applied sciences

MDPI

Article

In-Motion Coarse Alignment Method Based on Position Loci and Optimal-REQUEST for SINS

Haoqian Huang * and Jiaying Wei

College of Energy and Electrical Engineering, Hohai University, Nanjing 211100, China; xtcwjy@hhu.edu.cn
* Correspondence: hqhuang@hhu.edu.cn

Abstract: In this paper, an improved in-motion coarse alignment method is proposed for a strapdown inertial navigation system (SINS) using position loci obtained from the Global Positioning System (GPS). The difference from the popular coarse alignment methods is that the proposed algorithm uses GPS position loci information to form the vector observation, and does not need velocity information, which expands the application range of in-motion coarse alignment. In addition, this paper utilizes the Optimal-REQUEST algorithm to reduce the influence of random errors contained in the vector observation. The Optimal-REQUEST algorithm is an adaptive iterative updating algorithm, which can adaptively adjust the gain of the filter according to the loss function. Simulation results confirmed that the proposed algorithm can suppress the impact of random errors effectively. The pitch, roll and yaw angles calculated by the proposed algorithm were improved by 51.95%, 53.80% and 63.03% compared with the comparison algorithms.

Keywords: coarse alignment; strapdown inertial navigation system; attitude estimation; observation vector

check for updates

Citation: Huang, H.; Wei, J. In-Motion Coarse Alignment Method Based on Position Loci and Optimal-REQUEST for SINS. *Appl. Sci.* **2022**, *12*, 7113. https://doi.org/10.3390/app12147113

Academic Editor: Yutaka Ishibashi

Received: 28 May 2022
Accepted: 11 July 2022
Published: 14 July 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

A strapdown inertial navigation system (SINS) is an autonomous navigation system that does not rely on any external information. A SINS has the characteristics of excellent concealment because it does not radiate energy to the outside. Thus, it can work in a variety of complex environments, such as air, ground, and underwater [1–3]. In practical navigation applications, an integrated navigation system (INS) is usually used to improve navigation accuracy. An INS usually takes a SINS as the main navigation system and the Global Positioning System (GPS) or other systems, where errors do not accumulate with time, are used as auxiliary navigation systems [4–6]. Among many INSs, SINS/GPSs have attracted the attention of many researchers because of their wide application range and high efficiency [7]. In SINS/GPS research, initial alignment has been one of the significant focal areas [8].

Initial alignment is the process of calculating the initial attitude, position, and velocity of vehicles [9,10]. In a SINS/GPS integrated system, the initial position and velocity can be easily obtained by GPS. Therefore, the essence of initial alignment is to calculate the initial attitude. In general, initial alignment is divided into two stages: coarse alignment and fine alignment [11]. The coarse alignment period estimates and provides a rough attitude matrix for the fine alignment. Then, the fine alignment uses the liner model and filter technology based on the results of coarse alignment. Thus, the accuracy of coarse alignment greatly affects the accuracy of the whole integrated navigation process [12–14].

Several researchers have devoted themselves to devising new coarse alignment algorithms. In [15–17], the coarse alignment algorithms are studied based on gravitational apparent motion. The authors of [15] used gravitational apparent motion vectors at three moments to construct an attitude matrix and then to obtain latitude information by calculating the geometric relationship among the apparent motions. However, the method

proposed in [15] is easily affected by random noise contained in accelerometer measurement. However, when the measurement time interval is very short, or the measurement quantity is inaccurate, the apparent motion of gravity will be collinear. In order to avoid collinearity among apparent motions, the authors of [16] propose an online sensor data denoising method and study the design of a novel reconstruction method for apparent motion Through the reconstruction of vectors, the method can increase the accuracy of self-alignment. In addition, in [17], a fast self-alignment method with real-time estimated latitude to obtain high accuracy with unknown geographic latitude constraints and limited time is proposed. Unfortunately, the above three coarse alignment methods can only be applied to the static base or swaying base and limit the application range of a SINS. Therefore, researchers have begun to investigate in-motion coarse alignment methods based on an external equipment auxiliary.

When a vehicle is moving, the coarse alignment process cannot be completed using measurement information from inertial sensors alone, and it is necessary to engage the assistance of external equipment. GPS is the most popular auxiliary equipment for land vehicles, and many coarse alignment methods based on GPS have been studied in recent years. The authors of [18] investigated an optimization-based alignment (OBA) algorithm in an innovative way. The OBA algorithm uses infinite vector observations to transform the coarse alignment problem into an attitude determination problem. In [19], the approach reported in [18] is further developed by utilization of devised velocity integration formulae to construct the vector observation. Based on the above two algorithms, the basic problem of in-motion alignment can be solved. Unfortunately, both algorithms cannot suppress the impact of outliers contained in the outputs of GPS and inertial measurement units. Thus, in [20], a robust optimization-based alignment method to detect the outliers and improve the accuracy of in-motion coarse alignment is proposed. It is worth noting that the above in-motion coarse alignment methods are all based on the ground velocity. However, not all GPS receivers can export the ground velocity and some of them can only provide the position information of the vehicle. This limits the application of in-motion coarse alignment. Therefore, ref. [21] investigated an in-motion coarse alignment method for the SINS/GPS integrated system using position loci. The vector observation, based on position trajectory, further expands the application range of coarse alignment on the moving base. However, ref. [21] uses a filter-QUEST algorithm to calculate the attitude matrix, which has only limited ability to reduce the errors of vector observation contained in the inertial sensors and GPS measurement. Furthermore, the propagation noises are treated as fixed fading memory values in the filter-QUEST algorithm, which is similar to the REQUEST algorithm, and this makes both of the algorithms suboptimal.

To improve the performance of in-motion coarse alignment, this paper extends the research of [21] and proposes a new alignment method based on an Optimal-REQUEST algorithm and position loci (OP-PL). This paper employs a vector observation calculation method based on position loci. Then, the Optimal-REQUEST is utilized to suppress the noise in the measurement vectors and to filter the measurement noises by adaptively adjusting the fading memory value. Finally, the objective of improving the accuracy of coarse alignment is achieved. The contributions of the proposed in-motion coarse alignment are summarized as follows:

1. The proposed OP-PL utilizes position information to construct the observation vectors, which can expand the application range of the in-motion coarse alignment algorithm, because the proposed method only utilizes the position information which is available from every GPS receiver.
2. The outliers contained in the GPS positioning outputs can be suppressed as the outliers can be smoothed by GPS position loci; this can reduce the noise impact of outliers on observation vectors.
3. The proposed algorithm employs the Optimal-REQUEST algorithm to further suppress random noise contained in the observation vectors. The Optimal-REQUEST overcomes the defect of the REQUEST algorithm of the use of an empirical and con-

stant gain to filter the random noise. Therefore, the Optimal-REQUEST algorithm can optimally filter random noise using a cost function, and can improve the stability and accuracy of the coarse alignment.

The remainder of the paper is organized as follows: Section 2 introduces the traditional method of in-motion coarse alignment and defines the problem. Next, Section 3 elaborates the principle of vector observation based on the position loci, and Section 4 introduces the proposed Optimal-REQUEST algorithm. The simulation experiments conducted are described in Section 5. Finally, the conclusions and suggestions for future work are presented in Section 6. See Appendix A for the proof of some formulas.

2. Traditional Method of In-Motion Coarse Alignment and Related Work

In this section, the traditional method of in-motion coarse alignment is briefly described. Then, the limits and practical issues of constructing a vector using velocity measurement are introduced.

2.1. Traditional Construction Method Using Velocity Measurement

The core issue for in-motion coarse alignment is to find the initial attitude matrix C_b^n. According to the chain rule of the attitude matrix, the initial attitude matrix C_b^n can be divided into three matrices [11]:

$$\mathrm{C}_b^n(t) = \mathrm{C}_{n(0)}^{n(t)}\mathrm{C}_{b(0)}^{n(0)}\mathrm{C}_{b(t)}^{b(0)} \tag{1}$$

where n-frame denotes the navigation frame aligned with East-North-Up (ENU) geodetic axes, and b-frame denotes the body frame aligned with inertial measurement unit axes, namely right-forward-upper. $\mathrm{C}_{n(t)}^{n(0)}$ and $\mathrm{C}_{b(t)}^{b(0)}$ encode the attitude changes of the navigation frame and the body frame from the initial time to the current time, respectively. Using the attitude matrix differential equation, $\mathrm{C}_{n(t)}^{n(0)}$ and $\mathrm{C}_{b(t)}^{b(0)}$ can be calculated as follows:

$$\dot{\mathrm{C}}_{n(t)}^{n(0)} = \mathrm{C}_{n(t)}^{n(0)}\omega_{in}^{n}\times \tag{2}$$

$$\dot{\mathrm{C}}_{b(t)}^{b(0)} = \mathrm{C}_{b(t)}^{b(0)}\omega_{ib}^{b}\times \tag{3}$$

where i-frame denotes the Earth-center-fixed orthogonal reference frame, which does not rotate with the alignment process, ω_{in}^n denotes the rotation of n-frame relative to the i-frame, and ω_{ib}^b denotes the rotation of the b-frame relative to the i-frame. After the above calculation, the determination of C_b^n is translated into the determination of $\mathrm{C}_{b(0)}^{n(0)}$, which is a constant matrix during the in-motion coarse alignment.

