

Research on Optimal Eight-sample Rotation Vector Algorithm for SINS Attitude Updating



Lian-Peng Li^{1*}, Lun Xie¹, Zhong Su²

¹ School of Computer and Communication Engineering, University of Science and Technology Beijing
Beijing 10083, China, CN
llpstudy@163.com, xielun@ustb.edu.cn

² University of Beijing Information Science and Technology
Beijing 100010, China, CN
sz@bistu.edu.cn

Received 24 July 2017; Revised 19 September 2017; Accepted 19 October 2017

Abstract. The rotation vector algorithm is widely used in the solution of high dynamic carrier attitude, which can make proper compensation to the rotation of the noncommutativity error. In order to truly reflect the attitude information of the carrier, the more intense that the vector need to fit the curve, the higher the degree of the curve needed. In the paper, the algorithm drift of SINS mathematical platform caused by the base coning motion are analysed in detail, and made a deduction of eight-sample rotation vector algorithm. Under the coning motion, made a simulation by comparing the algorithm drift of five-sample, six-sample and eight-sample. The results show that eight-sample rotation vector algorithm can significantly improve the coning error compensation precision in high dynamic environment.

Keywords: algorithm drift, eight-sample, rotation vector, SINS

1 Introduction

SINS inertial device is fixedly connected to the carrier, and the system establish mathematics platform through attitude updating to isolate angular movement of the carrier. It does not require the establishment has a complex electromechanical physical platform, and has many advantages, such as low cost, small size, high reliability. It has become the basis for most current navigation systems. When the carrier is vibrating at the same angle frequency along two axis of the body, in the third axis of the body will commutate the constant angular velocity, which is the coning motion. Under high dynamic environment, the carrier is usually working under the condition of high speed, high rotation, high overload and so on. As far as SINS attitude updating algorithm, the coning motion is the worst working environment [1]. If we can ensure that the drift of the algorithm is the smallest under the condition of the coning motion, we can make sure that the algorithm drift is the smallest in the rest of the environments.

The coning error is produced by the algorithm drift of the mathematical platform. In essence, it is caused by the non-commutative property of the finite rotation. Reference [2] puts forward an algorithm to avoid questions of quaternion normalization or attitude matrix orthogonality constrains in the process of solving attitude. Reference [3] gives five-sample, six-sample rotation vector optimization algorithm. While the amount of computation increases with the complexity of the sample. With the development of computer technology, computational complexity is no longer a major factor restricting precision. The more intense maneuver of the carrier, the higher order is needed to fit the curve [4]. This paper deduced the eight-sample rotation vector algorithm, and the algorithm is optimized in the coning motion environment. On this basis, the work of comparing the five-sample, six-sample and eight-sample rotation vector drift error has been done under different frequencies. For the above work has a certain reference

* Corresponding Author

value to solve attitude exactly in high dynamic environment.

2 The Estimation of Eight: Sample Rotation Vector

2.1 Rotation Vector Differential Equation

The differential equation is

$$\dot{\Phi} = \omega_{nb}^b + \frac{1}{2}\Phi \times \omega_{nb}^b + \frac{1}{12}\Phi \times (\Phi \times \omega_{nb}^b). \quad (1)$$

In (1), Φ is the equivalent rotation vector corresponding to the position change of the body coordinate system from time t_k to t_{k+1} ; ω_{nb}^b is the angular rate of the body in this time period. Assume $\Phi(h)$ is the rotation vector in this time period, and $h = t_{k+1} - t_k$. The angular velocity of the carrier is fitted with a seven order equation [5].

$$\omega_{nb}^b(t_k + \tau) = \mathbf{a} + 2\mathbf{b}\tau + 3\mathbf{c}\tau^2 + 4\mathbf{d}\tau^3 + 5\mathbf{e}\tau^4 + 6\mathbf{f}\tau^5 + 7\mathbf{g}\tau^6 + 8\mathbf{j}\tau^7. \quad (2)$$

The Taylor expansion of $\Phi(h)$

$$\begin{aligned} \Phi(h) = & \Phi(0) + h\dot{\Phi}(0) + \frac{h^2}{2!}\ddot{\Phi}(0) + \frac{h^3}{3!}\dddot{\Phi}(0) + \frac{h^4}{4!}\Phi^{(4)}(0) + \frac{h^5}{5!}\Phi^{(5)}(0) + \frac{h^6}{6!}\Phi^{(6)}(0) + \frac{h^7}{7!}\Phi^{(7)}(0) \\ & + \frac{h^8}{8!}\Phi^{(8)}(0) + \frac{h^9}{9!}\Phi^{(9)}(0) + \dots \end{aligned} \quad (3)$$

In (3), $\Phi(0)$ rotation vector for a time interval of 0, which means $\Phi(0) = \mathbf{0}$.

