Research Article

Fast SINS Initial Alignment Method Based on Iterative Algorithms in Inertial Frame

Dongrui Yang ^{1,2}, Xiaosu Xu ^{1,2} and Yiqing Yao ^{1,2}

¹Key Laboratory of Micro-Inertial Instrument and Advanced Navigation Technology, Ministry of Education, Nanjing 210096, China

²School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China

Correspondence should be addressed to Xiaosu Xu; xxs@seu.edu.cn

Received 26 August 2020; Revised 19 September 2020; Accepted 29 September 2020; Published 13 October 2020

Academic Editor: Gaoge Hu

Copyright © 2020 Dongrui Yang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Iterative algorithms based on saved data have been researched for strapdown inertial navigation system to make the best use of sensor data, and different alignment algorithms can be carried out during the iterative process to fulfil initial alignment. This paper firstly analysed two interactive processes. The forward-backward process needs to change the normal system model in the backward process and have additional error factors. While the forward-forward process is easy to be carried out without changing the system model, the transition error will also decrease the accuracy. In addition, traditional iterative methods always need to save a large amount of data and do redundant computation, which is not highly effective, especially in view of the real-time property. To address these effects, a forward-forward solution in the inertial frame based on Kalman filter is analysed. This method calculates the attitude transfer matrix in the inertial frame to deal with angular disturbances and avoids saving all the original measurement data during the process, which makes the method more stable and faster to fulfil in the real-time system. Simulation and turntable test validates the performance of the proposed method.

1. Introduction

Strapdown inertial navigation system (SINS) is widely used in the field of navigation and positioning, such as vehicles, ships, submarines, and aircrafts [1]. Different from the Global Navigation Satellite System (GNSS), SINS is a dead-reckoning navigation system [2], which does not rely on external reference and is less susceptible to interference. However, SINS cannot be calibrated without reference information and will have error accumulation during the navigation process [3, 4]. The performance of SINS highly depends on the accuracy of the initial navigation parameters, among which the initial velocity and position can be easily obtained from external references such as GPS and DVL [5, 6]. But normally, the initial attitude needs to be obtained through the initial alignment process by the system, which becomes the main factor that influences the navigation accuracy. High precision of the alignment is of vital importance for inertial navigation and positioning.

Thus, the initial alignment is a key technology for SINS [7, 8].

Normally, the initial alignment could be divided into two stages: coarse alignment and fine alignment [9]. Coarse alignment utilizes the analytic method to roughly estimate the rotation matrix between the body frame and navigation frame in a short time. Ideally, the horizontal misalignment angles will be less than one degree, and the heading misalignment angle is constrained in several degrees [10]. Due to the small misalignment angles, the error model of SINS can be approximated to a linear model after coarse alignment. During the fine alignment stage, the misalignment angles will be decreased effectively and sensor errors could be estimated based on the error model of SINS. “Velocity” or “position” matching methods, which are based on Kalman filter (KF), are usually used as fine alignment methods [11–13]. KF is known to be an optimal estimator, which is used to estimate the misalignment angles and sensor errors in the alignment process. Many research studies have been

done to improve the performance of KF. Hu et al. [14] and Shen et al. [15] applied the nonlinear filtering technology in the integrated navigation system, which are used to improve the adaptiveness, but the alignment time is hard to decrease remarkably. Another fine alignment method is the compass method, which is derived from the platform initial navigation system [16]. Different from KF, the compass method is based on the classical control theory to realize initial alignment. It does not need precise mathematical models and noise models, whereas it cannot estimate the sensor bias.

Precision and time are the most important indexes during the initial alignment process of SINS. Many solutions have been proposed to shorten the alignment time or increase the alignment accuracy [17–21]. However, it is hard to shorten the time while increasing the accuracy as there is contradiction between alignment time and accuracy [22]. For SINS, inertial sensors are installed directly on the carrier, and the mathematical platform is calculated in the navigation process. In this case, the relationship between the carrier and sensors is constant, and different navigation results can be obtained with the same set of sensor data but with different attitude matrixes. Then, the navigation errors could be used to calculate the misalignment angles during the fine alignment process. The same set of sensor data of SINS can be transformed to different projected data in the mathematical platform with different attitude matrixes, and these projected data can be used in the alignment method to generate control signals to update the attitude matrix. Then, new projected data can be obtained from the new attitude matrix. Based on this, the same set of sensor data can be used in the fine alignment process repeatedly to get a better attitude matrix with less navigation errors.

Fast initial alignment methods for SINS based on the saved data and iterative calculation have been researched a lot in recent years [17, 22–25]. The iterative process could be divided into forward-forward process and forward-backward process, and then the normal fine alignment algorithm is combined with the iterative process. Under ideal conditions, only a short set of sensor data is enough to fulfil the alignment, but the sensor errors and different motion conditions will have a bad effect on the accuracy of practical application [5]. Normally, a few seconds or several minutes of data will be needed. On the other hand, the normal iterative process needs to save all the measurement data from the inertial sensors to update navigation parameters during each iterative period, which will be a considerable amount of data need to be saved and calculated per second. And during the iterative process, new measurement data also need to be saved to transform the alignment result to the real-time process. The more time used in the iterative process, the more data need to be saved and more errors will be accumulated. It is inefficient with saving all the sensor data, and this will cause redundant calculation.

This paper analyses the typical iterative process and researches the improved alignment method in the inertial frame with forward-forward process. Error model of the inertial frame is adopted to fulfil the alignment, while decreasing the amount of saved data and shortening the time of iterative calculation. The rest of this paper is organized as

follows: the error model of SINS in the navigation frame is introduced in Section 2. Then, iterative process is further analysed, and simulation of the forward-forward process and forward-backward process is given in Section 3. In Section 4, the errors of the alignment method based on iterative process are analysed. Then, the alignment method in the inertial frame based on forward-forward process is proposed in Section 5. Simulation and turntable test are done in Section 6 to validate the proposed method. Finally, conclusions are drawn in Section 7.

