The Journal of Navigation (2026), 1–18
doi:10.1017/S0373463326101660

CAMBRIDGE UNIVERSITY PRESS

RESEARCH ARTICLE

High-dimensional system error model and real-time estimation scheme for GNSS/INS integrated system

Linzhouting Chen[1], Hongjian Gao[1], Zhanchao Liu[2] and Jiancheng Fang[2]

[1]College of Aerospace Engineering, Guizhou Institute of Technology, Guiyang, Guizhou, China
[2]School of Instrumentation Science and Opto-Electronic Engineering, Beihang University, Beijing, China
Corresponding author: Linzhouting Chen; Email: chenlzt@163.com

Received: 02 April 2025 UTC; **Revised:** 25 May 2026 UTC; **Accepted:** 29 June 2026 UTC

Keywords: computational complexity; high-dimensional error model; INS error; memory usage; real-time estimation

Abstract

This paper establishes an advanced high-dimensional system error model with 42 Kalman filter states for a GNSS/INS integrated system, which sufficiently considers the influence of multiple INS crucial error factors according to the remote sensing survey requirement. Moreover, to overcome the heavy computational burden imposed by high-dimensional matrix operation, an optimal state estimation scheme with high-efficiency Kalman filter is proposed by comprehensively making use of matrix sparsity and symmetry, which retains the estimating optimality with low computation complexity. It is demonstrated through flight experiment that the proposed 42-dimensional error model offers better estimation and navigation precision than the other models, and the proposed real-time estimation scheme can effectively reduce the computation and memory usage to improve the overall performance of GNSS/INS.

1. Introduction

The global navigation satellite system (GNSS) and inertial navigation system (INS) are integrated with a Kalman filter to obtain the optimal state estimation (Cui et al., 2019; Liu et al., 2023), which combines the advantages of both technologies to offer an accurate and complete navigation solution, and thus has been widely used for motion compensation in remote sensing applications.

The navigation solution errors in the GNSS/INS integrated system mainly arise from initialisation errors, inertial sensor errors and processing approximations, which include gravity disturbances, lever-arm errors and timing synchronisation errors. The initialisation errors are generally calibrated by GNSS alignment, which uses position and velocity derived from GNSS as Kalman filter observations (Zhong et al., 2018; Zhong et al., 2023). Additionally, the inertial sensor errors, including biases, scale factor errors, cross-coupling errors, higher-order errors, random noise and more, make up 70%–80% of error sources (Lo and Gao, 2021; Zhong et al., 2024), which are normally compensated by building an exact error model. Knudsen (1985) proposed a 93-state INS error model for the Litton inertial navigation unit, which contained position errors, velocity errors, attitude errors, vertical channel errors, gyro and accelerometer bias, gyro and accelerometer scale factor errors, gyro and accelerometer misalignments, gyro and accelerometer initial thermal transients, etc. Although the INS error model with 93 states is the fairly accurate characterisation in theory, the high degree of dimensionality would result in the huge calculation burden. Lewantowicz and Keen (1991) constructed a 39-state INS error model, which eliminated a number of states from the 93-state Litton model state space and does not produce significant errors. Additionally, the research results have demonstrated that the 39-state is a sufficient substitute for the 93-state model without a significant loss of accuracy.

In commercial integrated navigation software, a simple INS error model is often adopted for low-dynamics application, which only consists of 15 states including position/velocity/attitude errors and

© The Author(s), 2026. Published by Cambridge University Press on behalf of The Royal Institute of Navigation

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

gyro/accelerometer constant bias. The 15-state INS error model provides a rough approximation to the accelerometer and gyro behaviours, which is accounted for by modelling each bias state as the sum of a constant state and white system noise. The simplification of error model has led to increase the processing efficiency of integrated navigation, but can involve a certain accuracy loss of navigation solution. In the high-dynamic environment, such as ballistic missile, rocket and indexed INS applications, the size effect and higher-order errors may also occur and behave like a bias that varies with the amplitude of the vibration. Gao et al. (2018) and Hu and Wang (2023) respectively built their INS error models considering the influence of accelerometers size effect and quadratic errors, which were estimated as Kalman filter states through optimal estimation. In aerial remote sensing application, the INS error propagation is affected by the aircraft manoeuvres. Sun et al. (2010) developed a 36-state INS error model for the aerial remote sensing application with random constants to characterise the accelerometer and gyro scale factor errors, the effect of which on navigation error growth depended on the aircraft manoeuvres. In the 36-state INS error model, the accelerometer and gyro cross-coupling errors are supposed to be completely compensated after calibration. However, the calibration results generally have some deviations compared with the truth, which would have direct influence on the navigation accuracy. Chen and Fang (2013) established a 45-state INS error model for airborne InSAR application, which was built on the feature of the 15-state INS error model by augmenting the system states with scale factor errors, cross-coupling errors, random walks errors and first-order Markov process errors. The simulation results show that the 45-state INS error model has higher navigation accuracy compared with the traditional 15-state INS error model. However, in the actual environment, the dynamic component of the accelerometer and gyro biases is typically approximately 10% of the static component, so accelerometer and gyro biases can be simplified by modelling each bias as the sum of a constant and a Markov state, and the modelled errors adequately overbound their real counterparts.

However, the system error model of GNSS/INS is conditioned on the estimated Kalman filter states, which depends not only on the gyro and accelerometer measurement errors, but also upon some approximate processing errors. Skog and Handel (2011), He et al. (2022) and Li et al. (2024) adopted the software-based time synchronisation method to remove the timing error between GNSS and INS, where the time synchronisation error was included as a Kalman filter state to be estimated and compensated. Fang et al. (2014), Wang et al. (2020) and Luo et al. (2024) constructed their respective statistical models of the gravity disturbance, which are embedded into the system error model for the gravity disturbance estimation and removal online by a Kalman filter. Xiong et al. (2015), Chen et al. (2020) and Zhang et al. (2020) extended lever-arm errors between GNSS and INS as Kalman filter states, which were estimated and compensated in real time by adopting the closed-loop correction architecture.

Especially for the high-precision GNSS/INS integrated system, various kinds of error factors should be considered seriously. In this paper, we establish a high-dimensional error model of the loosely coupled GNSS/INS integration, which is composed of 42 error states including position/velocity/attitude errors, gyro/accelerometer static biases, gyro/accelerometer dynamic biases, gyro/accelerometer scale factor errors, gyro/accelerometer cross-coupling errors, gravity disturbance and lever-arm errors. The main contributions of this paper are summarised as follows.

1) Taking full account of remote sensing application, inertial sensors, integrated architecture, choice of coordinate frame and other factors, we meticulously describe the modelling of inertial sensor errors and remarkable approximate processing errors, and develop a high-dimensional INS state-propagation model by augmenting gyro and accelerometer biases, scale factor errors, cross-coupling errors, gravity disturbance and lever arm uncertainty.
2) With the aim of enhancing the real-time performance of a GNSS/INS integrated system, the high-dimensional matrices are reasonably sparsified and partitioned into blocks, then we propose a classified processing strategy for matrix products featuring either asymmetric or symmetric structures to optimise the tradeoff between multiplication burden and temporary variables space.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

The remaining part of this paper is organised as follows. The major error sources contributing to the accelerometer and gyro outputs are discussed in Section 2. Section 3 illustrates the proposed high-dimensional system error model for GNSS/INS integration and a real-time optimal state estimation scheme using matrix sparsity and symmetry. In Section 4, the experimental results are described and analysed in detail. The conclusion is presented in Section 5.

2. INS error modelling

2.1. Inertial sensor error modelling

First, following inertial sensor calibration and compensation, gyro and accelerometer biases are respectively denoted by the vectors $\boldsymbol{\varepsilon}$ and ∇, which can be split into two parts

$$\begin{cases} \boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_{\mathrm{s}} + \boldsymbol{\varepsilon}_{\mathrm{d}} \\ \nabla = \nabla_{\mathrm{s}} + \nabla_{\mathrm{d}} \end{cases}. \tag{1}$$

where $\boldsymbol{\varepsilon}_{\mathrm{s}}$ and ∇_{s} are static components, $\boldsymbol{\varepsilon}_{\mathrm{d}}$ and ∇_{d} are dynamic components.

Based on analysing the noise characteristics of inertial sensors, gyro and accelerometer biases could be represented as the sum of random constant and first-order Markov processes, tuned to each inertial sensor type (Nassar and El-Sheimy, 2006). The static bias states $\boldsymbol{\varepsilon}_{\mathrm{s}}$ and ∇_{s} are modelled with zero known time variation, while the dynamic bias states $\boldsymbol{\varepsilon}_{\mathrm{d}}$ and ∇_{d} are modelled with deterministic time variation as follows:

$$\begin{cases} \dot{\boldsymbol{\varepsilon}}_{\mathrm{s}} = \boldsymbol{0}, & \dot{\nabla}_{\mathrm{s}} = \boldsymbol{0} \\ \dot{\boldsymbol{\varepsilon}}_{\mathrm{d}} = -\frac{\boldsymbol{\varepsilon}_{\mathrm{d}}}{\boldsymbol{\tau}_{\varepsilon}} + \boldsymbol{w}_{\varepsilon}, & \dot{\nabla}_{\mathrm{d}} = -\frac{\nabla_{\mathrm{d}}}{\tau_{\nabla}} + \boldsymbol{w}_{\nabla} \end{cases}. \tag{2}$$

where τ_{ε} and τ_{∇} are the correlation times of gyro and accelerometer dynamic biases, $\boldsymbol{w}_{\varepsilon}$ and $\boldsymbol{w}_{\nabla}$ denote the excitation white noise of $\boldsymbol{\varepsilon}_{\mathrm{d}}$ and ∇_{d}.

