

Article

An Adaptive Fast Incremental Smoothing Approach to INS/GPS/VO Factor Graph Inference

Zhaoxu Tian , Yongmei Cheng and Shun Yao

School of Automation, Northwestern Polytechnical University, Xi'an 710072, China

* Correspondence: tianzx1991@mail.nwpu.edu.cn

Abstract: In response to asynchronous and delayed sensors within multi-sensor integrated navigation systems, the computational complexity of joint optimization navigation solutions persistently rises. This paper introduces an adaptive fast integrated navigation algorithm for INS/GPS/VO based on factor graph. The factor graph model for INS/GPS/VO is developed subsequent to individual modeling of the Inertial Navigation System (INS), Global Positioning System (GPS), and Visual Odometer (VO) using the factor graph model approach. Additionally, an Adaptive Fast Incremental Smoothing (AFIS) factor graph optimization algorithm is proposed. The simulation results demonstrate that the factor-graph-based integrated navigation algorithm consistently yields high-precision navigation outcomes even amidst dynamic changes in sensor validity and the presence of asynchronous and delayed sensor measurements. Notably, the AFIS factor graph optimization algorithm significantly enhances real-time performance compared to traditional Incremental Smoothing (IF) algorithms, while maintaining comparable real-time accuracy.

Keywords: integrated navigation; factor graph; inertial navigation system; global positioning system; visual odometry



Citation: Tian, Z.; Cheng, Y.; Yao, S. An Adaptive Fast Incremental Smoothing Approach to INS/GPS/VO Factor Graph Inference. *Appl. Sci.* **2024**, *14*, 5691. <https://doi.org/10.3390/app14135691>

Academic Editors: Manuel Armada and Roemi Fernandez

Received: 20 May 2024

Revised: 24 June 2024

Accepted: 27 June 2024

Published: 29 June 2024



Copyright: © 2024 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

Recently, multi-sensor integrated navigation has emerged as the primary approach for achieving long-term, high-precision navigation in diverse and complex environments [1–4]. Navigation systems, including the Inertial Navigation System (INS), Global Positioning System (GPS), and Visual Odometer (VO), find widespread applications in unmanned aerial vehicle navigation systems. The INS boasts high short-term accuracy, robust autonomy, comprehensive data, frequent data updates, and impressive resistance to interference. However, it is prone to accumulating and diverging positioning errors over time [5,6]; GPS offers high positioning accuracy, yet the functioning of GPS receivers is significantly influenced by carrier maneuvers, susceptible to interference, and subject to human control; VO estimates carrier pose changes using visual information at consecutive instances, effectively mitigating the accumulation of INS errors in the absence of GPS. Consequently, achieving long-term, high-precision, and stable navigation is feasible by leveraging information from multiple navigation devices to create an integrated navigation system [7–11].

In general, the integrated navigation system comprises multiple navigation subsystems. Divergences in data frequency among various navigation sources and delays in the arrival of navigation data at the fusion center, attributed to the distinct working principles of each navigation method, necessitate heightened demands on information fusion methodologies within integrated navigation. Presently, the predominant information fusion algorithms for integrated navigation heavily depend on the Kalman filter (KF) and have undergone numerous refinements to enhance navigation accuracy and stability. Furthermore, the KF mandates the establishment of a state space model for the integrated navigation system. Adjustments to this model are required when the efficacy of sensors in the integrated navigation system undergoes changes, markedly amplifying the intricacy

of algorithmic design [12–14] and constraining the potential utilization of more effective sensor information to enhance navigation performance. Additionally, KF generally confines its estimation to the current state at the present time, and the formulated measurement equation solely delineates the mapping relationship between measurement information and the current state. Complexities arise in accommodating measurement information, such as VO, which characterizes constraint relationships between diverse states at different times and proves challenging to integrate within the information fusion framework based on KF.

The factor graph represents a category of probability graph models tasked with encoding the probability relationships between navigation state variables and measurements obtained from navigation systems. The state variables are denoted as variable nodes, while each sensor measurement is identified as a factor node. The integration of measurements from diverse sensors, potentially operating asynchronously, necessitates the establishment of connections between the factors derived from these measurements and the corresponding nodes within the factor graph [15,16]. The factor graph model exhibits notable flexibility, enabling a plug-and-play functionality for sensors, a feature that has garnered considerable attention in the field of navigation [17–19]. In contrast to state estimation methods in filter-based integrated navigation systems, factor-graph-based integrated navigation system state estimation methods abstain from marginalizing historical states. Instead, they undertake the estimation of all states within the integrated navigation system, a task akin to solving a nonlinear least squares problem.

Batch optimization is extensively employed for information fusion in integrated navigation systems [20–22]. However, as batch optimization recalculates all variables of the nonlinear least squares problem, computational complexity escalates over time. In reference [23], the fixed-lag smoothing algorithm is implemented to address this issue by maintaining several past states in a sliding window. To sustain a consistent and manageable computational cost, all variables within the sliding window undergo recalculation upon obtaining a new measurement. This operation, however, proves unnecessary since some states remain unchanged. The fixed-lag smoothing algorithm is deemed sub-optimal in comparison to batch optimization unless the smooth lag is sufficiently extensive. Furthermore, the fixed-lag smoother marginalizes variables outside of the smoothing lag, posing challenges for the navigation system in establishing effective closed-loop factors. This compromises the outstanding performance of factor-graph-based integrated navigation.

The incremental smoothing (IF) technology is employed to efficiently and swiftly compute the complete navigation solution [24–27]. In contrast to batch optimization and fixed-lag smoothing, the incremental smoothing algorithm updates only a subset of nodes altered in the factor graph during each iteration. However, the incremental smoothing algorithm incrementally computes the system's square root information matrix and optimizes all state variables at each iteration. Despite its faster calculation speed compared to batch optimization, the dimension of state variables expands with the progression of navigation time, inevitably amplifying the computational complexity for each global optimization.

Given the above analysis, the processing of asynchronous and delayed measurements in traditional KF-based integrated navigation methods is relatively complex. Additionally, the current factor graph algorithms (including batch optimization, fixed-lag smoothing, and incremental smoothing) cannot meet the accuracy and real-time demands of integrated navigation systems. To address the mentioned problems, this paper proposes an improved INS/GPS/VO adaptive fast integrated navigation algorithm based on factor graph. The main contributions of this paper are as follows:

- (a) A novel factor-graph-based INS/GPS/VO integrated navigation model is formulated, which exhibits commendable efficacy in managing dynamic alterations in sensor effectiveness, as well as accommodating asynchronous and delayed sensors;
- (b) The adaptive fast incremental smoothing algorithm is proposed, which demonstrates superior performance compared to the traditional incremental smoothing approach.

This paper is organized as follows. The proposed factor graph model for the INS/GPS/VO integrated navigation system is presented in Section 2.1. The AFIS factor graph optimization algorithm is derived in Section 2.2. The simulation results and analysis are shown in Section 3. The discussion is shown in Section 4. The conclusions are drawn in Section 5.

2. Materials and Methods

2.1. Factor Graph Model of INS/GPS/VO Integrated Navigation System

2.1.1. INS Factor Graph Model

The efficacy of INS heavily relies on the high-frequency measurement of carrier motion states facilitated by the Inertial Measurement Unit (IMU). This frequency is typically considerably higher than the output frequency of the navigation system. Consequently, the INS measurement data contain a substantial amount of redundant information compared to the navigation system output. Therefore, the idea of equivalent factors can be adopted [28] when establishing the INS factor graph model. State variables requiring output are introduced as variable nodes. The IMU measurement data between consecutive time points of adjacent variable nodes is integrated and defined as equivalent factor nodes. This approach serves to reduce the update frequency of factor nodes, thereby enhancing the real-time performance of the algorithm.

Figure 1 shows the construction scheme of the INS factor graph model. First, the output frequency of navigation state variables is determined based on the actual system application requirements. Subsequently, the update period for equivalent factors is selected. Navigation state variables earmarked for output, identified as target variables, are incorporated as variable nodes into the factor graph. Additionally, the inertial sensor bias variable is introduced to the variable node to rectify the measurement error of the inertial sensor in IMU. The bias factor node is defined using the error propagation function. During an update period, the IMU measurement data is utilized to establish a pre-integration factor node, connecting it to the corresponding variable nodes in the factor graph.

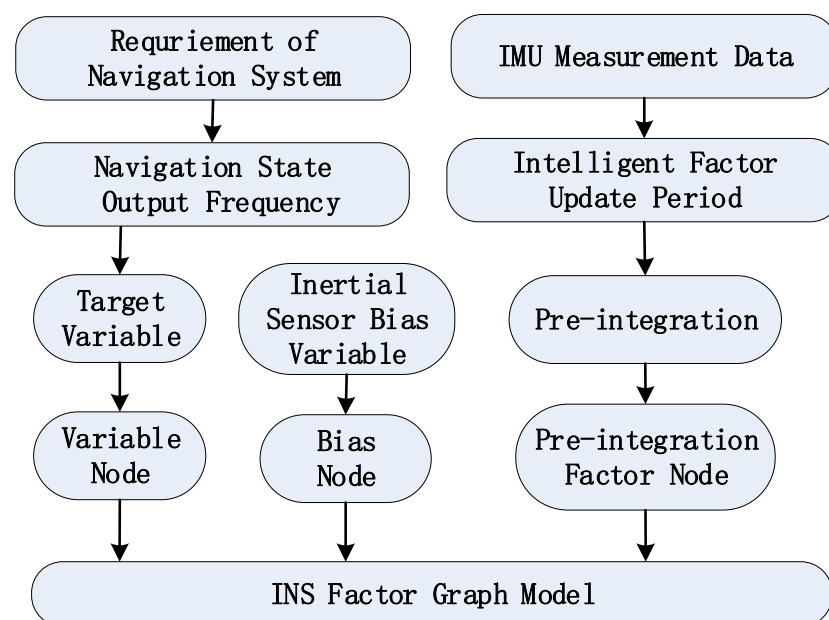


Figure 1. Construct scheme of INS factor graph model.

