Dynamic Calibration of Triaxial Accelerometers with Simple Setup

Federico Pedersini

***Abstract*— In the dynamic techniques for calibration of triaxial accelerometers, the sensor is held in motion during data acquisition; the acquired time-varying acceleration is in general more informative than the constant acceleration acquired in static calibration approaches. Dynamic methods, however, typically require complex and expensive calibration setups based on high-precision moving benches, to guarantee accurate a-priori knowledge of the acceleration applied to the sensors under test.**

This article presents a dynamic calibration procedure working with a very simple and inexpensive setup: a tilted, freely-rotating bench, made of a simple wheel mounted on a static support. The accelerometer is fastened to the bench, which is then manually set into rotation; the acceleration signal is then acquired while the bench is freely rotating.

No a-priori knowledge is required about the bench rotation: the proposed calibration procedure estimates both the calibration parameters and the bench motion.

To keep the setup as simple as possible, an effort has been made to minimize also the necessity of prior knowledge of the bench geometry: only the distance of the sensor from the rotation axis needs to be known, which can be easily obtained through direct measurement on the bench.

The proposed calibration has been tested both on synthetic data, to prove the absence of estimation biases and to evaluate the potential accuracy of the estimated parameters, and on real data from a MEMS triaxial accelerometer, to assess the practical usability and measure the actual precision of the procedure.

***Index Terms*— Accelerometers, Dynamic Calibration, IMU, MEMS, Sensor Modeling**

I. Introduction

MEMS accelerometers are widely used in many fields, like in motion analysis applications, where accelerometers are part of the Inertial Measurement Units (IMU). In such applications the acceleration signal, besides being combined with gyroscopic and magnetometric measurements [1]–[4], is integrated over time to obtain the velocity, and integrated twice to obtain the displacement. The major limitation of this technique is the steadily growing error, or *drift*, affecting the obtained velocity and displacement, caused by the integration over time of an acceleration signal affected by systematic errors, like offsets and differences in gain along the three axes. For this reason, an accurate calibration of the accelerometer is of crucial importance to minimize these drifts.

Many different approaches have been proposed in literature for the calibration of triaxial accelerometer [2], [4]–[13], featuring a large variety of calibration setups and sensor models. In general, more sophisticated calibration approaches are able to estimate more complex accelerometer models. Complex models consider, for instance, nonorthogonal sensing axes [5], [7], or nonlinear components of the sensor response [6], [9]. However, for most commercially-available triaxial MEMS accelerometers, a six-parameter model, considering independent values for the three components (along the sensing axes) of offset and gain, is normally more than sufficient to fully exploit the accuracy potential of the sensor.

F. Pedersini is with the University of Milan, Department of Computer Science – via Giovanni Celoria 18, I-20133 Milan, Italy – e-mail: federico.pedersini@unimi.it.

The accelerometer calibration techniques presented in literature can be subdivided into two main groups: *static* and *dynamic* methods. In static calibrations, each acceleration sample is acquired while the sensor is held in a static position [4], [5], [7], [10]–[12]. Consequently, these procedures need to carry out many data acquisitions, one for each time the sensor is positioned and held fixed on a new orientation. Conversely, in dynamic calibrations, the acceleration samples are acquired continuously, while the sensor is undergoing a controlled motion [2], [6], [9], [13], [14]. Consequently, dynamic calibrations typically need a single data sequence, which makes the acquisition process simpler, compared to static procedures.