2. Error Model of SINS

In order to avoid confusion, the different coordinate frames in this paper are defined as follows:

The i -frame is the earth-centred initially fixed orthogonal coordinate frame. The e -frame is the earth-centred earth-fixed (ECEF) orthogonal reference frame. The n -frame is the navigation coordinate frame, aligned with east-north-up (ENU) geodetic axes.

The n_0 -frame is the navigation inertial coordinate frame. It is formed by fixing the n -frame at the beginning of the process.

The b -frame is the orthogonal reference frame aligned with inertial measurement unit (IMU) axes.

The b_0 -frame is the body inertial coordinate frame. It is formed by fixing the b -frame at the beginning of the process.

The n -frame has followed through three Euler angles rotation to b -frame. The rotation transformation relationship between the n -frame and b -frame can be described by the direct cosine matrix (DCM) C_n^b [24]. However, there are several kinds of measurement errors and disturbances in the actual navigation system, so rotational errors usually exist between the calculated navigation coordinate frame (n' -frame) and the ideal n -frame. $\phi = [\phi_E \ \phi_N \ \phi_U]^T$ is used to describe the misalignment angle between the n -frame and n' -frame, and the rotation transformation relationship between C_n^b and $C_{n'}^b$ is

$$C_n^b = C_{n'}^b C_n^{n'} \quad (1)$$

From [26, 27], the attitude error equations and velocity error equations of SINS on swaying base can be simplified as follows:

$$\dot{\phi} = \phi \times \omega_{in}^n - \delta \omega_{in}^n - \varepsilon^n, \quad (2)$$

$$\delta \dot{V}^n = -\phi \times C_b^n f^b + \delta V^n \times (2\omega_{ie}^n + \omega_{en}^n) + \nabla^n, \quad (3)$$

where ω_{in}^n is the rotating rate of the n -frame with respect to the i -frame, which is projected onto the n -frame, and $\varepsilon^n = [\varepsilon_E \ \varepsilon_N \ \varepsilon_U]^T = C_b^n \varepsilon^b$, ε^b is the gyro drift rate. $\delta V^n = [\delta V_E \ \delta V_N \ \delta V_U]^T$ is the velocity error, f^b is the accelerometer measurement, $\omega_{ie}^n = [0 \ \omega_{ie} \cos L \ \omega_{ie} \sin L]^T$, and L is the latitude, ω_{ie} is the magnitude of Earth rotation rate, and ω_{en}^n is the angular rate of n -frame with respect to the e -frame, which is projected onto the n -frame. $\nabla^n = [\nabla_E \ \nabla_N \ \nabla_U] = C_b^n \nabla^b$ and ∇^b is the accelerometer bias.

3. Iterative Navigation Algorithms for SINS

With inertial sensors being directly installed on the carrier of SINS, the mathematical platform in SINS replaces the physical platform in the traditional platform inertial navigation system (PINS) [28]. For SINS, when the carrier is stationary, the measurement data of the ideal error-free inertial sensors stay the same, which means the attitude could be calculated by using one sampling of sensor data repeatedly. The attitude relationship between the b -frame and n -frame with the DCM C_n^b is determined at the beginning and stays the same through the whole navigation process. As a result, with a period of saved measurement data, the attitude could also be calculated with iterative algorithms [5].

Iterative algorithms could be divided into forward-forward algorithm and forward-backward algorithm, and the design of them is shown in Figure 1. During the normal process from t_0 to t_n , the sensor data is saved, and the normal forward navigation algorithm is performed. Then, the data from t_0 to t_n is used to do the forward process repeatedly as shown in Figure 1(a), while the backward process is performed with the data from t_n to t_0 after the forward process in Figure 1(b). The data of time t_a^b means the data at time point t_a in the b th iterative period of the process.

3.1. Algorithms of Forward Navigation Process and Backward Navigation Process. Assuming that the attitude, velocity, and position are all calculated or given at the origin of the inertial navigation, velocity and attitude during the forward navigation process can be updated as follows [28]:

$$\dot{C}_b^n = C_b^n [\omega_{nb}^\times], \quad (4)$$

$$\dot{\mathbf{V}}^n = C_b^n \mathbf{f}^b - (2\omega_{ie}^n + \omega_{en}^n) \times \mathbf{V}^n + \mathbf{g}^n, \quad (5)$$

where

$$\omega_{nb}^\times = \omega_{ib}^\times - C_n^b (\omega_{ie}^n + \omega_{en}^n), \quad (6)$$

and $[\cdot]^\times$ denotes the skew symmetric matrix of the vector $\cdot$.

The forward process is from the origin to the end, while the backward process is from the end to the origin, which is a reverse process of the normal navigation process. Similarly, the backward navigation process can be obtained by rearranging equations (4) and (5) as follows:

$$\dot{C}_b^{\leftarrow n} = C_b^{\leftarrow n} [\omega_{nb}^{\leftarrow \times}], \quad (7)$$

$$\dot{\mathbf{V}}^{\leftarrow n} = C_b^{\leftarrow n} \mathbf{f}^{\leftarrow b} - \left(2\omega_{ie}^{\leftarrow n} + \omega_{en}^{\leftarrow n} \right) \times \mathbf{V}^{\leftarrow n} + \mathbf{g}^{\leftarrow n}, \quad (8)$$

where the superscript " $\leftarrow$ " denotes the backward process [23, 25].

The backward navigation process is just like the playback of the forward process. The carrier has the same attitude at the same sampling point t_k no matter in the forward or

backward process, which is $C_b^{\leftarrow n}(t_k) = C_b^n(t_k)$. Therefore, compared with the forward process, the backward process has converse angular velocity. As the attitude is the integration of angular rate, the angular rate could be obtained as $\omega_{ib}^{\leftarrow b}(t_k) = -\omega_{ib}^b(t_k)$, $\omega_{ie}^{\leftarrow n}(t_k) = -\omega_{ie}^n(t_k)$, and $\omega_{en}^{\leftarrow n}(t_k) = -\omega_{en}^n(t_k)$. Meanwhile, the system should have the same position in the backward process as the forward process at the same sampling point t_k , which means the velocity should be converse as $\mathbf{V}(t_k)_b = -\mathbf{V}^n(t_k)$, and the output of accelerometer should be $\mathbf{f}^{\leftarrow b}(t_k) = \mathbf{f}^b(t_k)$.