2.2. Related Work

To calculate the matrix $\mathrm{C}_{b(0)}^{n(0)}$, ref. [18] proposes an algorithm using vector observation. Using the traditional vector observation method, we can acquire the measurement vectors and the reference vectors based on the velocity measurement. The specific force equation under the navigation frame is given by [18]:

$$\dot{\mathrm{v}}^n = \mathrm{C}_b^n\mathrm{f}^b - (2\omega_{ie}^n + \omega_{en}^n)\times \mathrm{v}^n + \mathrm{g}^n \tag{4}$$

Combining Equations (1)–(3), the following equation can be obtained:

$$\mathrm{C}_{b(0)}^{n(0)}\mathrm{C}_{b(t)}^{b(0)}\mathrm{f}^b = \mathrm{C}_{n(t)}^{n(0)}\dot{\mathrm{v}}^n + \mathrm{C}_{n(t)}^{n(0)}(2\omega_{ie}^n + \omega_{en}^n)\times \mathrm{v}^n - \mathrm{C}_{n(t)}^{n(0)}\mathrm{g}^n \tag{5}$$

Then, the reference vectors and the measurement vectors can be acquired, and the initial alignment problem is transformed into an attitude determination problem.

$$\mathrm{C}_{b(0)}^{n(0)}\alpha_v = \beta_v \tag{6}$$

where

$$\alpha_v = \int_0^t \mathrm{C}_{b(\tau)}^{b(0)}\mathrm{f}^b d\tau \tag{7}$$

$$\beta_v = \mathrm{C}_{n(t)}^{n(0)}\mathrm{v}^n - \mathrm{v}^n(0) + \int_0^t \mathrm{C}_{n(\tau)}^{n(0)}\omega_{ie}^n \times \mathrm{v}^n d\tau - \int_0^t \mathrm{C}_{n(\tau)}^{n(0)}\mathrm{g}^n d\tau \tag{8}$$

From Equation (8), it can be seen that the measurement vector β_v is calculated based on the ground velocity v^n; this requires that the GPS receiver can output the ground velocity. This construction method for the observation vector is the method reported in many current alignment studies [12–14]. However, some GPS receivers can only output the position of the vehicle, and the traditional construction method of vector observation is not suitable for in-motion coarse alignment in this situation. Furthermore, the noise contained in the GPS and inertial sensor measurements inevitably affects the accuracy of observation. Thus, it is necessary to change the construction method of vector observation and expand the application scope of the traditional OBA algorithms.

3. The Principle of Vector Observation Based on Position Loci

In this section, a construction method of vector observation based on position loci is derived, and the detailed recursive formulae are also investigated.

3.1. Position Integration Formula

Based on the principles of inertial navigation, the position and the velocity under the navigation frame are shown as:

$$\dot{\mathrm{p}}^n = \mathrm{R}_c\mathrm{v}^n \tag{9}$$

where $\mathrm{p}^n = \begin{bmatrix} L & \lambda & h \end{bmatrix}^{\mathrm{T}}$; L, λ, h denote the vehicle's latitude, longitude, and height, respectively. The local curvature matrix R_c is defined below.

$$\mathrm{R}_c = \begin{bmatrix} 0 & \frac{1}{R_M+h} & 0 \\ \frac{\sec L}{R_N+h} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{10}$$

where R_M and R_N denote the meridian radius and the transverse radius of curvature based on the WGS-84 reference ellipsoid, respectively.

As is well-known, the expression of position integration can be obtained by integrating the velocity. Combining with Equation (9), the specific position integration can be calculated as:

$$\mathrm{r}^n = \int_0^t \mathrm{v}^n d\tau = \int_0^t \mathrm{R}_c^{-1}\dot{\mathrm{p}}^n d\tau \tag{11}$$

where $\mathrm{r}^n(0) = 0$.

Based on the relation between position and velocity, Equations (6)–(8) can be reconstructed through integral calculation, as follows:

$$\mathrm{C}_{b0}^{n0}\int_0^t \alpha_v d\tau = \int_0^t \beta_v d\tau \tag{12}$$

The integral calculation formulae on both sides of the equation are shown below [22]:

$$\alpha_p = \int_0^t \alpha_v d\tau = \int_0^t \int_0^\tau \mathrm{C}_{b(\sigma)}^{b0}\mathrm{f}^b d\sigma d\tau \tag{13}$$

$$\beta_p = \int_0^t \mathrm{C}_{n(\tau)}^{n0}\dot{\mathrm{r}}^n d\tau - t\mathrm{v}^n(0) + \int_0^t \int_0^\tau \mathrm{C}_{n(\sigma)}^{n0}(\omega_{ie}^n\times)\dot{\mathrm{r}}^n d\sigma d\tau - \int_0^t \int_0^\tau \mathrm{C}_{n(\sigma)}^{n0}\mathrm{g}^n d\sigma d\tau \quad (14)$$

Comparing Equation (14) with Equation (8), we notice that the measurement vectors are constructed by the GPS position information of the vehicle. Moreover, the effects of outliers, coming from the GPS position information, can be weakened through position-loci integration. Thus, the proposed algorithm is more robust than the traditional algorithms.

3.2. Discrete Recursive Formulae

In the previous subsection, we obtained the vector construction method based on position information. However, all the formulae are continuous and cannot be calculated in practical experiments. Thus, the discrete recursive formulae are studied in this subsection [21].

The reference vector α_p can be approximated as below:

$$\alpha_p(M) = \sum_{k=0}^{M-1}\Delta t_G \sum_{m=0}^{k-1}\mathrm{C}_{b(t_m)}^{b0}\int_{t_m}^{t_{m+1}}\mathrm{C}_{b(\sigma)}^{b(t_m)}\mathrm{f}^b d\sigma + \sum_{k=0}^{M-1}\mathrm{C}_{b(t_k)}^{b0}\int_{t_k}^{t_{k+1}}\int_{t_k}^{\tau}\mathrm{C}_{b(\sigma)}^{b(t_k)}\mathrm{f}^b d\sigma d\tau \quad (15)$$

where Δt_G and M represent the sampling time of GPS and the current time, respectively. Specifically, $t = t_M = M\Delta t_G$, $t_k = k\Delta t_G$ and $t_m = m\Delta t_G$. The integral term can be calculated by the two-sample method, which is as follows:

$$\int_{t_m}^{t_{m+1}}\mathrm{C}_{b(\sigma)}^{b(t_m)}\mathrm{f}^b d\sigma = \Delta\mathrm{v}_1 + \Delta\mathrm{v}_2 + \frac{1}{2}(\Delta\theta_1 + \Delta\theta_2)\times(\Delta\mathrm{v}_1 + \Delta\mathrm{v}_2) + \frac{2}{3}(\Delta\theta_1\times\Delta\mathrm{v}_2 + \Delta\mathrm{v}_1\times\Delta\theta_2) \quad (16)$$

$$\int_{t_k}^{t_{k+1}}\int_{t_k}^{\tau}\mathrm{C}_{b(\sigma)}^{b(t_k)}\mathrm{f}^b d\sigma d\tau = \frac{\Delta t_G}{30}(25\Delta\mathrm{v}_1 + 5\Delta\mathrm{v}_2 + 12\Delta\theta_1\times\Delta\mathrm{v}_1 + 8\Delta\theta_1\times\Delta\mathrm{v}_2 + 2\Delta\mathrm{v}_1\times\Delta\theta_1 + 2\Delta\theta_2\times\Delta\mathrm{v}_2) \quad (17)$$

where

$$\begin{cases} \Delta\mathrm{v}_1 = \sum\limits_{l=1}^{\frac{D}{2}}\mathrm{f}_l^b\Delta t_S & \Delta\mathrm{v}_2 = \sum\limits_{l=\frac{D}{2}+1}^{D}\mathrm{f}_l^b\Delta t_S \\ \Delta\theta_1 = \sum\limits_{l=1}^{\frac{D}{2}}\omega_{ib,l}^b\Delta t_S & \Delta\theta_2 = \sum\limits_{l=\frac{D}{2}+1}^{D}\omega_{ib,l}^b\Delta t_S \end{cases} \quad (18)$$

where D denotes the half sampling time of GPS, and Δt_s denotes the sampling time of the accelerometers and gyroscopes.

After acquiring the recursive formulae of the reference vector, the measurement vector is derived in detail. The first integration term of Equation (14) is shown below:

$$\int_0^t \mathrm{C}_{n(\tau)}^{n0}\dot{\mathrm{r}}^n d\tau = \mathrm{C}_{n(t)}^{n0}\mathrm{r}^n - \int_0^t \mathrm{C}_{n(\tau)}^{n0}(\omega_{in}^n\times)\mathrm{r}^n d\tau = \mathrm{C}_{n(t_M)}^{n0}\mathrm{r}^n - \sum_{k=0}^{M-1}\mathrm{C}_{n(t_k)}^{n0}\int_{t_k}^{t_{k+1}}\mathrm{C}_{n(\tau)}^{n(t_k)}(\omega_{in}^n\times)\mathrm{r}^n d\tau \quad (19)$$

The second integration term can be given by:

$$\begin{aligned} &\int_0^t\int_0^\tau \mathrm{C}_{n(\sigma)}^{n0}(\omega_{ie}^n\times)\dot{\mathrm{r}}^n d\sigma d\tau = \int_0^t \mathrm{C}_{n(\tau)}^{n0}(\omega_{ie}^n\times)\mathrm{r}^n d\tau - \int_0^t\int_0^\tau \mathrm{C}_{n(\sigma)}^{n0}(\omega_{in}^n\times)(\omega_{ie}^n\times)\mathrm{r}^n d\sigma d\tau \\ &= \sum_{k=0}^{M-1}\mathrm{C}_{n(t_k)}^{n0}\int_{t_k}^{t_{k+1}}\mathrm{C}_{n(\tau)}^{n(t_k)}(\omega_{ie}^n\times)\mathrm{r}^n d\tau - \sum_{k=0}^{M-1}\Delta t_G\sum_{m=0}^{k-1}\mathrm{C}_{n(t_m)}^{n0}\int_{t_m}^{t_{m+1}}\mathrm{C}_{n(\tau)}^{n(t_m)}(\omega_{in}^n\times)(\omega_{ie}^n\times)\mathrm{r}^n d\tau \\ &- \sum_{k=0}^{M-1}\mathrm{C}_{n(t_k)}^{n0}\int_{t_k}^{t_{k+1}}\int_{t_k}^{\tau}\mathrm{C}_{n(\sigma)}^{n(t_k)}(\omega_{in}^n\times)(\omega_{ie}^n\times)\mathrm{r}^n d\sigma d\tau \end{aligned} \quad (20)$$

