

Mathematical and Computer Modelling of Dynamical Systems: Methods, Tools and Applications in Engineering and Related Sciences

Publication details, including instructions for authors and subscription information:

<http://www.tandfonline.com/loi/nmcm20>

An approach to benchmarking of loosely coupled low-cost navigation systems

R. González^a, J.I. Giribet^b & H.D. Patiño^c

^a GridTICs and Laboratorio de Computación Reconfigurable, Facultad Regional Mendoza, Universidad Tecnológica Nacional, Rodriguez 273, M5502AJE Mendoza, Argentina

^b Departamento de Electrónica, Universidad de Buenos Aires and CONICET, Av. Paseo Colón 850, C1063ACV Ciudad Autónoma de Buenos Aires, Argentina

^c Instituto de Automática, Universidad Nacional de San Juan, Av. San Martín Oeste 1112, J5400ARL San Juan, Argentina

Published online: 02 Sep 2014.



[Click for updates](#)

To cite this article: R. González, J.I. Giribet & H.D. Patiño (2014): An approach to benchmarking of loosely coupled low-cost navigation systems, Mathematical and Computer Modelling of Dynamical Systems: Methods, Tools and Applications in Engineering and Related Sciences, DOI: [10.1080/13873954.2014.952642](https://doi.org/10.1080/13873954.2014.952642)

To link to this article: <http://dx.doi.org/10.1080/13873954.2014.952642>

PLEASE SCROLL DOWN FOR ARTICLE

Taylor & Francis makes every effort to ensure the accuracy of all the information (the "Content") contained in the publications on our platform. However, Taylor & Francis, our agents, and our licensors make no representations or warranties whatsoever as to the accuracy, completeness, or suitability for any purpose of the Content. Any opinions and views expressed in this publication are the opinions and views of the authors, and are not the views of or endorsed by Taylor & Francis. The accuracy of the Content should not be relied upon and should be independently verified with primary sources of information. Taylor and Francis shall not be liable for any losses, actions, claims, proceedings, demands, costs, expenses, damages, and other liabilities whatsoever or

howsoever caused arising directly or indirectly in connection with, in relation to or arising out of the use of the Content.

This article may be used for research, teaching, and private study purposes. Any substantial or systematic reproduction, redistribution, reselling, loan, sub-licensing, systematic supply, or distribution in any form to anyone is expressly forbidden. Terms & Conditions of access and use can be found at <http://www.tandfonline.com/page/terms-and-conditions>

An approach to benchmarking of loosely coupled low-cost navigation systems

R. González^{a*}, J.I. Giribet^b and H.D. Patiño^c

^a*GridTICs and Laboratorio de Computación Reconfigurable, Facultad Regional Mendoza, Universidad Tecnológica Nacional, Rodriguez 273, M5502AJE Mendoza, Argentina;*

^b*Departamento de Electrónica, Universidad de Buenos Aires and CONICET, Av. Paseo Colón 850, C1063ACV Ciudad Autónoma de Buenos Aires, Argentina; 'Instituto de Automática, Universidad Nacional de San Juan, Av. San Martín Oeste 1112, J5400ARL San Juan, Argentina*

(Received 9 October 2013; accepted 5 August 2014)

New solutions to the navigation problem related to low-cost integrated navigation systems (INS) are often published. Since these new solutions are generally compared with ad hoc mathematical models that are not fully exposed, one cannot be sure of the relative improvements. In this work, complete mathematical model for a low-cost INS is suggested to be used as a benchmarking. As far as the authors' knowledge, a benchmarking for low-cost INS has not been previously reported. Shown INS comprises a strapdown inertial navigation system, loosely coupled to a GPS receiver. The INS mathematical model is based upon classical navigation equations and classical sensor models, both from recognized authors. The algorithm that details the INS operation is also presented. The benchmarking is provided as an open-source toolbox for MATLAB. Additionally, this work can be taken as a starting point for new practitioners in the INS field. To validate the INS mathematical model, real-world data sets from three different Micro Electro-Mechanical Systems (MEMS) inertial measurement units (IMU) and a GPS receiver are processed. It is observed that obtained RMS errors from the three INS are coherent with the quality of corresponding MEMS IMU. This confirms that the proposed benchmarking is a suitable tool to evaluate objectively new solutions to low-cost INS.

Keywords: navigation; system modelling; inertial sensors; signal and information processing; aerospace systems; robotic systems

1. Introduction

An integrated navigation system (INS) is an electronic device mainly composed of an onboard computer, an inertial measurement unit (IMU), which contains accelerometers and gyroscopes, in addition to aiding sensors such as a GPS receiver, a barometer, and/or magnetometers. It provides estimates of position and velocity, as well as orientation parameters for a vehicle. Recently, with advances in Micro Electro-Mechanical Systems (MEMS) technology, commercial off-the-shelf MEMS-based inertial sensors are available. INS using MEMS IMU are a subject of great interest today due to their particular characteristics, such as light weight, low power consumption, and low cost, which make them suitable for several applications, e.g., navigation of aerial and ground unmanned micro-robots.

*Corresponding author. Email: rodraz@frm.utn.edu.ar

Despite of navigation equations from recognized authors [1–4] may lead to similar results for a particular reference frame (ECI, ECEF, local), more information is needed to fully describe an INS. For instance, different integration architectures (tight vs. loosely coupled) or diverse sensors error models will drastically produce unequal outcomes. Moreover, an open-loop correction architecture, as opposite to closed-loop, is impractical for loosely coupled INS ([3], Section 12.1.1). Hence, a complete mathematical model of the navigation system against which a new solution is compared must be provided in order to replicate shown results.

New approaches to the navigation problem related to low-cost INS are often proposed in the literature, for instance, modifications to nonlinear filters to estimate system states or novel integration strategies. Generally, these new solutions are compared with ad hoc mathematical models which are not fully shown, commonly by space restrictions (e.g. see [5,6]). In other works, navigation data are processed using proprietary software that usually does not reveal mathematical models due to copyright issues (e.g. see [7]).

A completely defined INS mathematical model can be a useful tool to contrast new solutions for low-cost INS, in order to obtain fair numerical comparisons. The use of a well-known benchmarking would lead to get unbiased conclusions about the advantages of new navigation approaches with respect to former solutions. To the best of authors' knowledge, a benchmarking for low-cost INS has not been previously reported in the literature.

In this article, an approach to benchmarking for low-cost INS is presented. It consists of a strapdown inertial navigation system (SINS), loosely coupled to a GPS receiver, and an extended Kalman filter (EKF) to accomplish the fusion of sensors. The main INS algorithm that exposes the sequence in which equations must be solved is also provided. It tries to clarify some implementation aspects. The proposed mathematical model is based upon classical navigation equations and classical sensor models. The benchmarking is given as an open-source toolbox for MATLAB, *de facto* standard programming language for simulation and mathematical computing. It can be downloaded from [8]. The authors believe that this is the first work that suggests an approach to benchmarking for low-cost INS.

