An Improved Algorithm Based on Particle Filter for 3D UAV Target Tracking

Wenjing Xie, Li Wang, *Senior Member, IEEE*, Bo Bai, *Senior Member, IEEE*,
Bile Peng, *Member, IEEE*, and Zhiyong Feng, *Senior Member, IEEE*

***Abstract*—The widespread application of unmanned aerial vehicles (UAVs) urgently requires an effective tracking algorithm as technical support. Particle filter has been widely applied in maneuvering target tracking, however, there has been no suitable solution to the trade-off between weight degeneracy and particle diversity during the process of resampling. In this paper, we propose an improved particle filter algorithm based on systematic resampling with additional random perturbation. This method ensures that particle filter maintains particle diversity and reduces weight degeneracy under environments with different noise types, simultaneously. The simulation results demonstrate that the proposed algorithm generates more accurate filtered trajectory than generic particle filter, especially under the environment with low noise.**

***Index Terms*—particle filter, target tracking, resampling**

I. Introduction

The proliferation of small maneuvering targets such as unmanned aerial vehicles (UAVs) has raised the demand for target tracking algorithms with better performance in flight control areas. In the meantime, the working environment of UAVs is usually in remote areas where UAVs are prone to loss of control. In order to effectively track a maneuvering target like UAV, it is necessary to locate the present position and predict the following motion state. The measurement indicators in the positioning problem include Angle of Arrival (AOA), Time of Arrival (TOA), Time Difference of Arrival (TDOA) and Received Signal Strength (RSS) [1]. For a maneuvering target capable of flying, the motion state model extends from two-dimensional (2D) to three-dimensional (3D). Considering the feasibility and accuracy of outdoor measurement, AOA and distance from targets to measuring equipment can be adopted as indicators to describe the position of 3D maneuvering targets, which results in the nonlinearity of motion state model. Due to the instability of outdoor measurement conditions, the noise distribution is not limited to Gaussian distribution.

In order to track targets more accurately, several filter-based algorithms have been proposed to reduce the effect of measurement noise during tracking process. Kalman Filter (KF), which utilizes the mean and covariance of the state in its update rule, was first introduced into the linear target tracking [2]. For motion models that cannot be represented linearly, Extended Kalman Filter (EKF), by assuming that all transformations are quasi-linear, linearizes all nonlinear transformations and substitutes Jacobian matrices for the linear transformations in KF [3]. To address the deficiencies of linearization in EKF, Unscented Kalman Filter (UKF) is developed by providing a more direct and explicit mechanism for transforming mean and covariance information [4].

The aforementioned three filters are only suitable for Gaussian noise environment, and the application scenarios are 2D mostly. With the growth of maneuvering targets, the motion model is no longer limited to 2D, which has put forward higher requirements for target tracking algorithms. Particle Filter (PF), which is designed for nonlinear systems without limitation of the noise distribution type [5], is the most promising algorithm for high-dimensional scenario with various noise features. However, particle filter still have some problems when being applied to target tracking, such as the trade-off between maintaining particle diversity and discarding invalid particles during resampling. To solve this problem, researches have proposed various solutions, including: 1) eliminating the dependency between samples [6]; 2) optimizing the process of weight assignment [7]; 3) decomposing resampling into multiple steps [8]. Although these works have achieved significant performance, they consider that the external conditions obey constant distribution. To ensure the accuracy of tracking, 3D system model should be more robust to the noise variation.

Enlightened by previous works, this paper establishes a 3D nonlinear flight model of UAV, and based on the model, an improved algorithm, which adds random perturbation instead of modifying the resampling process, is proposed. Specifically, after resampling, the particles are averaged based on the reassigned weights. In order to supplement the loss of diversity caused by discarding particles, a random perturbation is added to the weighted mean, so that the predicted prior information at the next moment is not completely consistent with the

This work was supported in part by the National Natural Science Foundation of China under Grant 61571056 and 61871416, in part by the Beijing Science and Technology Nova Program under Grant xx2018083, in part by the Fundamental Research Funds for the Central Universities under Grant 2018XKJC03, and in part by the Beijing Municipal Natural Science Foundation under Grant L172010.