The angular increment

$$\Delta\theta(\tau) = \int_0^\tau \omega_{nb}^b(t_k + \tau) d\tau. \quad (4)$$

According to (2) and (4), calculate each of their derivative

$$\begin{aligned} \omega_{nb}^b(t_k) &= \omega_{nb}^b(t_k + \tau)|_{\tau=0} = \mathbf{a}; \\ \dot{\omega}_{nb}^b(t_k) &= \dot{\omega}_{nb}^b(t_k + \tau)|_{\tau=0} = 2\mathbf{b}; \\ \ddot{\omega}_{nb}^b(t_k) &= \ddot{\omega}_{nb}^b(t_k + \tau)|_{\tau=0} = 6\mathbf{c}; \\ \dddot{\omega}_{nb}^b(t_k) &= \dddot{\omega}_{nb}^b(t_k + \tau)|_{\tau=0} = 24\mathbf{d}; \\ \omega_{nb}^{b(4)}(t_k) &= \omega_{nb}^{b(4)}(t_k + \tau)|_{\tau=0} = 120\mathbf{e}; \\ \omega_{nb}^{b(5)}(t_k) &= \omega_{nb}^{b(5)}(t_k + \tau)|_{\tau=0} = 720\mathbf{f}; \\ \omega_{nb}^{b(6)}(t_k) &= \omega_{nb}^{b(6)}(t_k + \tau)|_{\tau=0} = 5040\mathbf{g}; \\ \omega_{nb}^{b(7)}(t_k) &= \omega_{nb}^{b(7)}(t_k + \tau)|_{\tau=0} = 40320\mathbf{j}; \\ \omega_{nb}^{b(i)}(t_k) &= \omega_{nb}^{b(i)}(t_k + \tau)|_{\tau=0} = 0. \quad i = 8, 9, 10 \dots \end{aligned} \quad (5)$$

Then, which can be obtained as

$$\begin{aligned} \Delta\theta(0) &= \Delta\theta(\tau)|_{\tau=0} = \mathbf{0}; \\ \Delta\dot{\theta}(0) &= \Delta\dot{\theta}(\tau)|_{\tau=0} = \mathbf{a}; \\ \Delta\ddot{\theta}(0) &= \Delta\ddot{\theta}(\tau)|_{\tau=0} = 2\mathbf{b}; \\ \Delta\dddot{\theta}(0) &= \Delta\dddot{\theta}(\tau)|_{\tau=0} = 6\mathbf{c}; \\ \Delta\theta^{(4)}(0) &= \Delta\theta^{(4)}(\tau)|_{\tau=0} = 24\mathbf{d}; \\ \Delta\theta^{(5)}(0) &= \Delta\theta^{(5)}(\tau)|_{\tau=0} = 120\mathbf{e}; \\ \Delta\theta^{(6)}(0) &= \Delta\theta^{(6)}(\tau)|_{\tau=0} = 720\mathbf{f}; \end{aligned} \quad (6)$$

$$\begin{aligned}\Delta\theta^{(7)}(0) &= \Delta\theta^{(7)}(\tau)|_{\tau=0} = 5040\mathbf{g}; \\ \Delta\theta^{(8)}(0) &= \Delta\theta^{(8)}(\tau)|_{\tau=0} = 42320\mathbf{j}; \\ \Delta\theta^{(i)}(0) &= \Delta\theta^{(i)}(\tau)|_{\tau=0} = 0; \quad i=9,10,11,\dots\end{aligned}$$

Generally, due to the amount of the general attitude updating h is milliseconds, and Φ is also a small number. So according to (1), when calculate the $\Phi(\tau)$'s derivative as $\tau = 0$, the third item in the (1) can be regarded as a small amount of the two order and neglected. $\Phi(\tau)$ in the second item can be replaced by angular increment

$$\Phi(\tau) \approx \Delta\theta(\tau) \quad (7)$$

Then the (1) can be expressed as

$$\dot{\Phi}(\tau) = \omega_{nb}^b(t_k + \tau) + \frac{1}{2}\Delta\theta(\tau)\omega_{nb}^b(t_k + \tau) \quad (8)$$