Although the primary contribution of this work is to supply a benchmarking for low-cost INS, it might be also beneficial for new practitioners in the INS area. Since it summarizes INS formulas in the discrete domain, it can be used to understand better a typical loosely coupled INS. Conventional literature on this subject gives complete navigation solutions for different frames projections. Generally, this detailed amount of information can be overwhelming for an INS newcomer who may not have a solid background on navigation basis, stochastic processes and control theory. Furthermore, classical literature does not explain practical INS issues, e.g., which clock must be considered as the master clock of the system, GPS clock or SINS clock.

Suggested benchmarking is validated by applying a practical procedure. Field data were collected from three MEMS IMU and a GPS receiver. Reference values are obtained by fusing navigation-grade IMU and DGPS measurements. Three INS are set up and processed using the benchmarking. Lastly, systems performances are analysed.

The rest of this article is organized as follows. [Section 2](#) shows mathematical models of sensors. [Section 3](#) exhibits the mathematical model of a loosely coupled INS, including EKF equations. [Section 4](#) shows the INS algorithm and comments some developmental issues. In [Section 5](#), method to validate the benchmarking is explained. Finally, in [Section 6](#) concluding remarks are commented.

2. Sensors mathematical models

This section exposes the mathematical models of inertial sensors and GPS.

All the equations presented in this work are in the discrete-time domain, except otherwise stated. Being the current interval sample t_k , variables with subscript $(-)$ correspond to the former interval sample, $t_{(k-1)}$. Besides, variables with subscript $(+)$ belong to the later interval sample, $t_{(k+1)}$.

Noisy measurements have as superscript a tilde (\sim). Estimated variables that depend on noisy measurements have as superscript a hat ($\hat{\cdot}$). A skew symmetric matrix operator is defined as $S_{\{3 \times 3\}} = [\mathbf{r} \times]$ where $\mathbf{r}_{\{3 \times 1\}}$ is a vector ([4], Equation B.15). Lastly, the symbol ‘o’ means entrywise product.

2.1. Gyroscopes

Outputs of orthogonal triaxial gyroscopes are modelled as,

$$\tilde{\boldsymbol{\omega}}^b = \boldsymbol{\omega}_{ib}^b + \boldsymbol{\eta}_g + \mathbf{b}_g + \boldsymbol{\eta}_{g\delta b}, \quad (1)$$

where $\boldsymbol{\omega}_{ib}^b$ represents the ideal gyroscopes outputs ([2], Equation 3.29), and the rest of terms represent different types of errors.

A gyroscope has several sources of errors as angle random walk noise (ARW), rate random walk noise, rate ramp noise, biases, cross-coupling errors, scale factor errors, and axes misalignments, among others. The proposed benchmarking takes only three errors into account which are considered to be dominant in MEMS gyroscopes: ARW and static and dynamic biases. ARW can be expressed as Gaussian white-noise sequence $\boldsymbol{\eta}_g \sim N(\mathbf{0}, \boldsymbol{\sigma}_g^2)$, where $\boldsymbol{\sigma}_g^2 = [\sigma_{gx}^2, \sigma_{gy}^2, \sigma_{gz}^2]^T$. Usually, gyroscope manufacturers deliver ARW in root-PSD units that must be converted to International System (SI) units. If ARW is $N\left(\frac{\text{deg}}{\sqrt{\text{h}}}\right)$, then ([4], Section 4.6.3.2),

$$\mathbf{n}_g = \frac{N}{60} \frac{\pi}{180} \left(\frac{\text{rad/s}}{\sqrt{\text{Hz}}} \right). \quad (2)$$

Static bias \mathbf{b}_g varies every time the gyroscope is turned on but stays constant throughout operation ([3], Section 4.4.1), hence it is modelled as a random constant process ([4], Section 4.6.3.1). On the other hand, dynamic bias $\delta\mathbf{b}_g$ is usually observed at low frequencies and is associated with correlation time τ_g and variance $\sigma_{g\delta b}^2$ ([3], Section 4.4.1). Thus, it is modelled as a scalar Gauss-Markov process ([4], Section 4.6.3.3). Values from τ_g and $\sigma_{g\delta b}^2$ define the discrete sequence $\boldsymbol{\eta}_{g\delta b}$ [9].

2.2. Accelerometers

The proposed error model of accelerometers is similar to that of gyroscopes,

$$\tilde{\mathbf{f}}^b = \mathbf{f}^b + \boldsymbol{\eta}_f + \mathbf{b}_f + \boldsymbol{\eta}_{f\delta b}, \quad (3)$$

where accelerometer’s random noise is given as velocity random walk (VRW) as $\boldsymbol{\eta}_f \sim N(\mathbf{0}, \boldsymbol{\sigma}_f^2)$ with variance $\boldsymbol{\sigma}_f^2 = [\sigma_{fx}^2, \sigma_{fy}^2, \sigma_{fz}^2]^T$. \mathbf{b}_f denotes the static bias and $\boldsymbol{\eta}_{f\delta b}$ is a

discrete sequence related to dynamic bias noise $\delta \mathbf{b}_f$, which is modelled as a scalar Gauss-Markov process with correlation time τ_f and variance $\sigma_{f\delta b}^2$.

As with gyroscopes, VRW is commonly provided in units that must be changed to SI units. Thus, if VRW is $N\left(\frac{\text{m/s}}{\sqrt{\text{h}}}\right)$, then,

$$\mathbf{n}_f = \frac{N}{60} \left(\frac{\text{m/s}^2}{\sqrt{\text{Hz}}} \right). \quad (4)$$

2.3. GPS receiver

2.3.1. GPS position

A low-cost GPS usually delivers estimates of position only in the navigation frame (n-frame) as $\tilde{\mathbf{p}}_G^n = [\tilde{\gamma}_G, \tilde{\lambda}_G, \tilde{h}_G]^T$, where $\tilde{\gamma}_G$ denotes geodetic latitude, $\tilde{\lambda}_G$ denotes geodetic longitude, both in radians, and \tilde{h}_G denotes geodetic altitude in meters above the reference ellipsoid ([4], Section 2.3.2.1).

