International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 10 Issue: 11 | Nov 2023 www.irjet.net p-ISSN: 2395-0072

Experiment to determine a method for effortless static calibration of the inertial measurement unit on a small unmanned aerial vehicle

David Kofi Oppong, Joshua Ampofo, Anthony Agyei-Agyemang, Eunice Akyereko Adjei, Kwasi Kete Bofah, Victoria Serwaa-Bonsu Acheamfour

Department of Mechanical Engineering, Kwame Nkrumah University of Science and Technology, Kumasi – Ghana

---

Abstract - *This paper details a series of activities carried out in an effort to develop a means for static calibration of the micro-electro-mechanical-systems inertial measurement unit on a small unmanned aerial vehicle. The study involved collecting raw acceleration and angular velocity data from the inertial measurement unit on an Ardupilot flight controller and processing the measurements to determine the calibration parameters. An established calibration procedure provided benchmark parameters against which the experimental results were compared. We discovered that it is possible to calibrate the inertial measurement unit of a small unmanned aerial vehicle by collecting about a minute's worth of static data. Although the values of the calibration parameters differ in some ways from those of the benchmark calibration algorithm, an accurate representation of the true inputs to the system is obtained. We explore an approach that requires no special equipment or precise maneuvers in the calibration of these sensors.*

Key Words: Static calibration, inertial measurement unit, micro-electro-mechanical systems, unmanned aerial vehicle

1. INTRODUCTION

The use of small unmanned aerial vehicles (UAVs) has been proven in several scenarios, including surveying and mapping, package delivery, surveillance, precision agriculture, and recreation [1]. Most of these UAVs come with micro-electro-mechanical system (MEMS) sensors, usually accelerometers and gyroscopes, which provide low-cost, small form-factor sensor solutions. Owing to manufacturing (and installation) imperfections, errors are inherent in the output of these sensors [2]. Three of the most prominent are misalignment, bias, and scale factor error. The goal of calibration is to eliminate these, and accurately describe the relationship between the true value of the measurement and the sensor output.

Standard calibration equipment usually costs significantly more than the MEMS sensors themselves [3], and an alternative calibration procedure is often desired. Several researchers have proposed calibration methods that do not require expensive equipment. [4] used a set of 18 positions (6 flat faces and 12 edges of the sensor) to collect accelerometer data, and 9 positions (coordinate axes perpendicular and parallel to a hinge) to collect gyroscope data for calibration. [5] used a similar approach with accelerometers; with the sensor mounted on a patient, they collected data and used a threshold to determine quasi-static states, on which they applied an estimator to determine the sensitivity and offset of the accelerometer. [6] also avoided expensive calibration equipment by collecting data with the inertial measurement unit (IMU) moved by hand and placed in several static positions. These approaches require the sensors to be placed in several different configurations in order to determine the calibration parameters. Sometimes this is not possible or convenient, especially when the sensors are rigidly attached to the airframe of a small UAV, and a method that requires absolutely no movement of the platform is desirable.

We begin by examining models of these MEMS sensors, which give us an idea of the calibration parameters to be determined.

2. SENSOR MODELS

It is necessary to define a sufficiently accurate representation of the sensors if they are to be calibrated properly. [7] provide the following models.

2.1 Accelerometers

$$a_s = K_a \cdot R_a \cdot a_p + b \tag{1}$$

where,

a_s is the accelerometer output in the sensor axes frame; a_p is the sensor platform acceleration in body-fixed coordinates, that is, the desired value;

$R_a = [T_a]^{-1}$ where T_a is the alignment matrix: $T_a = \begin{bmatrix} 1 & -\alpha_{yz} & \alpha_{zy} \\ \alpha_{xz} & 1 & -\alpha_{zx} \\ -\alpha_{xy} & \alpha_{yx} & 1 \end{bmatrix}$

the term α_{ij} in T_a is the rotation of the *i*-accelerometer sensitivity axis around the j-th platform axis.

