

Article

Customizable Stochastic High-Fidelity Model of the Sensors and Camera Onboard a Fixed Wing Autonomous Aircraft

Eduardo Gallo *  and Antonio Barrientos 

Centro de Automática y Robótica, Universidad Politécnica de Madrid-Consejo Superior de Investigaciones, c/José Cutiérrez Abascal 2, 28006 Madrid, Spain; antonio.barrientos@upm.es

* Correspondence: e.gallo@alumnos.upm.es or edugallo@yahoo.com; Tel.: +34-600-643-104

Abstract: The navigation systems of autonomous aircraft rely on the readings provided by a suite of onboard sensors to estimate the aircraft state. In the case of fixed wing vehicles, the sensor suite is usually composed by triads of accelerometers, gyroscopes, and magnetometers, a Global Navigation Satellite System (GNSS) receiver, and an air data system (Pitot tube, air vanes, thermometer, and barometer), and it is often complemented by one or more digital cameras. An accurate representation of the behavior and error sources of each of these sensors, together with the images generated by the cameras, is indispensable for the design, development, and testing of inertial, visual, or visual–inertial navigation algorithms. This article presents realistic and customizable models for each of these sensors; a ready-to-use C++ implementation is released as open-source code so non-experts in the field can easily generate realistic results. The pseudo-random models provide a time-stamped series of the errors generated by each sensor based on performance values and operating frequencies obtainable from the sensor’s data sheets. If in addition, the simulated true pose (position plus attitude) of the aircraft is provided, the camera model generates realistic images of the Earth’s surface that resemble those taken with a real camera from the same pose.

Keywords: aircraft sensors; IMU model; camera model; simulation; pseudo-random



Citation: Gallo, E.; Barrientos, A.

Customizable Stochastic High-Fidelity Model of the Sensors and Camera Onboard a Fixed Wing Autonomous Aircraft. *Sensors* **2022**, *22*, 5518. <https://doi.org/10.3390/s22155518>

Academic Editor: Jaroslaw Pytko

Received: 8 June 2022

Accepted: 18 July 2022

Published: 24 July 2022

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



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

1. Introduction

The sensors onboard an autonomous aircraft measure various aspects of the aircraft *real or actual state* $\mathbf{x} = \mathbf{x}_{\text{TRUTH}}$ and provide these measurements to the aircraft guidance, navigation, and control (GNC) system. The outputs of these sensors, collectively known as the *sensed state* $\tilde{\mathbf{x}} = \mathbf{x}_{\text{SENSED}}$, represent the only link between the real but unknown actual states and the GNC system in charge of achieving an actual trajectory that deviates as little as possible from the guidance targets (Figure 1).

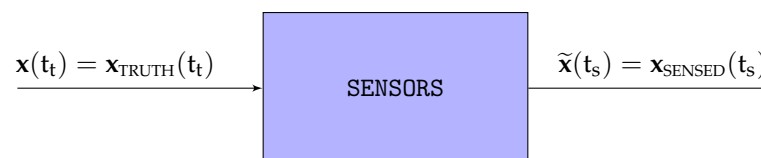


Figure 1. Sensors flow diagram.

Researchers and engineers designing, developing, or testing aircraft navigation systems require realistic renditions of the time variations of both the actual and sensed states to analyze the behavior of their algorithms in simulation. It is only after validating the algorithms under a wide range of conditions that these can be installed onboard the aircraft and field tested. The obtainment of realistic $\mathbf{x} = \mathbf{x}_{\text{TRUTH}}$ states covering the different maneuvers to be analyzed is generally achieved by discrete integration of the aircraft equations of motion coupled with the applicable guidance targets. Realistic $\tilde{\mathbf{x}} = \mathbf{x}_{\text{SENSED}}$ sensor outputs are however quite difficult to generate given their stochastic nature, the various underlying

technologies, and the lack of detailed error models from the sensor manufacturers. In addition, realistic-looking images of the Earth's surface that resemble those taken from a real aircraft are also necessary to test visual navigation algorithms.

Note that although the actual aircraft state varies continuously in the real world, in simulation, it is usually the outcome of a high-frequency discrete integration process [1] that results in $\mathbf{x}(t_t) = \mathbf{x}_{\text{TRUTH}}(t_t)$, where $t_t = t \cdot \Delta t_{\text{TRUTH}}$. The sensed trajectory $\tilde{\mathbf{x}}(t_s) = \mathbf{x}_{\text{SENSED}}(t_s)$, where $t_s = s \cdot \Delta t_{\text{SENSED}}$, is however intrinsically discrete, although the working frequency of the different sensors may vary. This article considers that all sensors operate at the same rate of Δt_{SENSED} , with the exception of the GNSS receiver and the onboard camera, which work at Δt_{GNSS} and Δt_{IMG} , respectively. It is also assumed that all sensors are fixed to the aircraft structure in a strapdown configuration and that their measurement processes are instantaneous and time synchronized with each other at their respective frequencies.

The sensed states or sensed trajectory can be defined as a time-stamped series of state vectors $\tilde{\mathbf{x}} = \mathbf{x}_{\text{SENSED}}$ that groups the measurements provided by the different onboard sensors (1) (note that when present, the super index represents the frame or reference system in which a certain variable is viewed; if two sub-indexes are present, it implies that the vector goes from the first frame to the second. For example, $\omega_{\text{IB}}^{\text{B}}$ represents the angular velocity from the inertial frame F_{I} to the body frame F_{B} viewed in body. This article makes use of the body frame F_{B} , which is rigidly attached to the aircraft structure with origin in its center of mass [2], the NED frame F_{N} also centered on the aircraft center of mass with axes in the North–East–Down directions [2], and the inertial frame F_{I} , which is usually considered as centered in the Sun with axes fixed with respect to other stars [3]), comprising the only view of the actual states at the disposal of the navigation system. Its components are listed in Table 1. Note that the *specific force* \mathbf{f}_{IB} is defined as the non-gravitational acceleration experienced by the aircraft body with respect to an inertial frame [4].

$$\tilde{\mathbf{x}} = \mathbf{x}_{\text{SENSED}} = \left[\tilde{\mathbf{f}}_{\text{IB}}^{\text{B}}, \tilde{\omega}_{\text{IB}}^{\text{B}}, \tilde{\mathbf{B}}^{\text{B}}, \tilde{\mathbf{x}}_{\text{GDT}}, \tilde{\mathbf{v}}^{\text{N}}, \tilde{p}, \tilde{T}, \tilde{\mathbf{v}}_{\text{TAS}}, \tilde{\alpha}, \tilde{\beta}, \mathbf{I} \right]^{\text{T}} \quad (1)$$

Table 1. Components of sensed trajectory.

Components	Variable	Measured by	Acronym	Rate
Specific force	$\tilde{\mathbf{f}}_{\text{IB}}^{\text{B}}$	Accelerometers	ACC	Δt_{SENSED}
Inertial angular velocity	$\tilde{\omega}_{\text{IB}}^{\text{B}}$	Gyroscopes	GYR	Δt_{SENSED}
Magnetic field	$\tilde{\mathbf{B}}^{\text{B}}$	Magnetometers	MAG	Δt_{SENSED}
Geodetic coordinates	$\tilde{\mathbf{x}}_{\text{GDT}}$	GNSS receiver	GNSS	Δt_{GNSS}
Ground velocity	$\tilde{\mathbf{v}}^{\text{N}}$	GNSS receiver	GNSS	Δt_{GNSS}
Air pressure	\tilde{p}	Barometer	OSP	Δt_{SENSED}
Air temperature	\tilde{T}	Thermometer	OAT	Δt_{SENSED}
Airspeed	$\tilde{\mathbf{v}}_{\text{TAS}}$	Pitot tube	TAS	Δt_{SENSED}
Angle of attack	$\tilde{\alpha}$	Air vanes	AOA	Δt_{SENSED}
Angle of sideslip	$\tilde{\beta}$	Air vanes	AOS	Δt_{SENSED}
Image	\mathbf{I}	Digital camera	CAM	Δt_{IMG}

Following a review of the objectives, state of the art, and novelty in Sections 1.1–1.3, the following sections provide detailed descriptions of the stochastic models representing the errors present in the measurements of the different sensors: Section 2 describes the inertial sensors (accelerometers and gyroscopes), Section 3 focuses on the magnetometers, GNSS receiver, and air data system, and Section 4 presents the tool employed to generate realistic images that resemble what a real camera would view if located at the same position and attitude. Although the camera differs from all other sensors in that it does not provide a measurement or reading but a digital image, in this article, it is indeed considered a sensor as it provides the navigation system with information about its surroundings that can be employed for navigation. Section 5 describes sensor calibration activities that are indispensable for the determination of various parameters present in the sensor models. Section 6 discusses the main characteristics of the models, with special emphasis on the

input seeds that control its stochastic properties; it also includes an example on how to customize the models for the case of a low Size, Weight, and Power (SWaP) aircraft. The conclusions are presented in Section 7.

1.1. Objectives

Sensor manufacturers do not provide models with which to estimate the errors introduced by their products, this is, the differences between the actual and sensed states. Instead, they usually publish data sheets that contain selected performance parameters in different formats and units, with few or no instructions for their interpretation. In the case of high-grade inertial sensors, an Allan variance curve is sometimes provided.

Faced with this situation, researchers willing to understand how their GNC systems will perform when the aircraft is equipped with given sensors face a difficult choice, especially in the case of low-cost sensors. One possibility is to adopt a sensor model from the literature and rely on the Allan variance curve (if available) to identify the required parameters, although it is not always clear how to do so [5]. If the Allan curve is not available, a second possibility is to obtain the curve by analyzing the sensor outputs when placed on a test bench, but the process is time consuming, requires specialized equipment and know-how, and the results are only valid for the specific tested hardware [5]. The end result is that researchers often rely on simple sensor models, which although easy to implement, fail to provide realistic outputs of the errors introduced by each sensor type. This has negative consequences for the performance of their GNC algorithms, which may not work as desired when faced with the real sensor outputs instead of the simulated ones employed for their development.

The first objective of this article is to address the need for realistic error models that can be quickly customized by the user based exclusively on the performance parameters contained in the data sheets provided by the sensor manufacturers, without the need for specific test equipment nor expert knowledge in the behavior of the different sensors. By implementing the described models or employing the provided open-source C++ code [6], the end user can quickly obtain realistic pseudo-random results without any expertise in the behavior of the different onboard sensors. Once the user introduces the desired performance parameters and operating frequencies, the simulated outputs of all onboard sensors rely exclusively on two input seeds (one identifying the airframe and the other identifying the specific flight). Different pairs of seeds can be employed as part of a Monte Carlo simulation, or the same pair can be used repeatedly in case the same outputs are required for further analysis. The use of the proposed models hence enables researchers to quickly obtain realistic time-stamped series of the values of $\tilde{\mathbf{x}} = \mathbf{x}_{\text{SENSED}}$ with which to feed their simulations.

The second objective of this article is to develop a camera model capable of generating realistic images, resembling what a real camera would record if mounted on the aircraft, so the resulting images of the Earth's surface can be employed for the development and testing of visual and visual-inertial navigation systems. The resulting Earth Viewer application, described in Section 4.2, is also released as open-source code within [6].

1.2. State of the Art

The standard means for conveying the performance of a given single-axis inertial sensor is by means of its Allan variance curve [5], even though manufacturers do not always make it available, in particular for low-cost sensors. Although standards exist for how to generate the curve [7–9], it is not always clear how to convert the Allan variance information into a suitable sensor model [5]. Various Allan curve translation methods [10–19] have been available for a long time, but [5] provides the first clear exposition of the underlying ideas, issues, and trade-offs between the different methods. More recent attempts to identify the required parameters involve the use of maximum likelihood estimators [20] as well as machine learning [21].

As described in [5], the Allan variance is a well-known time domain analysis technique originally developed to analyze the frequency stability of oscillators [22–24], which has been successfully adopted to communicate the performances of inertial sensors and to characterize their stochastic errors [11,13,15,18,19,25–29].

The various error sources that influence the output of an inertial sensor are described in multiple navigation textbooks, such as [2,4,30–33]. In addition to the single-axis sensor model described in [5], the system noise and random walk contributions of the final error of a single-axis sensor are discussed in detail in [34,35], which constitute the basis for the model presented in Section 2.2. The added difficulty of combining three inertial sensors into a triad, treated in Section 2.5, is discussed in [2,31,32]. Basic magnetometer and GNSS receiver models can also be found in these navigation textbooks.

1.3. Novelty

The main contribution of this article is that it provides customizable, stochastic, and realistic models for the errors introduced by the various sensors onboard a fixed wing aircraft, with special emphasis on the inertial ones (accelerometers and gyroscopes), without relying on the Allan variance curves, as is the case in the rest of the literature reviewed in Section 1.2. The required characteristics of the models are the following:

- *Customizable* so the user can employ the values that better resemble the performances of the specific equipment being modeled.
- *Stochastic* to properly represent the nature of the different random processes involved, while ensuring that the time variation of the errors generated by each sensor can be repeated if so desired.
- *Realistic* to provide a faithful description of the variation with time of the measurement errors, including as few simplifications as possible.

This enables researchers to quickly generate pseudo-random time-stamped series of the errors introduced by each sensor without the need for the expert know-how in the behavior of each sensor required to process the Allan variance curve nor the expensive and time-consuming process required to generate the curve independently. The results can be employed to feed Monte Carlo simulations that require the sensor readings as inputs, such as those required to analyze the behavior of GNC algorithms.

To develop comprehensive models whose parameters can be obtained exclusively from the data sheets published by the manufacturers, the authors have built on established models for the system noise and random walk contributions to single-axis sensors as well as the scale factor and cross-coupling contributions that appear when using sensor triads. The comprehensive models take into consideration the influence of the true relative pose (position plus attitude) of the sensor triad with respect to the platform as well as the uncertainty in the processor's knowledge about such poses. The contribution of the various calibration procedures on the required parameters is also discussed.

The second contribution of this article is the release of the Earth Viewer application, which is capable of providing realistic and distortion-free images of the Earth surface that resemble what a real camera would record when mounted on the aircraft. To the knowledge of the authors, this is the first time that a tool capable of considering the six degrees of freedom of the camera pose has been published. These images can be employed to test the behavior of visual and visual inertial navigation systems.

2. Inertial Sensors

Inertial sensors comprise *accelerometers* and *gyroscopes*, which measure the specific force and inertial angular velocity about a single axis, respectively [36]. An *inertial measurement unit* (IMU) encompasses multiple accelerometers and gyroscopes, usually three of each, obtaining three-dimensional measurements of the specific force and angular rate [2] viewed in the platform frame F_P (Section 2.5). However, the individual accelerometers and gyroscopes are not aligned with the F_P axes but with those of the non-orthogonal accelerometers F_A and gyroscopes F_Y frames, which are also defined in Section 2.5. The output of the inertial

sensors must hence first be transformed from the F_A and F_Y frames to F_P , as described in Sections 2.6 and 2.7, and then from the F_P frame to the body frame F_B as explained in Section 2.8, where they can be employed by the navigation system. The accelerometers and gyroscopes are assumed to be infinitesimally small and located at the IMU reference point (Section 2.8), which coincides with the origin of these three frames ($O_P = O_A = O_Y$).

The IMU is physically attached to the aircraft structure in a strapdown configuration, so both the displacement T_{BP}^B and the Euler angles $\phi^{BP} = [\psi^P, \theta^P, \zeta^P]^T$ that describe the relative position and rotation between the body F_B and platform F_P frames are constant. Accelerometers can be divided by their underlying technology into pendulous and vibrating beams, while gyroscopes are classified into spinning mass, optical (ring laser or fiber optic), and vibratory [4]. Current inertial sensor development is mostly focused on *micro machined electromechanical system* (MEMS) sensors (there exist both pendulous and vibrating beam MEMS accelerometers, but all MEMS gyroscopes are vibratory), which makes direct use of the chemical etching and batch processing techniques used by the electronics integrated circuit industry to obtain sensors with small size, low weight, rugged construction, low power consumption, low price, high reliability, and low maintenance [30]. On the negative side, the accuracy of MEMS sensors is still low, although tremendous progress has been achieved in the last two decades, and more is expected in the future.

There is no universal classification of inertial sensors according to their performance, although they can be broadly assigned into five different categories or grades: marine (submarines and spacecraft), aviation (commercial and military), intermediate (small aircraft and helicopters), tactical (unmanned air vehicles and guided weapons), and automotive (consumer) [4]. The full range of grades covers approximately six orders of magnitude of gyroscope performance and only three for the accelerometers, but higher performance is always associated with bigger size, weight, and cost. Tactical grade IMUs cover a wide range of performance values but can only provide a stand-alone navigation solution for a few minutes, while automotive grade IMUs are unsuitable for navigation.

The different errors that appear in the measurements provided by accelerometers and gyroscopes are described in Section 2.1. Section 2.2 presents a model for the measurements of a single inertial sensor, while Sections 2.3 and 2.4 focus on how to obtain the specific values for white noise and bias on which the model relies from the documentation. Section 2.5 describes the reference systems required to represent the IMU measurements. Additional errors appear when three accelerometers or gyroscopes are employed together, and these are modeled in Section 2.6 for accelerometers and Section 2.7 for gyroscopes. The analysis of the inertial sensors concludes with Section 2.9, which provide a comprehensive error model for the IMU measurements. The final model also depends on the relative position of the IMU with respect to the body frame, which is described in Section 2.8.

2.1. Inertial Sensor Error Sources

In addition to the accelerometers and gyroscopes, an IMU also contains a processor, storage for the calibration parameters, one or more temperature sensors, and a power supply. As described below, each sensor has several error sources, but each of them has four components: *fixed* contribution, *temperature*-dependent variation, *run-to-run* variation, and *in-run* variation [4,37]. The first two can be measured at the laboratory (at different temperatures) and the calibration results can be stored in the IMU so the processor can later compensate the sensor outputs based on the reading provided by the temperature sensor. Calibration, however, increases manufacturing costs, so it may be absent in the case of inexpensive sensors. The run-to-run variation results in a contribution to a given error source that varies every time the sensor is employed but remains constant within a given run. It cannot be compensated by the IMU processor but can be calibrated by the navigation system every time it is turned on with a process known as fine alignment [2,4,31]. The in-run contribution to the error sources slowly varies during execution and cannot be calibrated in the laboratory nor by the navigation system.

Let us now discuss the different sources of error that influence an inertial sensor [4,38–40]:

- The *bias* is an error exhibited by all accelerometers and gyroscopes that is independent of the underlying specific force or angular rate being measured, and it comprises the dominant contribution to the overall sensor error. It can be defined as any nonzero output when the sensor input is zero [37], and it can be divided into its static and dynamic components. The static part, also known as fixed bias, *bias offset*, turn-on bias, or bias repeatability, comprises the run-to-run variation, while the dynamic component, known as in-run bias variation, *bias drift*, or bias instability (or stability), is typically about 10% of the static part and slowly varies over periods of the order of one minute. As the bias is the main contributor to the overall sensor error, its value can be understood as a sensor quality measure. Table 2 provides approximate values for the inertial sensor biases according to the IMU grade [4].

While the bias offset can be greatly reduced through fine alignment [2,4,31], the bias drift cannot be determined and needs to be modeled as a stochastic process. It is mostly a warm-up effect that should be almost non-existent after a few minutes of operation, and it corresponds to the minimum point of the sensor's Allan curve [7–9,39]. It is generally modeled as a random walk process obtained by the integration of a white noise signal coupled with limits that represent the conclusion of the warm up process.

- The *scale factor error* is the departure of the input output gradient of the instrument from unity following unit conversion at the IMU processor. It represents a varying relationship between sensor input and output caused by aging and manufacturing tolerances. As it is a combination of fixed contribution plus temperature-dependent variation, most of it can be eliminated through calibration (Section 5.1).
- The *cross-coupling error* or non-orthogonality error is a fixed contribution that arises from the misalignment of the sensitive axes of the inertial sensors with respect to the orthogonal axes of the platform frame due to manufacturing limitations, and it can also be highly reduced through calibration. The scale factor and cross-coupling errors are in the order of 10^{-4} and 10^{-3} for most inertial sensors, although they can be higher for some low-grade gyroscopes. The cross-coupling error is equal to the sine of the misalignment, which is listed by some manufacturers.
- *System noise* or random noise is inherent to all inertial sensors and can combine electrical, mechanical, resonance, and quantization sources. It can originate at the sensor itself or at any other electronic equipment that interferes with it. System noise is a stochastic process usually modeled as white noise because its noise spectrum is approximately white, and it cannot be calibrated as there is no correlation between past and future values. A white noise process is characterized by its power spectral density (PSD), which is constant as it does not depend on the signal frequency. It corresponds to the 1 s crossing of the sensor's Allan curve [7–9,39].