The accuracy of dynamic calibration approaches strongly depends on how much and how accurate the available prior knowledge is, about the motion imposed to the sensor. Calibration approaches employing high-precision mechanical setups can reach high levels of accuracy, as they rely on a very accurate knowledge of the sensor motion [2], [9], [11], [14]. However, such setups are typically cumbersome and very expensive, therefore they are typically adopted for top-grade accelerometers, like those used in seismic or navigation applications. Conversely, for common applications employing commercial-grade MEMS sensors, the cost for such setups would be prohibitive, therefore it is reasonable to follow a somehow opposite approach: a calibration based on a much simpler mechanical setup, providing almost no a-priori knowledge about the sensor motion. In this case, the missing motion information should be determined as well, together with the sensor parameters. A self-calibration approach is therefore

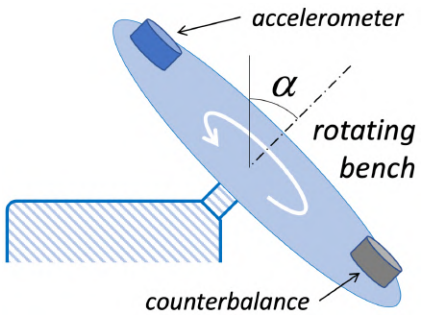

Fig. 1. The calibration setup. The accelerometer to calibrate is mounted on a rotating bench, together with a properly placed counterbalance, to balance the rotation. The bench is tilted with respect to the vertical; the tilt angle α is determined by the calibration procedure.

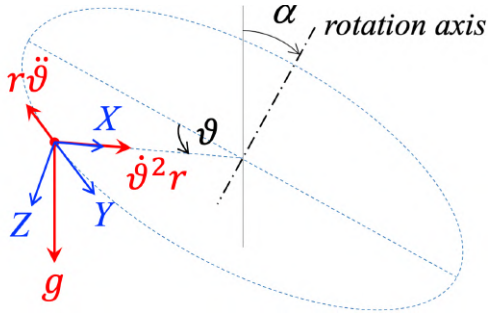

Fig. 2. Geometric description of the bench motion. During rotation, the sensor undergoes three accelerations: the centripetal acceleration $\dot{\vartheta}^2 r$, directed towards the rotation axis, gravity g, along the vertical, and the tangential deceleration $r\ddot{\vartheta}$, due to rotating frictions.

necessary [7], [12], which estimates both the sensor motion and the sensor model.

The aim of this work is to obtain a maximally simple, yet accurate procedure for the calibration of commercial-grade triaxial accelerometers. We propose a calibration procedure which exploits a particularly simple mechanical setup, built with common materials and needing no accurate mechanical constraints on its geometry or motion. The proposed setup is sketched in Fig. 1. It simply consists of a tilted, freely rotating bench carrying the accelerometer. To perform a calibration, the bench is manually put into rotation and then the acceleration data are acquired while the bench rotates. The rotation speed and the tilt angle of the bench do not need to be known, as they are estimated by the calibration procedure. The proposed calibration has been designed for a six-parameters sensor model (offset and gain for each axis), which is well suited for consumer-grade accelerometers, but this method is potentially able to calibrate also more complex sensor models (e.g. models with nonlinear sensor responses).

The article is organized as follows: Section II describes the setup and the sensor/bench model, Section III describes the calibration algorithm, and Section IV presents the performance obtained with synthetically generated data and with experimental data acquired from a triaxial MEMS accelerometer.

II. Calibration Setup and Model

A. Mechanical Setup

A sketch of the proposed calibration setup is shown in Fig. 1. The calibration bench consists of a circular plate mounted on a wheel, which is fastened to a static basement and is free to rotate. The rotation axis is tilted with respect to the vertical. The accelerometer under test is fastened to the bench, preferably near the periphery, to maximize the radius of its circular trajectory. A proper counterbalance is applied on the opposite side and its position is adjusted in order to balance the rotating bench (an unbalanced wheel would lead to undesired periodic fluctuations in the angular velocity on each turn). To perform a calibration, the bench is manually put into rotation and, while the bench freely rotates, acceleration data are acquired from the sensor. The data acquired over several turns of the bench are the input of the calibration algorithm. The rotating bench should have a substantial moment of inertia and preferably low rotating friction, so that once the bench has been set into rotation, its angular velocity decreases slowly enough to consider the angular deceleration very low during the acquisition time.