Second, according to the requirements of remote sensing survey, the aircraft should move as far as possible along trajectories with little vibrations. So, vibration-induced errors, such as size effect and higher-order errors can be neglected in aerial remote sensing application. However, circular trajectories can occur when the aircraft is surveying an area, where the scale factor and cross-coupling errors of inertial sensors produce navigation solution errors that increase with time. The scale factor and cross-coupling errors for a trio of gyros and accelerometers mounted with nominally orthogonal sensitive axes can be shown as the following equations:

$$\begin{cases} \delta\omega^{\mathrm{b}}_{\mathrm{x_scale_cross}} = \delta S_{\mathrm{g_x}}\omega^{\mathrm{b}}_{\mathrm{x}} + \delta C_{\mathrm{g_xy}}\omega^{\mathrm{b}}_{\mathrm{y}} + \delta C_{\mathrm{g_xz}}\omega^{\mathrm{b}}_{\mathrm{z}} \\ \delta\omega^{\mathrm{b}}_{\mathrm{y_scale_cross}} = \delta C_{\mathrm{g_yx}}\omega^{\mathrm{b}}_{\mathrm{x}} + \delta S_{\mathrm{g_y}}\omega^{\mathrm{b}}_{\mathrm{y}} + \delta C_{\mathrm{g_yz}}\omega^{\mathrm{b}}_{\mathrm{z}} \\ \delta\omega^{\mathrm{b}}_{\mathrm{z_scale_cross}} = \delta C_{\mathrm{g_zx}}\omega^{\mathrm{b}}_{\mathrm{x}} + \delta C_{\mathrm{g_zy}}\omega^{\mathrm{b}}_{\mathrm{y}} + \delta S_{\mathrm{g_z}}\omega^{\mathrm{b}}_{\mathrm{z}} \\ \delta f^{\mathrm{b}}_{\mathrm{x_scale_cross}} = \delta S_{\mathrm{a_x}}f^{\mathrm{b}}_{\mathrm{x}} + \delta C_{\mathrm{a_xy}}f^{\mathrm{b}}_{\mathrm{y}} + \delta C_{\mathrm{a_xz}}f^{\mathrm{b}}_{\mathrm{z}} \\ \delta f^{\mathrm{b}}_{\mathrm{y_scale_cross}} = \delta C_{\mathrm{a_yx}}f^{\mathrm{b}}_{\mathrm{x}} + \delta S_{\mathrm{a_y}}f^{\mathrm{b}}_{\mathrm{y}} + \delta C_{\mathrm{a_yz}}f^{\mathrm{b}}_{\mathrm{z}} \\ \delta f^{\mathrm{b}}_{\mathrm{z_scale_cross}} = \delta C_{\mathrm{a_zx}}f^{\mathrm{b}}_{\mathrm{x}} + \delta C_{\mathrm{a_zy}}f^{\mathrm{b}}_{\mathrm{y}} + \delta S_{\mathrm{a_z}}f^{\mathrm{b}}_{\mathrm{z}} \end{cases} \tag{3}$$

where x-axis, y-axis and z-axis in the body frame (b-frame) complete the right-handed orthogonal set; $\delta\omega^{\mathrm{b}}_{\mathrm{x_scale_cross}}$, $\delta\omega^{\mathrm{b}}_{\mathrm{y_scale_cross}}$, $\delta\omega^{\mathrm{b}}_{\mathrm{z_scale_cross}}$ and $\delta f^{\mathrm{b}}_{\mathrm{x_scale_cross}}$, $\delta f^{\mathrm{b}}_{\mathrm{y_scale_cross}}$, $\delta f^{\mathrm{b}}_{\mathrm{z_scale_cross}}$ represent the total angular rate and specific force measurement deviations caused by the scale factor and cross-coupling errors; $\delta S_{\mathrm{g_x}}$, $\delta S_{\mathrm{g_y}}$, $\delta S_{\mathrm{g_z}}$ and $\delta S_{\mathrm{a_x}}$, $\delta S_{\mathrm{a_y}}$, $\delta S_{\mathrm{a_z}}$ denote gyro and accelerometer scale factor errors; $\delta C_{\mathrm{g_xy}}$, $\delta C_{\mathrm{g_xz}}$, $\delta C_{\mathrm{g_yx}}$, $\delta C_{\mathrm{g_yz}}$, $\delta C_{\mathrm{g_zx}}$, $\delta C_{\mathrm{g_zy}}$ and $\delta C_{\mathrm{a_xy}}$, $\delta C_{\mathrm{a_xz}}$, $\delta C_{\mathrm{a_yx}}$, $\delta C_{\mathrm{a_yz}}$, $\delta C_{\mathrm{a_zx}}$, $\delta C_{\mathrm{a_zy}}$ are gyro and accelerometer cross-coupling errors; $\omega^{\mathrm{b}}_{\mathrm{x}}$, $\omega^{\mathrm{b}}_{\mathrm{y}}$, $\omega^{\mathrm{b}}_{\mathrm{z}}$ and $f^{\mathrm{b}}_{\mathrm{x}}$, $f^{\mathrm{b}}_{\mathrm{y}}$, $f^{\mathrm{b}}_{\mathrm{z}}$ are gyro angular rate outputs and accelerometer specific force outputs.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

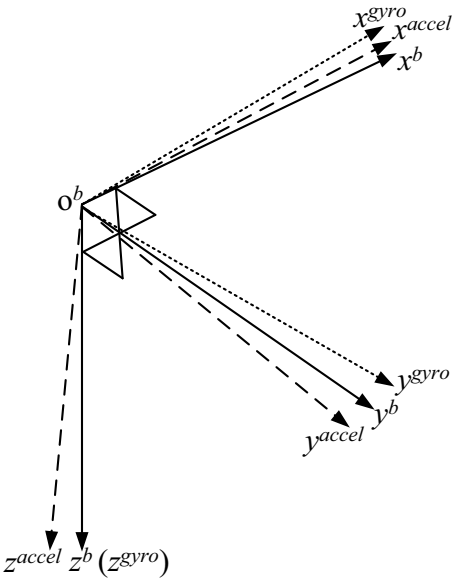


Figure 1. *Misalignment of gyro and accelerometer sensitive axes (Groves, 2013).*

If we define the z^b-axis as the sensitive axis of the z gyro and let the sensitive axis of the *y* gyro lie in the x–y plane of the *b*-frame (as shown in Figure 1), three components of gyro cross-coupling errors may be eliminated, which are δC_{g_xy}, δC_{g_xz} and δC_{g_yz}. So, Equation (3) may be simplified to the following matric expression:

$$\begin{cases} \delta\boldsymbol{\omega}^{b}_{scale_cross} = \boldsymbol{M}_{g}\boldsymbol{\omega}^{b} \\ \delta\boldsymbol{f}^{b}_{scale_cross} = \boldsymbol{M}_{a}\boldsymbol{f}^{b} \end{cases} \tag{4}$$

where

$$\delta\boldsymbol{\omega}^{b}_{scale_cross} = \left[\delta\omega^{b}_{x_scale_cross}\ \delta\omega^{b}_{y_scale_cross}\ \delta\omega^{b}_{z_scale_cross}\right]^{T},$$
$$\delta\boldsymbol{f}^{b}_{scale_cross} = \left[\delta f^{b}_{x_scale_cross}\ \delta f^{b}_{y_scale_cross}\ \delta f^{b}_{z_scale_cross}\right]^{T},$$

$$\boldsymbol{\omega}^{b} = \begin{bmatrix} \omega^{b}_{x} \\ \omega^{b}_{y} \\ \omega^{b}_{z} \end{bmatrix}, \boldsymbol{f}^{b} = \begin{bmatrix} f^{b}_{x} \\ f^{b}_{y} \\ f^{b}_{z} \end{bmatrix}, \boldsymbol{M}_{g} = \begin{bmatrix} \delta S_{g_x} & 0 & 0 \\ \delta C_{g_yx} & \delta S_{g_y} & 0 \\ \delta C_{g_zx} & \delta C_{g_zy} & \delta S_{g_z} \end{bmatrix}, \boldsymbol{M}_{a} = \begin{bmatrix} \delta S_{a_x} & \delta C_{a_xy} & \delta C_{a_xz} \\ \delta C_{a_yx} & \delta S_{a_y} & \delta C_{a_yz} \\ \delta C_{a_zx} & \delta C_{a_zy} & \delta S_{a_z} \end{bmatrix}.$$

Additionally, the scale factor and cross-coupling errors are typically modelled with no deterministic time variation. Thus,

$$\begin{cases} \dot{\boldsymbol{M}}_{g} = \boldsymbol{0} \\ \dot{\boldsymbol{M}}_{a} = \boldsymbol{0} \end{cases} \tag{5}$$

Finally, combining Equations (1) and (4), the main error sources of the gyro and accelerometer outputs in aerial remote sensing application can be shown as the following equations:

$$\begin{cases} \delta\boldsymbol{\omega}^{b} = \boldsymbol{M}_{g}\boldsymbol{\omega}^{b} + \varepsilon_{s} + \varepsilon_{d} \\ \delta\boldsymbol{f}^{b} = \boldsymbol{M}_{a}\boldsymbol{f}^{b} + \nabla_{s} + \nabla_{d} \end{cases} \tag{6}$$

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

2.2. Approximate processing error modeling

First, according to the inertial navigation equations in the local-navigation frame (n-frame) (Mohinder et al., 2020), the acceleration could be expressed in terms of the specific force, centrifugal acceleration and gravity,

$$\dot{\boldsymbol{V}}^{\mathrm{n}} = \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}}\boldsymbol{f}^{\mathrm{b}} - (2\boldsymbol{\omega}_{\mathrm{ie}}^{\mathrm{n}} + \boldsymbol{\omega}_{\mathrm{en}}^{\mathrm{n}}) \times \boldsymbol{V}^{\mathrm{n}} + \boldsymbol{g}^{\mathrm{n}} \tag{7}$$

where $\boldsymbol{V}^{\mathrm{n}}$ and $\omega_{\mathrm{ie}}^{\mathrm{n}}$ are the kinematic velocity and the earth rotation vector in the n-frame; $\boldsymbol{\omega}_{\mathrm{en}}^{\mathrm{n}}$ is known as the rotating angular velocity which arises from the rotation of the n-frame axes as the frame centre moves with respect to the Earth; $C_{\mathrm{b}}^{\mathrm{n}}$ denotes the transformation matrix between b-frame and n-frame; and $\boldsymbol{g}^{\mathrm{n}}$ represents the gravity vector in n-frame.

Typically, we use an empirical model (WGS84) to obtain the gravity vector, $\boldsymbol{g}_{\mathrm{m}}^{\mathrm{n}}$, which applies the scaling law to calculate the gravity value with various altitude (Manon et al., 2017),

$$\begin{cases} \boldsymbol{g}_{\mathrm{m}}^{\mathrm{n}} = \begin{pmatrix} 0 & 0 & \gamma \end{pmatrix}^{\mathrm{T}} \\ \gamma = 9.780325 \times (1 + 0.005302\sin^2 L - 0.0000058\sin^2 2L) \\ \qquad -(3.0877 - 0.0044\sin^2 L) \times 10^{-6} \quad h + 0.072 \times 10^{-12} h^2 \end{cases} \tag{8}$$

where γ denotes the vertical component of $\boldsymbol{g}_{\mathrm{m}}^{\mathrm{n}}$; and L and h are latitude and altitude.