To simplify the alignment matrix, we define the sensor and body axes such that their x-axes coincide, and the body y-axis lies in the plane defined by the sensor x- and y-axes.

As a result [8], $\alpha_{xz} = \alpha_{xy} = \alpha_{yx} = 0,$ and $T_a = \begin{bmatrix} 1 & -\alpha_{yz} & \alpha_{zy} \\ 0 & 1 & -\alpha_{zx} \\ 0 & 0 & 1 \end{bmatrix}$.

$K_a = \begin{bmatrix} k_{x_a} & 0 & 0 \\ 0 & k_{y_a} & 0 \\ 0 & 0 & k_{z_a} \end{bmatrix}$, the sensitivity (scale factor) matrix; and *b* is the sensor bias.

By multiplying K_a and R_a, we obtain the 3 x 3 matrix, $F = \begin{bmatrix} k_{x_a} & k_{x_a}\alpha_{yz} & -k_{x_a}\alpha_{zy} \\ 0 & k_{y_a} & k_{y_a}\alpha_{zx} \\ 0 & 0 & k_{z_a} \end{bmatrix}$, where the products of the elements of the alignment matrix have been dropped.

Thus, calibrating the accelerometer requires the computation of nine parameters: the nonzero elements of F and bias.

2.2 Gyroscopes

$$\omega_s = K_g \cdot R_g \cdot \omega_p + b \quad (2)$$

where,

ω_s is the gyroscope output in the sensor frame; ω_p is the sensor platform rotational velocity in body-fixed coordinates, that is, the desired value;

$R_g = [T_g]^{-1}$ where T_g is the alignment matrix defined as, $T_g = \begin{bmatrix} 1 & -\gamma_{yz} & \gamma_{zy} \\ \gamma_{xz} & 1 & -\gamma_{zx} \\ -\gamma_{xy} & \gamma_{yx} & 1 \end{bmatrix}$, where the term γ_{ij} is the rotation of the i gyroscope sensitivity axis around the j-th platform axis.

Again, we define the sensor and body axes such that their x-axes coincide, and the body y-axis lies in the plane defined by the sensor x- and y-axes.

As a result, $\gamma_{xz} = \gamma_{xy} = \gamma_{yx} = 0,$ and $T_a = \begin{bmatrix} 1 & -\gamma_{yz} & \gamma_{zy} \\ 0 & 1 & -\gamma_{zx} \\ 0 & 0 & 1 \end{bmatrix}$.

$K_g = \begin{bmatrix} k_{x_g} & 0 & 0 \\ 0 & k_{y_g} & 0 \\ 0 & 0 & k_{z_g} \end{bmatrix}$ is the sensitivity (scale factor) matrix; and b is the sensor bias. By multiplying K_g and R_g, we obtain the 3 x 3 matrix, $E = \begin{bmatrix} k_{x_g} & k_{x_g}\gamma_{yz} & -k_{x_g}\gamma_{zy} \\ 0 & k_{y_g} & k_{y_g}\gamma_{zx} \\ 0 & 0 & k_{z_g} \end{bmatrix}$,

where the products of the elements of the alignment matrix have been dropped.

Therefore, calibrating the gyroscope is a problem of determining the six components of E and the three components of the bias.

3. EXPERIMENTAL PROCEDURE

The experiment was carried out using an APM 2.8 flight controller running the Ardupilot multicopter firmware. The hardware was connected to a computer via the USB interface and telemetry logs of the raw inertial measurement unit were taken. Mission planner [9], an opensource ground station software, provided the computer interface to the controller.

3.1 Benchmark data

Following [10], two sets of benchmark data were collected: one for the calibration of the accelerometer and the other for the gyroscope.

For the accelerometer, each sensor axis was placed parallel and antiparallel to the gravity vector. Approximately 20 seconds of data was collected in each case.