A GPS receiver provides position with certain dispersion. Thus, GPS position is considered to be added with three Gaussian white noises sequences, viz, $\eta_\gamma \sim N(0, \sigma_\gamma^2)$, $\eta_\lambda \sim N(0, \sigma_\lambda^2)$ and $\eta_h \sim N(0, \sigma_h^2)$ as

$$\tilde{\gamma}_G = \gamma + \eta_\gamma \quad (5a)$$

$$\tilde{\lambda}_G = \lambda + \eta_\lambda \quad (5b)$$

$$\tilde{h}_G = h + \eta_h. \quad (5c)$$

In general, GPS manufacturers give horizontal position (2D) accuracy expressed as circular error probable (CEP) in meters, which is the radius of the circle centred on the correct location that contains 50% of the expected horizontal position errors ([4], Section 4.9.1.2). Considering that latitude and longitude standard deviations, $\sigma_{\gamma m}$ and $\sigma_{\lambda m}$ respectively, are equal, then ([4], Table 4.2)

$$\sigma_{\gamma m} = \sigma_{\lambda m} = 0.8493 \cdot \text{CEP (m)}. \quad (6)$$

GPS standard deviations are also needed in units of radians and can be obtained as follows,

$$\sigma_\gamma = \frac{\sigma_{\gamma m}}{R_M(\gamma) + h} \quad (7a)$$

$$\sigma_\lambda = \frac{\sigma_{\lambda m}}{(R_N(\gamma) + h) \cos \gamma}, \quad (7b)$$

where R_M and R_N are given by Equation (11a) and (11b).

2.3.2. GPS velocity

Additionally, most low-cost GPS receivers provide estimates of velocity in the vehicle-carried North-East-Down (NED) frame ([10], Section 2.2.4).

GPS velocity vector $\tilde{\mathbf{v}}_G^u = [\tilde{v}_{N_G}, \tilde{v}_{E_G}, \tilde{v}_{D_G}]^T$ is modelled as true NED velocity $\mathbf{v}^u = [v_N, v_E, v_D]^T$ plus a Gaussian white noise sequence $\boldsymbol{\eta}_v \sim \mathbf{N}(\mathbf{0}, \boldsymbol{\sigma}_v^2)$ as

$$\tilde{\mathbf{v}}_G = \mathbf{v}^n + \boldsymbol{\eta}_{\mathbf{v}}, \quad (8)$$

where σ_v^2 is given by GPS manufacturer.

3. Integrated navigation mathematical model

This section presents the mathematical model of a loosely coupled strapdown inertial navigation system, in addition to EKF equations to fusion sensors information.

3.1. Strapdown inertial navigation system

A strapdown inertial navigation system (SINS) is an electronic device rigidly mounted on a vehicle, compound of a microcomputer and inertial sensors (accelerometers and gyroscopes), whose axes are aligned with respect to the vehicle body frame (b-frame), to calculate continuously estimates of orientation, position and velocity. Figure 1 shows a diagram of a SINS.

Table 1 shows some important constants from the World Geodetic System 1984 (WGS84) Earth model to be used next.

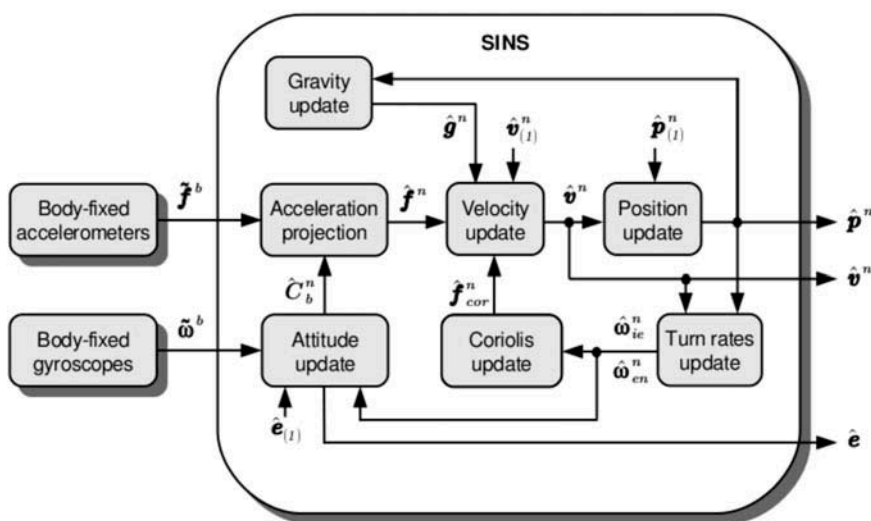


Figure 1. Diagram of a strapdown inertial navigation system.

Table 1. Constants from WGS84 Earth model.

Name	Symbol	Value	
Semi-major axis	a	6378137.0	m
Eccentricity	e	0.0818191908426	
Earth angular rate	Ω	7.292115E-5	rad/s

3.1.1. Turn rates

The turn rate of the Earth expressed in the navigation frame (n-frame) is ([2], Equation 3.72)

$$\hat{\omega}_{ie}^n = [\Omega \cos \hat{\gamma}_{(-)}, 0, -\Omega \sin \hat{\gamma}_{(-)}]^T. \quad (9)$$

The turn rate of the n-frame respect to the Earth expressed in the n-frame, known as the transport rate, is ([3], Equation 5.37)

$$\hat{\omega}_{en}^n = \left[\frac{\hat{v}_{E(-)}}{\hat{R}_{N(-)} + \hat{h}_{(-)}}, \frac{-\hat{v}_{N(-)}}{\hat{R}_{M(-)} + \hat{h}_{(-)}}, \frac{-\hat{v}_{E(-)} \tan \hat{\gamma}_{(-)}}{\hat{R}_{N(-)} + \hat{h}_{(-)}} \right]^T, \quad (10)$$

where R_M is the meridian radius of curvature ([4], Equation 2.6) and R_N is the normal radius of curvature ([4], Equation 2.7)

$$\hat{R}_M = \frac{a(1 - e^2)}{(1 - e^2 \sin^2 \hat{\gamma})^{3/2}} \quad (11a)$$

$$\hat{R}_N = \frac{a}{(1 - e^2 \sin^2 \hat{\gamma})^{1/2}}. \quad (11b)$$

3.1.2. Attitude

Quaternion attitude representation is selected to propagate changes in body orientation. A quaternion vector is defined as,

$$\mathbf{q} = [\boldsymbol{\varrho}, q_4]^T = [q_1, q_2, q_3, q_4]^T, \quad (12)$$

where q_1, q_2, q_3 , and q_4 are real numbers. A quaternion has a vector part, $\boldsymbol{\varrho} = [q_1, q_2, q_3]^T$, $[q_1, q_2, q_3]^T$, and a scalar part, q_4 , and satisfies the constraint given by $\mathbf{q}^T \mathbf{q} = 1$.