L. Wang (*Corresponding author*) is with the School of Electronic Engineering, Beijing University of Posts and Telecommunications, the Key Laboratory of the Universal Wireless Communications, Ministry of Education, and she is also with the Beijing Laboratory of Work Safety Intelligent Monitoring, Beijing 100876, China (e-mail: liwang@bupt.edu.cn).

W. Xie is with the School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: xiewenjing@bupt.edu.cn).

B. Bai is with the Future Network Theory Laboratory, Huawei Technologies Co., Ltd., Shatin Hong Kong (e-mail: baibo8@huawei.com).

B. Peng is with the Department of Electrical Engineering, Chalmers University of Technology, Gothenburg 41258, Sweden (e-mail: bile.peng@chalmers.se).

Z. Feng is with the Key Laboratory of Universal Wireless Communications, Ministry of Education, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: fengzy@bupt.edu.cn).

978-1-5386-8088-9/19/$31.00 ©2019 IEEE

filtered value of the current moment. In summary, the main contributions of this paper are two-fold:

(1) A 3D flight model is established according to the maneuvering characteristics of UAV, where particle filter tracks a total of nine states in three dimensions.
(2) Compared with conventional algorithms solving the problem caused by resampling, such as regularized particle filter, the proposed algorithm introduces minor modification to generic particle filter, which effectively reduces computational complexity.

The remainder of this paper is organized as follows. Section II introduces the principle of particle filter by mathematical derivation. Section III illustrates the system model. Section IV deduces the formula of weight update and gives the framework of the improved algorithm. Section V validates the accuracy and computational complexity of the proposed algorithm and demonstrates the superiority of this work. Finally, Section VI concludes this paper.

II. Preliminaries on Particle Filter

Particle filter is based on Monte Carlo and recursive Bayesian [9]. The implementation can be described as

1) Prediction: The following state is predicted based on existing prior knowledge.

2) Update: The prediction is amended based on the latest measurement to obtain the posterior probability density.

The principle of particle filter will be gradually explained according to the following five concepts.

A. Bayesian Filter

The state and measurement equation of a system are given as follows

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

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

where x, y, v, u, f, and h denote the state of system, measurements, process noise, measurement noise, state transition function, and measurement function, respectively.

The posterior probability of the present state x_k results from $y_{1:k}$ (measurements from time 1 to k), which is expressed as $p(x_k|y_{1:k})$. Assume the state transition obeys first-order Markov model, i.e., x_k is only related to x_{k-1} and y_k is only related to x_k.

1) Prediction: The probability of x_k can be derived as

$$p(x_k\,|y_{1:k-1}) = \int p(x_k\,|x_{k-1})p(x_{k-1}\,|y_{1:k-1})dx_{k-1}, \tag{3}$$

where $p(x_k|x_{k-1})$ is decided by Eq.(1).

2) Update: The update step obtains $p(x_k|y_{1:k})$ by amending $p(x_k|y_{1:k-1})$ with y_k

$$p(x_k\,|y_{1:k}) = \frac{p(y_k\,|x_k)p(x_k\,|y_{1:k-1})}{p(y_k\,|y_{1:k-1})}. \tag{4}$$

For nonlinear and non-Gaussian system, it is difficult to obtain the analytical solution of posterior probability. To solve the problem, Monte Carlo Sampling (MCS) is introduced.

B. Monte Carlo Sampling

Suppose particles can be sampled from a target probability distribution $p(x)$, and the expectation of functions obeying $p(x)$ can be estimated [10], MCS obtains the expectation of $p(x)$ by averaging the sampled values instead of integrating

$$E[f(x)] = \int f(x)p(x\,|y_{1:k})dx \approx \frac{1}{N}\sum\nolimits_{i=1}^{N} f(x_i), \tag{5}$$

where x_i denotes the ith sample from $p(x)$.