By calculating, each of (1) its derivative are as follows

$$\ddot{\Phi}(\tau) = \dot{\omega}_{nb}^b(t_k + \tau) + \frac{1}{2}\Delta\dot{\theta}(\tau) \times \omega_{nb}^b(t_k + \tau) + \frac{1}{2}\Delta\theta(\tau) \times \dot{\omega}_{nb}^b(t_k + \tau) \quad (9)$$

$$\begin{aligned}\ddot{\Phi}(\tau) &= \dot{\omega}_{nb}^b(t_k + \tau) + \frac{1}{2}\Delta\ddot{\theta}(\tau) \times \omega_{nb}^b(t_k + \tau) + \frac{1}{2}\Delta\dot{\theta}(\tau) \times \dot{\omega}_{nb}^b(t_k + \tau) + \frac{1}{2}\Delta\dot{\theta}(\tau) \times \dot{\omega}_{nb}^b(t_k + \tau) \\ &+ \frac{1}{2}\Delta\theta(\tau) \times \ddot{\omega}_{nb}^b(t_k + \tau)\end{aligned} \quad (10)$$

.....

$$\Phi^{(14)}(\tau) = 621\Delta\theta^{(8)}(\tau) \times \omega_{nb}^{(5)b}(t_k + \tau) + 798\Delta\theta^{(7)}(\tau) \times \omega_{nb}^{(6)b}(t_k + \tau) + 753\Delta\theta^{(6)}(\tau) \times \omega_{nb}^{(7)b}(t_k + \tau) \quad (11)$$

$$\Phi^{(15)}(\tau) = 1419\Delta\theta^{(8)}(\tau) \times \omega_{nb}^{(6)b}(t_k + \tau) + 1551\Delta\theta^{(7)}(\tau) \times \omega_{nb}^{(7)b}(t_k + \tau) \quad (12)$$

$$\Phi^{(16)}(\tau) = 2970\Delta\theta^{(8)}(\tau) \times \omega_{nb}^{(7)b}(t_k + \tau) \quad (13)$$

$$\Phi^{(i)}(\tau) = \mathbf{0} \quad ; \quad i=17,18,19,\dots \quad (14)$$

Taking the values of $\omega_{nb}^{b(i)}(t_k)$, $\Delta\theta^{(i)}(0)$ and $\tau = 0$ into the above equations form (9) to (14). Furthermore, takes the results into (3). And then, $\Phi(h)$ can be expressed as

$$\begin{aligned}\Phi(h) &= ah + bh^2 + \frac{(6c+a \times b)}{6} \times h^3 + \frac{6d+a \times c}{4} h^4 + \frac{10e+2a \times d+b \times c}{10} h^5 + \frac{30f+5a \times e+4b \times d}{30} h^6 + \frac{35g+5a \times f+5b \times e+2c \times d}{35} h^7 + \\ &\frac{56j+7a \times g+8b \times f+5c \times e}{56} h^8 + \frac{28a \times j+35b \times g+27c \times f+10d \times e}{252} h^9 + \frac{8b \times j+7c \times g+4d \times f}{60} h^{10} + \frac{20c \times j+14d \times g+5e \times f}{165} h^{11} + \\ &\frac{64d \times j+35e \times f}{660} h^{12} + \frac{20e \times j+7f \times g}{286} h^{13} + \frac{8320128f \times j}{871782912} h^{14} + \frac{2682409000g \times j}{1307674368000} h^{15}\end{aligned} \quad (15)$$

2.2 Solving Differential Equations

In order to represent the h , b , c , d , e , f and j these come from the above formula by the angular increment, sampled eight times of gyro outputs in an attitude-updating cycle [6]. The results is as follows:

$$\Delta\theta_i = \int_{\frac{i-1}{8}h}^{\frac{i}{8}h} \omega_{nb}^b(t_k + \tau) d\tau, i=1,2,3,4,5,6,7,8 \tag{16}$$

The next part is to calculate $\Delta\theta_1, \Delta\theta_2, \Delta\theta_3, \Delta\theta_4, \Delta\theta_5, \Delta\theta_6, \Delta\theta_7, \Delta\theta_8$ in turn. The calculated results are in the matrix of P .