Equations to update $\hat{\mathbf{q}}$ are ([11], Equation 7.39)

$$\begin{aligned} \hat{\omega}_c^b &= \tilde{\omega}^b - \hat{C}_{n(-)}^b (\hat{\omega}_{ie}^n + \hat{\omega}_{en}^n) + \hat{\mathbf{b}}_{g(-)} + \delta \hat{\mathbf{b}}_{g(-)} \\ &= [\hat{\omega}_{cx}^b, \hat{\omega}_{cy}^b, \hat{\omega}_{cz}^b] \end{aligned} \quad (13a)$$

$$m = \|\hat{\omega}_c^b\| \quad (13b)$$

$$c_1 = \frac{\hat{\omega}_{cx}^b}{m} \sin(0.5m\delta t_S) \quad (13c)$$

$$c_2 = \frac{\hat{\omega}_{cy}^b}{m} \sin(0.5m\delta t_S) \quad (13d)$$

$$c_3 = \frac{\hat{\omega}_{cz}^b}{m} \sin(0.5m\delta t_S) \quad (13e)$$

$$c_4 = \cos(0.5m\delta t_S) \quad (13f)$$

$$\hat{\mathbf{q}} = \begin{bmatrix} c_4 & c_3 & -c_2 & c_1 \\ -c_3 & c_4 & c_1 & c_2 \\ c_2 & -c_1 & c_4 & c_3 \\ -c_1 & -c_2 & -c_3 & c_4 \end{bmatrix} \hat{\mathbf{q}}_{(-)} \quad (13g)$$

$$\hat{\mathbf{q}} = \frac{\hat{\mathbf{q}}}{\|\hat{\mathbf{q}}\|}. \quad (13h)$$

Vectors $\hat{\mathbf{b}}_g$ and $\delta\hat{\mathbf{b}}_g$ are provided by the EKF (Equation (21b)). The time between inertial measurements is δt_S . Updated $\hat{\mathbf{q}}$ is a unit vector to within first order. Commonly, a brute-force normalization should be done to insure $\mathbf{q}^T \mathbf{q} = 1$ (Equation (13h)). Initial quaternion $\hat{\mathbf{q}}_{(1)}$ can be obtained from Euler angles ([2], Equation 3.65).

A direction cosine matrix $\hat{C}_{b\{3 \times 3\}}^n$ is needed for projecting measurements from the b-frame to the n-frame. Elements of \hat{C}_b^n are updated using quaternion as follows ([2], Equation 3.63):

$$\hat{C}_{b(1,1)}^n = \hat{q}_1^2 - \hat{q}_2^2 - \hat{q}_3^2 + \hat{q}_4^2 \quad (14a)$$

$$\hat{C}_{b(1,2)}^n = 2(\hat{q}_1\hat{q}_2 - \hat{q}_3\hat{q}_4) \quad (14b)$$

$$\hat{C}_{b(1,3)}^n = 2(\hat{q}_1\hat{q}_3 + \hat{q}_2\hat{q}_4) \quad (14c)$$

$$\hat{C}_{b(2,1)}^n = 2(\hat{q}_1\hat{q}_2 + \hat{q}_3\hat{q}_4) \quad (14d)$$

$$\hat{C}_{b(2,2)}^n = -\hat{q}_1^2 + \hat{q}_2^2 - \hat{q}_3^2 + \hat{q}_4^2 \quad (14e)$$

$$\hat{C}_{b(2,3)}^n = 2(\hat{q}_2\hat{q}_3 - \hat{q}_1\hat{q}_4) \quad (14f)$$

$$\hat{C}_{b(3,1)}^n = 2(\hat{q}_1\hat{q}_3 - \hat{q}_2\hat{q}_4) \quad (14g)$$

$$\hat{C}_{b(3,2)}^n = 2(\hat{q}_2\hat{q}_3 + \hat{q}_1\hat{q}_4) \quad (14h)$$

$$\hat{C}_{b(3,3)}^n = -\hat{q}_1^2 - \hat{q}_2^2 + \hat{q}_3^2 + \hat{q}_4^2. \quad (14i)$$

Roll ($\hat{\phi}$), pitch ($\hat{\theta}$) and yaw ($\hat{\psi}$) angles are updated using matrix \hat{C}_b^n as ([2], Equation 3.66)

$$\hat{\phi} = \arctan \left[\hat{C}_{b(3,2)}^n / \hat{C}_{b(3,3)}^n \right] \quad (15a)$$

$$\hat{\theta} = \arcsin \left[-\hat{C}_{b(3,1)}^n \right] \quad (15b)$$

$$\hat{\psi} = \arctan 2 \left[\hat{C}_{b(2,1)}^n, \hat{C}_{b(1,1)}^n \right]. \quad (15c)$$

3.1.3. Gravity

Equations to evaluate the estimated local gravity into the n-frame are ([2], Equation 3.89–3.91)

$$\hat{R}_0 = \sqrt{\hat{R}_M \hat{R}_N} \quad (16a)$$

$$\hat{g}(0) = 9.780318 \{ 1 + 5.3024E - 3 \sin^2[\hat{\gamma}_{(-)}] - 5.9E - 6 \sin^2[2\hat{\gamma}_{(-)}] \} \quad (16b)$$

$$\hat{g} = \hat{g}(0) \left[1 + \frac{\hat{h}_{(-)}}{\hat{R}_{0(-)}} \right]^{-2} \quad (16c)$$

$$\hat{\mathbf{g}}^n = [0, 0, \hat{g}]^T. \quad (16d)$$

3.1.4. Velocity

SINS velocity $\hat{\mathbf{v}}^n = [\hat{v}_N, \hat{v}_E, \hat{v}_D]^T$ is represented in the vehicle-carried NED coordinate system ([10], Section 2.2.4). $\hat{\mathbf{v}}^n$ is determined by the following equations ([2], Equation 3.69)

$$S = [\hat{\mathbf{v}}_{(-)}^n \times] \quad (17a)$$

$$\hat{\mathbf{f}}_c^n = \hat{C}_b^n [\tilde{\mathbf{f}}^b + \hat{\mathbf{b}}_{f(-)} + \delta \hat{\mathbf{b}}_{f(-)}] \quad (17b)$$

$$\Delta \hat{\mathbf{f}}^n = \hat{\mathbf{f}}_c^n + S(2\hat{\omega}_{ie}^n + \hat{\omega}_{en}^n) + \hat{\mathbf{g}}^n \quad (17c)$$

$$\hat{\mathbf{v}}^n = \hat{\mathbf{v}}_{(-)}^n + \Delta \hat{\mathbf{f}}^n \delta t_S, \quad (17d)$$

vectors $\hat{\mathbf{b}}_f$ and $\delta \hat{\mathbf{b}}_f$ are given by the EKF (Equation (21b)).

3.1.5. Position

As with GPS, SINS position is represented in the n-frame as $\hat{\mathbf{p}}^n = [\hat{\gamma}, \hat{\lambda}, \hat{h}]^T$ and is calculated in a specific order as ([3], Equation 5.49)

$$\hat{h} = \hat{h}_{(-)} - \hat{v}_D \delta t_S \quad (18a)$$

$$\hat{\gamma} = \hat{\gamma}_{(-)} + \frac{\hat{v}_N \delta t_S}{\hat{R}_{M(-)} + \hat{h}} \quad (18b)$$

$$\hat{\lambda} = \hat{\lambda}_{(-)} + \frac{\hat{v}_E \delta t_S}{\left[\hat{R}_N(\hat{\gamma}) + \hat{h} \right] \cos \hat{\gamma}}. \quad (18c)$$

3.2. Extended Kalman filter

The Kalman filter is an algorithm for linear systems that operates recursively on noisy input and output data to produce a statistically optimal estimate of the system states. The EKF is a nonlinear extension of the Kalman filter, which linearizes about an estimate of the current mean and covariance. This algorithm can fuse information coming from IMU and GPS to get state estimates with less errors when compared with using sensors separately.