The difference between the actual and modeled gravity fields is referred to as the gravity disturbance $\delta\boldsymbol{g}^{\mathrm{n}}$

$$\delta\boldsymbol{g}^{\mathrm{n}} = \boldsymbol{g}^{\mathrm{n}} - \boldsymbol{g}_{\mathrm{m}}^{\mathrm{n}} \tag{9}$$

In common GNSS/INS application scenarios, the effects of $\delta\boldsymbol{g}^{\mathrm{n}}$ are absorbed by the z-axis accelerometer bias state and the components of the attitude error states resolved about the horizontal axes (Groves, 2013). However, for high-precision GNSS/INS in remote sensing application, it is essential to estimate $\delta\boldsymbol{g}^{\mathrm{n}}$ as separate Kalman filter states which could be modelled as exponentially correlated fixed-variance first-order Markov processes

$$\delta\dot{\boldsymbol{g}}^{\mathrm{n}} = -\frac{\delta\boldsymbol{g}^{\mathrm{n}}}{\boldsymbol{\tau}_{\mathrm{g}}} + \boldsymbol{w}_{\mathrm{g}} \tag{10}$$

where τ_{g} is the correlation time of the gravity disturbance and $\boldsymbol{w}_g$ denotes the excitation white noise of $\delta\boldsymbol{g}^{\mathrm{n}}$.

As gravity disturbance errors are spatially correlated rather than temporally correlated, the correlation time τ_{g} is velocity-dependent, which can be expressed by using the presumably constant velocity of the vehicle $\bar{V}$ and the correlation distance β.

$$\tau_{\mathrm{g}} = \beta\bar{V} \tag{11}$$

Equation (11) is substituted into Equation (10) to reorganise

$$\delta\dot{\boldsymbol{g}}^{\mathrm{n}} = -\frac{\delta\boldsymbol{g}^{\mathrm{n}}}{\beta\bar{V}} + \boldsymbol{\omega}_{\mathrm{g}} \tag{12}$$

Second, the position and velocity solutions of GNSS are used as observations to estimate the INS errors. Normally, a GNSS receiving antenna needs to be mounted onto the aircraft top for good reception and INS is always fixedly connected with the remoting sensor under the airframe.

The spatial vector from INS location to GNSS receiving antenna (as shown in Figure 2), named lever arm, may bring non-negligible observation errors, which will lead to the INS/GNSS accuracy

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

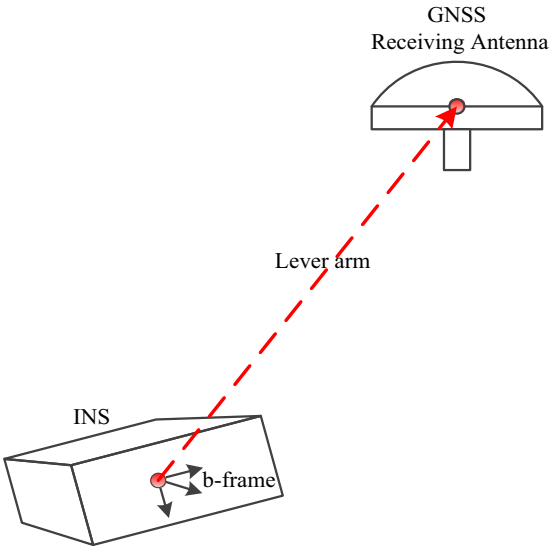


Figure 2. *Schematic of lever arm.*

degradation. Therefore, the lever arm effect should be properly compensated that generally converts the GNSS antenna phase centre to an INS center as follows:

$$\begin{cases} \boldsymbol{P}^{\mathrm{n}}_{\mathrm{INS}} = \boldsymbol{P}^{\mathrm{n}}_{\mathrm{GNSS}} - \boldsymbol{C}^{\mathrm{n}}_{\mathrm{b}} \boldsymbol{l}^{\mathrm{b}}_{\mathrm{GNSS}} \\ \boldsymbol{V}^{\mathrm{n}}_{\mathrm{INS}} = \boldsymbol{V}^{\mathrm{n}}_{\mathrm{GNSS}} - \boldsymbol{C}^{\mathrm{n}}_{\mathrm{b}} \left(\boldsymbol{\omega}^{\mathrm{b}}_{\mathrm{nb}} \times \boldsymbol{l}^{\mathrm{b}}_{\mathrm{GNSS}} \right) \\ \boldsymbol{A}^{\mathrm{n}}_{\mathrm{INS}} = \boldsymbol{A}^{\mathrm{n}}_{\mathrm{GNSS}} - \boldsymbol{C}^{\mathrm{n}}_{\mathrm{b}} \left(\boldsymbol{\omega}^{\mathrm{b}}_{\mathrm{nb}} \times \boldsymbol{\omega}^{\mathrm{b}}_{\mathrm{nb}} \times \boldsymbol{l}^{\mathrm{b}}_{\mathrm{GNSS}} \right) - \boldsymbol{C}^{\mathrm{n}}_{\mathrm{b}} \left(\dot{\boldsymbol{\omega}}^{\mathrm{b}}_{\mathrm{nb}} \times \boldsymbol{l}^{\mathrm{b}}_{\mathrm{GNSS}} \right) \end{cases} \tag{13}$$

where $\boldsymbol{P}^{\mathrm{n}}_{\mathrm{GNSS}}$, $\boldsymbol{V}^{\mathrm{n}}_{\mathrm{GNSS}}$ and $\boldsymbol{A}^{\mathrm{n}}_{\mathrm{GNSS}}$ denote the actual position, velocity and acceleration solutions in GNSS antenna phase centre. Additionally, $\boldsymbol{P}^{\mathrm{n}}_{\mathrm{INS}}$, $\boldsymbol{V}^{\mathrm{n}}_{\mathrm{INS}}$ and $\boldsymbol{A}^{\mathrm{n}}_{\mathrm{INS}}$ denote the actual position, velocity and acceleration solutions in the INS centre. Here, $\boldsymbol{\omega}^{\mathrm{b}}_{\mathrm{nb}}$ is known as the rotating angular velocity which arises from the rotation of the *b*-frame axes as the frame centre moves with respect to the *n*-frame. Furthermore, $\boldsymbol{l}^{\mathrm{b}}_{\mathrm{GNSS}}$ denotes the actual lever arm vector of GNSS with respect to INS in the *b*-frame.

However, if it is not practical to accurately measure the lever arm, the uncertainty may also be estimated as Kalman filter states and can be modelled as follows:

$$\begin{cases} \boldsymbol{l}^{\mathrm{b}}_{\mathrm{GNSS_measure}} = \boldsymbol{l}^{\mathrm{b}}_{\mathrm{GNSS}} + \delta \boldsymbol{l}^{\mathrm{b}} \\ \dot{\delta} \boldsymbol{l}^{\mathrm{b}} = 0 \end{cases} \tag{14}$$

where $\boldsymbol{l}^{\mathrm{b}}_{\mathrm{GNSS_measure}}$ denotes the measured lever arm and $\delta \boldsymbol{l}$ denotes the lever arm uncertainty.

3. High-dimensional system error model and real-time estimation scheme

3.1. System error model and state selection

For GNSS/INS integration, the system error model depends on the Kalman filter states, and closely correlates with the integration architecture, coordinate systems used in navigation, inertial sensor errors, approximate calculation errors and application scenario (Groves, 2013). The most prevalently used GNSS/INS integration architecture in remote sensing application is the loosely coupled scheme because of its simplicity that applies to any INS and GNNS equipment. In *n*-frame, the coordinate axes are aligned with the topographic directions, such as east (E), north (N) and up (U), and the error propagation equations of INS navigation are given as follows:

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

$$\begin{cases} \delta\dot{L} = -\frac{V_N \cdot \delta h}{(R_m+h)^2} + \frac{\delta V_N}{R_m+h}, \delta\dot{h} = \delta V_U \\ \delta\dot{\lambda} = \frac{V_E \cdot \sec L \cdot \tan L \cdot \delta L}{R_n+h} - \frac{V_E \cdot \sec L \cdot \delta h}{(R_n+h)^2} + \frac{\sec L \cdot \delta V_E}{R_n+h} \\ \delta\dot{\boldsymbol{V}}^n = \boldsymbol{C}_b^n \cdot \delta\boldsymbol{f}^b - \boldsymbol{\psi}^n \times (\boldsymbol{C}_b^n \cdot \boldsymbol{f}^b) - (2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \delta\boldsymbol{V}^n \\ \qquad +(2\delta\boldsymbol{\omega}_{ie}^n + \delta\boldsymbol{\omega}_{en}^n) \times V^n + \delta\boldsymbol{g}^n \\ \dot{\boldsymbol{\psi}}^n = \boldsymbol{C}_b^n \cdot \delta\boldsymbol{\omega}^b + \boldsymbol{\psi}^n \times (\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) + \delta\boldsymbol{\omega}_{ie}^n + \delta\boldsymbol{\omega}_{en}^n \end{cases} \tag{15}$$

where λ denotes longitude; R_m and R_n denote the main curvature radius of meridian and prime vertical; $\boldsymbol{V}^n = [V_E\ V_N\ V_U]^T$ denotes the velocity vector in the n-frame; $\boldsymbol{\psi}^n = [\psi_E\ \psi_N\ \psi_U]^T$ denotes the attitude misalignment vector in the n-frame; and $\boldsymbol{C}_b^n = [C_{11}\ C_{12}\ C_{13};\ C_{21}\ C_{22}\ C_{23};\ C_{31}\ C_{32}\ C_{33}]$ denotes the transformation matrix between the b-frame and n-frame.

For the GNSS/INS integration with a loosely coupled scheme, the core partition of the system state vector comprises the position/velocity/attitude errors shown in the left-hand side of Equation (15). The choice of other error states to estimate depends on their effect on the GNSS/INS integrated navigation solution. The accelerometer biases and gravity disturbance errors should be estimated because the attitude error estimations would be contaminated by the effects of them. The gyro biases produce a quadratic increase of velocity error with the elapse of time, which should be also estimated. With a circular or oval trajectory, the scale factor and cross-coupling errors will result in an oscillating attitude error, and their inclusion in the system state vector can effectively improve the estimation of the accelerometer and gyro biases. The lever arm errors in the aircraft platform of the remote sensing application obviously exceed the cm-level error in GNSS measurement, which should also be included as the system states. Therefore, a system error model of GNSS/INS is developed here for estimating position, velocity and attitude errors referenced to the n-frame, together with inertial sensor errors and approximate processing errors. The state vector $\boldsymbol{X}$ consists of 42 error states such that $\boldsymbol{X} = [\delta L\ \delta\lambda\ \delta h\ \delta V_E\ \delta V_N\ \delta V_U\ \psi_E\ \psi_N\ \psi_U \nabla_{dx} \nabla_{dy} \nabla_{dz}\ \boldsymbol{\varepsilon}_{dx}\ \boldsymbol{\varepsilon}_{dy}\ \boldsymbol{\varepsilon}_{dz}\ \delta g_E\ \delta g_N\ \delta g_U \nabla_{sx} \nabla_{sy} \nabla_{sz}\ \delta S_{a_x}\ \delta S_{a_y}\ \delta S_{a_z}\ \delta C_{a_xy}\ \delta C_{a_xz}\ \delta C_{a_yx}\ \delta C_{a_yz}\ \delta C_{a_zx}\ \delta C_{a_zy}\ \boldsymbol{\varepsilon}_{sx}\ \boldsymbol{\varepsilon}_{sy}\ \boldsymbol{\varepsilon}_{sz}\ \delta S_{g_x}\ \delta S_{g_y}\ \delta S_{g_z}\ \delta C_{g_yx}\ \delta C_{g_zx}\ \delta C_{g_zy}\ \delta l_x\ \delta l_y\ \delta l_z]^T$.