Form the calculation above, we can get

$$ah + bh^2 + ch^3 + dh^4 + eh^5 + fh^6 + gh^7 + jh^8 = \Delta\theta_1 + \Delta\theta_2 + \Delta\theta_3 + \Delta\theta_4 + \Delta\theta_5 + \Delta\theta_6 + \Delta\theta_7 + \Delta\theta_8 \tag{17}$$

The results of the above calculation are expressed in a matrix

$$\Theta = PH \tag{18}$$

In (18), the specific content of Θ, P, H is described as

$$\Theta = \begin{bmatrix} \Delta\theta_1 \\ \Delta\theta_2 \\ \Delta\theta_3 \\ \Delta\theta_4 \\ \Delta\theta_5 \\ \Delta\theta_6 \\ \Delta\theta_7 \\ \Delta\theta_8 \end{bmatrix}, H = \begin{bmatrix} ah \\ bh^2 \\ ch^3 \\ dh^4 \\ eh^5 \\ fh^6 \\ gh^7 \\ jh^8 \end{bmatrix} \tag{19}$$

$$P = \begin{bmatrix} \frac{1}{8} & \frac{1}{64} & \frac{1}{512} & \frac{1}{4096} & \frac{1}{32768} & \frac{1}{262144} & \frac{1}{2097152} & \frac{1}{16777216} \\ \frac{1}{8} & \frac{3}{64} & \frac{7}{512} & \frac{15}{4096} & \frac{31}{32768} & \frac{63}{262144} & \frac{127}{2097152} & \frac{255}{16777216} \\ \frac{1}{8} & \frac{5}{64} & \frac{19}{512} & \frac{65}{4096} & \frac{211}{32768} & \frac{665}{262144} & \frac{2059}{2097152} & \frac{6305}{16777216} \\ \frac{1}{8} & \frac{7}{64} & \frac{37}{512} & \frac{175}{4096} & \frac{781}{32768} & \frac{3367}{262144} & \frac{14197}{2097152} & \frac{58975}{16777216} \\ \frac{1}{8} & \frac{9}{64} & \frac{61}{512} & \frac{369}{4096} & \frac{2101}{32768} & \frac{11529}{262144} & \frac{61741}{2097152} & \frac{325089}{16777216} \\ \frac{1}{8} & \frac{11}{64} & \frac{91}{512} & \frac{671}{4096} & \frac{4651}{32768} & \frac{31031}{262144} & \frac{201811}{2097152} & \frac{1288991}{16777216} \\ \frac{1}{8} & \frac{13}{64} & \frac{127}{512} & \frac{1105}{4096} & \frac{9031}{32768} & \frac{70993}{262144} & \frac{543607}{2097152} & \frac{4085185}{16777216} \\ \frac{1}{8} & \frac{15}{64} & \frac{169}{512} & \frac{1695}{4096} & \frac{15961}{32768} & \frac{144495}{262144} & \frac{1273609}{2097152} & \frac{11012415}{16777216} \end{bmatrix} \tag{20}$$

By matrix operation

$$H = P^{-1}\Theta \tag{21}$$

Thus from (21), the range from ah to jh^8 can be expressed by $\Delta\theta_i$. So the expression of $\Phi(h)$ can be replaced by ah and so on. Making use of MATLAB to carry out the following calculation: select the two-row elements of the matrix P^{-1} in turn, and select the two-column elements in these two-row elements to make a cross multiply. Next, the difference of the cross multiply can be obtained. Moreover, a matrix consists of 28 rows and 28 columns. The last step is that the matrix of each-row element multiplied by the last 28 corresponding coefficients of (15). Finally, a matrix is composed of 1 row and 28 columns by

calculating the sum of each column element.