The SINS error model is obtained by perturbing the mechanization equations (Section 3.1) and is given by a series of first-order differential equations. The resulting system is error-oriented and linear, although time-variant. Figure 2 depicts how SINS, GPS and EKF work together. Corrections in \hat{x} are fed back and used it as prior information to calculate estimates at the SINS. This scheme is known as close-loop correction ([3], Section 12.1.1).

The continuous state-space model of the system is ([12], Section 7.1)

$$\delta \dot{\hat{x}}_{(t)} = F_{(t)} \delta \hat{x}_{(t)} + G_{(t)} \mathbf{u}_{(t)} + \zeta_{(t)} \quad (19a)$$

$$\delta \hat{\mathbf{y}}_{(t)} = H \delta \hat{\mathbf{x}}_{(t)} + \mathbf{v}_{(t)}. \quad (19b)$$

The discrete state-space model of the system is ([12], Section 5.5)

$$\delta \hat{\mathbf{x}}_{(+)} = \Phi \delta \hat{\mathbf{x}} + G \mathbf{u} + \zeta \quad (20a)$$

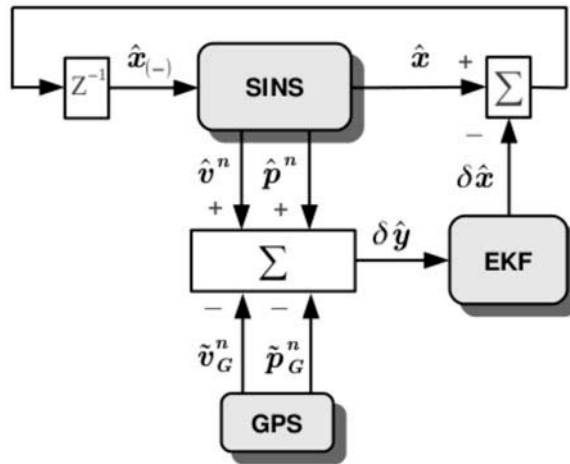


Figure 2. Diagram of EKF and SINS/GPS system integration.

$$\delta\hat{\mathbf{y}} = H\delta\hat{\mathbf{x}} + \mathbf{v}. \quad (20b)$$

Vectors $\zeta \sim \mathcal{N}(\mathbf{0}, Q)$ and $\mathbf{v} \sim \mathcal{N}(\mathbf{0}, R)$, Gaussian distributions with zero mean and known covariance matrix, are driving noise and measurement noise, respectively. Vectors $\mathbf{u} \in \mathbb{R}^{12}$, $\delta\hat{\mathbf{x}} \in \mathbb{R}^{21}$ and $\delta\hat{\mathbf{y}} \in \mathbb{R}^6$ are defined as

$$\mathbf{u} = \left[\tilde{\omega}^{bT}, \tilde{\mathbf{f}}^{bT}, \boldsymbol{\eta}_{g\delta b}^T, \boldsymbol{\eta}_{f\delta b}^T \right]^T \quad (21a)$$

$$\delta\hat{\mathbf{x}} = \left[\delta\hat{\mathbf{e}}^T, \delta\hat{\mathbf{v}}^{nT}, \delta\hat{\mathbf{p}}^{nT}, \hat{\mathbf{b}}_g^T, \hat{\mathbf{b}}_f^T, \delta\hat{\mathbf{b}}_g^T, \delta\hat{\mathbf{b}}_f^T \right]^T \quad (21b)$$

$$\delta\hat{\mathbf{y}} = \left[\delta\hat{\mathbf{y}}_v^T, \delta\hat{\mathbf{y}}_p^T \right]^T \quad (21c)$$

$$\delta\hat{\mathbf{y}}_v = [\hat{\mathbf{v}}^n - \hat{\mathbf{v}}_G^n] \quad (21d)$$

$$\delta\hat{\mathbf{y}}_p = \hat{T}_p^r [\hat{\mathbf{p}}^n - \hat{\mathbf{p}}_G^n] + \hat{C}_b^n \mathbf{l}_{ba}^b \quad (21e)$$

$$\hat{T}_p^r = \text{diag} \left\{ \left[\left(\hat{R}_M + \hat{h} \right), \left(\hat{R}_N + \hat{h} \right) \cos(\hat{\gamma}), -1 \right] \right\}, \quad (21f)$$

where \mathbf{l}_{ba}^b is the lever arm from the GPS antenna to the IMU, and \hat{T}_p^r is the curvilinear-to-Cartesian position change transformation matrix ([3], Equation 12.81).

Being $I_{\{3 \times 3\}}$ an identity matrix and $\mathbf{0}_{\{3 \times 3\}}$ a null matrix, state-space matrices are

$$F_{(t)\{21 \times 21\}} = \begin{bmatrix} F_{ee} & F_{ev} & F_{ep} & -\hat{C}_b^n & \mathbf{0} & -\hat{C}_b^n & \mathbf{0} \\ F_{ve} & F_{vv} & F_{vp} & \mathbf{0} & \hat{C}_b^n & \mathbf{0} & \hat{C}_b^n \\ \mathbf{0} & F_{pv} & F_{pp} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\frac{1}{\tau_g} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\frac{1}{\tau_f} \end{bmatrix} \quad (22)$$

$$G_{\{21 \times 21\}} = \begin{bmatrix} -\hat{C}_b^n & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{C}_b^n & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & I \end{bmatrix} \quad (23)$$

$$H_{\{6 \times 21\}} = \begin{bmatrix} \mathbf{0} & I & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \hat{T}_p^r & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}. \quad (24)$$

Submatrices F_{xx} are shown in [2] (Equation 12.28).

Covariance matrix Q is defined as ([3], Equation 12.67)

$$Q_{\{12 \times 12\}} = \text{diag}([n_g^{2T}, n_f^{2T}, n_{g\delta b}^{2T}, n_{f\delta b}^{2T}]), \quad (25)$$

where n_g^2 (Equation (2)), n_f^2 (Equation (4)), $n_{g\delta b}^2$ and $n_{f\delta b}^2$ are the power spectral densities (PSD) of, respectively, the gyroscopes random noises, accelerometers random noises, gyroscopes dynamic biases and accelerometers dynamic biases. $n_{g\delta b}^2$ and $n_{f\delta b}^2$ are obtained as ([3], Equation 12.68 and 12.69)

$$n_{g\delta b}^2 = \sigma_{g\delta b}^2 \circ \tau_g \quad (26a)$$

$$n_{f\delta b}^2 = \sigma_{f\delta b}^2 \circ \tau_f. \quad (26b)$$

Covariance matrix R is defined as,

$$R_{\{6 \times 6\}} = \text{diag}([\sigma_v^{2T}, \sigma_{\gamma m}^2, \sigma_{\lambda m}^2, \sigma_h^2]). \quad (27)$$

Diagonal entries from R are provided by the GPS error profile. Matrix \hat{T}_p^r in Equation (21e) motivates the use of $\sigma_{\gamma m}$ and $\sigma_{\lambda m}$ in R , instead of σ_γ and σ_λ . This way, diagonal elements from R are of the same order, which produces better numerical stability in calculating the Kalman gain, K (Algorithm 1).