For the gyroscope, the z-axis was placed parallel and perpendicular to the gravity vector for 20 seconds each. Then, each sensitive axis was placed antiparallel to gravity, and the flight controller rotated anticlockwise about that particular axis through an angle of 360^0 in increments of 90^0. After each 90-degree rotation, the sensor was left idle for approximately 20 seconds. Finally, the z-axis was again placed parallel and perpendicular to the gravity vector, and 20 seconds of data collected in each case.

3.2 Test data

Given that our aim was to develop a method for static calibration of the inertial measurement unit, we collected about a minute's worth of accelerometer and gyroscope

data[1] from the stationary flight controller with its x-axis pointing north (with the help of a portable magnetic compass) and the z-axis parallel to the gravity vector (with the help of a plumbline).

4. BENCHMARK CALIBRATION PROCEDURE

The calibration parameters were first obtained using the procedure suggested by [10] on the benchmark data.

4.1 Accelerometer bias

The sensor bias was calculated using,

$$b_i = \frac{a_{i,p_i} + a_{i,ap_i}}{2} \quad (3)$$

where *i* is the sensor axis (x, y or z), a_{i,p_i} is the acceleration recorded along the *i* axis when the sensitive axis, *i*, is placed parallel to the gravity vector, and a_{i,ap_i}, the acceleration recorded along the *i* axis when the sensitive axis, *i*, is placed anti-parallel to the gravity vector. Because the data was collected for about 20 seconds in each case, $a_{p,i}$ and $a_{ap,i}$ represent the average values.

4.2 Accelerometer coupling matrix

Determining the matrix encapsulating the scale factors and alignment (coefficient of a_p in $a_s = K_a \cdot R_a \cdot a_p + b$ (*1*) involves the computation of two matrices, U+ and U- as follows:

$$U+= \begin{bmatrix} a_{x,p_x} & a_{x,p_y} & a_{x,p_z} \\ a_{y,p_x} & a_{y,p_y} & a_{y,p_z} \\ a_{z,p_x} & a_{z,p_y} & a_{z,p_z} \end{bmatrix}$$

where the notation a_{i,p_j} represents the accelerometer value along the *i* axis when the sensor is positioned so that its sensitive axis, *j*, is parallel to the gravity vector;

$$U-= \begin{bmatrix} a_{x,ap_x} & a_{x,ap_y} & a_{x,ap_z} \\ a_{y,ap_x} & a_{y,ap_y} & a_{y,ap_z} \\ a_{z,ap_x} & a_{z,ap_y} & a_{z,ap_z} \end{bmatrix}$$

where a_{i,ap_j} is the accelerometer value along the *i* axis when the sensor is positioned such that its sensitive axis, *j*, is aligned anti-parallel to the gravity vector.

It is also necessary to compute the quantity,

$$Q = mean(A, B, C) \quad (4)$$

where, *A* is the norm of the first column of U+, *B* is the norm of column 2 of U+ and *C* is the norm of the last column of U+, although U– can be used instead of U+.

The coupling matrix is then equal to $\frac{1}{2Q}\big((U+) - (U-)\big)$.

4.3 Gyroscope bias

Gyroscope bias was modelled as a linear equation in time,

$$b_i = b_{0_i} + b_{1_i} \cdot t \quad (5)$$

where, *i*, is the sensor axis.

Knowing the time elapsed over the collection of benchmark gyroscope data when the z-axis of the sensor was placed parallel to the gravity vector allows the constants b_{0_x}, b_{0_y}, b_{1_x} and b_{1_y} to be determined. b_{0_z} and b_{1_z} are obtained using the data obtained and the time elapsed when the z-axis is perpendicular to the gravity vector.

4.4 Gyroscope coupling matrix

The matrix of scale factors and alignment (coefficient of ω_p in $\omega_s = K_g \cdot R_g \cdot \omega_p + b$ (*2*) is obtained by dividing the result of integrating the gyroscope output by the total angular displacement when it is rotated through 360^0 in 90^0 increments.