The INS functions as a dead reckoning system, amalgamating measurements of acceleration and angular velocity to derive information regarding the position, velocity, and attitude of the carrier. Utilizing the pre-integration method enables the synthesis of IMU data within an update cycle, establishing equivalent factor nodes. The integrated navigation system leverages auxiliary navigation methods to swiftly accomplish the initial alignment

of the INS, facilitating the acquisition of initial position, velocity, and attitude information of the carrier. Subsequently, by directly pre-integrating acceleration and angular velocity measurements within the body frame, significant reductions in computational complexity and improvements in the real-time performance of the algorithm are achieved [29]. The output time of the navigation state is denoted by t_k , $k = 0, 1, \dots$, and consequently, the equivalent factor update period will be $T = t_{k+1} - t_k$. By pre-integrating IMU measurement data within an update period $t_k \sim t_{k+1}$ under the body frame, the calculation formula for position, velocity, and attitude increments is obtained as

$$\begin{cases} \Delta p_{t_k+1}^{t_k} = \iint_{t_k}^{t_{k+1}} \left(C_{bt}^{bt_k} \left(f_t^b - \nabla_f \right) \right) dt^2 \\ \Delta v_{t_k+1}^{t_k} = \int_{t_k}^{t_{k+1}} \left(C_{bt}^{bt_k} \left(f_t^b - \nabla_f \right) \right) dt \\ \Delta \phi_{t_k+1}^{t_k} = \int_{t_k}^{t_{k+1}} \left(E_{bt}^{bt_k} \left(\omega_t^b - \nabla_\omega \right) \right) dt \end{cases} \quad (1)$$

where $C_{bt}^{bt_k}$ and $E_{bt}^{bt_k}$ are the rotation matrix and rotation rate matrix from the body frame of current time to the body frame of time t_k ; f_t^b and ω_t^b are the specific force and the angular acceleration in the body frame; ∇_f and ∇_ω are the bias variables of the accelerometer and gyroscope. According to the principle of INS dead reckoning, the carrier's position, velocity, and attitude at the time t_{k+1} can be further obtained as

$$\begin{cases} p_{t_{k+1}}^n = p_{t_k}^n + (t_{k+1} - t_k) v_{t_k}^n + \frac{1}{2} (t_{k+1} - t_k)^2 g^n + C_{bt_k}^n \Delta p_{t_k+1}^{t_k} \\ v_{t_{k+1}}^n = v_{t_k}^n + (t_{k+1} - t_k) g^n + C_{bt_k}^n \Delta v_{t_k+1}^{t_k} \\ \phi_{t_{k+1}}^n = \phi_{t_k}^n + C_{bt_k}^n \Delta \phi_{t_k+1}^{t_k} \end{cases} \quad (2)$$

where g^n is gravity vector.

Equation (2) is the discrete form of the INS system state equation, which can be uniformly expressed as

$$\begin{aligned} \hat{x}_{k+1} &= h(x_k, \alpha_k, z_k^{IMU}) \\ \hat{\alpha}_{k+1} &= g(\alpha_k) \end{aligned} \quad (3)$$

where x_k represents the navigation state variable, including position, velocity, and attitude angle; α_k represents the bias variable of the inertial sensor; z_k^{IMU} represents the measured value of the inertial sensor; $\hat{x}_{k+1}$ is the predicted value of the navigation state value obtained from the state equation at time t_{k+1} ; Conceptually, each equation in (3) defines distinct factors: an IMU factor f^{IMU} linking the navigation nodes x_k , $\hat{x}_{k+1}$ and the bias node α_k , alongside a bias factor f^{Bias} connecting the bias nodes α_k and $\hat{\alpha}_{k+1}$. In practice, consecutive IMU measurements are amalgamated into a singular factor, as depicted in Figure 2. Nonetheless, delineating explicit expressions for the conventional IMU factor f^{IMU} remains beneficial.

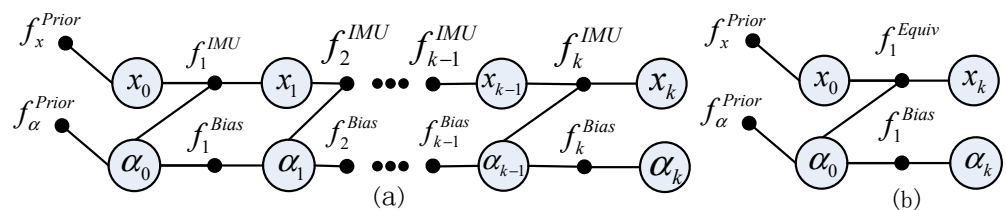


Figure 2. The comparison between IMU factor and equivalent IMU factor. (a) Factor graph model using IMU factor; (b) Factor graph model using equivalent IMU factor.

The conventional IMU factor f^{IMU} corresponding to a specific IMU measurement z_k^{IMU} is defined as follows. The IMU measurement z_k^{IMU} and the current estimate of the state x_k, α_k are utilized to forecast the values for the state $\hat{x}_{k+1}$. The discrepancy between this

prediction and the current estimate x_{k+1} is encapsulated as the error function represented by the factor

$$f_{k+1}^{IMU}(x_{k+1}, x_k, \alpha_k) \doteq d(x_{k+1} - h(x_k, \alpha_k, z_k)) \quad (4)$$

where the operator $d(\cdot)$ denotes a specific cost function. Upon the addition of the new variable node x_{k+1} to the graph, an appropriate reasonable initial value for x_{k+1} is necessary. This value can be derived, for instance, from the prediction $h(\hat{x}_k, \hat{\alpha}_k, z_k)$, where $\hat{x}_k$ and $\hat{\alpha}_k$ represent the most recent estimates of the state x_k and the measurement α_k , respectively.

Similarly, the bias factor associated with the modeled IMU errors is expressed as

$$f_{k+1}^{Bias}(\alpha_{k+1}, \alpha_k) \doteq d(\alpha_{k+1} - g(\alpha_k)) \quad (5)$$

with α_{k+1} and α_k represented as variable nodes in the factor graph.

Due to the high frequency of inertial measurement data (generally greater than 200 Hz), the inertial system has a high calculation frequency, which means that the number of variable nodes and corresponding cost functions in the navigation system factor graph model increase sharply, leading to a sharp increase in the computational complexity of joint optimization. Instead, consecutive IMU measurements can be combined into an equivalent IMU factor [29], relating between two distant navigation and calibration nodes, thereby avoiding introducing new variables into the optimization at IMU rate. The equivalent IMU factor is given by

$$f^{Equiv}(x_j, x_i, \alpha_k) \doteq d(x_j - h^{Equiv}(x_i, \alpha_i, \Delta x_{i \rightarrow j})) \quad (6)$$

where $\Delta x_{i \rightarrow j}$ relates between the navigation states at the two time instances x_i and x_j , comprising the accumulated change in position, velocity, and attitude. h^{Equiv} represents the nonlinear function that predicts x_j given x_i , α_i and $\Delta x_{i \rightarrow j}$.

The factor f^{Equiv} encapsulates all IMU readings spanning the temporal interval between t_i and t_j . However, integrating it into the factor graph entails introducing solely the variable x_i . This stands in juxtaposition to the conventional IMU factor scenario, where variables are added at the IMU sampling rate. Figure 2 shows the comparison between using the conventional IMU factor and using the equivalent IMU factor. The shown factor graph contains prior, IMU, equivalent IMU, and bias factors.

From Equations (3), (5), and (6), the equivalent IMU factor node $f_{k+1}^{Equiv}(x_{k+1}, x_k, \alpha_k)$ is connected to the state variable node x_k , bias variable node α_k , and state variable node x_{k+1} ; The bias factor node $f_{k+1}^{Bias}(\alpha_k, \alpha_{k+1})$ connects the bias variable node α_k and the bias variable node α_{k+1} . Figure 3a,b shows the factor graph models at two distinct time intervals. New nodes are continually integrated into the model as time progresses; therefore, the INS factor graph model is actually a dynamic graph model.

2.1.2. Prior Factor

Prior information $p(x_0)$ signifies that the navigation system possesses a predetermined understanding of the initial variables prior x_0 to real-time navigation. For instance, INS can determine the initial state accuracy of the carrier through initial alignment. The available prior information $p(x_0)$ can be subjected to additional factorization, resulting in discrete priors pertaining to relevant variables. Each of these priors is encapsulated by an individual prior factor. Termed as a unary factor, the prior factor is expressed as

$$f^{Prior}(x_0) \doteq d(x_0) \quad (7)$$

In the case of a Gaussian distribution, prior information is typically specified through a mean μ_0 and a covariance matrix Σ_0 .