The key problem in the above is that posterior probability is unknown, which results in impossibility of sampling particles.

C. Importance Sampling

Instead of sampling from posterior probability, Importance Sampling samples from a known distribution $q(\cdot)$ [11]. Thus

$$\begin{aligned} E[f(x_k)] &= \int f(x_k)\frac{p(x_k\,|y_{1:k})}{q(x_k\,|y_{1:k})}q(x_k\,|y_{1:k})dx_k \\ &\approx \frac{\frac{1}{N}\sum_{i=1}^{N} W_k(x_k^{(i)})f(x_k^{(i)})}{\frac{1}{N}\sum_{i=1}^{N} W_k(x_k^{(i)})}, \\ &\left(W_k(x_k) = \frac{p(y_{1:k}\,|x_k)p(x_k)}{q(x_k\,|y_{1:k})} \propto \frac{p(x_k\,|y_{1:k})}{q(x_k\,|y_{1:k})}\right). \end{aligned} \tag{6}$$

Each particle has corresponding normalized weight. The larger the weight is, the higher the credibility is. However, $p(x_k|y_{1:k})$ is recalculated every time a sample is added. To simplify calculation, weights can be calculated recursively.

D. Sequential Importance Sampling (SIS)

It can be assumed that importance distribution $q(\cdot)$ satisfies

$$q(x_k\,|x_{0:k-1}, y_{1:k}) = q(x_k\,|x_{k-1}, y_{1:k}). \tag{7}$$

Thus the recursion of particle weights can be expressed as

$$w_k^{(i)} \propto \frac{p(x_{0:k}^{(i)}\,|y_{1:k})}{q(x_{0:k}^{(i)}\,|y_{1:k})} = w_{k-1}^{(i)}\frac{p(y_k\left|x_k^{(i)}\right.)p(x_k^{(i)}\left|x_{k-1}^{(i)}\right.)}{q(x_k^{(i)}\left|x_{k-1}^{(i)}, y_k\right.)}. \tag{8}$$

E. Resampling

In SIS, part of particle weights will decrease to be negligible gradually (weights degeneracy). The number of effective particles (N_{eff}) denotes the degree of weights degeneracy as

$$N_{eff} \approx \frac{1}{\sum\limits_{i=1}^{N} {(w_k^{(i)})}^2}. \tag{9}$$

Resampling is a frequently-taken solution to weights degeneracy [5], during which particles with small weight are discarded and replaced by new ones.

By resampling, the expectation can be expressed as

$$\tilde{p}(x_k\,|y_{1:k}) = \sum_{j=1}^{N}\frac{1}{N}\delta(x_k - x_k^{(j)}) = \sum_{i=1}^{N}\frac{n_i}{N}\delta(x_k - x_k^{(i)}), \tag{10}$$

where $x_k^{(i)}$ denotes particle at time k, $x_k^{(j)}$ denotes $x_k^{(i)}$ after resampling, n_i denotes the number of times $x_k^{(i)}$ is copied.

By combining resampling and SIS, generic particle filter algorithm is formed.

III. System Model

This paper considers a scenario that a ground-measuring equipment locates a small 3D maneuvering UAV via three measurements indicators: the distance from the UAV to the ground-measuring device, elevation angle, and azimuth, as shown in Fig.1. The measurement is performed in a spherical coordinate system in which measuring device is at the spherical center. Angular variation caused by the same displacement of UAV moving at different positions is uneven, which makes the ordinary 3D acceleration model become a completely nonlinear model in spherical coordinates.

In order to display the trajectory more intuitively, after the positioning result is filtered, the transformation from spherical coordinates to Cartesian coordinates is performed, thus the trajectory can be displayed in the form of 3D Cartesian coordinate system.