Based on INS error propagations, inertial sensor error models and approximate calculation error models, the system state equation for Kalman filter state estimation, at time, t, is given as

$$\dot{\boldsymbol{X}}(t) = \boldsymbol{F}(t) \cdot \boldsymbol{X}(t) + \boldsymbol{G}(t) \cdot \boldsymbol{w}(t) \tag{16}$$

where $\boldsymbol{F}$ is the 42-dimensional system matrix and $\boldsymbol{G}$ is the system noise distribution matrix with 42×9 dimensions. The specific forms of $\boldsymbol{F}$ and $\boldsymbol{G}$ can be derivable from Equations (6), (10), (14) and (15). Here, $\boldsymbol{w}$ is the 9-dimensional system noise vector.

3.2. Optimal state estimation scheme with high-efficiency Kalman filter

The transition matrix $\boldsymbol{\Phi}_{k/k-1}$ is derived from the power-series expansion of system matrix $\boldsymbol{F}$ and propagation interval T,

$$\boldsymbol{\Phi}_{k/k-1} = \sum_{r=0}^{+\infty} \frac{\boldsymbol{F}_{k-1}^r T^r}{r!} \tag{17}$$

where $\boldsymbol{F}_{k-1}$ can be calculated by $(\boldsymbol{F}(t\text{–}T) + \boldsymbol{F}(t))/2$.

The differences between measurements output by INS and GNSS are taken as the observation vector $\boldsymbol{Z}_k$, and the discrete-time state-space model is as shown in the following equations:

$$\begin{cases} \boldsymbol{X}_k = \boldsymbol{\Phi}_{k/k-1}\boldsymbol{X}_{k-1} + \boldsymbol{W}_{k-1} \\ \boldsymbol{Z}_k = \boldsymbol{H}_k \cdot \boldsymbol{X}_k + \boldsymbol{v}_k \end{cases} \tag{18}$$

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

where $\boldsymbol{v}_k$ denotes the observation noise vector, whose components are considered as zero-mean random white noise. Additionally, $\boldsymbol{W}_{k-1}$ is the discrete system noise vector such that

$$\boldsymbol{W}_{k-1} = \int_{t-T}^{t} \boldsymbol{\Phi}_{k/k-1} \boldsymbol{G}(\tau) \boldsymbol{w}(\tau) d\tau \tag{19}$$

Here, $\boldsymbol{Z}$ is derived from the difference of position, velocity and acceleration solutions between INS and GNSS, calculated as

$$\boldsymbol{Z} = \begin{bmatrix} \delta \boldsymbol{P}^{\mathrm{n}} + \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} \boldsymbol{l}_{\mathrm{GNSS}}^{\mathrm{b}} \times \boldsymbol{\psi}^{\mathrm{n}} + \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} \delta \boldsymbol{l}^{\mathrm{b}} \\ \delta \boldsymbol{V}^{\mathrm{n}} + \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} \boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}} \times \boldsymbol{l}_{\mathrm{GNSS}}^{\mathrm{b}} \times \boldsymbol{\psi}^{\mathrm{n}} + \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} \boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}} \times \delta \boldsymbol{l}^{\mathrm{b}} \\ (\boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} \boldsymbol{f}^{\mathrm{b}} + \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} (\boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}} \times \boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}} + \dot{\boldsymbol{\omega}}_{\mathrm{nb}}^{\mathrm{b}}) \times \boldsymbol{l}_{\mathrm{GNSS}}^{\mathrm{b}}) \times \boldsymbol{\psi}^{\mathrm{n}} + \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} \delta \boldsymbol{f}^{\mathrm{b}} \\ + \delta \boldsymbol{g}^{\mathrm{n}} + \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} (\boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}} \times \boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}} + \dot{\boldsymbol{\omega}}_{\mathrm{nb}}^{\mathrm{b}}) \times \delta \boldsymbol{l}^{\mathrm{b}} \end{bmatrix} + \boldsymbol{v} \tag{20}$$

Thus, the specific forms of the observation matrix $\boldsymbol{H}$ are defined as follows:

$$\begin{cases} \boldsymbol{H} = \begin{bmatrix} \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{H}1 & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times6} & \boldsymbol{0}_{3\times9} & \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \boldsymbol{H}2 & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times6} & \boldsymbol{0}_{3\times9} & \boldsymbol{H}6 \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{H}3 & \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} & \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} & \boldsymbol{H}4 & \boldsymbol{H}5 & \boldsymbol{0}_{3\times9} & \boldsymbol{H}7 \end{bmatrix} \\ \boldsymbol{H}1 = \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} [\boldsymbol{l}_{\mathrm{GNSS}}^{\mathrm{b}} \times] \quad \boldsymbol{H}2 = \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} [\boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}} \times \boldsymbol{l}_{\mathrm{GNSS}}^{\mathrm{b}} \times] \\ \boldsymbol{H}3 = \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} [\boldsymbol{f}^{\mathrm{b}} \times] + \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} [(\boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}} \times \boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}} + \dot{\boldsymbol{\omega}}_{\mathrm{nb}}^{\mathrm{b}}) \times \boldsymbol{l}_{\mathrm{GNSS}}^{\mathrm{b}} \times] \\ \boldsymbol{H}4 = \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} \begin{bmatrix} f_{\mathrm{x}}^{\mathrm{b}} & 0 & 0 \\ 0 & f_{\mathrm{y}}^{\mathrm{b}} & 0 \\ 0 & 0 & f_{\mathrm{z}}^{\mathrm{b}} \end{bmatrix} \quad \boldsymbol{H}5 = \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} \begin{bmatrix} f_{\mathrm{y}}^{\mathrm{b}} & f_{\mathrm{z}}^{\mathrm{b}} & 0 & 0 & 0 & 0 \\ 0 & 0 & f_{\mathrm{x}}^{\mathrm{b}} & f_{\mathrm{z}}^{\mathrm{b}} & 0 & 0 \\ 0 & 0 & 0 & 0 & f_{\mathrm{x}}^{\mathrm{b}} & f_{\mathrm{y}}^{\mathrm{b}} \end{bmatrix} \\ \boldsymbol{H}6 = \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} [\boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}} \times] \quad \boldsymbol{H}7 = \boldsymbol{C}_{\mathrm{b}}^{\mathrm{n}} [(\boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}} \times \boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}} + \dot{\boldsymbol{\omega}}_{\mathrm{nb}}^{\mathrm{b}}) \times] \end{cases} \tag{21}$$

where [*×] denotes the anti-symmetric matrix of vector *.

The state vector estimation $\hat{\boldsymbol{X}}_{\boldsymbol{k}}$ is solved according to the discrete-time Kalman filter algorithm, as

$$\begin{cases} \hat{\boldsymbol{X}}_{k/k-1} = \boldsymbol{\Phi}_{k/k-1} \hat{\boldsymbol{X}}_{k-1} \\ \hat{\boldsymbol{X}}_k = \hat{\boldsymbol{X}}_{k/k-1} + \boldsymbol{K}_k (\boldsymbol{Z}_k - \boldsymbol{H}_k \hat{\boldsymbol{X}}_{k/k-1}) \\ \boldsymbol{K}_k = \boldsymbol{P}_{k/k-1} \boldsymbol{H}_k^{\mathrm{T}} (\boldsymbol{H}_k \boldsymbol{P}_{k/k-1} \boldsymbol{H}_k^{\mathrm{T}} + \boldsymbol{R}_k)^{-1} \\ \boldsymbol{P}_{k/k-1} = \boldsymbol{\Phi}_{k/k-1} \boldsymbol{P}_{k-1} \boldsymbol{\Phi}_{k/k-1}^{\mathrm{T}} + \boldsymbol{Q}_{\boldsymbol{k-1}} \\ \boldsymbol{P}_k = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k) \boldsymbol{P}_{k/k-1} (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k)^{\mathrm{T}} + \boldsymbol{K}_k \boldsymbol{R}_k \boldsymbol{K}_k^{\mathrm{T}} \end{cases} \tag{22}$$

where $\boldsymbol{K}_k$ and $\boldsymbol{P}_k$ represent the Kalman gain matrix and the error covariance matrix. Here, $\boldsymbol{R}_k$ and $\boldsymbol{Q}_k$ are the measurement noise covariance matrix and the system noise covariance matrix, which are determined by the following equation:

$$\begin{cases} \boldsymbol{R}_k = E[\boldsymbol{v}_k \boldsymbol{v}_k^{\mathrm{T}}] \\ \boldsymbol{Q}_k = E[\boldsymbol{W}_k \boldsymbol{W}_k^{\mathrm{T}}] \end{cases} \tag{23}$$

In Equation (22), the computation burden of propagating the error covariance matrix $\boldsymbol{P}_{k/k-1}$ is most heavy, which holds more than 70% of the computation time in the whole Kalman filter process (Ca et al., 2006). Additionally, $\boldsymbol{\Phi}_{k/k-1} \boldsymbol{P}_{k-1} \boldsymbol{\Phi}_{k/k-1}^{\mathrm{T}}$ is the core part of the $\boldsymbol{P}_{k/k-1}$ calculation, which requires

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

$2N^3$ multiplications and N^2 temporary variables with the conventional matrix dot-product (N represents the dimensionality of the system states). Therefore, efforts to achieve speed optimisation of the Kalman filter should focus on reducing the computation burden of $\boldsymbol{\Phi}_{k/k-1}\boldsymbol{P}_{k-1}\boldsymbol{\Phi}_{k/k-1}^{\mathrm{T}}$. To ensure as much GNSS/INS accuracy as possible, the commonly used dimension-reduction method will not be taken into account. In this study, due to large amount of zero elements contained in the transition matrix, $\boldsymbol{\Phi}_{k/k-1}$ can be used to construct an obvious sparse matrix. From Equation (17), the value of T is set to ten milliseconds according to the IMU sampling frequency in our study, so the power series can be limited to the first order in $\boldsymbol{F}_{k-1}$, and the transition matrix $\boldsymbol{\Phi}_{k/k-1}$ is

$$\boldsymbol{\Phi}_{k/k-1} = \begin{bmatrix} \boldsymbol{\Phi}1_{9\times9} & \boldsymbol{\Phi}2_{9\times9} & \boldsymbol{\Phi}3_{9\times24} \\ \boldsymbol{0}_{9\times9} & \boldsymbol{\Phi}4_{9\times9} & \boldsymbol{0}_{9\times24} \\ \boldsymbol{0}_{24\times9} & \boldsymbol{0}_{24\times9} & \boldsymbol{I}_{24\times24} \end{bmatrix} \tag{24}$$

