SINS/GPS Integrated Navigation System using an Improved Particle Filter based on State Reconstruction

Longhua Ma*. Bo Guan**

** School of Aeronautics and Astronautics, Zhejiang University, Hangzhou, 310027 China (e-mail: lhma@iipc.zju.edu.cn).*
*** Department of Control Science and Engineering, Zhejiang University, Hangzhou, 310027 China (e-mail: 20932087@zju.edu.cn).*

Abstract: There exist nonlinear models in the integrated navigation system of strapdown inertial navigation system (SINS) and GPS. So it is appropriate to use particle filters to estimate the states. This paper focuses on the nonlinear problems when there exists large initial azimuth error in the SINS errors. In this paper, particles are driven to the regions of high probability by applying the error correction technique of Dynamic Matrix Control (DMC) to general particle filters and propose an improved particle filter. The proposed particle filter is then applied to the high-dimensional state model of SINS/GPS integrated navigation system. The simulation results show that the new algorithm doesn't need accurate error models of inertial measurement unit (IMU) but can still perform well and achieve more accurate estimates than unscented Kalman filter (KF).

Keywords: particle filter, state reconstruction, DMC, SINS/GPS.

1. INTRODUCTION

For SINS/GPS integrated system, GPS is a referenced navigation system and SINS is to be corrected for cumulative errors. It is the most popular way of integrated systems and often used to improve the navigation performance of SINS and GPS. However, the nonlinearity of SINS/GPS in some situations has a serious effect on the filtering performance of the system. So nonlinear filter algorithms have to be applied to the nonlinear models of SINS/GPS errors of which the state models may be nonlinear or the measurement models or both.

Particle filters can be applied to nonlinear models or non-Gaussian noise models. The more nonlinear model, or the more non-Gaussian noise, the more potential particle filters have (Gustafsson et al., 2002). To handle the nonlinear filter problems of SINS/GPS, particle filters have been introduced (Carvalho et al., 1997). But the dimension of the state space representing a complete set of SINS errors is quite high(≥15) and a direct application of particle filters shows that a vast number of particles are needed for particle filters to perform well (Nordlund & Gustafsson, 2001). That's because particle filters have the drawback of high computational complexity for high-dimensional system models and moreover, their filtering performance degrades quickly when the dimension increases. Paul Quang et al. (2010) have proved that the filtering error can grow exponentially with the dimension in the case of a linear tracking model. In high-dimensional systems, authors in (Bengtsson et al., 2008; Snyder et al., 2009) have proved that the particle size needs to essentially grow exponentially with the dimension to avoid weight degeneracy. So particle filters are not efficient when applied to high-dimensional systems (Daum et al., 2003).

In order to reduce the required particle size and avoid degeneracy for high-dimensional system models, the traditional method is to apply marginalization techniques (Nordlund & Gustafsson, 2001) to particle filters, which marginalize out the linear states to be estimated with Kalman filter. The combination of Kalman filter (KF) and particle filters is called Rao-Blackwised particle filters (RBPFs) (Doucet et al., 2000). RBPFs can reduce the computational complexity when there exists linear Gaussian state-space substructure in the system models. But if the system models don't have linear Gaussian parts, RBPFs are degenerated to general particle filters and become too computationally expensive. Authors in (Chorin et al., 2010; Leeuwen, 2010) exploit implicit sampling to remedy the high computational complexity, which firstly find regions of high probability and then look for particles that assume them and then guide them one by one towards the high probability regions. In this way the computational complexity can be reduced apparently. In this paper, particles are driven to the regions of high probability according to their observation errors and propose an improved particle filter by fusing general particle filters and the error correction technique of Dynamic Matrix Control (DMC). The fusion process is called state reconstruction modifying particles' states based on their observation errors. Due to the introduction of the error correction technique, the inertial-measurement-unit (IMU) errors can be regarded as the uncertainty errors of SINS and so the SINS errors can be modeled as a 9-dimension state model with Gaussian white noises. The computer simulation is performed to compare the improved particle filter with unscented Kalman filter (KF) (Julier & Uhlmann, 2004) in performance. The simulation results demonstrate that, compared with unscented KF, the improved algorithm can perform better and lead more accurate estimates.

The rest of the paper is organized as follows. Section II describes the system model of SINS/GPS when there exists large initial azimuth error. Then a general particle filter algorithm is presented in Section III. Section IV studies the error correction technique of DMC and proposes the improved particle filter algorithm based on state reconstruction. Section V presents the simulation results, and finally section VI concludes the paper.