5. PROPOSED CALIBRATION PROCEDURE

It is required that the sensor platform be mounted parallel to the local horizontal plane (with the help of a plumbline), and the forward direction towards the north (with the aid of a portable magnetic compass). The sensors may then be calibrated.

5.1 Accelerometer

The bias in the sensor output is a DC component; it is the value of the output when the input is zero. It may be determined in the following way:

1. The mean is computed for the output of each axis of the accelerometer

[1] The duration of 1 minute was a compromise between the time required for the sensor output to settle to steady state and the maximum wait time before flight after power-on.

2. The Euclidean norm of the three-component mean vector is calculated and subtracted from the sensor output along the *z*-axis
3. The mean, R_i, is computed for the output of *Step 2* (*i* represents the sensitive axis: *x, y or z*)
4. The output of *Step 2* is then passed through a high-pass filter using a cut-off frequency, $f_c = 0.5\ Hz$, adapted from [5].
5. The mean of the filter output, R_{fi} is determined; again, *i* represents the specific axis: *x, y or z*
6. The bias is determined by solving *Equation* ***Error! Reference source not found.*** for *b*, using $a_s = \begin{bmatrix} R_x \\ R_y \\ R_z \end{bmatrix}$, $a_p = \begin{bmatrix} R_{fx} \\ R_{fy} \\ R_{fz} \end{bmatrix}$ and, $F = K_a \cdot R_a$, equal to the 3 x 3 identity matrix.

The remaining parameters (components of matrix *F*) are determined next.

7. Because the sensor platform has been positioned parallel to the local horizontal, the gravity vector sensed has only a vertical component, equal to the acceleration due to gravity [11]; this is taken to be equal to the norm of the sensor output as calculated in *Step 2* above.

8. The first two components of the diagonal of matrix F are forced to a constant value: their effect is absorbed by the off-diagonal elements, which are now labelled α, β, γ and μ.
Hence,
$$F = \begin{bmatrix} k_{x_a} & k_{x_a}\alpha_{yz} & -k_{x_a}\alpha_{zy} \\ 0 & k_{y_a} & k_{y_a}\alpha_{zx} \\ 0 & 0 & k_{z_a} \end{bmatrix} = \begin{bmatrix} constant & \alpha & \beta \\ 0 & constant & \gamma \\ 0 & 0 & \mu \end{bmatrix}$$

9. As a consequence of *Step 7*, the value of α is "don't care", and can be set equal to 0. Thus,
$$F = \begin{bmatrix} constant & 0 & \beta \\ 0 & constant & \gamma \\ 0 & 0 & \mu \end{bmatrix}$$

10. *Equation* ***Error! Reference source not found.*** may now be solved for the three unknowns (β, γ, μ) using,

$a_s = \begin{bmatrix} R_x \\ R_y \\ R_z \end{bmatrix}$, $a_p = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}$ and b equal to the bias (calculated earlier).

5.2 Gyroscope

1. The mean, R_i, is computed for the output of each axis of the gyroscope (*i* represents the specific axis: *x, y or z*)

2. The sensor output (for each axis) is then passed through a high-pass filter using a cut-off frequency, $f_c = 0.5\ Hz$, adapted from [5].

3. The mean of the filter output, R_{fi} is determined

4. The bias is determined by solving *Equation* ***Error! Reference source not found.*** for *b*, using $\omega_s = \begin{bmatrix} R_x \\ R_y \\ R_z \end{bmatrix}$, $\omega_p = \begin{bmatrix} R_{fx} \\ R_{fy} \\ R_{fz} \end{bmatrix}$ and, $E = K_g \cdot R_g$, equal to the 3 x 3 identity matrix.

Next, the components of matrix E are determined.