The bench tilt α and the time-varying angular velocity $\dot{\vartheta}(t)$ are not known; they are estimated by the calibration procedure. The only parameter that needs to be predetermined is the accelerometer rotation radius r, defined as the distance of the sensor from the rotation axis. This distance can be easily measured once the sensor is mounted on the bench; the rotation axis can be easily localized on the top of the wheel hub (as shown in Fig. 9).

B. Modelling Sensor and Motion

The geometric model describing the bench motion is represented in Fig. 2. During the rotation, the acceleration applied to the sensor is the resultant of three components: gravity g ($9.8067\,\mathrm{m/s^2}$ in our lab in Milan, Italy), directed downwards, the centripetal acceleration $r\,\dot{\vartheta}(t)^2$, directed toward the rotation center, and the deceleration $r\,\ddot{\vartheta}(t)$, directed tangentially to the rotation.

Let us consider a local reference frame rigid to the bench, as shown in Fig. 2: let X be the radial axis, lying on the rotation plane and pointing toward the rotation axis, Y be the tangential axis and Z be orthogonal to the rotation plane. The three components of the acceleration vector $\mathbf{a}(t)$ along these axes result to be

$$\mathbf{a}(t) = \begin{cases} a_X(t) = r\,\dot{\vartheta}^2(t) & + g\,\sin\alpha\,\cos\vartheta(t) \\ a_Y(t) = r\,\ddot{\vartheta}(t) & + g\,\sin\alpha\,\sin\vartheta(t) \\ a_Z(t) = & g\,\cos\alpha \end{cases} \quad (1)$$

where α is the bench tilt, $\vartheta(t)$ is the angular position (or *phase*) of the sensor, $\dot{\vartheta}(t)$ is the angular velocity and $\ddot{\vartheta}(t)$ the angular acceleration of the bench.

1) Modelling the bench rotation: Due to the several sources of friction, like air drag on the rotating wheel and frictions in the ball bearings, the angular velocity $\dot{\vartheta}(t)$ slowly decreases during the acquisition. Due to the noise superimposed to the accelerometer's output signal, a data acquisition over several turns is highly desirable to obtain the best-achievable accuracy in calibration. For this reason, the decrease of $\dot{\vartheta}(t)$, although

slow, cannot be neglected. The time-varying angular speed $\dot{\vartheta}(t)$ can be described by taking into account the angular acceleration $\ddot{\vartheta}(t)$ in the model (1).

Experimental free-rotation tests evidenced that the dominant source of friction is the drag force of air on the wheel spokes, generating turbulence while rotating; such force is proportional to the squared velocity of the moving part [15]. For rotating parts, however, this velocity is $\dot{\vartheta}(t)\, r$, therefore the drag force depends also on the rotating radius. Considering also that this force is strongly dependent on the shape of each moving part, we can conclude that it is unfeasible to define an accurate analytical model of the total friction force causing the angular deceleration $\ddot{\vartheta}(t)$.

For this reason, an accurate model describing the time-dependency of the angular velocity has been found experimentally. To this aim, we acquired data from MEMS accelerometers mounted on the bench undergoing free rotation, and derived, from the acquired acceleration, the angular velocity $\dot{\vartheta}(t)$. These data have shown that $\dot{\vartheta}(t)$ fits surprisingly well to a cubic polynomial function of t. We modeled therefore the angular velocity $\dot{\vartheta}(t)$, and consequently the phase $\vartheta(t)$, as follows:

$$\begin{aligned} &\dot{\vartheta}(t) = \omega_0 + \omega_1\, t + \omega_2\, t^2 + \omega_3\, t^3 \\ \rightarrow \quad &\vartheta(t) = \vartheta_0 + \omega_0\, t + \frac{\omega_1}{2}\, t^2 + \frac{\omega_2}{3}\, t^3 + \frac{\omega_3}{4}\, t^4 \end{aligned} \tag{2}$$

Figure 3 shows a typical example of cubic fitting on more than 200 samples of $\dot{\vartheta}(t)$ (each sample being the average speed of a half turn of the bench), derived from the acceleration signal of a MEMS sensor rotating for 90 seconds. As the plots show, the fitting accuracy is very high, by far more than needed for this application: the deviation from the fitting curve is less than $0.018\,\mathrm{rad/s}$, with $\sigma_{\dot{\vartheta}} = 6.12\,\mathrm{mrad/s}$.

