applied sciences

MDPI

Article

A Variable Gain Complementary Filtering Fusion Algorithm Based on Distributed Inertial Network and Flush Air Data Sensing

Weiguang Shao [1,*], Jianwen Zang [1], Jin Zhao [2] and Kai Liu [1]

[1] School of Aeronautics and Astronautics, Dalian University of Technology, Dalian 116081, China; zangjw@mail.dlut.edu.cn (J.Z.)
[2] Shenyang Aircraft Design Institute, Shenyang 110034, China
* Correspondence: 32203116@mail.dlut.edu.cn; Tel.: +86-180-9886-2080

Abstract: Aiming at the problem that the accuracy of flight parameters calculated by the traditional flush air data sensing (FADS) and inertial navigation system (INS) complementary filter cannot meet the fine real-time control of the aircraft, a data fusion algorithm based on distributed inertial network and FADS is proposed. This method converts the inertial navigation parameters calculated by the distributed inertial network into flight parameters, and using the least squares fitting theory, the flight parameters are obtained from the pressure data measured by the FADS pressure hole. Then, by adjusting the filter constant of the complementary filtering algorithm, the flight parameters calculated by the inertial network are fused with the flight parameters calculated by the air data sensing to obtain high-precision flight parameters. Finally, the simulation results show that the proposed filtering algorithm can keep the flight parameter estimation error less than 0.01 degrees throughout the flight phase. Compared with the traditional complementary filter, the estimation error of the proposed algorithm is smaller.

Keywords: data fusion; inertial network; flush air data sensing

check for updates

Citation: Shao, W.; Zang, J.; Zhao, J.; Liu, K. A Variable Gain Complementary Filtering Fusion Algorithm Based on Distributed Inertial Network and Flush Air Data Sensing. *Appl. Sci.* **2023**, *13*, 8090. https://doi.org/10.3390/app13148090

Academic Editors: Xiangbo Xu and Kai Shen

Received: 14 April 2023
Revised: 8 July 2023
Accepted: 8 July 2023
Published: 11 July 2023

Copyright: © 2023 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

For the real-time control, navigation and post-flight data analysis of aircraft flying in the atmosphere, it is necessary to accurately measure the flight parameters [1,2]. The flush air data sensing (FADS) system has no components outside the surface of the aircraft and can be used for modern advanced aircraft such as hypersonic and stealth aircraft. In addition, the whole system has no mechanical moving parts, which reduces the maintenance time and cost, and it is considered to be one of the most promising air data measurement systems for measurements such as angle of attack and sideslip angle [3–5]. FADS relies on an array of pressure sensors distributed at the front of the aircraft to measure the pressure on the aircraft surface and indirectly obtain flight parameters from the pressure distribution. However, when the aircraft is flying at a high angle of attack, the sensitivity of the air pressure on the leeward side to the angle of attack will be seriously reduced. In addition, when the aircraft is in high dynamic maneuvering, the maneuvering of the flight itself will disrupt the original flow field, which in effect will increase the measurement error of the aircraft pressure. The above two points lead to the accuracy of FADS to be drastically reduced when the aircraft is in a high angle of attack or high dynamic maneuvering flight conditions [6]. Unlike FADS, according to the measured acceleration and angular velocity of the aircraft, the inertial navigation system (INS) calculates the velocity, position and attitude of the aircraft through the navigation algorithm and then obtains the atmospheric data according to the atmospheric model [7]. Because the information source of INS is the acceleration and angular velocity of the aircraft, it will not be affected by external interference when using inertial navigation information and an atmospheric model to solve

flight parameters. However, the outstanding disadvantage of INS is that the accuracy decreases with the increase of time due to the influence of drift error; when the navigation time is long, the accuracy of the inertial navigation will be greatly reduced. It is also very difficult to accurately estimate the flight parameters of the aircraft by relying solely on an inertial navigation system [8].

Therefore, it is necessary to fuse the flight parameters estimated by INS and FADS and that the two complement each other to meet the accurate estimation of flight parameters in various flight states. At present, there are two main fusion algorithms for FADS and INS [9–18]. One is a data fusion algorithm based on complementary filtering, the other is a data fusion algorithm based on Kalman filtering. In reference [19], a FADS/INS fusion algorithm based on complementary filtering is designed and verified on the X-34 aircraft, the experiment proves the effectiveness of the algorithm. The essence of the algorithm is a simple complementary filter, the unbiased estimation of the state value of the atmospheric parameters is obtained by combining the parameters of the FADS and the INS parameters along the trajectory. In Reference [20], a complementary filtering algorithm based on FADS measurement results and modified by inertial navigation system is designed to calculate the angle of attack and sideslip angle, and the effectiveness of the algorithm is proved by simulation. Reference [21] proposes to use the angle of attack calculated by INS to correct the angle of attack solved by FADS and fuse the parameters solved by the two through complementary filtering. The simulation results show that the method can finitely improve the measurement accuracy of the aircraft in stable flight and high maneuvering flight.

References [19–21] introduce the fusion algorithm based on complementary filtering. In addition, there is a fusion algorithm based on Kalman filtering. In reference [22], the wind speed estimation method based on Kalman filtering technology and the information fusion technology of FADS/INS were studied; however, the selected Kalman filter state does not contain atmospheric data. Beck R [23] designed a FADS/INS data fusion algorithm based on the Kalman filter and applied it to Mars exploration to estimate atmospheric data. In this method, the surface pressure of the aircraft is taken as the measured value, and the atmospheric density, atmospheric static pressure and wind speed vector are selected as the system state variables to establish the system model. Cheng [24] designed an integrated navigation fusion algorithm for an inertial navigation system and an atmospheric data system based on Kalman filter theory, and the effectiveness of the algorithm was verified by simulation, which proves that the integration of an inertial navigation system and an atmospheric data system is feasible. Jiang [25] proposed the unscented kalman filter (UKF) algorithm to fuse FADS and INS data to estimate the real-time atmospheric data of the aircraft, the algorithm estimates the parameters such as angle of attack and sideslip angle by calculating the speed and height of the aircraft. The simulation results showed that the algorithm has improved the estimation accuracy and system stability compared with the original estimation methods such as Kalman filter.

The above two theories, the fusion algorithm based on complementary filtering and the fusion algorithm based on Kalman filtering, have their own advantages and disadvantages. The structure of the fusion algorithm based on complementary filtering is simple and easy to implement. However, the requirements for filter parameters during maneuvering flight and cruise are contradictory, so the fusion algorithm based on complementary filtering cannot maintain high accuracy during the whole flight process. In addition, when the flight time is long, the accuracy of the fusion algorithm of INS and FADS based on complementary filtering will also decrease due to the significant decline in the accuracy of the inertial navigation system [26]. Furthermore, the fusion algorithm based on the Kalman filter has high accuracy when using a high-order filtering model, but the computational complexity is high and the calculation speed is slow, which is not conducive to real-time estimation. If the low-order model is used, the calculation accuracy decreases rapidly.

At present, the INS and FADS fusion algorithm based on complementary filtering is widely used. However, due to the influence of the aircraft elastic deformation and the sensor error, the accumulation of these errors when the flight time is long reduces the

accuracy of the flight parameters solved by INS, which leads to the low accuracy of flight parameters solved by the fusion algorithm. Moreover, due to the conflicting parameter requirements of the filter during maneuvering flight and cruise flight, the fusion algorithm based on complementary filtering cannot maintain high accuracy throughout the flight phase. Therefore, a variable gain complementary filtering algorithm [26] based on the inertial network [27] is designed in this paper. By fusing the measurement information of inertial sensors at different positions, the accuracy of flight parameters solved by INS is improved, so as to improve the accuracy of the flight parameters solved by the fusion algorithm. In addition, the filtering parameters are adjusted according to different stages to ensure the accuracy of fusion in different flight stages of the aircraft.

The general research idea of this paper is shown in Figure 1. Inertial sensors (accelerometers and gyroscopes) are installed in different parts of the aircraft as inertial nodes, according to the angular velocity and acceleration measured by the network node sensors. The Kalman filter is designed to estimate the elastic deformation of the child nodes, and the elastic deformation compensation algorithm is established to compensate the elastic deformation of the child nodes. Moreover, the dynamic analysis model of the rotation transformation matrix is established to solve the dynamic rotation transformation matrix of each node. The dynamic rotation transformation matrix and distributed data fusion algorithm are used to fuse the data of all nodes to obtain the navigation state, and then the flight parameters of INS are calculated by the inertial velocity. In addition, polynomial fitting is performed on the pressure values measured by FADS to obtain the flight parameters solved by FADS. Finally, the variable gain complementary filtering algorithm is used to fuse the flight parameters of inertial navigation and FADS to obtain high-precision flight parameters.