Some research studies deal with the output of the gyro and accelerometer similarly in the backward process as $\omega_{ib}^{\leftarrow b}(t_k) = -\omega_{ib}^b(t_k)$ and $\mathbf{f}^{\leftarrow b}(t_k) = -\mathbf{f}^b(t_k)$, which means the velocity in contiguous period of forward-backward navigation process is axisymmetric as $\mathbf{V}(t_k) = \mathbf{V}^n(t_k)$, but the position will change at the same sampling point. This approach has little effect on the stationary base, but the position error will have bad effect on the accuracy of the system with position changing or if the iterative process lasts for a long time.

3.2. Simulations for Iterative Navigation Process. The simulation is produced into two stages: (1) error-free sensor data is produced during time t_0 to t_n and saved with the ideal navigation parameters, and then the normal forward navigation process is performed; (2) the saved data are used to fulfil the forward or backward navigation of the forward-forward or forward-backward process. Velocity and attitude are changing during the simulation period, which are being updated using the saved measurement data through the whole process.

The attitude and velocity curves of one direction of the two iterative navigation solutions are shown in Figures 2 and 3.

As shown in Figures 2(a) and 3(a), the forward-forward navigation process is the repetition of the normal navigation process with the same saved data from t_0 to t_n . Each repetition should start with the same initial position and velocity, as well as the attitude. The carrier gets the same attitude and velocity during $t_0^i \sim t_n^i$ as those during $t_0 \sim t_n$.

Figures 2(b) and 3(b) show that, during the period $t_0^1 \sim t_n^1$ of the backward period, t_n^1 and t_0^1 are the start and end time, respectively, while t_0^1 is also the start time of the forward period $t_0^2 \sim t_n^2$. The attitude and velocity at t_n^i in the forward process is equal to t_n^{i+1} of the backward process, and the attitude and velocity at t_0^i in the backward process is equal to t_0^{i+1} in the forward process. Velocity and attitude at t_n^i are set as the initial parameters of t_n^{i+1} , and then the simulation seems to back to the origin and get the same attitude, velocity, and position as those at t_0 in the backward process.

It would be found from Figure 2(b) that the attitude curve of the whole forward-backward process is a coherent curve, while from Figure 3(b), the velocity at t_n^i and t_0^{i+1} are reversed. Different from the forward-backward process, forward-forward process is like combining several same processes.

FIGURE 1: Two kinds of iterative navigation solution methods: (a) forward-forward process and (b) forward-backward process.

FIGURE 2: Attitude curves. (a) Forward-forward process; (b) Forward-backward process.

FIGURE 3: Velocity curves. (a) Forward-forward process; (b) Forward-backward process.

4. The Drawback of Traditional Iterative Alignment Algorithm

The fine alignment methods such as compass method or Kalman filter are used during the navigation process to fulfil alignment for the normal system. Similarly, the fine alignment algorithm could also be added to the iterative process to fulfil the alignment. Different from the normal system using the measurement data in real-time, the forward-forward process is not a continuous process for attitude and velocity between different periods, while the forward-backward process needs to change the framework of the navigation algorithm compared to the normal process; as a result, the fine alignment method could not be used directly during the iterative process. Meanwhile, the iterative process will also cause other errors besides normal alignment errors.

4.1. Forward-Forward Process. During the forward-forward process, C_n^b will be updated in the first period $t_0^1 \sim t_n^1$ as usual, and $C_n^b(t_0^2)$ is calculated from $C_n^b(t_n^1)$ at time t_n^1 to run the forward process again. The DCM transferred from the end of period l to the start of period $l+1$ could be calculated as follows:

$$C_n^b(t_0^{l+1}) = C_n^b(t_0^{l+1}) = C_b^b(t_0^{l+1}) C_n^b(t_n^l) C_n^n(t_n^l) = C_b^b C_n^b(t_n^l) (C_n^n)^T, \quad (9)$$

where $C_b^{b_0}$ is the rotation matrix of the body frame from t_0 to t_n , $C_n^{n_0}$ is the rotation matrix of the navigation frame from t_0 to t_n . $C_b^{b_0}$ and $C_n^{n_0}$ are both constant matrices during different iterative periods, which just need to be calculated once during the whole alignment process using ω_{ib}^b and ω_{in}^n as follows:

$$\dot{C}_b^{b_0} = C_b^{b_0} [\omega_{ib}^b \times], \quad (10)$$

$$\dot{C}_n^{n_0} = C_n^{n_0} [\omega_{in}^n \times] = C_n^{n_0} [(\omega_{ie}^n + \omega_{en}^n) \times], \quad (11)$$

and in the period $t_0^l \sim t_n^l$, C_n^b is updated by the saved data based on equation (4).

The forward-forward process is not a continuous process for calculating the attitude matrix as it needs the transition based on equation (9) between every two periods. Influenced by the sensor bias and measurement errors, the output of gyro is not ideal, and $C_b^{b_0}$ will not be accurate as well, which will cause the transition error between every two periods. The velocity is also not continuous during the whole process as it will be given an initial value (such as 0 on the stationary base) at the start of every period, and as a result, the error of velocity needs to be accumulated from the initial value over again during each period. While the misalignment angles are estimated from the velocity error based on the system error model by the alignment method, this solution is not efficient enough.

4.2. Forward-Backward Process. From Section 3, the forward-backward process does not need the transition of C_n^b between different periods. The attitude at the end of the

forward process is just the attitude at the start of the next backward process, where there is $C_n^b(t_n^l) = C_n^b(t_n^{l+1})$. The attitude calculation of the forward-backward process is a continuous process, so the backward period can be considered as an extended process of the forward period. However, as the navigation algorithm is different during the backward process, the alignment algorithm also needs to be changed in the backward process, and the error equations need to be reanalysed [21].