According to this model, the five parameters $\langle \vartheta_0, \omega_0 \ldots \omega_3 \rangle$ show to be appropriate to accurately model the phase $\vartheta(t)$ during the bench rotation. The acceleration $\mathbf{a}(t)$ in (1) can be therefore modeled as a function of seven parameters: r, α, and the five motion parameters:

$$\mathbf{a}(t) = f\left(r, \alpha, \vartheta_0, \omega_0 \ldots \omega_3,\ t\right). \tag{3}$$

2) Modelling the sensor misalignment: Aiming to keep the calibration procedure as simple as possible, we did not impose any geometrical constraints on how to fasten the sensor onto the bench. Consequently, the relative orientation of the sensor reference $\langle xyz \rangle$ with respect to the bench reference $\langle XYZ \rangle$ is unknown, as schematized in Fig 4. This sensor misalignment can be modeled as the rotation bringing one reference frame onto the other, or, more precisely, as the 3D rotation matrix $\mathbf{R}$ transforming a vector defined in the bench reference $\langle XYZ \rangle$ into the same vector in the sensor's reference $\langle xyz \rangle$.

Naming $\mathbf{a} = [a_X, a_Y, a_Z]^T$ the *applied* acceleration vector defined in (1) and referred to the bench reference $\langle XYZ \rangle$, and $\mathbf{s} = [s_x, s_y, s_z]^T$ the *sensed* acceleration vector, corresponding to the same vector in the sensor's reference $\langle xyz \rangle$, the sensor misalignment can be modeled as the rotation matrix linking the two reference frames, that is:

$$\begin{bmatrix} s_x \\ s_y \\ s_z \end{bmatrix} = \mathbf{R} \begin{bmatrix} a_X \\ a_Y \\ a_Z \end{bmatrix}, \quad \mathbf{R} = \mathrm{rotmat}\left[\theta, \phi, \psi\right] \tag{4}$$

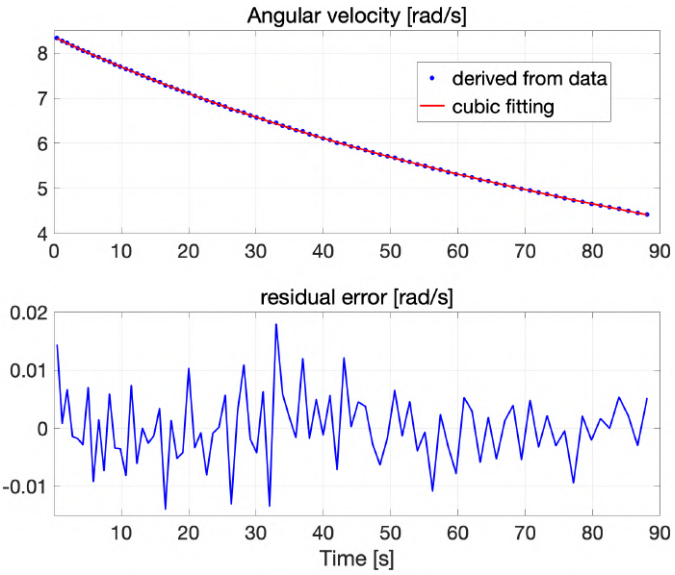

Fig. 3. Cubic fitting (red line) of the samples of angular velocity (blue dots) $\dot{\vartheta}(t)$, for a 90 s-long data acquisition (one of those considered in the experimental tests presented in Sec. IV-B) while the bench is freely rotating. The standard deviation of the residual error (plotted above) is $\sigma[e_{\dot{\vartheta}}] = 6.12 \times 10^{-3}\,\mathrm{rad/s}$.