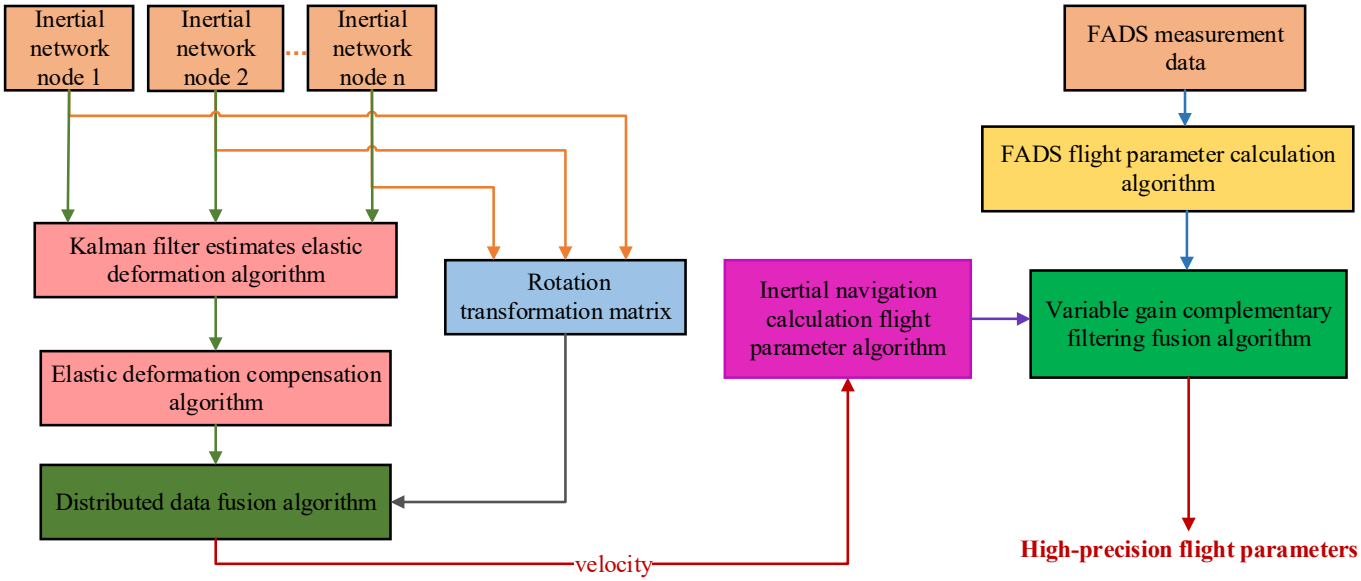


Figure 1. Fusion algorithm based on distributed inertial network and flush air data sensing.

2. Distributed Inertial Network System and Aircraft Elastic Deformation

2.1. Inertial Network System Structure

In order to ensure the reliability of the inertial navigation system, the single node of the inertial network adopts redundant inertial sensors, that is, the number of similar inertial sensors in a single node exceeds three. According to the normal theory, the reliability of the navigation system increases with the increase of the number of inertial sensors. However, with the increase of the number of inertial sensors, the cost, volume and weight of the navigation system also increase accordingly, and when the number of sensors increases to a certain number, the growth of the reliability of the system is not obvious [28].

The number of inertial sensors in a single node of the inertial network has been studied in detail in reference [28]. According to the proposed theory, after considering the reliability, economy and weight of the inertial navigation system, the optimal number of sensors in a single node of the inertial network is 6. This paper adopts the theory proposed in reference [28] and configures 6 similar inertial sensors in a single node.

After determining that the number of sensors in a single node is six, the configuration orientation of these six sensors in the three-dimensional space is determined by the principle of minimum mean square error of least squares estimation. This is discussed in detail in reference [29]. Finally, it is found that the measurement axes of the six inertial sensors are along the normal direction of the six planes of the dodecahedron, and the angle with the coordinate system is 31.7°. This configuration structure minimizes the mean square error of the least squares estimation. This configuration structure is also used in this paper.

Each inertial network node is composed of the above-mentioned six sensors and microprocessors. Each network node is connected to other nodes by a wired or wireless network. The inertial information and local navigation estimation information of each node are shared in the network, thus avoiding a single point of failure [30], as shown in Figure 2. In the microprocessor of each inertial network node, the distributed information fusion algorithm is used to realize the navigation information fusion.

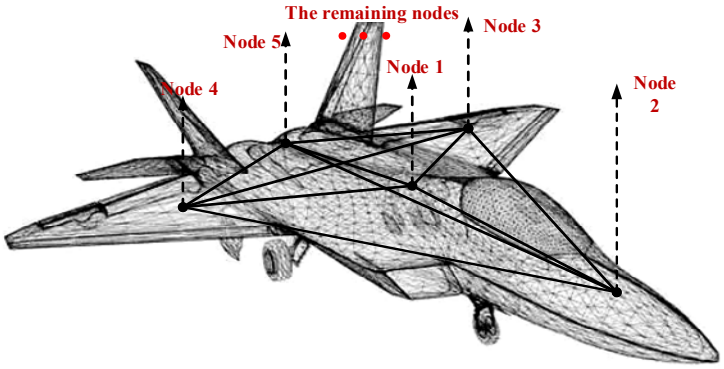


Figure 2. Installation and communication relationship of inertial network nodes.

2.2. Inertial Network Dynamic Measurement Model

If the aircraft structure is rigid, the dynamic relationship between different node frames can be described by fixed rotation transformations, which can be precisely determined from the geometry of node locations at the time of installation. In high-speed flight and high dynamic maneuvers, the airframe should be considered as a flexible structure. If the flexible structure of an aircraft is ignored, continuing to use the static rotation matrix will lead to a large deviation of local state estimation. Accordingly, it is necessary to develop the dynamic relationships between the network nodes and to estimate these dynamic transformation matrices during flight [31].

This paper adopts a calculation method of a dynamic rotation transformation matrix. This method is based on the development of analytical dynamic models of the transformation matrices. The body frame of node 1 is used as a reference frame to represent the relative rotation motion of the other frames and the measured angular velocities [31].

The initial transformation matrix is measured when the aircraft is stationary on the ground. Taking the dynamic conversion between 1 and 2 nodes as an example, I is the inertial coordinate system, 1 represents the local body coordinate system of node 1 and 2 represents the local body coordinate system of node 2.

$$\begin{cases} w(x)^1_{I/2} = w(x)^1_{I/1} + w(x)^1_{1/2} \\ w(y)^1_{I/2} = w(y)^1_{I/1} + w(y)^1_{1/2} \\ w(z)^1_{I/2} = w(z)^1_{I/1} + w(z)^1_{1/2} \end{cases} \quad (1)$$

The angular rate is represented by a skew-symmetric matrix, and the superscript 1 denotes that the angular rate vectors are expressed in terms of the 1 body coordinates.

$$\Omega_{I/2}^1 = \Omega_{I/1}^1 + \Omega_{1/2}^1 \tag{2}$$

Considering the differential equation of the rotation matrix $\Omega_{1/2}^1 = -\dot{C}_1^2 C_2^1$, (2) can be written as

$$\dot{C}_1^2 = \left(\Omega_{I/1}^1 - \Omega_{I/2}^1\right) C_1^2 \tag{3}$$

The measured values of the gyroscope and accelerometer are the angular velocity and acceleration of the node local body system relative to the inertial coordinate system in the node local body coordinate system. In the above Formula (3), $\Omega_{I/1}^1$ can be estimated from the measured value of the gyroscope at node 1, and $\Omega_{I/2}^1$ cannot be obtained directly from the measured value of the gyroscope at node 2.

The skew-symmetric matrix $\Omega_{I/2}^1$ can be represented by rotation transformation matrix as follows:

$$\Omega_{I/2}^1 = C_2^1 \Omega_{I/2}^2 C_1^2 \tag{4}$$

Clearly, the rotation transformation matrix is an orthogonal matrix; Equations (3) and (4) form a nonlinear matrix differential equation group, where the initial matrices can be measured when the aircraft is stationary, and these differential equations are solved at each measurement time to obtain a dynamic rotation transformation matrix. In the above formula, node 1 is usually the main node installed at the center of gravity of the aircraft.

Similarly, the rotation transformation matrices $C_1^i (i = 3, 4 \ldots n)$ can be obtained by diagonal multiplication, such as.

$$C_2^3 = C_1^3 C_2^1 = C_1^3 \left(C_1^2\right)^{-1} \tag{5}$$

2.3. Aircraft Elastic Deformation Model

The inertial sensors are installed in different parts of the aircraft with a distributed structure, and the deflection motion of the carrier structure limits the further improvement of the navigation accuracy of the inertial network system. In order to effectively overcome this adverse effect, it is necessary to further establish a more accurate and practical model of structural deflection motion and study a certain algorithm to correct and compensate for the influence of structural deflection.

The elastic deformation causes an additional rotation angle θ from the master node to the child node, and when the elastic deformation changes with time, there will be an additional angular velocity $\dot{\theta}$ measured by the child node gyroscope, which cannot be measured by the master node. Assuming that there is an installation error angle ψ between the primary and secondary nodes, the total misalignment angle between the primary and secondary nodes is δ, then the relationship between the three can be given by the following:

$$\delta = \theta + \psi \tag{6}$$

The body between the main and sub-nodes constitutes a large elastic system, and in the case of deformation, process noise is selected to describe the motion on each axis. Therefore, this kind of deflection motion can be regarded as a second-order Markov process, the parameters of the model can be determined by structural mechanics experiments [32].

The flexural motion model represented by the second-order Markov process can be written as

$$\ddot{\theta} + 2\beta\theta + \beta^2\theta = \rho \tag{7}$$

In Formula (7), $\theta = \begin{bmatrix} \theta_x & \theta_y & \theta_z \end{bmatrix}^T$ is the elastic deformation angle vector, where the variance $\sigma^2 = \begin{bmatrix} \sigma_x^2 & \sigma_y^2 & \sigma_z^2 \end{bmatrix}^T$; moreover, the parameter $\beta = 2.146/\tau$, τ is the correlation

time of the random process of the elastic deformation of three axes; ρ is a white noise with a certain variance, and its spectral density $Q = 4\beta^3\delta^2$, that is, $\rho \sim N(0, Q)$.