5. The first two components of the diagonal elements of *E* are forced to a constant value, and their effect is absorbed by the off-diagonal elements, which are now represented as σ, τ, ξ and ν.
Hence,
$$E = \begin{bmatrix} k_{x_g} & k_{x_g}\gamma_{yz} & -k_{x_g}\gamma_{zy} \\ 0 & k_{y_g} & k_{y_g}\gamma_{zx} \\ 0 & 0 & k_{z_g} \end{bmatrix} = \begin{bmatrix} constant & \sigma & \tau \\ 0 & constant & \xi \\ 0 & 0 & \nu \end{bmatrix}.$$

6. With the aid of a GPS receiver, the location (latitude-longitude) of the sensor platform is determined and used to compute the direction cosine matrix,
$$C_{ecef}^{ned} = \begin{bmatrix} -sin(lat) * cos(lon) & -sin(lat) * sin(lon) & cos(lat) \\ -sin(lon) & cos(lon) & 0 \\ -cos(lat) * cos(lon) & -cos(lat) * sin(lon) & -sin(lat) \end{bmatrix}$$
that transforms the Earth's rotation vector, $\omega_e = \begin{bmatrix} 0 \\ 0 \\ 7.292115e-5 \end{bmatrix} rad/s$ from the Earth frame to the local navigational frame. It is realized that the East component of Earth's rotation rate turns out to be zero, and hence, the value of σ is "don't care" and can be set equal to 0. As a result, matrix *E* is reduced to, $\begin{bmatrix} constant & 0 & \tau \\ 0 & constant & \xi \\ 0 & 0 & \nu \end{bmatrix}$.

7. *Equation* ***Error! Reference source not found.*** may now be solved for the three unknowns (τ, ξ, ν) using,

$\omega_s = \begin{bmatrix} R_x \\ R_y \\ R_z \end{bmatrix}$, $\omega_p = C_{ecef}^{ned} * \omega_e$, and b equal to the bias (calculated earlier).

6. RESULTS AND DISCUSSION

6.1 Results

The tables below summarize the experimental results.

Table -1: Gyroscope bias: benchmark vs. proposed calibration procedures

Sensor axis	$b_0\ (rad/s)$			$b_1\ (rad/s^2)$
	Benchmark	Proposed	% Error	Benchmark
x	0.1901E-3	0.2E-3	+10.8	0.0016E-3
y	-0.1953E-3	-0.4E-3	+108.2	-0.0007E-3
z	-0.5823E-3	-1.1E-3	+81.0	-0.0017E-3

Table -2: Gyroscope coupling matrix: benchmark vs. proposed procedures

Benchmark	Proposed	% Error
$\begin{bmatrix} 0.0128 & 0.0005 & 0.0002 \\ 0.0126 & 0.0171 & -0.0007 \\ 0.0118 & 0.0179 & 0.0199 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 8.5258 \\ 0 & 1 & -0.0283 \\ 0 & 0 & 0.1586 \end{bmatrix}$	$\begin{bmatrix} 7.7377E3 & 100 & 4.8885E6 \\ 100 & 5.7486E3 & 3.8037E3 \\ 100 & 100 & 6.9610E2 \end{bmatrix}$

Table -3: Accelerometer bias: benchmark vs. proposed procedures

Sensor axis	$b\ (m/s^2)$		
	Benchmark	Proposed	% Error
x	-0.0060	-0.1015	+1.58E3
y	0.1310	0.0470	-64.1
z	0.8262	0.0007	-99.9

Table -4: Accelerometer coupling matrix: benchmark versus proposed procedures

Benchmark	Proposed	% Error
$\begin{bmatrix} -0.9650 & -0.0382 & -0.0245 \\ 0.0293 & -0.9674 & 0.0064 \\ 0.0599 & 0.0408 & -0.9735 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 5.003E-9 \\ 0 & -1 & -8.290E-7 \\ 0 & 0 & -1.000 \end{bmatrix}$	$\begin{bmatrix} -3.622 & 100 & 100.00 \\ 100 & -3.371 & 100.01 \\ 100 & 100 & -2.717 \end{bmatrix}$

6.2 Discussion

The tables compare the results obtained using the benchmark calibration algorithm with those obtained using the proposed algorithm. The gyroscope bias was most accurately computed for the x-axis, with the proposed algorithm recording a deviation of only 10.8%. Although the y- and z-axis biases were over-predicted, the signs agreed with the benchmark values. The signs again agreed with the accelerometer biases, although the deviation was much greater. Bias drift of the gyroscopes was observed to be minimal as its coefficients were of the order of $10^{-6}\ rad/s^2$.