Then, the eight-sample rotation vector algorithm can be expressed as

$$\begin{aligned}
 \Phi = & \Delta\theta_1 + \Delta\theta_2 + \Delta\theta_3 + \Delta\theta_4 + \Delta\theta_5 + \Delta\theta_6 + \Delta\theta_7 + \Delta\theta_8 \\
 & -164432.567598473\Delta\theta_1\Delta\theta_2 + 671132.544017758\Delta\theta_1\Delta\theta_3 \\
 & -1042191.06723711\Delta\theta_1\Delta\theta_4 + 782663.992317715\Delta\theta_1\Delta\theta_5 \\
 & -279592.955604012\Delta\theta_1\Delta\theta_6 + 31291.25929746\Delta\theta_1\Delta\theta_7 \\
 & +3581.13863130481\Delta\theta_1\Delta\theta_8 - 1191045.14143486\Delta\theta_2\Delta\theta_3 \\
 & +1519140.78057143\Delta\theta_2\Delta\theta_4 + 60109.8620532868\Delta\theta_2\Delta\theta_5 \\
 & -1156830.42094525\Delta\theta_2\Delta\theta_6 + 738632.490518788\Delta\theta_2\Delta\theta_7 \\
 & -149963.228308971\Delta\theta_2\Delta\theta_8 + 1157407.75865753\Delta\theta_3\Delta\theta_4 \\
 & -5402372.18747198\Delta\theta_3\Delta\theta_5 + 6220039.78901697\Delta\theta_3\Delta\theta_6 \\
 & -2994522.46566882\Delta\theta_3\Delta\theta_7 + 541490.082229038\Delta\theta_3\Delta\theta_8 \\
 & +6759864.21132678\Delta\theta_4\Delta\theta_5 - 8739006.42284086\Delta\theta_4\Delta\theta_6 \\
 & +4346880.95315698\Delta\theta_4\Delta\theta_7 - 796282.211873293\Delta\theta_4\Delta\theta_8 \\
 & +4601231.65541143\Delta\theta_5\Delta\theta_6 - 2933961.14399365\Delta\theta_5\Delta\theta_7 \\
 & +589729.336755194\Delta\theta_5\Delta\theta_8 + 836207.665252431\Delta\theta_6\Delta\theta_7 \\
 & -221242.889632138\Delta\theta_6\Delta\theta_8 + 33921.2851240353\Delta\theta_7\Delta\theta_8
 \end{aligned} \tag{22}$$

3 Cone Motion Algorithm Optimization and Algorithm Drift

As far as SINS attitude updating algorithm, the coning motion is the worst working environment. It will lead to serious drift of the mathematical platform. So the coning motion is often used as the environmental condition in the rotation vector optimization algorithm [7].

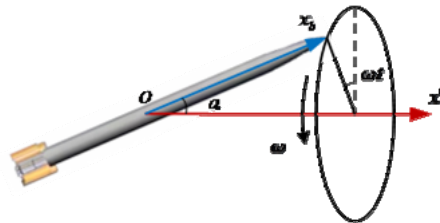


Fig. 1. Schematic diagram of the cone motion

Quaternion describes the carrier's real-time orientation that relative to the reference coordinate. The quaternion update formula of cone motion is

$$\mathbf{Q}(t+h) = \mathbf{Q}(t) \otimes \mathbf{q}(h) \tag{23}$$

Typical movement of the rotary cone quaternion can be assumed to be as follows

$$\mathbf{Q}(t) = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} \cos \frac{\alpha}{2} \\ 0 \\ \sin \frac{\alpha}{2} \cos \omega t \\ \sin \frac{\alpha}{2} \sin \omega t \end{bmatrix} \tag{24}$$

Angular velocity of the body coordinate system is:

$$\boldsymbol{\omega}(t) = \begin{bmatrix} -2\omega \sin^2 \frac{\alpha}{2} \\ -\omega \sin \alpha \sin \omega t \\ \omega \sin \alpha \cos \omega t \end{bmatrix} \quad (25)$$

By analyzing (16) with (24), $\Delta\theta_i$ can be obtained as follows

$$\Delta\theta_i = \begin{bmatrix} -\frac{1}{4}\omega h \sin^2 \frac{\alpha}{2} \\ \sin \alpha \left[\cos\left(\omega t + \frac{i}{8}\omega h\right) - \cos\left(\omega t + \frac{i-1}{8}\omega h\right) \right] \\ \sin \alpha \left[\sin\left(\omega t + \frac{i}{8}\omega h\right) - \sin\left(\omega t + \frac{i-1}{8}\omega h\right) \right] \end{bmatrix}, \quad i = 1, 2, 3, 4, 5, 6, 7, 8 \quad (26)$$