The last integration term is calculated as follows:

$$\begin{aligned}
&\int_0^t \int_0^\tau \mathrm{C}_{n(\sigma)}^{n0}\mathrm{g}^n d\sigma d\tau = \sum_{k=0}^{M-1}\int_{t_k}^{t_{k+1}} \int_0^\tau \mathrm{C}_{n(\sigma)}^{n0}\mathrm{g}^n d\sigma d\tau = \sum_{k=0}^{M-1}\int_{t_k}^{t_{k+1}} \left(\int_0^{t_k} \mathrm{C}_{n(\sigma)}^{n0}\mathrm{g}^n d\sigma + \int_{t_k}^\tau \mathrm{C}_{n(\sigma)}^{n0}\mathrm{g}^n d\sigma\right) d\tau \\
&= \sum_{k=0}^{M-1}\Delta t_G \int_0^{t_k} \mathrm{C}_{n(\sigma)}^{n0}\mathrm{g}^n d\sigma + \sum_{k=0}^{M-1}\int_{t_k}^{t_{k+1}} \int_{t_k}^\tau \mathrm{C}_{n(\sigma)}^{n0}\mathrm{g}^n d\sigma d\tau \\
&= \sum_{k=0}^{M-1}\Delta t_G \sum_{m=0}^{k-1} \mathrm{C}_{n(t_m)}^{n0}\int_{t_m}^{t_{m+1}} \mathrm{C}_{n(\sigma)}^{n(t_m)}\mathrm{g}^n d\sigma + \sum_{k=0}^{M-1} \mathrm{C}_{n(t_k)}^{n0}\int_{t_k}^{t_{k+1}} \int_{t_k}^\tau \mathrm{C}_{n(\sigma)}^{n(t_k)}\mathrm{g}^n d\sigma d\tau
\end{aligned} \tag{21}$$

Substituting Equations (19)–(21) into Equation (14), the complete formula for the measurement vector is shown below:

$$\begin{aligned}
\beta_p &= \mathrm{C}_{n(t_M)}^{n0}\mathrm{r}^n - \sum_{k=0}^{M-1} \mathrm{C}_{n(t_k)}^{n0}\int_{t_k}^{t_{k+1}} \mathrm{C}_{n(\tau)}^{n(t_k)}(\omega_{in}^n\times)\mathrm{r}^n d\tau - t\mathrm{v}^n(0) + \sum_{k=0}^{M-1} \mathrm{C}_{n(t_k)}^{n0}\int_{t_k}^{t_{k+1}} \mathrm{C}_{n(\tau)}^{n(t_k)}(\omega_{ie}^n\times)\mathrm{r}^n d\tau \\
&- \sum_{k=0}^{M-1}\Delta t_G \sum_{m=0}^{k-1} \mathrm{C}_{n(t_m)}^{n0}\int_{t_m}^{t_{m+1}} \mathrm{C}_{n(\tau)}^{n(t_m)}(\omega_{in}^n\times)(\omega_{ie}^n\times)\mathrm{r}^n d\tau - \sum_{k=0}^{M-1} \mathrm{C}_{n(t_k)}^{n0}\int_{t_k}^{t_{k+1}} \int_{t_k}^\tau \mathrm{C}_{n(\sigma)}^{n(t_k)}(\omega_{in}^n\times)(\omega_{ie}^n\times)\mathrm{r}^n d\sigma d\tau \\
&- \sum_{k=0}^{M-1}\Delta t_G \sum_{m=0}^{k-1} \mathrm{C}_{n(t_m)}^{0}\int_{t_m}^{t_{m+1}} \mathrm{C}_{n(\sigma)}^{n(t_m)}\mathrm{g}^n d\sigma - \sum_{k=0}^{M-1} \mathrm{C}_{n(t_k)}^{0}\int_{t_k}^{t_{k+1}} \int_{t_k}^\tau \mathrm{C}_{n(\sigma)}^{n(t_k)}\mathrm{g}^n d\sigma d\tau
\end{aligned} \tag{22}$$

Assuming that the terms related to $(\omega_{ie}^n\times)$ and $(\omega_{in}^n\times)$ are so small compared with other terms in Equation (22) that they can be ignored during calculation, then the measurement vectors can be approximately determined as:

$$\beta_p = \mathrm{C}_{n(t_M)}^{n0}\mathrm{r}^n - t\mathrm{v}^n(0) - \sum_{k=0}^{M-1}\Delta t_G \sum_{m=0}^{k-1} \mathrm{C}_{n(t_m)}^{n0}\int_{t_m}^{t_{m+1}} \mathrm{C}_{n(\sigma)}^{n(t_m)}\mathrm{g}^n d\sigma - \sum_{k=0}^{M-1} \mathrm{C}_{n(t_k)}^{n0}\int_{t_k}^{t_{k+1}} \int_{t_k}^\tau \mathrm{C}_{n(\sigma)}^{n(t_k)}\mathrm{g}^n d\sigma d\tau \tag{23}$$

Meanwhile, the attitude matrix $\mathrm{C}_{n(t)}^{n(t_k)}$ can be determined approximately as follows:

$$\mathrm{C}_{n(t)}^{n(t_k)} = \mathrm{I} + \frac{\sin\|\varphi_n\|}{\|\varphi_n\|}(\varphi_n\times) + \frac{1-\cos\|\varphi_n\|}{\|\varphi_n\|^2}(\varphi_n\times)^2 \approx \mathrm{I} + (\varphi_n\times) \tag{24}$$

where $\varphi_n \approx \int_{t_k}^t \omega_{ie}^n d\tau \approx (t-t_k)\omega_{in}^n$. The reason for this approximation is that ω_{in}^n slowly changes during the coarse alignment. Further discretizing the Equation (23), we obtain the following formulae.

$$\begin{aligned}
\int_{t_m}^{t_{m+1}} \mathrm{C}_{n(\sigma)}^{n(t_m)}\mathrm{g}^n d\sigma &\approx \int_{t_m}^{t_{m+1}} [\mathrm{I} + (\varphi_n\times)]\mathrm{g}^n d\sigma = \int_{t_m}^{t_{m+1}} \left[\mathrm{I} + (\sigma - t_m)(\omega_{in}^n\times)\right]\mathrm{g}^n d\sigma \\
&\approx \left(\Delta t_G \mathrm{I} + \tfrac{\Delta t_G^2}{2}(\omega_{in}^n\times)\right)\mathrm{g}^n
\end{aligned} \tag{25}$$

$$\begin{aligned}
\int_{t_k}^{t_{k+1}} \int_{t_k}^\tau \mathrm{C}_{n(\sigma)}^{n(t_k)}\mathrm{g}^n d\sigma d\tau &\approx \int_{t_k}^{t_{k+1}} \int_{t_k}^\tau [\mathrm{I} + (\varphi_n\times)]\mathrm{g}^n d\sigma d\tau = \int_{t_k}^{t_{k+1}} \int_{t_k}^\tau \left[\mathrm{I} + (\sigma - t_m)(\omega_{in}^n\times)\right]\mathrm{g}^n d\sigma d\tau \\
&\approx \left(\tfrac{\Delta t_G^2}{2}\mathrm{I} + \tfrac{\Delta t_G^3}{6}(\omega_{in}^n\times)\right)\mathrm{g}^n
\end{aligned} \tag{26}$$

Finally, we acquire the recursive calculation process for the observation vector based on the formulae described:

$$\beta_p \approx \mathrm{C}_{n(t_M)}^{n0}\mathrm{r}^n - t\mathrm{v}^n(0) - \sum_{k=0}^{M-1}\Delta t_G \sum_{m=0}^{k-1} \mathrm{C}_{n(t_m)}^{n0}\left(\Delta t_G \mathrm{I} + \frac{\Delta t_G^2}{2}(\omega_{in}^n\times)\right)\mathrm{g}^n - \sum_{k=0}^{M-1} \mathrm{C}_{n(t_k)}^{n0}\left(\frac{\Delta t_G^2}{2}\mathrm{I} + \frac{\Delta t_G^3}{6}(\omega_{in}^n\times)\right)\mathrm{g}^n \tag{27}$$

$$\mathrm{r}^n \approx \sum_{k=1}^{M} \mathrm{R}_c^{-1}(k)(\mathrm{p}_k^n - \mathrm{p}_{k-1}^n) \tag{28}$$

where p_k^n represents the position information of the vehicle at time instant k, and $R_c^{-1}(k)$ is the inverse matrix of Equation (10). It is worth noting that the position information can be acquired through GPS directly.

4. Attitude Determination Using the Optimal-REQUEST Algorithm

Based on the above calculations, models of the observation vector are acquired. However, the results of coarse alignment are inevitably affected by GPS measurement errors in the process of constructing vectors. Thus, an Optimal-REQUEST algorithm, based on position loci (OR-PL), is proposed to reduce the impact of errors and to determine the initial attitude matrix $C_{b(0)}^{n(0)}$. The Optimal-REQUEST algorithm utilizes the measurement vectors and reference vectors to construct a K-matrix. Moreover, an iteration algorithm, which can adjust the gain according to the measurement error, is applied to update the K-matrix. The optimal attitude quaternion can then be obtained by calculating the K-matrix eigenvector corresponding to its largest eigenvalue.