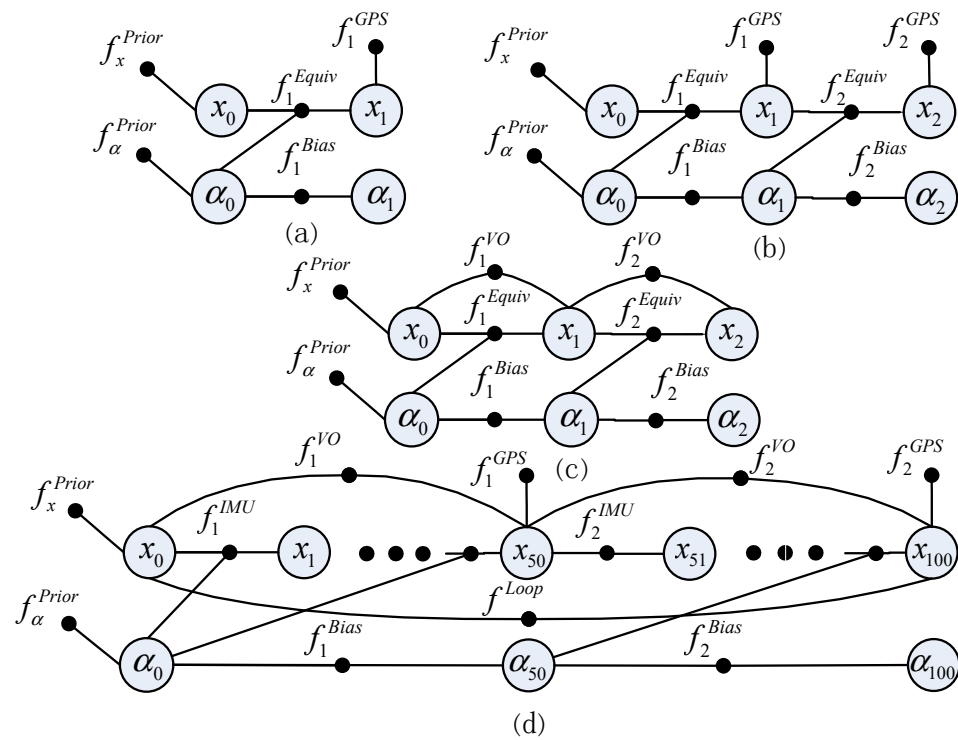


Figure 3. Factor graph model including factors and nodes at different times. (a,b) Factor graphs that include GPS, prior, equivalent IMU, and bias factors at time t_1 and t_2 ; (c) Factor graphs with VO, prior, equivalent IMU, and bias factors; (d) Factor graphs with GPS, VO, prior, IMU, and bias factors. Which is equivalent to the factor graph represented in (c), utilizing conventional IMU factors. Assuming that GPS and VO measurements are acquired every 50 IMU measurements. Additionally, bias nodes are incorporated at the same frequency as depicted in (c).

2.1.3. GPS Factor

The GPS measurement equation is

$$z_k^{GPS} = h^{GPS}(x_k) + n^{GPS} \quad (8)$$

where n^{GPS} represents the measurement noise, and h^{GPS} stands for the measurement function, which establishes the relationship between the measurement z_k^{GPS} and the carrier's position.

The GPS factor equation is a unary factor defined as

$$f^{GPS}(x_k) \doteq d(z^{GPS} - h^{GPS}(x_k)) \quad (9)$$

which is solely connected to the node at time t_k . The GPS factor f^{GPS} can be integrated into the graph along with an equivalent IMU factor f^{IMU} .

Figure 3b illustrates examples of factor graphs encompassing GPS measurements and data from various sensors operating at different rates. Additionally, it presents prior factor nodes associated with navigation and bias factor nodes.

2.1.4. VO Factor

When the true location of the target, as estimated by the stereo vision system, remains unknown, the vision system is limited to acquiring solely the relative position between the carrier and the target at the current time. Direct recovery of the carrier's positioning information is not possible. VO offers an indirect method to ascertain the relative relationship between the carrier's position state at two consecutive moments. This is achieved by utilizing the target as an intermediary reference position to estimate the relative position

concerning the same target at both adjacent moments. It is evident that the measurement equation established by the visual system is intricately connected to the temporal states during imaging. Despite the inability of this measurement equation to satisfy the prerequisites of the KF, it equivalently introduces a factor node into the system factor graph model within the factor-graph-based state estimation algorithm.

The VO measurement equation is given by

$$z^{VO} = h^{VO}(x_k, x_{k+1}) + n^{VO} \quad (10)$$

where n^{VO} is the measurement noise, $h^{VO}(x_k, x_{k+1})$ is the measurement function, z^{VO} is the measurement, x_k and x_{k+1} represent the state of VO at the associated time, respectively. The VO factor equation is a binary factor defined as

$$f^{VO}(x_k, x_{k+1}) = d(z^{VO} - h^{VO}(x_k, x_{k+1})) \quad (11)$$

Figure 3c illustrates the VO factor node added into the INS factor graph model.

2.1.5. Loop Factor

In general, VO encapsulates the relative relationship between the carrier's positional state at consecutive time points. When the carrier retraces its path to a prior position over a defined duration, constraints can be defined for these temporal state variables. Techniques like loop closure detection [30–32] offer a means to establish these constraints. This process is analogous to introducing a closed-loop factor node into the factor graph.

The Loop measurement equation is given by

$$z_{ij}^{Loop} = h^{Loop}(x_i, x_j) + n^{Loop} \quad (12)$$

where n^{Loop} represents the measurement noise, $h^{Loop}(x_i, x_j)$ denotes the measurement function, z_{ij}^{Loop} signifies the measurement. The Loop factor equation is defined by a binary factor

$$f^{Loop}(x_i, x_j) = d(z_{ij}^{Loop} - h^{Loop}(x_i, x_j)) \quad (13)$$

Figure 3d shows a factor graph in a multi-sensor and asynchronous scenario. The factor graph includes Loop, VO, GPS, IMU, bias, and prior factors. The IMU factor is replaced by the equivalent IMU factor as shown in Figure 2.

2.2. Factor Graph Inference Algorithm for Integrated Navigation

The factor graph model converts the solution of the maximum posterior estimation for integrated navigation state variables into factor graph inference. The inference algorithm employed for the factor graph is the sum-product algorithm [33], widely applied in practical engineering for precise or approximate reasoning, with documented success [34]. In light of the imperative for robust real-time performance in navigation systems, incremental inference technology [35] is employed. This approach is driven by the continuous expansion of the integrated navigation factor graph model, evolving dynamically over time.

2.2.1. Factor Graph Algorithm

In the factor graph model, the set of variable nodes is represented as X ; The set of factor nodes is represented as F ; The set of all edges of connecting nodes is represented as E . Thus, the factor graph can be represented as

$$G = (X, F, E) \quad (14)$$

Based on the theory of factor graphs, the factor graphs G describe the factorization of functions $f(X)$, which can be denoted as

$$f(X) = \prod_i f_i(X_i) \quad (15)$$

where X_i represents the set of variable nodes connected to factor nodes f_i , $X_i \subseteq X$.

The problem needs to be solved in the factor graph inference algorithm to find the optimal estimate value $\hat{X}$ of X , so that the function $f(X)$ takes the maximum value. The Maximum A Posteriori (MAP) estimate is expressed as

$$\hat{X} = \operatorname{argmax}_X \left(\prod_i f_i(X_i) \right) \quad (16)$$

For the factors defined in Equations (5)–(7), (9), (11), and (13), assuming a Gaussian noise model, the cost function can usually be defined as $d(a) \triangleq \exp\left(-\frac{1}{2}\|a\|_{\Sigma}^2\right)$. Here, $\|a\|_{\Sigma}^2 \doteq a^T \Sigma^{-1} a$ represents the squared of the Mahalanobis distance, Σ represents the covariance matrix. Then, the factor f_i assumes the following form

$$f_i(X_i) \propto \exp\left(\frac{1}{2}\|a_i\|_{\Sigma_i}^2\right) \quad (17)$$

Therefore, taking the maximum value of the above equation is equivalent to the least squares problem given by

$$\operatorname{argmin}_X (-\log f(X)) = \operatorname{argmin}_X \frac{1}{2} \sum_i \|a_i\|_{\Sigma_i}^2 \quad (18)$$

The expression of factor nodes in factor graph models is usually nonlinear and can be linearized through Taylor expansion. When the initial estimation $\hat{X}_k$ of the system state X_k is given, the update increment Δ of state X_k can be transformed into solving a standard least squares solution

$$\Delta^* = \operatorname{argmin}_{\Delta} (-\log f(\Delta)) = \operatorname{argmin}_{\Delta} \|J(\hat{X}_k)\Delta - b(\hat{X}_k)\|^2 \quad (19)$$

where $\Delta = \{\delta X_k\}$ represents the update increment for the current system variable X_k , $J(\hat{X}_k)$ and $b(\hat{X}_k)$ are the Jacobian matrices and residuals of the cost functions at X_k ; then, the Gaussian Newton iteration method is used to calculate the optimal estimation value $\hat{X}_k$.