Take (26) into (22) and we can calculate the value of $\Phi(h)$. For the solution process is too complex, we need to use MATLAB to calculate. After the $\Phi(h)$ has been calculated, can we obtain the change of quaternion attitude $q(h)$ by the following formula

$$q(h) = \cos \frac{\Phi}{2} + \frac{\Phi}{\Phi} \sin \frac{\Phi}{2} \quad (27)$$

$$C_b^R = \begin{bmatrix} 1 - 2(q_2^2 + q_3^2) & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & 1 - 2(q_1^2 + q_3^2) & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & 1 - 2(q_1^2 + q_2^2) \end{bmatrix} \quad (28)$$

If the reference coordinate system is R , navigation coordinate system is n , and fixed coordinate system b is body coordinate system coordinates carrier fixedly connected to the body coordinate system. Then the coordinate transformation matrix C_b^R is the attitude matrix C_b^n .

$$C_n^b = \begin{bmatrix} \cos \gamma \cos \psi + \sin \gamma \sin \psi \sin \theta & -\cos \gamma \sin \psi + \sin \gamma \cos \psi \sin \theta & -\sin \gamma \cos \theta \\ \sin \psi \cos \theta & \cos \psi \cos \theta & \sin \theta \\ \sin \gamma \cos \psi - \cos \gamma \sin \psi \sin \theta & -\sin \gamma \sin \psi - \cos \gamma \cos \psi \sin \theta & \cos \gamma \cos \theta \end{bmatrix} \quad (29)$$

In (29), ψ is the yaw angle; γ is the roll angle and θ is the pitch angle. (29) can be simplified

$$C_n^b = (C_b^R)^T = \begin{bmatrix} T_{11} & T_{21} & T_{31} \\ T_{12} & T_{22} & T_{32} \\ T_{13} & T_{23} & T_{33} \end{bmatrix} \quad (30)$$

$$\begin{cases} \gamma = \arctan\left(-\frac{T_{31}}{T_{33}}\right) \\ \psi = \arctan\left(\frac{T_{12}}{T_{22}}\right) \\ \theta = \arcsin(T_{32}) \end{cases} \quad (31)$$

Taking the (27) into (23) and doing the iterative calculation, combining with (31) can solve coning motion attitude updating. Based on the assumption that under the condition of cone motion, the multiplication cross between the two angular increments with the same time interval is equal to the contribution of the cone. And then we obtain the angular increments h in the update cycle [8].

$$\begin{aligned}
 \Phi(h) = & \Delta\theta_1 + \Delta\theta_2 + \Delta\theta_3 + \Delta\theta_4 + \Delta\theta_5 + \Delta\theta_6 + \Delta\theta_7 + \Delta\theta_8 + \frac{k_1}{7}(\Delta\theta_1 \times \Delta\theta_2 + \Delta\theta_2 \times \Delta\theta_3 + \Delta\theta_3 \times \Delta\theta_4 + \Delta\theta_4 \times \Delta\theta_5 + \\
 & \Delta\theta_5 \times \Delta\theta_6 + \Delta\theta_6 \times \Delta\theta_7 + \Delta\theta_7 \times \Delta\theta_8) + \frac{k_2}{6}(\Delta\theta_1 \times \Delta\theta_3 + \Delta\theta_2 \times \Delta\theta_4 + \Delta\theta_3 \times \Delta\theta_5 + \Delta\theta_4 \times \Delta\theta_6 + \Delta\theta_5 \times \Delta\theta_7 + \\
 & \Delta\theta_6 \times \Delta\theta_8) + \frac{k_3}{5}(\Delta\theta_1 \times \Delta\theta_4 + \Delta\theta_2 \times \Delta\theta_5 + \Delta\theta_3 \times \Delta\theta_6 + \Delta\theta_4 \times \Delta\theta_7 + \Delta\theta_5 \times \Delta\theta_8) + \frac{k_4}{4}(\Delta\theta_1 \times \Delta\theta_5 + \Delta\theta_2 \times \Delta\theta_6 \\
 & + \Delta\theta_3 \times \Delta\theta_7 + \Delta\theta_4 \times \Delta\theta_8) + \frac{k_5}{3}(\Delta\theta_1 \times \Delta\theta_6 + \Delta\theta_2 \times \Delta\theta_7 + \Delta\theta_3 \times \Delta\theta_8) + \frac{k_6}{2}(\Delta\theta_1 \times \Delta\theta_7 + \Delta\theta_2 \times \Delta\theta_8) + k_7(\Delta\theta_1 \times \Delta\theta_8)
 \end{aligned} \quad (32)$$