Algorithm 1 presents a simplified version of EKF algorithm for one iteration ([2], Section 13.6.2.3). The master INS clock is the GPS clock. EKF evolves between GPS time steps, δt_G . Between GPS measurements, SINS keeps updating estimates of attitude, velocity and position.

Algorithm 1 Simplified extended Kalman filter

- 1: Update $\delta t_G = t_G - t_{G(-)}$ (Equation (21c)), F (Equation (22)), and G (Equation (23)).
 - 2: $\Phi = I + F\delta t_G$
 - 3: $Q_k = GQG^T\delta t_G$
 - 4: $K = (P_{(-)}H^T)(R + HP_{(-)}H^T)^{-1}$
 - 5: $\delta\hat{\mathbf{x}} = K\delta\hat{\mathbf{y}}$
 - 6: $P_t = (1 - KH)P$
 - 7: $P = \Phi P_t \Phi^T + Q_k$
 - 8: $P = \frac{1}{2}(P + P^T)$
-

The covariance matrix $P_{\{21 \times 21\}}$ must be set up initially before execution of Algorithm (1). It is formed as a diagonal matrix whose individual elements are chosen according to the expected initial variances of $\delta\hat{\mathbf{x}}$ (Equation (21b)) as

$$P_{(1)} = \text{diag}[\sigma_\phi^2, \sigma_\theta^2, \sigma_\psi^2, \sigma_v^{2T}, \sigma_\gamma^2, \sigma_\lambda^2, \sigma_h^2, b_g^{2T}, b_f^{2T}, \sigma_{g\delta b}^{2T}, \sigma_{g\delta f}^{2T}] \quad (28)$$

The Kalman filter may have serious numerical problems ([13], Chapter 6). To avoid this situation, it is necessary but not sufficient that covariance matrix P be symmetric at each iteration. Line 8 of Algorithm 1 ensures this condition ([4], Section 5.9.1).

4. INS algorithm

This section shows the main INS algorithm that exhibits the sequence in which navigation equations must be solved. In addition, some features of the proposed INS model are commented.

A loosely coupled SINS/GPS system is chosen between other types of integration schemes since it can be implemented with most low-cost GPS receivers, which usually only deliver position data in the navigation frame (n-frame). Besides, this frame has the property that horizontal (2D) and vertical errors are decoupled. In addition, n-frame position is represented in a more intuitive way when compared, for instance, with ECEF coordinates.

Chosen sensors for proposed INS are inertial sensors and GPS because they are ubiquitous in most navigation systems. This list of sensors does not pretend to be fixed. If a practitioner wants to compare her system with the proposed INS model using additional sensors, such as magnetometers or an odometer, she has to modify $\delta\hat{\mathbf{y}}$ (Equation (21c)), H (Equation (24)) and R (Equation (27)) in the provided MATLAB toolbox. Lastly, the practitioner has to report only the changes with respect to the benchmarking, instead of the complete model. Generally speaking, since the benchmarking is provided as open-source code, any change that has to be made to achieve a fair comparison between a particular INS and the proposed benchmarking is able to be carried out.

Algorithm 2 details the operation of the INS. All system variables are in SI units.

Vector $\hat{\mathbf{x}}$ is defined as

$$\hat{\mathbf{x}} = \left[\hat{\mathbf{e}}^T, \hat{\mathbf{v}}^{nT}, \hat{\mathbf{p}}^{nT}, \hat{\mathbf{b}}_g^T, \hat{\mathbf{b}}_f^T, \delta\hat{\mathbf{b}}_g^T, \delta\hat{\mathbf{b}}_f^T \right]^T. \quad (29)$$

Formulas for quaternion correction are ([1], Equation 7.34, A.172a, and A.173)

$$\hat{\mathbf{q}} = \hat{\mathbf{q}} + \frac{1}{2} \Xi(\hat{\mathbf{q}}) \delta\hat{\mathbf{e}} \quad (30a)$$

$$\Xi(\hat{\mathbf{q}})_{\{4 \times 3\}} = \begin{bmatrix} q_4 I_{\{3 \times 3\}} + [\boldsymbol{\varrho} \times] \\ -\boldsymbol{\varrho}^T \end{bmatrix}, \quad (30b)$$

where $\boldsymbol{\varrho}$ comes from Equation (12) and $\delta\hat{\mathbf{e}}$ is delivered by the EKF (Equation (21b)).

Attitude is initialized with estimated Euler angles, typically at rest. Position and velocity are initialized with corresponding GPS measurements.

Algorithm 2 INS algorithm.