4.1. Measurement Update

When the measurement vectors and reference vectors are obtained based on the position loci, the initial attitude matrix $C_{b(0)}^{n(0)}$ can be determined by minimizing the following formula:

$$\frac{1}{2}\sum_{i=1}^{n} a_i \|b_i - Ar_i\|^2 = 1 - q^T K q \tag{29}$$

where $b_i = \frac{\beta_p}{\|\beta_p\|}$, $r_i = \frac{\alpha_p}{\|\alpha_p\|}$ and $\sum_{i=1}^{n} a_i = 1$. The matrix A is the initial attitude matrix $C_{b(0)}^{n(0)}$. The construction method of the K-matrix is shown below:

$$K_{k/k} = \begin{bmatrix} S_k - \sigma_k I_3 & z_k \\ z_k^T & \sigma_k \end{bmatrix} \tag{30}$$

where the parameters in Equation (30) are defined as follows:

$$B_k \triangleq \sum_{i=1}^{n} a_i b_i r_i^T,\ S_k \triangleq B_k + B_k^T \tag{31}$$

$$z_k \triangleq \sum_{i=1}^{n} a_i b_i \times r_i,\ \sigma_k \triangleq tr(B_k) \tag{32}$$

According to the REQUEST algorithm, the propagation formula of the K-matrix is given by:

$$K_{k+1/k} = \Phi_k K_{k/k} \Phi_k^T \tag{33}$$

During the coarse alignment, the initial attitude matrix $C_{b(0)}^{n(0)}$ stays constant. Thus, the propagation coefficient matrix $\Phi_k = I_4$.

When the new observation vectors are obtained at time $k + 1$, we construct the corresponding incremental K-matrix, namely δK_{k+1}. The specific construction method is the same as Equations (30)–(32), and the only difference is that the vectors acquired at time $k + 1$ are used to calculate δK_{k+1}.

Next, the estimated $K_{k+1/k+1}$ can be updated by the combination of prior estimation $K_{k+1/k}$ and the incremental matrix δK_{k+1} at time $k + 1$, namely,

$$K_{k+1/k+1} = (1 - \rho_{k+1}) \frac{m_k}{m_{k+1}} K_{k+1/k} + \rho_{k+1} \frac{\delta m_{k+1}}{m_{k+1}} \delta K_{k+1} \tag{34}$$

where m_k, m_{k+1} and δm_{k+1} are scalar coefficients, and the specific relationship of the three scalars is as follows:

$$m_{k+1} = (1 - \rho_{k+1})m_k + \rho_{k+1}\delta m_{k+1} \tag{35}$$

The fading memory factor ρ_{k+1} is the core of the REQUEST algorithm and the Optimal-REQUEST algorithm. The REQUEST algorithm uses the fading memory concept to treat the measurement noise, and this makes the algorithm suboptimal. However, the proposed OP-PL algorithm can optimally filter the measurement noise by calculating the value of the fading memory factor adaptively.

4.2. Stochastic Models

To deduce the error models of the proposed OP-PL algorithm accurately, we assume that the reference vector r_{k+1} is error-free and that the measurement vector b_{k+1} is corrupted by the noisy vector $\delta\mathrm{b}_{k+1}$.

$$\mathrm{b}_{k+1} = \mathrm{A}_{k+1}\mathrm{r}_{k+1} + \delta\mathrm{b}_{k+1} \tag{36}$$

Define the following noise matrix:

$$\mathrm{V}_{k+1} = \begin{bmatrix} \mathrm{S}_b - \sigma_b\mathrm{I}_3 & \mathrm{z}_b \\ \mathrm{z}_b^{\mathrm{T}} & \sigma_b \end{bmatrix} \tag{37}$$

where the parameters in Equation (37) are defined as:

$$\mathrm{B}_b \triangleq \sum_{i=1}^{n} a_i\delta\mathrm{b}_{k+1}(\mathrm{r}_{k+1})^{\mathrm{T}},\ \mathrm{S}_b \triangleq \mathrm{B}_b + \mathrm{B}_b^{\mathrm{T}} \tag{38}$$

$$\mathrm{z}_b \triangleq \sum_{i=1}^{n} a_i\delta\mathrm{b}_{k+1} \times \mathrm{r}_{k+1},\ \sigma_b \triangleq tr(\mathrm{B}_b) \tag{39}$$

It is worth noting that V_{k+1} is made up of $\delta\mathrm{b}_{k+1}$ and r_{k+1}, and $\delta\mathrm{b}_{k+1}$ is random. Thus, we conclude that V_{k+1} is also random. The linear relation of V_{k+1} in $\delta\mathrm{b}_{k+1}$ is used to deduce a stochastic model for V_{k+1} and $\delta\mathrm{b}_{k+1}$. The relationship between $\delta\mathrm{K}_{k+1}$ and V_{k+1} is given by:

$$\delta\mathrm{K}_{k+1} = \delta\mathrm{K}_{k+1}^{o} + \mathrm{V}_{k+1} \tag{40}$$

where $\delta\mathrm{K}_{k+1}^{o}$ denotes the error-free matrix.

The mean and the covariance of $\delta\mathrm{b}_k$ are constructed as follows:

$$E[\delta\mathrm{b}_k] = 0 \tag{41}$$

$$E\left[\delta\mathrm{b}_k\delta\mathrm{b}_{k+i}^{\mathrm{T}}\right] = \mu_k(\mathrm{I}_3 - \mathrm{b}_k\mathrm{b}_{k+i}^{\mathrm{T}})\delta_{k,k+i} \tag{42}$$

where $\delta_{k,k+i}$ represents the Kronecker delta function and μ_k represents the variance of the component of $\delta\mathrm{b}_k$. It is obvious that $\delta\mathrm{b}_k$ is white noise with zero-mean. Thus, V_k also has the characteristics of zero-mean white noise, namely,

$$E\left[\mathrm{V}_k\mathrm{V}_{k+i}^{\mathrm{T}}\right] = 0 \tag{43}$$

where $i \neq 0$. To further derive the uncertainty of the matrix, we introduce a special form of uncertainty, namely P_{xx}. The proof of P_{xx} is shown in Appendix A.

The P-matrix of V_k is given by:

$$\Re_k = E\left[\mathrm{V}_k\mathrm{V}_k^{\mathrm{T}}\right] = \begin{bmatrix} \Re_{11} & \Re_{12} \\ \Re_{21} & \Re_{22} \end{bmatrix} \tag{44}$$

The calculation formulae of each element in the matrix are shown below and the detailed derivation process is shown in [23].

$$\Re_{11} = \frac{\mu_k}{n_k}\sum_{i=1}^{n_k}\left[3-(\mathrm{r}_i^{\mathrm{T}}\mathrm{b}_i)^2\right]\mathrm{I}_3+(\mathrm{b}_i^{\mathrm{T}}\mathrm{r}_i)(\mathrm{b}_i\mathrm{r}_i^{\mathrm{T}}+\mathrm{r}_i\mathrm{b}_i^{\mathrm{T}})+(\mathrm{r}_i\times)(\mathrm{b}_i\mathrm{b}_i^{\mathrm{T}})(\mathrm{r}_i\times)^{\mathrm{T}} \tag{45}$$

$$\Re_{12} = 0 \tag{46}$$

$$\Re_{21} = 0^{\mathrm{T}} \tag{47}$$

$$\Re_{22} = \frac{2\mu_k}{n_k} \tag{48}$$

where μ_k is the variance for all the n_k simultaneous observations calculated at time t_k.

4.3. Calculation of Optimal Gain

After the measurement update formulae and error models are deduced, the cost function can be determined through the estimated errors of the K-matrix. Next, the fading memory factor ρ_{k+1} can be updated adaptively according to the cost function.

We define the estimation errors of the K-matrix as:

$$\Delta\mathrm{K}_{k+1/k} \triangleq \mathrm{K}^o_{k+1/k} - \mathrm{K}_{k+1/k} \tag{49}$$

$$\Delta\mathrm{K}_{k+1/k+1} \triangleq \mathrm{K}^o_{k+1/k+1} - \mathrm{K}_{k+1/k+1} \tag{50}$$

where $\mathrm{K}^o_{k+1/k}$ and $\mathrm{K}^o_{k+1/k+1}$ denote the a priori and the a posteriori estimation with error-free, respectively. $\Delta\mathrm{K}_{k+1/k}$ and $\Delta\mathrm{K}_{k+1/k+1}$ denote the errors of the a priori and the a posteriori estimation.

