Spacecraft Autonomous GPS Navigation based on Polytopic Linear Differential Inclusion

Zhen Chen, Jialin Li and Xiangdong Liu
(School of Automation, Key Laboratory for Intelligent Control & Decision of Complex System, Beijing Institute of Technology, Beijing 100081, China)
(E-mail: lijialin_bit@163.com)

Aiming at improving the poor real-time performance of existing nonlinear filtering algorithms applied to spacecraft autonomous navigation based on Global Positioning System (GPS) measurements and simplifying the algorithm design of navigation algorithms, a spacecraft autonomous navigation algorithm based on polytopic linear differential inclusion is proposed in this paper. Firstly, it is demonstrated that the nonlinear estimation error system of spacecraft autonomous navigation can be modelled as a polytopic linear differential inclusion system model according to the idea of global linearization. Thus, the filtering of a nonlinear system simplified to a filtering of a polytopic linear system with coefficients. Secondly, Tensor-Product (TP) model transformation is applied to determine the polytopic linear differential inclusion system model. The model error introduced by global linearization is reduced and the compromise between computational complexity and modelling accuracy is realised. Finally, a spacecraft autonomous navigation algorithm based on polytopic linear differential inclusion is designed by combining multi-model Kalman filtering with data fusion. Compared with an Extended Kalman Filter (EKF), the proposed algorithm is simpler and easier to implement since it need not update the Jacobian matrices online. Simulation results demonstrate the same estimation accuracy of the proposed algorithm to that of EKF.

KEYWORDS

Submitted: 15 September 2014. Accepted: 8 October 2014. First published online: 20 November 2014.

1. INTRODUCTION. Autonomous navigation technology enables a spacecraft to determine its position and velocity by itself, without need for ground tracking. This technology can not only improve the reliability of a space mission, but also reduce operational complexity while working.

The capability of spacecraft autonomous navigation relies mainly on its navigation method and navigation algorithm. A variety of autonomous navigation methods have
been proposed so far: photoelectric autonomous navigation (fully autonomous navigation), including horizon-sensing (Ning and Fang, 2007), magnetometer-based navigation (Shorshi and Bar-Itzhack, 1995), star sensor (Ning and Fang, 2008) and inertial navigation (Xu and Fang, 2008); inter-satellite link (semi-autonomous navigation), including Doppler measurement (Ning and Fang, 2008) and Global Positioning System (Choi et al., 2010; Chiaradia et al., 2003; Yoon et al., 2000; Bolandi et al., 2013) and so on. Unlike other autonomous navigation methods, GPS has the following advantages: (1) global coverage; (2) low cost; (3) able to determine positions with centimetre level accuracy and measure velocities with high precision by Differential GPS (DGPS) (Chiaradia et al., 2003). Therefore, GPS is widely applied to spacecraft autonomous navigation.

A nonlinear filtering algorithm is employed for autonomous navigation. The filtering algorithms applied to spacecraft autonomous navigation can be divided into three classes according to the handling methods of nonlinear problems. The first is the approximation analytic method, which approximates the nonlinear measurement and transition equations with the Taylor series expansion, e.g. Extended Kalman Filter (EKF). The second is the deterministic sampling method, which approximates the underlying density functions of state-vectors by producing a variety of sigma points according to a rule, e.g. Unscented Kalman Filter (UKF). The last is a method based on the Monte Carlo simulation, which approximates the posterior density functions on the basis of the law of large numbers, e.g. Particle Filter (PF). EKF is widely used in spacecraft autonomous navigation (Shorshi and Bar-Itzhack, 1995; Yim et al., 2000; Zhang et al., 2009). Compared with other nonlinear filtering algorithms, EKF has an algorithm design of low complexity and its estimation accuracy satisfies the requirements of practical applications. However, EKF needs to linearize the measurement and transition equations in real time, which leads to a large online calculation quantity. It will also introduce large approximation model errors that may lead to poor filtering performance or sometimes divergence of the filter in a highly nonlinear and non-Gaussian system.