When the GPS and VO measurement frequency are different from the system output frequency and there is a time delay, it is only necessary to define the GPS or VO measurement values at time t_k as factor nodes and connect them to the INS factor graph model. If GPS measurements are missing, the corresponding GPS factor node is not connected to the INS factor graph model at time t_k . Figure 4b illustrates the INS/GPS/VO factor model, reflecting the lower frequencies of GPS and VO measurements compared to the system output frequency, and the occurrence of missing GPS measurements.

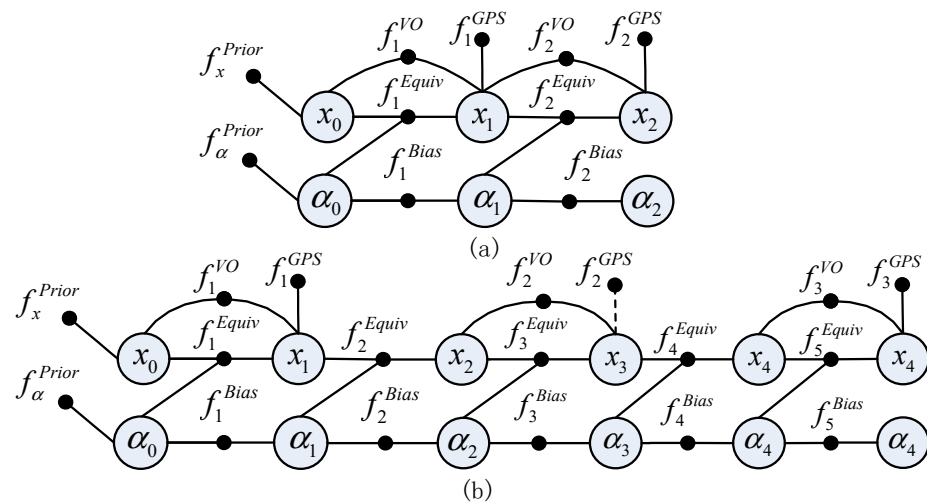


Figure 4. Factor graph model for asynchronous, delayed, and missing measurements. (a) GPS and VO measurement frequency are the same as the system output frequency; (b) GPS and VO measurement frequency lower than system output frequency, and the missing GPS measurement (measurement f_2^{GPS} is missing).

The process of constructing the factor graph model for the integrated navigation system described above reveals that the factor nodes corresponding to each sensor are independent and do not affect each other in the factor graph model. For instance, in the case of GPS effectiveness change, if GPS fails suddenly, we simply suppress the addition of GPS factor nodes and transition to an INS/VO factor graph model. Conversely, when GPS becomes effective, adding GPS factor nodes reinstates the INS/GPS/VO factor graph model. The use of factor graph models to describe integrated navigation problems can flexibly handle dynamic changes in sensor effectiveness, and the integrated navigation algorithm based on factor graphs has strong scalability and flexibility.

2.2.2. IS Algorithm

The factor graph model intuitively manifests as the increasing number of variables in X over time. It is clearly unable to meet the real-time requirements of the system, if every factor graph inference involves re-solving the optimal estimate for all variables X . Taking the factor graph model shown in Figure 3a as an example, variable elimination [36] is adopted and all nodes are eliminated from the factor graph by the order of $x_0 \rightarrow \alpha_0 \rightarrow x_1 \rightarrow \alpha_1$, as shown in Figure 4a. A Bayesian network is a directed acyclic graph that encodes the conditional densities of each variable. The Bayesian network is equivalent to the upper triangular matrix R resulting from the Jacobian matrix factorization $J = QR$ [37], thereby representing the square root information matrix of the system. The corresponding Jacobian matrix J_{t_1} and the upper triangular matrix R_{t_1} at time t_1 , as shown in Figure 3a and obtained through a QR factorization of $J_{t_1} = QR_{t_1}$, take the following form

$$J_{t_1} = \begin{bmatrix} x_0 & \alpha_0 & x_1 & \alpha_1 \\ \times & & & \\ \times & \times & \times & \\ & \times & & \\ & \times & & \times \\ & & \times & \end{bmatrix}, R_{t_1} = \begin{bmatrix} x_0 & \alpha_0 & x_1 & \alpha_1 \\ \times & \times & \times & \\ & \times & \times & \times \\ & & \times & \times \\ & & & \times \end{bmatrix} \quad (20)$$

where $\times$ represents non-zero elements in matrix entries J_{t_1} and R_{t_1} . Each linearized factor is denoted by corresponding rows in the Jacobian matrix J_{t_1} and the upper triangular matrix R_{t_1} corresponds to the square root information matrix of the system.

As seen, part of the Bayesian network is changed when new nodes are added, assuming that the new measurement w and residual r are added at time t_2 . Compared to the factor graph at time t_1 , the new state variable node x_2 and IMU bias nodes α_2 are added into the factor graph at time t_2 . Additionally, GPS, bias, and equivalent IMU factors are also introduced. The new factor graph is shown in Figure 3b, with the new added factors f_2^{Equiv} and f_2^{GPS} . The updated Jacobian matrix is $J_{t_2} = \begin{bmatrix} J_{t_1} \\ w \end{bmatrix}$ and the corresponding residual is $b^* = \begin{bmatrix} b \\ r \end{bmatrix}$; thus, the upper triangular matrix R_{t_2} is given by

$$R_{t_2} = \begin{bmatrix} Q^T & \\ & I \end{bmatrix} \begin{bmatrix} J_{t_1} \\ w \end{bmatrix} = \begin{bmatrix} R_{t_1} \\ w \end{bmatrix}, d^* = \begin{bmatrix} Q^T & \\ & I \end{bmatrix} \begin{bmatrix} b \\ r \end{bmatrix} = \begin{bmatrix} d \\ r \end{bmatrix} \quad (21)$$

where d^* is the residual term of the upper triangular matrix R_{t_2} after QR factorization of the Jacobian matrix J_{t_2} . The updated Jacobian matrix J_{t_2} and the upper triangular matrix R_{t_2} can be shown as

$$J_{t_2} = \begin{bmatrix} x_0 & \alpha_0 & x_1 & \alpha_1 & x_2 & \alpha_2 \\ \times & & \times & & & \\ \times & \times & & & & \\ & \times & & & & \\ & \times & & & & \\ & & \times & & & \\ & & \overline{\times} & \overline{\times} & \overline{\times} & \\ & & & \overline{\times} & & \overline{\times} \\ & & & & \overline{\times} & \overline{\times} \end{bmatrix}, R_{t_2} = \begin{bmatrix} x_0 & \alpha_0 & x_1 & \alpha_1 & x_2 & \alpha_2 \\ \times & \times & \times & & & \\ & \times & \times & \times & & \\ & & \overline{\times} & \overline{\times} & \overline{\times} & \\ & & & \overline{\times} & \overline{\times} & \overline{\times} \\ & & & & \overline{\times} & \overline{\times} \\ & & & & & \overline{\times} \end{bmatrix} \quad (22)$$

where the new or changed non-zero variables are denoted as $\overline{\times}$. Obviously, two new variables x_2 and α_2 are added into the factor graph and the corresponding Jacobian matrix has two new rows. Compared to the matrix R_{t_1} , only part of the variables are changed so that it is not necessary to recalculate the whole matrix R_{t_2} . The corresponding Bayes net is described in Figure 5b, and the changed variables are shown in red.

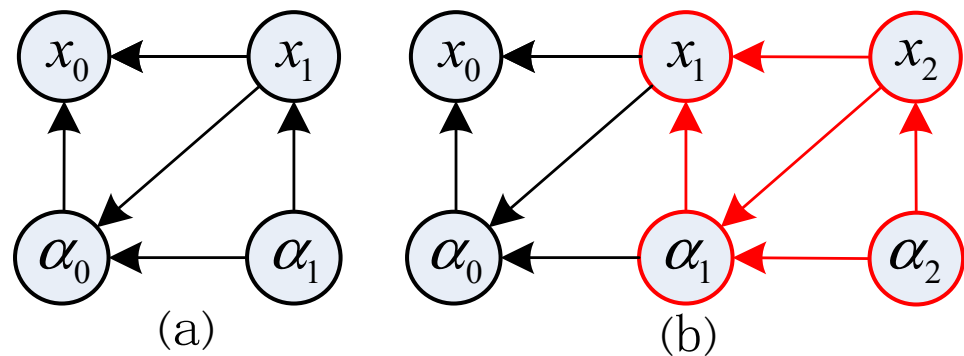


Figure 5. Bayesian network corresponding to the factor graph at different times. (a) Bayesian network corresponding to the elimination of the factor graph at time t_1 ; (b) Incorporating new measurements into a factor graph at time t_2 , introducing additional nodes as x_2 and α_2 .