2. SYSTEM MODEL FOR SINS/GPS INTEGRATED SYSTEM

In this paper, integrated SINS/GPS is studied through loosely coupled architecture. The SINS attitude, velocity and position are corrected by comparing estimates of velocity and position generated by GPS with estimates of the same quantities provided by SINS. The local level geographic frame ENU (East-North-Up) is chosen as the navigation frame (denoted as n). The axes of body frame (denoted as b) are lateral axis, longitudinal axis and vertical axis respectively. When there exists large initial azimuth error, the state model of SINS errors becomes nonlinear.

2.1 State Model

The state model consists of the SINS attitude error equation, velocity error equation and position error equation. When attitude errors are small, the attitude error equation can be approximated as linear time-varying equation with Gaussian noise approximations. But if attitude errors are too large to be approximated, the attitude error equation becomes nonlinear time-varying equation. In this paper, the simulation is aimed to investigate the performance of SINS/GPS with large initial azimuth error and small horizontal attitude errors. The state model of SINS is

$$\dot{\boldsymbol{X}} = \boldsymbol{F}\boldsymbol{X} + \boldsymbol{G}(\delta\boldsymbol{\varphi}) + \boldsymbol{\Gamma}\boldsymbol{W} \tag{1}$$

where $\boldsymbol{X}$ is given as

$$\boldsymbol{X} = \begin{pmatrix} \delta\boldsymbol{\varphi} \\ \delta\boldsymbol{v}^n \\ \delta\boldsymbol{p} \end{pmatrix} \tag{2}$$

where $\delta\boldsymbol{\varphi}$, $\delta\boldsymbol{v}^n$, $\delta\boldsymbol{p}$ are the vector of attitude errors, velocity errors and position errors respectively. $\boldsymbol{F}$ is given as

$$\boldsymbol{F} = \begin{pmatrix} 0_{3\times3} & \boldsymbol{M}_{\varphi v} & \boldsymbol{M}_{\varphi p} \\ 0_{3\times3} & \boldsymbol{M}_{vv} & \boldsymbol{M}_{vp} \\ 0_{3\times3} & \boldsymbol{M}_{pv} & \boldsymbol{M}_{pp} \end{pmatrix} \tag{3}$$

where the elements of $\boldsymbol{F}$ are shown as follows

$$\boldsymbol{M}_{\varphi v} = \begin{pmatrix} 0 & -\dfrac{1}{R+h} & 0 \\ \dfrac{1}{R+h} & 0 & 0 \\ \dfrac{\tan L}{R+h} & 0 & 0 \end{pmatrix} \tag{4}$$

$$\boldsymbol{M}_{\varphi p} = \begin{pmatrix} 0 & 0 & \dfrac{v_N}{(R+h)^2} \\ -\Omega\sin L & 0 & -\dfrac{v_E}{(R+h)^2} \\ \Omega\cos L + \dfrac{v_E\sec^2 L}{R+h} & 0 & -\dfrac{v_E\tan L}{(R+h)^2} \end{pmatrix} \tag{5}$$

$$\boldsymbol{M}_{vv} = (\boldsymbol{v}^n\times)\boldsymbol{M}_{\varphi v} - ((2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n)\times) \tag{6}$$

$$\boldsymbol{M}_{vp} = (\boldsymbol{v}^n\times)\begin{pmatrix} 0 & 0 & \dfrac{v_N}{(R+h)^2} \\ -2\Omega\sin L & 0 & -\dfrac{v_E}{(R+h)^2} \\ 2\Omega\cos L + \dfrac{v_E\sec^2 L}{R+h} & 0 & -\dfrac{v_E\tan L}{(R+h)^2} \end{pmatrix} \tag{7}$$

$$\boldsymbol{M}_{pv} = \begin{pmatrix} 0 & \dfrac{1}{R+h} & 0 \\ \dfrac{\sec L}{R+h} & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{8}$$

$$\boldsymbol{M}_{pp} = \begin{pmatrix} 0 & 0 & -\dfrac{v_N}{(R+h)^2} \\ \dfrac{v_E\sec L\tan L}{R+h} & 0 & -\dfrac{v_E\sec L}{(R+h)^2} \\ 0 & 0 & 0 \end{pmatrix} \tag{9}$$