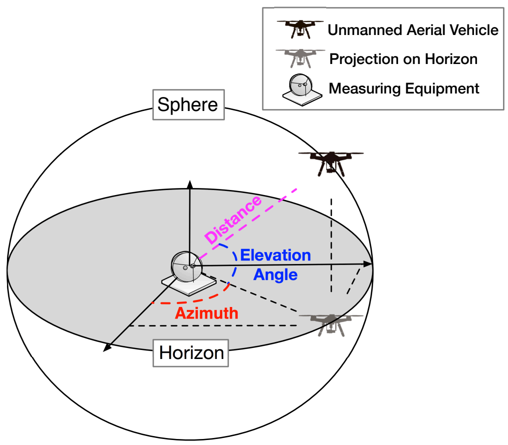

Fig. 1. Measurement Scenario in Spherical Coordinates

A. Motion State Model

The motion state model is built in 3D Cartesian coordinate. The UAV performs uniform acceleration linear motion on the three axes, which can be described by the following equation of motion state

$$\mathcal{S}_k = \begin{pmatrix} 1 & t & \frac{t^2}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & t & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & t & \frac{t^2}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & t & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & t & \frac{t^2}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & t \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \mathcal{S}_{k-1} + u_{k-1}, \tag{11}$$

where u_{k-1} denotes the noise caused by state transition process at time $k-1$ (i.e., process noise), t denotes one - time step and $\mathcal{S}_k$ denotes the motion state at time k. State $\mathcal{S}$ contains displacement, velocity, and acceleration of the target in three axes respectively

$$\mathcal{S} = \begin{pmatrix} x & v_x & a_x & y & v_y & a_y & z & v_z & a_z \end{pmatrix}^{\mathrm{T}}. \tag{12}$$

Considering that the actual measurement indicators are distance, elevation angle and azimuth, after generating a flight path of UAV according to the motion state equation, it needs to be converted into spherical coordinates

$$\begin{aligned} r_k &= \sqrt{x_k^2 + y_k^2 + z_k^2}, \\ \theta_k &= \frac{\pi}{180} \cdot \arctan \frac{z_k}{\sqrt{x_k^2 + y_k^2}}, \\ \varphi_k &= \frac{\pi}{180} \cdot \arctan \frac{y_k}{x_k}, \end{aligned} \tag{13}$$

where r_k, θ_k and φ_k denote distance, elevation angle and azimuth at time k respectively.

The value of the inverse tangent function is $[-\pi, \pi]$, which may cause the calculated elevation angle to be negative. Therefore, to prevent from being physically meaningless, the calculated negative result need to be adjusted to the range of $[0, 2\pi]$ according to the periodicity of the trigonometric function.

Then, the measurement noise is added to the trajectory of UAV generated according to the motion equation

$$M_k = \begin{pmatrix} r_k & \theta_k & \varphi_k \end{pmatrix}^T + v_k, \tag{14}$$

where M_k and v_k denotes the measurement and measurement noise at time k, respectively. In this way, the ultimate measurements of particle filter are obtained.

B. Process of Filtering

1) Initialization: The value of initial state matrix S is set according to the flight characteristics of UAV, and magnitude and probability distribution type of process noise u and measurement noise v are defined respectively. In the meantime, the number of particles N and flight time step T are initialized.

2) Sampling: At time k, the motion state is predicted based on all predictions of particle filter from time 1 to $k-1$, and the measurements are obtained by adding noise to motion state equation. Then, the weights of all particles are calculated and normalized.

Algorithm 1: Systematic Resampling

Input: W_i (Weight of particle i, i=1:N)
Output: n_i (Number of particle i, i=1:N)
1 Generate N random numbers:
2 $\widetilde{u_k} = \frac{(k-1)+u}{N}$ $(u \sim U[0,1])$
3 Calculate n_i:
4 n_i= the number of $\widetilde{u_k} \in \left(\sum_{s=1}^{i-1} W_s, \sum_{s=1}^{i} W_s\right]$
5 Reset the sampled values of N particles.

3) Resampling: At each moment, the effective particle number N_{eff} is updated based on the weights of particles to determine whether resampling is performed. Conventional resampling methods include Simple Random Resampling, Stratified Resampling, Systematic Resampling, and Residual Resampling. By comparing probability density variance, Root Mean Square Error (RMSE), and time complexity [12], Systematic Resampling is selected in this paper.