The diagonal elements of the accelerometer coupling matrices agreed for both algorithms with a maximum error of 3.62%. The off-diagonal elements showed the greatest deviation, reaching up to 100% of the benchmark value, and this is the case because of fixing those values to zero. For the gyroscope, the deviations were bigger, although the signs were preserved.

The first two elements of the diagonals of the coupling matrix, using the proposed algorithm, are taken to be constants, and a reasonable starting value is an absolute value of 1[2]. The value 1 works well for the gyroscope, but leads to inversion of the *x*- and *y*-inputs to the accelerometer, hence they are negated for the accelerometer.

A quick referral was made to the sensor datasheets to validate the outputs of the proposed calibration algorithm. ***Error! Reference source not found.*** [12] provides a summary of the error limits for the accelerometer and gyroscope.

Table -5: Inertial sensor datasheet specifications

Sensor	Bias	Sensitivity scale factor	Cross-axis sensitivity
Gyroscope	±0.35 rad/s	±0.97	±0.02
Accelerometer	±0.49 m/s² *(x/y-axis)* ±0.78 m/s² *(z-axis)*	±0.97	±0.02

It is observed that the outputs of the proposed calibration algorithm fall within the limits specified by the datasheet.

7. EVALUATION

To further validate the proposed calibration algorithm, benchmark data used to calibrate the accelerometer was applied, and Equations **Error! Reference source not found.** and **Error! Reference source not found.** solved for the true inputs. For comparison, the benchmark algorithm was also used. *Figures 1* to *6* show the agreement between the proposed and benchmark algorithms. In some cases, particularly with angular velocities, the proposed algorithm provided a smoother representation of the inputs.