where R, h, L, Ω are the radius of the earth, the local altitude, the local latitude and the earth rate respectively. $\boldsymbol{v}^n = (v_E, v_N, v_U)^T$ is the vector of velocities projected on the navigation frame. $\boldsymbol{\omega}_{ie}^n$ is the rotation rate of the earth frame with respect to the inertial frame projected on the navigation coordinate frame. $\boldsymbol{\omega}_{en}^n$ is the rotation rate of the navigation frame with respect to the earth frame projected on the navigation coordinate frame too. $\boldsymbol{G}(\delta\boldsymbol{\varphi})$ is given as

$$\boldsymbol{G}(\delta\boldsymbol{\varphi}) = \begin{pmatrix} g(\delta\boldsymbol{\varphi})\boldsymbol{\omega}_{in}^n \\ -g(\delta\boldsymbol{\varphi})\boldsymbol{f}^n \\ 0_{3\times1} \end{pmatrix} \tag{10}$$

where $\boldsymbol{\omega}_{in}^n$ is the rotation rate of the navigation frame with respect to the inertial frame projected on the navigation coordinate frame. $\boldsymbol{f}^n$ is the non-gravitational special force vector. $g(\delta\boldsymbol{\varphi})$ is written as

$$g(\delta\boldsymbol{\varphi}) = -\begin{pmatrix} \cos\delta\psi - 1 & \sin\delta\psi & -\delta\phi \\ -\sin\delta\psi & \cos\delta\psi - 1 & \delta\theta \\ \delta\theta\sin\delta\psi + \delta\phi\cos\delta\psi & \delta\phi\sin\delta\psi - \delta\theta\cos\delta\psi & 0 \end{pmatrix} \tag{11}$$

Here $\delta\boldsymbol{\varphi} = (\delta\theta, \delta\phi, \delta\psi)^T$. $\boldsymbol{\Gamma}$, $\boldsymbol{W}$ are written as

$$\boldsymbol{\Gamma} = \begin{pmatrix} -C_b^n & 0_{3\times3} \\ 0_{3\times3} & C_b^n \\ 0_{3\times3} & 0_{3\times3} \end{pmatrix} \quad (12)$$

$$\boldsymbol{W} = \begin{pmatrix} \delta\boldsymbol{\omega}_{ib}^b \\ \delta\boldsymbol{f}_{ib}^b \end{pmatrix} \quad (13)$$

where C_b^n is the direction cosine matrix from the body frame to the navigation frame. $\delta\boldsymbol{\omega}_{ib}^b$ is the vector of gyro errors and $\delta\boldsymbol{f}_{ib}^b$ is the vector of accelerometer errors.
The discretization of the continuous-time state model (1) then can be formulated by

$$\boldsymbol{X}_k = (I + T\boldsymbol{F}_{k-1})\boldsymbol{X}_{k-1} + T\boldsymbol{G}_{k-1} + \boldsymbol{\Gamma}_{k-1}\boldsymbol{W}_{k-1} \quad (14)$$

where T is the filtering time interval between $k-1$ and k. $\boldsymbol{W}_{k-1}$ are Gaussian white noises of the accelerometer and gyro errors in the form of discrete time. $\boldsymbol{\Gamma}_{k-1}$ and $\boldsymbol{G}_{k-1}$ are expressed as

$$\boldsymbol{\Gamma}_{k-1} = \boldsymbol{\Gamma}\cdot T \quad (15)$$

$$\boldsymbol{G}_{k-1} = \left[\boldsymbol{G}(\delta\boldsymbol{\varphi})\right]_{k-1} \quad (16)$$

2.2 Measurement Model

The measurement model describes the relations between the GPS position and velocity measurements and the system states. So the measurement can be achieved by subtracting the GPS position and velocity from the SINS position and velocity respectively. The measurement model is

$$\boldsymbol{Z}_k = \boldsymbol{H}_k\boldsymbol{X}_k + \boldsymbol{V}_k \quad (17)$$

where $\boldsymbol{V}_k$ is the vector of the measurement noises. $\boldsymbol{H}_k$ is given by

$$\boldsymbol{H}_k = \begin{pmatrix} 0_{3\times3} & I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & I_{3\times3} \end{pmatrix} \quad (18)$$

3. SEQUENTIAL IMPORTANCE RESAMPLE PARTICLE FILTER

Particle filter can solve nonlinear and non-Gaussian estimation problems. Considering the following nonlinear discrete-time model

$$\begin{cases} \boldsymbol{X}_k = f(\boldsymbol{X}_{k-1}, \boldsymbol{W}_{k-1}) \\ \boldsymbol{Y}_k = h(\boldsymbol{X}_k, \boldsymbol{V}_k) \end{cases} \quad (19)$$

where $\boldsymbol{X}_k$ denotes the state vector at k$^{\text{th}}$ time interval, $\boldsymbol{Y}_k$ the measurements. $\boldsymbol{W}_{k-1}$ are the process noises with known probability density function (PDF). $\boldsymbol{V}_k$ are the measurement noises which also have known PDF. They are assumed to be mutually independent. The SIR particle filter algorithm is given in the following algorithm (Gustafsson et al., 2002).