Similarly to Equation (34), reconstruct the equation with the error-free K-matrix, namely:

$$\mathrm{K}^o_{k+1/k+1} = (1-\rho_{k+1})\frac{m_k}{m_{k+1}}\mathrm{K}^o_{k+1/k} + \rho_{k+1}\frac{\delta m_{k+1}}{m_{k+1}}\delta\mathrm{K}^o_{k+1} \tag{51}$$

After subtracting Equations (34) and (51), substitute Equations (40), (49) and (50) into the following formula:

$$\Delta\mathrm{K}_{k+1/k+1} = (1-\rho_{k+1})\frac{m_k}{m_{k+1}}\Delta\mathrm{K}_{k+1/k} + \rho_{k+1}\frac{\delta m_{k+1}}{m_{k+1}}\mathrm{V}_{k+1} \tag{52}$$

The P matrix of $\Delta\mathrm{K}_{k+1/k}$ and $\Delta\mathrm{K}_{k+1/k+1}$ are defined as below

$$\mathrm{P}_{k+1/k} \triangleq E\left[\Delta\mathrm{K}_{k+1/k}\Delta\mathrm{K}^{\mathrm{T}}_{k+1/k}\right] \tag{53}$$

$$\mathrm{P}_{k+1/k+1} \triangleq E\left[\Delta\mathrm{K}_{k+1/k+1}\Delta\mathrm{K}^{\mathrm{T}}_{k+1/k+1}\right] \tag{54}$$

Considering Equation (52), the following formula can be calculated as:

$$\begin{aligned}\Delta\mathrm{K}_{k+1/k+1}\Delta\mathrm{K}^{\mathrm{T}}_{k+1/k+1} &= \left[(1-\rho_{k+1})\tfrac{m_k}{m_{k+1}}\right]^2\Delta\mathrm{K}_{k+1/k}\Delta\mathrm{K}^{\mathrm{T}}_{k+1/k}\\ -(1-\rho_{k+1})\rho_{k+1}\tfrac{m_k\delta m_{k+1}}{m^2_{k+1}}&(\Delta\mathrm{K}_{k+1/k}\mathrm{V}^{\mathrm{T}}_{k+1}+\mathrm{V}_{k+1}\Delta\mathrm{K}^{\mathrm{T}}_{k+1/k}) + (\rho_{k+1}\tfrac{\delta m_{k+1}}{m_{k+1}})^2\mathrm{V}_{k+1}\mathrm{V}^{\mathrm{T}}_{k+1}\end{aligned} \tag{55}$$

As mentioned, $\delta\mathrm{b}_k$ and V_k represent the white noise process with zero-mean. Moreover, $\Delta\mathrm{K}_{k+1/k}$ consists of V_i, where i is an integer from 1 to k. Thus, $\Delta\mathrm{K}_{k+1/k}$ has no relation with V_{k+1}, namely,

$$E\left[\Delta\mathrm{K}_{k+1/k}\mathrm{V}^{\mathrm{T}}_{k+1}\right] = 0 \tag{56}$$

Then, we take the expectation of Equation (55), and simplify the following formulae:

$$\mathrm{P}_{k+1/k+1} \triangleq E\left[\Delta\mathrm{K}_{k+1/k+1}\Delta\mathrm{K}_{k+1/k+1}^{\mathrm{T}}\right] = \left[(1-\rho_{k+1})\frac{m_k}{m_{k+1}}\right]^2 E\left[\Delta\mathrm{K}_{k+1/k}\Delta\mathrm{K}_{k+1/k}^{\mathrm{T}}\right] + (\rho_{k+1}\frac{\delta m_{k+1}}{m_{k+1}})^2 E\left[\mathrm{V}_{k+1}\mathrm{V}_{k+1}^{\mathrm{T}}\right] \quad (57)$$

$$\mathrm{P}_{k+1/k+1} = \left[(1-\rho_{k+1})\frac{m_k}{m_{k+1}}\right]^2 \mathrm{P}_{k+1/k} + (\rho_{k+1}\frac{\delta m_{k+1}}{m_{k+1}})^2 \Re_{k+1} \quad (58)$$

Based on Equation (58), we define the cost function J_{k+1}:

$$J_{k+1} = tr(E\left[\Delta\mathrm{K}_{k+1/k+1}\Delta\mathrm{K}_{k+1/k+1}^{\mathrm{T}}\right]) = tr(\mathrm{P}_{k+1/k+1}) \quad (59)$$

The calculation of the fading memory factor is then transformed by solving the minimization problem of Equation (59). We use the derivation to obtain the expression of the fading memory factor.

$$J_{k+1}(\rho_{k+1}) = \left[(1-\rho_{k+1})\frac{m_k}{m_{k+1}}\right]^2 tr(\mathrm{P}_{k+1/k}) + (\rho_{k+1}\frac{\delta m_{k+1}}{m_{k+1}})^2 tr(\Re_{k+1}) \quad (60)$$

$$\frac{dJ_{k+1}}{d\rho_{k+1}} = 2\left[\left(\frac{m_k}{m_{k+1}}\right)^2 tr(\mathrm{P}_{k+1/k}) + \left(\frac{\delta m_{k+1}}{m_{k+1}}\right)^2 tr(\Re_{k+1})\right]\rho_{k+1} - 2\left(\frac{m_k}{m_{k+1}}\right)^2 tr(\mathrm{P}_{k+1/k}) = 0 \quad (61)$$

Finally, the stationary point ρ_{k+1}^* can be determined using the extremum of J_{k+1}.

$$\rho_{k+1}^* = \frac{m_k^2 tr(\mathrm{P}_{k/k})}{m_k^2 tr(\mathrm{P}_{k/k}) + \delta m_{k+1}^2 tr(\Re_{k+1})} \quad (62)$$

4.4. Algorithm Summary

Compared with the REQUEST algorithm, the OP-PL algorithm uses the updated estimation error to define the special cost function and calculates the scalar gain ρ_{k+1}^*. Thus, the OP-PL can better reduce the impact of GPS measurement errors which are contained in the observation vectors. For clear explanation, the whole in-motion coarse alignment procedure is shown in Table 1 and Figure 1.

Table 1. In-motion coarse alignment based on the OP-PL algorithm.

Initialization	$\boldsymbol{k = 0}$, $\boldsymbol{m_0 = \delta m_0}$, $\boldsymbol{\rho_0^* = 1}$, $\mathbf{C}_{\boldsymbol{n(0)}}^{\boldsymbol{n0}} = \mathbf{C}_{\boldsymbol{b(0)}}^{\boldsymbol{b0}} = \mathbf{I}_3$, $\mathbf{K}_{0/0} = \delta\mathbf{K}_0$, $\mathbf{P}_{0/0} = \Re_0$
Step 1	$k = k + 1$
Step 2	Update $\mathrm{C}_{n(t)}^{n(0)}$ and $\mathrm{C}_{b(t)}^{b(0)}$ using Equations (2) and (3)
Step 3	Calculate and normalize α_p using Equations (15)–(18)
Step 4	Calculate and normalize β_p using Equations (27) and (28)
Step 5	Calculate $\delta\mathrm{K}_k$ and $\Re_k$ using Equations (30)–(32) and (44)–(48)
Step 6	Calculate ρ_k^* and m_k using Equations (62) and (35)
Step 7	Update $\mathrm{K}_{k/k}$ using Equation (34)
Step 8	Calculate the attitude matrix $\mathrm{C}_{b(0)}^{n(0)}$
Step 9	Calculate the attitude matrix $\mathrm{C}_{b(k)}^{n(k)}$
Step 10	Back to the Step 1 until the end

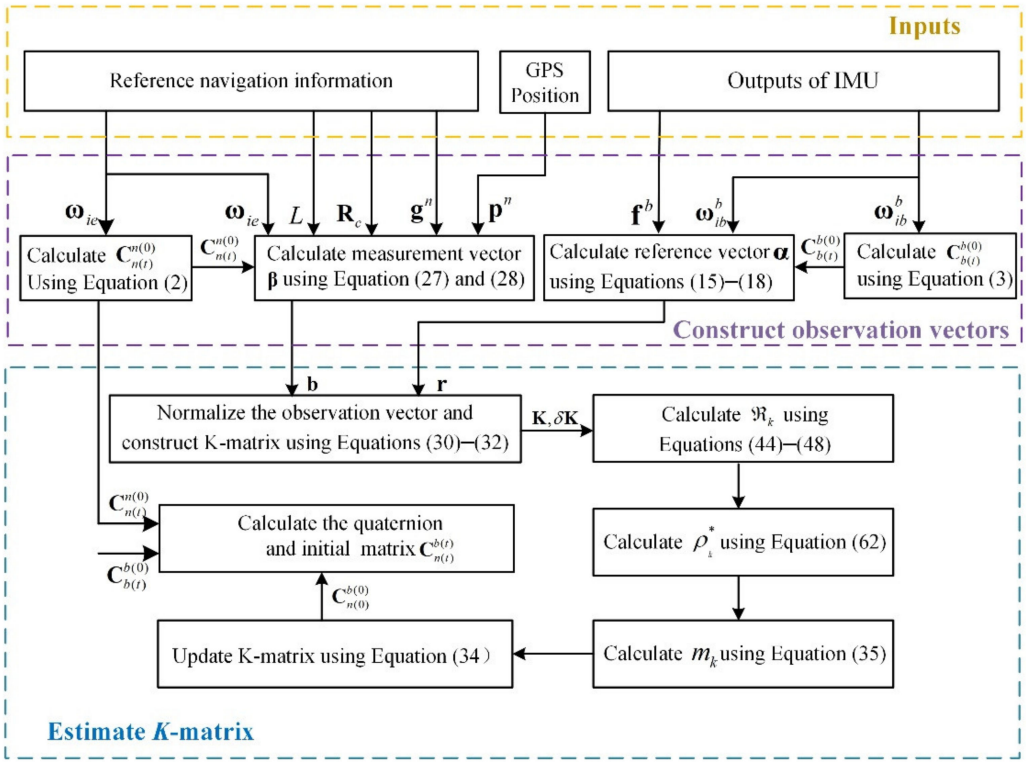


Figure 1. In-motion coarse alignment procedure.