- 1: Set $\sigma_{g\delta b}$, $\sigma_{f\delta b}$, $\boldsymbol{\tau}_g$, and $\boldsymbol{\tau}_f$
- 2: Set \mathbf{n}_g , \mathbf{n}_f , $\mathbf{n}_{g\delta b}$, and $\mathbf{n}_{f\delta b}$
- 3: Set σ_γ , σ_λ , $\sigma_{\gamma m}$, $\sigma_{\gamma n}$, σ_h , and σ_v
- 4: Load $\tilde{\boldsymbol{\omega}}^b$ (Equation (1)), $\tilde{\mathbf{f}}^b$ (Equation (3)), $\tilde{\mathbf{p}}_G^n$ (Equation (5a-c)), $\tilde{\mathbf{v}}_G^n$ (Equation (8)), \mathbf{t}_S , and \mathbf{t}_G
- 5: Set $\hat{\mathbf{e}}_{(1)} = \left[\hat{\phi}_{(1)}, \hat{\theta}_{(1)}, \hat{\psi}_{(1)} \right]$, $\hat{\mathbf{p}}_{(1)} = \tilde{\mathbf{p}}_{G(1)}^n$, and $\hat{\mathbf{v}}_{(1)} = \tilde{\mathbf{v}}_{G(1)}^n$
- 6: Calculate $\hat{\mathbf{q}}_{(1)}$ with $\hat{\mathbf{e}}_{(1)}$ ([2], Equation 3.65)
- 7: Calculate $\hat{\mathbf{C}}_{b(1)}^n$ with $\hat{\mathbf{q}}_{(1)}$ (Equation (14a-i))

```

8: Set  $\hat{\mathbf{b}}_{g(1)} = \mathbf{0}$ ,  $\hat{\mathbf{b}}_{f(1)} = \mathbf{0}$ ,  $\delta\hat{\mathbf{b}}_{g(1)} = \mathbf{0}$ , and  $\delta\hat{\mathbf{b}}_{f(1)} = \mathbf{0}$ 
9: Set  $\mathbf{I}_{ba}^n$ ,  $\mathbf{Q}$  (Equation (25)),  $\mathbf{R}$  (Equation (27)), and  $\mathbf{P}_{(1)}$  (Equation (28))
10: Guarantee that  $\mathbf{t}_{G(1)} < \mathbf{t}_{S(1)} = < \mathbf{t}_{G(2)}$ 
11: Guarantee that  $\mathbf{t}_{S(k_S-1)} < \mathbf{t}_{G(k_G)} = < \mathbf{t}_{S(k_S)}$ 
12: Set  $i = 2$ 
13: for  $k = 2$  to  $k_G$  do
14:   while  $\mathbf{t}_{S(i)} = < \mathbf{t}_{G(k)}$  do
15:     Update  $\hat{\omega}_{ie}^n$  and  $\hat{\omega}_{en}^n$  (Equations (9) and (10))
16:     Update  $\hat{\mathbf{q}}$  (Equation (13a–h))
17:     Update  $\hat{\mathbf{C}}_b^n$  (Equation (14a–i))
18:     Update  $\hat{\mathbf{e}}$  (Equation (15a–c))
19:     Update  $\hat{\mathbf{g}}^n$  (Equation (16a–d))
20:     Update  $\hat{\mathbf{v}}^n$  (Equation (17a–d))
21:     Update  $\hat{\mathbf{p}}^n$  (Equation (18a–c))
22:      $i = i + 1$ 
23:   end while
24:   Update  $\delta\hat{\mathbf{x}}$  (Algorithm 1)
25:   Correct  $\hat{\mathbf{x}} : \hat{\mathbf{x}} = \hat{\mathbf{x}} - \delta\hat{\mathbf{x}}$ 
26:   Correct  $\hat{\mathbf{q}}$  (Equation (30a) and (b))
27:   Correct  $\hat{\mathbf{C}}_b^n$  (Equation (14a–i))
28: end for

```

The INS algorithm considers that discrete-time series $\tilde{\omega}^b$ and $\tilde{\mathbf{f}}^b$ are synchronized, i.e., each is associated with the same time sequence \mathbf{t}_S with k_S elements. On the other hand, GPS discrete-time series $\tilde{\mathbf{p}}_G^n$ and $\tilde{\mathbf{v}}_G^n$ are related to time sequence \mathbf{t}_G with k_G elements. Generally, low-cost GPS receivers deliver estimates at lower frequencies than an IMU, then $k_G < k_S$. Sequences \mathbf{t}_S and \mathbf{t}_G must be adjusted for two reasons: to execute the SINS before the EFK at least once and to provide inertial data until the last GPS measurement.

The benchmarking has certain limitations. Neither corrections in computation of attitude are made for conning effects, nor velocity is corrected for sculling effects. In addition, GPS position is derived under the assumption that it is only affected by a zero-mean Gaussian noise. In some particular cases, noise distribution of GPS position should be considered as non-Gaussian.

5. Validation method

The method to validate the suggested INS mathematical model takes a practical approach. Three INS are set up using real-world data sets that were collected on field from three different MEMS IMU and a GPS. Then, obtained performances are analysed and checked if they are coherent with the quality of used sensors.

5.1. Experiment setup

Several IMU and a Novatel OEM4 dual-frequency GPS data sets are provided by [14], which were logged during the open-sky trajectory shown in Figure 3. In addition, a DGPS system was implemented using a Topcon Legacy GPS base station.

The trajectory took place in the vicinity of The Ohio State University, USA. The globe with a circle marks the beginning of the trajectory. On the other hand, the globe with a square points its end. The trajectory presents high and low dynamics. Firstly, the vehicle

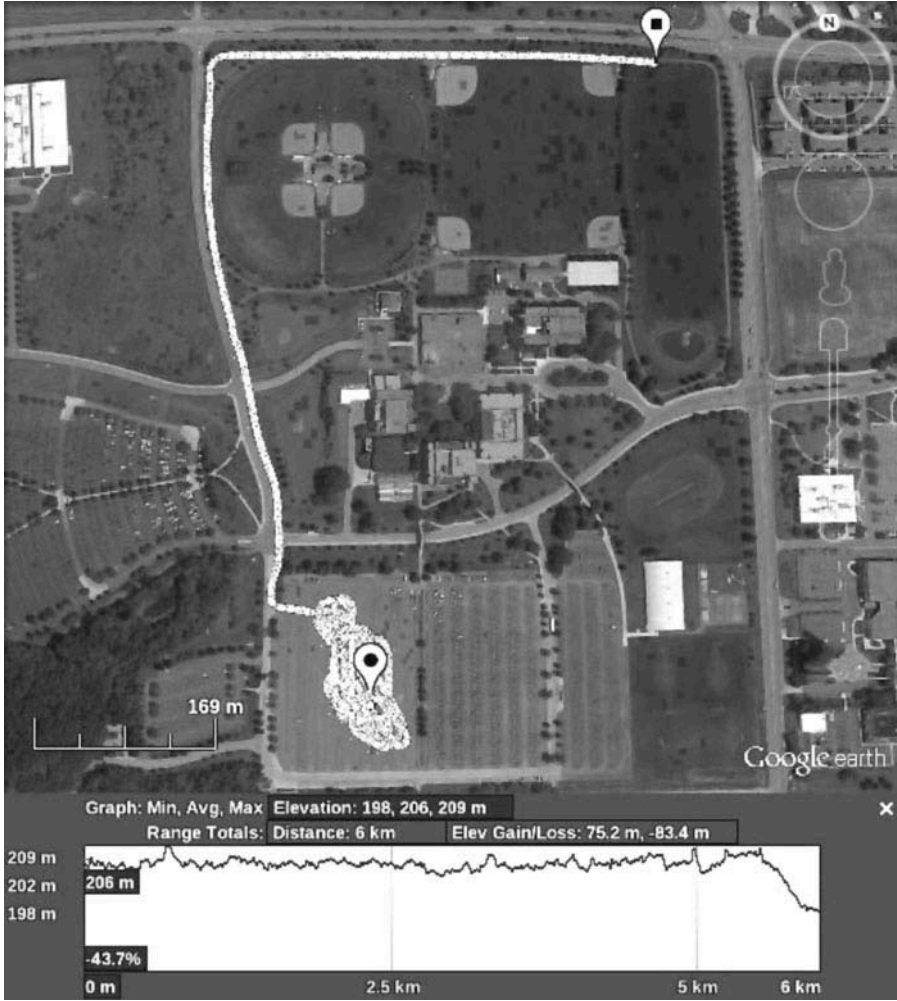


Figure 3. Trajectory during which data sets were generated (image courtesy of Google Earth [15]).

performs several ‘8’ patterns on a parking. Then, it is driven along a road. The path covers 14.67 minutes throughout which the availability of GPS signal is 100%.