I. Initialization. Generate N particles according to the PDF $p(\boldsymbol{X}_0)$ of initial state which is assumed to be known.

II. Measurement update. Update the weights by the likelihood.

$$w_k^i = w_{k-1}^i p(\boldsymbol{Y}_k \mid \boldsymbol{X}_k^i) = w_{k-1}^i p_{V_k}(\boldsymbol{Y}_k - h(\boldsymbol{X}_k^i)), i = 1, \cdots, N$$

Here $p_{V_k}(\bullet)$ is the PDF of the measurement noises. Then normalize their weights

$$w_k^i = \frac{w_k^i}{\sum_{i=1}^N w_k^i}, i = 1, \cdots, N$$

III. Resampling. Take N particles with replacement from the set $\{\boldsymbol{X}_k^i\}_{i=1}^N$ and the probability to take the particle i is w_k^i.

IV. Prediction.

$$\boldsymbol{X}_{k+1}^i = f(\boldsymbol{X}_k^i, \boldsymbol{W}_k^i), i = 1, \cdots, N$$

V. Let $k := k+1$ and iterate to item 2.

4. AN IMPROVED PARTICLE FILTER BASED ON STATE RECONSTRUCTION

In order to reduce the number of particles and solve particle degeneracy, particles should be moved to high probability region as much as possible. In this paper, an improved particle filter is presented. Firstly, update particles' states and then reconstruct the states by state reconstruction before calculating weights. The principle of state reconstruction is error-correction, which is to update particles' states by increment inputs $\vartriangle\boldsymbol{U}$ based on new observation and desired observation.

System model which is nonlinear in the state equation and linear in the measurement equation considered in this paper is as follows

$$\begin{cases} \boldsymbol{X}_k = f\left(\boldsymbol{X}_{k-1}, \boldsymbol{W}_{k-1}\right) + \boldsymbol{B}\boldsymbol{U}_k \\ \boldsymbol{Y}_k = h\boldsymbol{X}_k + \boldsymbol{V}_k \end{cases} \quad (20)$$

where $\boldsymbol{X}_k \in \mathbb{R}^n$ is the state vector at k$^{\text{th}}$ time interval, $\boldsymbol{U} \in \mathbb{R}^m$ system inputs, $\boldsymbol{W}_{k-1} \in \mathbb{R}^q$ process noises, $\boldsymbol{Y}_k \in \mathbb{R}^p$ measurements and $\boldsymbol{V}_k \in \mathbb{R}^r$ measurement noises. Here the distributions of $\boldsymbol{W}_{k-1}$, $\boldsymbol{V}_k$ and $\boldsymbol{X}_0$ are assumed to be independent with known PDF.

Dynamic matrix control theory was proposed by Cutler and others (Cutler & Ramakar, 1980). It is based on a discrete time step response model that calculates a desired change of the manipulated variables, or the inputs, which minimizes the error between the desired outputs and their status in real-time. Referring to the theory of DMC, because the measurement equation is linear, after knowing the numerical transfer coefficients a_{ij} which represents the response of output $\boldsymbol{Y}_i$ ($i = 1:p$) of unit step input $\boldsymbol{U}_j$ ($j = 1:m$), the outputs can be predicted by linear superposition of the predicted response of outputs of any input increment. For example, if the input $\boldsymbol{U}_j$ has an increment $\vartriangle\boldsymbol{U}_j$ between $k-1$ and k, then the output $\tilde{\boldsymbol{Y}}_i(k)$ can be updated as follows

$$\hat{\boldsymbol{Y}}_i(k) = \tilde{\boldsymbol{Y}}_i(k) + a_{ij}\vartriangle\boldsymbol{U}_j(k) \quad (21)$$

where $\tilde{\boldsymbol{Y}}_i(k)$ denotes the output without any change to the input $\boldsymbol{U}_j$ from $k-1$ to k. Provided every input has an increment between $k-1$ and k, then (18) becomes

$$\hat{\boldsymbol{Y}}_i(k)=\tilde{\boldsymbol{Y}}_i(k)+\sum_{j=1}^{m} a_{ij}\vartriangle\boldsymbol{U}_j(k) \tag{22}$$