In order to improve the estimation accuracy and considering the lower approximation model errors than EKF, the application of UKF in spacecraft autonomous navigation has also received great attention (Ali and Fang, 2006; Cheng et al., 2014; Liu et al., 2011). UKF uses the actual nonlinear model and a set of sigma sample points produced by the unscented transformation to capture the mean and covariance of states (Julier and Uhlmann, 2004). This has proved to have a better performance than EKF in nonlinear system state estimation. Nevertheless, UKF has not only a complicated algorithm design and large online calculation quantity, but also serious influence on the preference in system state estimation, caused by the uncertainty of the true distribution (Song and Han, 2008). The poor stability of UKF limits its application in spacecraft autonomous navigation to some extent. Compared with UKF, PF has not only a high-precision state estimation, but also stable performance with an uncertain distribution. PF approximates the posterior density function of state vector by obtaining a random sample. It achieves the optimal estimation by adjusting the weights of samples and changing samples based on measured data (Song et al., 2009). However, the large online calculation quantity caused by vast samples and the complicated algorithm design also limit the application of PF in spacecraft autonomous navigation, in spite of the above advantages. A lot of modified nonlinear filtering algorithms based on EKF, UKF and
PF have appeared in recent years (Ning and Fang, 2008; Li et al., 2012; Yang et al., 2014). In order to improve the performance of UKF in case of uncertain noise statistic characteristics, a noise statistic estimator was designed to estimate uncertain noise characteristics online for UKF by Cheng et al. (2014). Moreover, the single algorithms can be integrated to enhance their estimation performance. Ning and Fang (2008) proposed the unscented particle filter by combining UKF with PF to enhance the stability of UKF, while according to the research of Yang et al. (2014), by designing a switch-mode information fusion filter, the iterated square root unscented filter was combined with EKF to strengthen the reliability of autonomous navigation. However, it will be more and more difficult for the increasingly complex algorithm to provide real-time navigation, especially for modern spacecraft with limited computing resources.

Aiming at improving the poor real-time performance of the nonlinear filtering algorithms applied to spacecraft autonomous navigation, an effective spacecraft autonomous navigation algorithm based on Polytopic Linear Differential Inclusion (PLDI) is proposed in this paper. Firstly it is demonstrated that the estimation error system of spacecraft autonomous navigation can be modelled as a Polytopic Linear Differential Inclusion Systems (PLDI) model and the problem of the nonlinear filtering is converted to the problem of multiple linear time-invariant systems in the form of convex combination on linear filtering. Then the PLDI model of spacecraft autonomous navigation is determined by TP model transformation. And on this basis the spacecraft autonomous navigation algorithm based on PLDI is proposed by combining multi-model Kalman filtering with data fusion. This algorithm has better performance in state estimation and lower complexity of algorithm design than EKF.

This paper is structured as follows: after this introduction, the orbit dynamic model and GPS measurement model are introduced in Section 2. In Section 3, it is demonstrated that the estimation error system of spacecraft autonomous navigation can be modelled as a PLDI model. Meanwhile, the methods for determining the PLDI model and the spacecraft autonomous navigation algorithm based on PLDI
are given. The navigation algorithm is verified by using simulation in Section 4 and the simulation results of the algorithm will be compared with that of EKF. Conclusions are drawn in Section 5.

2. SPACECRAFT AUTONOMOUS GPS NAVIGATION MODEL

2.1. Orbit dynamic model. WGS84 is selected as the reference coordinate system, and we suppose that the high order terms of disturbance caused by the Earth, Sun, Moon or atmosphere are ignored. The equations of this model are given below (Ning and Fang, 2007):

\[
\begin{align*}
\frac{dx}{dt} &= v_x + w_x \\
\frac{dy}{dt} &= v_y + w_y \\
\frac{dz}{dt} &= v_z + w_z \\
\frac{dv_x}{dt} &= -\mu \frac{x}{r^3} \left[ 1 - J_2 \left( \frac{R_e}{r} \right) \left( 7.5 \frac{z^2}{r^2} - 1.5 \right) \right] + a_x + w_{vx} \\
\frac{dv_y}{dt} &= -\mu \frac{y}{r^3} \left[ 1 - J_2 \left( \frac{R_e}{r} \right) \left( 7.5 \frac{z^2}{r^2} - 1.5 \right) \right] + a_y + w_{vy} \\
\frac{dv_z}{dt} &= -\mu \frac{z}{r^3} \left[ 1 - J_2 \left( \frac{R_e}{r} \right) \left( 7.5 \frac{z^2}{r^2} - 4.5 \right) \right] + a_z + w_{vz} \\
\frac{da_x}{dt} &= -\alpha_x a_x + w_{ax} \\
\frac{da_y}{dt} &= -\alpha_y a_y + w_{ay} \\
\frac{da_z}{dt} &= -\alpha_z a_z + w_{az}
\end{align*}
\]

where \( r \) represents \( \sqrt{x^2 + y^2 + z^2} \), \( x, v_x, y, v_y, z, v_z \) are satellite positions and velocities on the three axes respectively; \( a_x, a_y, a_z \) are part of the perturbing accelerations on the three axes respectively which can be approximated by the Singer model; \( \alpha_x, \alpha_y, \alpha_z \) are reciprocals of the correlation time constants on the three axes respectively; \( R_e \) is the earth radius; \( \mu \) is the gravitational constant of the earth; \( J_2 \) is the second zonal coefficient.