4) Update: Averaging the N sampled values to obtain the posteriori prediction value of particle filter at time k.

IV. Improved Particle Filter Algorithm

Particle filter is an effective solution for analyzing nonlinear dynamic system, and its accuracy can approximate the optimal estimation. However, the complexity of particle filter is higher than other filters and resampling does not fundamentally address the problem of weight degeneracy.

In addition, since the resampled particle set contains multiple repeating particles, resampling may result in loss of particle diversity, which is more severe in the environment with low noise. To solve the problem, regularized particle filter is proposed, which resamples from a continuous approximation of the posterior probability density function. However, this process requires calculation of the Epanechnikov kernel and is difficult to be applied to high-dimensional system.

In this paper, resampling is only performed when it is detected that the number of effective particles N_{eff} is below the pre-set threshold ξ. And in order to minimize the additional computational complexity that the proposed algorithm brings, we do not modify the resampling process of the particle filter, but add random perturbation after resampling to improve the particle diversity in the form of appropriate noise.

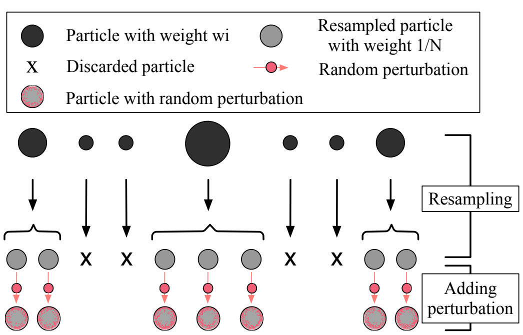

Fig. 2. Demonstration of Adding Perturbation

In the proposed algorithm, importance distribution $q(\cdot)$ in Eq.(8) is selected as

$$q(x_k^{(i)} \left| x_{k-1}^{(i)}, y_k \right.) = p(x_k^{(i)} \left| x_{k-1}^{(i)} \right.), \tag{15}$$

where $p(x_k^{(i)} \left| x_{k-1}^{(i)} \right.)$ can be obtained according to Eq.(1), thus Eq.(8) can be derived as

$$w_k^{(i)} \propto w_{k-1}^{(i)} p\left(y^{(i)} \left| x_k^{(i)} \right. \right). \tag{16}$$

After resampling, $w_{k-1}^{(i)} = \frac{1}{N}$, thus Eq.(8) can be further simplified as

$$w_k^{(i)} \propto p\left(y^{(i)} \left| x_k^{(i)} \right. \right), \tag{17}$$

Algorithm 2: Improved Resampling Algorithm

Input: N, T, δ, ξ, R, M_{3*T} (Measurements)
Output: $\mathcal{P}$ (Ultimate filtered result)
for i = *2, 3, ..., T* **do**
  ***Step 1*: Sampling and Calculating Weights**
  for k = *1:N* **do**
    $S_k = F \cdot S_{k-1} + u_{k-1}$;
    Convert S_k to χ_k (Spherical Coordinates);
    $\sigma_k = M_{:,i} - \chi_k$;
    $W_k = \frac{1}{\sqrt{2\pi |R|}} \cdot e^{-\frac{\sigma_k^T \cdot R' \cdot \sigma_k}{2}}$;
  end
  ***Step 2*: Weights Normalization**
  $W_{sum} = \sum_{k=1}^{N} W_k$;
  for k = *1:N* **do**
    $W_k = \frac{W_k}{W_{sum}}$;
  end
  ***Step 3*: Resampling and Adding Random Perturbation**
  Perform **Algorithm 1** and calculate P_k (Filtered result at time k);
  $P_{mean} = \frac{1}{N} \cdot \sum_{k=1}^{N} P_k$;
  if $N_{eff} \leq \xi$ **then**
    for k = *1:N* **do**
      $P'_k = P_{mean} \cdot (1 + \delta)$;
    end
    $P_{mean}^{(i)} = \frac{1}{N} \cdot \sum_{k=1}^{N} P'_k$;
  else
    $P_{mean}^{(i)} = \frac{1}{N} \cdot \sum_{k=1}^{N} P_k$;
  end
end
$\mathcal{P} = \left\{ P_{mean}^{(i)} \right\}, (i = 2 : N)$;
End of algorithm.