Equation (22) can be expanded for other outputs and then can be written in matrix form as

$$\hat{\boldsymbol{Y}}(k)=\tilde{\boldsymbol{Y}}(k)+\boldsymbol{A}\vartriangle\boldsymbol{U}(k) \tag{23}$$

where

$$\boldsymbol{A}=\begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{p1} & \cdots & a_{pm} \end{pmatrix} \tag{24}$$

$$\vartriangle\boldsymbol{U}(k)=\begin{pmatrix} \vartriangle\boldsymbol{U}_1(k) \\ \vdots \\ \vartriangle\boldsymbol{U}_m(k) \end{pmatrix} \tag{25}$$

In order to avoid dramatic changes in the input increment $\vartriangle\boldsymbol{U}(k)$, soft constraints are set to the size of the manipulated inputs. Then the optimal quadratic performance index is obtained as

$$\min J(k)=\left(\boldsymbol{Y}_k-\hat{\boldsymbol{Y}}(k)\right)^T Q\left(\boldsymbol{Y}_k-\hat{\boldsymbol{Y}}(k)\right)+\vartriangle\boldsymbol{U}(k)^T R\vartriangle\boldsymbol{U}(k) \tag{26}$$

where $\boldsymbol{Y}_{k+1}$ are the desired outputs at kth time interval. Q is called error weight matrix and R is called control weight matrix. The former is to depress the tracking errors and the latter is to depress the size of the manipulated inputs' increment $\vartriangle\boldsymbol{U}(k)$. Q and R are both non-negative matrix and usually taken as diagonal matrix to simplify the modulation process.

Having equated the derivative of $J(k)$ given by (26) to 0, the optimal control increments is given as

$$\vartriangle\boldsymbol{U}(k)=\left(\boldsymbol{A}^T Q\boldsymbol{A}+R\right)^{-1}\boldsymbol{A}^T Q\left(\boldsymbol{Y}_k-\hat{\boldsymbol{Y}}(k)\right) \tag{27}$$

Put $\vartriangle\boldsymbol{U}(k)$ back into the system model (20), then the corrected states at kth time interval can be written as

$$\boldsymbol{X}_k=\hat{\boldsymbol{X}}_k+\boldsymbol{B}\vartriangle\boldsymbol{U}(k) \tag{28}$$

If the system model has no control inputs $\boldsymbol{U}$ and control matrix $\boldsymbol{B}$, the paper presents that state reconstruction can be implemented according to the observability of state variables (Goshen-Meskin & Bar-Itzhack, 1990).

(1) For states of high observability, the steps of state reconstruction are as above. The dimension of $\vartriangle\boldsymbol{U}(k)$ is equal to the number of states of high observability.

(2) For states of low observability relatively, state reconstruction can be realized by adding control matrix to the system model. For example, if the former m states of $\boldsymbol{X}_k$ all have relatively high observability and $\boldsymbol{X}_{j,k}$ ($j>m$) is one of the remaining states, which has lower observability or has no observability. The control matrix can be set as

$$\boldsymbol{B}=\begin{pmatrix} \boldsymbol{I}_{m\times m} \\ \vdots \\ c_{j,1\times m} \\ \boldsymbol{O} \end{pmatrix} \tag{29}$$

$$c_{j,1\times m}=\begin{pmatrix} c_{j,1} & \cdots & c_{j,m} \end{pmatrix} \tag{30}$$

$$c_{j,i}=\frac{\partial\boldsymbol{X}_{j,k}}{\partial\boldsymbol{X}_{i,k}}(i=1,\cdots,m) \tag{31}$$

where $\boldsymbol{I}$ is an identity matrix and $\boldsymbol{O}$ is a null matrix.

The improved particle filter can be obtained by combining the state reconstruction described as above with SIR particle filter (Gustafsson et al., 2002). The new particle filter algorithm is written as follows.