Since the error covariance matrix is symmetric about the diagonal, $\boldsymbol{P}_{k-1}$ may be partitioned as

$$\boldsymbol{P}_{k-1} = \begin{bmatrix} \boldsymbol{P}1_{9\times9} & \boldsymbol{P}2_{9\times9} & \boldsymbol{P}3_{9\times24} \\ \boldsymbol{P}2_{9\times9}^{\mathrm{T}} & \boldsymbol{P}4_{9\times9} & \boldsymbol{P}5_{9\times24} \\ \boldsymbol{P}3_{24\times9}^{\mathrm{T}} & \boldsymbol{P}5_{24\times9}^{\mathrm{T}} & \boldsymbol{P}6_{24\times24} \end{bmatrix} \tag{25}$$

From Equations (24) and (25), $\boldsymbol{\Phi}_{k/k-1}\boldsymbol{P}_{k-1}\boldsymbol{\Phi}_{k/k-1}^{\mathrm{T}}$ becomes

$$\boldsymbol{\Phi}_{k/k-1}\boldsymbol{P}_{k-1}\boldsymbol{\Phi}_{k/k-1}^{\mathrm{T}} = \begin{bmatrix} \begin{pmatrix} \boldsymbol{\Phi}1\boldsymbol{P}1\boldsymbol{\Phi}1^{\mathrm{T}} + \boldsymbol{\Phi}2\boldsymbol{P}4\boldsymbol{\Phi}2^{\mathrm{T}} + \boldsymbol{\Phi}3\boldsymbol{P}6\boldsymbol{\Phi}3^{\mathrm{T}} \\ +\boldsymbol{\Phi}1\boldsymbol{P}2\boldsymbol{\Phi}2^{\mathrm{T}} + \left(\boldsymbol{\Phi}1\boldsymbol{P}2\boldsymbol{\Phi}2^{\mathrm{T}}\right)^{\mathrm{T}} \\ +\boldsymbol{\Phi}1\boldsymbol{P}3\boldsymbol{\Phi}3^{\mathrm{T}} + \left(\boldsymbol{\Phi}1\boldsymbol{P}3\boldsymbol{\Phi}3^{\mathrm{T}}\right)^{\mathrm{T}} \\ +\boldsymbol{\Phi}2\boldsymbol{P}5\boldsymbol{\Phi}3^{\mathrm{T}} + \left(\boldsymbol{\Phi}2\boldsymbol{P}5\boldsymbol{\Phi}3^{\mathrm{T}}\right)^{\mathrm{T}} \end{pmatrix}_{9\times9} & \begin{pmatrix} \boldsymbol{\Phi}1\boldsymbol{P}2\boldsymbol{\Phi}4^{\mathrm{T}} \\ +\boldsymbol{\Phi}2\boldsymbol{P}4\boldsymbol{\Phi}4^{\mathrm{T}} \\ +\boldsymbol{\Phi}3\boldsymbol{P}5^{\mathrm{T}}\boldsymbol{\Phi}4^{\mathrm{T}} \end{pmatrix}_{9\times9} & \begin{pmatrix} \boldsymbol{\Phi}1\boldsymbol{P}3 \\ +\boldsymbol{\Phi}2\boldsymbol{P}5 \\ +\boldsymbol{\Phi}3\boldsymbol{P}6 \end{pmatrix}_{9\times24} \\ \begin{pmatrix} \boldsymbol{\Phi}1\boldsymbol{P}2\boldsymbol{\Phi}4^{\mathrm{T}} \\ +\boldsymbol{\Phi}2\boldsymbol{P}4\boldsymbol{\Phi}4^{\mathrm{T}} \\ +\boldsymbol{\Phi}3\boldsymbol{P}5^{\mathrm{T}}\boldsymbol{\Phi}4^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}}_{9\times9} & \left(\boldsymbol{\Phi}4\boldsymbol{P}4\boldsymbol{\Phi}4^{\mathrm{T}}\right)_{9\times9} & (\boldsymbol{\Phi}4\boldsymbol{P}5)_{9\times24} \\ \begin{pmatrix} \boldsymbol{\Phi}1\boldsymbol{P}3 \\ +\boldsymbol{\Phi}2\boldsymbol{P}5 \\ +\boldsymbol{\Phi}3\boldsymbol{P}6 \end{pmatrix}^{\mathrm{T}}_{24\times9} & (\boldsymbol{\Phi}4\boldsymbol{P}5)^{\mathrm{T}}{}_{24\times9} & \boldsymbol{P}6_{24\times24} \end{bmatrix} \tag{26}$$

In Equation (26), $\boldsymbol{\Phi}_{k/k-1}\boldsymbol{P}_{k-1}\boldsymbol{\Phi}_{k/k-1}^{\mathrm{T}}$ is obviously symmetric about the diagonal and displays much repetition calculation. To optimise the tradeoff between multiplication burden and temporary variables space, the computation methods for these matrix multiplications can be divided into two categories. One is the conventional matrix product method, which is used for the asymmetric structure cases, such as $\{\boldsymbol{\Phi}1\boldsymbol{P}2\boldsymbol{\Phi}2^{\mathrm{T}}, \boldsymbol{\Phi}1\boldsymbol{P}3\boldsymbol{\Phi}3^{\mathrm{T}}, \boldsymbol{\Phi}2\boldsymbol{P}5\boldsymbol{\Phi}3^{\mathrm{T}}, \boldsymbol{\Phi}1\boldsymbol{P}2\boldsymbol{\Phi}4^{\mathrm{T}}, \boldsymbol{\Phi}2\boldsymbol{P}4\boldsymbol{\Phi}4^{\mathrm{T}}, \boldsymbol{\Phi}3\boldsymbol{P}5^{\mathrm{T}}\boldsymbol{\Phi}4^{\mathrm{T}}, \boldsymbol{\Phi}1\boldsymbol{P}3, \boldsymbol{\Phi}2\boldsymbol{P}5, \boldsymbol{\Phi}3\boldsymbol{P}6, \boldsymbol{\Phi}4\boldsymbol{P}5\}$. This method requires relatively little multiplication computation, but needs to occupy a certain amount of temporary variables space.

$$(*)_{ij} = \sum_{k=1}^{q} \boldsymbol{\Phi}_{ik}\boldsymbol{P}_{kj} \tag{27}$$

where $(*)_{ij}$ denotes the element of ith row and jth column in the result matrix; and q denotes the order of $\boldsymbol{P}2$, $\boldsymbol{P}3$, $\boldsymbol{P}4$, $\boldsymbol{P}5$ and $\boldsymbol{P}6$. Each of $\{\boldsymbol{\Phi}1_{9\times9}\boldsymbol{P}2_{9\times9}\boldsymbol{\Phi}2_{9\times9}^{\mathrm{T}}$, $\boldsymbol{\Phi}2_{9\times9}\boldsymbol{P}4_{9\times9}\boldsymbol{\Phi}4_{9\times9}^{\mathrm{T}}$ and $\boldsymbol{\Phi}1_{9\times9}\boldsymbol{P}2_{9\times9}\boldsymbol{\Phi}4_{9\times9}^{\mathrm{T}}\}$ requires 1,458 multiplications and 81 temporary variables. Then, each of $\{\boldsymbol{\Phi}1_{9\times9}\boldsymbol{P}3_{9\times24}\boldsymbol{\Phi}3_{24\times9}^{\mathrm{T}}$ and $\boldsymbol{\Phi}2_{9\times9}\boldsymbol{P}5_{9\times24}\boldsymbol{\Phi}3_{24\times9}^{\mathrm{T}}\}$ requires 7,128 multiplications and 216 temporary variables. Additionally, $\boldsymbol{\Phi}3_{9\times24}P5_{24\times9}^{\mathrm{T}}$ $\boldsymbol{\Phi}4_{9\times9}^{\mathrm{T}}$ requires 2,673 multiplications and 81 temporary variables. Finally, each of

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

Table 1. *Multiplications and temporary variables requirement (system state dimension $N = 42$).*

Calculation $\boldsymbol{\Phi}_{k/k-1}\mathbf{P}_{k-1}\boldsymbol{\Phi}_{k/k-1}^{\mathrm{T}}$	Multiplications required	Temporary variables required
Conventional computation approach: simple matrix dot-product	148,176	1,764
Proposed computation approach: matrix sparsity + matrix symmetry + matrix exterior-product	87,399	756

$\{\boldsymbol{\Phi}1_{9\times9}\boldsymbol{P}3_{9\times24}$, $\boldsymbol{\Phi}2_{9\times9}\boldsymbol{P}5_{9\times24}$ and $\boldsymbol{\Phi}4_{9\times9}\boldsymbol{P}5_{9\times24}\}$ requires 1944 multiplications, and $\boldsymbol{\Phi}3_{9\times24}\boldsymbol{P}6_{24\times24}$ requires 5,184 multiplications.

The other one is the matrix exterior-product method, which is used for the symmetric structure cases, such as $\{\boldsymbol{\Phi}1\boldsymbol{P}1\boldsymbol{\Phi}1^{\mathrm{T}}$, $\boldsymbol{\Phi}2\boldsymbol{P}4\boldsymbol{\Phi}2^{\mathrm{T}}$, $\boldsymbol{\Phi}3\boldsymbol{P}6\boldsymbol{\Phi}3^{\mathrm{T}}$, $\boldsymbol{\Phi}4\boldsymbol{P}4\boldsymbol{\Phi}4^{\mathrm{T}}\}$. Compared with the former, this method has a higher multiplication computational cost, but requires no temporary variables space.

$$(*)_{ij} = \sum_{k=1}^{q} P_{kk}\Phi_{ik}\Phi_{jk} + \sum_{m=1}^{q-1}\sum_{k=m+1}^{q} P_{mk}(\Phi_{im}\Phi_{jk}+\Phi_{ik}\Phi_{jm}) \tag{28}$$

where $(*)_{ij}$ denotes the element of ith row and jth column in the result matrix; and q denotes the order of $\boldsymbol{P}1$, $\boldsymbol{P}4$ and $\boldsymbol{P}6$. Each of $\{\boldsymbol{\Phi}1_{9\times9}\boldsymbol{P}1_{9\times9}\boldsymbol{\Phi}1_{9\times9}^{T}$, $\boldsymbol{\Phi}2_{9\times9}\boldsymbol{P}4_{9\times9}\boldsymbol{\Phi}2_{9\times9}^{T}$, $\boldsymbol{\Phi}4_{9\times9}\boldsymbol{P}4_{9\times9}\boldsymbol{\Phi}4_{9\times9}^{T}\}$ requires 5,670 multiplications, and $\boldsymbol{\Phi}3_{9\times24}\boldsymbol{P}6_{24\times24}\boldsymbol{\Phi}3_{24\times9}^{T}$ requires 38,070 multiplications. Additionally, all of them have no need for temporary variables.