In case of sensor constant errors, the normal output of gyro could be described as

$$\tilde{\omega}_{ib}^b = \omega_{ib}^b + \varepsilon^b. \quad (12)$$

While in the backward process, it will be

$$\tilde{\omega}_{ib}^b = -\tilde{\omega}_{ib}^b = -\omega_{ib}^b - \varepsilon^b. \quad (13)$$

We hope that the form of the errors in the iterative alignment process is similar to the form of the errors comprised in the saved data. The ideal form of the output of the gyro with converse angular rate but the same bias in the backward process should be $-\omega_{ib}^b + \varepsilon^b$. Thus, there will be $2\varepsilon^b$ between the real form and the ideal form as shown in Figure 4. For example, if the ideal gyro output without bias during the saving period is sinusoidal as curve 1 in Figure 4, the real output will be curve 2 due to the gyro bias. The form of the ideal gyro output during the backward process of alignment was thought to be sinusoidal too, which is represented by curve 3. According to equation (12), the form of the data in the backward process will be curve 5. Curve 4 shows the form of gyro output if more data is sampled as normal. Another thing is the disturbance caused by the oscillatory error and low-frequency noise of sensors in the backward process [20, 22]. These errors will have a bad effect on the accuracy of the initial alignment.

5. Iterative Alignment Method in the Inertial Frame

From Sections 3 and 4, the main problem of the forward-forward process is the transition error at the start of each period from equation (9). The forward-backward process needs different system model and has different error model in the backward period. Besides, the updating of attitude and velocity based on equations (4)–(8) needs to save all the measurement data of gyros and accelerometers during both forward and backward process.

The inertial frame stays the same during the initial alignment process, which means the DCM of the initial attitude $C_b^{n_0}$ is a constant matrix. Transition using equation (9) is not needed for $C_b^{n_0}$ during different iterative periods. If $C_b^{n_0}$ could be calculated precisely, the alignment process could be fulfilled by a simple transition.

5.1. Alignment in the Inertial Frame. We rewrite equation (5) in n_0 -frame as

FIGURE 4: Schematic diagram of gyro error for one process switch.

$$\dot{\mathbf{V}}^{n_0} = \mathbf{C}_b^{n_0} \mathbf{f}^b - (2\hat{\boldsymbol{\omega}}_{ie}^{n_0} + \boldsymbol{\omega}_{en_0}^{n_0}) \times \mathbf{V}^{n_0} + \mathbf{C}_n^{n_0} \mathbf{g}^n. \quad (14)$$

However, the calculated results will have some errors. “ $\sim$ ” is used to describe the calculated result, and equation (14) yields as follows:

$$\dot{\hat{\mathbf{V}}}^{n_0} = \hat{\mathbf{C}}_b^{n_0} \tilde{\mathbf{f}}^b - (2\hat{\boldsymbol{\omega}}_{ie}^{n_0} + \hat{\boldsymbol{\omega}}_{en_0}^{n_0}) \times \hat{\mathbf{V}}^{n_0} + \hat{\mathbf{C}}_n^{n_0} \mathbf{g}^n, \quad (15)$$

$$\tilde{\mathbf{f}}^b = \mathbf{f}^b + \nabla^b = \mathbf{f}^b + \nabla_c^b + \nabla_r^b, \quad (16)$$

$$\hat{\mathbf{C}}_b^{n_0} = (\mathbf{I}_{3 \times 3} - \boldsymbol{\phi}^{n_0} \times) \mathbf{C}_b^{n_0}, \quad (17)$$

where ∇_c^b is the constant bias and ∇_r^b is the random error. $\boldsymbol{\phi}^{n_0}$ is the misalignment angle in the inertial frame, which could be considered as a small angle after coarse alignment [26]. The attitude matrix $\hat{\mathbf{C}}_b^n$ could be calculated by

$$\hat{\mathbf{C}}_b^n = \hat{\mathbf{C}}_b^n \hat{\mathbf{C}}_b^{n_0}, \quad (18)$$

$\hat{\boldsymbol{\omega}}_{in_0}^{n_0} = 0$ as n_0 is an inertial frame, and $\hat{\boldsymbol{\omega}}_{en_0}^{n_0}$ could be calculated as follows:

$$\hat{\boldsymbol{\omega}}_{en_0}^{n_0} = \hat{\boldsymbol{\omega}}_{ei}^{n_0} + \hat{\boldsymbol{\omega}}_{in_0}^{n_0} = -\hat{\boldsymbol{\omega}}_{ie}^{n_0}. \quad (19)$$

As the period of the alignment process is short, the Earth rotation rate is constant, and the error of latitude is minor. $\hat{\boldsymbol{\omega}}_{ie}^n$ is considered accurate in the process, and $\hat{\mathbf{C}}_b^n$ is also accurate as it is updated by $\hat{\boldsymbol{\omega}}_{ie}^n$, which is $\hat{\mathbf{C}}_b^n \approx \mathbf{C}_b^n$. $\mathbf{C}_b^n$ could be described by a function of the time t_k as the rotation vector $\boldsymbol{\omega}_{ie}^n$ is constant. The quaternion form of $\mathbf{C}_b^n(t_k)$ could be calculated as follows:

$$\mathbf{Q}_b^n(t_k) = \cos\left(\frac{\boldsymbol{\omega}_{ie}^n t_k}{2}\right) + \frac{\vec{\mathbf{u}}}{|\vec{\mathbf{u}}|} \sin\left(\frac{\boldsymbol{\omega}_{ie}^n t_k}{2}\right), \quad (20)$$

where $\vec{\mathbf{u}} = [0 \ 1 \ \tan L]^T$.