2.2. Measurement model of GPS. The theory of satellite navigation is illustrated by Figure 1.

The pseudo-range between spacecraft receiver and navigation satellite is regarded as the observation of the measurement equations. To determine unknown position vectors of the target spacecraft in space, the position information of three navigation satellites needs to be obtained. Considering that the clock bias can produce a big observation error that can neither be ignored nor be predicted, the clock bias is regarded as an unknown quantity that needs to be determined. As a result, the information of a fourth navigation satellite is needed. Now, the position information of four navigation satellites need to be obtained to determine the position vectors of the target spacecraft and clock bias of receiver using the least squares method. The number of observable satellites is assumed to be four, and the measurement
equations are given as follows:

\[
\begin{align*}
\rho_1 &= \sqrt{(x^1 - x)^2 + (y^1 - y)^2 + (z^1 - z)^2} + c\delta t_u + \epsilon_1 \\
\rho_2 &= \sqrt{(x^2 - x)^2 + (y^2 - y)^2 + (z^2 - z)^2} + c\delta t_u + \epsilon_2 \\
\rho_3 &= \sqrt{(x^3 - x)^2 + (y^3 - y)^2 + (z^3 - z)^2} + c\delta t_u + \epsilon_3 \\
\rho_4 &= \sqrt{(x^4 - x)^2 + (y^4 - y)^2 + (z^4 - z)^2} + c\delta t_u + \epsilon_4
\end{align*}
\]

(2)

where \([x^i, y^i, z^i]^T (i = 1, 2, 3, 4), [x, y, z]^T\) denote the position of observable satellites and position of spacecraft receiver in the WGS84 coordinate system respectively; \(c\) denotes the velocity of light; \(\delta t_u\) denotes the GPS clock bias; \(\rho_i (i = 1, 2, 3, 4)\) denotes the pseudo-range between spacecraft receiver and observable satellites; \(\epsilon_i\) is the measurement noise and assumed to be zero-mean white noise.

In practice, \(\delta f_u\) as the clock bias drift of receiver, and \(\delta t_u\) is caused by the clock bias drift with the following equation:

\[
\dot{\delta t}_u = \delta f_u
\]

(3)

In order to simplify the algorithm design, a spacecraft autonomous navigation algorithm design in case the dynamic and measurement models have high nonlinearities. However, updating the Jacobian matrices may lead to not only poor real-time performance, but also higher complexity of the algorithm design in case the dynamic and measurement models have high nonlinearities. In order to simplify the algorithm design, a spacecraft autonomous navigation algorithm based on PLDI (PLDI Kalman Filter (PKF)) is proposed in this Section.

Define \(\delta f_u\) as the clock bias drift of receiver, and \(\delta t_u\) is caused by the clock bias drift with the following equation:

\[
\dot{\delta f}_u = -\lambda_s \delta f_u + s
\]

(4)

where \(\lambda_s\) denotes the reciprocal of the correlation time constants for the model, \(s\) denotes the zero-mean Gaussian noise.

2.3. Autonomous navigation system model based on GPS measurement. \(X = [x, v_x, a_x, y, v_y, a_y, z, v_z, a_z, \delta t_u, \delta f_u]^T\) is regarded as the state variable of an autonomous navigation system, where \(\delta t_u\) and \(\delta f_u\) denote the clock bias and clock bias drift of GPS receiver respectively. According to Equations (1) to (4), the autonomous navigation model based on GPS measurement can be given by:

\[
\begin{align*}
\dot{X} &= f(X) + w \\
Z &= h(X) + \epsilon
\end{align*}
\]

(5)

where \(X \in \mathbb{R}^n\) and \(Z \in \mathbb{R}^m\) denote the state variable and observation which is comprised of the pseudo-range between target spacecraft and GPS navigation satellites; \(f(\cdot)\) and \(h(\cdot)\) denote the general equation of Equation (1) and Equation (2) respectively; \(w = (w_x, w_{vx}, w_{ax}, w_y, w_{vy}, w_{ay}, w_z, w_{az}, 0, w_t)^T, \epsilon = (\epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4)^T\) denotes the system noise and measurement noise respectively.

3. Spacecraft Autonomous Navigation Algorithm Based on PLDI. EKF needs to linearize the nonlinear system model in real time and update the Jacobian matrices online. However, updating the Jacobian matrices may lead to not only poor real-time performance, but also higher complexity of the algorithm design in case the dynamic and measurement models have high nonlinearities. In order to simplify the algorithm design, a spacecraft autonomous navigation algorithm based on PLDI (PLDI Kalman Filter (PKF)) is proposed in this Section.