[2] An absolute value of 1 is the ideal scale factor value

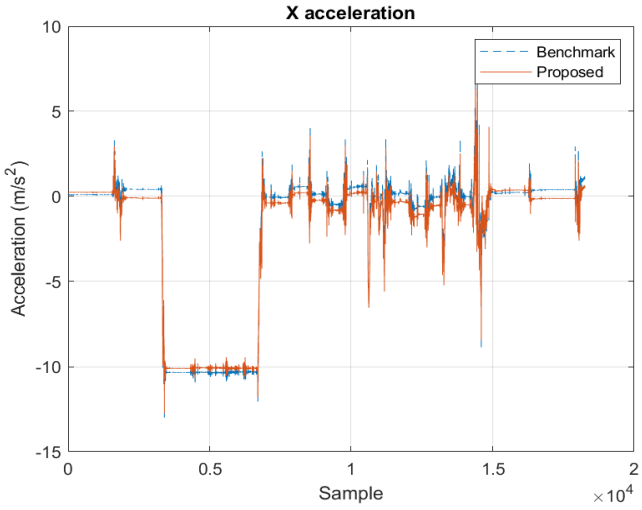

Fig -1: X-axis acceleration input - benchmark vs. proposed procedures

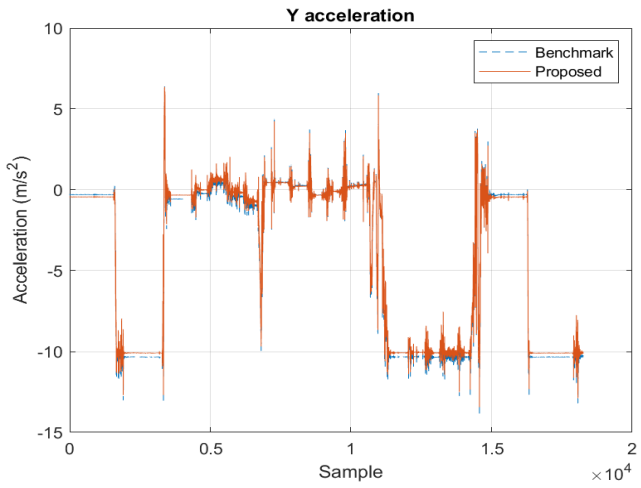

Fig -2: Y-axis acceleration input - benchmark vs. proposed procedures

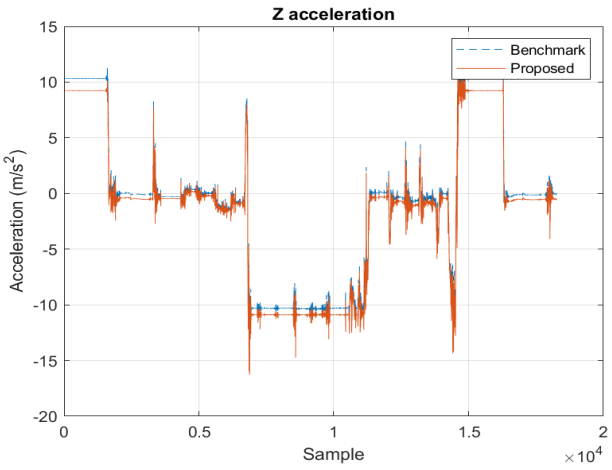

Fig -3: Z-axis acceleration input - benchmark vs. proposed procedures

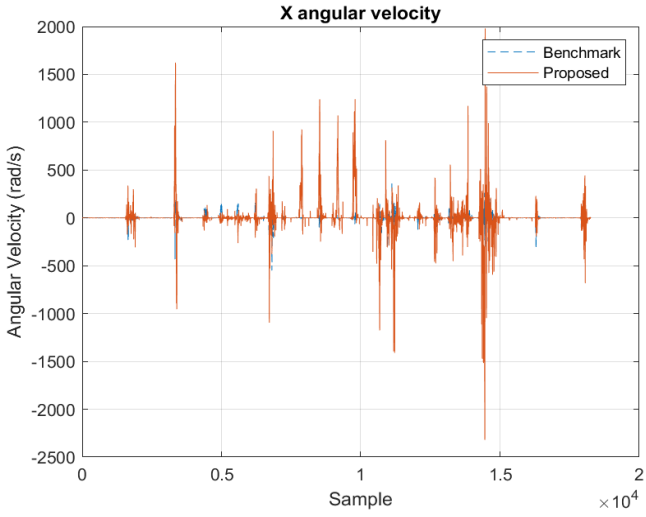

Fig -4: X-axis angular velocity input - benchmark vs. proposed procedures

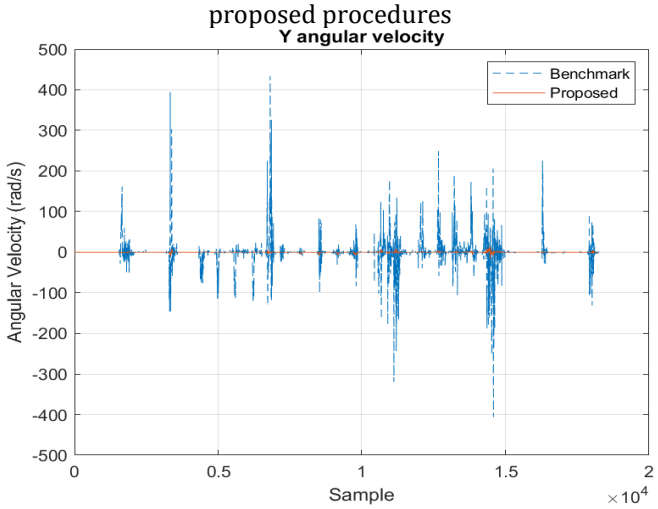

Fig -5: Y-axis angular velocity input - benchmark vs. proposed procedures

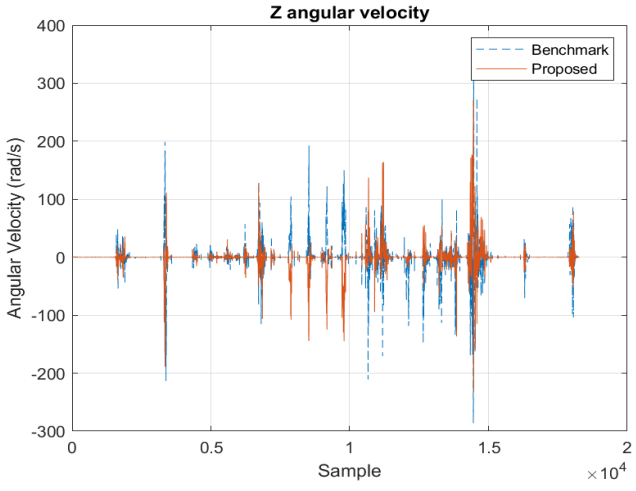

Fig -6: Z-axis angular velocity input - benchmark vs. proposed procedures

8. CONCLUSION

The experiment showed that it is possible to statically calibrate the inertial measurement unit on a small unmanned aerial vehicle to a good degree of accuracy without requiring special equipment. Although the calibration parameters differ from those obtained using a different calibration algorithm, their signs are largely the same, and the proposed method provides a good estimate of the true inputs to the system.

REFERENCES

[1] B. Terwilliger, D. Ison, J. Robbins and D. Vincenzi, Small Unmanned Aircraft Systems Guide: Exploring Designs, Operations, Regulations, and Economics, Newcastle: Aviation Supplies & Academics, Inc., 2017.

[2] A. G. Quinchia, G. Falco , E. Falletti, F. Dovis and C. Ferrer, "A Comparison between Different Error Modeling of MEMS Applied to GPS/INS Integrated Systems," Sensors, pp. 9549-9588, 2013.

[3] I. Skog and P. Handel, "Calibration of a MEMS Inertial Measurement Unit," Rio de Janeiro, Brazil, 2006.