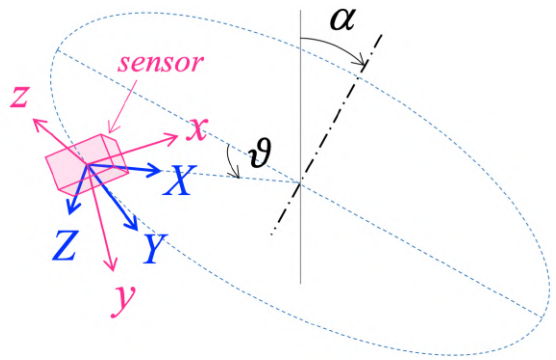

Fig. 4. Schematization of the sensor misalignment. The actual orientation of the sensing reference frame is unknown. This misalignment can be modeled as the 3D rotation that overlaps the bench reference frame $\langle XYZ \rangle$ onto the sensor reference frame $\langle xyz \rangle$.

being $\langle \theta, \phi, \psi \rangle$ the Euler angles defining the *misalignment* rotation matrix $\mathbf{R}$.

3) Modelling the sensor: We consider the classical six-parameters linear model [7], [14] considering the sensor's offsets and gains for each axis:

$$\mathbf{v} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = \begin{bmatrix} o_x \\ o_y \\ o_z \end{bmatrix} + \begin{bmatrix} \gamma_x & 0 & 0 \\ 0 & \gamma_y & 0 \\ 0 & 0 & \gamma_z \end{bmatrix} \begin{bmatrix} s_x \\ s_y \\ s_z \end{bmatrix} \tag{5}$$

where $\mathbf{v} = [v_x, v_y, v_z]$ is the *observed* acceleration generated by the accelerometer, $\mathbf{s} = [s_x, s_y, s_z]^T$ is the sensed acceleration defined in (4), $\mathbf{o} = [o_x, o_y, o_z]^T$ and $\mathbf{\Gamma} = \mathrm{diag}\left[\gamma_x, \gamma_y, \gamma_z\right]$ are the accelerometer *offset* and *gain* along the sensing axes.

The complete model, relating the observed data $\mathbf{v}(t)$ to the applied acceleration $\mathbf{a}(t)$ in (1), is obtained by combining equations (4) and (5):

$$\mathbf{v}(t) = \mathbf{o} + \mathbf{\Gamma}\, \mathbf{s}(t) = \mathbf{o} + \mathbf{\Gamma}\, \mathbf{R}\, \mathbf{a}(t) \tag{6}$$

where offset $\mathbf{o} = [o_x, o_y, o_z]$ and gain $\mathbf{\Gamma} = \text{diag}\,[\gamma_x, \gamma_y, \gamma_z]$ are the actual sensor parameters, the target of the calibration. Since the misalignment matrix $\mathbf{R}$ is defined by the Euler angles $\langle\theta, \phi, \psi\rangle$ and the time behavior of $\mathbf{a}(t)$ is defined by the seven motion parameters in (3), the complete calibration model is a set of 16 parameters, which can be seen as a vector $\mathbf{m}$ in the 16-dimensional model space:

$$\mathbf{v}(\mathbf{m}, t): \quad \mathbf{m} = \{\mathbf{o},\ \mathbf{\Gamma},\ \mathbf{R},\ r,\ \alpha,\ \vartheta_0,\ \omega_0 \ldots \omega_3\} \tag{7}$$

Aim of the calibration is to determine the parameter set $\mathbf{m}$, whereby $\mathbf{R}$, α, r, and the motion parameters in (3) define the bench motion, whereas $\mathbf{o}$ and $\mathbf{\Gamma}$ correspond to the desired sensor calibration.

C. Determinability of the problem

Striving for a very simple calibration procedure has lead us to conceive a calibration setup in which the geometry and the motion of the calibration bench are unknown a-priori. Unfortunately, so many unknowns make the calibration problem defined in (6) and (7) undetermined.