3.1. PLDIs model of autonomous navigation estimation error system. \(\hat{X}\) is assumed to be the optimal system state estimation. The system state and observed
estimation satisfy:
\[
\dot{\hat{X}} = f(\hat{X}) \\
\hat{Z} = h(\hat{X})
\] (6)

where \(\hat{Z}\) denotes the observed estimation of measurement model.

The state estimation error and observed residual can be defined as:
\[
\Delta X = X - \hat{X} \\
\Delta Z = Z - \hat{Z}
\]

Therefore, the estimation error model of autonomous navigation system can be given by:
\[
\Delta \dot{X} = f(X) - f(\hat{X}) + w \\
\Delta Z = h(X) - h(\hat{X}) + \varepsilon
\] (7)

**Theorem 1:** \(\Omega_x\) is a subset of n-dimensional space, \(\Omega_x \subseteq \mathbb{R}^n\), and the vector function \(f\) is defined as: \(\Omega_x \rightarrow \mathbb{R}^m\). If \(\Omega_x\) and \(f\) satisfy the following conditions:

1. \(f\) is continuously differentiable;
2. The arbitrary point \(X\) and vector \(H\) at the subset \(\Omega_x\) satisfy: \(X + aH \in \Omega_x\);
3. Partial derivatives of \(f\) vary in a bounded domain belonging to \(\mathbb{R}^{m \times n}\);

Then,
\[
f(X + H) - f(X) \in \text{Co}(\Omega_y)H
\] (8)

where \(0 \leq a \leq 1\), \(H \in \mathbb{R}^n\), \(\text{Co}(\Omega_y)\) is the convex hull generated by \(\Omega_y\).

**Proof:** The vector function \(f: \Omega_x \rightarrow \mathbb{R}^m\) is continuously differentiable. The arbitrary point \(X\) and \(X + H \in \Omega_x\) satisfy \(X + aH \in \Omega_x\), thus according to differential mean value theorem of vector function (de Boor, 2005) we have:
\[
f (X + H) - f (X) = \left[ \int_0^1 \nabla f (X + aH) da \right] H
\] (9)

where \(\nabla f\) is the partial derivative of \(f\), namely, the Jacobian matrix of \(f\). Therefore,
\[
f_i(X + H) - f_i(X) = \int_0^1 \sum_{j=1}^n \frac{\partial f_i}{\partial x_j}(X + aH)h_j \, da
\] (10)

where \(i = 1, 2 \cdots m\). Because partial derivatives of \(f\) vary in a bounded domain belonging to \(\mathbb{R}^{m \times n}\), the value of each element in \(\nabla f\) is bounded, and can be expressed as:
\[
\nabla f_{ij} = \frac{\partial f_i}{\partial x_j} \\
\nabla f_{ij}^\text{max} = \max\{\nabla f_{ij}(X + aH)\} \\
\nabla f_{ij}^\text{min} = \min\{\nabla f_{ij}(X + aH)\}
\] (11)
\[ \nabla f_{ij}(X + \alpha H) = \lambda_1 \nabla f_{ij}^{\text{max}} + \lambda_2 \nabla f_{ij}^{\text{min}} \]  \hspace{1cm} (12)

where,

\[ \lambda_1 = \frac{\nabla f_{ij}(X + \alpha H) - \nabla f_{ij}^{\text{min}}}{\nabla f_{ij}^{\text{max}} - \nabla f_{ij}^{\text{min}}} \]
\[ \lambda_2 = \frac{\nabla f_{ij}^{\text{max}} - \nabla f_{ij}(X + \alpha H)}{\nabla f_{ij}^{\text{max}} - \nabla f_{ij}^{\text{min}}} \]  \hspace{1cm} (13)

When \( \nabla f_{ij}^{\text{max}} = \nabla f_{ij}^{\text{min}} \), one of \( \lambda_1 \) and \( \lambda_2 \) can be assumed to be 1 and the other is assumed to be 0. As a result, the above equation holds. According to the above equation, ranges of \( \lambda_1 \) and \( \lambda_2 \) are \( 0 \leq \lambda_1 \leq 1, 0 \leq \lambda_2 \leq 1 \). Therefore, according to Equation (12):

\[ \nabla f_{ij}(X + \alpha H) \in \text{Co}(\nabla f_{ij}^{\text{max}}, \nabla f_{ij}^{\text{min}}) \]  \hspace{1cm} (14)

The following equation can be given by substituting the above equation into Equation (10):

\[ f_i(X + H) - f_i(X) \in \int_0^1 \sum_{j=1}^n \left[ \text{Co}(\nabla f_{ij}^{\text{max}}, \nabla f_{ij}^{\text{min}}) h_j \right] d\alpha \]  \hspace{1cm} (15)