System noise is sometimes referred to as random walk, which can generate confusion with the bias. The reason is that the inertial sensor outputs are always integrated to obtain ground velocity in the case of accelerometers and aircraft attitude in the case of gyroscopes. As the integration of a white noise process is indeed a random walk, the later term is commonly employed to refer to system noise. Table 3 contains typical values for accelerometer and gyroscope root PSD according to sensor grade [4].

- Other minor error sources not considered in this article are the g-dependent bias (sensitivity of spinning mass and vibratory gyroscopes to specific force), scale factor nonlinearity, and higher-order errors (spinning mass gyroscopes and pendulous accelerometers).

Table 2. Typical inertial sensor biases according to IMU grade.

IMU Grade	Accelerometer Bias [mg]	Gyroscope Bias [°/h]
Marine	0.01	0.001
Aviation	0.03–0.1	0.01
Intermediate	0.1–1	0.1
Tactical	1–10	1–100
Automotive	>10	>100

Table 3. Typical inertial sensor system noise according to IMU grade.

IMU Grade	Accelerometer Root PSD [m/s/h ^{0.5}]	Gyroscope Root PSD [°/h ^{0.5}]
Aviation	0.012	0.002
Tactical	0.06	0.03–0.1
Automotive	0.6	1

2.2. Single-Axis Inertial Sensor Error Model

As the inertial sensors provide measurements at equispaced discrete times $t_s = s \cdot \Delta t_{\text{SENSED}} = s \cdot \Delta t$, this section focuses on obtaining a discrete model for the bias and white noise errors of a single-axis inertial sensor. The results obtained here will be employed in the following sections to generate a comprehensive IMU model.

Let us consider a sensor in which the difference between its measurement at any given time $\tilde{x}(t)$ and the real value of the physical magnitude being measured at that same time $x(t)$ can be represented by a zero mean white noise Gaussian process $\eta_v(t)$ with spectral density σ_v^2 :

$$\tilde{x}(t) = x(t) + \eta_v(t) \quad (2)$$

Dividing (2) by $\Delta t_{\text{SENSED}} = \Delta t$ and integrating results in:

$$\frac{1}{\Delta t} \int_{t_0}^{t_0+\Delta t} \tilde{x}(t) dt = \frac{1}{\Delta t} \int_{t_0}^{t_0+\Delta t} [x(t) + \eta_v(t)] dt \quad (3)$$

Assuming that the measurement and real value are both constant over the integration interval (note that the stochastic process η_v cannot be considered constant over any interval) [34] yields

$$\tilde{x}(t_0 + \Delta t) = x(t_0 + \Delta t) + \frac{1}{\Delta t} \int_{t_0}^{t_0+\Delta t} \eta_v(t) dt \quad (4)$$

This expression results in the white noise sensor error $w(t)$, which is the difference between the sensor measurement $\tilde{x}(t)$ and the true value $x(t)$. Its mean and variance can be readily computed:

$$w(t_0 + \Delta t) = \frac{1}{\Delta t} \int_{t_0}^{t_0+\Delta t} \eta_v(t) dt \quad (5)$$

$$E[w(t_0 + \Delta t)] = 0 \quad (6)$$

$$\text{Var}[w(t_0 + \Delta t)] = \frac{\sigma_v^2}{\Delta t} \quad (7)$$

Based on these results, the white noise error can be modeled by a discrete random variable identically distributed to the above continuous white noise error, that is, one that results in the same mean and variance, where $N_v \sim N(0, 1)$ is a standard normal random variable:

$$w(s \Delta t_{\text{SENSED}}) = w(s \Delta t) = \frac{\sigma_v}{\Delta t^{1/2}} N_{vs} \quad (8)$$

Let us now consider a second model in which the measurement error or bias is given by a first-order random walk process or integration of a zero mean white noise Gaussian process $\eta_u(t)$ with spectral density σ_u^2 :

$$\dot{b}(t) = \eta_u(t) \rightarrow b(t_0 + \Delta t) = b(t_0) + \int_{t_0}^{t_0+\Delta t} \eta_u(t) dt \quad (9)$$

Its mean and variance can be quickly computed:

$$E[b(t_0 + \Delta t)] = E[b(t_0)] \quad (10)$$

$$\text{Var}[b(t_0 + \Delta t)] = \sigma_u^2 \Delta t \quad (11)$$

These results indicate that the bias can be modeled by a discrete random variable identically distributed to the continuous random walk above:

$$b(t_0 + \Delta t) = b(t_0) + \sigma_u \Delta t^{1/2} N_u \quad (12)$$

where $N_u \sim N(0, 1)$ is a standard normal random variable. Operating with the above expression results in the final expression for the discrete bias as well as its mean and variance:

$$b(s \Delta t) = B_0 N_{u0} + \sigma_u \Delta t^{1/2} \sum_{i=1}^s N_{ui} \quad (13)$$

$$E[b(s \Delta t)] = 0 \quad (14)$$

$$\text{Var}[b(s \Delta t)] = B_0^2 + \sigma_u^2 s \Delta t \quad (15)$$

A comprehensive single-axis sensor error model without a scale factor can hence be constructed by adding together the influence of the system noise provided by (8) and the bias given by (13) [35], while assuming that the standard normal random variables N_u and N_v are uncorrelated (note that the expected value and variance of each of the two discrete components of this sensor model coincide with those of their continuous counterparts, but their combined mean and variance provided by expressions (17) and (18) differ from that of the combination of the two continuous error models given by (5) and (9). This is the case even if considering that the two zero mean white noise Gaussian processes η_u and η_v are independent and hence uncorrelated. It is however possible to obtain a discrete model whose discrete bias and white noise components are not only identically distributed to those of their continuous counterparts [34], even adding the equivalence of covariance between the bias and the sensor error, but this results in a significantly more complex model that behaves similarly to the one above at all but has the shortest time samples after sensor initialization. The authors have decided not to do so in the model described in this article, reducing complexity with little or no loss of realism):

$$e_{BW}(s \Delta t) = \tilde{x}(s \Delta t) - x(s \Delta t) = B_0 N_{u0} + \sigma_u \Delta t^{1/2} \sum_{i=1}^s N_{ui} + \frac{\sigma_v}{\Delta t^{1/2}} N_{vs} \quad (16)$$

$$E[e_{BW}(s \Delta t)] = 0 \quad (17)$$

$$\text{Var}[e_{BW}(s \Delta t)] = B_0^2 + \sigma_u^2 s \Delta t + \frac{\sigma_v^2}{\Delta t} \quad (18)$$

The *discrete sensor error* or difference between the measurement provided by the sensor $\tilde{x}(s \Delta t_{\text{SENSED}}) = \tilde{x}(s \Delta t)$ at any given discrete time $s \Delta t$ and the real value of the physical variable being measured at that same discrete time $x(s \Delta t)$ is the combination of a bias or first-order random walk and a white noise process, and it depends on three parameters: the bias offset B_0 , the bias instability σ_u , and the white noise σ_v . The contributions of these three different sources to the sensor error as well as to its first and second integrals (gyroscopes measure angular velocity, and their output needs to be integrated once to obtain attitude, while accelerometers measure specific force and are integrated once to obtain velocity and twice to obtain position) are very different and inherent to many of the challenges encountered when employing accelerometers and gyroscopes for inertial navigation, as explained below.

Figures 2 and 3 represent the performance of a fictitious sensor of $B_0 = 1.6 \times 10^{-2}$, $\sigma_u = 4 \times 10^{-3}$, and $\sigma_v = 1 \times 10^{-3}$ working at a frequency of 100 Hz ($\Delta t = 0.01$ s), and they are intended to showcase the different behavior and relative influence on the total error of each of its three components. The figures show the theoretical variation with time of the sensor error mean (Figure 2) and standard deviation (Figure 3) given by (17) and (18) together with the average of fifty different runs. In addition, Figure 2 also includes ten of those runs to showcase the variability in results implicit to the random variables (although the data are generated at 100 Hz, for visibility purposes, the figure only employs 1 out of every 1000 points, so it appears far less noisy than the real data), while Figure 3 shows the theoretical contribution to the standard deviation of each of the three components. In

addition to the near equivalence between the theory and the average of several runs, the figures show that the bias instability is the commanding long-term factor in the deviation between the sensor measurement and its zero mean (the standard deviation of the bias instability grows with the square root of time while the other two components are constant). As discussed in Section 2.1, the bias drift or bias instability is indeed the most important quality parameter of an inertial sensor. This is also the case when the sensor output is integrated, as discussed below.

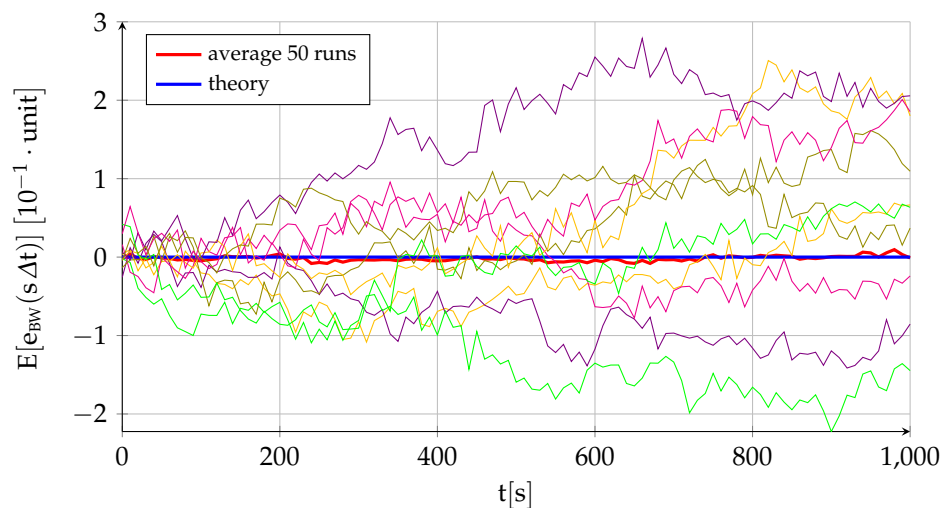


Figure 2. Propagation with time of sensor error mean.

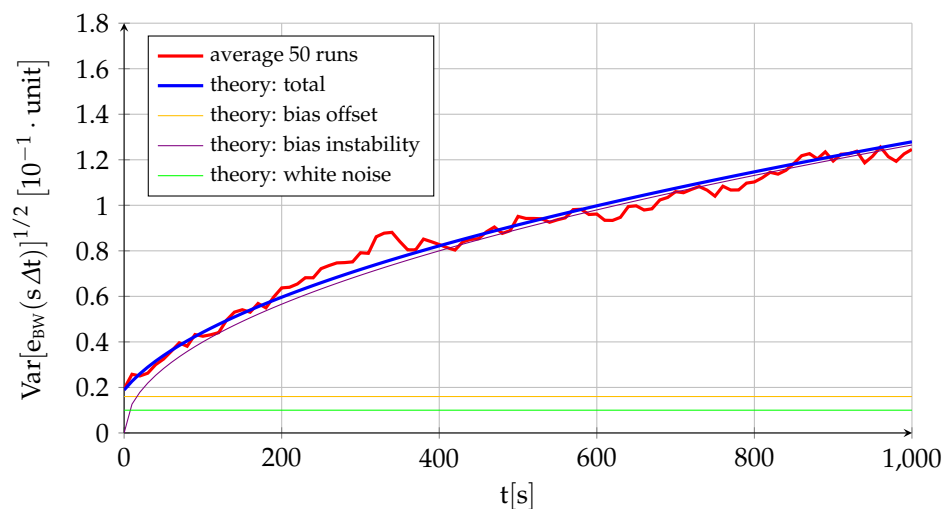


Figure 3. Propagation with time of sensor error standard deviation.

Let us integrate the sensor error over a timespan $s \Delta t$ to evaluate the growth with time of both its expected value and its variance (as the interest lies primarily in $s \gg 1$, a simple integration method such as the rectangular rule is employed):

$$\begin{aligned}
f_{\text{BW}}(s \Delta t) &= f_{\text{BW}}(0) + \int_0^{s \Delta t} e_{\text{BW}}(\tau) d\tau = f_{\text{BW}}(0) + \Delta t \sum_{i=1}^s e_{\text{BW}}(i \Delta t) \\
&= f_{\text{BW}}(0) + B_0 N_{u0} s \Delta t + \sigma_u \Delta t^{3/2} \sum_{i=1}^s (s-i+1) N_{ui} + \\
&\quad + \sigma_v \Delta t^{1/2} \sum_{i=1}^s N_{vi}
\end{aligned} \tag{19}$$

$$E[f_{\text{BW}}(s \Delta t)] = f_{\text{BW}}(0) \tag{20}$$

$$\begin{aligned}
\text{Var}[f_{\text{BW}}(s \Delta t)] &= B_0^2 (s \Delta t)^2 + \frac{\sigma_u^2}{6} \Delta t^3 s (s+1) (2s+1) + \sigma_v^2 (s \Delta t) \\
&\approx B_0^2 (s \Delta t)^2 + \frac{\sigma_u^2}{3} (s \Delta t)^3 + \sigma_v^2 (s \Delta t)
\end{aligned} \tag{21}$$

Figures 4 and 5 follow the same pattern as Figures 2 and 3 but applied to the error integral instead of to the error itself. They would represent the attitude error resulting from integrating the gyroscope output or the velocity error expected when integrating the specific force measured by an accelerometer. The conclusions are the same as before but significantly more accentuated. Not only is the expected value of the error constant instead of zero ($f_{\text{BW}}(0) = 3$ has been employed in the experiment), but the growth in the standard deviation (over a nonzero mean) is much quicker than before. The bias instability continues to be the dominating factor but now increases with a power of $t^{3/2}$, while the bias offset and white noise contributions also increase with time, although with powers of t and $t^{1/2}$, respectively. Let us continue the process and integrate the error a second time:

$$\begin{aligned}
g_{\text{BW}}(s \Delta t) &= g_{\text{BW}}(0) + \int_0^{s \Delta t} f_{\text{BW}}(\tau) d\tau = g_{\text{BW}}(0) + \Delta t \sum_{i=1}^s f_{\text{BW}}(i \Delta t) \\
&= g_{\text{BW}}(0) + f_{\text{BW}}(0) s \Delta t + \frac{B_0}{2} N_{u0} (s \Delta t)^2 + \\
&\quad + \sigma_u \Delta t^{5/2} \sum_{i=1}^s \sum_{j=1}^{s-i+1} j N_{ui} + \sigma_v \Delta t^{3/2} \sum_{i=1}^s s(s-i+1) N_{vi}
\end{aligned} \tag{22}$$

$$E[g_{\text{BW}}(s \Delta t)] = g_{\text{BW}}(0) + f_{\text{BW}}(0) (s \Delta t) \tag{23}$$

$$\begin{aligned}
\text{Var}[g_{\text{BW}}(s \Delta t)] &= \frac{B_0^2}{4} (s \Delta t)^4 + \sigma_u^2 \Delta t^5 \sum_{i=1}^s \left(\sum_{j=1}^{s-i+1} j \right)^2 + \frac{\sigma_v^2}{6} \Delta t^3 s (s+1) (2s+1) \\
&\approx \frac{B_0^2}{4} (s \Delta t)^4 + \frac{\sigma_u^2}{20} (s \Delta t)^5 + \frac{\sigma_v^2}{3} (s \Delta t)^3
\end{aligned} \tag{24}$$

Figures 6 and 7 show the same type of figures but applied to the second integral of the error ($g_{\text{BW}}(0) = 1.5$ has been employed in the experiment). In this case, the degradation of the results with time is even more intense to the point where the measurements are useless after a very short period of time. Unless corrected by the navigation system, this is equivalent to the error in position obtained by double integrating the output of the accelerometers.

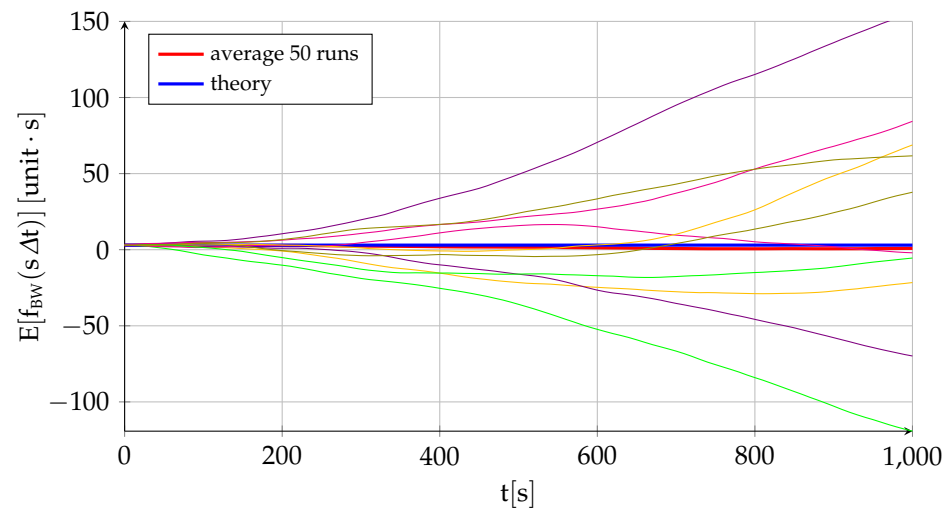


Figure 4. Propagation with time of first integral of sensor error mean.

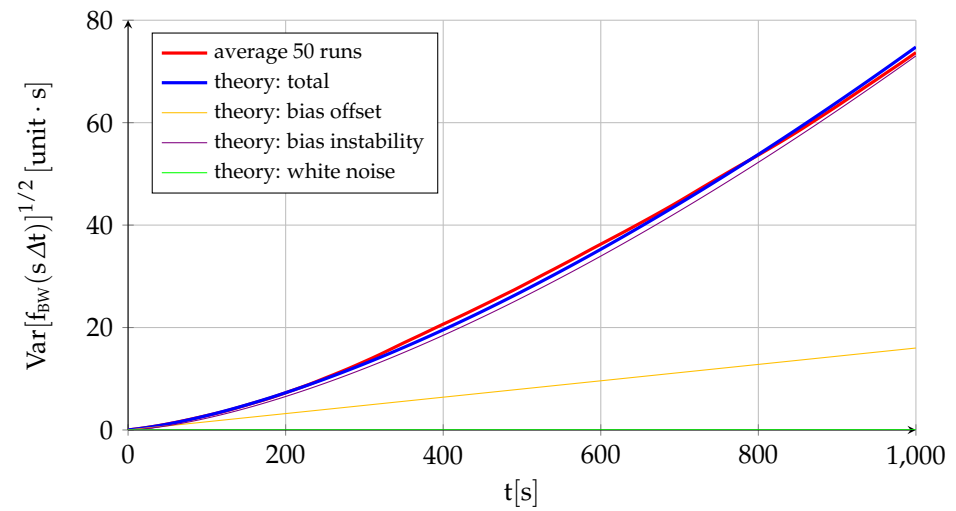


Figure 5. Propagation with time of first integral of sensor error standard deviation.