Using equations (15) and (14) and neglecting the second-order small quantity, the velocity error can be obtained as follows:

$$\begin{aligned} \delta \dot{\mathbf{V}}^{n_0} &= \left[\hat{\mathbf{C}}_b^{n_0} \tilde{\mathbf{f}}^b \times \right] \boldsymbol{\phi}^i - \hat{\boldsymbol{\omega}}_{ie}^{n_0} \times \delta \mathbf{V}^{n_0} + \hat{\mathbf{C}}_b^{n_0} (\nabla_c^b + \nabla_r^b) \\ &= \left[\hat{\mathbf{C}}_b^{n_0} \tilde{\mathbf{f}}^b \times \right] \boldsymbol{\phi}^i - (\mathbf{C}_n^{n_0} \boldsymbol{\omega}_{ie}^n) \times \delta \mathbf{V}^{n_0} + \hat{\mathbf{C}}_b^{n_0} \tilde{\mathbf{C}}_b^{b_0} (\nabla_c^b + \nabla_r^b). \end{aligned} \quad (21)$$

As $\boldsymbol{\phi}^{n_0}$ is irrelevant with the Earth rotation and carrier motion, equation (2) could be rewritten in the i -frame as

$$\dot{\boldsymbol{\phi}}^{n_0} = -\hat{\mathbf{C}}_b^{n_0} \tilde{\mathbf{C}}_b^{b_0} \boldsymbol{\varepsilon}^b = -\hat{\mathbf{C}}_b^{n_0} \tilde{\mathbf{C}}_b^{b_0} (\boldsymbol{\varepsilon}_c^b + \boldsymbol{\varepsilon}_r^b), \quad (22)$$

where $\boldsymbol{\varepsilon}_c^b$ is the constant bias of the gyro and $\boldsymbol{\varepsilon}_r^b$ is the random error.

The update process can be performed with Kalman filter to estimate $\boldsymbol{\phi}^{n_0}$. Set the state variable $\mathbf{X} = [\boldsymbol{\phi}^{n_0} \ \delta \mathbf{V}^{n_0} \ \boldsymbol{\varepsilon}_c^b \ \nabla_c^b]$, and the system equation could be obtained from equations (21) and (22) as

$$\begin{aligned} \dot{\mathbf{X}} &= \begin{bmatrix} 0 & 0 & -\hat{\mathbf{C}}_b^{n_0} \tilde{\mathbf{C}}_b^{b_0} \tilde{\mathbf{f}}^b & 0 \\ \hat{\mathbf{C}}_b^{n_0} \tilde{\mathbf{C}}_b^{b_0} \tilde{\mathbf{f}}^b \times (\hat{\mathbf{C}}_n^{n_0} \boldsymbol{\omega}_{ie}^n) \times & 0 & \hat{\mathbf{C}}_b^{n_0} \tilde{\mathbf{C}}_b^{b_0} \\ 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{X} \\ &+ \begin{bmatrix} -\hat{\mathbf{C}}_b^{n_0} \tilde{\mathbf{C}}_b^{b_0} \boldsymbol{\varepsilon}_r^b \\ \hat{\mathbf{C}}_b^{n_0} \tilde{\mathbf{C}}_b^{b_0} \nabla_r^b \\ 0 \\ 0 \end{bmatrix}. \end{aligned} \quad (23)$$

The measurement variable $\mathbf{Z}$ is set as

$$\mathbf{Z} = \delta \mathbf{V}^{n_0} = [0 \ \mathbf{I}_{3 \times 3} \ 0 \ 0] \mathbf{X} + \mathbf{R}, \quad (24)$$

where $\mathbf{R}$ is the measurement noise. The velocity of the carrier in the n_0 -frame could be integrated from equation (15) as

$$\hat{\mathbf{V}}^{n_0} = \int \left(\hat{\mathbf{C}}_b^{n_0} \tilde{\mathbf{f}}^b - \hat{\boldsymbol{\omega}}_{ie}^{n_0} \times \hat{\mathbf{V}}^{n_0} + \hat{\mathbf{C}}_n^{n_0} \mathbf{g}^n \right) dt. \quad (25)$$

5.2. The Process of the Forward-Forward Alignment Method in the Inertial Frame. The data during normal time t_0 to t_n could be used to form the real-time coarse alignment, and the related data could be saved. Then, the forward-forward process with Kalman filter in the inertial frame is implemented to fulfil the fine alignment. At the start of each period, the Kalman filter needs to be initialized. Figure 5 shows the process of the whole alignment.

Different from normal forward-forward process in the n -frame, the original sensor data does not need to be saved with the method above. From equation (23), at sampling time t_k during the period l , the information needs to be updated based on the sensor data which is $\hat{\mathbf{C}}_b^{b_0}(t_k)$ and $\hat{\mathbf{C}}_b^{b_0}(t_k) \tilde{\mathbf{f}}^b(t_k)$, which stay the same during different periods. $\hat{\mathbf{C}}_b^{b_0}(t_k)$ could be transformed into quaternion form as $\hat{\mathbf{Q}}_b^{b_0}(t_k)$, which has 4 elements but including all attitude information. And $\hat{\mathbf{C}}_b^{b_0}(t_k) \tilde{\mathbf{f}}^b(t_k)$ has 3 elements, and thus the elements need to be saved are reduced. During the iterative process, the real-time output of the gyro needs to be used to calculate the rotation matrix $\mathbf{C}_b^{b_0}(t_p)$ of the b -frame through equation (10) during time t_n to t_p .

FIGURE 5: Process of the alignment based on forward-forward process in inertial frame.