It is obvious from Table 1 that the proposed approach can significantly improve the computational efficiency. Ascribing to the comprehensive use of matrix sparsity and symmetry, the computational effort required to calculate $\boldsymbol{\Phi}_{k/k-1}\boldsymbol{P}_{k-1}\boldsymbol{\Phi}_{k/k-1}^{\mathrm{T}}$ can be almost halved, when the system state dimension is selected as 42. Furthermore, implementing the matrix exterior-product for the symmetrical matrix multiplications can also reduce the storage load by over 57%.

4. Experiment

To validate the proposed method, an aerial mapping experiment was conducted on an airplane platform equipped with the GNSS/INS integrated system. The flight experiment data were collected and processed jointly by Beihang University and Guizhou Institute of Technology.

4.1. Hardware configuration

The INS and GNSS data are acquired from two different GNSS/INS integrated systems containing a fibre optic gyroscope (FOG) IMU, a ring laser gyroscope (RLG) IMU, and two NovATel DL-V3 GNSS receivers. The two IMUs are fixed on the same mounting plate as shown in Figure 3, and the IMUs and GNSS specifications are listed in Table 2. The higher-precision GNSS/RLG_INS integrated system is used as the reference system and, meanwhile, the navigation solutions provided by the GNSS/FOG_INS integrated system are used to verify the performance of the proposed method.

4.2. Experimental arrangement

The flight experiment was conducted in Yangling, Shanxi Province, China, which lasted for approximately 1.2 hours. Before entering the survey region, the aircraft is designed to undergo a figure-of-eight manoeuvring to observe the heading error, and separate the roll and pitch errors from the horizontal accelerometer biases. During the execution of a figure-of-eight manoeuvring, a preliminary acceleration to 1.2∼1.5 times the level cruising speed is needed to establish an adequate kinetic energy

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

Table 2. *Specifications of FOG IMU, RLG IMU and DL-V3 GNSS receiver.*

IMU mode	FOG IMU	RLG IMU (Reference)
Rate	100 Hz	100 Hz
Gyroscope zero drift repeatability	<0.05°/h (1σ)	<0.01°/h (1σ)
Gyroscope zero drift stability	<0.05°/h (1σ)	<0.01°/h (1σ)
Accelerometer zero bias repeatability	<50 ug (1σ)	<20 ug (1σ)
Accelerometer zero bias stability	<50 ug (1σ)	<20 ug (1σ)
DL-V3 mode	RTK	POST-PROCESSED (Reference)
Rate	1 Hz	20 Hz
Position measurement noise	0.1 m (RMS)	0.05 m (RMS)
Velocity measurement noise	0.05 m/s (RMS)	0.01 m/s (RMS)

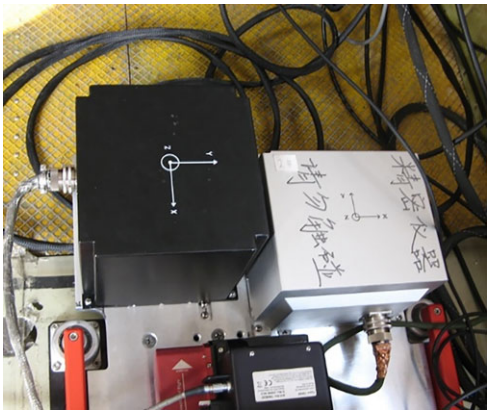


Figure 3. *Equipment installation.*

reserve prior to entry. The roll angle is typically maintained within a range of 15°–20° to generate the requisite centripetal force. Simultaneously, the pitch angle is moderately increased to approximately 8° to compensate for the loss of the vertical component of lift, thereby maintaining a constant altitude. The whole flight trajectory within the reference coordinates is shown in Figure 4.

4.3. Experiment data processing and results analysis

The GNSS measurements were post-processed using GrafNav^{TM} 8.30 software (NovAtel Inc.) in carrier phase difference mode with 0.05 m level position accuracy and 0.01 m/s level velocity accuracy, and the reference trajectories were generated by the RLG IMU/Post Processed GNSS integration using InertialExplorer^{TM} 8.30 post-processing software (NovAtel Inc.) with 0.005°/0.0025° level heading/attitude accuracy. Then, the RTK GNSS measurements were output directly by a DL-V3 receiver with 0.1 m level position accuracy and 0.05 m/s level velocity accuracy, and FOG IMU data and RTK GNSS data were integrated and processed by POSProcess^{TM} 3.0 (GNSS/INS Research Group, Beihang University and Guizhou Institute of Technology) for the loose coupling with the following various stochastic models. The specific configurations of various stochastic models are listed in Table 3.

The position and velocity solution are initialised from GNSS, and self-alignment is used to initialise the heading, pitch and roll. In the state initialisation of GNSS/INS integration according to Equation (22), the state estimates $\boldsymbol{X}_0$ should generally be initialised at zero. The initial error covariance $\boldsymbol{P}_0$, which reflects the state uncertainties and any covariance between state residuals, is assumed using

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

Table 3. *Configurations of various stochastic models.*

Error states	Model 1 (15 states, classical model)	Model 2 (27 states, commonly used model)	Model 3 (45 states, presented by Chen and Fang, 2013)	Model 4 (42 states, proposed in this paper)
Position, velocity, and attitude errors	✓	✓	✓	✓
Gyro, accelerometer static biases/drifts	✓	✓	✓	✓
Gyro, accelerometer dynamic biases/drifts		✓	✓	✓
Gyro, accelerometer scale factor errors		✓	✓	✓
Gyro, accelerometer cross-coupling errors			✓	✓
Gravity disturbances				✓
Lever-arm uncertainty errors				✓

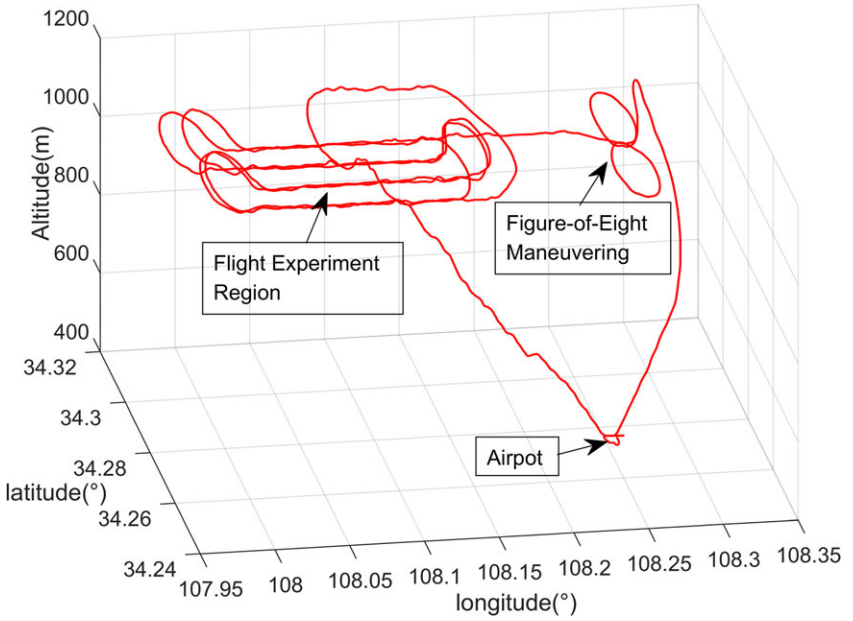


Figure 4. *Trajectory of flight.*

the prior knowledge of GNSS and INS performance parameters, and the measurement noise covariance matrix $\boldsymbol{R}$ and the system noise covariance matrix $\boldsymbol{Q}$ are determined based on device specifications in Table 2.

To verify the feasibility of the proposed stochastic model embedded in the GNSS/INS Kalman filter, the error states estimations were contrasted and analysed as follows. Figures 5 and 6 show the estimated static biases and drifts of accelerometer and gyro with three stochastic models. For model 1, the static bias and drift estimations vibrate intensely with the different flight strips in opposite directions, which means that model 1 is not capable for the fine description of GNSS/INS Kalman filter states and error covariance matrix varying with time. The behaviour of model 2 has improved significantly compared with model 1, but the estimation results are still not sufficiently ideal, i.e. the estimations of accelerometer three axes static bias and gyro Z-axis static drift are greater than 200 ug and 0.1°/h.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