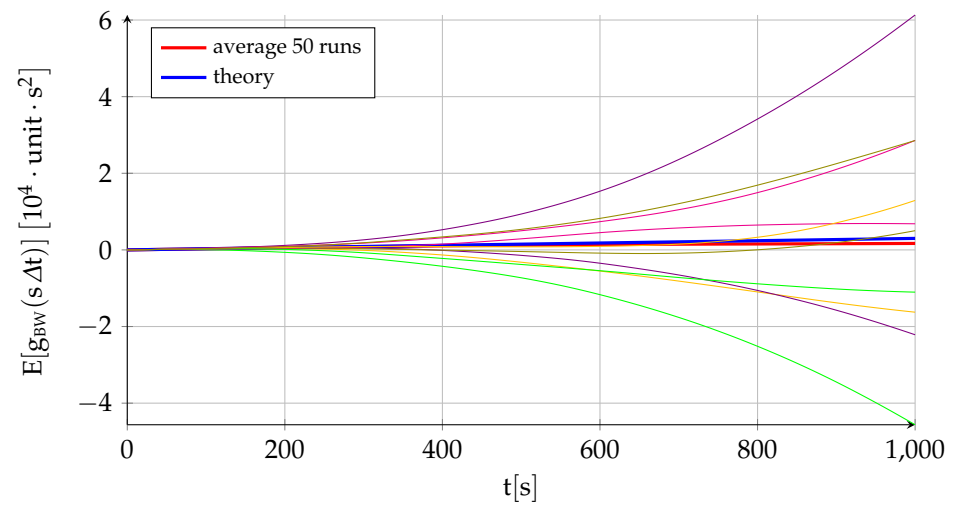


Figure 6. Propagation with time of second integral of sensor error mean.

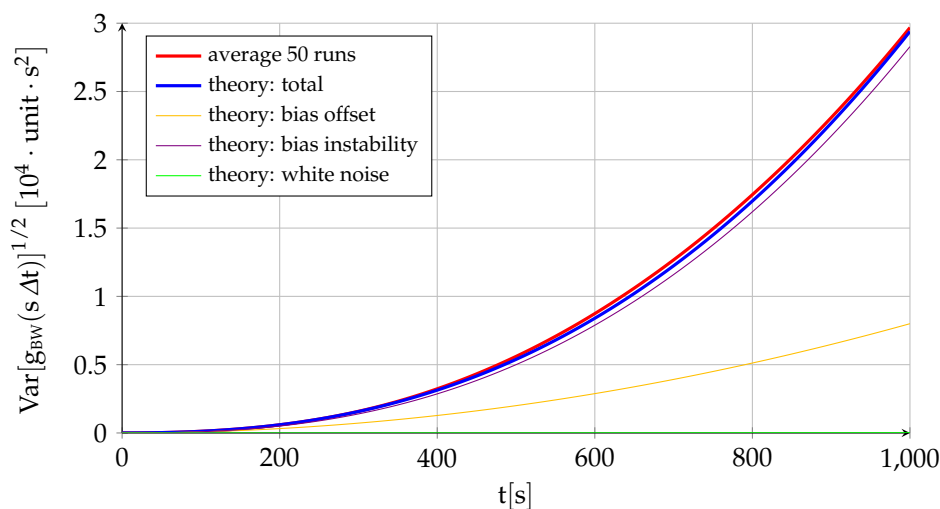


Figure 7. Propagation with time of second integral of sensor error standard deviation.

Let us summarize the main points of the single-axis inertial sensor discrete error model developed in this section, which includes the influence of the bias and the system error but not that of the scale factor and cross-coupling errors included in the three-dimensional error model of Section 2.9. The error $e_{BW}(s \Delta t)$, which applies to specific force for accelerometers and inertial angular velocity in the case of gyroscopes, depends on three factors: bias offset B_0 , bias drift σ_u , and white noise σ_v . Its mean is always zero, but the error standard deviation grows with time ($\propto t^{1/2}$) due to the bias drift with constant contributions from the bias offset and the white noise. When integrating the error to obtain $f_{BW}(s \Delta t)$, which is equivalent to ground velocity for accelerometers and attitude for gyroscopes, the initial speed error or initial attitude error $f_{BW}(0)$ becomes the fourth contributor, and an important one indeed, as it becomes the mean of the first integral error at any time. The standard deviation, which measures the spread over the nonzero mean, increases very quickly with time because of the bias instability ($\propto t^{3/2}$), with contributions from the offset ($\propto t$) and the white noise ($\propto t^{1/2}$). If integrating a second time to obtain $g_{BW}(s \Delta t)$, or position in case of the accelerometer, the initial position error $g_{BW}(0)$ turns into the fifth contributor. The expected value of the position error grows linearly with time due to the initial velocity error with an additional constant contribution from the initial position error, while the position standard deviation (measuring spread over a growing average value) grows extremely quick due mostly to the bias instability ($\propto t^{5/2}$) but also because of the bias offset ($\propto t^2$) and the white noise ($\propto t^{3/2}$). Table 4 shows the standard units of the different sources of error for both accelerometers and gyroscopes.

Table 4. Units for single-axis inertial sensor error sources.

Units	B_0	σ_u	σ_v	$f_{BW}(0)$	$g_{BW}(0)$
Accelerometer	m/s^2	$m/s^{2.5}$	$m/s^{1.5}$	m/s	m
Gyroscope	$^\circ/s$	$^\circ/s^{1.5}$	$^\circ/s^{0.5}$	$^\circ$	N/A

2.3. Obtainment of System Noise Values

This section focuses on the significance of system or white noise error σ_v and how to obtain it from sensor specifications, which often refer to the integral of the output instead of the output itself. As the integral of white noise is a random walk process, the angle random walk of a gyroscope is equivalent to white noise in the angular rate output, while velocity random walk refers to the specific force white noise in accelerometers [37]. The discussion that follows applies to gyroscopes but is fully applicable to accelerometers if replacing the angular rate by specific force and attitude or angle by ground velocity.

Angle random walk, measured in $(\text{rad}/\text{s}^{1/2})$, $(^\circ/\text{h}^{1/2})$, or equivalent, describes the average deviation or error that occurs when the sensor output signal is integrated due to system noise exclusively, without considering other error sources such as bias or scale factor [41]. If integrating multiple times and obtaining a distribution of end points at a given final time $s\Delta t$, the standard deviation of this distribution, containing the final angles at the final time, scales linearly with the white noise level σ_v , the square root of the integration step size Δt , and the square root of the number of steps s , as noted by the last term of (21). This means that an angle random walk of $1^\circ/\text{s}^{1/2}$ translates into a standard deviation for the error of 1° after 1 s, 10° after 100 s, and $1000^{1/2} \approx 31.6^\circ$ after 1000 s.

Manufacturers often provide this information as the power spectral density PSD of the white noise process in $(^\circ)^2/\text{h}^2/\text{Hz}$ or equivalent, where it is necessary to take its square root to obtain σ_v , or as the root PSD in $(^\circ/\text{h}/\text{Hz}^{1/2})$ that is equivalent to σ_v . Sometimes, it is even provided as the PSD of the random walk process, not the white noise, in units $(^\circ/\text{h})$ or equivalent. It is then necessary to multiply this number by the square root of the sampling interval Δt or divide it by the square root of the sampling frequency to obtain the desired σ_v value.

2.4. Obtainment of Bias Drift Values

This section describes the meaning of bias instability σ_u (also known as bias stability or bias drift) and how to obtain it from sensor specifications. As in the previous section, the discussion is centered on gyroscopes, but it is fully applicable to accelerometers as well. Bias instability can be defined as the potential of the sensor error to stay within a certain range for a certain time [42]. A small number of manufacturers directly provide sensor output changes over time, which directly relates with the bias instability (also known as in-run bias variation, bias drift, or rate random walk) per the second term of (18). If provided with an angular rate change of x ($^\circ/\text{s}$) (1σ) in t (s), then σ_u can be obtained as follows [34,43]:

$$\sigma_u = \frac{x}{t^{1/2}} \quad (25)$$

As the bias drift is responsible for the growth of sensor error with time (Figures 2 and 3), manufacturers more commonly provide bias stability measurements that describe how the bias of a device may change over a specified period of time [35], typically around 100 s. Bias stability is usually specified as a 1σ value with units $(^\circ/\text{h})$ or $(^\circ/\text{s})$, which can be interpreted as follows according to (16)–(18). If the sensor error (or bias) is known at a given time t , then a 1σ bias stability of $0.01^\circ/\text{h}$ over 100 s means that the bias at time $t + 100$ s is a random variable with the mean error at time t and standard deviation $0.01^\circ/\text{h}$, and expression (25) can be used to obtain σ_u . As the bias behaves as a random walk over time whose standard deviation grows proportionally to the square root of time, the bias stability is sometimes referred as a bias random walk.

In reality, bias fluctuations do not really behave as a random walk. If they did, the uncertainty in the bias of a device would grow without bound as the timespan increased, which is not the case. In practice, the bias is constrained to be within some range, and therefore, the random walk model is only a good approximation to the true process for short periods of time [35].

2.5. Platform, Accelerometers, and Gyroscopes Frames

The following sections make use of three different reference frames to describe the readings of accelerometers and gyroscopes:

- The *platform frame* F_p is a Cartesian reference system with its origin located at the IMU reference point (Section 2.8) and its three axes $\{\mathbf{i}_1^p, \mathbf{i}_2^p, \mathbf{i}_3^p\}$ forming a right-hand system that is loosely aligned with the aircraft body axes, so they point in the general directions of the aircraft fuselage (forward), aircraft wings (rightwards), and downward, respectively [2,31,32].

A proper definition of the platform frame is indispensable for navigation, as the calibrated outputs of the accelerometers and gyroscopes are based on it (Sections 2.6 and 2.7). The F_P frame can be obtained from the body frame F_B by a rotation best described by the Euler angles $\phi^{BP} = [\psi^P, \theta^P, \zeta^P]^T$ (these Euler angles correspond to the 3–2–1 (yaw, pitch, roll) convention employed in aeronautics) followed by a translation \mathbf{T}_{BP}^B (Section 2.8) from the aircraft center of mass to the IMU reference point.

- The *accelerometers frame* F_A is a non-orthogonal reference system also centered at the IMU reference point [2,31,32]. The basis vectors $\{\mathbf{i}_1^A, \mathbf{i}_2^A, \mathbf{i}_3^A\}$ are aligned with each of the three accelerometer's sensing axes (each accelerometer hence only senses the specific force component parallel to its sensing axis) (Section 2.6), but they are not orthogonal among them due to manufacturing inaccuracies. This implies that the angles between the F_A and F_P axes are very small.

It is always possible, with no loss of generality, to impose that \mathbf{i}_1^P coincides with \mathbf{i}_1^A and that \mathbf{i}_2^P is located in the plane defined by \mathbf{i}_1^A and \mathbf{i}_2^A . If this is the case, $\mathbf{i}_1^A \perp \mathbf{i}_2^P$, $\mathbf{i}_1^A \perp \mathbf{i}_3^P$, and $\mathbf{i}_2^A \perp \mathbf{i}_3^P$, and the relative attitude between the F_P and F_A frames can be defined by three independent small rotations.

- The \mathbf{i}_2^A axis can be obtained from \mathbf{i}_2^P by means of a small rotation $\alpha_{ACC,3}$ about \mathbf{i}_3^P .
- The \mathbf{i}_3^A axis can be obtained from \mathbf{i}_3^P by two small rotations: $\alpha_{ACC,1}$ about \mathbf{i}_1^P and $\alpha_{ACC,2}$ about \mathbf{i}_2^P .

Although the exact relationships can be obtained [31], and given that the angles are very small, it is possible to consider $\cos \alpha_{ACC,i} = 1$, $\sin \alpha_{ACC,i} = \alpha_{ACC,i}$, and $\alpha_{ACC,i} \cdot \alpha_{ACC,j} = 0 \forall i, j \in \{1, 2, 3\}, i \neq j$, resulting in the following transformations between free vectors viewed in the platform (\mathbf{v}^P) and accelerometer (\mathbf{v}^A) frames, respectively (As F_A is not orthogonal, the transformation matrices are denoted with \star to indicate that they are not proper rotation matrices):

$$\mathbf{v}^P = \mathbf{R}_{PA}^{\star} \mathbf{v}^A = \begin{bmatrix} 1 & 0 & 0 \\ \alpha_{ACC,3} & 1 & 0 \\ -\alpha_{ACC,2} & \alpha_{ACC,1} & 1 \end{bmatrix} \mathbf{v}^A \quad (26)$$

$$\mathbf{v}^A = \mathbf{R}_{AP}^{\star} \mathbf{v}^P = \begin{bmatrix} 1 & 0 & 0 \\ -\alpha_{ACC,3} & 1 & 0 \\ \alpha_{ACC,2} & -\alpha_{ACC,1} & 1 \end{bmatrix} \mathbf{v}^P \quad (27)$$

- The *gyroscopes frame* F_Y is similar to the accelerometers frame F_A defined above, but it is aligned with the gyroscopes' sensing axes instead of those of the accelerometers [2,31,32]. It is also a non-orthogonal reference system centered at the IMU reference point, but no simplifications can be made about the relative orientation of its axes $\{\mathbf{i}_1^Y, \mathbf{i}_2^Y, \mathbf{i}_3^Y\}$ with respect to those of F_P , so their relative attitude is defined by six small rotations $\alpha_{GYR,ij} \forall i, j \in \{1, 2, 3\}, i \neq j$, where $\alpha_{GYR,ij}$ is the rotation of \mathbf{i}_i^Y about \mathbf{i}_j^P . An approach similar to that employed with accelerometers leads to the following transformations between free vectors viewed in the platform (\mathbf{v}^P) and gyroscope (\mathbf{v}^Y) frames:

$$\mathbf{v}^P = \mathbf{R}_{PY}^{\star} \mathbf{v}^Y = \begin{bmatrix} 1 & -\alpha_{GYR,23} & \alpha_{GYR,32} \\ \alpha_{GYR,13} & 1 & -\alpha_{GYR,31} \\ -\alpha_{GYR,12} & \alpha_{GYR,21} & 1 \end{bmatrix} \mathbf{v}^Y \quad (28)$$

$$\mathbf{v}^Y = \mathbf{R}_{YP}^{\star} \mathbf{v}^P = \begin{bmatrix} 1 & \alpha_{GYR,23} & -\alpha_{GYR,32} \\ -\alpha_{GYR,13} & 1 & \alpha_{GYR,31} \\ \alpha_{GYR,12} & -\alpha_{GYR,21} & 1 \end{bmatrix} \mathbf{v}^P \quad (29)$$

2.6. Accelerometer Triad Sensor Error Model

An IMU is equipped with an accelerometer triad composed by three individual accelerometers, each of which measures the projection of the specific force over its sensing

axis as described in Section 2.2 while incurring in an error $e_{BW,ACC}$ that can be modeled as a combination of bias offset, bias drift, and white noise (16). The three accelerometers can be considered infinitesimally small and located at the IMU *reference point*, which is defined as the intersection between the sensing axes of the three accelerometers. As the accelerometer frame F_A is centered at the IMU reference point and its three non-orthogonal axes coincide with the accelerometers' sensing axes, (30) joins together the measurements of the three individual accelerometers:

$$\tilde{\mathbf{f}}_{IA}^A = \mathbf{S}_{ACC} (\mathbf{f}_{IA}^A + \mathbf{e}_{BW,ACC}^A) \tag{30}$$

where \mathbf{f}_{IA}^A is the specific force viewed in the accelerometer frame F_A , $\tilde{\mathbf{f}}_{IA}^A$ represents its measurement also viewed in F_A , $\mathbf{e}_{BW,ACC}^A$ is the error introduced by each accelerometer (16), and \mathbf{S}_{ACC} is a square diagonal matrix containing the scale factor errors $\{s_{ACC,1}, s_{ACC,2}, s_{ACC,3}\}$ for each accelerometer (Section 2.1). It is however preferred to obtain an expression in which the specific forces are viewed in the orthogonal platform frame F_P instead of the accelerometer frame F_A . As both share the same origin,

$$\tilde{\mathbf{f}}_{IP}^P = \mathbf{R}_{PA}^* \mathbf{S}_{ACC} (\mathbf{R}_{AP}^* \mathbf{f}_{IP}^P + \mathbf{e}_{BW,ACC}^A) \tag{31}$$

where \mathbf{R}_{PA}^* and \mathbf{R}_{AP}^* , defined by (26) and (27), contain the cross-coupling errors $\{\alpha_{ACC,1}, \alpha_{ACC,2}, \alpha_{ACC,3}\}$ generated by the misalignment of the accelerometer sensing axes. The scale factor and cross-coupling errors contain fixed and temperature-dependent error contributions (refer to Section 2.1) that can be modeled as normal random variables:

$$s_{ACC,i} = N(1, s_{ACC}^2) \quad \forall i \in \{1, 2, 3\} \tag{32}$$

$$\alpha_{ACC,i} = N(0, \alpha_{ACC}^2) \quad \forall i \in \{1, 2, 3\} \tag{33}$$

where s_{ACC} and α_{ACC} can be obtained from the sensor specifications. Equation (31) can be transformed to make it more useful by defining the accelerometer scale and cross-coupling error matrix \mathbf{M}_{ACC} :

$$\begin{aligned} \mathbf{M}_{ACC} &= \mathbf{R}_{PA}^* \mathbf{S}_{ACC} \mathbf{R}_{AP}^* = \begin{bmatrix} m_{ACC,11} & 0 & 0 \\ m_{ACC,21} & m_{ACC,22} & 0 \\ m_{ACC,31} & m_{ACC,32} & m_{ACC,33} \end{bmatrix} \\ &\approx \begin{bmatrix} s_{ACC,1} & 0 & 0 \\ \alpha_{ACC,3} (s_{ACC,1} - s_{ACC,2}) & s_{ACC,2} & 0 \\ \alpha_{ACC,2} (s_{ACC,3} - s_{ACC,1}) & \alpha_{ACC,1} (s_{ACC,2} - s_{ACC,3}) & s_{ACC,3} \end{bmatrix} \end{aligned} \tag{34}$$

Considering that the scale and cross-coupling errors are uncorrelated and very small, and taking into account the expressions for the mean and variance of the sum and product of two random variables [44], the different components $m_{ACC,ij}$ of \mathbf{M}_{ACC} can be obtained as follows $\forall i, j \in \{1, 2, 3\}$:

$$m_{ACC,ij} = N(1, s_{ACC}^2) \quad i = j \tag{35}$$

$$m_{ACC,ij} = N(0, [\sqrt{2} \alpha_{ACC} s_{ACC}]^2) = N(0, m_{ACC}^2) \quad i > j \tag{36}$$

$$m_{ACC,ij} = 0 \quad i < j \tag{37}$$

Let us also define the accelerometer error transformation matrix \mathbf{N}_{ACC} as

$$\mathbf{N}_{ACC} = \mathbf{R}_{PA}^* \mathbf{S}_{ACC} = \begin{bmatrix} n_{ACC,11} & 0 & 0 \\ n_{ACC,21} & n_{ACC,22} & 0 \\ n_{ACC,31} & n_{ACC,32} & n_{ACC,33} \end{bmatrix} = \begin{bmatrix} s_{ACC,1} & 0 & 0 \\ \alpha_{ACC,3} s_{ACC,1} & s_{ACC,2} & 0 \\ -\alpha_{ACC,2} s_{ACC,1} & \alpha_{ACC,1} s_{ACC,2} & s_{ACC,3} \end{bmatrix} \tag{38}$$

A process similar to that employed above leads to:

$$n_{ACC,ij} = N(1, s_{ACC}^2) \quad i = j \quad (39)$$

$$n_{ACC,ij} = N(0, \alpha_{ACC}^2 (1 + s_{ACC}^2)) \approx N(0, \alpha_{ACC}^2) \quad i > j \quad (40)$$

$$n_{ACC,ij} = 0 \quad i < j \quad (41)$$

Taking into account the expressions for the mean and variance of the sum and product of two random variables [44], and knowing that the cross-coupling errors are very small ($1 + \alpha_{ACC}^2 \approx 1$), it can be proven that the bias and white noise errors viewed in the platform frame F_P respond to a expression similar to (16):

$$\begin{aligned} \mathbf{e}_{BW,ACC}^P &= \mathbf{e}_{BW,ACC}^P(s \Delta t_{SENSED}) = \mathbf{e}_{BW,ACC}^P(s \Delta t) \\ &= \mathbf{N}_{ACC} \mathbf{e}_{BW,ACC}^A = B_{0ACC} \mathbf{N}_{u0,ACC} + \sigma_{uACC} \Delta t^{1/2} \sum_{i=1}^S \mathbf{N}_{ui,ACC} + \frac{\sigma_{vACC}}{\Delta t^{1/2}} \mathbf{N}_{vs,ACC} \end{aligned} \quad (42)$$

where each $\mathbf{N}_{u,ACC}$ and $\mathbf{N}_{v,ACC}$ is a random vector composed by three independent standard normal random variables $N(0, 1)$. Note that as the bias drift is mostly a warm-up process that stabilizes itself after a few minutes of operation, the random walk within (42) is not allowed to vary freely but is restricted to within a band of width $\pm 100 \sigma_{uACC} \Delta t^{1/2}$. The final model for the accelerometer measurements viewed in F_P results in

$$\tilde{\mathbf{f}}_{IP}^P = \mathbf{M}_{ACC} \mathbf{f}_{IP}^P + \mathbf{e}_{BW,ACC}^P \quad (43)$$

where \mathbf{M}_{ACC} is described in (34) through (37) and $\mathbf{e}_{BW,ACC}^P$ is provided by (42). This model relies on inputs for the bias offset B_{0ACC} , bias drift σ_{uACC} , white noise σ_{vACC} , scale factor error s_{ACC} , and cross-coupling error m_{ACC} . Section 6.2 provides an example on how to obtain these values from the data sheet provided by the accelerometer manufacturer.