Let \( \Omega_i \) be \( \Omega_i = \{a|a_j \in \{\nabla f_{ij}^{\text{max}}, \nabla f_{ij}^{\text{min}} \}, j = 1, 2\ldots n\} \), namely, each point at set \( \Omega_i \) is \( a \in \mathbb{R}^n \), \( a_j \) is the \( n \)-th element of \( a \) and its value is \( \nabla f_{ij}^{\text{max}} \) or \( \nabla f_{ij}^{\text{min}} \). Thus, the above equation can also be expressed as:

\[ f_i(X + H) - f_i(X) \in \int_0^1 \text{Co}(\Omega_i) H d\alpha \]  \hspace{1cm} (16)

Considering that the integral term is irrelevant to \( a \), the above equation is equivalent to:

\[ f_i(X + H) - f_i(X) \in \text{Co}(\Omega_i) H \]  \hspace{1cm} (17)

The above equation shows that each row in the Jacobian matrix of \( f \) can be expressed in the form of differential inclusion, \( \frac{\partial f_i}{\partial X} = \text{Co}(\Omega_i) \). The left side of Equation (9) can also be expressed as:

\[ f(X + H) - f(X) = \sum_{i=1}^m (f_i(X + H) - f_i(X)) I_{m \times 1}^i \]
\[ \in \sum_{i=1}^m [\text{Co}(\Omega_i) H] I_{m \times 1}^i \]  \hspace{1cm} (18)

Where \( I_{m \times 1}^i \) denotes the \( i \)-th unit of space \( \mathbb{R}^{m \times 1} \) whose elements are all 1. According to Equation (17), the above equation is equivalent to:

\[ f(X + H) - f(X) \in \text{Co}(\Omega_f) H \]  \hspace{1cm} (19)

where \( \Omega_f = \{a|a_{ij} \in \{\nabla f_{ij}^{\text{max}}, \nabla f_{ij}^{\text{min}} \}, i = 1, 2\ldots m; j = 1, 2\ldots n\} \) means that each point in \( \Omega_f \) is \( a \in \mathbb{R}^{m \times n} \), and the value of element in \( a, a_{ij}, \) is \( \nabla f_{ij}^{\text{max}} \) or \( \nabla f_{ij}^{\text{min}} \).

QED.
According to Theorem 1, the estimation error system represented by Equation (7) can be expressed as:

\[ \Delta \dot{X} \in Co(\mathbf{\Omega}_f)\Delta X + w \]
\[ \Delta Z \in Co(\mathbf{\Omega}_h)\Delta X + \varepsilon \]  

(20)

**Theorem 2:** Considering the nonlinear system represented by Equation (5), if \( f(\cdot), h(\cdot) \) are both continuously differentiable and the value set of the Jacobian matrices is a bounded set, the estimation error system can be modelled as follows:

\[ \Delta \dot{X} = A\Delta X + w \]
\[ \Delta Z = C\Delta X + \varepsilon \]  

(21)

where matrix \( A \) and matrix \( C \) are defined as:

\[ (A, C) = \sum_{i=1}^{l}\lambda_i(A_i, C_i) \]

where \( \lambda_i \) are weights of convex combination, and \( \sum_{i=1}^{l}\lambda_i = 1, 0 \leq \lambda_i \leq 1, A_i \) and \( C_i \) are system vertex matrices of the PLDIs model; \( l \) is number of the system vertex matrices.

**Proof:** The estimation error system represented by Equation (7) can be expressed as:

\[ \begin{pmatrix} \Delta \dot{X} \\ \Delta Z \end{pmatrix} = \begin{pmatrix} f(X + \Delta X) \\ h(X + \Delta X) \end{pmatrix} - \begin{pmatrix} f(X) \\ h(X) \end{pmatrix} + \begin{pmatrix} w \\ \varepsilon \end{pmatrix} \]  

(22)

For a nonlinear filtering algorithm with good estimation performance, the difference between state estimation and real state of system, \( \Delta X \), tends to be zero or very small. As a result, we can find a proper subset \( \mathbf{\Omega}_x \subset \mathbb{R}^n \) which includes \( \hat{X} \) and \( \hat{X} + \alpha\Delta X \), where \( 0 \leq \alpha \leq 1 \). Considering that nonlinear functions \( f(\cdot), h(\cdot) \) are both continuously differentiable and partial derivatives of them vary in a bounded domain, the above estimation error system can be expressed as:

\[ \begin{pmatrix} \Delta \dot{X} \\ \Delta Z \end{pmatrix} \in Co(\mathbf{\Omega}_{th})\Delta X + \begin{pmatrix} w \\ \varepsilon \end{pmatrix} \]  

(23)