From Equations (20)–(22), non-zero terms emerge below the diagonal of R_{t_1} when incorporating new measurements into the current measurement Jacobian matrix J_{t_1} and upper triangular matrix R_{t_1} . Exploring the prospect of localized QR decomposition on the upper triangular matrix segments with non-zero terms below the diagonal yields a fresh upper triangular matrix R_{t_2} . Analogously, the residual terms d^* are derived through corresponding operations. Figure 6 shows a schematic diagram of the incremental matrix R_{t_2} calculation.

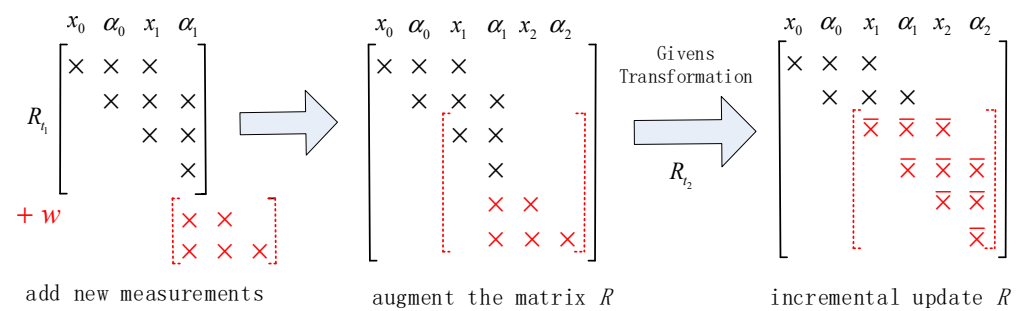


Figure 6. Schematic diagram of IS (the changed variables are shown in red).

Owing to the inherent sparsity of the upper triangular matrix R and the typical scenario where newly acquired measurement rows predominantly pertain to a limited set of variables, the region for local QR decomposition remains consistently compact and fixed with each iteration. Consequently, the majority of elements in the matrix R remain unaltered pre and post-conversion. The incremental smoothing algorithm, grounded in batch optimization principles, maximizes the leverage of the sparsity in the system's measurement Jacobian matrix J . It achieves this by progressively solving the matrix R , obviating the necessity for repeated decomposition of the measurement Jacobian matrix J during each batch optimization cycle. Consequently, the incremental smoothing algorithm effectively curtails the computational complexity of the system, ensuring precision in solutions and bolstering real-time performance.

2.2.3. AFIS Algorithm

From the analysis of incremental QR decomposition in Equation (21), non-zero terms emerge beneath the diagonal of the associated upper triangular matrix R upon the incorporation of new measurements into the integrated navigation system, necessitating local QR decomposition. However, the recently introduced measurement solely pertains to a subset of variables, influencing only the corresponding section of these variables in the upper triangular matrix (as shown in the red segment of the matrix R_{t_2} in Figure 5). Consequently, continuous trajectory smoothing is deemed unnecessary, suggesting a more judicious approach of optimizing the impacted variables directly during the resolution of the joint optimization equation (smooth when the system has requirement). This optimization retains the integrity of the system's matrix R . The matrix R of the entire system proves instrumental in executing global optimization, especially with the introduction of a loop factor in the system (optimizes variables within the closed loop) or when a navigation task requires comprehensive optimization of all navigation variables at a specific juncture (optimizes all navigation variables). Treating the matrix R corresponding to the entire system as a network enables a swift execution of the optimization algorithm, employing a blend of adaptive local and global optimization. This amalgamation is aptly named the Adaptive Fast Incremental Smoothing (AFIS) algorithm, elucidated in the schematic diagram presented in Figure 7.

At each moment, when new sensor measurements are added to the integrated navigation system, corresponding factor nodes and variable nodes need to be added to the factor graph model, which is equivalent to adding new rows to the linearized least squares equations of the system. Due to the fact that newly added sensors only affect their adjacent state variables, local QR decomposition can be used to update the changing parts of the square root mean information matrix R in the system. The state variables that generate changes can form an adaptive window, and only the state variables X' within that window are optimized and solved at each moment. Global state variables optimization is performed according to the system requirements. The system requirements here can be manually set, such as conducting global optimization every certain time or completing a certain maneuver task, depending on the computational requirements for algorithm complexity. The AFIS program flow chart is shown in Figure 8.

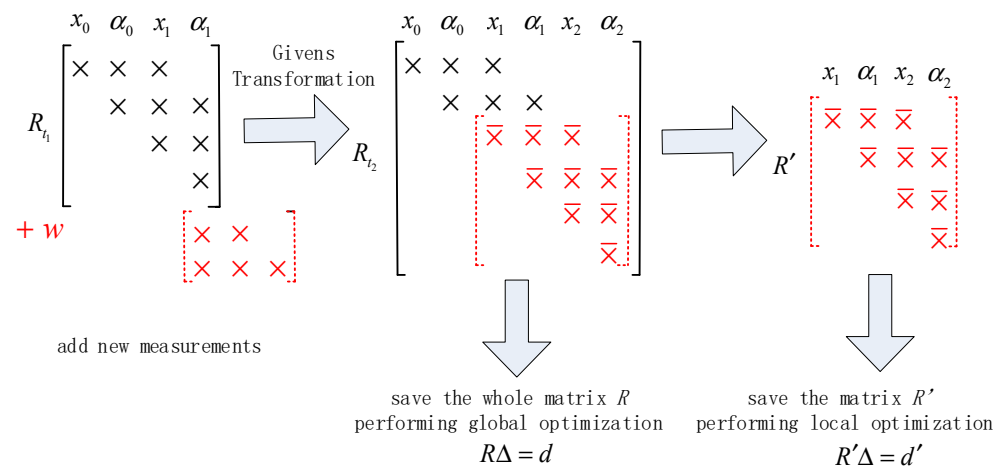


Figure 7. Schematic diagram of AFIS (the changed variables are shown in red).

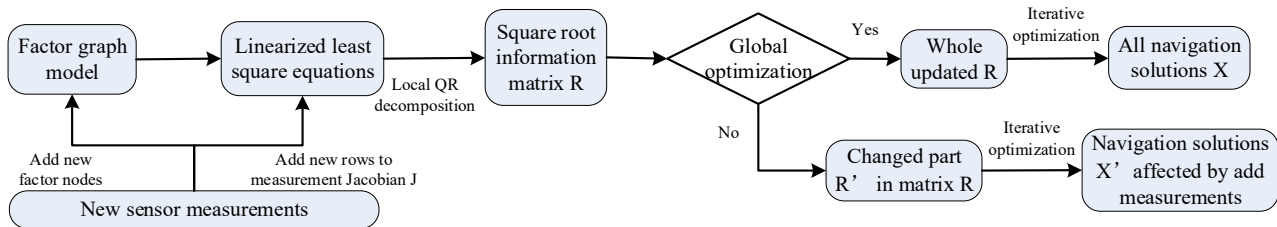


Figure 8. Program flow chart of the AFIS algorithm.

It can be seen that the AFIS algorithm is based on the foundations of the IS algorithm, focusing on extracting the variable regions directly influenced by measurements for subsequent local optimization. Clearly, the AFIS algorithm yields a notable reduction in the computational complexity of optimization solutions, thereby enhancing the real-time performance of the system. This algorithm not only exhibits adaptability in selectively choosing and optimizing the network based on the added type of measurement but also accommodates closed-loop factors, enabling it to undertake global optimization in alignment with navigation requirements.

3. Results

The simulation analysis is executed in the MATLAB 2016b environment on a desktop computer with an Inter® Core i5 3.3 GHz CPU (Intel, Santa Clara, CA, USA), with specific simulation conditions outlined as follows:

- (1) Scenario 1: The simulation commences from a starting position with latitude 34.246° , longitude 108.91° , and altitude set at 380 m. The initial speed is 0 m/s, heading north. Following a sequence of actions, including acceleration, turning, and deceleration, the trajectory reaches a destination specified by latitude 34.244° , longitude 108.9° , and altitude of 380 m. The maximum speed during the trajectory is 20 m/s, and the total flight time is 250 s;
- (2) Scenario 2: The simulation begins at a starting position defined by latitude 34.246° , longitude 108.91° , and altitude of 380 m, with an initial speed of 0 m/s and heading north. Executing actions such as acceleration, turning, and deceleration, the trajectory reaches the destination defined by latitude 34.244° , longitude 108.9° , and altitude of 380 m, then returns to the initial position. The maximum speed during this trajectory is 20 m/s, and the total flight time is 400 s;
- (3) The IMU initialization data can be acquired through the application of inertial navigation's differential equation based on the designed trajectory information. The constant

- drift and random drift of the gyroscope in INS is $1^\circ/\text{h}$ and $0.1^\circ/\text{h}$. The constant bias and random bias of the accelerometer is $100\ \mu\text{g}$ and $10\ \mu\text{g}$;
- (4) GPS measurement noise is characterized by a mean square error of 2 m for longitude and latitude, 5 m for height, and 0.2 m/s for speed in all three directions;
 - (5) The initial position of VO is the same as that of the carrier, and the VO position measurement noise is quantified by a mean square error of 1 m;
 - (6) The mean square error of the measurement noise for the Loop factor is 2 m, and the measurement equation is shown in Section 2.1.5;
 - (7) INS sampling frequency is set at 200 Hz, the GPS update frequency at 1 Hz, and the VO output frequency at 1 Hz.