For SINS without linear motion, $\delta \mathbf{V}^{n_0} = \hat{\mathbf{V}}^{n_0}$, which could be calculated from equation (25). ϕ^{n_0} is estimated by the Kalman filter. The attitude curves will converge after repeating the process for several times, and then the DCM of the real time t_p could be calculated as follows:

$$C_n^b(t_p) = C_b^{b(t_p)} C_n^{b(t_n)} \left(C_n^{n(t_p)} \right)^T, \quad (26)$$

where the quaternion form of $C_n^{n(t_p)}$ could be calculated as follows:

$$Q_n^{n(t_p)} = \cos\left(\frac{\omega_{ic}(t_p - t_n)}{2}\right) \frac{\vec{u}}{|\vec{u}|} \sin\left(\frac{\omega_{ic}(t_p - t_n)}{2}\right). \quad (27)$$

6. Analysis of the Method

In the normal process, the real attitude matrix $C_n^b(t_k)$ is changing all the time if the carrier is not stationary, which means $C_n^b(t_k)$ needs to be updated all the time during the iterative process. The b -frame $b(t_k)$ is changing with carrier rotation, and the n -frame $n(t_k)$ is changing with Earth rotation and carrier motion. As a result, the iterative navigation process in n -frame will need to save all the sensor data and the update the attitude with every saved data. If the sensor sampling rate is 200 Hz, the output of the gyro and accelerometer need to be saved at 200 Hz; meanwhile, the attitude and velocity during the iterative process need to be updated at 200 Hz. This will occupy a large amount of

storage and cost much more time for the iterative calculation.

Instead of calculating C_n^b directly, $C_{b_0}^{n_0}$ is estimated in the method from Section 5. The attitude in the inertial frame is steady during the iterative process as the motion and swing have no effect on the velocity and attitude of the inertial frame. The only parameter needs to be updated in the inertial frame during the iterative process is the velocity. If the solution frequency of Kalman filter during the forward-forward process is set as 1 Hz, the system information will be updated at 1 Hz. From equation (23), $\hat{C}_{b_0}^{b_0} \hat{f}^b$ and $\hat{C}_{b_0}^{b_0}$ are used during the alignment process, which need to be calculated and saved during the normal time from t_0 to t_n at 1 Hz, and $\hat{C}_{b_0}^{b_0}$ could also be transformed into quaternion form to save the storage space. By this, the data need to be saved is reduced by nearly 200 times compared to normal iterative process.

6.1. Simulation for Alignment in the Inertial Frame Based on Forward-Forward Process. The simulation is supposed to be carried out under a swing condition. The errors of sensor are listed in Table 1, and the swing parameters are listed in Table 2. The carrier is assumed without linear motion, and the initial latitude, longitude, and height are set as 32°, 118°, and 30 m, respectively. The sensor sampling rate is 200 Hz, while the Kalman Filter resolution cycle is set as 1 s. The simulation is carried out for 120 s, and the measurement data is generated and saved for alignment analysis. The initial attitude errors are set as 2.5°, 2.5°, and 4° for pitch, roll, and heading, respectively. The initial settings for the Kalman filter are given as follows:

$$\begin{aligned} X_0 &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T, \\ P_0 &= \text{diag}[5^\circ \ 5^\circ \ 10^\circ \ 1 \text{ m/s} \ 1 \text{ m/s} \ 1 \text{ m/s} \ 0.1^\circ/\text{h} \ 0.1^\circ/\text{h} \ 0.1^\circ/\text{h} \ 200 \mu\text{g} \ 200 \mu\text{g} \ 200 \mu\text{g}]^2, \\ R &= \text{diag}[0.1 \text{ m/s} \ 0.1 \text{ m/s} \ 0.1 \text{ m/s}]^2, \end{aligned} \quad (28)$$

where P_0 is the initial state covariance matrix. The obtained misalignment angles of attitude through the forward-forward alignment method in the inertial frame are shown in Figure 6, and statistic results of the misalignment angles are shown in Table 3.

It is clear to see in Figure 6 that the convergent estimation value of the attitude angles can be obtained within five times of the forward process with the saved IMU data. The pitch and roll misalignment angles could reach about $\pm 0.003^\circ$ in the second period of iterative process, and the

TABLE 1: Sensor errors.

	Gyro bias		Accelerometer error	
	Constant	Random	Constant	Random
x -axis	$0.02^\circ/\text{h}$	$0.01^\circ/\text{h}$	$50 \mu\text{g}$	$20 \mu\text{g}$
y -axis	$0.02^\circ/\text{h}$	$0.01^\circ/\text{h}$	$50 \mu\text{g}$	$20 \mu\text{g}$
z -axis	$0.02^\circ/\text{h}$	$0.01^\circ/\text{h}$	$50 \mu\text{g}$	$20 \mu\text{g}$

TABLE 2: Swing parameters.

	Pitch	Roll	heading
Amplitude ($^\circ$)	3	6	2
Frequency (Hz)	0.17	0.13	0.07

(a)

(b)

FIGURE 6: Continued.

FIGURE 6: Misalignment angle curves of forward-forward process: (a) pitch misalignment angle, (b) roll misalignment angle, and (c) heading misalignment angle.

TABLE 3: Statistic results of the misalignment angles.

	Pitch	Roll	heading
Calculated value (°)	-0.00277	0.00282	-0.0881
Theory value (°)	-0.00293	0.00293	-0.0897

heading misalignment angle converges to about 0.09° after two times of the iterative process. Although the heading angle needs nearly three times of the process to become convergent, the real time needed during the iterative process for the computer is quite little.

As the level angles are easy to converge during the alignment, more simulations are done for the heading angle to test the effect. The simulation is done with different lengths of the saved data and iterative times. The duration of data is set saving as T_s , and the number of times of the forward process needed during the alignment is M_f . The simulation chooses $T_s = 30$ s, 60 s, 100 s, 150 s, and 200 s and $M_f = 20, 10, 6, 4, 2$, respectively, where $T_s M_f = 600$ as the whole times of Kalman filter resolution period in the fine alignment process. The simulation condition is set as same as Tables 1 and 2. The curves of the heading misalignment angle with different T_s are shown in Figure 7.

From Figure 7, it could be found that $T_s = 30$ s and 60 s is too short for each process to converge. For $T_s \geq 100$ s, the alignment could fulfil in limited times of the process and has similar accuracy. Even though the saved data could be used for numerous times, the attitude will not converge in each period with too short saved data. In view of different alignment conditions, data lasting for about 100 s~200 s is usually chosen to be saved to fulfil the fine alignment.