Where \( \mathbf{\Omega}_{th} \subset \mathbb{R}^{(m+n)\times n} \) is the value set of partial derivative of nonlinear function \( g(X) = (f(X)^T h(X)^T)^T \) with respect to the system state variable. The above equation represents a Linear Differential Inclusion system (LDIs). In general, \( Co(\mathbf{\Omega}_{th}) \) can be approximated by an endo-tangent polytope with any accuracy (Xie et al., 2005; Li, 2002). Therefore, there will always be a polytope \( \mathbf{\Omega} \) satisfying:

\[ Co(\mathbf{\Omega}_{th}) \approx \mathbf{\Omega} = \left\{ \begin{pmatrix} A \\ C \end{pmatrix} \right\} = \sum_{i=1}^{l}\lambda_i\begin{pmatrix} A_i \\ C_i \end{pmatrix}, 0 \leq \lambda_i \leq 1 \} \]  

(24)

Thus, Equation (23) can be expressed in the form of Equation (21).

QED.

According to Theorem 1, Theorem 2 and the autonomous navigation estimation error system indicated by Equation (7), an estimation error system can be modelled as the PLDIs model. According to the TP model transformation (Baranyi, 2004), the PLDIs model can be determined by the following steps:
Step 1: Determine the bound domain $\Omega_p$. Considering that parameters included by the Jacobian matrices are positions on three axes, $x, y, z$, $\Omega_p$ can be expressed as $\Omega_p = [c_1, d_1] \times [c_2, d_2] \times [c_3, d_3]$.

Step 2: Divide $\Omega_p$ into $I_n$ grids uniformly distributed on each dimensionality: $p_n \in \{p_{n,1}, p_{n,2}, \ldots, p_{n,I_n}\}$ where $c_n \leq p_{n,1} \leq p_{n,2} \leq \cdots \leq p_{n,I_n} \leq d_n, n = 1, 2, 3$;

Step 3: Sample the functions $S(p_n, in)$ over the hyper rectangular grid and store the sample matrices in the tensor $S$;

Step 4: Execute the Higher Order Singular Value Decomposition (HOSVD) on each dimensionality of tensor $S$, and discard not only zero n-mode singular values but also part of the non-zero n-mode singular values. Normalised n-mode matrices $U_n$, and tensor $S$ are approximated by: $\hat{S} = \bar{G}' \otimes \bar{U}'_n$

Step 5: Extract the vertex matrices $C_j$ for the PLDIs model from $\bar{G}'$.

The vertex systems of autonomous navigation system which is modelled as a PLDIs model are determined by TP model transformation. It means that the Jacobian matrix is expressed in the form of convex combination, which is composed by a variety of constant matrices. $S(p)$ is defined as a matrix composed by Jacobian matrices:

$$S(p) = \left( \frac{\partial h}{\partial X} \right), p=X$$

According to TP model transformation, $S(p)$ can be approximately expressed in the form of convex combination, composed by a variety of constant matrices.

$$S(p) \approx \sum_{i=1}^{l} \lambda_i(p) S_i = \sum_{i=1}^{l} \lambda_i(p) C_i$$

(25)

where $\lambda_i(p)$ are weights of convex combination, and $\sum_{i=1}^{l} \lambda_i(p) = 1, 0 \leq \lambda_i(p) \leq 1$, $C_i$ is the constant matrices.

Therefore, the autonomous navigation estimation error system can be described by a PLDIs model as follows:

$$\Delta \dot{X} = \Phi \Delta X + B \nu$$
$$\Delta Z = C \Delta X + D \nu$$

(26)

where

$$B = \begin{bmatrix} I_{11 \times 11} & 0_{11 \times 4} \end{bmatrix}, D = \begin{bmatrix} 0_{4 \times 11} & I_{4 \times 4} \end{bmatrix}, \nu = \begin{bmatrix} w \varepsilon \end{bmatrix}$$

$$C \in \left\{ C_j \mid C_j = \sum_{j=1}^{l} \lambda_j C_j, 0 \leq \lambda_j \leq 1, \sum_{j=1}^{l} \lambda_j = 1 \right\}$$

where $C_j$ denotes the j-th vertex system matrices, and $C_j$ is constant matrix; $l$ is number of the vertex system matrices.