The coordinates of the sub-node i in the main node coordinate system are $\begin{bmatrix} x_i^1 & y_i^1 & z_i^1 \end{bmatrix}^T$. Considering the constraints of the wing structure, the relative displacement caused by the deflection deformation is set as

$$P_f^1 = \begin{bmatrix} P_{fx}^1 \\ P_{fy}^1 \\ P_{fz}^1 \end{bmatrix} = \begin{bmatrix} 0 \\ x_i^1\theta_{fz} - x_i^1\theta_{fx} \\ -x_i^1\theta_{fy} \end{bmatrix} \quad V_f^1 = \begin{bmatrix} V_{fx}^1 \\ V_{fy}^1 \\ V_{fz}^1 \end{bmatrix} = \begin{bmatrix} 0 \\ x_i^1\omega_{fz} - x_i^1\omega_{fx} \\ -x_i^1\omega_{fy} \end{bmatrix} \tag{8}$$

3. Aircraft Elastic Deformation Estimation and Inertial Network Algorithm Design

3.1. Aircraft Elastic Deformation Estimation Algorithm

The elastic deformation of flight affects the performance of inertial navigation. If the error angle between the master node and the child node cannot be accurately estimated, and the deflection deformation at the child node cannot be compensated, it will lead to the child node (such as the weapon system) to misestimate the navigation parameters such as attitude, speed and position and reduce the accuracy of the inertial network fusion algorithm. Therefore, the Kalman filter is used to estimate the misalignment angle, the elastic deformation angle and the elastic deformation angle rate in real time by using the sampling data of the gyroscope and accelerometer of the child node and the master node. When the misalignment angle reaches a certain accuracy, the error of the child node is corrected once.

The state equation and measurement equation can be written in the form

$$\begin{aligned} \dot{X} &= FX + w \\ Z &= HX + v \end{aligned} \tag{9}$$

The Kalman filter state vector is given by

$$X = \begin{bmatrix} \Phi_x & \Phi_y & \Phi_z & \delta V_x & \delta V_y & \delta V_z & \varepsilon_x & \varepsilon_y & \varepsilon_z & \nabla_x & \nabla_y & \nabla_z \\ \mu_x & \mu_y & \mu_z & \theta_x & \theta_y & \theta_z & w_x & w_y & w_z & & & \end{bmatrix}^{\mathrm{T}} \tag{10}$$

where $\Phi_x \quad \Phi_y \quad \Phi_z$ is attitude angle error; $\delta V_x \quad \delta V_y \quad \delta V_z$ is velocity error; $\varepsilon_x \quad \varepsilon_y \quad \varepsilon_z$ is gyroscope bias error; $\nabla_x \quad \nabla_y \quad \nabla_z$ is accelerometer bias error; $\mu_x \quad \mu_y \quad \mu_z$ is initial installation angle error; $\theta_x \quad \theta_y \quad \theta_z$ is elastic deformation angle; $w_x \quad w_y \quad w_z$ is elastic deformation angular velocity.

The measurement vector is given by

$$Z = \begin{bmatrix} \varphi_x & \varphi_y & \varphi_z & \delta V_x & \delta V_y & \delta V_z \end{bmatrix}^{\mathrm{T}} \tag{11}$$

The state transition matrix F is obtained by equivalent transformation of inertial navigation error equation [33], the F matrix is as follows.

$$F = \begin{bmatrix} -\left[\left(\omega_{en}^n + \omega_{ie}^n\right)\times\right] & 0_{3\times3} & -C_{bs}^n & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ \left[f^n\times\right] & -\left[\left(\omega_{en}^n + 2\omega_{ie}^n\right)\times\right] & 0_{3\times3} & C_{bs}^n & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & F_1 & F_2 \end{bmatrix} \tag{12}$$

The values of F_1 and F_2 in Formula (12) are given by

$$F_1 = \begin{bmatrix} -\beta_x^2 & 0 & 0 \\ 0 & -\beta_y^2 & 0 \\ 0 & 0 & -\beta_z^2 \end{bmatrix}, \; F_2 = \begin{bmatrix} -2\beta_x & 0 & 0 \\ 0 & -2\beta_y & 0 \\ 0 & 0 & -2\beta_z \end{bmatrix} \tag{13}$$

The measurement matrix H is given by the relationship between the selected state vector and the measurement vector.

$$H = \begin{bmatrix} I_{3\times3} & 0_{3\times3} & 0_{3\times6} & -C_{bs}{}^{n} & -C_{bs}{}^{n}0_{3\times3} \\ 0_{3\times3} & I_{3\times3} & 0_{3\times6} & 0_{3\times3} & 0_{3\times3}\ 0_{3\times3} \end{bmatrix} \tag{14}$$

The Kalman filter is used to estimate the misalignment angles, installation error angle, elastic deformation angle and elastic deformation angular velocity, and then the navigation state of the child node is compensated by using the estimated error state.

3.2. Distributed Data Fusion Algorithm

As shown in Figure 3, after receiving the measurement information of its own nodes and other nodes, the micro-computer of the node first calculates the rotation transformation matrix of each node and the covariance of the local estimation error of the least squares estimation and then uses the maximum likelihood estimation method to obtain a more accurate and reliable local inertial state vector of each node. The multi-sample algorithm is used to solve the local inertial state vector to obtain the corresponding navigation state (attitude, speed, position). Combined with the information of other nodes, the local Kalman filter is constructed to obtain the local estimation of the node. Finally, the local state estimation of the other nodes is received, and local information fusion is performed to obtain higher performance local navigation state estimation. The measurement of data and the calculation of the rotation transformation matrix have been described in detail in the previous content of this paper. In the following, the measurement fusion algorithm, the navigation state solution algorithm, the local Kalman filter and the navigation state fusion algorithm are introduced in detail.

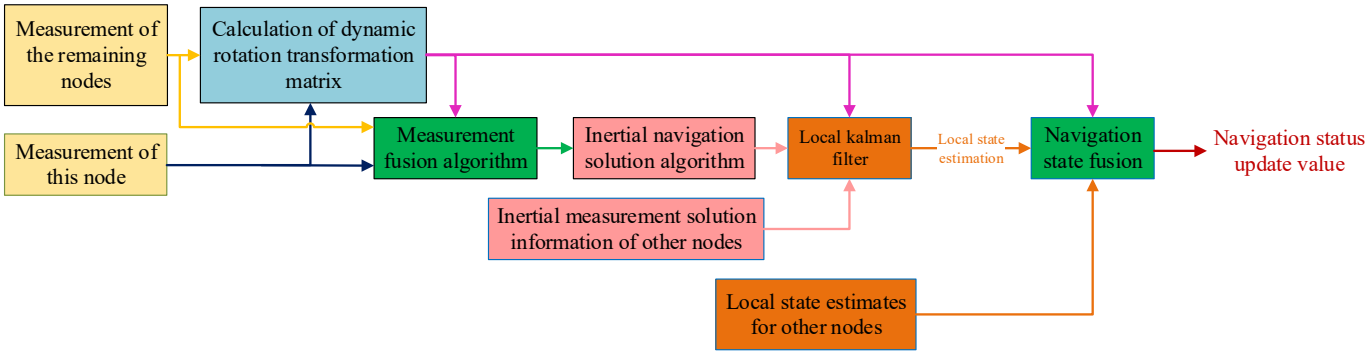


Figure 3. Distributed inertial network data fusion algorithm.

3.2.1. Measurement Fusion Algorithm

Taking node 1 as an example, the measurement equation is as follows:

$$m_1 = Hx_1 = HC_i^1 x_i \tag{15}$$

where $C_i^1 (i = 2 \ldots n)$ is the dynamic rotation transformation matrix, x_1 is the 3-dimensional local vector of node 1 (the acceleration or angular velocity of node 1), m_1 is the measured value of 6 sensors of node 1 and H is the installation matrix of nodes. The single node

of the network in this paper selects the dodecahedron installation of the six sensors, the installation matrix can be determined as follows:

$$H = \begin{bmatrix} 0.5257 & 0 & 0.8507 \\ -0.5257 & 0 & 0.8507 \\ 0.8507 & 0.5257 & 0 \\ 0.8507 & -0.5257 & 0 \\ 0 & 0.8507 & 0.5257 \\ 0 & 0.8507 & -0.5257 \end{bmatrix} \tag{16}$$

The three-dimensional local vector x in the measurement equation is estimated by the least squares method.

$$\hat{x}_1 = \left(H^T H\right)^{-1} H^T m_1 \tag{17}$$

The covariance matrix of the estimated value is of the form

$$P_x = (H^T H)^{-1} H^T R H (H^T H)^{-1} \tag{18}$$

where R is the covariance matrix of the measurement error. Because the sensors in a single node are independent, the covariance matrix is the diagonal matrix of the variance of each sensor.

All sensors of a single node are independent, and their measurement results satisfy the Gaussian probability distribution in the same frame. The errors of the local inertial state estimates are also a Gaussian distributed random vector. The Gaussian probability density function of the multi-dimensional (independent and identically distributed) normal distribution is as follows.

$$p(x) = \frac{1}{(2\pi)^{D/2}|\Lambda|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \Lambda^{-1}(x-\mu)\right) \tag{19}$$

where Λ is the covariance matrix of random variable x, D is the dimension of random variable x and μ represents mathematical expectation.