3.1. Simulation of Sensor Effectiveness with Dynamic Changes

This section focuses on integrated INS/GPS navigation within Scenario 1, assuming a synchronous output frequency between INS and GPS, with the GPS measurements being unavailable during the period from 90 s to 160 s. The objective is to compare the fusion accuracy of the Kalman filtering (KF) and factor-graph-based (FG) method without time delay in measurement. The evaluation specifically examines the factor graph algorithm's capacity to accommodate dynamic changes in sensor effectiveness. Figure 9 shows the real-time position error of both KF and FG.

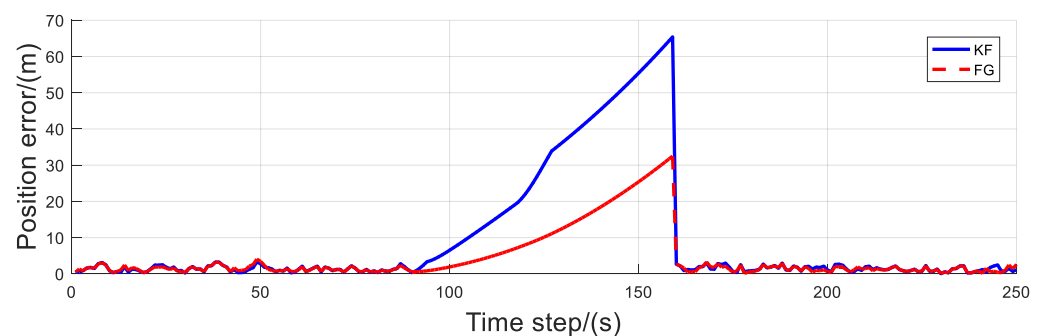


Figure 9. Position error of KF and FG with dynamic changes in GPS effectiveness.

In this experiment, GPS measurements were not available from 90 to 160 s, and the efficacy of sensors in the integrated navigation system underwent two changes. The integrated navigation system transitions from INS/GPS mode to INS individual mode when the GPS measurements are unavailable at 90 s. The inclusion of factor node f^{GPS} is inhibited in the factor graph model, resulting in a rapid accumulation of position errors due to the lack of INS correction. Subsequently, as the GPS signal becomes effective again at 160 s, the integrated navigation system shifts from INS individual mode to INS/GPS mode, and the factor node f^{GPS} is reintroduced to the factor graph model, correcting the position error.

Upon analysis, it is evident that during transitions between INS and INS/GPS modes prompted by changes in sensor effectiveness, only factor nodes associated with invalidated sensors need to be suppressed or new factor nodes related to effective sensors need to be added to the factor graph model. No additional adjustments are required, ensuring a seamless switch between different modes of the integrated navigation system and enhancing its overall robustness.

3.2. Simulation of Sensor Asynchrony and Delay

Under simulation scenario 1, assuming stable and effective measurement information from all sensors within the INS/GPS integrated navigation system, we established various asynchronous and delay conditions between the GPS measurement frequency and the INS output frequency. This aimed to assess the factor graph algorithm's capability in managing sensor asynchrony and delay. The term "asynchronous" denotes the discrepancy in output

frequencies between INS and GPS. We assumed a time delay in GPS measurements and configured delay times as $dt = 0.1$ s, $dt = 0.3$ s, and $dt = 0.5$ s, respectively. We subsequently compared these results with the integrated navigation of FG and KF without delay ($dt = 0$ s). The simulation results are shown in Figure 10 and Table 1.

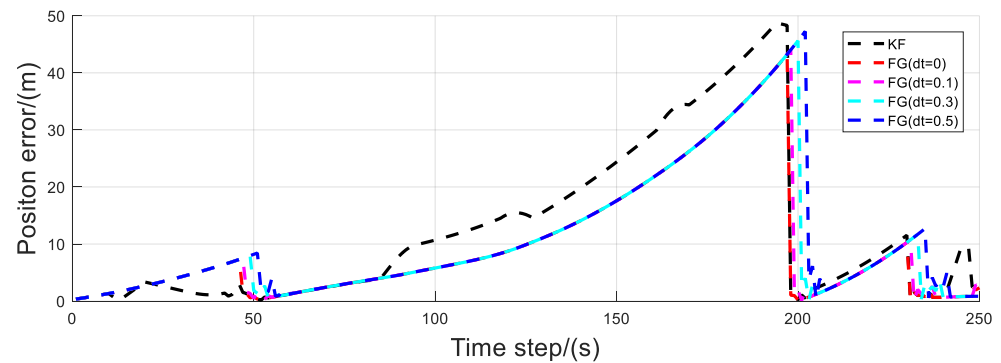


Figure 10. Position error of KF and FG in different measurement delays.

Table 1. RMSE statistical result of KF and FG in different measurement delays.

Integrated Navigation Method	Without Delay		dt = 0.1 s	dt = 0.3 s	dt = 0.5 s
	KF	FG	FG	FG	FG
Radial Error (m)	18.32	14.27	14.57	15.17	15.81

From Figure 10 and Table 1, it is evident that the positional error increases to a certain extent in the presence of asynchrony and delay. This occurs primarily because the system output frequency surpasses the GPS measurement frequency, compelling reliance solely on INS without GPS measurements. Notably, the FG algorithm exhibits a lesser sensitivity to delay times, sustaining comparable accuracy to the KF algorithm even in the absence of delay measurements. The FG algorithm requires no adjustment for sensor asynchrony and delay. Varied delay measurements merely alter the sequencing of corresponding cost functions in deriving global cost functions, without affecting the optimization results. Instead, sensor measurement values received must be systematically denoted as factor nodes, progressively connected to corresponding variable nodes in the factor graph. Subsequently, the incremental inference algorithm is employed to deduce the factor graph, yielding the optimal estimate of the navigation state.

3.3. AFIS Algorithm Analysis

The section primarily examines the accuracy and real-time performance of the adaptive fast incremental smoothing factor-graph-based (AFIS-FG) algorithm, batch optimization factor-graph-based (Batch-FG) algorithm, fixed-lag smoothing factor-graph-based (FixLag-FG) algorithm, and incremental smoothing (IS-FG) algorithm. The IS-FG algorithm and our proposed AFIS-FG algorithm are shown in Sections 2.2.2 and 2.2.3, respectively. The Batch-FG algorithm recalculates all state variables of the nonlinear least squares problem, which is shown in references [20–22]. The FixLag-FG algorithm [23] maintains several past states in a sliding window, with all variables within the sliding window undergoing recalculation upon obtaining a new measurement, while FixLag-FG (5) means the length of the sliding window is 5. The simulation aims at the INS/GPS/VO integrated navigation in scenario 1 and scenario 2. The simulation results are shown in Figures 11 and 12, Tables 2 and 3.

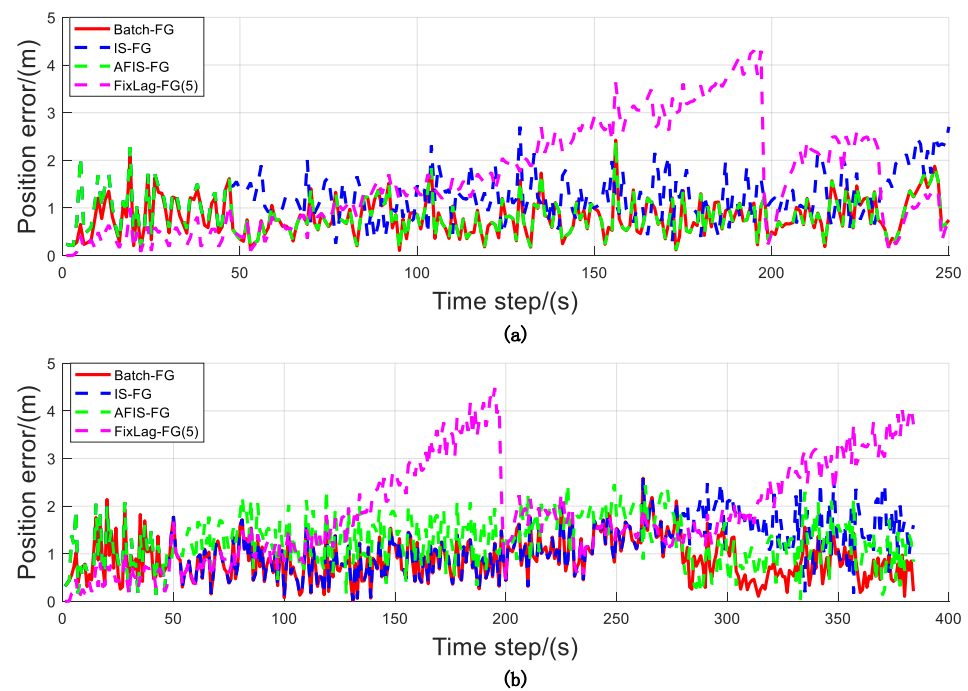


Figure 11. Radial position error of different algorithms in different scenarios. (a) Position error in scenario 1; (b) Position error in scenario 2.

