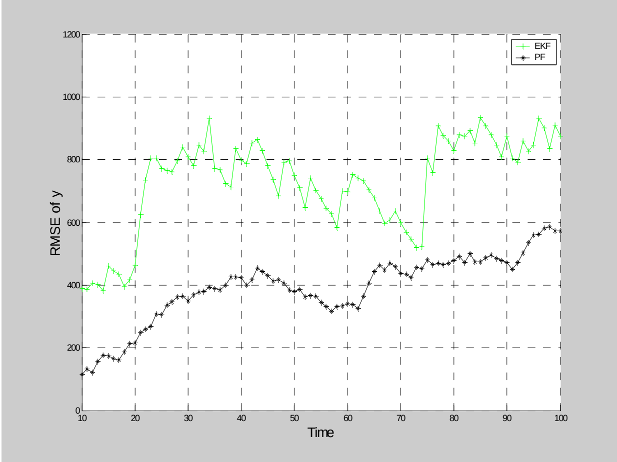

Fig.6 RMS position error of y-axis

4.2 Glint Noise entironment

Glint probability $\varepsilon = 0.05$
Covariance matrix for background Gaussian noise $P_1 = diag(2500m^2, 1\deg^2)$
Covariance matrix for glint spikes Gaussian noise $P_2 = diag(2500m^2, 100\deg^2)$
Figure 5-8 shows the RMS error of position and velocity. Table 1 gives the average RMS at different ε .

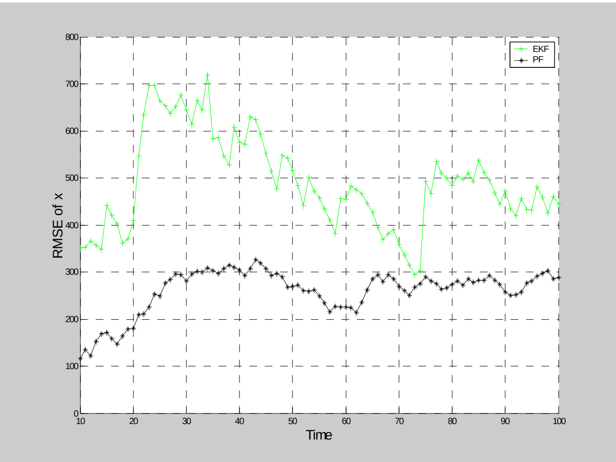

Fig.5 RMS position error of x-axis

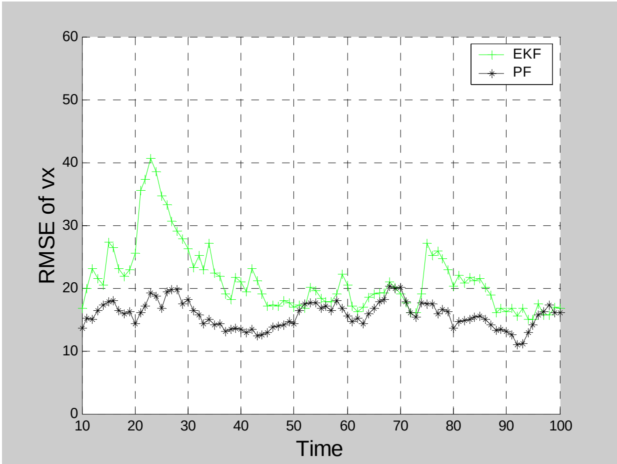

Fig.7 RMS velocity error of x-axis

Table 1 RMS of different glint probability

Glint probability ε	Position of X-axis (m)		Velocity of X-axis (m/s)		Position of Y-axis (m)		Velocity of Y-axis (m/s)	
	EKF	PF	EKF	PF	EKF	PF	EKF	PF
0.05	490	260	21	16	726	390	32	21
0.1	611	266	24	16	899	403	36	21
0.2	859	270	33	17	1246	420	50	24

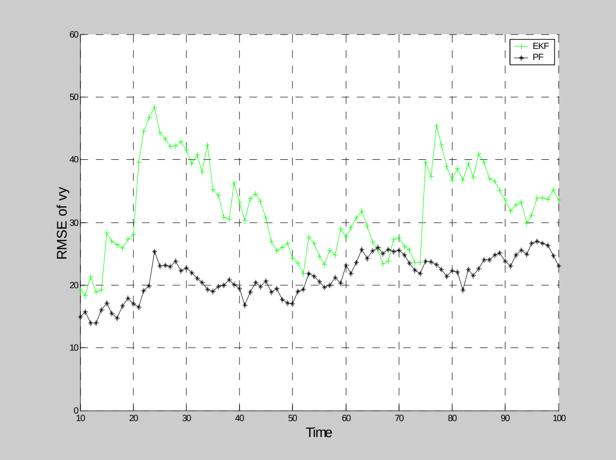

Fig.8 RMS velocity error of y-axis

4.3 Result Analysis

By looking the results, figs.1-4 shows when in Gaussian environment EKF has a little better tracking accuracy than MCMC-PF, since we only use 1000 particles in MCMC-PF. If we increase particle numbers, the accuracy of MCMC-PF also will increase. Figs.5-8 shows in glint noise environment the performance of MCMC-PF is clearly superior to that of the EKF, since glint noise is non-Gaussian. From table 1 we can see that RMS of EKF rapidly increases as the glint probability ε increasing, and RMS of MCMC-PF only little increase. It is because that with the glint probability ε increasing the glint noise is more non-Gaussian. MCMC-PF is robust since it uses exact noise PDF.

5 Conclusion

We have considered the application of the MCMC-PF algorithm for the problem of target tracking when the measurement noise is corrupted by glint. It has been shown that the in Gaussian environment both EKF and MCMC-PF have almost the same tracking performance, and that in glint noise environment the performance of MCMC-PF is clearly superior to that of the EKF.

Acknowledgements

This work was jointly supported by National Natural Science Foundation of China (60375008), Shanghai Key Scientific Project (02DZ15001), China PH.D Discipline Special Foundation (20020248029) and China Aviation Science Foundation (02D57003).

References

[1] Bar-Shalom Y. X.R Li. *Estimation with Applications to Tracking and Navigation*. Artech House, New York, 2001.

[2] Lerro D. Bar-Shalom Y. Tracking with Debiased Consistent Converted Measurements Versus EKF. *IEEE Trans. AES*, 29(3):1015-1022, 1993.

[3] Julier S.J. Uhlmann J.K. *Proc. the 11th International Sumposium on Aerospace/Defence Sensing, Simulation and Controls. A new extension of the Kalman filter to nonlinear systems*. Orlando, Florida, 1997.

[4] Skolnik M.L. *Introduction to Radar Systems*. McGraw-Hill, New York, 1980.

[5] Hewer G.A. Martin R.D. and Zeh J. Robust Preprocessing for Kalman Filtering of Glint Noise. *IEEE Trans. AES*, 23:120-128, 1987.

[6] Djuric P.M. Kotecha J.H. and et.al. Particle Filtering. *IEEE Trans. Signal Processing*, 20:19-28, 2003.

[7] Van der Merwe R. Doucet F. and et.al. The Unscented Particle Filter. Technical Report CUED/F-INFENG/TR 380, Cambridge University Engineering Department, Cambridge, England, 2000.

[8] Arulampalam M.S. Maskell S. and et.al. A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking. *IEEE Trans. Signal Processing*, 50:174-188, 2002.