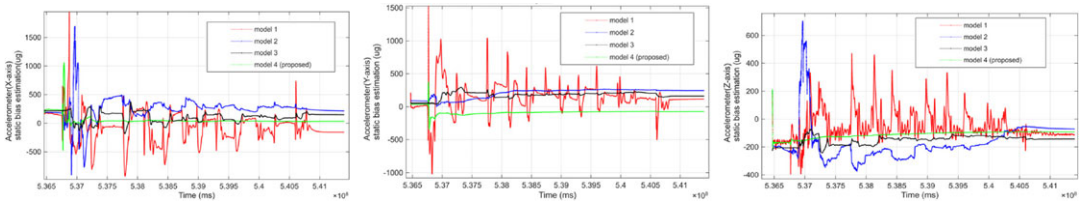


Figure 5. *Estimated acce. static biases comparison of models 1, 2, 3 and 4.*

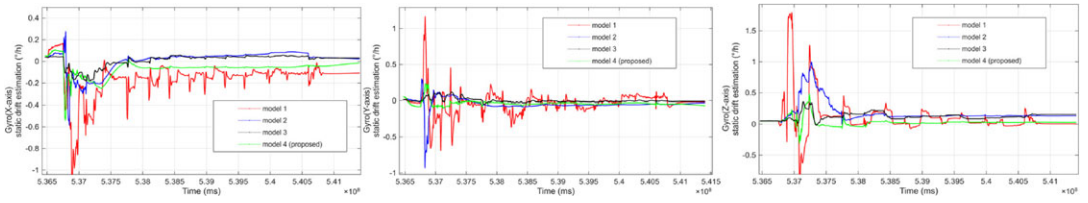


Figure 6. *Estimated gyro static drifts comparison of models 1, 2, 3 and 4.*

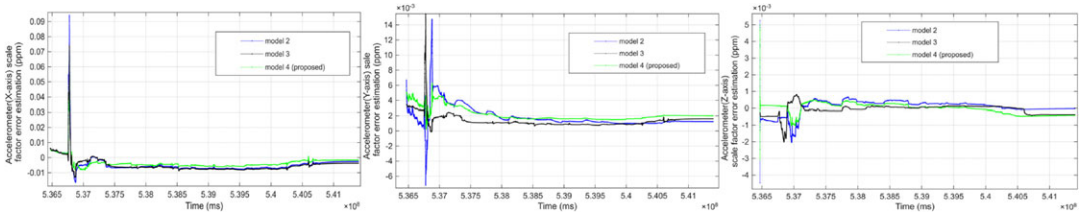


Figure 7. *Estimated acce. scale factor errors comparison of models 2, 3 and 4.*

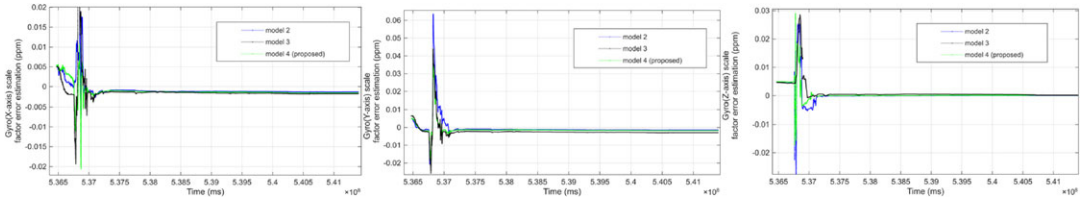


Figure 8. *Estimated gyro scale factor errors comparison of models 2, 3 and 4.*

Although achieving better results than model 2, model 3 has not yielded the expected results. The proposed model 4 has the best performance among the four stochastic models where the estimated accelerometer biases for three axes are approximately 40 ug, 80 ug, 90 ug, and the estimated gyro drifts for three axes are approximately 0.03°/h, 0.06°/h, 0.04°/h, respectively. The estimated static biases/drifts of accelerometer/gyro with the proposed model 4 are very close to the nominal value in Table 2.

Figures 7 and 8 show the estimated scale factor errors of accelerometer and gyro with model 2, model 3 and model 4. In general, the estimation effects of the proposed model 4 are almost coincident with those of model 2 and model 3 where the estimated accelerometer scale factor errors for the three axes are approximately 0.005 ppm, 0.0015 ppm, 0.0004 ppm, and the estimated gyro scale factor errors for the three axes are approximately 0.0013 ppm, 0.0015 ppm, 0.00003 ppm, respectively. However, the proposed model 4 has a faster convergence speed of the scale factor error estimations than model 2 and model 3. Figures 9 and 10 show the estimated cross-coupling errors of accelerometer and gyro with model 3 and model 4. The proposed model 4 outperforms model 3 to some extent where the

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

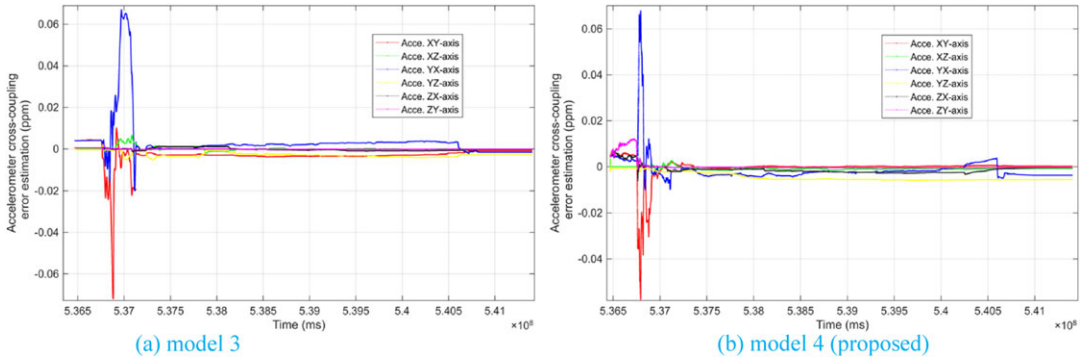


Figure 9. *Estimated acce. cross-coupling errors comparison of models 3 and 4.*

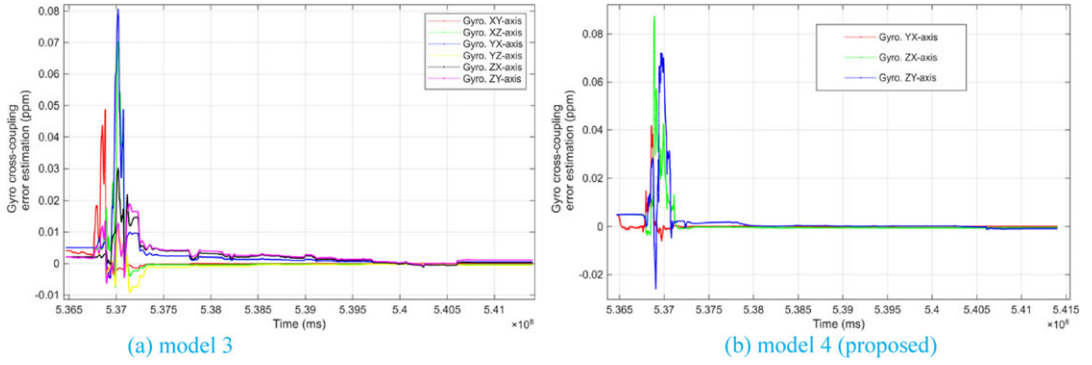


Figure 10. *Estimated gyro cross-coupling errors comparison of models 3 and 4.*

accelerometer cross-coupling error estimations for the six interaxial directions are approximately 0.0002 ppm, 0.001 ppm, 0.0025 ppm, 0.0055 ppm, 0.0025 ppm, 0.0005 ppm, and the gyro cross-coupling error estimations for the three interaxial directions are approximately 0.0001 ppm, 0.0002 ppm, 0.0004 ppm, respectively. Figure 11 shows the estimated lever-arm uncertainty errors with the proposed model 4, and the estimations for the three axes are approximately 0.16 m, 0.07 m, and 0.28 m. The estimated results of accelerometer and gyro cross-coupling errors and lever-arm uncertainty errors with the proposed model 4 are the best in accordance with expectations.

To validate the performance of the proposed stochastic model in GNSS/INS integration, the navigation accuracies with various stochastic models are compared and analysed statistically as follows. It can be seen from Figures 12–14 that the position, velocity and attitude errors of the proposed model 4 are much more stable and lower than the three other models, and meanwhile, its improvement is more evident where there is no obvious correlation between the navigation error and the manoeuvre trajectory. As seen in Table 4, the proposed model 4 produces an RMS position error of 0.2577 m, an RMS velocity error of 0.0435 m/s, an RMS heading error of 0.0176°, an RMS pitch error of 0.0044° and an RMS roll error of 0.0041°. Compared with the other three models, the proposed model 4 shows outstanding performance in GNSS/INS integration, where the percentage improvement in position, velocity, heading, pitch and roll accuracy was enhanced by 62.97% (45.11%, 27.47%), 22.32% (13.52%, 8.42%), 67.82% (54.87%, 24.46%), 39.73% (35.29%, 15.38%), and 51.76% (36.92%, 31.67%) against model 1 (model 2, model 3).

Further, to evaluate the real-time performance and computational complexity of the proposed estimation scheme, the elapsed time and memory requirement are compared between the conventional computation approach and the proposed computation approach (see Table 1). The memory requirement

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

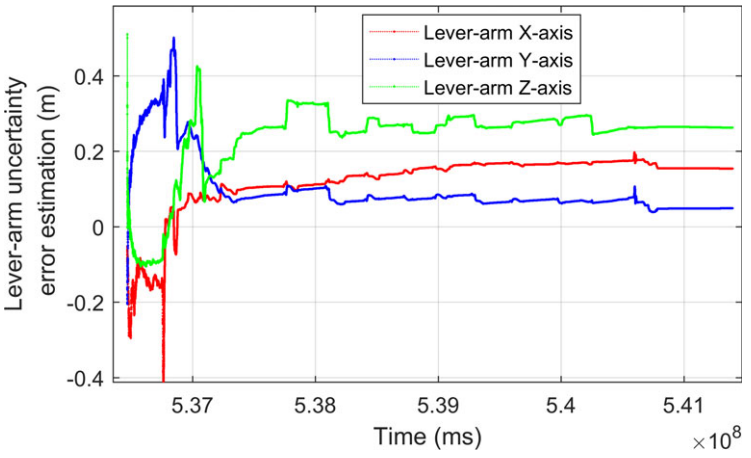


Figure 11. *Estimated lever-arm uncertainty errors of model 4.*

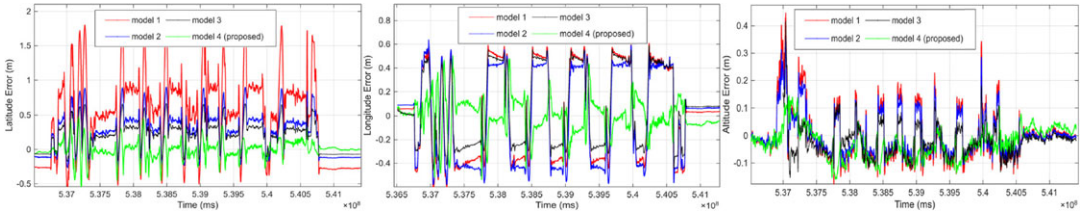


Figure 12. *Position errors comparison of models 1, 2, 3 and 4.*

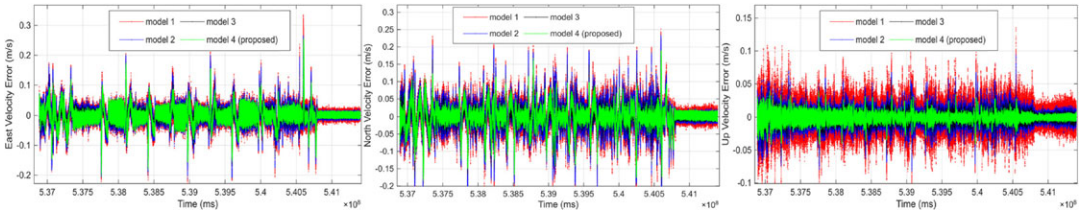


Figure 13. *Velocity errors comparison of models 1, 2, 3 and 4.*

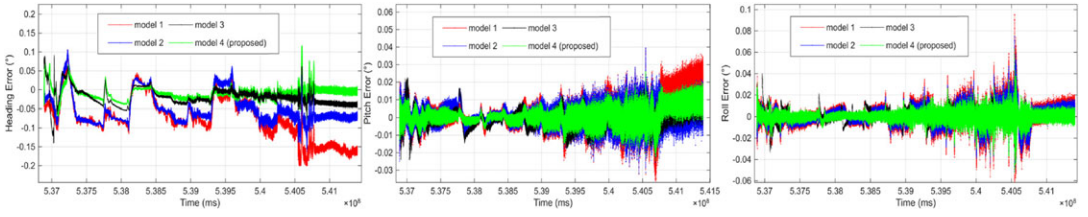


Figure 14. *Attitude errors comparison of models 1, 2, 3 and 4.*

is calculated by summing the memory sizes of all the temporary variables necessary for a complete iteration of the Kalman filter. The elapsed time is obtained by measuring the time duration of one Kalman filter iteration (see Equation (22)). The profiling approach has been considered to measure the elapsed time by MATLAB on a test machine with Intel Core i5-4200 CPU at 2.60 GHz and 4 GB of RAM. As seen in Table 5, the conventional computation approach entails a computational burden and