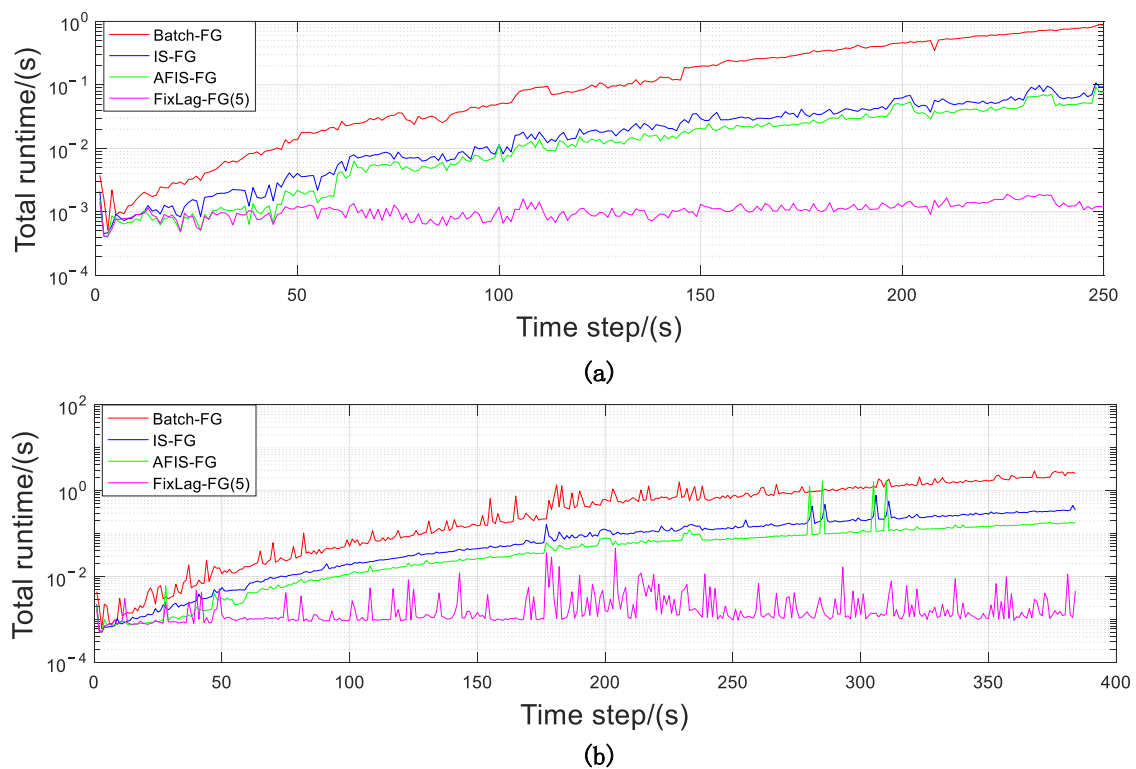


Figure 12. Comparison of total runtime of different algorithms in different scenarios. (a) Total runtime in scenario 1; (b) Total runtime in scenario 2.

Table 2. RMSE statistical result of different algorithms in two different scenarios.

Optimization Algorithm		Batch-FG	IS-FG	AFIS-FG	FixLag-FG (5)
Radial Error (m)	Scenario 1	0.907	1.308	0.940	1.942
	Scenario 2	1.235	1.570	1.554	2.588

Table 3. Total runtime statistical result of different algorithms in two different scenarios.

Optimization Algorithm		Batch-FG	IS-FG	AFIS-FG	FixLag-FG (5)
Time (s)	Scenario 1	54.6	7.4	4.7	0.2665
	Scenario 2	257.1	32.5	20.9	0.3971

From Figure 12 and Table 3, it is evident that FixLag (5) exhibits the shortest optimization time, clocking in at 0.2665 s in scenario 1 and 0.3971 s in scenario 2. This efficiency arises from its exclusive focus on optimizing and updating local state variables within the smoothing window at each moment. In contrast, the optimization times of Batch-FG, IS-FG, and AFIS-FG gradually escalate as simulation time progresses. Batch-FG stands out with the lengthiest optimization time, recorded at 54.6 in scenario 1 and 257.1 s in scenario 2. This is attributed to Batch-FG's overarching approach, involving unified processing of all received measurements at all times. This approach is akin to batch processing, entailing the update of the system's upper triangular matrix R at each time, followed by global optimization.

IS-FG secures the second shortest optimization time, totaling 7.4 s in scenario 1 and 32.5 s in scenario 2. This is a consequence of IS-FG's incremental update of the upper triangular matrix R of the system at each moment, a departure from the Batch-FG's approach. AFIS-FG follows with the third shortest optimization time, amounting to 4.9 s in scenario 1 and 20.9 s in scenario 2. AFIS-FG employs incremental updates to the upper triangular matrix of the system upon receiving measurements, focusing its optimization on locally affected state variables. Additionally, global optimization is executed by retaining the upper triangular matrix in the presence of a loop factor in the factor graph.

The precision analysis from Figure 11 and Table 2 reveals that the accuracy achieved by IS-FG and AFIS-FG is comparable to that of Batch-FG, surpassing the performance of FixLag-FG (5). This discrepancy arises due to the distinct optimization strategies employed by these algorithms. Specifically, Batch-FG, IS-FG, and AFIS-FG optimize global variables, whereas FixLag-FG (5) exclusively focuses on variables within a smoothing window. Notably, during instances of a loop factor in the system (as evident in the 310 s segment in Figure 11b), Batch-FG, IS-FG, and AFIS-FG leverage this loop factor to rectify the current time state, thereby enhancing real-time navigation accuracy. In contrast, FixLag-FG (5) lacks this capability.

4. Discussion

In summary, this paper proposes a factor-graph-based INS/GPS/VO integrated navigation method. Compared with traditional filtering-based (such as KF) integrated navigation methods, the factor-graph-based integrated navigation method can achieve plug-and-play functionality when facing asynchronous and delayed sensor measurements, and can handle multi-state correlated measurements such as VO, while also having high navigation accuracy. Sections 3.1 and 3.2 also verified this point. Therefore, when dealing with multi-state correlated measurements and facing asynchronous and delayed sensor measurements, using factor-graph-based integrated navigation methods has better accuracy and flexibility, and can achieve plug-and-play functionality. Although its computational complexity is higher than that of filter-based combined navigation methods, this aspect can optimize the algorithm to reduce its computational complexity. Furthermore, we also propose an adaptive fast incremental smoothing (AFIS) algorithm. Compared with existing factor graph

optimization algorithms based on incremental smoothing (IS), the AFIS algorithm proposed in this paper can greatly improve the speed of factor graph optimization algorithms while maintaining comparable accuracy, which can better meet the real-time requirements of factor-graph-based integrated navigation methods. Simultaneously, the algorithm can be divided into two components: global optimization and local optimization. Local optimization pertains to optimizing newly added measurement-related partial state variables at each moment, whereas global optimization involves optimizing state variables continuously, which can be performed at regular intervals according to system requirements. Although the accuracy of global optimization at any given moment may not match that of traditional incremental smoothing methods, it is evident that this optimization mode offers greater flexibility. Section 3.3 also verified that the AFIS algorithm notably enhances computational speed while maintaining comparable accuracy to the IS algorithm.

5. Conclusions

This paper introduces an adaptive fast integrated navigation algorithm for INS/GPS/VO, employing a factor graph framework. The study assesses the algorithm's efficacy in managing dynamic changes in sensor effectiveness, handling sensor measurements with asynchronous and delayed presence, and optimizing the factor graph. The simulation results demonstrate that when the effectiveness of the sensor changes, it only needs to suppress or add its factor nodes into the factor graph model to achieve seamless switching of the integrated navigation system working mode, which makes the system highly robust. For asynchronous and delayed sensors, the factor graph algorithm directly defines received sensor measurements as factor nodes linked to the respective variable nodes in the factor graph. This approach eliminates the need to synchronize measurements from different sensors to the same time, thereby mitigating the introduction of additional human error. In comparison to traditional incremental smoothing factor graph optimization algorithms, the AFIS algorithm proposed in this paper significantly enhances optimization speed while maintaining comparable accuracy. The excellent scalability and flexibility of the factor graph model confers substantial advantages in addressing multi-sensor integrated navigation challenges. Consequently, the INS/GPS/VO integrated navigation algorithm, based on the factor graph, holds considerable reference value for diverse multi-sensor integrated navigation systems in various application environments.