6.2. Turntable Test. Considering engine vibration and angular motion of the carrier, the turntable test is executed to

FIGURE 7: Forward-forward process with different T_s .

verify the validity of the proposed method. The IMU used in this test is developed by Casic33s as shown in Figure 8; and fibre optic gyros and quartz accelerometers are used in this type of IMU, and the precision parameters of sensors are listed in Table 4. In this IMU, the updating frequency of sensor data is 200 Hz. The experimental environment consists of a turntable, IMU, computer, time-synchronization signal generator, data acquisition card, local area network, serial communication port, and so on. IMU is installed in the turntable with the x , y , and z axes coinciding with the axes of intermediate, inner, and outer frames, respectively.

A time-synchronization signal of 200 Hz is introduced to make the IMU data and turntable data synchronous. Under this experimental environment, once the time-synchronization signal is acting, the current angle information of turntable will be sent to the computer via serial

FIGURE 8: Turntable and IMU.

TABLE 4: Sensor errors.

Gyro		Accelerometer	
Bias	$<0.02^\circ/\text{h}$ (1σ)	Bias	$<5 \times 10^{-4} \text{ g}$
Repetitiveness of bias	$<0.01^\circ/\text{h}$ (1σ)	Repetitiveness of bias	$<2.5 \times 10^{-4} \text{ g}$ (1σ)
Random walk	$<0.006^\circ/\sqrt{\text{h}}$	Repetitiveness of scale factor	$<3.5 \times 10^{-4} \text{ g}$ (1σ)
Nonlinearity of scale factor	$\leq 50 \text{ ppm}$ (1σ)	Temperature coefficient of scale factor	$<6 \times 10^{-5} \text{ g}$ ($-40 \sim 40^\circ\text{C}$)

communication port and be saved at 200 Hz, and then the current sensor data and alignment results will be collected for alignment solution. With the data from turntable and IMU stored in the computer, evaluation can be executed on the alignment results. The algorithms are carried out based on VC++.

At first, the inner frame rotates to 6° , the intermediate frame rotates to 357° , and the outer frame rotates to 25° (these values can be selected randomly), which is set as the swing center. The swing amplitude and frequency of the turntable are set in Table 5.

The test lasts for 600 s, during which the measurement data of IMU and the attitude of turntable were saved. Three alignment schemes are compared: (1) iterative alignment with Kalman filter in the inertial frame (scheme A); (2) iterative alignment with Kalman filter in the navigation frame (scheme B); and (3) normal alignment process with Kalman filter in the navigation frame (scheme C). The Kalman filter resolution cycle is set as 1 s. 5 iterative navigation solution cycles are required to finish the alignment in scheme A and scheme B with each cycle using the saved data of 120 s. The calculated attitudes of the three schemes are

TABLE 5: Turntable swing parameters.

	Pitch	Roll	heading
Amplitude ($^\circ$)	4	6	4
Frequency (Hz)	0.125	0.1	0.1

compared with the attitude of the turntable, and then the misalignment angles are calculated. It should be noted that scheme C used 600 s to fulfil the process and update 600 times with Kalman filter, while scheme A and scheme B also update 600 times but use the data in the first 120 s for 5 times. The curves of misalignment angles are shown in Figure 9 and the RMS values calculated from the last 30 value of the estimation errors are shown in Table 6.

From Figure 9, it could be seen that scheme A and scheme B had oscillation during the iterative process, especially at the switching point. Figure 9(c) shows that during the first 120 s, scheme B and scheme C had the same result as they used the same algorithm, but scheme B is different since 120 s as it had a transition to fulfil the iterative process. For the pitch and roll angles, the three schemes had similar effect. While for the heading angle, scheme B is not stable. The period of 120 s is not

FIGURE 9: Turntable test. (a) Pitch misalignment angle. (b) Roll misalignment angle. (c) Heading misalignment angle.

TABLE 6: Statistic results of the misalignment angles.

	Scheme A	Scheme B	Scheme C
Pitch (°)	0.0038	0.0048	0.0042
Roll (°)	0.0019	0.0014	0.0024
Heading (°)	0.0031	0.0467	0.0019
Alignment time (s)	122	126	600

enough for the heading angle to converge under this condition with scheme B. Based on the comparison of Table 6, it can be seen that the alignment accuracy of scheme A could reach the same level as scheme C but with the least time. As scheme B needs to save all the IMU data during the first 120 s and use all the data to update the attitude at 1 Hz, the alignment time needed for scheme A is less than scheme B.

7. Conclusion

In this paper, iterative calculation methods, which include forward-forward method and forward-backward method, used in the SINS are discussed. Based on the saved sensor data, the iterative methods could improve the data utilization. For the iterative methods used to fulfil the alignment, the forward-forward process in the navigation frame is easy to carry out but it is also not efficient enough as it needs to deal with the

transition errors at the beginning of every period. The forward-backward process needs to change the system model and use different updating equations, which is hard to carry out, and the sensor errors as well as the oscillation errors will lead to accuracy degradation. The forward-forward process in the inertial frame is used to deal with these problems. Just like the forward-forward process in the navigation frame, this method uses the same updating equations during different periods. Meanwhile, the attitude calculation in the inertial frame is separated from the motion influence of the b -frame, which means no transition errors will be added during different periods. The data need to be saved and calculated is also projected to be much less. The simulation and the turntable test both indicate that the proposed method can fulfil the initial alignment effectively, and the alignment accuracy is not decreased.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The research was supported in part by the National Natural Science Foundation of China (51775110 and 61921004).