3.2. Spacecraft autonomous navigation algorithm. The PLDIs model of spacecraft autonomous navigation system is determined by TP model transformation as above, and the PKF algorithm will be given by following steps based on the above results.
Forecast states:

\[ \hat{X}_{k|k-1} = \Phi_{k|k-1} \hat{X}_{k-1} \]  

Forecast estimation error covariance matrix:

\[ \hat{P}_{k|k-1} = \Phi_{k|k-1} \hat{P}_{k-1} (\Phi_{k|k-1})^T + Q_k \]  

Calculate the observed residual:

\[ \Delta Z_k = Z_k - \hat{Z}_k = Z_k - h(\hat{X}_{k|k-1}) \]  

Calculate gain: Calculate different filtering gains according to different \( C_j \).

\[ K_{j,k} = \hat{P}_{k|k-1} C_j^T \left( C_j \hat{P}_{k|k-1} C_j^T + R_k \right)^{-1} \]  

Update estimation error covariance matrix: Different estimation error covariance matrices are obtained by different gains \( K_{j,k} \) and \( C_j \), and the global estimation error covariance matrix is obtained according to data fusion.

\[ \hat{P}_{j,k} = (I_n - K_{j,k} C_j) \hat{P}_{k|k-1} (I_n - K_{j,k} C_j)^T + K_{j,k} R_k K_{j,k}^T \]  

\[ \hat{P}_{k}^{-1} = \sum_{j=1}^{l} s_j (\hat{P}_{j,k})^{-1} / \sum_{j=1}^{l} s_j \]  

Calculate correcting value of state estimation: Different correcting values are obtained for different gains \( K_{j,k} \), and the global correcting value can be given according to data fusion.

\[ \Delta \hat{X}_{j,k} = K_{j,k} \Delta Z_k \]  

\[ \Delta \hat{X}_k = \sum_{j=1}^{l} \left( s_j \Delta \hat{X}_{j,k} \right) / \sum_{j=1}^{l} s_j \]  

Update state estimation:

\[ \hat{X}_k = \hat{X}_{k|k-1} + \Delta \hat{X}_k \]  

where \( j = 1 \ldots l \), \( s_j \) denotes the j-th n-mode singular matrix of the n-mode matrix. The PKF navigation algorithm proposed in this section takes the place of EKF. The difference between the nonlinear filtering algorithm used by PKF navigation algorithm and that used by EKF navigation algorithm is that the Jacobian matrix is represented by convex combination composed by a variety of constant matrices. Compared with EKF, the algorithm proposed needs not to update the Jacobian matrix online and the algorithm design is simpler.

4. SIMULATION RESULTS. In order to verify the good performance of the PKF navigation algorithm, a numerical experiment is conducted and a comparative analysis of results based on EKF and PKF navigation algorithm will be given. The parameters for simulation are set as: \( \mu = 398600 \cdot 44 \times 10^9 \text{ m}^3/\text{s}^2 \), \( J_2 = 1.08263 \times 10^{-3} \), orbital radius of the target spacecraft \( a = 7154440 \text{ m} \), eccentricity ratio \( e = 0.0010311 \).
orbit inclination $i = 98.5^\circ$, RAAN $\Omega = 202.669^\circ$, argument of perigee $\omega = 90^\circ$; time of simulation is 8000 s, and simulation step size is 1 s. The results of numerical experiment are given as follows.

The Monte Carlo simulation experiment was conducted to verify the good performance of the PKF navigation algorithm. The performance of position estimation and velocity estimation based on PKF is shown in Figures 2 to 5. The position estimation error means on the three axes are given in Figure 2 and the mean square error of absolute position estimation error is given in Figure 4. The velocity estimation error means on the three axes are given in Figure 3 and the mean square error of absolute velocity estimation error is given in Figure 5. The red lines in Figures 2 and 3 indicate the $3\sigma$ error of the corresponding estimation error, and blue lines in Figures 2 and 3 indicate the estimation error mean based on PKF.
Figures 2 and 3 show that the position estimation error mean and velocity estimation error mean based on PKF vary within 3δ error of corresponding estimation error, which means that the filter is stable. It can be seen in Figure 2 that the position estimation error means on the three axes based on PKF are less than 5 m. Figure 4 shows that the absolute position estimation error based on PKF converges near zero over time and gradually becomes stable. During the stable stage, the absolute position estimation error based on PKF is less than 2 m. Figure 3 shows that the velocity estimation error means on the three axes based on PKF are all less than 0.5 m/s, and the absolute velocity estimation error based on PKF is less than 0.2 m/s. These figures show that the PKF navigation algorithm has not only good performance of position estimation and velocity estimation, but also good stability.

The performance of position estimation and velocity estimation based on EKF and PKF, achieved by Monte Carlo simulation experiment, is given in Figures 6 to 9.
Figure 6. Mean square errors of position estimation error.

Figure 7. Mean square errors of velocity estimation error.

Figure 8. Mean square error of absolute position estimation error.
The mean square errors of position estimation error on the three axes based on the two algorithms are shown in Figure 6. The mean square errors of velocity estimation error on the three axes based on the two algorithms are shown in Figure 7. The mean square errors of absolute position estimation error based on the two algorithms are shown in Figure 8. The mean square errors of absolute velocity estimation error based on the two algorithms are shown in Figure 9. Considering that the mean square errors of position estimation error based on PKF gradually become stable after 2000 s, simulation data after 4000 s are selected for statistical analysis. The maximal mean square errors of position estimation errors on the three axes are shown in Table 1. The maximal mean square errors of velocity estimation error on the three axes are shown in Table 2. The maximal mean square errors of absolute position estimation error and absolute velocity estimation error are shown in Table 3.