5. Simulation

The simulation tests carried out to verify the accuracy and rapidity of the proposed OP-PL algorithm are described in this section. The trajectory of the vehicle was designed to follow a zigzag movement, and the actual information of the vehicle was known. Thus, the actual angles were used as reference angles to calculate the errors of three angles. The zigzag trajectory and the reference information are shown in Figures 2 and 3, respectively. It can be seen that the outliers, which corrupt the measurement of GPS, were added to the measurement. The output frequencies of GPS and the simulated inertial measurement unit (IMU) were set as 5 Hz and 200 Hz, respectively. The constant bias and random noise of the gyroscopes and accelerometers are listed in Table 2. The initial latitude and longitude were set as 32.00° and 118.00°, respectively. Throughout the in-motion coarse alignment process, it was assumed that the vehicle was moving on the horizontal plane; the initial values of the attitude angles were all set as 0°. The simulation was conducted using Matlab 2016b with an alignment time of 300 s. The proposed OP-PL algorithm was compared with the QUEST algorithm and the REQUEST algorithm to verify performance.

Figures 4–6 show the different performances of the REQUEST algorithm and the proposed OP-PL algorithm. We chose three different values of the fading memory factor, which were 0.1, 0.01 and 0.001, to compare with the optimal adaptive gain ρ^*. Figure 7 shows the optimal gain of the proposed algorithm in the simulation. Considering the results from three angles, we found that the REQUEST algorithm was unstable during simulation. From Figures 4 and 5, it is evident that the REQUEST algorithm showed similar performance to the proposed algorithm when 0.1 was chosen as the value of the fading memory factor. The 'MEAN' and 'RMSE' represent the mean and the root mean square error of the attitude errors. The MEAN and RMSE errors of the blue and black curves were calculated by collecting data from 150 s to 300 s, where the blue curve represents $\rho = 0.1$ and the black curve represents the optimal adaptively gain. The pitch MEAN errors of the blue and black curves were 0.0269° and −0.0760°, respectively. The pitch RMSE errors of the blue and black curves were 0.0425° and 0.0824°, respectively. Although the accuracy of the pitch angle calculated by the proposed OP-PL algorithm was lower than the REQUEST,

the convergence rate of the pitch angle calculated by the proposed OP-PL algorithm was half that calculated by the REQUEST algorithm.

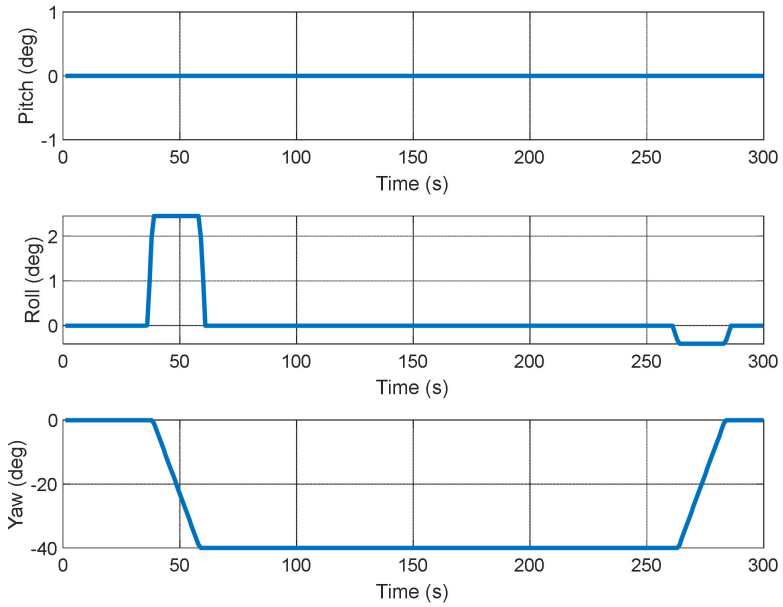


Figure 2. Reference angles.

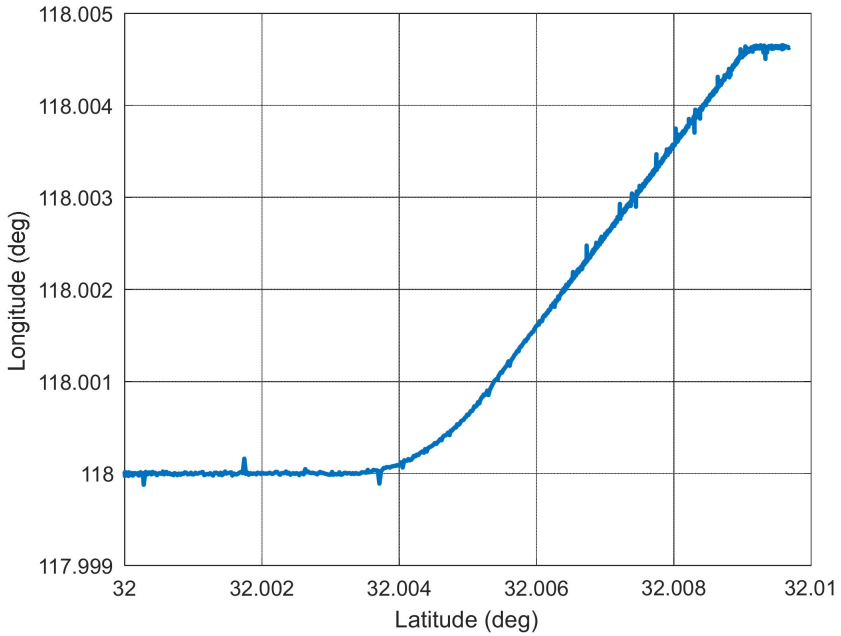


Figure 3. Reference trajectory.

Table 2. Parameter setting of inertial measurement units.

Sensors	Constant Bias	Random Noise
Gyroscopes	0.01 deg/h	0.005 deg/$\sqrt{\text{h}}$
Accelerometer	100 μg	100 μg/$\sqrt{\text{Hz}}$

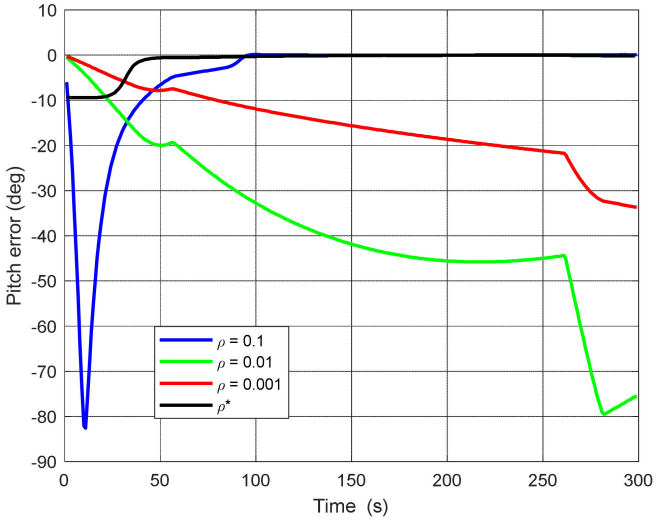


Figure 4. Pitch comparisons of different ρ.

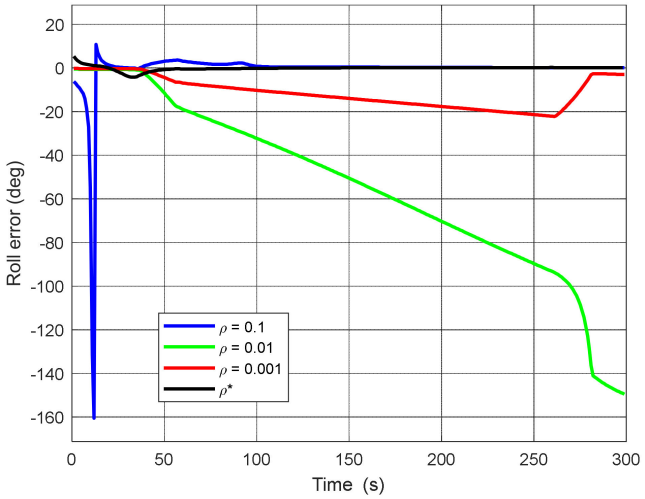


Figure 5. Roll comparisons of different ρ.

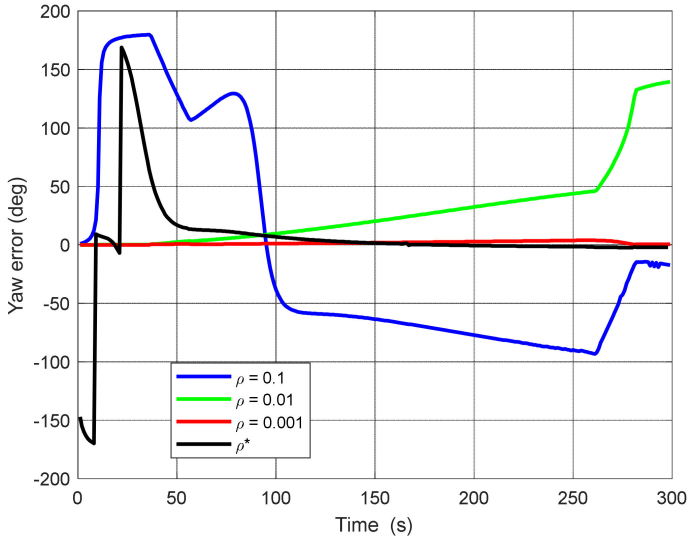


Figure 6. Yaw comparisons of different ρ.