The dimension of the local inertial state vector is 3, and $P_{x,l}$ is the covariance matrix of the error of the local inertial state estimate. Taking node 1 as an example, the probability density distribution function of the local inertial state can be written as

$$\begin{cases} P_1(x) = \frac{1}{\sqrt{(2\pi)^3 det P_{x,1}}} \exp\left[-\frac{1}{2}(\hat{x}_1 - x_1)^T P_{x,1}^{-1}(\hat{x}_1 - x_1)\right] \\ P_2(x) = \frac{1}{\sqrt{(2\pi)^3 det P_{x,2}}} \exp\left[-\frac{1}{2}\left(\hat{x}_2 - C_1^2 x_1\right)^T P_{x,2}^{-1}\left(\hat{x}_2 - C_1^2 x_1\right)\right] \\ \vdots \\ P_n(x) = \frac{1}{\sqrt{(2\pi)^3 \det P_{x_1 n}}} \exp\left[-\frac{1}{2}\left(\hat{x}_n - C_1^n x_1\right)^T P_{x_1 n}^{-1}\left(\hat{x}_n - C_1^n x_1\right)\right] \end{cases} \tag{20}$$

The objective of the fusion of inertial measurements is to generate the optimal estimates of all the local states of inertia, defining the optimality criterion as the maximum of the conditional probabilities. Then, since the measurements of all nodes are independent, the conditional probability density function of the true local inertial state of each node can be written as

$$L = P_1(x) P_2(x) \cdots P_n(x) \tag{21}$$

The state corresponding to the maximum value of the established likelihood function is obtained.

$$x_1 = \left(\sum_{i=1,2\ldots n} C_i^1 P_{x,i}^{-1} C_1^i\right)^{-1} \sum_{i=12\cdots n} C_i^1 P_{x_i i}^{-1} \hat{x}_i \tag{22}$$

where $\hat{x}_i (i = 1, 2 \ldots n)$ is the local vector obtained by the least squares estimation, substituting the formula of the least squares estimation into (22), it can be written as

$$x_1 = \left(\sum_{i=1,2\ldots n} C_i^1 P_{x,i}^{-1} C_1^i \right)^{-1} \sum_{i=1,2\ldots n} C_i^1 P_{x_i i} \left(H^T H \right)^{-1} H^T m_i \tag{23}$$

Let

$$H_i^* = \left(H^T H \right)^{-1} H^T \tag{24}$$

Then the measurement fusion equation can be given by

$$x_1 = \left(\sum_{i=1,2\ldots n} C_i^1 P_{x,i}^{-1} C_1^i \right)^{-1} \sum_{i=1,2\ldots n} C_i^1 P_{x_i i} H^* m_i \tag{25}$$

The above formula is the inertial measurement fusion algorithm of node 1, m_l $(l = 1, 2, 3 \cdots n)$ is the local inertial redundant measurement vector, $P_{x,l}$ is the covariance matrix of the local inertial measurement fusion estimation error and H is the installation matrix of nodes.

The inertial measurement fusion algorithm of the remaining nodes is similar to that of node 1, and the local state vector after fusion is as follows.

$$x_J = \left[\sum_{l=i,j,\mathrm{cg}} C_l^J P_{x,l}^{-1} C_J^l \right]^{-1} \sum_{l=i,j,\mathrm{cg}} C_l^J P_{x,l}^{-1} H_l^* m_l, J = 1, 2 \ldots n \tag{26}$$

The local inertial state vector obtained by (26) is solved by the multi-sample algorithm to obtain the navigation state value [33]. Therefore, it is not described in detail in this paper. Then, the Kalman filter model is established.

3.2.2. Distributed State Fusion Algorithm

The local dynamic model on each node of the inertial network system can be described as

$$\begin{aligned} x_J(t_k) &= \Phi_J(t_k, t_{k-1}) x_J(t_{k-1}) + G_J(t_{k-1}) w_J(t_{k-1}) \\ z_J(t_k) &= D_J(t_k) x_J(t_k) + v_J(t_k) \end{aligned} \tag{27}$$

where $J = 1, 2, 3 \cdots n$ represents each node; the local state vector x_J can be divided into local system state $x_{1,J}$ and local sensor error state $x_{2,J}$.

$$x_J = \begin{bmatrix} x_{1J}^{\mathrm{T}} & x_{2J}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \tag{28}$$

The local system states at the nodes are called similar states, and the transformation between these similar states is given by the dynamic transformation matrix. The measurement vector z_J represents all the observations (including the observations of the node) after receiving the inertial measurement information of other nodes, which can be decomposed into two sub-vectors.

$$z_J = \begin{bmatrix} z_{JL}^{\mathrm{T}} & z_{JA}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \tag{29}$$

where z_{JL} is the measurement vector provided by the sensor of this node, and z_{JA} is the combination of inertial measurements absorbed from other nodes. Since the two measurement vectors are independent of each other, the following decomposition can be obtained as follows:

$$\begin{aligned} D_J &= \begin{bmatrix} D_{JL}^{\mathrm{T}} & D_{JA}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \\ v_J &= \begin{bmatrix} v_{JL}^{\mathrm{T}} & v_{JA}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \\ R_J &= \mathrm{blockdiag} \begin{bmatrix} R_{JL} & R_{JA} \end{bmatrix} \end{aligned} \tag{30}$$

The local Kalman filter uses the assimilated sensor measurements to estimate the local state, and the local state fusion filter combines the local estimation with the estimation after assimilation of other nodes to update the local estimation. The Kalman filter algorithm is applied to the local dynamic model on each node, and the local state estimation $\hat{x}$ and covariance matrix P_J can be obtained.

$$\begin{aligned} &P_J^{-1}(t_k^+) = P_J^{-1}(t_k^-) + D_J^{\mathrm{T}}(t_k)R_J^{-1}(t_k)D_J(t_k) \\ &= P_J^{-1}(t_k^-) + \sum_{k=JL,JA} D_l^{\mathrm{T}}(t_k)R_l^{-1}(t_k)D_l(t_k) \\ &P_J^{-1}(t_k^+)\hat{x}_J(t_k^+) = P_J^{-1}(t_k^-)\hat{x}_J(t_k^-) + D_J^{\mathrm{T}}(t_k)R_J^{-1}(t_k)z_J(t_k) \\ &= P_J^{-1}(t_k^-)\hat{x}_J(t_k^-) + \sum_{l=JL,JA} D_l^{\mathrm{T}}(t_k)R_l^{-1}(t_k)z_l(t_k) \end{aligned} \tag{31}$$

The mean square error matrix of state one-step prediction and state one-step prediction in Formula (31) is

$$\begin{aligned} \hat{x}_J(t_k^-) &= \Phi_J(t_k, t_{k-1})\overline{x}_J(t_{k-1}) \\ P_J(t_k^-) &= \Phi_J(t_k, t_{k-1})\overline{P}_J(t_{k-1}^+)\Phi_J^{\mathrm{T}}(t_k, t_{k-1}) \\ &\quad + G_J(t_{k-1})Q_J(t_{k-1})G_J^{\mathrm{T}}(t_{k-1}) \end{aligned} \tag{32}$$

where $\overline{x}$ and $\overline{P}$ are the local state updates obtained by the local state fusion filter. Therefore, it is only necessary to calculate the values of $\overline{x}$ and $\overline{P}$ at each moment, so that the filter can work normally and output the final result. The calculation method will be explained below.

Where we introduce the nonlinear functional model.

$$J = \int_a^b f^2(t)dt \tag{33}$$

In the optimal control system, the above equation can be expressed as

$$J = \int_a^b (x^T P x)dt \tag{34}$$

After discretization of the Equation (34), it can be written as

$$J = \sum x^T P x \tag{35}$$

A state fusion filter is established on each node. Taking node 1 as an example, define the quadratic cost function of node 1 as follows:

$$F_1 = \sum_{J=1-n} \left(C_J^1 x_J - x_1\right)^{\mathrm{T}} C_J^1 P_J^{-1} C_1^J \left(C_J^1 X_J - x_1\right) \tag{36}$$

where x_1 is the true local similar state value of node 1, and F_1 is a cost function to measure the displacement between the local state estimation and the true value.

The state fusion filter is designed to minimize F_1, which is called the minimum weighted mean square error criterion, and the local state update values are as follows:

$$\begin{aligned} &\overline{x}_1 = \overline{P}_1{}^{-1}(P_1{}^{-1}\hat{x}_1 + C_2{}^1P_2{}^{-1}\hat{x}_2 + C_3{}^1P_3{}^{-1}\hat{x}_3 + \cdots + C_n{}^1P_n{}^{-1}\hat{x}_n) \\ &\overline{P}_1{}^{-1} = (P_1{}^{-1} + C_2{}^1P_2{}^{-1}C_1{}^2 + C_3{}^1P_3{}^{-1}C_1{}^3 + \cdots + C_n{}^1P_n{}^{-1}C_1{}^n)^{-1} \end{aligned} \tag{37}$$

Similarly, the update equation at the other nodes is as follows:

$$\begin{aligned}
&\overline{x}_2 = \overline{P}_2{}^{-1}(C_1{}^2P_1{}^{-1}\hat{x}_1 + P_2{}^{-1}\hat{x}_2 + C_3{}^2P_3{}^{-1}\hat{x}_3 + \cdots + C_n{}^2P_n{}^{-1}\hat{x}_n)\\
&\overline{P}_2{}^{-1} = (C_1{}^2P_2{}^{-1}C_2{}^1 + P_2{}^{-1} + C_3{}^2P_3{}^{-1}C_2{}^3 + \cdots + C_n{}^2P_n{}^{-1}T_2{}^n)^{-1}\\
&\qquad\qquad\vdots\\
&\overline{x}_n = \overline{P}^{-1}(C_1{}^nP_n{}^{-1}\hat{x}_1 + C_2{}^nP_2{}^{-1}\hat{x}_2 + C_3{}^nP_3{}^{-1}\hat{x}_3 + \cdots + P_n{}^{-1}\hat{x}_n)\\
&\overline{P}_n{}^{-1} = (C_1{}^nP_n{}^{-1}T_1{}^2 + C_2{}^nP_2{}^{-1}T_n{}^2 + C_3{}^nP_3{}^{-1}T_n{}^3 + \cdots + P_n{}^{-1})^{-1}
\end{aligned} \tag{38}$$

4. The Fusion of Inertial Network and FADS to Solve Flight Parameters

4.1. The Flight Parameters Are Solved by the Fusion of Inertial Network

Many flight parameters exist, like angle of attack, side slip angle, Mach-Number, static and dynamic drag, etc. Due to the limitation of space, this paper will use the commonly used angle of attack as the flight parameter to be solved, and the solving process of the remaining parameters is similar to that of the angle of attack.

According to Newton's law, the inertial navigation system passes the three-axis acceleration and angular velocity information under the inertial frame. After coordinate transformation, the attitude angle, velocity and position information of the aircraft are obtained by integral, and the angle of attack is estimated by inertial velocity.