where $p\left(y^{(i)} \left| x_k^{(i)} \right. \right)$ denotes the probability that the measurement y appears under the condition that the state x occurs. According to Eq.(2), in the distribution of y, the mean is the true measurement and the variance equals noise variance. Therefore, assume that noise obeys Gaussian distribution, w_k can be obtained by

$$w_k = \frac{1}{\sqrt{2\pi |R|}} e^{-\frac{(y_k - y)^T R' (y_k - y)}{2}}, \tag{18}$$

where R is the covariance matrix of measurement noise u.

However, resampling may result in the loss of particle diversity, which is prone to serious error in filtering especially under environment with low noise. In order to maintain the diversity, state values of all particles need to be added with a random perturbation δ after resampling. Perturbation δ

combines resampling result of the present time k with the randomness of target motion in the following time $k+1$. That is, the values of all particles at present time k are set as the resampling result, and then each particle is added with independent and identically distributed noise.

In this way, the complexity of particle filter is reduced while ensuring the diversity of particles.

V. Numerical Results And Performance Evaluation

Since the resampled set contains multiple repeating particles, resampling may result in loss of particle diversity. To some extent, the random distribution of noise is beneficial to provide particle diversity during the filtering process. In the scenario of this paper, measurement noise of the UAV during flight is greatly affected by environmental factors.

In order to adapt to the scenario, and simultaneously test the performance of the proposed algorithm under different noise conditions, simulation is performed under normal noise and low noise separately.

A. Simulation under Normal Noise

As Fig.3 shows, under the measurement environment with normal noise, both the proposed algorithm and generic particle filter perform well.

TABLE I: Simulation Parameters

Symbol	Denotation	Value
N	Number of Particles	10000
T	Simulation Step Size	202
v	Process Noise	0.3%
u_{normal}	Normal Measurement Noise	1%
u_{low}	Low Measurement Noise	0.14%
ξ	Threshold of N_{eff}	0.5
δ	Random Perturbance	5%

When UAV is flying smoothly, the proposed algorithm performs as well as generic particle filter.

When UAV is maneuvering such as steering, both the filtered trajectories of the proposed algorithm and generic particle filter will be significantly shifted from the real trajectory. However, as Fig.3 demonstrates, the offset between real value and filtered value is smaller and decreases faster using the proposed algorithm than using generic particle filter. This demonstrates that when UAV is maneuvering, the random perturbation can ensure particle diversity after resampling, thus effectively improving the robustness of filtering performance.