2.7. Gyroscopes Triad Sensor Error Model

The IMU is also equipped with a triad of gyroscopes, each of which measures the projection of the inertial angular velocity over its sensing axis as described in Section 2.2. The obtainment of the gyroscope triad model is fully analogous to that of the accelerometers in the previous section, with the added difficulty that the transformation between the gyroscope frame F_Y and platform frame F_P relies on six small angles instead of three. The starting point hence is:

$$\tilde{\omega}_{IP}^P = \mathbf{R}_{PY}^* \mathbf{S}_{GYR} (\mathbf{R}_{YP}^* \omega_{IP}^P + \mathbf{e}_{BW,GYR}^Y) \quad (44)$$

where ω_{IP}^P is the inertial angular velocity viewed in the platform frame F_P , $\tilde{\omega}_{IP}^P$ represents its measurement also viewed in F_P , $\mathbf{e}_{BW,GYR}^Y$ is the error introduced by each gyroscope (16), \mathbf{S}_{GYR} is a square diagonal matrix containing the scale factor errors $\{s_{GYR,1}, s_{GYR,2}, s_{GYR,2}\}$, and \mathbf{R}_{PY}^* and \mathbf{R}_{YP}^* , defined by (28) and (29), contain the cross-coupling errors $\alpha_{GYR,12}, \alpha_{GYR,21}, \alpha_{GYR,13}, \alpha_{GYR,31}, \alpha_{GYR,23}, \alpha_{GYR,32}$ generated by the misalignment of the gyroscope sensing axes.

Operating in the same way as in Section 2.6 leads to:

$$\begin{aligned} \mathbf{e}_{BW,GYR}^P &= \mathbf{e}_{BW,GYR}^P(s \Delta t_{SENSED}) = \mathbf{e}_{BW,GYR}^P(s \Delta t) \\ &= B_{0GYR} \mathbf{N}_{u0,GYR} + \sigma_{uGYR} \Delta t^{1/2} \sum_{i=1}^S \mathbf{N}_{ui,GYR} + \frac{\sigma_{vGYR}}{\Delta t^{1/2}} \mathbf{N}_{vs,GYR} \end{aligned} \quad (45)$$

$$\tilde{\omega}_{IP}^P = \mathbf{M}_{GYR} \omega_{IP}^P + \mathbf{e}_{BW,GYR}^P \quad (46)$$

where each $\mathbf{N}_{ui,GYR}$ and $\mathbf{N}_{v,GYR}$ is a random vector composed by three independent standard normal random variables $N(0, 1)$. As in the case of the accelerometers, the random walk within (45) representing the bias drift is not allowed to vary freely but is restricted to within a band of width $\pm 100 \sigma_{uGYR} \Delta t^{1/2}$. This model relies on inputs for the bias offset B_{0GYR} , bias

drift σ_{UGYR} , white noise σ_{VGYR} , scale factor error s_{GYR} , and cross-coupling error m_{GYR} , which can be obtained from the gyroscope specifications. An example of this process is included in Section 6.2. The gyroscope scale and cross-coupling error matrix \mathbf{M}_{GYR} responds to:

$$\begin{aligned} \mathbf{M}_{\text{GYR}} &= \mathbf{R}_{\text{PY}}^* \mathbf{S}_{\text{GYR}} \mathbf{R}_{\text{YP}}^* = \begin{bmatrix} m_{\text{GYR},11} & m_{\text{GYR},12} & m_{\text{GYR},13} \\ m_{\text{GYR},21} & m_{\text{GYR},22} & m_{\text{GYR},23} \\ m_{\text{GYR},31} & m_{\text{GYR},32} & m_{\text{GYR},33} \end{bmatrix} \\ &\approx \begin{bmatrix} s_{\text{GYR},1} & \alpha_{\text{GYR},23} (s_{\text{GYR},1} - s_{\text{GYR},2}) & \alpha_{\text{GYR},32} (s_{\text{GYR},3} - s_{\text{GYR},1}) \\ \alpha_{\text{GYR},13} (s_{\text{GYR},1} - s_{\text{GYR},2}) & s_{\text{GYR},2} & \alpha_{\text{GYR},31} (s_{\text{GYR},2} - s_{\text{GYR},3}) \\ \alpha_{\text{GYR},12} (s_{\text{GYR},3} - s_{\text{GYR},1}) & \alpha_{\text{GYR},21} (s_{\text{GYR},2} - s_{\text{GYR},3}) & s_{\text{GYR},3} \end{bmatrix} \end{aligned} \quad (47)$$

$$m_{\text{GYR},ij} = N(1, s_{\text{GYR}}^2) \quad i = j \quad (48)$$

$$m_{\text{GYR},ij} = N(0, 2 \alpha_{\text{GYR}}^2 s_{\text{GYR}}^2) = N(0, m_{\text{GYR}}^2) \quad i \neq j \quad (49)$$

2.8. Mounting of Inertial Sensors

Equations (43) and (46) contain the relationships between the specific force $\mathbf{f}_{\text{IP}}^{\text{P}}$ and inertial angular velocity $\boldsymbol{\omega}_{\text{IP}}^{\text{P}}$ and their measurements $(\tilde{\mathbf{f}}_{\text{IP}}^{\text{P}}, \tilde{\boldsymbol{\omega}}_{\text{IP}}^{\text{P}})$ when evaluated and viewed in the platform frame F_{P} . However, from the point of view of the navigation system, both magnitudes need to be evaluated and viewed in the body frame F_{B} instead of F_{P} . These equations thus need to be modified so they relate $\mathbf{f}_{\text{IB}}^{\text{B}}$ with $\tilde{\mathbf{f}}_{\text{IB}}^{\text{B}}$ as well as $\boldsymbol{\omega}_{\text{IB}}^{\text{B}}$ with $\tilde{\boldsymbol{\omega}}_{\text{IB}}^{\text{B}}$, respectively, as described in Section 2.9 below. To do that, it is necessary to define the relative pose (position plus attitude) between the F_{P} and F_{B} frames, and to distinguish between the true position $\mathbf{T}_{\text{BP}}^{\text{B}}$ and attitude ϕ^{BP} , and their estimations by the IMU processor ($\hat{\mathbf{T}}_{\text{BP}}^{\text{B}}$ and $\hat{\phi}^{\text{BP}}$). Note that the IMU, represented by the platform frame F_{P} , should be mounted in the aircraft as close as possible to the center of gravity (this reduces errors, as described in Section 2.9), and it is loosely aligned with the aircraft body axes.

To increase the realism, this article assumes that the real displacement $\mathbf{T}_{\text{BP}}^{\text{B}}$ between the two frames is deterministic, while the relative rotation $\phi^{\text{BP}} = [\psi^{\text{P}}, \theta^{\text{P}}, \zeta^{\text{P}}]^{\text{T}}$ is stochastic. In this way, each simulation run exhibits a slightly different IMU platform attitude with respect to the aircraft body:

- As the IMU reference point is fixed with respect to the structure but the aircraft center of mass slowly moves as the fuel load diminishes, it is possible to establish a linear model that provides the displacement between the origins of both frames according to the aircraft mass (the aircraft masses m_{full} and m_{empty}) when the fuel tank is fully loaded or empty as inputs, as are the displacements between the IMU reference point and the aircraft center of mass $\mathbf{T}_{\text{BP,full}}^{\text{B}}$ and $\mathbf{T}_{\text{BP,empty}}^{\text{B}}$:

$$\mathbf{T}_{\text{BP}}^{\text{B}} = \mathbf{f}(\mathbf{m}) = \mathbf{T}_{\text{BP,full}}^{\text{B}} + \frac{m_{\text{full}} - \mathbf{m}}{m_{\text{full}} - m_{\text{empty}}} (\mathbf{T}_{\text{BP,empty}}^{\text{B}} - \mathbf{T}_{\text{BP,full}}^{\text{B}}) \quad (50)$$

- The platform Euler angles respond to the stochastic model provided by (51), in which each specific Euler angle is obtained as the product of the user-selected standard deviations ($\sigma_{\psi^{\text{P}}}, \sigma_{\theta^{\text{P}}}, \sigma_{\zeta^{\text{P}}}$) by a single realization of a standard normal random variable $N(0, 1)$ ($N_{\psi^{\text{P}}}, N_{\theta^{\text{P}}}$, and $N_{\zeta^{\text{P}}}$).

$$\phi^{\text{BP}} = [\sigma_{\psi^{\text{P}}} N_{\psi^{\text{P}}}, \sigma_{\theta^{\text{P}}} N_{\theta^{\text{P}}}, \sigma_{\zeta^{\text{P}}} N_{\zeta^{\text{P}}}]^{\text{T}} \quad (51)$$

Once the real pose between the F_{P} and F_{B} frames is established, it is necessary to specify its estimation employed by the IMU processor in the comprehensive model introduced in

Section 2.9, which is discussed in Section 5.2. Stochastic models are employed for both the translation $\hat{\mathbf{T}}_{BP}^B$ and rotation $\hat{\phi}^{BP}$, changing their values from one execution to the next:

$$\hat{\mathbf{T}}_{BP}^B = \mathbf{T}_{BP}^B + \left[\sigma_{\hat{\mathbf{T}}_{BP}^B} N_{\hat{\mathbf{T}}_{BP,1}^P}, \sigma_{\hat{\mathbf{T}}_{BP}^B} N_{\hat{\mathbf{T}}_{BP,2}^P}, \sigma_{\hat{\mathbf{T}}_{BP}^B} N_{\hat{\mathbf{T}}_{BP,3}^P} \right]^T \quad (52)$$

$$\hat{\phi}^{BP} = \phi^{BP} + \left[\sigma_{\hat{\phi}^{BP}} N_{\hat{\phi}^P}, \sigma_{\hat{\phi}^{BP}} N_{\hat{\theta}^P}, \sigma_{\hat{\phi}^{BP}} N_{\hat{\xi}^P} \right]^T \quad (53)$$

As in the previous case, the model relies on two user-selected standard deviations ($\sigma_{\hat{\mathbf{T}}_{BP}^B}$ and $\sigma_{\hat{\phi}^{BP}}$), as well as six realizations of a standard normal random variable $N(0, 1)$, which are denoted as $N_{\hat{\phi}^P}$, $N_{\hat{\theta}^P}$, $N_{\hat{\xi}^P}$, $N_{\hat{\mathbf{T}}_{BP,1}^P}$, $N_{\hat{\mathbf{T}}_{BP,2}^P}$, and $N_{\hat{\mathbf{T}}_{BP,3}^P}$. Section 6.2 suggests values for the five required settings (σ_{ψ^P} , σ_{θ^P} , σ_{ξ^P} , $\sigma_{\hat{\mathbf{T}}_{BP}^B}$, $\sigma_{\hat{\phi}^{BP}}$), although they can be adjusted by the user.

\mathbf{T}_{BP} can be considered quasi-stationary as it slowly varies based on the aircraft mass, and the relative position of their axes ϕ^{BP} remains constant because the IMU is rigidly attached to the aircraft structure. Although Euler angles have been employed in this section, from this point on, it is more practical to employ the rotation matrix \mathbf{R}_{BP} to represent the rotation between two different frames [45]. The time derivatives of \mathbf{T}_{BP} and \mathbf{R}_{BP} are hence zero:

$$\dot{\mathbf{T}}_{BP} = \dot{\mathbf{R}}_{BP} = \mathbf{0} \longrightarrow \mathbf{v}_{BP} = \mathbf{a}_{BP} = \boldsymbol{\omega}_{BP} = \alpha_{BP} = \mathbf{0} \quad (54)$$

2.9. Comprehensive Inertial Sensor Error Model

Two considerations are required to establish the measurement equations for the inertial sensors viewed in the body frame F_b . First, let us apply the composition rules of Appendix A.6 considering F_I as F_0 , F_B as F_1 , and F_P as F_2 , which results in:

$$\boldsymbol{\omega}_{IP} = \boldsymbol{\omega}_{IB} \quad (55)$$

$$\alpha_{IP} = \alpha_{IB} \quad (56)$$

$$\mathbf{v}_{IP} = \mathbf{v}_{IB} + \hat{\boldsymbol{\omega}}_{IB} \mathbf{T}_{BP} \quad (57)$$

$$\mathbf{a}_{IP} = \mathbf{a}_{IB} + \hat{\alpha}_{IB} \mathbf{T}_{BP} + \hat{\boldsymbol{\omega}}_{IB} \hat{\boldsymbol{\omega}}_{IB} \mathbf{T}_{BP} \quad (58)$$

Second, it is also necessary to consider that as $\hat{\mathbf{R}}_{BP}$ is a rotation matrix in which all rows and columns are unitary vectors, the projection of the F_P frame bias and white noise errors $\mathbf{e}_{BW,ACC}^P$ and $\mathbf{e}_{BW,GYR}^P$ onto the F_B frame does not change their stochastic properties:

$$\mathbf{e}_{BW,ACC}^B(s\Delta t) = \hat{\mathbf{R}}_{BP} \mathbf{e}_{BW,ACC}^P = \mathbf{B}_{0ACC} \mathbf{N}_{u0,ACC} + \sigma_{uACC} \Delta t^{1/2} \sum_{i=1}^s \mathbf{N}_{ui,ACC} + \frac{\sigma_{vACC}}{\Delta t^{1/2}} \mathbf{N}_{vs,ACC} \quad (59)$$

$$\mathbf{e}_{BW,GYR}^B(s\Delta t) = \hat{\mathbf{R}}_{BP} \mathbf{e}_{BW,GYR}^P = \mathbf{B}_{0GYR} \mathbf{N}_{u0,GYR} + \sigma_{uGYR} \Delta t^{1/2} \sum_{i=1}^s \mathbf{N}_{ui,GYR} + \frac{\sigma_{vGYR}}{\Delta t^{1/2}} \mathbf{N}_{vs,GYR} \quad (60)$$

As the inertial angular velocity does not change when evaluated in the F_B and F_P frames (55), its measurement in the body frame can be derived from (46) by first projecting it from F_B to F_P based on the real rotation matrix \mathbf{R}_{BP} and then projecting back the measurement into F_B based on the estimated rotation matrix $\hat{\mathbf{R}}_{BP}$. The bias and white noise error is also projected according to (60):

$$\tilde{\boldsymbol{\omega}}_{IB}^B = \hat{\mathbf{R}}_{BP} \mathbf{M}_{GYR} \mathbf{R}_{BP}^T \boldsymbol{\omega}_{IB}^B + \mathbf{e}_{BW,GYR}^B \quad (61)$$

The expression for the specific force measurement is significantly more complex because the back and forth transformations of the specific force between the F_B and F_P frames need to consider the influence of the lever arm \mathbf{T}_{BP} , as indicated in (58). The additional

terms introduce errors in the measurements, so as indicated in Section 2.8, it is desirable to locate the IMU as close as possible to the aircraft center of mass.

$$\hat{\mathbf{f}}_{IB}^B = \hat{\mathbf{R}}_{BP} \mathbf{M}_{ACC} \mathbf{R}_{BP}^T (\mathbf{f}_{IB}^B + \hat{\alpha}_{IB}^B \mathbf{T}_{BP}^B + \hat{\omega}_{IB}^B \hat{\omega}_{IB}^B \mathbf{T}_{BP}^B) - \hat{\alpha}_{IB}^B \hat{\mathbf{T}}_{BP}^B - \hat{\omega}_{IB}^B \hat{\omega}_{IB}^B \hat{\mathbf{T}}_{BP}^B + \mathbf{e}_{BW,ACC}^B \quad (62)$$

Note that this expression cannot be directly evaluated as the estimated values for the inertial angular velocity and acceleration ($\hat{\omega}_{IB}^B$, $\hat{\alpha}_{IB}^B$) are unknown by the IMU until obtained by the navigation filter. The IMU can however rely on the gyroscope readings, directly replacing $\hat{\omega}_{IB}^B$ with $\tilde{\omega}_{IB}^B$ and computing $\tilde{\alpha}_{IB}^B$ based on the difference between the present and previous $\tilde{\omega}_{IB}^B$ readings, resulting in:

$$\hat{\mathbf{f}}_{IB}^B = \hat{\mathbf{R}}_{BP} \mathbf{M}_{ACC} \mathbf{R}_{BP}^T (\mathbf{f}_{IB}^B + \hat{\alpha}_{IB}^B \mathbf{T}_{BP}^B + \hat{\omega}_{IB}^B \tilde{\omega}_{IB}^B \mathbf{T}_{BP}^B) - \hat{\alpha}_{IB}^B \hat{\mathbf{T}}_{BP}^B - \hat{\omega}_{IB}^B \tilde{\omega}_{IB}^B \hat{\mathbf{T}}_{BP}^B + \mathbf{e}_{BW,ACC}^B \quad (63)$$

Table 5 lists the error sources contained in the comprehensive inertial sensor error model represented by (61), (63). The first two columns list the different error sources, while the third column specifies their origin according to the criterion established in the first paragraph of Section 2.1. The section where each error is described appears on the fourth column, which is followed by the seeds (refer to Section 6 for the meaning of the terms $Y_{i,A}$ and $Y_{j,F}$) employed to ensure the results variability for different aircraft ($Y_{i,A}$) as well as different flights ($Y_{j,F}$).

Table 5. Inertial sensor error sources.

Error	Source	Description	Seeds		
Bias Offset	\mathbf{B}_{0ACC} , \mathbf{B}_{0GYR}	run-to-run	Section 2.2	$U_{j,E,ACC}$, $U_{j,E,GYR}$	$Y_{j,F}$
Bias Drift	σ_{UACC} , σ_{UGYR}	in-run	Section 2.2	$U_{j,E,ACC}$, $U_{j,E,GYR}$	$Y_{j,F}$
System Noise	σ_{VACC} , σ_{VGYR}	in-run	Section 2.2	$U_{j,E,ACC}$, $U_{j,E,GYR}$	$Y_{j,F}$
Scale Factor	S_{ACC} , S_{GYR}	fixed and T	Section 2.6, Section 2.7	$U_{i,A,ACC}$, $U_{i,A,GYR}$	$Y_{i,A}$
Cross-Coupling	\mathbf{m}_{ACC} , \mathbf{m}_{GYR}	fixed	Section 2.6, Section 2.7	$U_{i,A,ACC}$, $U_{i,A,GYR}$	$Y_{i,A}$
Lever Arm	\mathbf{T}_{BP} , $\sigma_{\hat{\mathbf{T}}_{BP}}^B$	fixed	Section 2.8	$U_{i,A,PLAT}$	$Y_{i,A}$
IMU Attitude	σ_{ψ^P} , σ_{θ^P} , σ_{ξ^P} , $\sigma_{\hat{\phi}^{BP}}$	fixed	Section 2.8	$U_{i,A,PLAT}$	$Y_{i,A}$

Note that all the required error sources (2nd column) need to be specified by the user. As an example, section 6.2 suggests values appropriate for a low SWaP aircraft. It is worth pointing out that all errors are modeled as stochastic variables or processes (with the exception of the \mathbf{T}_{BP} displacement between the body and platform frames, which is deterministic), as expressions (61), (63) rely on the errors provided by (59), (60), the scale and cross-coupling matrices given by (34), (47), and the transformations given by (50), (51), (52), (53).

In the case of the accelerometer triad, the stochastic nature of the fixed and run-to-run error contributions to the model relies on three realizations of normal distributions for the bias offset, three for the scale factor errors, three for the cross-coupling errors, and nine for the mounting errors, while the in-run error contributions require three realizations each for the bias drift and system noise at every discrete sensor measurement. The gyroscope triad is similar but requires six realizations to model the cross-coupling errors instead of three while using the same six realizations as the accelerometer triad to model the true and estimated rotation between the F_B and F_P frames.