U_B, V_B, W_B are the components of inertial velocity in the body coordinate system. Without considering the wind disturbance, the calculation formula of angle of attack can be written as

$$\hat{\alpha} = \arctan\frac{W_B}{U_B} \tag{39}$$

4.2. The Flight Parameters Are Solved by FADS

From the pressure distribution measured by the pressure sensor array, FADS derives the flight parameters indirectly (angle of attack, angle of sideslip, dynamic pressure, static pressure, Mach number). When the pressure measurement value is obtained by the pressure hole array, the algorithm for solving the flight parameters of the FADS system can be roughly divided into two categories. One is based on the semi-empirical formula model, which is mainly suitable for establishing the relationship between the surface pressure distribution and the atmospheric data according to the physical principle. The other is a data-based solution method, which directly uses interpolation, fitting and other methods to calculate the atmospheric data according to the pressure value measured by the pressure hole array [34–39].

In this paper, the polynomial fitting method is used to solve the flight parameters (attack), and only the quadratic term and the following terms are considered. This is because if the degree of the polynomial is too high, the output will be very sensitive to the input error. At the same time, the high degree polynomial takes into account the cross product of all pressure values, and the polynomial is too complex. The pressure values measured by six pressure measuring holes are taken as input values, and the corresponding angles of attack are taken as output values, respectively. The angle of attack can be expressed as

$$\begin{aligned}
&\alpha = a_0 + a_1p_1 + a_2p_2 + a_3p_3 + a_4p_4 + a_5p_5 + a_6p_6\\
&+a_7p_1^2 + a_8p_2^2 + a_9p_3^2 + a_{10}p_4^2 + a_{11}p_5^2 + a_{12}p_6^2 +\\
&a_{13}p_1p_2 + a_{14}p_1p_3 + a_{15}p_1p_4 + a_{16}p_1p_5 + a_{17}p_1p_6\\
&+a_{18}p_2p_3 + a_{19}p_2p_4 + a_{20}p_2p_5 + a_{21}p_2p_6 + a_{22}p_3p_4\\
&+a_{23}p_3p_5 + a_{24}p_3p_6 + a_{25}p_4p_5 + a_{26}p_4p_6 + a_{27}p_5p_6
\end{aligned} \tag{40}$$

The process of calculating the undetermined coefficient $a_0 - a_{27}$ according to the input and output data is to find the value that minimizes the residual vector 2 norm of the calculated value and the sample value. The optimization objective function is as follows:

$$J_\alpha = \min \|\alpha_c - \alpha\|_2^2 \tag{41}$$

where the subscript *c* represents the calculated value. In this way, it is actually an objective function optimization problem. The commonly used methods are gradient method, Newton method and so on. In this study, the Levenberg-Marquardt method is used. The LM method is used to approach the minimum value of the two norms by function approximation. The specific steps are shown in Figure 4.

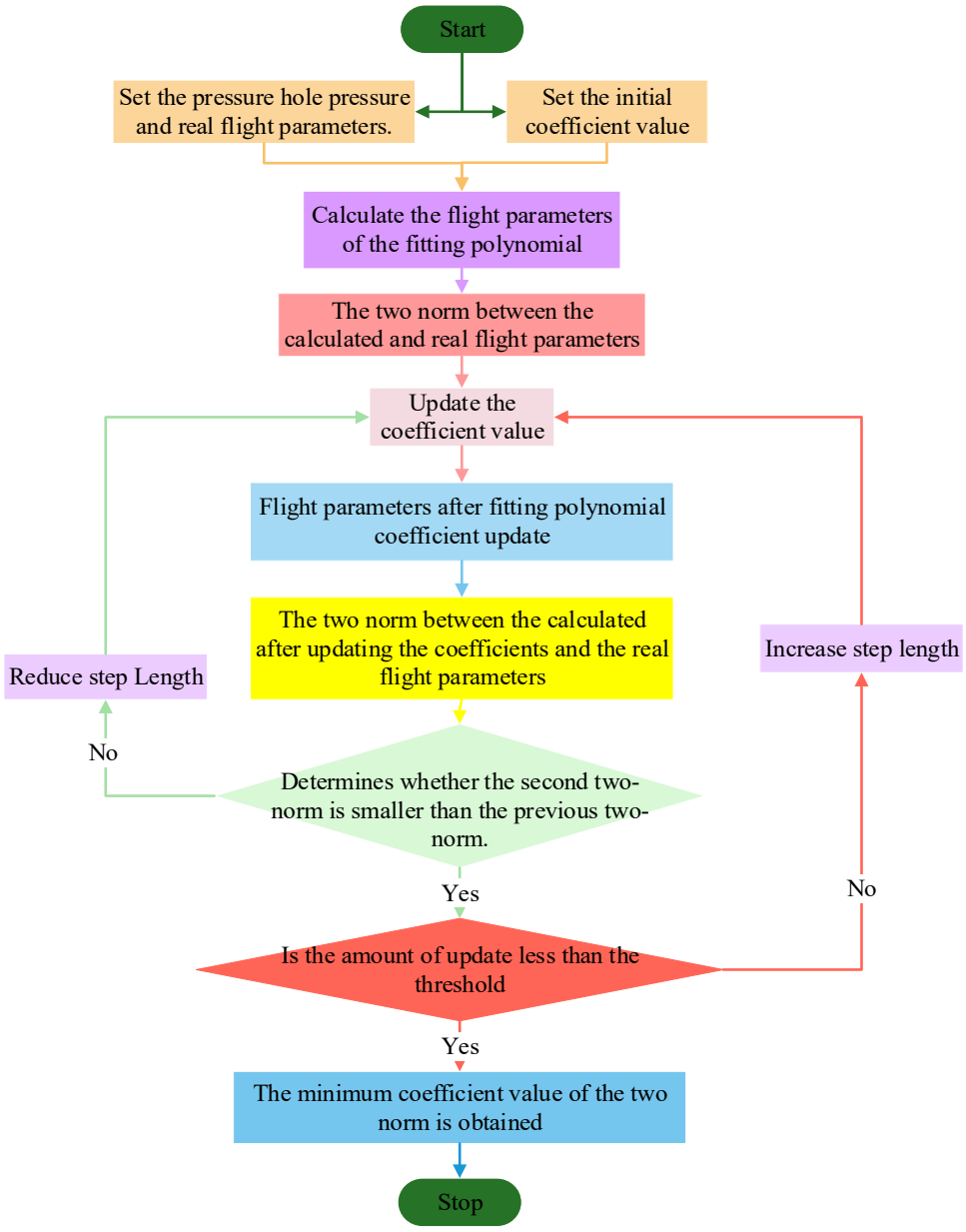


Figure 4. LM algorithm to solve the least square coefficient.

4.3. Fusion of Inertial Network and FADS

The inertial navigation system (INS) can respond to the change of angle of attack in time during maneuvering flight, but there is a large error in steady-state flight. The error of FADS is small in steady-state flight, while there is a large error due to the delay of pressure propagation in maneuvering flight. The two are complementary. The angle of attack of FADS and the angle of attack of INS can be fused by complementary filtering. The FADS angle of attack passes through a low-pass filter to take advantage of its precise steady-state component, while the angle of attack of INS is filtered through a high-pass filter, which

takes advantage of its timely response to the dynamic change of angle of attack and is then superimposed. The output angle of attack can be written as

$$\alpha = \frac{1}{\tau S + 1}(\alpha_{FADS} - \alpha_{INS}) + \alpha_{INS} = \frac{\tau s}{\tau S + 1}\alpha_{NNS} + \frac{1}{\tau s + 1}\alpha_{FADS} \tag{42}$$

However, the filter constant τ of the complementary filter has a significant effect on the output. The larger the τ is, the less the proportion of the FADS angle of attack component in the output result after fusion is, and the inertial angle of attack component has a great influence on the fusion result. Combined with the above analysis, the angle of attack error of INS is smaller during maneuvering flight, and the angle of attack error of FADS is smaller during steady flight. Therefore, when the aircraft is maneuvering, in order to improve the fusion accuracy during maneuvering, the τ can be selected larger, so that the fusion result is more affected by INS. In order to improve the fusion accuracy in level flight, the τ can be selected smaller, so that the fusion results are more affected by FADS.

In order to achieve this goal, a variable gain complementary filtering algorithm is designed. The filter constant τ in the complementary filter can change with the change of flight state, and the better angle of attack fusion results can be maintained throughout the flight phase. The change of the gain is most directly based on the change rate of the angle of attack. Considering that as long as the absolute value of the change rate of the angle of attack is large, whether the change rate is positive or negative, the filter constant τ is larger. Therefore, the change rule of the filter constant is set to

$$\tau = k|\dot{\alpha}| = k\left|\frac{\alpha_{k+1} - \alpha_k}{\Delta t}\right| \tag{43}$$

Bring the above formula into the complementary filtering Formula (42), and the fusion formula of the variable gain complementary filtering algorithm can be obtained. However, because the inertial angle of attack can quickly respond to the change rate of the angle of attack, while the FADS angle of attack has a large delay, the change rate of the angle of attack is very close to the change rate of the inertial angle of attack, and the change rate of the angle of attack can be replaced by the change rate of the inertial angle of attack.