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use , available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

Table 4. *Navigation accuracies of GNSS/INS with various stochastic models.*

Root mean square error (RMSE)	Model 1	Model 2	Model 3	Proposed model 4
Latitude error (m)	0.5724	0.2761	0.1972	0.1314
Longitude error (m)	0.3847	0.3727	0.2874	0.2147
Altitude error (m)	0.0926	0.0730	0.0687	0.0550
Total position error (m)	0.6959	0.4695	0.3553	0.2577
East velocity error (m/s)	0.0331	0.0307	0.0292	0.0284
North velocity error (m/s)	0.0412	0.0382	0.0364	0.0323
Up velocity error (m/s)	0.0185	0.0115	0.0088	0.0064
Total velocity error (m/s)	0.0560	0.0503	0.0475	0.0435
Heading error (°)	0.0547	0.0390	0.0233	0.0176
Pitch error (°)	0.0073	0.0068	0.0052	0.0044
Roll error (°)	0.0085	0.0065	0.0060	0.0041

Table 5. *Comparison of computational complexity.*

Computational complexity (Model 4)	Conventional approach	Proposed approach
Elapsed time (ms)	0.051	0.030
Memory requirement (KB)	24.468	21.023

requires a considerable memory size for model 4 (42-dimension), due to high-dimension matrix multiplication with the dot-product method. Meanwhile, the computational complexity can be significantly reduced by using the proposed computation approach, where the processing speed is increased by more than 41.18% and the memory requirement is decreased by approximately 14.08%.

5. Conclusion

In this paper, multiple INS crucial errors are discussed and modelled according to the remote sensing survey requirement, and a more accurate high-dimensional (42 states) system error model of the GNSS/INS Kalman filter is established that comprehensively considers the influence of inertial sensor errors and approximate processing errors. Then, a real-time estimation scheme with a high-efficiency Kalman filter is proposed for the 42-dimensional system error model, which avoids heavy computational burden as well as retaining optimality. It was demonstrated through flight experiments that the proposed 42-dimensional model offers better estimation and navigation precision than the other models. Finally, compared with the conventional matrix product approach, experiment results show that the proposed estimation scheme has nearly 41.18% and 14.08% performance improvements on the processing speed and the memory requirement.

Nevertheless, this study has some limitations. First, the accelerometer and gyro biases are modelled as the sum of a constant and a Markov state, which is only a rough approximation of the sensor behaviour. Second, the gravity disturbance is modelled as an exponentially correlated fixed-variance first-order Markov process, which performs well in flat terrain regions but yields significant errors over mountain ranges and ocean trenches. To address the above limitations, future work can be pursued as along the following directions:

(1) representing the accelerometer and gyro biases as second- and third-order autoregressive models with tuning to each sensor type;
(2) using a precision gravity model (such as the 4,730,400-coefficient EGM 2008 gravity model), or a gravity gradiometer to measure gravitational variations in real time.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

Overall, the proposed high-dimensional system error model and real-time estimation scheme presented in this paper are of great significance for improving the navigation accuracy of the GNSS/INS integrated system. The aforementioned limitations do not compromise the validity of the main conclusions, but instead highlight key directions for further improvement.

Funding statement. The authors disclosed receipt of the following financial support for the research, authorship and/or publication of this paper. This work was financially supported in part by the National Natural Science Foundation of China under Grant 61763005, Guizhou Provincial Key Technology R&D Program under Grant [2026]025 and XKBF [2025]032, and Guizhou Provincial Major Scientific and Technological Program under Grant XKBF[2025]031.

Competing interests. The authors declare that they have no conflict of interest.

References

Ca, J., Fang, J. and Shen, W. (2006). Fast Data Fusion Method for Integrated Navigation System and Hardware in Loop Simulation. *Journal of Beijing University of Aeronautics and Astronautic*, **32**(11), 1281–1285.

Chen, L. and Fang, J. (2013). Improving the Error Model for POS Application to Airborne InSAR. *Proceedings of International Conference on Measurement, Instrumentation and Automation*, Guangzhou, China, 521–529.

Chen, J., Zhang, D., Han, G. and Li, Z. (2020). A Method for Lever Arm Estimation in INS/GPS Integration Using Direct Unscented Kalman Filter. *Proceedings of IEEE International Conference on Computer and Communication*, Chengdu, China, 985–990.

Cui, B., Wei X., Chen, X., Li, J. and Li, L. (2019). On Sigma-Point Update of Cubature Kalman Filter for GNSS/INS Under GNSS-Challenged Environment. *IEEE Transactions on Vehicular Technology*, **68**(9), 8671–8682.

Fang, J., Chen, L. and Yao, J. (2014). An Accurate Gravity Compensation Method for High-Precision Airborne POS. *IEEE Transactions on Geoscience and Remote Sensing*, **52**(8), 4564–4573.

Gao, P., Li, K., Song, T. and Liu, Z. (2018). An Accelerometers-Size-Effect Self-Calibration Method for Triaxis Rotational Inertial Navigation System. *IEEE Transactions on Industrial Electronics*, **65**(2), 1655–1664.

Groves, P. (2013). *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*. Norwood, USA: Artech House.

He, T., Li, X., Zhang, X. and Zhang, H. (2022). Influence of Lever-Arm and Time Synchronization Errors on Loosely Coupled INS/GNSS Integrated Navigation System. Proceedings of International Symposium on Test Automation & Instrumentation, Beijing, China, 400-406.

Hu, X. and Wang, Z. (2023). Self-Calibration for Size Effect with Constrained Virtual Points. *IEEE Transactions on Industrial Electronics*, **70**(2), 2095–2105.

Knudsen, L. (1985). Performance Accuracy (Truth Model/Error Budget) Analysis for the LN-93 Inertial Navigation Unit. California: *Litton Guidance and Control Systems Inc*.

Lewantowicz, H. and Keen, W. (1991). Graceful Degradation of GPS/INS Performance with Fewer Than Four Satellites. *Proceedings of National Technical Meeting of The Institute of Navigation*, Phoenix, AZ, USA, 269-275.

Li, S., Li, X., Zhou, Y., Wang, S. and Chen, S. (2024). Accuracy- and Simplicity-Oriented Self-Calibration Approach for In-Vehicle GNSS/INS/Vision System with Observability Analysis. *IEEE Transactions on Intelligent Transportation Systems*, **25**(3), 2681–26950.

Liu, J., Chen, X., and Wang, J. (2023). Strong Tracking UKF-Based Hybrid Algorithm and Its Application to Initial Alignment of Rotating SINS With Large Misalignment Angles. *IEEE Transactions on Industrial Electronics*, **70**(8), 8334–8343.

Lo, S. and Gao, G. (2021). *Position, Navigation, and Timing Technologies in the 21st Century: Integrated Satellite Navigation, Sensor Systems, and Civil Applications*. New York, USA: Wiley.

Luo, K., Yu, R., Wu, M., Cao, J. and Huang, Y. (2024). A Novel Gravity Disturbance Compensation Inertial Navigation Method Based on Newtonian Mechanics. *IEEE Transactions on Instrumentation and Measurement*, **73**(7), 864–876.

Manon, K., Jeroen, D. and Thomas, B. (2017). *Using Inertial Sensors for Position and Orientation Estimation*. Boston, USA: Now.

Mohinder, S., Angus, P. and Chri, G. (2020). *Global Navigation Satellite Systems, Inertial Navigation, and Integration*. New York, USA: Wiley.

Nassar, S. and El-Sheimy, N. (2006). A Combined Algorithm of Improving INS Error Modeling and Sensor Measurements for Accurate for Accurate INS/GPS Navigation. *GPS Solutions*, **10**(1), 29–39.

Skog, I. and Handel, P. (2011). Time Synchronization Errors in Loosely Coupled GPS-Aided Inertial Navigation Systems. *IEEE Transactions on Intelligent Transportation Systems*, **12**(4), 1014–1023.

Sun, H., Yuan, X. and Fu, J. (2010). The GPS/INS Integrated Positioning and Orientation in the Aerial Remote Sensing Based on the High-order INS Stochastic Error Model. *Acta Geodaetica et Cartographica Sinica*, **39**(1), 28–33.

Wang, L., Yan, Q., Fei, C. and Zheng, W. (2020). On-Line Compensation of Disturbing Gravity for Ballistic Missiles in the Launching Area. Proceedings of Chinese Control Conference, Shenyang, China, 3357-3361.

Xiong, Z., Peng, H., Wang, J., Wang, R. and Liu, J. (2015). Dynamic Calibration Method for SINS Lever-Arm Effect for HCVs. *IEEE Transactions on Aerospace and Electronic Systems*, **51**(4), 2760–2771.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660

Zhang, Q., Hu, Y. and Niu, X. (2020). Required Lever Arm Accuracy of Non-Holonomic Constraint for Land Vehicle Navigation. *IEEE Transactions on Vehicular Technology*, **69**(8), 8305–8316.

Zhong, M., Guo, J. and Zhou, D. (2018). Adaptive In-Flight Alignment of INS/GPS Systems for Aerial Mapping. *IEEE Transactions on Aerospace and Electronic Systems*, **54**(3), 1184–1196.

Zhong, Y., Chen, X., Zhou, Y. and Wang, J. (2023). Adaptive Particle Filtering with Variational Bayesian and Its Application for INS/GPS Integrated Navigation. *IEEE Sensors Journal*, **23**(17), 19757–19770.

Zhong, Y., Chen, X., Zhang, F., Sha, M. and Li, Q. (2024). Enhance Robustness Performance of Resonator Fiber Optical Gyroscope System Using Internal Model Control. *IEEE Transactions on Instrumentation and Measurement*, **73**(1), 1–12.

Cite this article: Chen, L., Gao, H., Liu, Z., and Fang, J. (2026). High-dimensional system error model and real-time estimation scheme for GNSS/INS integrated system. *The Journal of Navigation* 1–18. https://doi.org/10.1017/S0373463326101660

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA METROPOLITANA, on 15 Nov 2026 at 11:04:18, subject to the Cambridge Core terms of use , available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0373463326101660