I. Initialization. Generate N particles according to the PDF $p(\boldsymbol{X}_0)$ of initial state which is assumed to be known. The weight of each particle is assigned $1/N$.
II. State reconstruction. Firstly, update the states of particles and then implement the steps of state reconstruction as above. If there is no input in system model, state reconstruction can be realized according to (29), (30) and (31).
III. Measurement update. Update the weights by the likelihood.
$$w_k^i=w_{k-1}^i p(\boldsymbol{Y}_k\mid\boldsymbol{X}_k^i)=w_{k-1}^i p_{V_k}(\boldsymbol{Y}_k-h(\boldsymbol{X}_k^i)),\ i=1,\cdots,N$$
Here $p_{V_k}(\bullet)$ is the PDF of the measurement noises. Then normalize their weights
$$w_k^i=\frac{w_k^i}{\sum_{i=1}^{N} w_k^i},\ i=1,\cdots,N$$
IV. Resampling. Take N particles with replacement from the set $\{\boldsymbol{X}_k^i\}_{i=1}^N$ and the probability to take the particle i is w_k^i.
V. Prediction.
$$\boldsymbol{X}_{k+1}^i=f(\boldsymbol{X}_k^i,\boldsymbol{W}_k^i)+\boldsymbol{B}\boldsymbol{U}_k,\ i=1,\cdots,N$$
VI. Let $k:=k+1$ and iterate to item 2.

5. SIMULATION RESULTS

In order to evaluate the filtering performance of the proposed particle filter algorithm in SINS/GPS, simulation test is carried out based on the INS toolbox of MATLAB in the laboratory situation. The initial velocity is assumed 0 in all axes. The object accelerates along straight line at 10 m/s² for ten seconds followed by climbing, horizontal 90° turning, straight cruising, climbing, horizontal 90° turning and straight cruising without deceleration. The flight trajectory and velocity profiles are shown in the Fig. 1 and Fig. 2 respectively. In the flight trajectory, climbing is accompanied by the pitching angle change and horizontal turning by the rolling angle change and the azimuth angle change. So the flight trajectory involves all attitude angle change process.

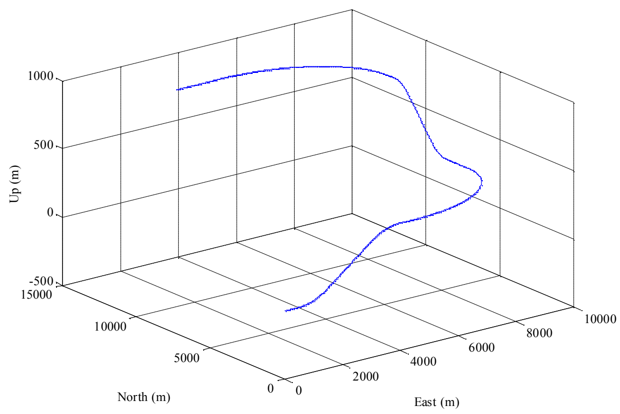

Fig. 1. Flight trajectory

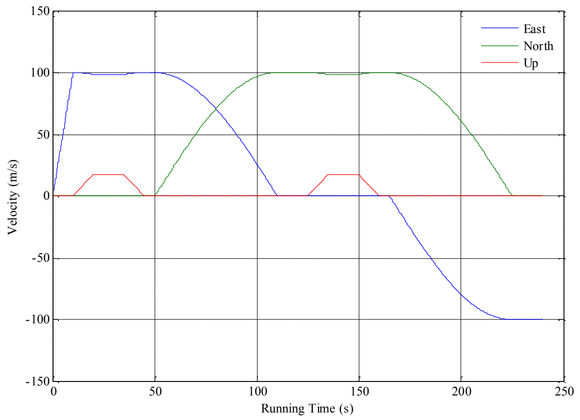

Fig. 2. Velocity profiles

For the GPS data generation, position accuracy is assumed to be 10 m and velocity accuracy is 1 m/s in all axes. It is also assumed that IMU in all axes has the same specifications. The tactical-grade sensors (Pusa, 2009) for IMU are presented as follows (without considering scale factor errors)

$$\nabla = 100\mu\text{g}, \quad w_a \sim N(0, 10\mu\text{g})$$
$$\varepsilon = 0.1^\circ/\text{h}, \quad w_g \sim N(0, 0.01^\circ/\text{h})$$

According to (1), $\boldsymbol{W}$ denotes IMU errors, including random bias, nonlinearity error, scale factor error, misalignment error and random white noise and so on. The new algorithm allows the IMU errors viewed as the uncertainty errors of SINS. So there is no need to extend the dimension of the system model and $\boldsymbol{W}$ is approximated to Gaussian white noise. The new proposed algorithm allows reducing the dimension of the state model and so can reduce the number of required particles to perform well.