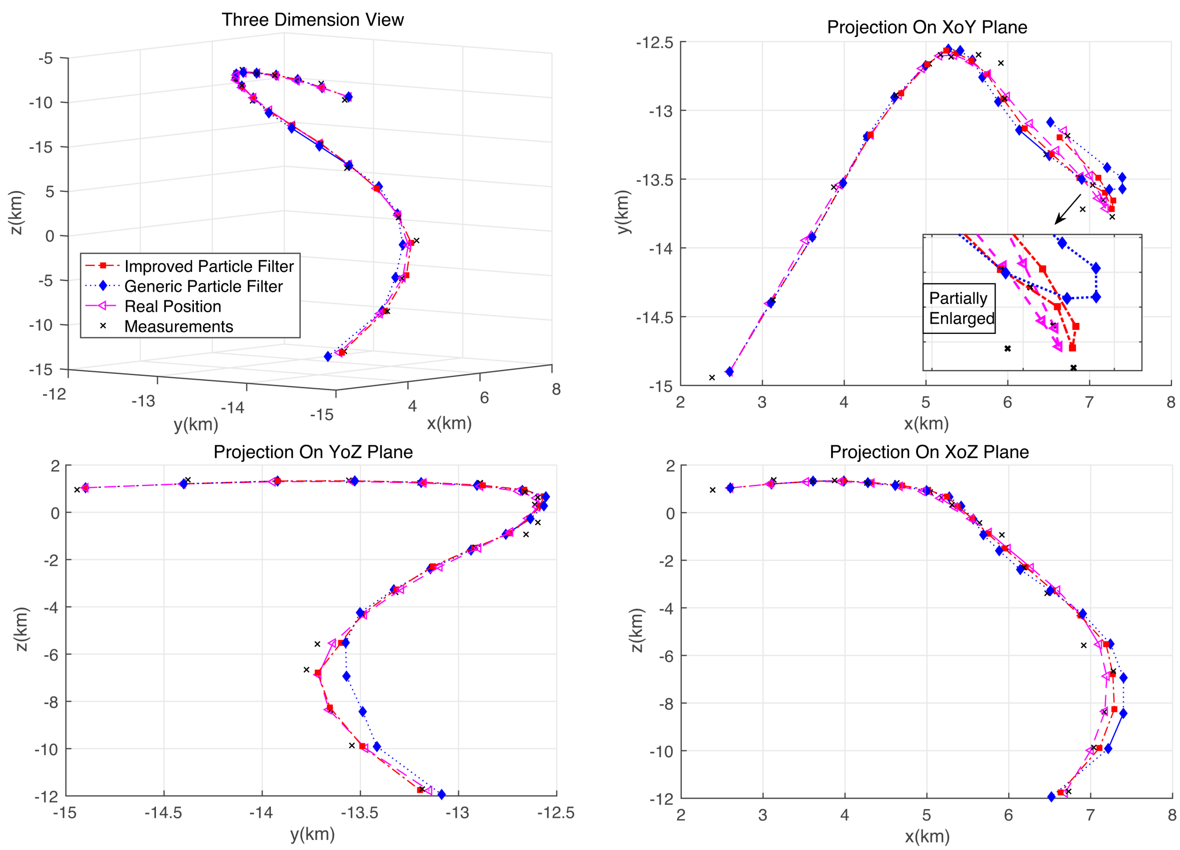

Fig. 3. Simulation under Normal Noise

B. Simulation under Low Noise

Smaller measurement noise means that the measurement is close to the real value. In the actual measurement, device cannot detect the noise magnitude, which causes that the accuracy of filtered results is not even comparable to the measurements. In the meantime, low noise cannot guarantee the diversity of particles during resampling, which probably causes the filtered value to deviate from the real value significantly.

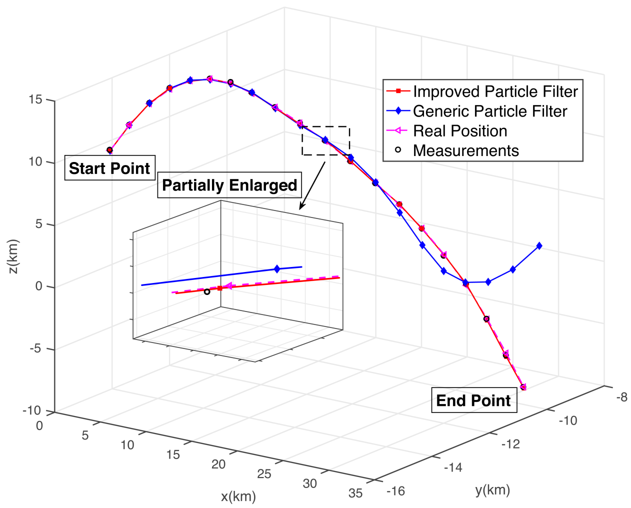

Fig. 4. Simulation under Low Noise

As Fig.4 demonstrates, filtered value of the proposed algorithm approximates the real value well. However, for lack of particle diversity, generic particle filter is prone to serious errors when measurement noise is low. Due to the particle diversity provided by the proposed algorithm, under the low-noise measurement environment, the proposed algorithm significantly outperforms generic particle filter.