Three MEMS IMU are selected from [14], namely, Xbow IMU400 CD (XBOW1), Xsens MTi (XSENS1) and Gladiator Landmark 10 (GLAD), to implement three integrated systems: XBOW1/GPS, XSENS1/GPS and GLAD/GPS. A reference data set is obtained by fusing navigation-grade Honeywell IMU H764 G-1 and DGPS measurements.

The same three MEMS IMU units used in [14] were profiled in [16], in which IMU units error characteristics were determined by using the Allan variance and PSD techniques. MEMS IMU noises terms ARW, VRW, $\sigma_{g\delta b}$, $\sigma_{f\delta b}$, τ_g and τ_f are taken from [16] (Table 4.4), ARW and VRW from AV column. It can be seen from that Table that XBOW1 and XSENS1 are of similar quality, and, on the contrary, GLAD presents a lower grade. GSP position error is 1.5 m CEP, and $\sigma_v = 0.03$ m/s, as stated in the GPS datasheet [17]. Lever arm vectors and $\hat{e}_{(1)}$ are provided.

Table 2. RMS errors for the three INS and GPS-only solution.

		XBOW1/GPS	XSENS1/GPS	GLAD/GPS	GPS only
Φ	deg	0.3020	0.6446	1.3390	–
Θ	deg	0.5717	0.6484	1.4247	–
Ψ	deg	3.7353	4.3586	6.6148	–
v_N	m/s	0.1570	0.1705	0.1717	0.1837
v_E	m/s	0.1465	0.1466	0.1698	0.1384
v_D	m/s	0.0181	0.0242	0.0720	0.0530
Γ	m	0.4223	0.4199	0.4444	0.6752
Λ	m	0.3426	0.3290	0.4193	0.4631
H	m	0.2960	0.2963	0.2979	1.3915

5.2. Result analysis

Table 2 displays the RMS errors with respect to the reference data set, for the three proposed INS and the GPS-only solution.

Table 2 exposes that XBOW1/GPS and XSENS1/GPS performances are close, since these IMU have also similar error profiles. On the other hand, GLAD/GPS shows a worse performance, which is coherent with GLAD quality. Moreover, it can be seen that the three integrated systems present smaller position errors than the GPS-only solution. All these scenarios are expected.

6. Conclusions

In this work, an approach to benchmarking for low-cost INS is proposed. This article shows the complete mathematical model of a loosely coupled INS, as well as the error models of inertial sensors and a GPS receiver. Plus, the main algorithm that details the INS operation is also exposed.

Suggested benchmarking is based on classical navigation equations. It is provided as an open-source toolbox for MATLAB and can be downloaded from [8]. The benchmarking is intended to generate a particular benchmark against which a novel INS solution can be objectively contrasted.

Validation of the benchmarking is achieved through processing real-world data sets from three MEMS IMU and a GPS receiver. It is observed that performances of the three INS are coherent with corresponding MEMS IMU error characteristics.

In conclusion, it is demonstrated that the proposed benchmarking is a suitable tool to evaluate new solutions to low-cost INS.

Acknowledgements

The authors would like to thank to Dr. Charles K. Toth (from The Ohio State University, USA), Dr. Allison Kealy, and Azmir Hasnur-Rabiain (both from The University of Melbourne, Australia) for generously sharing IMU and GPS data sets, and in particular, for Hasnur-Rabiain’s unselfish help. They would also like to thank to Dr. C.A. Catania (from the Universidad Nacional de Cuyo, Argentina) for his comments on this work.

Funding

This work was supported by the Agencia Nacional de Promoción Científica y Tecnológica, Argentina; and the Universidad Tecnológica Nacional, Argentina [grant number FONCyT IP-PRH 2007].

References

- [1] R.M. Rogers, *Applied Mathematics in Integrated Navigation Systems*, 2nd ed., American Institute of Aeronautics and Astronautics, Inc., Reston, VA, 2003.
- [2] D.H. Titterton and J.L. Weston, *Strapdown Inertial Navigation Technology*, 2nd ed., Institution of Engineering and Technology/The American Institute of Aeronautics and Astronautics, London/Reston, 2004.
- [3] P.D. Groves, *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*, Artech House, Norwood, MA, 2008.
- [4] J. Farrell, *Aided Navigation: GPS with High Rate Sensors*, McGraw-Hill Professional, New York, NY, 2008.
- [5] R. van der Merwe and E.A. Wan, *Sigma-point Kalman filters for integrated navigation*, Proceedings of the 60th Annual Meeting of The Institute of Navigation, Dayton, OH, June 2004, pp. 641–654.
- [6] S. Godha and M.E. Cannon, *Integration of DGPS with a low cost MEMS-based inertial measurement unit (IMU) for land vehicle navigation application*, Proceedings of the 18th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2005), Long Beach, CA, September 2005, pp. 333–345.
- [7] A.K. Brown and Y. Lu, *Performance test results on an integrated GPS/MEMS inertial navigation package*, Proceedings of the 17th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2004), Long Beach, CA, September 2004.
- [8] *Benchmarking for loosely-coupled low-cost navigation systems*, (2014). Available at <http://www.github.com/rodrazalez/navego>
- [9] J.M. Strus, M. Kirkpatrick, and J.W. Sinko, *Development of a high accuracy pointing system for maneuvering platforms*, Proceedings of the 20th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2007), Fort Worth, TX, September 2007, pp. 2541–2549.
- [10] G. Cai, B.M. Chen, and T.H. Lee, *Unmanned Rotorcraft Systems*, Springer, London, 2011.
- [11] J.L. Crassidis and J.L. Junkins, *Optimal Estimation of Dynamic Systems*, 2nd ed., Chapman and Hall/CRC, Boca Raton, FL, 2011.
- [12] R.G. Brown and P.Y.C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering*, 3rd ed., John Wiley & Sons, Inc., Hoboken, NJ, 1997.
- [13] M.S. Grewal and A.P. Andrews, *Kalman Filtering: Theory and Practice Using MATLAB*, 3rd ed., John Wiley & Sons, Inc., Hoboken, NJ, 2008.
- [14] C. Toth, D. Brzezinska, N. Politi, and A. Kealy, *Reference data set for performance evaluation of MEMS-based integrated navigation solutions*, FIG Working Week 2011, Marrakech, May 2011.
- [15] Google Earth. Imagery © 2014 DigitalGlobe, U.S. Geological Survey, USDA Farm Service Agency. The Ohio State University, USA, 2014. 40.001319N, 83.039045W.
- [16] A. Hasnur-Rabiain, *Performance evaluation of MEMS based INS/GPS integration*, Master's thesis, Department of Geomatics, The University of Melbourne, 2010.
- [17] Novatel Inc., *OEM4 Family User Manual (OM-20000046 Rev 19)*, Calgary, Novatel Inc., 2005.