Because the state model (1) has no any manipulated input, it should calculate control matrix according to the state observability of the states for the proposed algorithm. Velocity errors and position errors are fully observable states because they have external measurement information provided by GPS. No matter what kind of movement, they all have high observability relatively. The azimuth attitude angle error has much lower observability than the level attitude angle errors, which even have the same observability as the velocity errors and position errors (Li et al., 2010). So the paper choose the velocity errors, position errors and the level attitude angle errors as the states of high observability and so the dimension of $\Delta\boldsymbol{U}(k)$ is 8. In the simulation, the error weight matrix and the control weight matrix are chosen as

$$Q = \boldsymbol{I}_{6\times6} \tag{32}$$

$$R = 10\boldsymbol{I}_{8\times8} \tag{33}$$

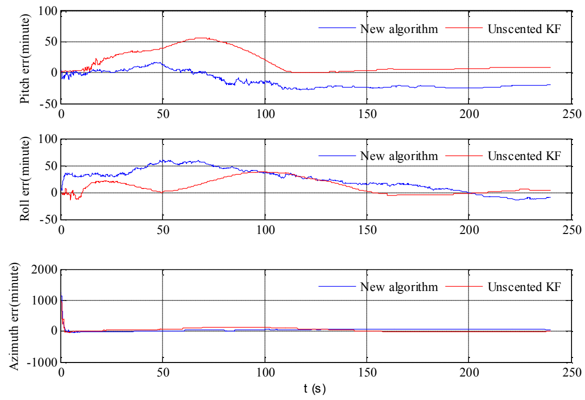

Fig. 3. Attitude errors

Throughout the simulation, the following initial errors are assumed: initial attitude angle errors [0 0 20°], initial velocity errors 1 m/s in all axes, initial position errors 10 m in all axes. Unscented KF (Julier & Uhlmann, 2004) is used to compare with the new algorithm in the simulation. In this paper, 100 particles are utilized for the improved particle filter. The attitude errors, velocity errors and position errors are shown in Fig. 3, Fig. 4 and Fig. 5 respectively. It can be seen from Fig. 3 that the converged roll angle error are around 0 and the azimuth angle error rapidly converges to 0 although there exists large error in the transient response of the proposed algorithm, which proves that the proposed algorithm has good convergence performance for the roll error and rapid convergence for the azimuth error. The pitch angle error converges within 110 seconds and the level is about 22 minute of arc. So the algorithm has almost the same convergence ability as Unscented KF for the roll error and azimuth error, but worse for the pitch angle error. From Fig. 4 and Fig. 5, the velocity errors and position errors are rather small in the whole flight trajectory, which clearly demonstrates that the new algorithm has rather good convergence and stability performance for the velocity errors and position errors. But for unscented KF, the convergence and stability performance is not as good as the proposed algorithm because of the larger errors during the whole flight trajectory. The root mean square error (RMSE) of the velocity errors and position errors for the new algorithm are shown in Table 1 and Table 2 respectively. It apparently shows that the new algorithm has much better performance for SINS/GPS than GPS–only accuracy.

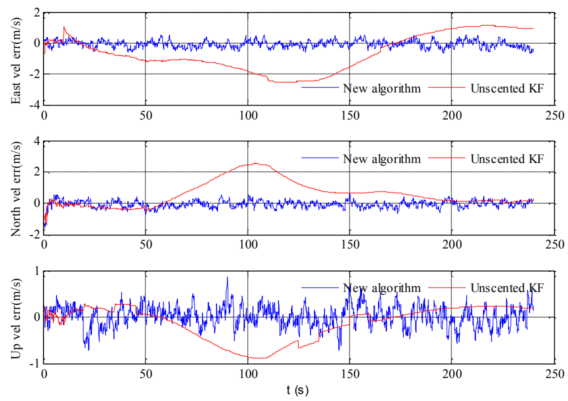

Fig. 4. Velocity errors

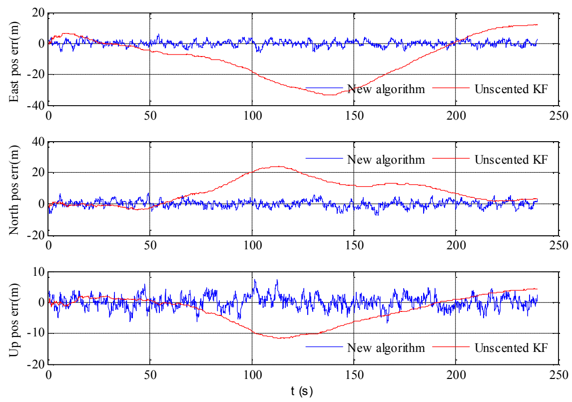

Fig. 5. Position errors

Table 1. Velocity errors RMSE

Velocity Direction	RMSE
East velocity error	0.2259
North velocity error	0.2293
Up velocity error	0.2250