The angle of attack based on variable gain complementary filtering can be written as

$$\begin{aligned}\alpha_{k+1} = {} & \frac{2k\left(\alpha_{\mathrm{INS},k+1} - \alpha_{\mathrm{INS},k}\right) - \Delta t^2}{2k\left(\alpha_{\mathrm{INS},k+1} - \alpha_{\mathrm{INS},k}\right) + \Delta t^2}\alpha_k + \frac{\Delta t^2}{2k\left(\alpha_{\mathrm{INS},k+1} - \alpha_{\mathrm{INS},k}\right) + \Delta t^2} \\ & \left(\alpha_{\mathrm{FADS},k+1} + \alpha_{\mathrm{FADS},k}\right) + \frac{2k\left(\alpha_{\mathrm{INS},k+1} - \alpha_{\mathrm{INS},k}\right)^2}{2k\left(\alpha_{\mathrm{INS},k+1} - \alpha_{\mathrm{INS},k}\right) + \Delta t^2}\end{aligned} \tag{44}$$

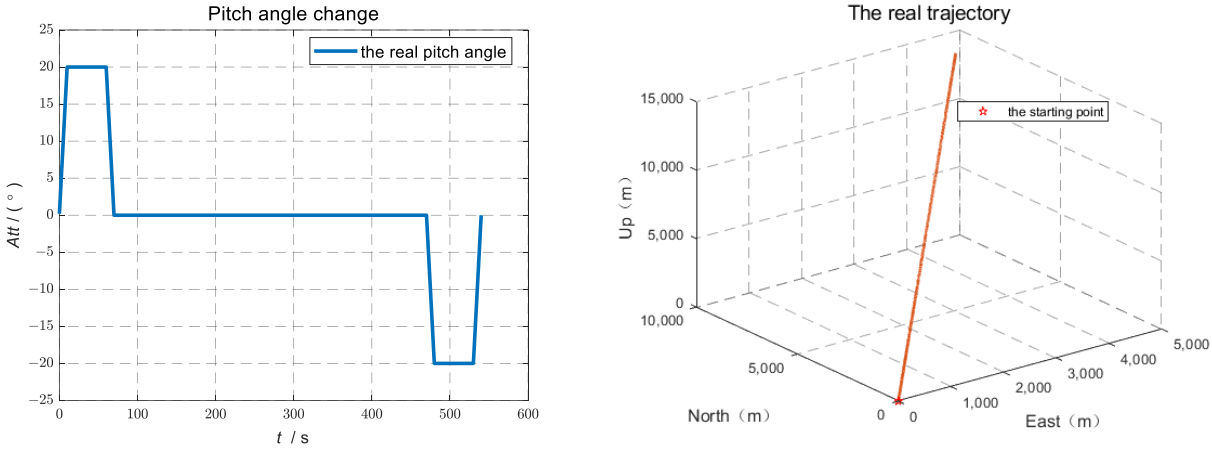
5. Mathematical Simulation Analysis

5.1. Trajectory and Sensor Data Simulation

The flight trajectory is set as follows, and the simulation time is 540 s. In the simulation, the flight trajectory of the aircraft is simplified, and the maneuvering operation is carried out during the flight. The initial angle and velocity are 0, the starting point is the Aerospace Laboratory of Dalian University of Technology, the initial latitude, longitude and height are 38.8880°, 121.541724°, 380 m, and the trajectory changes are shown in the following Figure 5.

In order to express the position clearly, the position increment information is drawn, with the change relative to the original position; ΔLat represents the latitude increment, ΔLon represents the longitude increment and ΔHgt represents the height increment.

Since the increase of nodes only changes the complexity and amount of calculation, we assume that the inertial network is composed of two inertial navigation nodes in the simulation process, which are the main node cg installed at the center of gravity of the aircraft and a local node i located at the wing of the aircraft. The sensor accuracy of the two nodes is shown in Table 1.



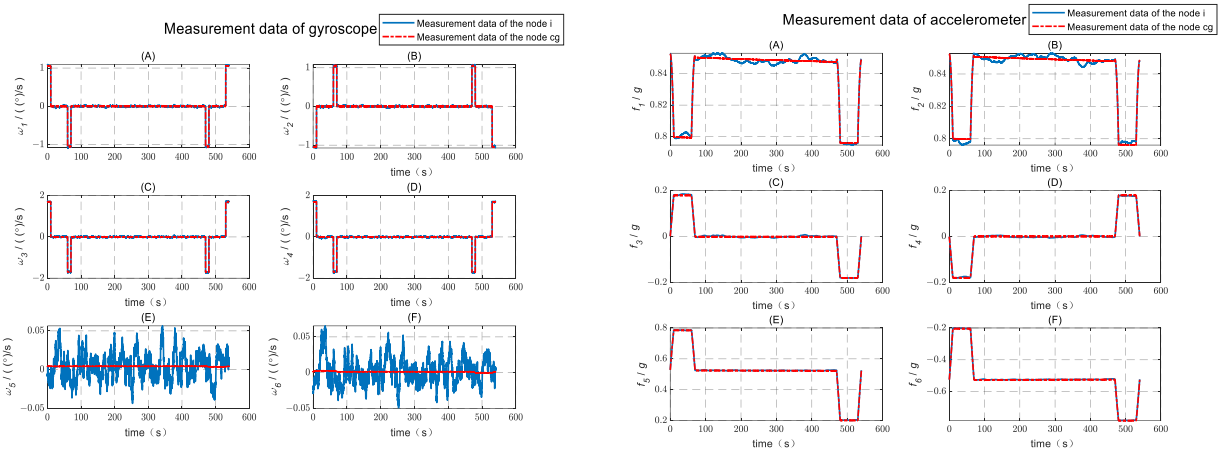
(**a**) (**b**)

Figure 5. The figure of the real trajectories: (**a**) Pitch angle diagram of real trajectory; (**b**) 3D diagram of real trajectory.

Table 1. Simulation parameters of inertial sensors.

Sensor Parameters	Node cg	Node i
Gyroscope bias error (°/h)	0.01	0.04
Gyroscope noise (°/sqrt (h))	0.003	0.17
Accelerometer bias error (ug)	100	200
Accelerometer noise (°/sqrt(h))	25	30
Gyroscope sampling frequency (Hz)	100	100
Accelerometer sampling frequency (Hz)	100	100
Markov correlation time (s)	0	[10; 20; 30]
Markov process variance (min)	0	[6; −10; 7]
Gyroscope installation error (min)	0	[5; 0; 0]

The sensors of the two nodes have sensor errors, different from the node cg, and the sensors of the node *i* are also affected by the elastic deformation. The sampling values of the two nodes are shown in the following Figure 6.



(**a**) (**b**)

Figure 6. Data collected by sensors of node cg and node *i*: (**a**) The data comparison diagram collected by the gyroscopes of node cg and node *i*; (**b**) The data comparison diagram collected by the accelerometers of node cg and node *i*.

5.2. Aircraft Elastic Deformation Estimation and Compensation

The installation error angle, elastic deformation angle and elastic deformation angular rate are estimated by the Kalman filter, and the results are as follows.

It can be seen from Figure 7, the misalignment angle, installation angle error, elastic deformation angle and elastic deformation angular velocity are all accurate, and the convergence speed is faster through the Kalman estimation. Then, the estimated value is used to correct the error of the child node.

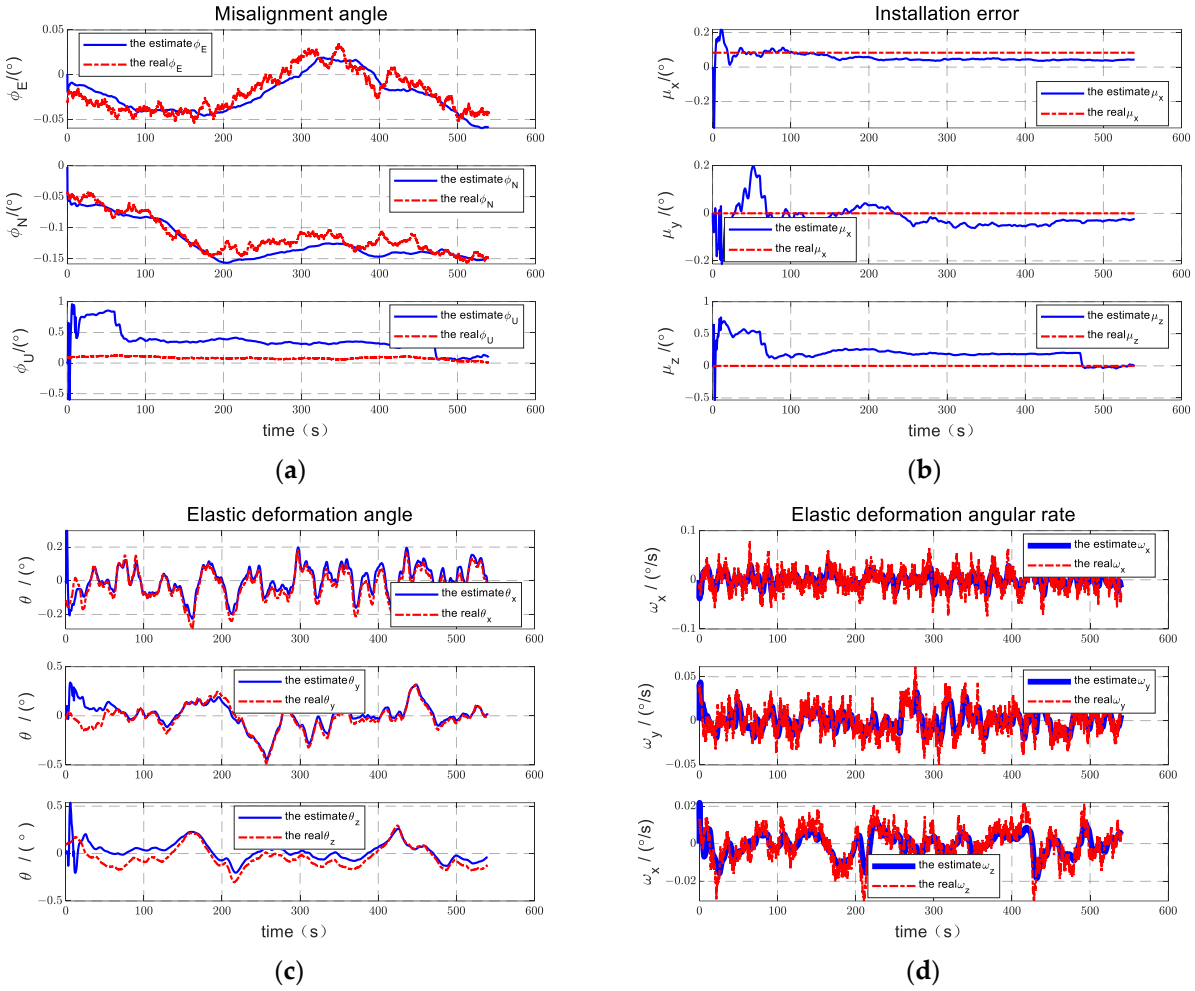


Figure 7. The comparison figures of the Kalman filter estimated value and actual value: (**a**) The comparison diagram of the Kalman filter estimated misalignment angle and real misalignment angle; (**b**) The comparison diagram of the Kalman filter estimated installation error angle and real installation error angle; (**c**) The comparison diagram of the Kalman filter estimated elastic deformation angle and real elastic deformation angle; (**d**) The comparison diagram of the Kalman filter estimated elastic deformation angular velocity and real elastic deformation angular velocity.