[4] W. T. Fong, S. K. Ong and A. Y. C. Nee, "Methods for in-field user calibration of an inertial measurement unit without external equipment," Measurement Science and Technology, vol. 19, pp. 1-11, 2008.

[5] J. C. Lötters, J. Schipper, P. H. Veltink, W. Olthuis and P. Bergveld, "Procedure for in-use calibration of triaxial accelerometers in medical applications," Sensors and Actuators A: Physical, vol. Volume 68, no. 1–3, pp. 221-228, 1998.

[6] D. Tedaldi, A. Pretto and E. Menegatti, "A Robust and Easy to Implement Method for IMU Calibration without External Equipments," Manila, 2013.

[7] A. Saxena, G. Gupta, V. Gerasimov and S. Ourselin, "In use parameter estimation of inertial sensors by detecting multilevel quasi-static states," Lecture notes in computer science, vol. 3684, pp. 595-601, 2005.

[8] ArduPilot Development Team, "Mission Planner Home," 2023. [Online]. Available: https://ardupilot.org/planner/. [Accessed 7 6 2023].

[9] F. Ferraris, U. Grimaldi and M. Parvis, "Procedure for effortless in-field calibration of three-axis rate gyros and accelerometers," Sensors and Materials, vol. 7, no. 5, pp. 311-330, 1995.

[10] B. L. Stevens, F. L. Lewis and E. N. Johnson, Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems, 3rd ed., Hoboken, New Jersey: John Wiley & Sons, Inc., 2016.

[11] B. L. Stevens, F. L. Lewis and E. N. Johnson, Aircraft Control and Simulation: Dynamics, Controls Design and Autonomous Systems, Hoboken, New Jersey: John Wiley & Sons, Inc. , 2016.

[12] InvenSense Inc., MPU-6000 and MPU-6050 Product Specification Revision 3.4, Sunnyvale, CA : InvenSense Inc., 2013.