References

- [1] J. L. Weston and D. H. Titterton, "Modern inertial navigation technology and its application," *Electronics & Communication Engineering Journal*, vol. 12, no. 2, pp. 49–64, 2000.
- [2] Z. Xiong, Y. Hao, and F. Sun, "SINS rapid in-motion alignment based on nonlinear filter, position, location, & navigation symposium," in *Proceedings of the 2006 IEEE/ION Position, Location, And Navigation Symposium*, IEEE, Coronado, CA, USA, April 2006.
- [3] J. Tong, X. Xu, L. Hou, Y. Li, J. Wang, and L. Zhang, "An ultra-short baseline positioning model based on rotating array & reusing elements and its error analysis," *Sensors*, vol. 19, no. 20, p. 4373, 2019.
- [4] G. Hu, W. Wang, Y. Zhong, B. Gao, and C. Gu, "A new direct filtering approach to INS/GNSS integration," *Aerospace Science and Technology*, vol. 77, pp. 755–764, 2018.
- [5] C. Shen, Y. Zhang, J. Tang, H. Cao, and J. Liu, "Dual-optimization for a MEMS-INS/GPS system during GPS outages based on the cubature Kalman filter and neural networks," *Mechanical Systems and Signal Processing*, vol. 133, p. 106222, 2019.
- [6] T. Jinwu, X. Xiaosu, Z. Tao, Z. Liang, and L. Yao, "Study on installation error analysis and calibration of acoustic transceiver array based on SINS/USBL integrated system," *IEEE Access*, vol. 6, pp. 66923–66939, 2018.
- [7] R. M. Rogers, *Applied Mathematics in Integrated Navigation Systems*, American Institute of Aeronautics & Astronautics Inc, Reston, VA, USA, 2007.
- [8] X. Xu, X. Xu, T. Zhang, Y. Li, and J. Tong, "A kalman filter for SINS self-alignment based on vector observation," *Sensors*, vol. 17, no. 2, p. 264, 2017.
- [9] M. Liu, Y. Gao, G. Li, X. Guang, and S. Li, "An improved alignment method for the strapdown inertial navigation system (SINS)," *Sensors*, vol. 16, no. 5, p. 621, 2016.
- [10] X. Xu, X. Xu, T. Zhang, Y. Li, and Z. Wang, "A coarse alignment method based on digital filters and reconstructed observation vectors," *Sensors*, vol. 17, no. 4, p. 709, 2017.
- [11] W. G. S. Xinlong, "A fast and accurate initial alignment method for strapdown inertial navigation system on stationary base," *Journal of Control Theory and Applications*, vol. 3, pp. 145–149, 2005.
- [12] I. Y. Bar-Itzhack and N. Berman, "Control theoretic approach to inertial navigation systems," *Journal of Guidance Control & Dynamics*, vol. 10, pp. 1442–1453, 2012.
- [13] P. D. Groves, "Principles of GNSS, inertial, and multisensor integrated navigation systems," *Industrial Robot*, vol. 67, pp. 191–192, 2013.
- [14] G. Hu, L. Ni, B. Gao, X. Zhu, W. Wang, and Y. Zhong, "Model predictive based unscented kalman filter for hypersonic vehicle navigation with INS/GNSS integration," *IEEE Access*, vol. 8, pp. 4814–4823, 2020.
- [15] C. Shen, Y. Zhang, X. Guo et al., "Seamless GPS/inertial navigation system based on self-learning square-root cubature kalman filter," *IEEE Transactions on Industrial Electronics*, 2020.
- [16] L. I. Yao, X. U. Xiao-Su, and W. U. Bing-Xiang, "Gyro-compass self-alignment of SINS," *Journal of Chinese Inertial Technology*, vol. 16, no. 4, pp. 386–389, 2008.
- [17] X. Liu, X. Xu, Y. Liu, and L. Wang, "A fast and high-accuracy transfer alignment method between M/S INS for ship based on iterative calculation," *Measurement*, vol. 51, pp. 297–309, 2014.
- [18] W. Gao, Y. Ben, X. Zhang, Q. Li, and F. Yu, "Rapid fine strapdown INS alignment method under marine mooring condition," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 4, pp. 2887–2896, 2011.
- [19] T. Du, L. Guo, and J. Yang, "A fast initial alignment for SINS based on disturbance observer and Kalman filter," *Transactions of the Institute of Measurement and Control*, vol. 38, no. 10, pp. 1261–1269, 2016.
- [20] J. Li, J. Xu, L. Chang, and F. Zha, "An improved optimal method for initial alignment," *Journal of Navigation*, vol. 67, no. 4, pp. 727–736, 2014.
- [21] W. Gao, B. Lu, and C. Yu, "Forward and backward processes for INS compass alignment," *Ocean Engineering*, vol. 98, pp. 1–9, 2015.
- [22] X. Liu, X. Xu, L. Wang, and Y. Liu, "A fast compass alignment method for SINS based on saved data and repeated navigation solution," *Measurement*, vol. 46, no. 10, pp. 3836–3846, 2013.
- [23] X. Liu, X. Xu, Y. Liu, and L. Wang, "A rapid transfer alignment method for SINS based on the added backward-forward SINS resolution and data fusion," *Mathematical Problems in Engineering*, vol. 2013, Article ID 401794, 10 pages, 2013.
- [24] J. Sun, X.-S. Xu, Y.-T. Liu, T. Zhang, and Y. Li, "Initial alignment of large azimuth misalignment angles in SINS based on adaptive UPF," *Sensors*, vol. 15, no. 9, pp. 21807–21823, 2015.
- [25] G. Yan, W. Yan, and D. Xu, "On reverse navigation algorithm and its application to SINS gyro-compass in-movement alignment," in *Proceedings of the 27th Chinese Control Conference*, Kunming, China, July 2008.
- [26] Y. Y. Qin, *Inertial Navigation*, Beijing Science Press, Beijing, China, 2nd edition, 2014.
- [27] G. Hu, B. Gao, Y. Zhong, and C. Gu, "Unscented kalman filter with process noise covariance estimation for vehicular ins/gps integration system," *Inform Fusion*, vol. 64, 2020.
- [28] D. H. Titterton and J. L. Weston, *Strapdown Inertial Navigation Technology*, Institution of Electrical Engineers, London, UK, 2004.