In (32), $k_1, k_2, k_3, k_4, k_5, k_6, k_7$ are the coefficients that needed to be optimized. The goal of optimization is to minimize the drift of the algorithm in the coning motion environment. Reference [4] and [9], it is proved that the optimal factors that reserved 30 significant digits are as follows

$$\begin{aligned}
 k_1 &= 4.797496296561896974232881093253 \\
 k_2 &= 2.522625311893548106657982240851 \\
 k_3 &= 2.655608324038598864700158238978 \\
 k_4 &= 1.962535994387389100548923847856 \\
 k_5 &= 1.524329329681241127889234795729 \\
 k_6 &= 0.989572679185493012389495953091 \\
 k_7 &= 0.500899439210667366836584019325
 \end{aligned}$$

The calculation formula of drift error for algorithm can be expressed as

$$\Phi_e = |2q_1 - \Phi_x| \quad (33)$$

In (33), Φ_e is the error of rotation vector. $\Phi_e = |\Phi_e|$, Φ_x is the value of the first row in matrix that the calculated in the process.

According to the (39), (40) and the optimization coefficients, the algorithm error generated in the attitude update cycle is as follows

$$|\dot{\Phi}_{er}| = \frac{\alpha^2 \omega (\omega h)^{16}}{1.350685271475316e+19} \quad (34)$$

3 Simulation Analysis and Comparison of Algorithm Drift

In order to verify the performance and correctness of the optimal eight-sample rotation vector algorithm derived by this paper, create the following simulation by MATLAB.

Assuming the coning motion frequency $f=1\text{Hz}$, the update cycle $h=0.024\text{s}$, the half cone angle $\alpha=1^\circ$, the simulation time t is 10s. Under these conditions, we use the eight-sample rotation vector algorithm angle to calculate the attitude angle. The simulation results are as follows

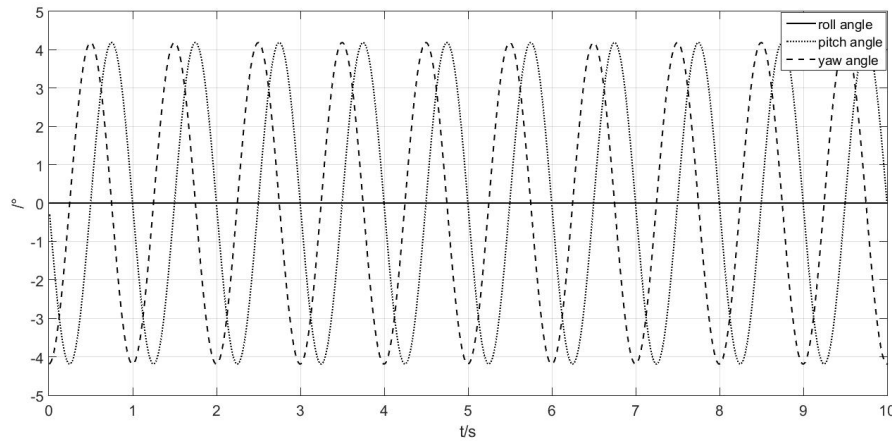


Fig. 2. Attitude angle

From Fig. 2, the trend of the curve reflects the setted parameters. The simulation result shows that eight-sample rotation vector optimization algorithm can correctly calculate the attitude information of the carrier in coning motion. This indicates the correctness of algorithm.

Combined with the [6] and [10], table 1 lists the algorithm drift of five-sample, six-sample and eight-sample rotation vector at the frequency of 2, 4, 8, 15, 30, 50Hz. And the attitude updating period is $h=10\text{ms}$, the half cone angle is $\alpha=1^\circ$.