Expressions (61), (63) can be rewritten to show the measurements as functions of the full errors (\mathbf{E}_{ACC} , \mathbf{E}_{GYR}), which represent all the errors introduced by the inertial sensors with the exception of white noise.

$$\tilde{\mathbf{f}}_{IB}^B(s \Delta t) = \mathbf{f}_{IB}^B(s \Delta t) + \mathbf{E}_{ACC}(s \Delta t) + \frac{\sigma_{VACC}}{\Delta t^{1/2}} \mathbf{N}_{vs,ACC} \quad (64)$$

$$\tilde{\boldsymbol{\omega}}_{IB}^B(s \Delta t) = \boldsymbol{\omega}_{IB}^B(s \Delta t) + \mathbf{E}_{GYR}(s \Delta t) + \frac{\sigma_{VGYR}}{\Delta t^{1/2}} \mathbf{N}_{vs,GYR} \quad (65)$$

3. Non-Inertial Sensors

This section describes the different non-inertial sensors usually installed onboard a fixed wing autonomous aircraft, such as a triad of *magnetometers* to measure the Earth's magnetic field, a *GNSS receiver* that provides absolute position and velocity measurements, and the *air data system*, which in addition to the pressure altitude and temperature also provides a measurement of the airspeed and the airflow angles.

3.1. Magnetometers

Magnetometers measure magnetic field intensity along a given direction and are very useful for estimating the aircraft heading. Although other types exist, magnetoinductive and magnetoresistive sensors are generally employed for navigation due to their accuracy and small size [4,30]. As with the inertial sensors, three orthogonal magnetometers are usually employed in a strapdown configuration to measure the magnetic field with respect to the body frame F_B .

Unfortunately, magnetometers do not only measure the Earth's magnetic field \mathbf{B} but also that generated by the aircraft permanent magnets and electrical equipment (known as hard iron magnetism) as well as the magnetic field disturbances generated by the aircraft ferrous materials (soft iron magnetism). For that reason, the magnetometers should be placed in a location inside the aircraft that minimizes these errors. On the positive side, magnetometers do not exhibit the bias instability present in inertial sensors, and the error of an individual sensor can be properly modeled by the combination of bias offset and white noise. A triad of magnetometers capable of measuring the magnetic field in three directions adds the same scale factor (nonlinearity) and cross-coupling (misalignment) errors as those present in the inertial sensors, together with the transformation between the magnetic axes and the body ones.

Modeling the behavior of a triad of magnetometers is simpler but less precise than that of inertial sensors, as the effect of the fixed hard iron magnetism is indistinguishable from that of the run-to-run bias offset, while the fixed effect of soft iron magnetism is indistinguishable from that of the scale factor and cross-coupling error matrix. This has several consequences. First of all is that magnetometers cannot be calibrated at the laboratory before being mounted in the aircraft as in the case of inertial sensors (Section 5.1) but are instead calibrated once attached to the aircraft by a process known as swinging (Section 5.3), which is less precise, as the aircraft attitude during swinging cannot be determined with so much accuracy as it would be in a laboratory setting. Second is that defining a magnetic platform frame to then transform the results into body axes serves no purpose, as the magnetometer readings are only valid, this is, contain the effects of hard and soft iron magnetism, once they are attached to the aircraft, and then they can be directly measured in body axes. Third is that percentage-wise, the errors induced by the magnetometers are bigger than those of the inertial sensors. The implemented model is the following:

$$\tilde{\mathbf{B}}^B = \mathbf{B}_{HI,MAG} + \mathbf{B}_{0,MAG} + \mathbf{M}_{MAG} \mathbf{R}_{BN} \mathbf{B}^N + \mathbf{e}_{W,MAG}^B \quad (66)$$

$$\tilde{\mathbf{B}}^B(s \Delta t) = \mathbf{B}_{HI,MAG} \mathbf{N}_{HI,MAG} + \mathbf{B}_{0,MAG} \mathbf{N}_{u0,MAG} + \mathbf{M}_{MAG} \mathbf{R}_{BN} \mathbf{B}^N + \frac{\sigma_{V,MAG}}{\Delta t^{1/2}} \mathbf{N}_{vs,MAG} \quad (67)$$

where $\tilde{\mathbf{B}}^B$ is the measurement viewed in F_B , $\mathbf{B}_{HI,MAG}$ is the fixed hard iron magnetism, $\mathbf{B}_{0,MAG}$ is the run-to-run bias offset, \mathbf{M}_{MAG} is a fixed matrix combining the effects of soft iron

magnetism with the scale factor and cross-coupling errors, and \mathbf{B}^N is the real magnetic field including local anomalies. $\mathbf{N}_{HI,MAG}$, $\mathbf{N}_{u0,MAG}$ and $\mathbf{N}_{vs,MAG}$ are uncorrelated normal vectors of size three each composed of three uncorrelated standard normal random variables $N(0, 1)$. The soft iron, scale factor and cross-coupling matrix \mathbf{M}_{MAG} does not vary with time and is computed as follows:

$$\mathbf{M}_{MAG} = \begin{bmatrix} s_{MAG} & \mathbf{m}_{MAG} & \mathbf{m}_{MAG} \\ \mathbf{m}_{MAG} & s_{MAG} & \mathbf{m}_{MAG} \\ \mathbf{m}_{MAG} & \mathbf{m}_{MAG} & s_{MAG} \end{bmatrix} \circ \mathbf{N}_{m,MAG} \quad (68)$$

In this expression, $\mathbf{N}_{m,MAG}$ contains nine outputs of a standard normal random variable $N(0, 1)$, and the symbol \circ represents the Hadamart or element-wise matrix product.

Table 6 lists the error sources contained in the magnetometer model in the same format as the previous section, noting that soft iron magnetism is included in both the scale factor and cross-coupling errors. Note that all the required error sources ($B_{HI,MAG}$, $B_{0,MAG}$, $\sigma_{v,MAG}$, s_{MAG} , \mathbf{m}_{MAG}) need to be specified by the user. As an example, section 6.2 suggests values appropriate for a low SWaP aircraft. The stochastic nature of the fixed and run-to-run error contributions to the magnetometer model relies on three realizations of normal distributions for the hard iron magnetism, three for the bias offset, three for the scale factor errors, and six for the cross-coupling errors, while the in-run error contributions require three realizations for system noise at every discrete sensor measurement.

Table 6. Magnetometer error sources.

Error	Source	Seeds
Hard Iron	$B_{HI,MAG}$	fixed
Bias Offset	$B_{0,MAG}$	run-to-run
System Noise	$\sigma_{v,MAG}$	in-run
Scale Factor	s_{MAG}	fixed
Cross Coupling	\mathbf{m}_{MAG}	fixed

Expression (67) can be rewritten to show the measurements as functions of the magnetometer full error \mathbf{E}_{MAG} , which represents all the errors introduced by the magnetometers with the exception of white noise:

$$\tilde{\mathbf{B}}^B(s \Delta t) = \mathbf{B}^B(s \Delta t) + \mathbf{E}_{MAG}(s \Delta t) + \frac{\sigma_{v,MAG}}{\Delta t^{1/2}} \mathbf{N}_{vs,MAG} \quad (69)$$

3.2. Global Navigation Satellite System Receiver

A GNSS receiver enables the determination of the aircraft position and absolute velocity based on signals obtained from various constellations of satellites, such as GPS, GLONASS, and Galileo. The position is obtained by triangulation based on the accurate satellite position and time contained within each signal. Instead of derivating the position with respect to time, which introduces noise, GNSS receivers obtain the vehicle absolute velocity by measuring the Doppler shift between the constant satellite frequencies and those measured by the receiver.

It is important to note that because of the heavy processing required to fix a position based on the satellite signals, GNSS receivers are not capable of working at the high frequencies characteristic of inertial and air data sensors, so Δt_{GNSS} is usually a multiple of Δt_{SENSED} . The position error of a GNSS receiver can be modeled as the sum of a zero mean white noise process plus slow varying ionospheric effects [33] modeled as the sum of the bias offset plus a random walk. This random walk is modeled with a frequency of 1/60 Hz

($\Delta t_{\text{ION}} = 60$ s) and linearly interpolated in between. The ground velocity error is modeled exclusively with a white noise process.

$$\mathbf{e}_{\text{GNSS,POS}}(g \Delta t_{\text{GNSS}}) = \tilde{\mathbf{x}}_{\text{GDT}} - \mathbf{x}_{\text{GDT}} = \sigma_{\text{GNSS,POS}} \mathbf{N}_{g,\text{GNSS,POS}} + \mathbf{e}_{\text{GNSS,ION}}(g \Delta t_{\text{GNSS}}) \quad (70)$$

$$\mathbf{e}_{\text{GNSS,VEL}}(g \Delta t_{\text{GNSS}}) = \tilde{\mathbf{v}}^N - \mathbf{v}^N = \sigma_{\text{GNSS,VEL}} \mathbf{N}_{g,\text{GNSS,VEL}} \quad (71)$$

$$\mathbf{e}_{\text{GNSS,ION}}(g \Delta t_{\text{GNSS}}) = \mathbf{e}_{\text{GNSS,ION}}(i \Delta t_{\text{ION}}) + \frac{r}{f_{\text{ION}}} \left(\mathbf{e}_{\text{GNSS,ION}}((i+1)\Delta t_{\text{ION}}) - \mathbf{e}_{\text{GNSS,ION}}(i \Delta t_{\text{ION}}) \right) \quad (72)$$

$$g = f_{\text{ION}} \cdot i + r \quad 0 \leq r < f_{\text{ION}} \quad (73)$$

$$\mathbf{e}_{\text{GNSS,ION}}(i \Delta t_{\text{ION}}) = B_{0,\text{GNSS,ION}} \mathbf{N}_{u0,\text{GNSS,ION}} + \sigma_{\text{GNSS,ION}} \sum_{j=1}^i \mathbf{N}_{j,\text{GNSS,ION}} \quad (74)$$

$$f_{\text{ION}} = \Delta t_{\text{ION}} / \Delta t_{\text{GNSS}} = 60 \quad (75)$$

where $\sigma_{\text{GNSS,POS}}$, $\sigma_{\text{GNSS,ION}}$, $B_{0,\text{GNSS,ION}}$, and $\sigma_{\text{GNSS,VEL}}$ are user-supplied inputs (Section 6.2 contains an example on how to fill up these values), and $\mathbf{N}_{g,\text{GNSS,POS}}$, $\mathbf{N}_{g,\text{GNSS,VEL}}$, $\mathbf{N}_{u0,\text{GNSS,ION}}$, and $\mathbf{N}_{j,\text{GNSS,ION}}$ are uncorrelated normal vectors of size three, each composed of three uncorrelated standard normal random variables $N(0, 1)$. In addition, note that as both g and f_{ION} are integers, the quotient remainder theorem guarantees that there exist unique integers i and r that comply with (73) [46].

Table 7 lists the error sources contained in the GNSS receiver model in the same format as previous sections. Note that all errors are modeled as stochastic variables or processes. Three realizations of a normal distribution are required for the run-to-run error contributions, while the in-run error contributions require three realizations each for position and velocity at every discrete sensor measurement, plus an extra three when corresponding for the ionospheric error.

Table 7. GNSS receiver error sources.

Error	Source	Seeds
Bias Offset	$B_{0,\text{GNSS,ION}}$	$u_{j,\text{E,GNSS}}$ $Y_{j,\text{F}}$
System Noise	$\sigma_{\text{GNSS,POS}}$, $\sigma_{\text{GNSS,VEL}}$, $\sigma_{\text{GNSS,ION}}$	$v_{j,\text{E,GNSS}}$ $Y_{j,\text{F}}$

3.3. Air Data System

The mission of the air data system is to measure the aircraft pressure altitude H_p [47,48] by means of the atmospheric pressure p , the outside air temperature T , the airspeed v_{TAS} , and the angles of attack α and sideslip β that provide the orientation of the aircraft structure with respect to the airflow.

A barometer or static pressure sensor, generally part of the Pitot tube as explained below [30], measures atmospheric pressure, which can be directly translated into pressure altitude [47,48]. Air data systems are also equipped with a thermometer to measure the external air temperature T . The implemented models, where OSP stands for outside static pressure and OAT means outside air temperature, include contributions from both bias offsets ($B_{0,\text{OSP}}$, $B_{0,\text{OAT}}$) and random noises (σ_{OSP} , σ_{OAT}):

$$e_{\text{OSP}}(s \Delta t_{\text{SENSED}}) = e_{\text{OSP}}(s \Delta t) = \tilde{p}(s \Delta t) - p(s \Delta t) = B_{0,\text{OSP}} N_{0,\text{OSP}} + \sigma_{\text{OSP}} N_{s,\text{OSP}} \quad (76)$$

$$e_{\text{OAT}}(s \Delta t_{\text{SENSED}}) = e_{\text{OAT}}(s \Delta t) = \tilde{T}(s \Delta t) - T(s \Delta t) = B_{0,\text{OAT}} N_{0,\text{OAT}} + \sigma_{\text{OAT}} N_{s,\text{OAT}} \quad (77)$$

where $N_{0,\text{OSP}}$, $N_{s,\text{OSP}}$, $N_{0,\text{OAT}}$, and $N_{s,\text{OAT}}$ are uncorrelated standard normal random variables $N(0, 1)$.

A *Pitot probe* is a tube with no outlet pointing directly into the undisturbed air stream, where the values of the air variables (temperature, pressure, and density) at its dead end resemble the total or stagnation variables of the atmosphere prior to its deceleration inside

the Pitot [49]. Measuring the air flow total pressure p_t at the tube dead end enables the estimation of the aircraft airspeed v_{TAS} .

The air data system is also capable of measuring the direction of the air stream with respect to the aircraft, which is represented by the angles of attack α and sideslip β . To do so, it can be equipped with two air vanes that align themselves with the unperturbed air stream or with a more complex multi-hole Pitot probe. In all three cases, the errors can also be modeled by a combination of bias offsets (B_{0TAS} , B_{0AOA} , B_{0AOS}) and random noises (σ_{TAS} , σ_{AOA} , σ_{AOS}):

$$e_{TAS}(s \Delta t_{SENSED}) = e_{TAS}(s \Delta t) = \tilde{v}_{TAS}(s \Delta t) - v_{TAS}(s \Delta t) = B_{0TAS} N_{0,TAS} + \sigma_{TAS} N_{s,TAS} \quad (78)$$

$$e_{AOA}(s \Delta t_{SENSED}) = e_{AOA}(s \Delta t) = \tilde{\alpha}(s \Delta t) - \alpha(s \Delta t) = B_{0AOA} N_{0,AOA} + \sigma_{AOA} N_{s,AOA} \quad (79)$$

$$e_{AOS}(s \Delta t_{SENSED}) = e_{AOS}(s \Delta t) = \tilde{\beta}(s \Delta t) - \beta(s \Delta t) = B_{0AOS} N_{0,AOS} + \sigma_{AOS} N_{s,AOS} \quad (80)$$

where $N_{0,TAS}$, $N_{s,TAS}$, $N_{0,AOA}$, $N_{s,AOA}$, $N_{0,AOS}$, and $N_{s,AOS}$ are uncorrelated standard normal random variables $N(0, 1)$. Table 8 lists the error sources contained in the air data sensor model represented by (76), (77), (78), (79), and (80) in the same format as previous tables. Note that all errors are modeled as stochastic variables or processes. The stochastic nature of the run-to-run error contributions to the models relies on five realizations of normal distributions for the bias offsets, while the in-run error contributions require five realizations for the system noises at every discrete sensor measurement.

Table 8. Air data sensor error sources.

Error	Source	Seeds
Bias Offset	$B_{0OSP}, B_{0OAT}, B_{0TAS}, B_{0AOA}, B_{0AOS}$	run-to-run
System Noise	$\sigma_{OSP}, \sigma_{OAT}, \sigma_{TAS}, \sigma_{AOA}, \sigma_{AOS}$	in-run
		$u_{j,EOSP}, u_{j,EOAT}, u_{j,ETAS}, u_{j,EAOA}, u_{j,EAOS}$
		$Y_{j,F}$

4. Camera

Image generation is a power and data-intensive process that cannot work at the high frequencies characteristic of inertial and air data sensors, so Δt_{IMG} is usually significantly higher than Δt_{SENSED} but not as much as Δt_{GNSS} . The camera is considered rigidly attached to the aircraft structure, and it is assumed that the shutter speed is sufficiently high so that all images are equally sharp, and that the image generation process is instantaneous. In addition, the camera ISO setting remains constant during the flight, and all generated images are noise free. The model also assumes that the visible spectrum radiation reaching all patches of the Earth's surface remains constant, and the terrain is considered Lambertian [50], so its appearance at any given time does not vary with the viewing direction. The combined use of these assumptions implies that a given terrain object is represented with the same luminosity in all images, even as its relative pose (position and attitude) with respect to the camera varies. Geometrically, a perspective projection or pinhole camera model [50] is employed, which in addition is perfectly calibrated and hence shows no distortion. Table 9 lists the required configuration parameters, which shall be provided by the user.

Table 9. Camera parameters.

Parameter	Symbol	Unit
Focal length	f	mm
Image width	S_H	px
Image height	S_V	px
Pixel size	S_{PX}	mm/px
Principal point horizontal location	c_1^{IMG}	px
Principal point vertical location	c_2^{IMG}	px
Horizontal field of view	Θ_H	°
Vertical field of view	Θ_V	°

4.1. Mounting of Camera

The digital camera can be located anywhere on the aircraft structure as long as its view of the terrain is unobstructed by other platform elements. It is desirable that the lever arm or distance between the camera optical center and the aircraft center of mass is as small as possible to reduce the negative effects of any camera alignment error. With respect to its orientation, the camera should be facing down to show a balanced view of the ground during level flight, but minor deviations are not problematic.

As in the case of the IMU platform, the model considers that the camera location is deterministic but its orientation is stochastic. The expressions below are hence analogous to those employed in Section 2.8, where each specific camera Euler angle is obtained as the product of the standard deviations (σ_{ψ^C} , σ_{θ^C} , σ_{ξ^C}) by a single realization of a standard normal random variable $N(0, 1)$ (N_{ψ^C} , N_{θ^C} , and N_{ξ^C}):

$$\mathbf{T}_{BC}^B = \mathbf{f}(\mathbf{m}) = \mathbf{T}_{BC,full}^B + \frac{\mathbf{m}_{full} - \mathbf{m}}{\mathbf{m}_{full} - \mathbf{m}_{empty}} \left(\mathbf{T}_{BC,empty}^B - \mathbf{T}_{BC,full}^B \right) \quad (81)$$

$$\phi_{BC} = \left[90^\circ + \sigma_{\psi^C} N_{\psi^C}, \sigma_{\theta^C} N_{\theta^C}, \sigma_{\xi^C} N_{\xi^C} \right]^T \quad (82)$$

In addition to the true translation and rotation between the F_B and F_C frames given by the previous equations, the model also requires the accuracy with which they are known to the navigation system. The determination of the camera position $\hat{\mathbf{T}}_{BC}^B$ and rotation $\hat{\phi}^{BC} = [\hat{\psi}^C, \hat{\theta}^C, \hat{\xi}^C]^T$ is discussed in Section 5.4. As in previous cases, stochastic models are considered for both the translation $\hat{\mathbf{T}}_{BC}^B$ and rotation $\hat{\phi}^{BC}$, changing their values from one run to another:

$$\hat{\mathbf{T}}_{BC}^B = \mathbf{T}_{BC}^B + \left[\sigma_{\hat{\mathbf{T}}_{BC}^B} N_{\hat{\mathbf{T}}_{BC,1}^B}, \sigma_{\hat{\mathbf{T}}_{BC}^B} N_{\hat{\mathbf{T}}_{BC,2}^B}, \sigma_{\hat{\mathbf{T}}_{BC}^B} N_{\hat{\mathbf{T}}_{BC,3}^B} \right]^T \quad (83)$$

$$\hat{\phi}^{BC} = \phi_{BC} + \left[\sigma_{\hat{\phi}^{BC}} N_{\hat{\phi}^C}, \sigma_{\hat{\phi}^{BC}} N_{\hat{\theta}^C}, \sigma_{\hat{\phi}^{BC}} N_{\hat{\xi}^C} \right]^T \quad (84)$$

where $N_{\hat{\phi}^C}$, $N_{\hat{\theta}^C}$, $N_{\hat{\xi}^C}$, $N_{\hat{\mathbf{T}}_{BC,1}^B}$, $N_{\hat{\mathbf{T}}_{BC,2}^B}$, and $N_{\hat{\mathbf{T}}_{BC,3}^B}$ are six realizations of a standard normal random variable $N(0, 1)$. Section 6.2 provides an example of the standard deviations required to fill up the model, although they can be adjusted by the user.