C. Computational Complexity

In order to verify that the proposed algorithm brings no additional computational load, it runs in the same simulation environment with generic particle filter. As Fig.5 demonstrates, since the resampling process remains unchanged, the introduction of random perturbation does not significantly increase the computational complexity in different simulation steps.

VI. Conclusions

This work modeled the flight state equation of UAV in 3D scenario, which actually contains state information of 9 dimensions. In order to quickly and accurately track high-dimensional maneuvering target, an improved particle filter algorithm with random perturbation terms, which adapts to the noise magnitude and noise type under different measurement environments, is proposed. Since the calculation of random perturbation is not large and the resampling process is not modified, the proposed algorithm is consistent with generic particle filter in computational complexity. In the meantime, an effective trade-off between increasing the particle diversity and reducing the number of particles with smaller weights in the filtering process is realized. The effectiveness of the proposed algorithm has been witnessed in the simulation results.

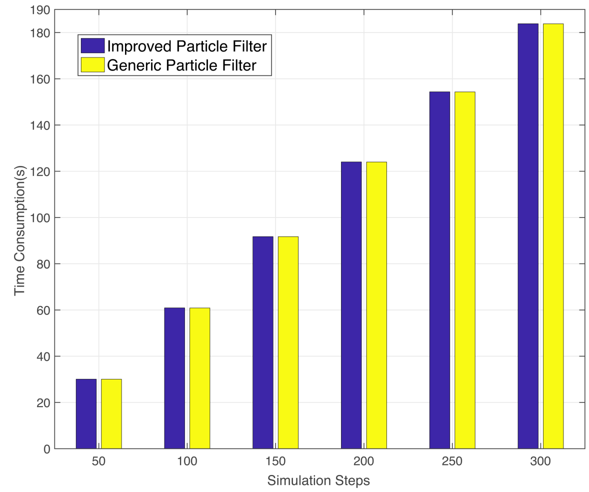

Fig. 5. Comparison of Computational Complexity

References

[1] S. Gezici, Z. Tian, G. B. Giannakis, H. Kobayashi, A. F. Molisch, H. V. Poor, and Z. Sahinoglu, "Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 70–84, July 2005.

[2] R. A. Singer, "Estimating optimal tracking filter performance for manned maneuvering targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-6, no. 4, pp. 473–483, July 1970.

[3] H. W. Sorenson, Ed., *Kalman Filtering: Theory and Application*. Piscataway, NJ: IEEE, 1985.

[4] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, March 2004.

[5] M. S. 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, no. 2, pp. 174–188, February 2002.

[6] R. Lamberti, Y. Petetin, F. Desbouvries, and F. Septier, "Independent resampling sequential monte carlo algorithms," *IEEE Transactions on Signal Processing*, vol. 65, no. 20, pp. 5318–5333, October 2017.

[7] A. Daniyan, Y. Gong, and S. Lambotharan, "An improved resampling approach for particle filters in tracking," in *2017 22nd International Conference on Digital Signal Processing (DSP)*, August 2017, pp. 1–5.

[8] N. Zheng, Y. Pan, X. Yan, and R. Huan, "Hierarchical resampling architecture for distributed particle filters," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2012, pp. 1565–1568.

[9] P. M. Djuric, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo, and J. Miguez, "Particle filtering," *IEEE Signal Processing Magazine*, vol. 20, no. 5, pp. 19–38, September 2003.

[10] D. Guo and X. Wang, "Quasi-monte carlo filtering in nonlinear dynamic systems," *IEEE Transactions on Signal Processing*, vol. 54, no. 6, pp. 2087–2098, June 2006.

[11] P. J. Smith, M. Shafi, and H. Gao, "Quick simulation: a review of importance sampling techniques in communications systems," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 4, pp. 597–613, May 1997.

[12] R. Douc and O. Cappe, "Comparison of resampling schemes for particle filtering," in *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005.*, September 2005, pp. 64–69.