Table 1. Algorithm drift comparison

Cone Motion Frequency/Hz	The optimal five-sample $/(^{\circ}/h)$	The optimal six-sample $/(^{\circ}/h)$	The optimal eight-sample $/(^{\circ}/h)$
2	$1.43207283 \times 10^{-17}$	$1.48261384 \times 10^{-20}$	$1.09589326 \times 10^{-36}$
4	$2.93288517 \times 10^{-14}$	$3.03639314 \times 10^{-17}$	$1.43640922 \times 10^{-31}$
8	$6.00654883 \times 10^{-11}$	$6.21853314 \times 10^{-14}$	$1.88273029 \times 10^{-26}$
15	$6.04837911 \times 10^{-8}$	$6.26183971 \times 10^{-11}$	$8.23777142 \times 10^{-22}$
30	$1.23870804 \times 10^{-4}$	$1.28242477 \times 10^{-7}$	$4.86674375 \times 10^{-18}$
50	$3.41432771 \times 10^{-2}$	$3.53482684 \times 10^{-5}$	$6.37893837 \times 10^{-13}$

From Table 1, based on the compared five-sample, six-sample and eight-sample rotation vector algorithm drift, we can get: When the frequency is 2Hz, compared with five-sample and six-sample, the eight-sample algorithm drift error is little more than 10 orders of magnitude. All in all, each sample algorithm drift will increase with the increase of the coning motion's frequency. In the same frequency, the eight-sample rotation vector algorithm drift is far less than five-sample and six-sample algorithm drift errors.

4 Conclusion

In order to truly reflect the movement of the carrier under high dynamic environment, the more intense the angular motion of the carrier requires higher order curves to fit. This paper deduces the eight-sample rotation vector, attitude updating formula and algorithm optimization in coning motion environment. Simulation is done based on the optimal eight-sample rotation vector algorithm. The result shows the formula we got is true. Under the same cone motion frequency, the algorithm drift error of eight-sample rotation vector algorithm is far less than the low-sample algorithm drift error. The inhibitory effect of noncommutativity error is more obvious, and the reason is that eight-sample rotation vector algorithm increase the adjacent angle's noncommutativity error compensation.

Although the eight-sample rotation vector algorithm has the defects of complex computation with large computational quantity. Along with the development of multi-core processor, its high accuracy of the prominent advantages will be paid more and more attention. The application of high-sample rotation

vector algorithm will be more and more extensive.

Acknowledgements

This work is supported by National Key Research and Development Program (NO. 2017YFB1302104).

References

- [1] J. Collin, MEMS IMU carouseling for ground vehicles, *IEEE Transactions on Vehicular Technology* 64(6)(2015) 2242-2251.
- [2] H.F. Grip, T.I. Fossen, T.A. Johansen, A. Saberi, Nonlinear observer for GNSS-aided inertial navigation with quaternion-based attitude estimation, in: *Proc. American Control Conference (ACC)*, 2013.
- [3] Z.S. Shariatmadar, K. Faez, Finger-knuckle-print recognition via encoding local-binary-pattern, *Journal of Circuits, Systems and Computers* 22(6)(2013) 1350050.
- [4] V. Renaudin, C. Combettes, Magnetic, acceleration fields and gyroscope quaternion (MAGYQ)-based attitude estimation with smartphone sensors for indoor pedestrian navigation, *Sensors* 14(12)(2014) 22864-22890.
- [5] M. Wang, W. Wu, J. Wang, X. Pan, High-order attitude compensation in coning and rotation coexisting environment, *IEEE Transactions on Aerospace and Electronic Systems* 51(2)(2015) 1178-1190.
- [6] R. Costanzi, F. Fanelli, N. Monni, A. Ridolfi, B. Allotta, An attitude estimation algorithm for mobile robots under unknown magnetic disturbances, *IEEE/ASME Transactions on Mechatronics* 21(4)(2016) 1900-1911.
- [7] L. Li, Z. Su, A Kind of Attitude Algorithm for High Dynamic IMU, in: *Proc. 16th Asia Simulation Conference and SCS Autumn Simulation Multi-Conference*, 2016.
- [8] F. Guo, L. Xie, J. Chen, C. Song, S. Wang, H. Yu, Research of SINS/DVL/OD integrated navigation system based on observability analysis, in: *Proc. 2016 35th Chinese Control Conference (CCC)*, 2016.
- [9] H. Nobahari, H. Mohammadkarimi, Accuracy analysis of an integrated inertial navigation system in slow maneuvers, *Navigation* 64(2)(2017) 197-211.
- [10] H. Liu, X. Wang, Y. Zhong, Quaternion-based robust attitude control for uncertain robotic quadrotors, *IEEE Transactions on Industrial Informatics* 11(2)(2015) 406-415.