Author Contributions: Conceptualization, Z.T.; methodology, Z.T.; software, Z.T. and S.Y.; validation, Z.T. and S.Y.; formal analysis, Z.T. and S.Y.; investigation, Z.T. and S.Y.; resources, Z.T., Y.C. and S.Y.; data curation, Z.T.; writing—original draft, Z.T.; writing—review and editing, Z.T., Y.C. and S.Y.; visualization, Z.T.; supervision, Y.C.; project administration, Y.C.; funding acquisition, Y.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work is funded by the National Key Laboratory on Blind Signal Processing, grant number 61424131903, and the National Natural Science Foundation of China, grant number U20B2067.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

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

References

1. Li, W.J.; Fu, Z.Y. Unmanned aerial vehicle positioning based on multi-sensor information fusion. *Geo-Spat. Inf. Sci.* **2018**, *21*, 302–310. [[CrossRef](#)]
2. Zhu, Y.; Cheng, X.; Zhou, L.; Liu, Q. Information fusion algorithm for asynchronous multi-sensors in integrated navigation systems. *J. Southeast Univ. (Nat. Sci. Ed.)* **2018**, *48*, 195–200.
3. Li, S.; Li, X.; Wang, H.; Zhou, Y.; Shen, Z. Multi-GNSS PPP/INS/Vision/LiDAR tightly integrated system for precise navigation in urban environments. *Inf. Fusion* **2023**, *90*, 218–232. [[CrossRef](#)]

4. Lee, Y.D.; Kim, L.W.; Lee, H.K. A tightly-coupled compressed-state constraint Kalman Filter for integrated visual-inertial-global navigation satellite system navigation in GNSS-degraded environments. *IET Radar Sonar Navig.* **2022**, *16*, 1344–1363. [\[CrossRef\]](#)
5. Jiang, W.; Li, Y.; Rizos, C. Optimal data fusion algorithm for navigation using triple integration of PPP-GNSS, INS and terrestrial ranging system. *IEEE Sens. J.* **2015**, *15*, 5634–5644. [\[CrossRef\]](#)
6. Li, Q.; Ben, Y.Y.; Yu, F.; Tan, J.B. Transversal strapdown INS based on reference ellipsoid for vehicle in the polar region. *IEEE Trans. Veh. Technol.* **2016**, *65*, 7791–7795. [\[CrossRef\]](#)
7. Dryanovski, I.; Valenti, R.G.; Xiao, J.Z. An open-source navigation system for micro aerial vehicles. *Auton. Robot.* **2013**, *34*, 177–188. [\[CrossRef\]](#)
8. Zhang, T.S.; Zhang, H.P.; Ban, Y.L.; Niu, X.J.; Liu, J.N. Tracking loop model and hardware prototype verification of GNSS/INS deep integration. In Proceedings of the 5th China Satellite Navigation Conference, Nanjing, China, 20–23 May 2014; pp. 553–572.
9. Pfeifer, T.; Weissig, P.; Lange, S.; Protzel, P. Robust factor graph optimization—A comparison for sensor fusion applications. In Proceedings of the 2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation, Berlin, Germany, 6–9 September 2016; pp. 1–4.
10. Jiang, C.; Chen, S.; Bo, Y.; Sun, Z.; Lu, Q. Implementation and performance evaluation of a fast relocation method in a GPS/SINS/CSAC integrated navigation system hardware prototype. *IEICE Electron. Express* **2017**, *14*, 1–8. [\[CrossRef\]](#)
11. Fu, Q.; Quan, Q.; Cai, K.Y. Robust pose estimation for multirotor UAVs using off-board monocular vision. *IEEE Trans. Ind. Electron.* **2017**, *64*, 7942–7951. [\[CrossRef\]](#)
12. Xu, T.L.; Cui, P.Y.; Cui, H.T. Design and implementation of multi-sensor integrated navigation system of land vehicle. *Syst. Eng. Electron.* **2008**, *30*, 686–691.
13. Aftatah, M.; Lahrech, A.; Abounada, A.; Soulhi, A. A GPS/INS/Odometer data for land vehicle localization in GPS denied environment. *Mod. Appl. Sci.* **2017**, *11*, 62–75. [\[CrossRef\]](#)
14. Li, Z.K.; Wang, J.; Gao, J.X.; Yao, Y.F. The application of adaptive federated filter in GPS-INS-Odometer integrated navigation. *Acta Geod. Cartogr. Sin.* **2016**, *45*, 157–163.
15. Indelman, V.; Williams, S.; Kaess, M.; Dellaert, F. Information fusion in navigation systems via factor graph based incremental smoothing. *Robot. Auton. Syst.* **2013**, *61*, 721–738. [\[CrossRef\]](#)
16. Wu, N.; Li, B.; Wang, H.; Xing, C.W.; Kuang, J.M. Distributed cooperative localization based on Gaussian message passing on factor graph in wireless networks. *Sci. China Inf. Sci.* **2015**, *58*, 1–15. [\[CrossRef\]](#)
17. Wen, W.; Pfeifer, T.; Bai, X.; Hsu, L. Factor graph optimization for GNSS/INS integration: A comparison with the extended Kalman filter. *Navigation* **2021**, *68*, 315–331. [\[CrossRef\]](#)
18. Chen, M.; Xiong, Z.; Xiong, J.; Wang, R. A hybrid cooperative navigation method for UAV swarm based on factor graph and Kalman filter. *Int. J. Distrib. Sens. Netw.* **2022**, *18*, 105826. [\[CrossRef\]](#)
19. Jiang, Y.; Ding, W.; Gao, Y. GNSS precise positioning for smartphones based on the integration of factor graph optimization and solution separation. *Measurement* **2022**, *203*, 111924. [\[CrossRef\]](#)
20. Mourikis, A.I.; Roumeliotis, S.I. A multi-state constraint Kalman filter for visionaided inertial navigation. In Proceedings of the 2007 IEEE International Conference on Robotics and Automation, Roma, Italy, 10–14 April 2007; pp. 3565–3572.
21. Mourikis, A.I.; Roumeliotis, S.I. A dual-layer estimator architecture for longterm localization. In Proceedings of the 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Anchorage, AK, USA, 24–26 June 2008; pp. 1–8.
22. Bryson, M.; Roberson, M.J.; Sukkariieh, S. Airborne smoothing and mapping using vision and inertial sensors. In Proceedings of the 2009 IEEE International Conference on Robotics and Automation, Kobe, Japan, 12–17 May 2009; pp. 3143–3148.
23. Sibley, G.; Matthies, L.; Sukhatme, G. Sliding window filter with application to planetary landing. *J. Field Robot.* **2010**, *27*, 587–608. [\[CrossRef\]](#)
24. Dellaert, F.; Kaess, M. Square Root SAM: Simultaneous localization and mapping via square root information smoothing. *Int. J. Robot. Res.* **2006**, *25*, 1181–1203. [\[CrossRef\]](#)
25. Kaess, M.; Ranganathan, A.; Dellaert, F. iSAM: Incremental smoothing and mapping. *IEEE Trans. Robot.* **2008**, *24*, 1365–1378. [\[CrossRef\]](#)
26. Williams, S.; Indelman, V.; Kaess, M.; Roberts, R.; Leonard, J.J.; Dellaert, F. Concurrent filtering and smoothing: A parallel architecture for real-time navigation and full smoothing. *Int. J. Robot. Res.* **2014**, *33*, 1544–1568. [\[CrossRef\]](#)
27. Indelman, V.; Gurfil, P.; Rivlin, E.; Rotstein, H. Real-Time Vision-Aided Localization and Navigation Based on Three-View Geometry. *IEEE Trans. Aerosp. Electron. Syst.* **2012**, *48*, 2239–2259. [\[CrossRef\]](#)
28. Carlone, L.; Kira, Z.; Beall, C.; Indelman, V.; Dellaert, F. Eliminating conditionally independent sets in factor graphs: A unifying perspective based on smart factors. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation, Hong Kong, China, 31 May–7 June 2014; pp. 4290–4297.
29. Lupton, T.; Sukkariieh, S. Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions. *IEEE Trans. Robot.* **2012**, *28*, 61–76. [\[CrossRef\]](#)
30. Bazeille, S.; Filliat, D. Combining odometry and visual loop-closure detection for consistent topo-metrical mapping. *RAIRO—Oper. Res.* **2011**, *44*, 365–377. [\[CrossRef\]](#)
31. Liu, C.; Cheng, R.; Zhao, L. Loop closure detection of visual SLAM based on point and line features. *J. Harbin Inst. Technol.* **2020**, *27*, 58–64.

32. An, S.; Zhu, H.; Wei, D.; Tsintotas, K.A.; Gasteratos, A. Fast and incremental loop closure detection with deep features and proximity graphs. *J. Field Robot.* **2022**, *39*, 473–493. [[CrossRef](#)]
33. Kschischang, F.R.; Frey, B.J.; Loeliger, H.A. Factor graphs and the sum-product algorithm. *IEEE Trans. Inf. Theory* **2001**, *47*, 498–519. [[CrossRef](#)]
34. Liu, J.W.; Cui, L.P.; Li, H.E.; Luo, X.L. Research and development on inference technique in probabilistic graphical models. *Comput. Sci.* **2015**, *42*, 1–18.
35. Kaess, M.; Johannsson, H.; Roberts, R.; Ila, V.; Leonard, J.J.; Dellaert, F. ISAM2: Incremental smoothing and mapping using the Bayes tree. *Int. J. Robot. Res.* **2012**, *31*, 216–235. [[CrossRef](#)]
36. Peyrard, N.; Givry, S.D.; Franc, A.; Robin, S.; Sabbadin, R.; Schiex, T.; Vignes, M. Exact and approximate inference in graphical models: Variable elimination and beyond. *Comput. Sci.* **2015**, *35*, 2454–2467.
37. Meyer, G.G.L.; Pascale, M. A family of parallel QR factorization algorithms. *Concurr. Pract. Exp.* **2015**, *8*, 461–473. [[CrossRef](#)]

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.