Table 2. Position errors RMSE

Position Direction	RMSE
East position error	1.8137
North position error	2.1560
Up position error	2.0752

6. CONCLUSIONS

The paper proposes an improved particle filter for high-dimensional system models by fusing general particle filters and the error correction technique of DMC, which can drive particles to the regions of high probability by modifying particles' states based on their observation errors. If system models haven't manipulated inputs, it is presented that control matrices can be determined by the size of observability of state variables to perform state reconstruction process.

The proposed improved particle filters provides better convergence and stability performance than unscented KF. Moreover, the algorithm doesn't depend on accurate models of accelerator errors and gyro errors and so can reduce the dimension of the system models. Though the simulation results can demonstrate that the new algorithm is a useful filter algorithm for SINS/GPS, it still has limitations on state models of which the measurement models should be linear models. A further analysis on the theoretical convergence and the reduction of computational complexity due to DMC involved will be investigated in the future work.

ACKNOWLEDGEMENT

This work is supported partially sponsored by the National Natural Science Foundation of China under Grant No.61070003 and the Natrual Science Foundation of Zhejiang Province of China under Grant No.R1090052.

REFERENCES

Bengtsson, T., Bickel, P., and Li B. (2008). Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems. *Probability and Statistics: Essays in Honor of David A. Freedman*, 2, 316–334.

Carvalho, H., Del Moral, P., Monin, A., Salut, G. (1997). Optimal Nonlinear Filtering in GPS/INS Integration. *IEEE Transactions on Aerospace and Electronic Systems*, 33(3), 835-850.

Chorin, A. J., Morzfeld, M. and Tu, X. (2010). Implicit particle filters for data assimilation. *Comm. Appl. Math. Comp.*, 5(2), 221-240.

Cutler, C.R, Ramakar, B.L. (1980). Dynamic Matrix Control – a Computer Control Algorithm. *Proceedings of the Joint Automatic Control Conference*, San Francisco, USA.

Daum, F., Co, R. and Huang J. (2003). Curse of Dimensionality and Particle Filters. *Proceedings of the IEEE Aerospace Conference*, 4, 1979-1986.

Doucet, A., de Freitas, N., Murphy, K., Russell, R. (2000). Rao-blackwellised Particle Filtering for Dynamic Baysian Networks. *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, Stanford, California, USA, 176-183.

Goshen-Meskin, D., Bar-Itzhack, I. Y. (1990). Observability Analysis of Piece-Wise Constant Systems with Application to Inertial Navigation. *Proceedings of the 29th IEEE Conference on Decision and Control*, Honolulu, Hawaii, USA, 821-826.

Gustafsson, F., Gunnarsson, F., Bergman, N., et al. (2002). Particle Filters for Positioning, Navigation and Tracking. *IEEE Transactions on Signal Processing*, 50(2), 425-437.

Julier, S., Uhlmann, J.K. (2004). Unscented Filtering and Nonlinear Estimation. *Proceedings of the IEEE*, 92(3), 401-422.

Leeuwen, P. van. (2010). Nonlinear data assimilation in geosciences: an extremely efficient particle filter. *Quarterly Journal of the Royal Meteorological Society*, 136, 1991–1999.

Li, Y., Li, Y., Rizos, C., Xu, X. (2010). Observability Analysis of SINS/GPS during In-motion Alignment using Singular Value Decomposition. *Proceedings of IEEE International Conference on Electrical Engineering and Automatic Control*, Zibo, Shandong, China, 26-28.

Nordlund, P-J., Gustafsson, F. (2001). Sequential Monte Carlo Filtering Techniques Applied to Integrated Navigation Systems. *Proceedings of the American Control Conference*, Arlington, Virginia, U.S.A.

Quang, P., Musso, C., Le Gland, F. (2010). An insight into the Issue of Dimensionality in Particle Filtering. *Proceedings of the 13th International Conference on Information Fusion*, UK.

Pusa, J. (2009). Strapdown Inertial Navigation System Aiding with Nonholonomic Constraints using Indirect Kalman Filtering. Tampere, Finland: Tampere University of Technology.

Snyder, C., Bengtsson, T., Bickel, P. and Anderson J. (2008). Obstacles to high-dimensional particle filtering. *Monthly Weather Review*, 136(12), 4629–4640.