The translation \mathbf{T}_{BC}^B between the origins of the F_B and F_C frames can be considered quasi-stationary, as it slowly varies based on the aircraft mass (81), and the relative position of their axes ϕ_{BC} remains constant because the camera is rigidly attached to the aircraft structure (82).

4.2. Earth Viewer

The camera model differs from all other sensor models described in this article in that it does not return a sensed variable $\tilde{\mathbf{x}}$ consisting of its real value \mathbf{x} plus a sensor error \mathbf{E} but instead generates a digital image simulating what a real camera would record based on the aircraft position and attitude as given by the actual or real state $\mathbf{x} = \mathbf{x}_{TRUTH}$. When provided with the camera pose with respect to the Earth at equally time-spaced intervals, the available model implementation [6] is capable of generating images that resemble the view of the Earth's surface that the camera would record if located at that particular pose. To do so, it relies on three software libraries:

- OpenSceneGraph [51] is an open-source high-performance 3D graphics toolkit written in C++ and OpenGL, used by application developers in fields such as visual simulation, games, virtual reality, scientific visualization and modeling. The library enables the representation of objects in a scene by means of a graph data structure, which allows grouping objects that share some properties to automatically manage rendering properties such as the level of detail necessary to faithfully draw the scene but without

considering the unnecessary detail that slows down the graphics hardware drawing the scene.

- osgEarth [52] is a dynamic and scalable 3D Earth surface rendering toolkit that relies on OpenSceneGraph, and it is based on publicly available *orthoimages* of the area flown by the aircraft. Orthoimages consist of aerial or satellite imagery geometrically corrected such that the scale is uniform; they can be used to measure true distances as they are accurate representations of the Earth surface, having been adjusted for topographic relief, lens distortion, and camera tilt. When coupled with a *terrain elevation model*, osgEarth is capable of generating realistic images based on the camera position as well as its yaw and pitch, but it does not accept the camera roll (in other words, the osgEarth images are always aligned with the horizon).
- Earth Viewer is a modification to osgEarth implemented by the authors, so it is also capable of accepting the bank angle of the camera with respect to the NED axes. Earth Viewer is capable of generating realistic Earth images as long as the camera height over the terrain is significantly higher than the vertical relief present in the image. As an example, Figure 8 shows two different views of a volcano in which the dome of the mountain, having very steep slopes, is properly rendered.

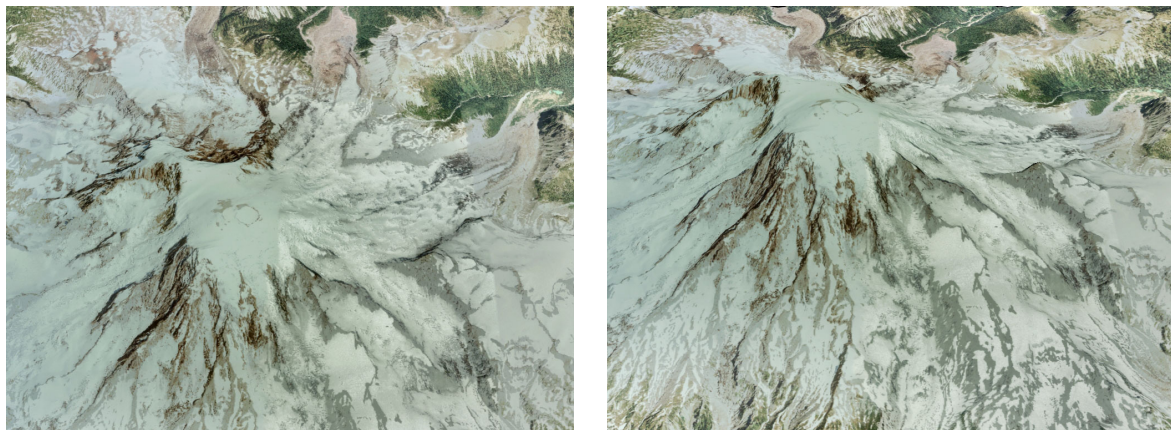


Figure 8. Example of Earth Viewer images.

5. Calibration Procedures

This section describes various calibration processes required for the determination of the fixed and run-to-run error contributions to the accelerometers, gyroscopes, magnetometers, and onboard camera. These procedures only need to be executed once and do not need to be repeated unless the sensors are replaced or their position inside the aircraft is modified (in addition, the swinging process of Section 5.3 needs to be performed every time new equipment is installed inside the aircraft, as this may modify the hard and soft iron magnetism and hence the magnetometer readings).

The calibration procedures include the laboratory calibration of the accelerometers and gyroscopes described in Section 5.1, the determination of the pose between the platform and body frames explained in Section 5.2, the magnetometer calibration or swinging described in Section 5.3, and the determination of the pose between the camera and body frames explained in Section 5.4. Their main objective is the determination of the fixed contributions to the sensor error models (refer to Section 2.1 for the different types of sensor error contributions, including fixed, run-to-run, and in-run), that is, the scale factor and cross-coupling errors of both inertial sensors and magnetometers ($\hat{\mathbf{M}}_{\text{ACC}}$, $\hat{\mathbf{M}}_{\text{GYR}}$, $\hat{\mathbf{M}}_{\text{MAG}}$) (note that $\hat{\mathbf{M}}_{\text{MAG}}$ also includes the soft iron magnetism), the magnetometers hard iron magnetism $\hat{\mathbf{B}}_{\text{HI,MAG}}^{\text{B}}$, the body to platform transformation ($\hat{\mathbf{T}}_{\text{BP}}^{\text{B}}$, $\hat{\phi}^{\text{BP}}$), and the body-to-camera transformation ($\hat{\mathbf{T}}_{\text{BC}}^{\text{B}}$, $\hat{\phi}^{\text{BC}}$). These procedures also provide estimations for the run-to-run error contributions ($\hat{\mathbf{B}}_{0\text{ACC}}$, $\hat{\mathbf{B}}_{0\text{GYR}}$, $\hat{\mathbf{B}}_{0\text{MAG}}$), but these need to be discarded, as they change every time the aircraft systems are switched on.

5.1. Inertial Sensors Calibration

Calibration is the process of comparing instrument outputs with known references to determine coefficients that force the outputs to agree with the references over a range of output values [31]. The IMU inertial sensors need to be calibrated to eliminate the fixed errors originated from manufacturing and also to determine their temperature sensitivity [32]. The calibration process requires significant material and time resources, but it greatly reduces the measurement errors. While high-grade IMUs are always factory calibrated, low-cost ones generally are not, so it is necessary to calibrate the IMU at the laboratory before mounting it on the aircraft [4].

The calibration process is executed at a location where the position \mathbf{x}_{GDT} and the gravity (gravity includes both gravitation and centrifugal accelerations) vector \mathbf{g}_c have been previously determined with great precision [31]. It relies on a three-axis table, which enables rotating the IMU with known angular velocities into a set of predetermined precisely controlled orientations [32,53]. Accelerometer and gyroscope measurements are then compared to reference values (gravity for the accelerometers, torquing rate plus Earth angular velocity for the gyroscopes) and the differences employed to generate corrections [31].

During calibration, the amount of time that the IMU is maintained stationary at each attitude, as well as the time required to rotate it between two positions, are trade-offs based on two opposing influences. On one side, longer periods of time are preferred as the negative influence of the system noise in the measurements tends to even out over time, while on the other, shorter times imply smaller variations of the bias drift over the measurement interval.

It is worth noting that as the calibration is performed before the IMU is installed on the aircraft, it relies on the platform frame F_P and the models contained in Sections 2.6 and 2.7. Although it is possible to use a calibration strategy based on selecting platform orientations that isolate sensor input onto a single axis (for example, gravity will only be sensed by the accelerometer that is placed vertically with respect to the Earth's surface) to then apply least squares techniques, in real life, it is better to employ state estimation techniques (the estimation filter not only relies on known gravity and angular velocity but also the fact that the IMU is stationary and hence its velocity is zero) to obtain estimates of the inertial sensor's scale factors, cross-coupling errors, and bias offsets [4,32]. The process is repeated at different temperatures so the IMU processor can later apply the correction based on the IMU sensor temperature [4].

The twenty-one coefficients estimated in the calibration process are listed in Table 10. Once the coefficients have been estimated, they can be introduced into the IMU processor so it automatically performs the corrections contained in (85) and (86):

$$\tilde{\mathbf{f}}_{\text{IP}}^{\text{P}} = \hat{\mathbf{M}}_{\text{ACC}}^{-1} \tilde{\mathbf{f}}_{\text{IP}}^{\text{P}} - \hat{\mathbf{B}}_{0\text{ACC}} \quad (85)$$

$$\tilde{\boldsymbol{\omega}}_{\text{IP}}^{\text{P}} = \hat{\mathbf{M}}_{\text{GYR}}^{-1} \tilde{\boldsymbol{\omega}}_{\text{IP}}^{\text{P}} - \hat{\mathbf{B}}_{0\text{GYR}} \quad (86)$$

This article assumes that the bias offset is exclusively a run-to-run source of error that varies every time the IMU is switched on, so the $\hat{\mathbf{B}}_{0\text{ACC}}$ and $\hat{\mathbf{B}}_{0\text{GYR}}$ coefficients obtained by calibration are discarded, as they have no relation to the offsets that occur during flight. Modeling the results obtained by the calibration process implies reducing the scale factor and cross-couplings errors found on the inertial sensors specifications by an arbitrary amount that can be specified by the user. To summarize, instead of applying (85), (86) to the measurements obtained by (61), (63), the model directly employs (61), (63) with reduced \mathbf{M}_{GYR} and \mathbf{M}_{ACC} values.

Table 10. Results of calibration process.

Estimation	#	Coefficients
$\hat{\mathbf{M}}_{\text{ACC}}$	6	$\{\hat{s}_{\text{ACC},i}, \hat{\mathbf{m}}_{\text{ACC},ij}\}$
$\hat{\mathbf{M}}_{\text{GYR}}$	9	$\{\hat{s}_{\text{GYR},i}, \hat{\mathbf{m}}_{\text{GYR},ij}\}$
$\hat{\mathbf{B}}_{0\text{ACC}}$	3	$\mathbf{B}_{0\text{ACC}} \hat{\mathbf{N}}_{u0,\text{ACC},i}$
$\hat{\mathbf{B}}_{0\text{GYR}}$	3	$\mathbf{B}_{0\text{GYR}} \hat{\mathbf{N}}_{u0,\text{GYR},i}$

5.2. Determination of the Platform Frame Pose

The true relative pose between the body and platform frames (F_B, F_P), given by \mathbf{T}_{BP}^B and ϕ^{BP} , as well as their estimated values $\hat{\mathbf{T}}_{BP}^B$ and $\hat{\phi}^{BP}$, play a key role in the readings generated by the inertial sensors, as explained in Section 2.9.

Considering that the position of the aircraft center of mass is known (in both full and empty tank configurations), the true displacement \mathbf{T}_{BP}^B can be determined with near exactitude based on the IMU attachment point to the aircraft, resulting in very low $\sigma_{\hat{\mathbf{T}}_{BP}^B}$ values to be employed for the estimation of $\hat{\mathbf{T}}_{BP}^B$ in (52).

With regard to the attitude ϕ^{BP} , after mounting the IMU platform so two of its axes are approximately aligned with the forward and down directions of an approximate aircraft plane of symmetry (with no particular need for accuracy), it is possible to estimate the angular deviation ϕ^{BP} by means of self-alignment [4], resulting in small $\sigma_{\hat{\phi}^{BP}}$ values when estimating $\hat{\phi}^{BP}$ in (53).

5.3. Swinging or Magnetometer Calibration

Magnetometer calibration is inherently more complex than that of the inertial sensors, as it must be performed with the sensors already mounted on the aircraft, as otherwise, it would not capture the fixed contributions of the hard iron and soft iron magnetisms (Section 3.1). The calibration process, known as *swinging*, relies on obtaining magnetometer readings while the aircraft is positioned at different attitudes that encompass a wide array of heading, pitch, and roll values [4], and it is executed at a location where the magnetic field is precisely known.

The accuracy of the results is very dependent of the precision with which the different aircraft attitudes can be determined during swinging. This can be done with self-alignment procedures [4] or with the use of expensive static instruments. In any case, attitude accuracy is always going to be inferior to that obtained with a three-axis table during inertial sensor calibration. Once the magnetic field readings are obtained, they are compared to the real magnetic field values, and expression (67) is employed with least squares techniques to obtain estimations of the bias (sum of hard iron magnetism $\mathbf{B}_{\text{HI,MAG}}$ and offset $\mathbf{B}_{0,\text{MAG}}$), and the scale factor and cross-coupling matrix \mathbf{M}_{MAG} , which also includes the soft iron magnetism. The process can be repeated several times to isolate the influence of hard iron magnetism (a fixed effect that does not change) from the offset, which is a run-to-run error source that changes every time the magnetometers are turned on.

The fifteen coefficients estimated in the swinging process are listed in Table 11. Once the coefficients have been estimated, they can be introduced into the processor so it automatically performs the corrections shown in (87):

$$\hat{\mathbf{B}}^B = \hat{\mathbf{M}}_{\text{MAG}}^{-1} \tilde{\mathbf{B}}^B - \hat{\mathbf{B}}_{\text{HI,MAG}} - \hat{\mathbf{B}}_{0,\text{MAG}} \quad (87)$$

This article assumes that the bias offset $\mathbf{B}_{0,\text{MAG}}$ is exclusively a run-to-run source of error that varies every time the magnetometer is switched on, so bias offset coefficients obtained by swinging are discarded, as they have no relation to the offsets that occur during flight. Modeling the results obtained by swinging implies reducing the hard iron bias $\mathbf{B}_{\text{HI,MAG}}$ and scale factor and cross-coupling errors \mathbf{M}_{MAG} found on the sensor's specifications by an arbitrary amount that can be specified by the user. To summarize,

instead of applying (87) to the measurements obtained by (67), the model directly employs (67) with reduced $\mathbf{B}_{\text{HI,MAG}}$ and \mathbf{M}_{MAG} values.

Table 11. Results of swinging process.

Estimation	#	Coefficients
$\hat{\mathbf{M}}_{\text{MAG}}$	9	$\{\hat{s}_{\text{MAG},i}, \hat{m}_{\text{MAG},i}\}$
$\hat{\mathbf{B}}_{\text{HI,MAG}}$	3	$\hat{\mathbf{B}}_{\text{HI,MAG},i}$
$\hat{\mathbf{B}}_{0,\text{MAG}}$	3	$\mathbf{B}_{0,\text{MAG}}, \hat{\mathbf{N}}_{\text{u}0,\text{MAG},i}$

5.4. Determination of the Camera Frame Pose

The images generated by the onboard camera, and simulated by means of the Earth Viewer application introduced in Section 4.2, do not only depend on the relative pose between the body and the Earth but also on that of the camera with respect to the aircraft structure, which is represented by the rotation ϕ_{BC} and displacement $\mathbf{T}_{\text{BC}}^{\text{B}}$ generated when mounting the camera, as described in Section 4.1. Visual navigation algorithms, however, rely on the navigation system best estimate of this pose, this is, $\hat{\phi}^{\text{BC}}$ and $\hat{\mathbf{T}}_{\text{BC}}^{\text{B}}$, which need to be estimated once the already calibrated camera has been mounted on the aircraft. The two-phase process requires a chess board such as that employed for camera calibration [50,54].

The first phase uses an optimization procedure quite similar to that used in calibration to determine the relative pose between the camera frame F_{C} and the one rigidly attached to the chessboard. Instead of using the location of each chess box corner in different images, this process relies on a single photo and imposes that all chess boxes are square and have the same size, which is enough to obtain a solution up to an unknown scale. The size of the chess boxes provides the scale required to unambiguously solve the identification problem with high precision.

The second step is to obtain the pose between the chessboard and body frames. This is a straightforward geometric optimization problem that relies on distance measurements between chessboard points and aircraft structure points whose coordinates in the F_{B} frame are known. The resulting accuracy depends on the accuracy with which these distances can be measured, so special equipment may be required given the importance of the final estimations for the success of the visual navigation algorithms.

Overall, this is a robust and accurate process if properly executed, which results in the user-selectable $\sigma_{\hat{\mathbf{T}}_{\text{BC}}^{\text{B}}}$, $\sigma_{\hat{\phi}^{\text{BC}}}$ values employed for the stochastic estimation of $\hat{\mathbf{T}}_{\text{BC}}^{\text{B}}$ and $\hat{\phi}^{\text{BC}}$ in each run by means of (83) and (84).

6. Discussion: Realism, Stochastic Properties and Customizable Inputs

This article provides models of the various sensors usually installed onboard a fixed wing autonomous aircraft, and it includes a ready-to-use open source C++ implementation [6] so researchers can quickly generate realistic, stochastic (pseudo-random), and customizable time-stamped series of the outputs of the different sensors, including images of the Earth's surface that closely resemble what a real camera would record from the same positions and attitudes.

- In terms of realism, the detailed descriptions of Sections 2–4 show that in addition to the usually present white noise and random walk contributions, the models also include key error sources such as scale factors and cross-coupling effects, which are challenging for navigation systems, since these often include some type of linearization, and inputs that should theoretically be restricted to a single axis (caused by maneuvering or turbulence) in fact result in outputs to all three sensor axes. Supplied with a series of time-varying aircraft positions and attitudes, the Earth Viewer application generates detailed distortion-free images, resembling what a real onboard camera would record if flying the same trajectory.

In addition, and only for the case of the camera and inertial sensors, the models do not only consider the relative pose (position and attitude) between the platform (IMU)

and camera frames (in which the information is recorded) and the body frame (into which they are converted for output), as well as their slow variations with time as a consequence of fuel load changes, but also take into account the inaccuracies in the aircraft processors' knowledge of these relative poses.

- With respect to randomness, the random nature of the outputs generated by the various sensors is reflected in the extensive use of stochastic processes and distributions within the models. Section 6.1 below explains how all sensor errors are derived from two input seeds (one identifying the aircraft or fixed errors, and another specifying the flight, or run-to-run and in-run error contributions). This facilitates the use of the models within Monte Carlo simulations while also enabling the same set of outputs to be reproduced if so desired.
- Regarding the customization, the realism in the models relies on multiple input parameters that need to be provided by the end user. Most of them can be found in the data sheets provided by the manufacturers (usually with different names and conventions, which are explained in previous sections), others depend on the methods employed for the mounting of the sensors onboard the aircraft, and some of them can be improved by means of the calibration procedures described in Section 5. Section 6.2 describes the process followed to obtain the parameters for the case of a small low SWaP aircraft, which constitutes the default configuration for the model's C++ implementation [6].

6.1. Stochastic Models and the Use of Input Seeds

As explained in the corresponding sections, the outputs of the different sensor models depend on the input seeds listed in Tables 5–8 and grouped together in Table 12 for convenience.

Table 12. Sensor seeds.