Figure 6 shows that the estimation errors based on PKF converge more slowly than those based on EKF. However, the estimation accuracies of position on the three axes based on the two algorithms are almost identical in the stable stage. Figure 7 shows that the trends of velocity estimation errors on the three axes based on PKF are similar to that based on EKF. Figure 8 shows that the estimation accuracies of absolute position based on the two algorithms are alike in stable stage, and both of them are less than 2 m. Figure 9 shows that the trends of absolute velocity estimation errors based on PKF and EKF are alike. Table 1 shows that the mean square errors of position estimation errors on the three axes based on PKF are all less than 3 m. The mean square errors of position estimation errors on the three axes based on PKF are larger than that based on EKF, but the differences between them are smaller than 0·5 m. Table 2 shows that the differences between mean square errors of velocity estimation errors on the three axes based on the two algorithms are all less than 0·1 m/s. Table 3 shows that the maximal mean square errors of absolute position based on PKF and EKF are 1·685 m and 1·439 m respectively, and the difference between them is less than 0·3 m. Besides, the maximal mean square errors of absolute velocity based on PKF and EKF in Table 3 are 0·336 m/s and 0·278 m/s respectively, and the difference between them is less than 0·1 m/s. These indicate that the estimation error.
accuracies of position and velocity based on the two algorithms are alike, and the estimation accuracy based on PKF is slightly lower than that based on EKF. The reason for the results is that in order to decrease the calculation quantity of the PKF navigation algorithm, a number of the non-zero n-mode singular values are discarded for a decreasing number of vertices in the PLDIs model, and the compromise between computational complexity and modelling accuracy is realised. As a result, the PLDIs model is an approximate model of original nonlinear estimation error system, rather than a fully accurate model.

In conclusion, the estimation accuracy of position and velocity based on PKF navigation algorithm is equivalent to that based on EKF.

5. CONCLUSION. With the aim of improving the poor real-time performance of nonlinear filtering algorithms applied to spacecraft autonomous navigation, a PKF navigation algorithm is proposed in this paper. This method is based on the fact that the spacecraft autonomous navigation estimation error system can be modelled as a Polytopic Linear Differential Inclusion (PLDI) model, and the position and velocity of spacecraft can be estimated by combining multi-model Kalman filtering with data fusion. Compared with EKF, PKF navigation algorithm does not need to update the Jacobian matrix online and the algorithm design is simpler. The simulation results show that the PKF navigation algorithm has a good navigation performance.

Table 1. Maximal mean square errors of position estimation error.

<table>
<thead>
<tr>
<th>Method</th>
<th>x/(m)</th>
<th>y/(m)</th>
<th>z/(m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EKF</td>
<td>1.881</td>
<td>1.593</td>
<td>2.290</td>
</tr>
<tr>
<td>PKF</td>
<td>2.026</td>
<td>1.554</td>
<td>2.840</td>
</tr>
</tbody>
</table>

Table 2. Maximal mean square errors of velocity estimation error.

<table>
<thead>
<tr>
<th>Method</th>
<th>x/(m/s)</th>
<th>y/(m/s)</th>
<th>z/(m/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EKF</td>
<td>0.404</td>
<td>0.329</td>
<td>0.462</td>
</tr>
<tr>
<td>PKF</td>
<td>0.466</td>
<td>0.333</td>
<td>0.563</td>
</tr>
</tbody>
</table>

Table 3. Maximal mean square error of absolute position, absolute velocity estimation error.

<table>
<thead>
<tr>
<th>Method</th>
<th>Mean square error of absolute position estimation error/(m/s)</th>
<th>Mean square error of absolute velocity estimation error/(m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EKF</td>
<td>0.278</td>
<td>1.439</td>
</tr>
<tr>
<td>FKF</td>
<td>0.336</td>
<td>1.685</td>
</tr>
</tbody>
</table>
and state estimation accuracy is similar to EKF. The PKF navigation algorithm simplifies the design of autonomous navigation algorithm, and has a simple online calculation. It is also relatively easy to implement. As a result, the PKF navigation algorithm has a better performance than EKF when applied in spacecraft autonomous navigation.

ACKNOWLEDGMENTS

This work was supported by National Natural Science Foundation of China (Grant No. 11372034) and Basic Research Foundation of Beijing Institute of Technology (No. 20130642010).

REFERENCES