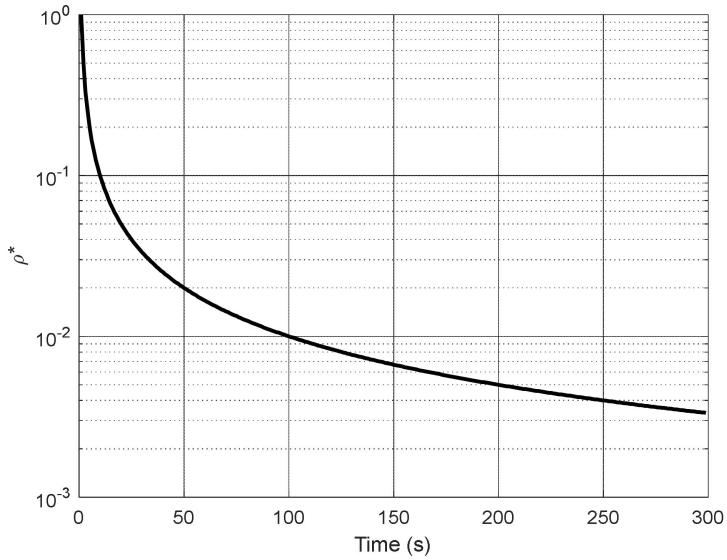


Figure 7. The curve of optimal ρ^*.

Moreover, the roll MEAN errors of the blue and black curves were 0.1875° and 0.1159°, respectively. The roll RMSE errors of the blue and black curves were 0.2114° and 0.1210°, respectively. From the RMSE data, it is evident that the proposed OP-PL algorithm performed better than the REQUEST algorithm for the calculation of the roll angle. It can also be seen from Figure 5 that the proposed OP-PL algorithm was very stable and that the convergence speed was still very fast.

However, when $\rho = 0.1$, the performance of the yaw angle was very poor. Although this value can produce excellent results in calculating the roll angle and the pitch angle, it was completely wrong for calculating the yaw angle. This indicates that the fixed gain was unreliable during the coarse alignment. When the value of ρ was selected as 0.1, the new K-matrix (δK_{k+1}) accounted for a relatively large proportion during the update of the K-matrix. This made it impossible to calculate the yaw angle correctly. If the value of ρ was decreased to 0.001, more error effects were accumulated because the old K-matrix ($K_{k+1/k}$) accounted for a relatively large proportion during the update of the K-matrix. This made it difficult to calculate the pitch and roll angles correctly.

To overcome the drawbacks of the REQUEST algorithm, the proposed algorithm uses a cost function to adjust the fading memory value adaptively. From Figure 7, we can see that, the value is selected as one at the beginning of the filter. Thus, more parts of the new measurements can be used during the update stage to speed up convergence. With increase in alignment time, a relatively low weight is assigned to the new measurement to ensure that the alignment process is steady.

To further verify the superiority of the proposed OP-PL algorithm, we chose a third algorithm for comparison. Based on the obtained simulation results, we chose $\rho = 0.001$ as the fading memory factor in the REQUEST algorithm. Figures 8–10 show the calculation results of the REQUEST, QUEST and the proposed OP-PL algorithm, which are represented by the yellow, green and brown curves, respectively. To better compare the errors among the three algorithms, we list the mean values and root mean square errors of the angle error values in Table 3, and present a box plot which is shown in Figure 11.

In Figures 8 and 9, the performance of the REQUEST algorithm can be seen to be still poor compared with the QUEST and the proposed OP-PL algorithm. The reason for this situation is that the REQUEST is a suboptimal algorithm, and it is hard to find an appropriate value to keep the algorithm stable. Figure 11 confirms the poor performance of REQUEST, where the three angle errors were obviously larger than for the other two algorithms. The MEANs and RMSEs of the pitch and roll angles, which were calculated by the REQUEST algorithm, were also the worst among the three algorithms.

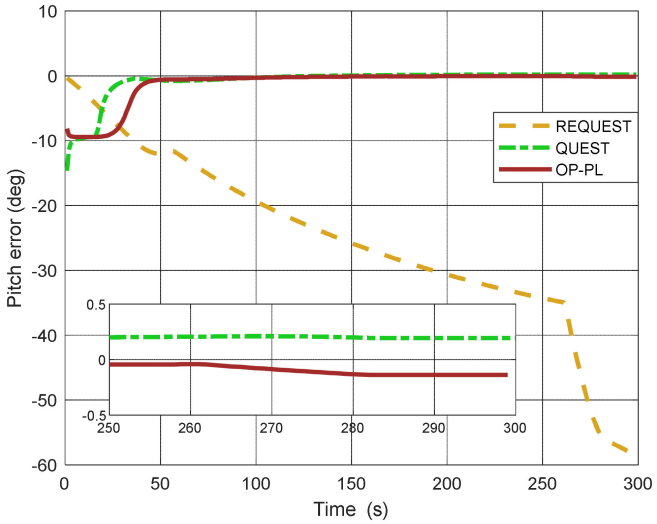


Figure 8. Curves of pitch angle errors.

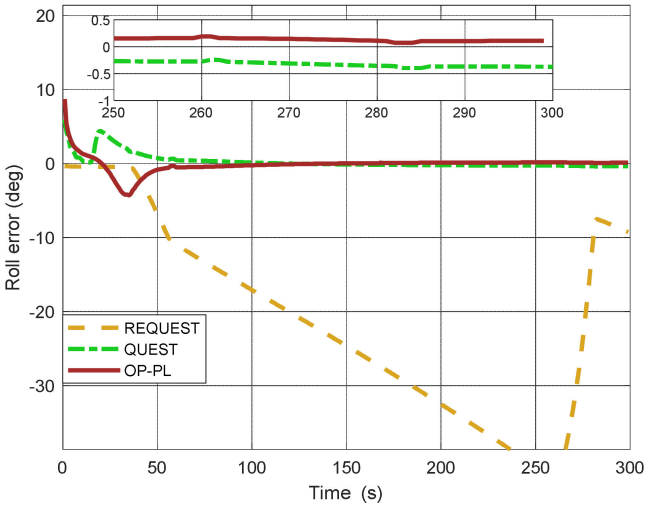


Figure 9. Curves of roll angle errors.

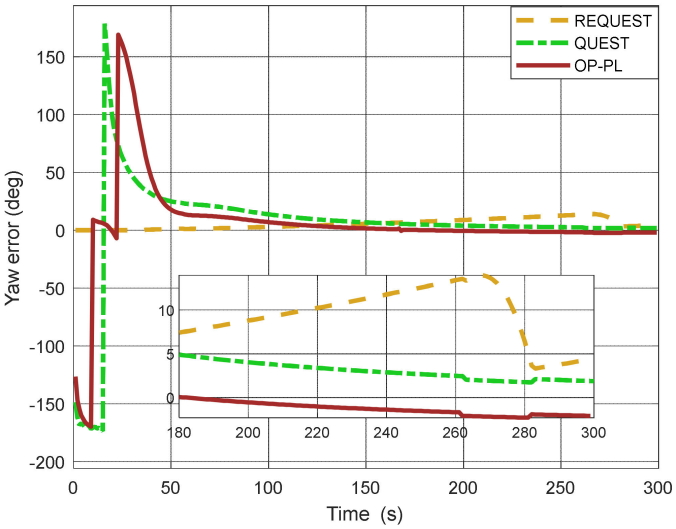


Figure 10. Curves of yaw angle errors.

Table 3. Error statistics of three algorithms in simulation test.

Items			REQUEST	QUEST	OP-PL
Pitch (deg)	1~150 s	MEAN	−14.7921	−1.7156	−2.3169
		RMSE	16.4321	3.5995	4.1962
	150~300 s	MEAN	−36.1971	0.1662	−0.0760
		RMSE	37.5654	0.1715	0.0824
Roll (deg)	1~150 s	MEAN	−11.6315	1.8145	−0.3420
		RMSE	14.3065	12.6798	1.5843
	150~300 s	MEAN	−29.8454	−0.2526	0.1159
		RMSE	31.4920	0.2619	0.1210
Yaw (deg)	1~150 s	MEAN	2.0683	5.7750	10.8685
		RMSE	2.6878	63.6506	57.2915
	150~300 s	MEAN	8.9417	3.5183	−0.9181
		RMSE	9.4477	3.7871	1.4000

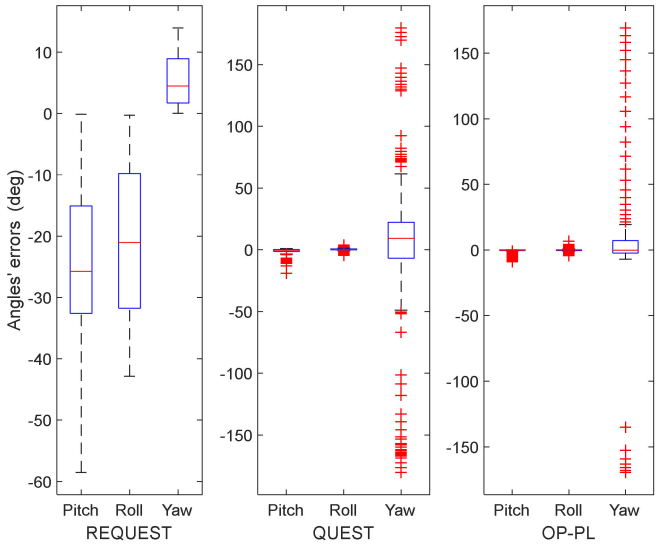


Figure 11. Box plot of angle errors.

Moreover, the OP-PL algorithm retains the merits of the REQUEST algorithm, as it is a time-varying recursive attitude quaternion estimator which preserves the quaternion unit-norm property. Compared with the QUEST algorithm, the OP-PL algorithm has an enhanced ability to reduce the impact of the measurement noise contained in the measurement vectors.