Type		Error Sources	Seeds
Aircraft	i	fixed	$U_{i,A,ACC}$, $U_{i,A,GYR}$, $U_{i,A,MAG}$, $U_{i,A,PLAT}$, $U_{i,A,CAM}$
Flight	j	run-to-run and in-run	$U_{j,E,ACC}$, $U_{j,E,GYR}$, $U_{j,E,MAG}$, $U_{j,E,OSP}$, $U_{j,E,OAT}$ $U_{j,E,TAS}$, $U_{j,E,AOA}$, $U_{j,E,AOS}$, $U_{j,E,GNSS}$

The following steps describe how these seeds are obtained in the available C++ implementation of the models [6]:

1. Initialize a discrete uniform distribution with any seed (any value is valid, so 1 was employed by the authors), which produces random integers where each possible value has an equal likelihood of being produced. Call this distribution a number of times equal or higher than twice the maximum number of runs to be executed (each run provides the variation of time of all sensors for an unlimited amount of time and corresponds to a single aircraft flight), divide them into two groups of the same size, and store the results for later use. These values, called $Y_{i,A}$ and $Y_{j,F}$, are, respectively, the *aircraft seeds* and the *flight seeds*, where i is the aircraft number representing given fixed error realizations (fixed error contributions vary from aircraft to aircraft but are constant for all flights of that aircraft), and j is the trajectory number representing run-to-run and in-run error realizations (run-to-run and in-run error contributions vary from one flight of a given aircraft to the next). The stored aircraft and flight seeds become the initialization seeds for each of the executions or runs, so this step does not need to be repeated.
2. Every time the simulator needs to obtain the errors generated by the different sensors (which usually correspond to a given flight), it is initialized with a given aircraft seed $Y_{i,A}$ together with a flight seed $Y_{j,F}$. As these seeds are the only inputs required for all the stochastic processes within the sensors, the results of a given run can always be repeated by employing the same two seeds.

The selected aircraft and flight seeds are then employed to initialize two different discrete uniform distributions. One is executed five times to provide the *fixed sensor seeds* ($v_{i,A,ACC}$, $v_{i,A,GYR}$, $v_{i,A,MAG}$, $v_{i,A,PLAT}$, and $v_{i,A,CAM}$), while the other is realized nine times to obtain the *run sensor seeds* ($v_{j,F,ACC}$, $v_{j,F,GYR}$, $v_{j,F,MAG}$, $v_{j,F,OSP}$, $v_{j,F,EOAT}$, $v_{j,F,ETAS}$, $v_{j,F,EOA}$, $v_{j,F,EAOS}$, and $v_{j,F,GNSS}$). These seeds hence become the initialization seeds for each of the different sensors described throughout this article.

3. Each sensor relies on either one or two standard normal distributions $N(0, 1)$, depending on whether its error model is based exclusively on run-to-run and in-run error contributions or it also contains fixed error sources. The normal distributions of every sensor are initialized with the corresponding seeds ($v_{i,A,XXX}$ and $v_{j,F,XXX}$) for that sensor.
4. Upon initialization, the *fixed* normal distribution of every sensor is employed to generate all the values corresponding to scale factors, cross couplings, hard iron magnetism, and mounting errors. The *run* normal distribution in turn is employed to generate the required bias offsets.
5. Once the model has been initialized, it is able to estimate the errors generated by each sensor working at the required sensor rate. As time advances, every time a sensor is called to provide a measurement, its already initialized and used run normal distribution is called to generate the corresponding random walk increments and white noises.

6.2. Example: Input Parameters for a Low SWaP Fixed Wing Aircraft

This section describes the process followed by the authors to obtain the input parameters required by the various models (listed in Tables 5–8) for the specific case of a small low SWaP fixed wing autonomous aircraft. These values constitute the default configuration of the supplied C++ implementation [6], but they can be modified by the user to reflect different hardware, mounting, or calibration procedures.

With respect to the operating frequencies, Table 13 reflects the considered values, which are all within the working range of the specific sensors described in the following paragraphs.

Table 13. Example of frequencies of the different sensors.

Discrete Time	Frequency	Rate
$t_t = t \cdot \Delta t_{\text{TRUTH}}$	500 Hz	0.002 s
$t_s = s \cdot \Delta t_{\text{SENSED}}$	100 Hz	0.01 s
$t_i = i \cdot \Delta t_{\text{IMG}}$	10 Hz	0.1 s
$t_g = g \cdot \Delta t_{\text{GNSS}}$	1 Hz	1 s

The gyroscope values correspond to the MEMS gyroscopes present inside the Analog Devices ADIS16488A IMU [55]. Table 14 shows its performances, which have been taken from the data sheet when possible and corrected when suspicious. A calibration process such as that described in Section 5.1 is assumed to eliminate 95% of the scale factor and cross-coupling errors.

Table 14. Example for gyroscopes performance values.

GYR	Spec	Unit	Variable	Value	Calibration	Unit
In-Run Bias Stability (1σ)	5.10	$^\circ/\text{h}$	σ_{UGYR}	1.42×10^{-4}	1.42×10^{-4}	$^\circ/\text{s}^{1.5}$
Angle Random Walk (1σ)	0.26	$^\circ/\text{h}^{0.5}$	σ_{VGYR}	4.30×10^{-3}	4.30×10^{-3}	$^\circ/\text{s}^{0.5}$
Nonlinearity ¹	0.01	%	S_{GYR}	3.00×10^{-4}	1.50×10^{-5}	-
Misalignment	± 0.05	$^\circ$	m_{GYR}	8.70×10^{-4}	4.35×10^{-5}	-
Bias Repeatability (1σ)	± 0.2	$^\circ/\text{s}$	$B_{0\text{GYR}}$	2.00×10^{-1}	2.00×10^{-1}	$^\circ/\text{s}$

¹ The 0.01% scale factor error obtained in [55] is considered too optimistic and hence modified to $0.03\% = 3.00 \times 10^{-4}$.

Table 14 contains three columns of data. The left most column (“Spec”) corresponds to data taken directly from the sensor’s specifications, which become converted in the middle column (“Value”) to the parameters shown in Sections 2.2–2.9 (the conversion between bias instability and σ_u uses a period of 100 s, as noted in Section 2.4). The right column (“Calibration”) contains the values after the calibration process, which reduces the scale factor and cross-coupling errors by 95%. A similar process based on the explanations of Sections 2.3 and 2.4 should be followed by the user to obtain the parameters applicable to different gyroscopes, noting that a certain level of arbitrariness is required to account for the effects of calibration.

The accelerometer values also correspond to the MEMS accelerometers present inside the Analog Devices ADIS16488A IMU [55]. All values shown in Table 15 have been taken from the data sheet. As in the case of the gyroscopes, a calibration process as that described in Section 5.1 is assumed to eliminate 95% of the scale factor and cross-coupling errors (the conversion between bias instability and σ_u uses a period of 100 s, as noted in Section 2.4). As in the previous case, a similar process should be followed by the user to obtain the parameters applicable to different accelerometers and calibration procedures.

Table 15. Example for accelerometers’ performance values.

ACC	Spec	Unit	Variable	Value	Calibration	Unit
In-Run Bias Stability (1σ)	0.07	mg	σ_{uACC}	6.86×10^{-5}	6.86×10^{-5}	$m/s^{2.5}$
Velocity Random Walk (1σ)	0.029	$m/s/h^{0.5}$	σ_{vACC}	4.83×10^{-4}	4.83×10^{-4}	$m/s^{1.5}$
Nonlinearity	0.1	%	s_{ACC}	1.00×10^{-3}	5.00×10^{-5}	-
Misalignment	± 0.035	°	m_{ACC}	6.11×10^{-4}	3.05×10^{-5}	-
Bias Repeatability (1σ)	± 16	mg	B_{0ACC}	1.57×10^{-1}	1.57×10^{-1}	m/s^2

The magnetometer values are shown in Table 16, where the white noise has been taken from [4] and the rest of the parameters correspond to the magnetometers present in the Analog Devices ADIS16488A IMU [55]. Although the value of hard and soft iron magnetism in aircraft is rather small, the authors have not been able to obtain trusted values for them. To avoid eliminating sources of error, the authors have decided to increase by 50% the values for bias offset, scale factor, and cross-coupling errors found in the literature, as shown in the compensation column (“Comp.”). As both result in a similar effect, the authors expect that the realism of the results will not be adversely affected. In the case of the bias, the authors have assigned most of the error to the fixed hard iron error $B_{HI,MAG}$ and the remaining to the run-to-run bias offset $B_{0,MAG}$.

Table 16. Example for magnetometers’ performance values.

MAG	Spec	Unit	Variable	Value	Comp.	Swinging	Unit
Output Noise	5	$nT \cdot s^{0.5}$	$\sigma_{v,MAG}$	5.00×10^0	5.00×10^0	5.00×10^0	$nT \cdot s^{0.5}$
Nonlinearity	0.5	%	s_{MAG}	5.00×10^{-3}	7.50×10^{-3}	7.50×10^{-4}	-
Misalignment	± 0.35	°	m_{MAG}	6.11×10^{-3}	9.16×10^{-3}	9.16×10^{-4}	-
Bias (1σ)	± 1500	nT	$B_{HI,MAG}$	1.50×10^3	1.75×10^3	1.75×10^2	nT
Repeatability			$B_{0,MAG}$		5.00×10^2	5.00×10^2	nT

A swinging process such as that described in Section 5.3 is assumed to eliminate 90% of the fixed error contributions, this is, the hard iron magnetism, the scale factor, and the cross-coupling error (the soft iron effect is combined with the scale factor and cross-coupling errors). The final results can be found in the rightmost column above. The user should follow a similar process to obtain the parameters applicable to each specific case. Note however that because of the influence of the hard and soft iron effects, and the intrinsic difficulty of swinging, the determination of the parameters is more arbitrary than in the case of the inertial sensors.

With respect to the GNSS receiver, the horizontal position accuracy shown in Table 17 corresponds to the U-blox NEO-M8 receiver data sheet [56], where CEP stands for circular error probability. As CEP is equivalent to 1.18 standard deviations [57], it enables the obtainment of $\sigma_{\text{GNSS,POS,HOR}}$. As no vertical position accuracy is available within [56], a conservative value for $\sigma_{\text{GNSS,POS,VER}}$ of twice that of $\sigma_{\text{GNSS,POS,HOR}}$ has been selected. The velocity accuracy also originates at the U-blox NEO-M8 receiver data sheet [56]. Assuming that it corresponds to a per-axis error of ± 0.05 m/s instead of CEP, and knowing that the 50% mark of a normal distribution lies at 0.67448 standard deviations, it is possible to obtain $\sigma_{\text{GNSS,VEL}}$.

Table 17. Example of GNSS receiver performance values.

GNSS	Spec	Unit	Variable	Value	Unit
Horizontal position accuracy (CEP 50%)	2.50	m	$\sigma_{\text{GNSS,POS,HOR}}$	2.12×10^0	m
Vertical position accuracy (CEP 50%)	N/A		$\sigma_{\text{GNSS,POS,VER}}$	4.25×10^0	m
Ionospheric random walk 1/60 Hz	N/A		$\sigma_{\text{GNSS,ION}}$	1.60×10^{-1}	m
Ionospheric bias offset	N/A		$B_{0,\text{GNSS,ION}}$	8.00×10^0	m
Velocity accuracy (50%)	0.05	m/s	$\sigma_{\text{GNSS,VEL}}$	7.41×10^{-2}	m/s

In regards to the air data system, the σ_{OSP} value shown in Table 18 originates at the ± 10 m altitude error listed in the specifications of the Aeroprobe air data system [58], which translates into ± 100 Pa at a pressure altitude of 1500 m. The σ_{OAT} value is taken from the Analog Devices ADT7420 temperature sensor [59]. With respect to σ_{TAS} , the Aeroprobe air data system specifications [58] list a maximum airspeed error of 1 m/s, which can be interpreted as 3σ , and hence results in the σ_{TAS} value included in the table. The multi-hole Pitot tube contained in the Aeroprobe air data system [58] measures both flow angles (attack and sideslip) with a maximum error of $\pm 1.0^\circ$. If interpreted as 3σ , this results in standard deviations σ_{AOA} and σ_{AOS} of 0.33° . Although never present in the data sheets, the bias offsets for all variables present in the the Section 3.3 air data system model have been set equal to the system noises to obtain further realism in the results.

Table 18. Example of air data system performance values.

Air Data System	Spec	Unit	Variable	Value	Unit
Altitude Error	± 10	m	σ_{OSP}	1.00×10^2	Pa
			$B_{0\text{OSP}}$	1.00×10^2	Pa
Temperature Error (3σ)	± 0.15	K	σ_{OAT}	5.00×10^{-2}	K
			$B_{0\text{OAT}}$	5.00×10^{-2}	K
Airspeed Error (max)	1	m/s	σ_{TAS}	3.33×10^{-1}	m/s
			$B_{0\text{TAS}}$	3.33×10^{-1}	m/s
Flow Angle Error (max)	± 1.0	°	σ_{AOA}	3.33×10^{-1}	°
			$B_{0\text{AOA}}$	3.33×10^{-1}	°
Flow Angle Error (max)	± 1.0	°	σ_{AOS}	3.33×10^{-1}	°
			$B_{0\text{AOS}}$	3.33×10^{-1}	°

Table 19 suggests appropriate values for the IMU platform and camera pose estimation errors described in Sections 5.2 and 5.4, together with totally subjective ones for their true attitude with respect to the body frame.

Table 19. Example for IMU and camera mounting accuracy values.

Concept	Variable	Value	Unit	Variable	Value	Unit
True Yaw Error	σ_{ψ^P}	0.5	°	σ_{ψ^C}	0.1	°
True Pitch Error	σ_{θ^P}	2.0	°	σ_{θ^C}	0.1	°
True Bank Error	σ_{ξ^P}	0.1	°	σ_{ξ^C}	0.1	°
Position Estimation Error	$\sigma_{T_{BP}^B}$	0.01	m	$\sigma_{T_{BC}^B}$	0.002	m
Attitude Estimation Error	$\sigma_{\hat{\phi}^{BP}}$	0.03	°	$\sigma_{\hat{\phi}^{BC}}$	0.01	°

7. Conclusions

This article presents realistic, stochastic, and customizable models for the errors generated by the sensors typically installed onboard a fixed wing aircraft. These can be used to generate pseudo-random time-stamped series of values that simulate the sensor outputs during flight as well as a series of images of the Earth's surface that resemble what would be recorded by a camera mounted on the aircraft.

The article provides instructions and an example of how to obtain the parameters on which the models rely based exclusively on the information usually displayed in the data sheets provided by the sensors' manufacturers, so the user can employ the values that better resemble the performances of the specific equipment being modeled. The models properly represent the stochastic nature of the different random processes involved, while ensuring that the time-stamped series of outputs generated by each sensor can be repeated if so desired. The various sensor models include the contributions of the most important sources of error, and they are intended to be used as inputs to Monte Carlo simulations that rely on the sensor outputs, such as those required to evaluate inertial, visual, and visual-inertial navigation systems.

The authors release an open-source C++ implementation of the described models.

Author Contributions: Investigation, E.G.; Methodology, E.G.; Software, E.G.; Supervision, A.B.; Writing—original draft, E.G.; Writing—review and editing, A.B. All authors have read the manuscript and agree to its published version.

Funding: This research has been possible thanks to the financing of the RoboCity2030- DIH-CM, Madrid Robotics Digital Innovation Hub, S2018/NMT-4331, funded by "Programas de Actividades I+D en la Comunidad Madrid" and cofunded by Structural Funds of the EU.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: An open source C++ implementation of the described models can be found at [6]. Its execution generates pseudo-random time-stamped series of the errors introduced by the different sensors.

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

Abbreviations

The following abbreviations are used in this manuscript:

ACC	ACCelerometer
AOA	Angle of Attack
AOS	Angle of Sideslip
CAM	CAMera
CEP	Circular Error Probability
GLONASS	GLObal NAVigation Satellite System
GNC	Guidance, Navigation, and Control
GNSS	Global Navigation Satellite System
GPS	Global Positioning System

GYR	GYroscope
IMU	Inertial Measurement Unit
ISO	International Organization for Standardization
MAG	MAGnetometer
MEMS	Micromachined ElectroMechanical System
NED	North–East–Down
OAT	Outside Air Temperature
OSP	Outside Static Pressure
PSD	Power Spectral Density
SWaP	Size, Weight, and Power
TAS	True Air Speed

Appendix A. Motion of Multiple Rigid Bodies

The equations employed in this article make use of positions, velocities, and accelerations (both linear and angular) that refer to different reference systems or rigid bodies, which are in continuous motion (translation and rotation) among themselves. Their relationships are obtained in this appendix based on the three reference systems shown in Figure A1: an inertial reference system $F_0\{\mathbf{0}_0, \mathbf{i}_1^0, \mathbf{i}_2^0, \mathbf{i}_3^0\}$ and two non-inertial reference systems $F_1\{\mathbf{0}_1, \mathbf{i}_1^1, \mathbf{i}_2^1, \mathbf{i}_3^1\}$ and $F_2\{\mathbf{0}_2, \mathbf{i}_1^2, \mathbf{i}_2^2, \mathbf{i}_3^2\}$, where $(\mathbf{T}_{01}, \mathbf{v}_{01}, \mathbf{a}_{01})$ are the position, linear velocity, and linear acceleration of the origin of F_1 with respect to F_0 , $(\mathbf{T}_{02}, \mathbf{v}_{02}, \mathbf{a}_{02})$ are those of the origin of F_2 with respect to F_0 , and $(\mathbf{T}_{12}, \mathbf{v}_{12}, \mathbf{a}_{12})$ are those of the origin of F_2 with respect to F_1 . Similarly, $(\omega_{01}, \alpha_{01})$ are the angular velocity and angular acceleration of F_1 with respect to F_0 , $(\omega_{02}, \alpha_{02})$ are those of F_2 with respect to F_0 , and $(\omega_{12}, \alpha_{12})$ are those of F_2 with respect to F_1 . Let us also consider that $\mathbf{R}_{01}, \mathbf{R}_{02}$, and \mathbf{R}_{12} are the rotation matrices among the three different rigid bodies.

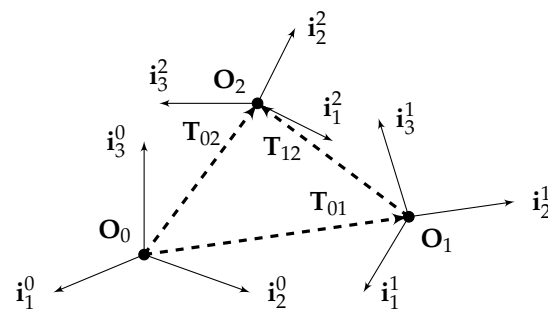


Figure A1. Reference system for combination of movements.