5.3. Distributed Information Fusion Algorithm Solves Navigation State

The height channel of the inertial navigation is divergent. Here, the barometric altimeter is used to damp the height direction. After distributed inertial network information fusion, the results are as follows.

Figure 8 shows the comparison of the pitch angle and 3-axis velocity of the two methods in the northeast navigation coordinate system. Because the flight parameters of this paper take the angle of attack as an example, only the curves of velocity and pitch angle are listed here. After about 600 s of navigation, the attitude error is within 0.1 degrees and the velocity error is within 1 m/s. Compared with the unfused pure navigation solution, the accuracy is significantly improved.

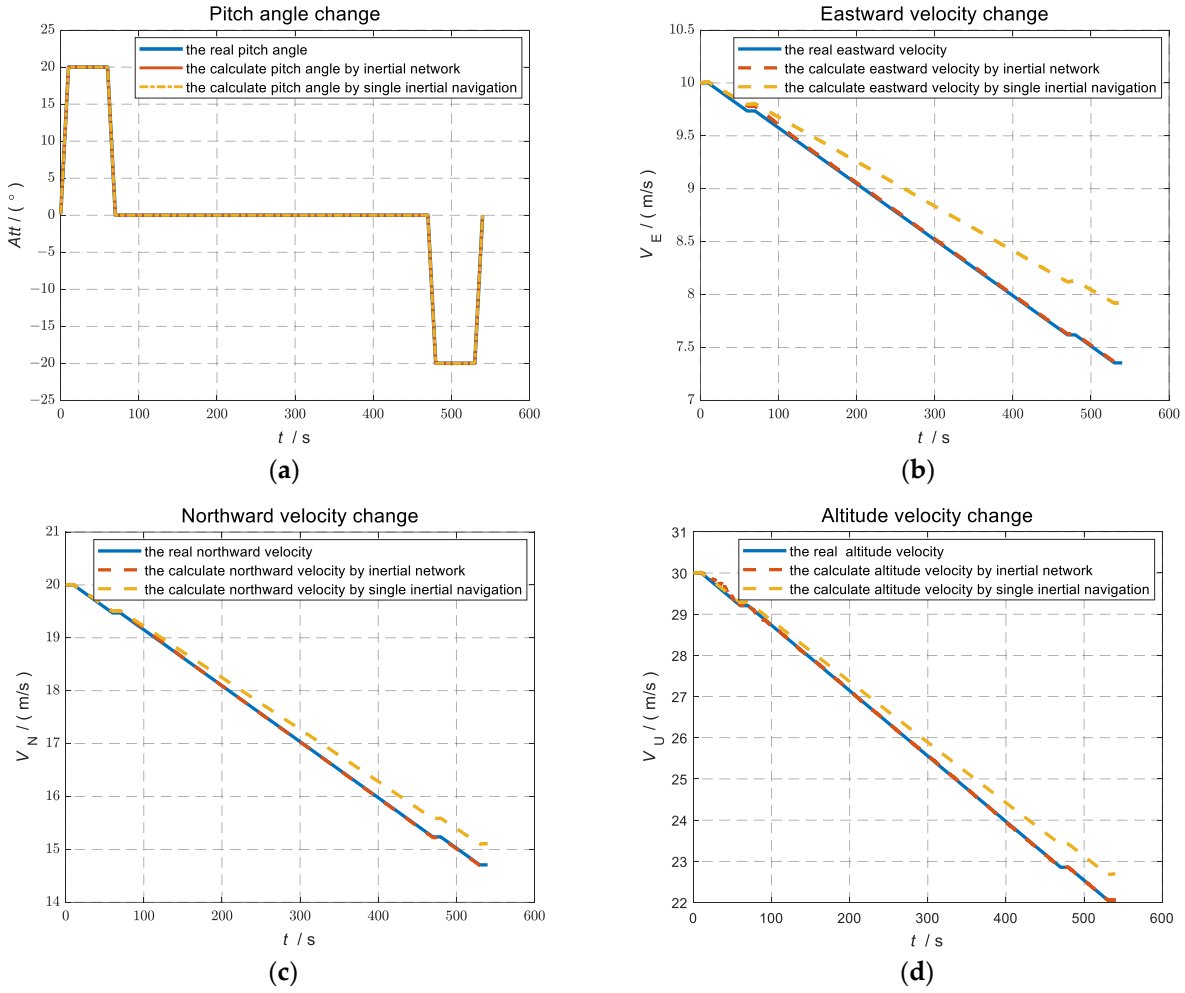


Figure 8. The comparison diagram of navigation parameters calculated by distributed fusion algorithm and inertial navigation alone: (**a**) The pitch angle comparison diagram of distributed fusion algorithm and single inertial navigation; (**b**) The east velocity comparison diagram of distributed fusion algorithm and single inertial navigation; (**c**) The north velocity comparison diagram of distributed fusion algorithm and single inertial navigation; (**d**) The comparison diagram of the altitude velocity between the distributed fusion algorithm and the single inertial navigation.

5.4. Distributed Inertial Network and FADS Fusion Algorithm Simulation

The flight parameters of this paper take the angle of attack as an example. The angle of attack solved by the distributed inertial network and the angle of attack solved by FADS are fused by the variable gain complementary filtering algorith, and compared with the traditional inertial navigation system(INS) and FADS complementary filtering algorithm. The comparison results are shown in the following Figure 9.

It can be seen from Figure 9a,b that in the whole flight phase, the angle of attack error output by the variable gain complementary filtering algorithm based on the inertial network and FADS is smaller than the angle of attack output by the traditional INS and FADS complementary filtering algorithm, which shows that the variable gain complementary filtering algorithm is effective and can ensure the accuracy of solving flight parameters in the whole flight phase. The angle of attack error output achieved through the traditional complementary filtering algorithm increases with time due to the cumulative effect of INS error. In 500 s, the relative error decreases due to the sudden increase of the angle of attack, but the absolute error is still increasing. It can be obtained from Table 2, at 540 s, the error reaches 0.104°, and the average error of the flight process is 0.0058°. The variable gain complementary filtering algorithm based on the inertial network and FADS can basically

avoid the cumulative effect of error due to the improvement of measurement accuracy caused by the inertial network and the continuous adjustment of the filtering parameters. At 540 s, the error is 0.002° and the average error is 0.0017°, values which are much smaller than in the traditional complementary filtering algorithm. Compared with the average error of the two algorithms, the accuracy of the variable gain complementary filtering algorithm based on the inertial network and FADS is improved by more than 2 times.

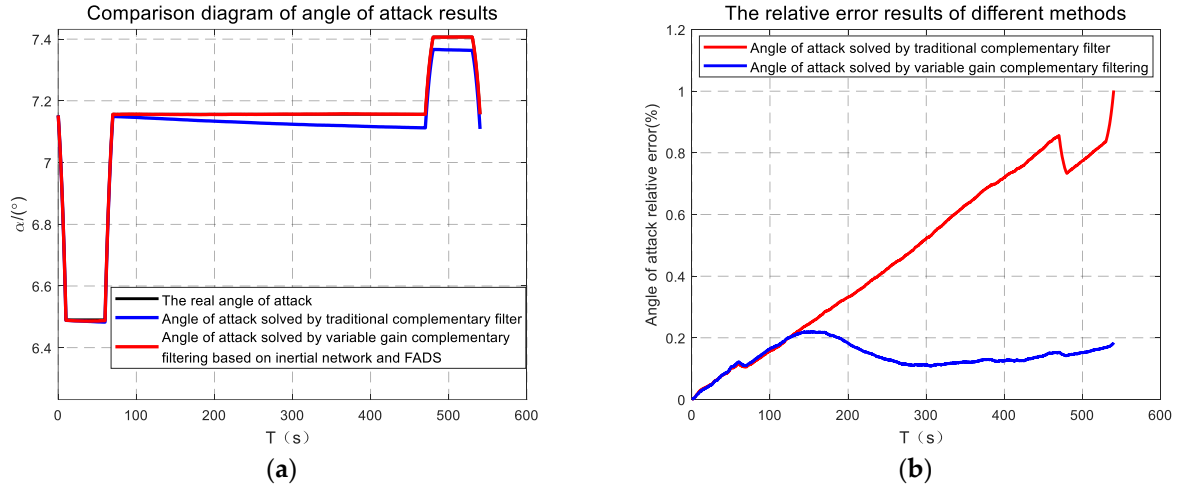


Figure 9. The contrast diagram of the angle of attack solved by the variable gain complementary filtering algorithm and the angle of attack solved by the traditional filtering algorithm: (**a**) The angle of attack comparison diagram of the two algorithms; (**b**) The relative error comparison diagram of the angle of attack solved by the two algorithms.

Table 2. The average error of the two algorithms.