From Figures 8–10, we can see that, although the convergence speed of the OP-PL algorithm was the same as that of the QUEST algorithm, the proposed algorithm had higher accuracy and the accuracy of the three angles were all improved. Moreover, as can be seen from Figure 11, although the outliers of the yaw angle calculated by OP-PL were similar to that calculated by the QUEST algorithm, the proposed algorithm was relatively stable, based on the calculation results from the three angles. The main cause of the outliers in the yaw angle was unstable performance from 0 to 50 s. After the algorithm became stable, the advantages of the proposed algorithm were still evident. The pitch angle, roll angle and yaw angle calculated by the OP-PL algorithm were reduced by approximately 51.95%, 53.80% and 63.03%, respectively, compared with the QUEST algorithm.

Therefore, the above analyses confirm that the OP-PL algorithm had better alignment accuracy than the comparison algorithms.

6. Conclusions and Future Work

This paper proposes an improved in-motion coarse alignment method based on position loci and an Optimal-REQUEST algorithm. The proposed algorithm only uses position information to construct the observation vector, which greatly expands the application scope of the coarse alignment algorithm because all GPS receivers can output the position information. Moreover, the effects of random noise contained in the vector observation are suppressed by the Optimal-REQUEST algorithm which can choose the filter gain adaptively. The results of simulation tests showed that the proposed OP-PL algorithm converged faster and had better accuracy than the comparison algorithms. The pitch angle, roll angle and yaw angle calculated by the OP-PL algorithm were reduced by approximately 51.95%, 53.80% and 63.03%, respectively. However, our work still has some limitations and areas that can be improved. In future work, we plan to verify our proposed algorithm in real vehicle experiments and to extend the application scope of the algorithm to underwater environments.

Author Contributions: Conceptualization, J.W.; Supervision, H.H.; Writing—original draft, J.W.; Writing—review & editing, H.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Natural Science Foundation of Jiangsu Province (BK20221500), the National Natural Science Foundation of China (61703098), and the Fujian Provincial Key Laboratory of Coast and Island Management Technology Study (FJCIMTS2019-03). This work is also supported in part by London Tech Bridge, the APEX Undersea Challenge, the European Regional Development Fund—Industrial Intensive Innovation Programme, the Research Fund from the Science and Technology on Underwater Vehicle Technology Laboratory (2021JCJQ-SYSJJ-LB06905), and the Water Science and Technology Project of Jiangsu Province under grant (2021072, 2021063).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

The definition of P_{xx} is shown below:

$$\mathrm{P}_{xx} = E\left[xx^{\mathrm{T}}\right] \tag{A1}$$

Taking a two-dimensional matrix as an example, the specific calculation method of P_{xx} is:

$$\mathrm{P}_{xx} = \begin{bmatrix} E\left[x_{11}^2\right] + E\left[x_{12}^2\right] & E[x_{11}x_{12}] + E[x_{12}x_{22}] \\ E[x_{11}x_{12}] + E[x_{12}x_{22}] & E\left[x_{12}^2\right] + E\left[x_{22}^2\right] \end{bmatrix} \tag{A2}$$

It is obvious that P_{xx} is a symmetric matrix with positive semi-definite. Considering the vector $\mathrm{X} = \begin{bmatrix} x_{11} & x_{12} & x_{21} & x_{22} \end{bmatrix}^{\mathrm{T}}$ with zero-mean, its covariance matrix is calculated as follows:

$$\mathrm{cov}(\mathrm{X}) \triangleq E\left[\mathrm{X}\mathrm{X}^{\mathrm{T}}\right] \tag{A3}$$

Then, we find that the trace of P_{xx} is equal to the trace of $\mathrm{cov}(\mathrm{X})$, namely, $tr(\mathrm{P}_{xx}) = tr(\mathrm{cov}(\mathrm{X}))$. This definition of the P_{xx} matrix expresses the uncertainty of the matrix in a more compact and convenient form. Therefore, we use this definition to evaluate the uncertainty of V_k.

References

1. Titterton, D.; Weston, J.L. *Strapdown Inertial Navigation Technology*, 2nd ed.; IET: London, UK, 2004.
2. Huang, H.; Tang, J.; Liu, C.; Zhang, B.; Wang, B. Variational Bayesian-Based Filter for Inaccurate Input in Underwater Navigation. *IEEE Trans. Veh. Technol.* **2021**, *70*, 8441–8452. [CrossRef]
3. Wang, D.; Xu, X.; Yang, Y.; Zhang, T. A Quasi-Newton Quaternions Calibration Method for DVL Error Aided GNSS. *IEEE Trans. Veh. Technol.* **2021**, *70*, 2465–2477. [CrossRef]
4. Abosekeen, A.; Iqbal, U.; Noureldin, A.; Korenberg, M.J. A Novel Multi-Level Integrated Navigation System for Challenging GNSS Environments. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 4838–4852. [CrossRef]
5. Huang, H.; Tang, J.; Song, R.; Tang, X. A novel matrix block algorithm based on cubature transformation fusing variational Bayesian scheme for position estimation applied to MEMS navigation system. *Mech. Syst. Signal Process.* **2022**, *166*, 108486. [CrossRef]
6. Liu, Y.; Fan, X.; Lv, C.; Wu, J.; Li, L.; Ding, D. An innovative information fusion method with adaptive Kalman filter for integrated INS/GPS navigation of autonomous vehicles. *Mech. Syst. Signal Process.* **2018**, *100*, 605–616. [CrossRef]
7. Wang, J.W.; Chen, X.Y.; Yang, P. Adaptive H-infinite Kalman Filter based on Multiple Fading Factors and its Application in Unmanned Underwater Vehicle. *ISA Trans.* **2021**, *108*, 295–304. [CrossRef]
8. Lin, Y.; Miao, L.; Zhou, Z. An Improved MCMC-Based Particle Filter for GPS-Aided SINS In-Motion Initial Alignment. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 7895–7905. [CrossRef]
9. Xu, J.N.; He, H.Y.; Qin, F.J. A Novel Autonomous Initial Alignment Method for Strapdown Inertial Navigation System. *IEEE Trans. Instrum. Meas.* **2017**, *66*, 2274–2282. [CrossRef]
10. Chang, L.; Li, Y.; Xue, B. Initial Alignment for a Doppler Velocity Log-Aided Strapdown Inertial Navigation System with Limited Information. *IEEE/ASME Trans. Mechatron.* **2017**, *22*, 329–338. [CrossRef]
11. Huang, H.; Wei, J.; Jin, C.; Tang, J. An Improved Initial Alignment Method using Kalman Filtering of the Vectorized K-matrix. In Proceedings of the 2021 International Conference on Sensing, Measurement & Data Analytics in the Era of Artificial Intelligence (ICSMD), Nanjing, China, 21–23 October 2021; pp. 1–5.
12. Zhang, T.; Zhu, Y.; Xu, X.; Wang, J.; Li, Y. In-Motion Coarse Alignment Based on the Vector Observation for SINS. *IEEE Trans. Instrum. Meas.* **2019**, *68*, 3740–3750. [CrossRef]
13. Luo, L.; Huang, Y.; Zhang, Z.; Zhang, Y. A New Kalman Filter-Based In-Motion Initial Alignment Method for DVL-Aided Low-Cost SINS. *IEEE Trans. Veh. Technol.* **2021**, *70*, 331–343. [CrossRef]
14. Xu, X.; Sun, Y.; Gui, J.; Yao, Y.; Zhang, T. A Fast Robust In-Motion Alignment Method for SINS with DVL Aided. *IEEE Trans. Veh. Technol.* **2020**, *69*, 3816–3827. [CrossRef]
15. Liu, X.X.; Liu, X.J.; Song, Q. A Novel Self-Alignment Method For SINS Based on Three Vectors of Gravitational Apparent Motion in Inertial Frame. *Measurement* **2015**, *62*, 47–62. [CrossRef]
16. Liu, Y.; Xu, X.; Liu, X.; Yao, Y.; Wu, L.; Sun, J. A Self-Alignment Algorithm for SINS Based on Gravitational Apparent Motion and Sensor Data Denoising. *Sensors* **2015**, *15*, 9827–9853. [CrossRef]
17. Wang, J.; Zhang, T.; Tong, J.; Li, Y. A Fast SINS Self-Alignment Method Under Geographic Latitude Uncertainty. *IEEE Sens. J.* **2020**, *20*, 2885–2894. [CrossRef]
18. Wu, M.P.; Wu, Y.X.; Hu, X.P. Optimization-based alignment for inertial navigation systems: Theory and algorithm. *Aerosp. Sci. Technol.* **2011**, *15*, 1–17. [CrossRef]
19. Wu, Y.; Pan, X. Velocity/Position Integration Formula Part I: Application to In-Flight Coarse Alignment. *IEEE Trans. Aerosp. Electron. Syst.* **2013**, *49*, 1006–1023. [CrossRef]
20. Xu, X.; Sun, Y.; Yao, Y.; Zhang, T. A Robust In-Motion Optimization-Based Alignment for SINS/GPS Integration. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 4362–4372. [CrossRef]
21. Xu, X.; Xu, D.; Zhang, T.; Zhao, H. In-Motion Coarse Alignment Method for SINS/GPS Using Position Loci. *IEEE Sens. J.* **2019**, *19*, 3930–3938. [CrossRef]
22. Yao, Y.; Xu, X.; Zhang, T.; Hu, G. An Improved Initial Alignment Method for SINS/GPS Integration with Vectors Subtraction. *IEEE Sens. J.* **2021**, *21*, 18256–18262. [CrossRef]
23. Oshman, Y.; Bar-Itzhack, I.; Choukroun, D. Optimal-REQUEST algorithm for attitude determination. *J. Guid. Control Dyn.* **2004**, *27*, 418–425.