Appendix A.1. Composition of Position

The relationship between the linear position vectors \mathbf{T}_{02} , \mathbf{T}_{12} , and \mathbf{T}_{01} can be established by vectorial arithmetics when expressed in the same reference frame or by coordinate transformation [45] when not so:

$$\mathbf{T}_{02}^0 = \mathbf{T}_{12}^0 + \mathbf{T}_{01}^0 = \mathbf{R}_{01} \mathbf{T}_{12}^1 + \mathbf{T}_{01}^0 \quad (\text{A1})$$

Appendix A.2. Composition of Linear Velocity

The derivation with time of (A1) results in:

$$\dot{\mathbf{T}}_{02}^0 = \dot{\mathbf{R}}_{01} \mathbf{T}_{12}^1 + \mathbf{R}_{01} \dot{\mathbf{T}}_{12}^1 + \dot{\mathbf{T}}_{01}^0 \quad (\text{A2})$$

The use of the relationship between the rotation matrix and its time derivative [45] results in (note that the wide hat $\hat{\cdot}$ refers to the skew-symmetric form of a vector):

$$\dot{\mathbf{T}}_{02}^0 = \mathbf{R}_{01} \hat{\omega}_{01}^1 \mathbf{T}_{12}^1 + \mathbf{R}_{01} \dot{\mathbf{T}}_{12}^1 + \dot{\mathbf{T}}_{01}^0 \quad (\text{A3})$$

Reordering and replacing the position time derivatives with their respective velocities results in the relationship between the linear velocity vectors \mathbf{v}_{02} , \mathbf{v}_{12} , and \mathbf{v}_{01} expressed in the inertial frame F_0 :

$$\mathbf{v}_{02}^0 = \mathbf{R}_{01} \mathbf{v}_{12}^1 + \mathbf{v}_{01}^0 + \mathbf{R}_{01} \hat{\boldsymbol{\omega}}_{01}^1 \mathbf{T}_{12}^1 \quad (\text{A4})$$

$$\mathbf{v}_{02}^0 = \mathbf{v}_{12}^0 + \mathbf{v}_{01}^0 + \hat{\boldsymbol{\omega}}_{01}^0 \mathbf{T}_{12}^0 \quad (\text{A5})$$

Appendix A.3. Composition of Linear Acceleration

The derivation with time of (A4) results in:

$$\dot{\mathbf{v}}_{02}^0 = \dot{\mathbf{R}}_{01} \mathbf{v}_{12}^1 + \mathbf{R}_{01} \dot{\mathbf{v}}_{12}^1 + \dot{\mathbf{v}}_{01}^0 + \dot{\mathbf{R}}_{01} \hat{\boldsymbol{\omega}}_{01}^1 \mathbf{T}_{12}^1 + \mathbf{R}_{01} \dot{\hat{\boldsymbol{\omega}}}_{01}^1 \mathbf{T}_{12}^1 + \mathbf{R}_{01} \hat{\boldsymbol{\omega}}_{01}^1 \dot{\mathbf{T}}_{12}^1 \quad (\text{A6})$$

Replacing the rotation matrix time derivative results in:

$$\dot{\mathbf{v}}_{02}^0 = \mathbf{R}_{01} \hat{\boldsymbol{\omega}}_{01}^1 \mathbf{v}_{12}^1 + \mathbf{R}_{01} \dot{\mathbf{v}}_{12}^1 + \dot{\mathbf{v}}_{01}^0 + \mathbf{R}_{01} \hat{\boldsymbol{\omega}}_{01}^1 \hat{\boldsymbol{\omega}}_{01}^1 \mathbf{T}_{12}^1 + \mathbf{R}_{01} \dot{\hat{\boldsymbol{\omega}}}_{01}^1 \mathbf{T}_{12}^1 + \mathbf{R}_{01} \hat{\boldsymbol{\omega}}_{01}^1 \dot{\mathbf{T}}_{12}^1 \quad (\text{A7})$$

Reordering, replacing the position, linear velocity, and angular velocity time derivatives with their respective velocities, linear accelerations and angular accelerations, results in the relationship between the linear acceleration vectors \mathbf{a}_{02} , \mathbf{a}_{12} , and \mathbf{a}_{01} expressed in the inertial frame F_0 :

$$\mathbf{a}_{02}^0 = \mathbf{R}_{01} \mathbf{a}_{12}^1 + \left(\mathbf{a}_{01}^0 + \mathbf{R}_{01} \hat{\boldsymbol{\alpha}}_{01}^1 \mathbf{T}_{12}^1 + \mathbf{R}_{01} \hat{\boldsymbol{\omega}}_{01}^1 \hat{\boldsymbol{\omega}}_{01}^1 \mathbf{T}_{12}^1 \right) + 2 \mathbf{R}_{01} \hat{\boldsymbol{\omega}}_{01}^1 \mathbf{v}_{12}^1 \quad (\text{A8})$$

$$\mathbf{a}_{02}^0 = \mathbf{a}_{12}^0 + \left(\mathbf{a}_{01}^0 + \hat{\boldsymbol{\alpha}}_{01}^0 \mathbf{T}_{12}^0 + \hat{\boldsymbol{\omega}}_{01}^0 \hat{\boldsymbol{\omega}}_{01}^0 \mathbf{T}_{12}^0 \right) + 2 \hat{\boldsymbol{\omega}}_{01}^0 \mathbf{v}_{12}^0 \quad (\text{A9})$$

The term on the left-hand side is called absolute acceleration, while the three right-hand side terms are usually named relative, transport, and Coriolis accelerations, respectively.

Appendix A.4. Composition of Angular Velocity

The relationship among the different frames angular velocities is given by the rotation matrix composition rule [45], which can be derivated with respect to time:

$$\mathbf{R}_{02} = \mathbf{R}_{01} \mathbf{R}_{12} \rightarrow \dot{\mathbf{R}}_{02} = \dot{\mathbf{R}}_{01} \mathbf{R}_{12} + \mathbf{R}_{01} \dot{\mathbf{R}}_{12} \quad (\text{A10})$$

Replacing the rotation matrix time derivatives results in:

$$\mathbf{R}_{02} \hat{\boldsymbol{\omega}}_{02}^2 = \hat{\boldsymbol{\omega}}_{01}^0 \mathbf{R}_{01} \mathbf{R}_{12} + \mathbf{R}_{01} \mathbf{R}_{12} \hat{\boldsymbol{\omega}}_{12}^2 = \hat{\boldsymbol{\omega}}_{01}^0 \mathbf{R}_{02} + \mathbf{R}_{02} \hat{\boldsymbol{\omega}}_{12}^2 = \mathbf{R}_{02} \hat{\boldsymbol{\omega}}_{01}^1 + \mathbf{R}_{02} \hat{\boldsymbol{\omega}}_{12}^2 \quad (\text{A11})$$

The relationship among the angular velocity vectors $\boldsymbol{\omega}_{02}$, $\boldsymbol{\omega}_{12}$, and $\boldsymbol{\omega}_{01}$ is hence the following:

$$\boldsymbol{\omega}_{02}^0 = \boldsymbol{\omega}_{12}^0 + \boldsymbol{\omega}_{01}^0 = \mathbf{R}_{01} \boldsymbol{\omega}_{12}^1 + \boldsymbol{\omega}_{01}^0 \quad (\text{A12})$$

Appendix A.5. Composition of Angular Acceleration

The derivation with time of (A12) results in:

$$\dot{\boldsymbol{\omega}}_{02}^0 = \dot{\mathbf{R}}_{01} \boldsymbol{\omega}_{12}^1 + \mathbf{R}_{01} \dot{\boldsymbol{\omega}}_{12}^1 + \dot{\boldsymbol{\omega}}_{01}^0 \quad (\text{A13})$$

Replacing the rotation matrix time derivatives results in:

$$\dot{\boldsymbol{\omega}}_{02}^0 = \mathbf{R}_{01} \hat{\boldsymbol{\omega}}_{01}^1 \boldsymbol{\omega}_{12}^1 + \mathbf{R}_{01} \dot{\boldsymbol{\omega}}_{12}^1 + \dot{\boldsymbol{\omega}}_{01}^0 \quad (\text{A14})$$

Reordering, replacing the angular velocity time derivatives with their respective angular accelerations, results in the relationship between the angular acceleration vectors α_{02} , α_{12} , and α_{01} expressed in the inertial frame F_0 :

$$\alpha_{02}^0 = \mathbf{R}_{01} \alpha_{12}^1 + \alpha_{01}^0 + \mathbf{R}_{01} \hat{\omega}_{01}^1 \omega_{12}^1 \quad (\text{A15})$$

$$\alpha_{02}^0 = \alpha_{12}^0 + \alpha_{01}^0 + \hat{\omega}_{01}^0 \omega_{12}^0 \quad (\text{A16})$$

Appendix A.6. Summary of Compositions

The final expressions of the compositions above (A1), (A5), (A9), (A12), and (A16) are all expressed in the inertial frame F_0 , but they are also valid in any other frame as long as all its components are converted into that frame (note that it is not the same to compute a time derivative (velocity, acceleration, or angular acceleration) in the inertial frame and then convert it into a different frame than to directly compute the derivative in a non-inertial frame):

$$\mathbf{T}_{02} = \mathbf{T}_{12} + \mathbf{T}_{01} \quad (\text{A17})$$

$$\mathbf{v}_{02} = \mathbf{v}_{12} + \mathbf{v}_{01} + \hat{\omega}_{01} \mathbf{T}_{12} \quad (\text{A18})$$

$$\mathbf{a}_{02} = \mathbf{a}_{12} + (\mathbf{a}_{01} + \hat{\alpha}_{01} \mathbf{T}_{12} + \hat{\omega}_{01} \hat{\omega}_{01} \mathbf{T}_{12}) + 2 \hat{\omega}_{01} \mathbf{v}_{12} \quad (\text{A19})$$

$$\boldsymbol{\omega}_{02} = \boldsymbol{\omega}_{12} + \boldsymbol{\omega}_{01} \quad (\text{A20})$$

$$\alpha_{02} = \alpha_{12} + \alpha_{01} + \hat{\omega}_{01} \boldsymbol{\omega}_{12} \quad (\text{A21})$$

References

- Gallo, E. Stochastic High Fidelity Simulation and Scenarios for Testing of Fixed Wing Autonomous GNSS-Denied Navigation Algorithms. *arXiv* **2021**, arXiv:2102.00883v3. [CrossRef]
- Farrell, J.A. *Aided Navigation, GPS with High Rate Sensors*; McGraw-Hill: New York, NY, USA, 2008; ISBN 0-071-49329-8.
- Etkin, B. *Dynamics of Atmospheric Flight*; John Wiley & Sons: Hoboken, NJ, USA, 1972; ISBN 0-486-44522-4.
- Groves, P.D. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*; Artech House: Norwood, MA, USA, 2008; ISBN 978-1-58053-255-6.
- Farrell, J.A.; Oliveira e Silva, F.; Rahman, F.; Wendel, J. IMU Error Modeling for State Estimation and Sensor Calibration: A Tutorial. *IEEE Control. Syst. Mag.* **2021**, *accepted*.
- Gallo, E. High Fidelity Model of the Sensors and Camera onboard a Low SWaP Fixed Wing UAV. C++ Open Source Code. 2021. Available online: https://github.com/edugallogithub/sensor_camera_model (accessed on 17 July 2022).
- Standard 952; IEEE Standard Specification Format Guide and Test Procedure for Single-Axis Interferometric Fiber Optic Gyros. IEEE: New York, NY, USA, 2003.
- Standard 647; IEEE Standard Specification Format Guide and Test Procedure for Single-Axis Laser Gyros. IEEE: New York, NY, USA, 2006.
- Standard 1293; IEEE Standard Specification Format Guide and Test Procedure for Linear, Single-Axis, Non-Gyroscopic Accelerometers. IEEE: New York, NY, USA, 1999.
- Baziw, J.; Leondes, C.T. In Flight Alignment and Calibration of Inertial Measurement Units—Part I: General formulation. *IEEE Trans. Aerosp. Electron. Syst.* **1972**, *8*, 439–449. [CrossRef]
- Dierendonck, A.J.; McGraw, J.B.; Brown, R.G. Relationship between Allan Variances and Kalman Filter Parameters. In Proceedings of the 16th Annual Precise Time and Time Interval Systems and Applications Meeting, Greenbelt, MD, USA, 27–29 November 1984.
- Ford, J.J.; Evans, M.E. Online Estimation of Allan Variance Parameters. *J. Guid. Control. Dyn.* **2000**, *23*, 980–987. [CrossRef]
- El-Sheimy, N.; Hou, H.; Niu, X. Analysis and Modeling of Inertial Sensors using Allan Variance. *IEEE Trans. Instrum. Meas.* **2008**, *57*, 140–149. [CrossRef]
- Zhang, X.; Li, Y.; Mumford, P.; Rizos, C. Allan Variance Analysis on Error Characters of MEMS Inertial Sensors for an FPGA-Based GPS/INS System. In *Proceedings of the International Symposium on GPS/GNNS*; University of New South Wales: Sydney, NSW, Australia, 2008; pp. 127–133.
- Xing, Z.; Gebre-Egziabher, D. Modeling and Bounding Low Cost Inertial Sensor Errors. In Proceedings of the 2008 IEEE/ION Position, Location and Navigation Symposium, Monterey, CA, USA, 5–8 May 2008; pp. 1122–1132. [CrossRef]
- Saini, V.; Rana, S.C.; Kube, M.M. Online Estimation of State Space Error Model for MEMS IMU. *J. Model. Simul. Syst.* **2010**, *1*, 219–225.
- Vaccaro, V.; Zaki, A. Statistical Modeling of Rate Gyros and Accelerometers. *IEEE Trans. Instrum. Meas.* **2012**, *61*, 673–684. [CrossRef]
- Hidalgo-Carrio, J.; Arnold, S.; Poulakis, P. On the Design of Attitude-Heading Reference Systems using the Allan Variance. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control.* **2016**, *63*, 656–665. [CrossRef]

19. Silva, F.O.; Hemerly, E.M.; Leite Filho, W.C. On the Error State Selection for Stationary SINS Alignment and Calibration Kalman Filters - Part II: Observability/Estimability Analysis. *Sensors* **2017**, *17*, 439. [[CrossRef](#)]
20. Nikolic, J.; Furgale, P.; Melzer, A.; Siegwart, R. Maximum Likelihood Identification of Inertial Sensor Noise Model Parameters. *IEEE Sens. J.* **2016**, *16*, 163–176. [[CrossRef](#)]
21. Li, Y.; Ruizhi, C.; Niu, X.; Zhuang, Y.; Gao, Z.; Hu, X.; El-Sheimy, N. Inertial Sensing Meets Machine Learning: Opportunity or Challenge? *IEEE Trans. Intell. Transp. Syst.* **2021**, 4712–4719. [[CrossRef](#)]
22. Allan, D.W. Statistics of Atomic Frequency Standards. *Proc. IEEE* **1966**, *54*, 221–230. [[CrossRef](#)]
23. Riley, W.J. *Handbook of Frequency Stability Analysis*; National Institute of Standards and Technology, U.S. Department of Commerce: Gaithersburg, MD, USA, 2008.
24. Barnes, J.A.; Chi, A.R.; Cutler, L.S.; Healey, D.J.; Leeson, D.B.; McGunigal, T.E.; Mullen, J.A., Jr.; Smith, W.L.; Sydnor, R.L.; Vessot, R.F.C.; et al. Characterization of Frequency Stability. *IEEE Trans. Instrum. Meas.* **1971**, *IM-20*, 105–120. [[CrossRef](#)]
25. Stebler, Y.; Guerrier, S.; Skaloud, J.; Victoria-Feser, M.P. A Framework for Inertial Sensor Calibration using Complex Stochastic Error Models. In Proceedings of the 2012 IEEE/ION Position, Location and Navigation Symposium, Myrtle Beach, SC, USA, 23–26 April 2012. [[CrossRef](#)]
26. Quinchia, A.G.; Falco, G.; Falletti, E.; Dovi, F.; Ferrer, C. A Comparison between Different Error Modeling of MEMS applied to GPS/INS Integrated Systems. *Sensors* **2013**, *13*, 9549–9588. [[CrossRef](#)]
27. Nassar, S.; Schwarz, K.P.; El-Sheimy, N.; Noureldin, A. Modeling Inertial Sensor Errors Using Autoregressive Models. *Navig. J. Inst. Navig.* **2014**, *51*, 259–268. [[CrossRef](#)]
28. Miao, Z.; Shen, F.; Xu, D.; He, K.; Tian, C. Online Estimation of Allan Variance Coefficients based on a Neural Extended Kalman Filter. *Sensors* **2015**, *15*, 2496–2524. [[CrossRef](#)]
29. Rudyk, A.V.; Semenov, A.O.; Kryvinska, N.; Semenova, O.O.; Kvasnikov, V.P.; Safonyk, A.P. Strapdown Inertial Navigation Systems for Positioning Mobile Robots—MEMS Gyroscopes Random Errors Analysis Using Allan Variance Method. *Sensors* **2020**, *20*, 4841. [[CrossRef](#)]
30. Titterton, D.; Weston, J. *Strapdown Inertial Navigation Technology*, 2nd ed.; The Institution of Engineering and Technology: Stevenage, UK, 2004. ISBN 0-863-41358-7.
31. Chatfield, A.B. *Fundamentals of High Accuracy Inertial Navigation*; Progress in Astronautics and Aeronautics; AIAA: Reston, VA, USA, 1997. ISBN 1-56347-243-0.
32. Rogers, R.M. *Applied Mathematics in Integrated Navigation Systems*, 3rd ed.; AIAA: Reston, VA, USA, 2007; ISBN 1-563-47927-3.
33. Kayton, M.; Fried, W.R. *Avionics Navigation Systems*, 2nd ed.; John Wiley & Sons: Hoboken, NJ, USA, 1997; ISBN 0-471-54795-6.
34. Crassidis, J.L. Sigma-Point Kalman Filtering for Integrated GPS and Inertial Navigation. *IEEE Trans. Aerosp. Electron. Syst.* **2006**, *42*, 750–756. [[CrossRef](#)]
35. Woodman, O.J. *An Introduction to Inertial Navigation*; Technical Report; University of Cambridge Computer Laboratory: Cambridge, UK, 2007.
36. Hibbeler, R.C. *Engineering Mechanics: Statics and Dynamics*, 4th ed.; Pearson: Upper Saddle River, NJ, USA, 2015; ISBN 978-0-13391-542-6.
37. Grewal, M.; Andrews, A. How Good Is Your Gyro? *IEEE Control. Syst. Mag.* **2010**, *30*, 12–86. [[CrossRef](#)]
38. Trusov, A.A. *Allan Variance Analysis of Random Noise Modes in Gyroscopes*; Technical Report; University of California: Berkeley, CA, USA, 2011.
39. Guide to Comparing Gyro and IMU Technologies: Micro-Electro-Mechanical Systems and Fiber Optic Gyros. KVH Fiber Optic Gyro. 2014. Available online: <https://caclase.co.uk/wp-content/uploads/2016/11/Guide-to-Comparing-Gyros-0914.pdf> (accessed on 17 July 2022).
40. Chow, R. *Evaluating Inertial Measurement Units*; Technical Report; Epsom Electronics America: Los Alamitos, CA, USA, 2011.
41. Stockwell, W. *Angle Random Walk*; Technical Report; Crossbow Technology Inc.: San Jose, CA, USA, 2003.
42. Renaut, F. *MEMS Inertial Sensors Technology*; Technical Report; Swiss Federal Institute of Technology: Zurich, Switzerland, 2013.
43. Farrenkopf, R.L. Analytic Steady-State Accuracy Solutions for Two Common Spacecraft Attitude Estimators. *J. Guid. Control.* **1974**, *1*, 282–284. [[CrossRef](#)]
44. Frishman, F. *On the Arithmetic Means and Variances of Products and Ratios of Random Variables*; Technical Report; Army Research Office: Adelphi, MD, USA, 1971.
45. Shuster, M.D. A Survey of Attitude Representations. *J. Astronaut. Sci.* **1993**, *41*, 439–517.
46. Pinter, C.C. *A Book of Abstract Algebra*, 2nd ed.; Dover Publications: New York, NY, USA, 1990; ISBN 0-486-47417-8.
47. *Manual of the ICAO International Standard Atmosphere*, 3rd ed.; Technical Report; ICAO DOC-7488/3; International Civil Aviation Organization: Montreal, QC, Canada, 2000.
48. Gallo, E. Quasi Static Atmospheric Model for Aircraft Trajectory Prediction and Flight Simulation. *arXiv* **2021**, arXiv:2101.10744v1. [[CrossRef](#)]
49. Eshelby, M.E. *Aircraft Performance, Theory and Practice*; Arnold: London, UK, 2000; ISBN 0-340-75897-X.
50. Ma, Y.; Soatto, S.; Kosecka, J.; Sastry, S.S. *An Invitation to 3-D Vision, From Images to Geometric Models*; Springer: Berlin/Heidelberg, Germany, 2001; ISBN 978-1-4419-1846-8.
51. Open Scene Graph. Available online: <http://openscenegraph.org> (accessed on 17 July 2022).
52. osgEarth. Available online: <http://osgearth.org> (accessed on 17 July 2022).

53. Tedaldi, D.; Pretto, A.; Menegatti, E. A Robust and Easy to Implement Method for IMU Calibration without External Equipments Required. *IEEE Int. Conf. Robot. Autom.* **2014**, 3042–3049. [[CrossRef](#)]
54. Kaehler, A.; Bradski, G. *Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2016; ISBN 978-1-491-93799-0.
55. Analog Devices ADIS16488A Ten Degrees of Freedom Inertial Sensor Data Sheet. 2015. Available online: <http://www.analog.com/media/en/technical-documentation/data-sheets/ADIS16488A.pdf> (accessed on 17 July 2022).
56. U-Blox NEO-M8 Data Sheet. Available online: <https://www.u-blox.com/en/product/neo-m8-series> (accessed on 17 July 2022).
57. GPS Position Accuracy Measures. 2003. Available online: http://www.gisresources.com/wp-content/uploads/2014/03/gps_book.pdf (accessed on 17 July 2022).
58. Aeroprobe Corporation Air Data Systems Data Sheet. Available online: www.aeroprobe.com (accessed on 17 July 2022).
59. Analog Devices ADT7420 Temperature Sensor. 2012. Available online: <http://www.analog.com/media/en/technical-documentation/data-sheets/ADT7420.pdf> (accessed on 17 July 2022).