Sensor Parameters	Complementary Filtering Algorithm Based on INS and FADS (°)	Variable Gain Complementary Filtering Algorithm Based on Inertial Network and FADS (°)
Angle of attack	0.0058	0.0017

6. Conclusions

In this paper, the latest progress of inertial navigation and FADS is reviewed, and a variable gain complementary filtering fusion algorithm based on a distributed inertial network and embedded atmospheric sensor is proposed. It is proved by mathematical simulation that the variable gain complementary filtering fusion algorithm based on the distributed inertial network and FADS can better maintain the measurement accuracy of flight parameters during the whole flight trajectory than the traditional INS/FADS complementary filtering algorithm and realize the accurate estimation of flight parameters.

This study can provide a scientific understanding of the effectiveness, accuracy and applicability of data fusion methods in the aviation industry. The algorithm is suitable for processes that require long endurance or have rich flight stages. In the practical application of the algorithm, integrity monitoring should be considered. Integrity refers to the ability to provide users with timely and effective warning information when the error caused by failure or performance deterioration exceeds the acceptable limit (alarm threshold) during the use of the navigation system. Integrity monitoring is essential, so it should be focused on in the practical applications of the algorithm.

Author Contributions: Conceptualization, W.S.; Methodology, W.S. and J.Z. (Jianwen Zang); Software, W.S.; Investigation, J.Z. (Jianwen Zang) and J.Z. (Jin Zhao); Writing—original draft, W.S.; Writing—review & editing, J.Z. (Jianwen Zang) and K.L.; Project administration, J.Z. (Jin Zhao). All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China, grant number U2141229 and foundation under grant JCJQ, grant number 2019-JCJQ-DA-001-131.

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.

References

1. Ding, Z.J.; Zhou, H.; Wu, D.S.; Wu, Y.C.; He, Y.Y. Review of Flush Air Data Sensing System. *Acta Astronaut.* **2019**, *40*, 247–257.
2. Takahashi, H.; Kodera, M.; Tani, K. Flush Air Data Sensing System for a Sharp-Nosed Hypersonic Vehicle with Curved-Wedge Forebody. *AIAA J.* **2020**, *58*, 4819–4831. [CrossRef]
3. Fan, X.; Jiang, Z.; Bai, X.; Zhang, S.; Liu, L. Design and Verification of Flush Air Data Sensing Module with Navigation and Temperature Information. *Aerospace* **2021**, *8*, 336. [CrossRef]
4. Wang, Y.; Xiao, Y.; Zhang, L.; Zhao, N.; Zhu, C. Flush Airdata System on a Flying Wing Based on Machine Learning Algorithms. *Aerospace* **2023**, *10*, 132. [CrossRef]
5. Xiao, D.B. Research on Algorithms and Key Technologies for Em-Bedded Atmospheric Data Sensing System. Master's Thesis, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 2008.
6. Wang, P.; Jin, X.; Zhang, W.M. Fusion Method for FADS /INS Integrated System. *Tactical Missile Technol.* **2012**, *6*, 70–73.
7. Song, L.J.; Xue, L.D.; Yan, Q.; Zhao, W.L.; Chen, X.Z. Review and Prospect of Inertial Technology Development Process. *Navig. Control* **2021**, *20*, 2029–2043.
8. Lu, C.; Li, R.B.; Liu, J.Y.; Lei, T.W.; Guo, Y. Calculation method for air data based on information from inertial navigation system and flight control system. *Control Decis.* **2017**, *32*, 363–367.
9. Baumann, E.; Pahle, J.W.; Davis, M.C.; White, J.T. X—43A flush air-data sensing system flight-test results. *J. Spacecr. Rocket.* **2010**, *47*, 48–61. [CrossRef]
10. Xiao, D.B.; Jiang, B.R.; Liu, P. Research on atmospheric data measurement method based on FADS/IMU information fusion. *Meas. Control Technol.* **2022**, *41*, 50–55.
11. Karlgaard, C.D.; Kutty, P.; Schoenenberger, M.; Munk, M.M.; Little, A.; Kuhl, C.A.; Shidner, J. Mars science laboratory entry atmospheric data system trajectory and atmosphere reconstruction. *J. Spacecr. Rocket.* **2014**, *51*, 1029–1047. [CrossRef]
12. Van Hove, B.; Karatekin, Ö.; Schleutker, T.; Gülhan, A. Exomars flush air data system: Entry simulation and atmospheric re-construction method. *J. Spacecr. Rocket.* **2019**, *4*, 1205–1219. [CrossRef]
13. Jiang, S.; Qiu, Q.; Yuan, S.; Shi, X.; Li, L.; Zhang, X.; Fu, K.; Qin, D.; Guo, F.; Wang, Z.; et al. Stochastic resonance for a bistable system with multiplicative noise and additive recycled noise. *Indian J. Phys.* **2022**, *96*, 3713–3719. [CrossRef]
14. Christopher, D.; Karlgaard, P.K. Coupled inertial navigation and flush air data sensing algorithm for atmosphere estimation. In Proceedings of the AIAA Atmospheric Flight Mechanics Conference, Kissimmee, FL, USA, 5–9 January 2015.
15. Arain, B.; Kendoul, F. Real-time wind speed estimation and compensation for improved flight. *IEEE Trans. Aerosp. Electron. Syst.* **2014**, *50*, 1599–1606. [CrossRef]
16. Yang, L.; Hou, Y.Z.; Zuo, G.; Liu, Y.; Guo, B. Aerodynamic characteristics measurement of Mars vehicles during entry flight. *Chin. J. Theor. Appl. Mech.* **2015**, *47*, 8–14.
17. Yang, S.J.; Zhao, J.C.; Yang, Z.H. Information fusion method of flush air data sensing system and inertial navigation. *Tactical Missile Technol.* **2016**, *2*, 95–100.
18. Kelly, G.M.; Findlay, J.T.; Compton, H.R. Shuttle subsonic horizontal wind estimation. *J. Spacecr. Rocket.* **1983**, *20*, 390–397. [CrossRef]
19. Ellsworth, J.C.; Whitmore, S.A. Simulation of a Flush Air-Data System for Transatmospheric Vehicles. *J. Spacecr. Rocket.* **2012**, *45*, 716–732. [CrossRef]
20. Ye, W.; Zheng, S.D.; Wen, R.H. A FADS/INS Combined Method for Measurement of Angle of Attack and Sideslip. *Aircr. Des.* **2007**, *27*, 14–18.
21. Yang, Y.; Lu, Y.P.; Wu, Z.G. Combination filter technology in flush air data sensing system. *Transducer Microsyst. Technol.* **2009**, *28*, 117–120.
22. Langelaan, J.W.; Alley, N.; Neidhoefer, J. Wind Field Estimation for Small Unmanned Aerial Vehicles. *J. Guid. Control Dyn.* **2012**, *34*, 1016–1030. [CrossRef]
23. Beck, R.; Karlgaard, C.; O'Keefe, S.; Siemers, P.; White, B.; Engelund, W.; Munk, M. Mars Entry Atmospheric Data System Modeling and Algorithm Development. In Proceedings of the AIAA Thermophysics Conference, San Antonio, TX, USA, 22–25 June 2009.
24. Cheng, C. Modeling of Air Data System and It's Application in Integrated Navigation. Master's Thesis, University of Electronic Science and Technology of China, Chengdu, China, 2019.
25. Jiang, B.R.; Liu, P.; Xiao, D.B. Air Data Estimation of FADS/INS Fusion Based on UKF. *J. Astronaut. Metrol. Meas.* **2022**, *4*, 31–36.

26. Xiao, D.B.; Jiang, B.R.; Zhang, Y.; Wang, Z.Q. Fusion method of FADS/INS based on variable gain complementary filtering. *Transducer Microsyst. Technol.* **2022**, *41*, 38–41.
27. Liu, H.Y.; Ye, W.S.; Huang, S.; Chen, Z.M. Distributed navigation method based on inertial sensor network. *Syst. Eng. Electron.* **2014**, *36*, 1374–1379.
28. Dai, X.Q. The Key Technology Research on Redundant Inertial Navigation System. Doctor's Thesis, Harbin Engineering University, Harbin, China, 2015.
29. Qin, Y.Y. Strapdown inertial navigation system. In *Inertial Navigation*, 3rd ed.; Zhu, X.Y., Ed.; Science Press: Beijing, China, 2014; pp. 250–350.
30. Yang, Y.J. Research of Distributed Navigation Based on Inertial Network. Master's Thesis, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 2013.
31. Allerton, D.J.; Jia, H. Distributed data fusion algorithms for inertial network system. *IET Radar Sonar Navig.* **2008**, *2*, 51–62. [CrossRef]
32. Yang, Y.J.; Liu, H.Y.; Wang, H.N. Transfer Alignment for Inertial Sensor Network. *Sci. Technol. Eng.* **2012**, *35*, 9608–9616.
33. Yan, G.M. Strapdown inertial navigation update algorithm and error analysis. In *Strapdown Inertial Navigation Algorithm and Integrated Navigation Principle*, 1st ed.; Li, A.M., Ed.; Northwest University of Technology Press: Xi'an, China, 2019; pp. 78–105.
34. Zheng, C.J. Research on Algorithms of Flush Air Data Sensing System. Master's Thesis, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 2005.
35. Song, X.Y. Research on Algorithms and Applications of Flush Air Data Sensing System. Master's Thesis, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 2008.
36. Shen, G.Q. Error Analysis of Flush Air Data Sensing System and Research on Methods of Error Eliminating. Master's Thesis, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 2012.
37. Sun, D.F. Research on FADS System Application Profile of Flying Wing Aircraft Layout. Master's Thesis, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 2014.
38. Qin, Y.M.; Zhang, C.; Dong, J.G. Experimental study on flush air data sensing system calibration in wind tunnel. *Acta Aerodyn. Sin.* **2015**, *33*, 488–492.
39. Cheng, G.Q.; Liu, W.Y.; Dou, X.X. Flush air data sensing system design for air breathing air-to-air missile. *Sci. Sin.* **2016**, *45*, 1193–1206.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.