To prove this fact, let us consider the ideal case of "no misalignment", with the accelerometer perfectly aligned to the bench reference. Referring to Fig. 4, this would mean that the references $\langle XYZ\rangle$ and $\langle xyz\rangle$ were coincident, or $\mathbf{R} = \mathbf{I_3}$. The calibration model in (6) would then reduce to $\mathbf{v}(t) = \mathbf{o} + \mathbf{\Gamma}\,\mathbf{a}(t)$. Writing explicitly the three components of $\mathbf{v}(t)$ in this case, we obtain:

$$\begin{aligned} v_x(t) &= o_x + \gamma_x\, r\, \dot{\vartheta}^2(t) + \gamma_x\, g \sin\alpha \cos\vartheta(t) \\ v_y(t) &= o_y + \gamma_y\, r\, \ddot{\vartheta}(t) \ + \gamma_y\, g \sin\alpha \sin\vartheta(t) \\ v_z(t) &= o_z + \gamma_z\, g \cos\alpha \end{aligned} \tag{8}$$

where we observe that $v_z(t)$ is constant. This fact enlightens the indeterminacy of this setup: any change in o_z could be masked by a proper change of the parameters γ_z and α, so that $o_z = -\gamma_z\, g \cos\alpha$, thus leaving $v_z(t)$ unchanged.

To make this point clear, let us consider a scalar change of the parameters α, r, and $\mathbf{\Gamma}$, changing respectively to α', r', and $\mathbf{\Gamma}'$ in such a way that:

$$\begin{aligned} \alpha \quad &\Rightarrow \quad \alpha': \quad \sin\alpha' = \beta \sin\alpha \\ r \quad &\Rightarrow \quad r': \quad r' = \beta\, r \\ \mathbf{\Gamma} \quad &\Rightarrow \quad \mathbf{\Gamma}': \quad \mathbf{\Gamma}' = \frac{1}{\beta}\,\mathbf{\Gamma} \end{aligned} \tag{9}$$

This change would leave $v_x(t)$ and $v_y(t)$ in (8) unchanged, as $\gamma'_x\, r' = \gamma_x\, r$ and $\gamma'_{x,y} \sin\alpha' = \gamma_{x,y} \sin\alpha$. Conversely, $v_z(t)$ would change, as $\gamma'_z\, g \cos\alpha' \neq \gamma_z\, g \cos\alpha$; but this change could be compensated by an opposite variation of o_z. More precisely, considering the latter addend of $v_z(t)$:

$$\begin{aligned} \gamma'_z\, g \cos\alpha' &= \frac{\gamma_z}{\beta}\, g \sqrt{1 - \beta^2 \sin^2\alpha} \\ &= \gamma_z\, g \cos\alpha \sqrt{\frac{\frac{1}{\beta^2} - \sin^2\alpha}{\cos^2\alpha}} \\ &= \delta \cdot \gamma_z\, g \cos\alpha \end{aligned} \tag{10}$$

where

$$\delta = \sqrt{\frac{\frac{1}{\beta^2} - \sin^2\alpha}{1 - \sin^2\alpha}} = \frac{1}{\beta}\sqrt{1 + (1 - \beta^2)\tan^2\alpha}. \tag{11}$$

This means that, with the parameters changing as in (9), the corresponding variation of $v_z(t)$ would be:

$$\Delta v_z = v'_z(t) - v_z(t) = (\delta - 1)\,\gamma_z\, g\, \cos\alpha. \tag{12}$$

Since Δv_z is constant over time, it can be nullified introducing an opposite change of the offset's z-component o_z:

$$\Delta o_z = o'_z - o_z = -\Delta v_z \ \Rightarrow\ v'_z(t) = v_z(t). \tag{13}$$

In conclusion, a variation of the parameters r, α, $\mathbf{\Gamma}$, o_z according to (9) and (13) would cause no change in the acquired data $\mathbf{v}(t)$. In other words, there is a null space in the function $\mathbf{v}(t) = f\,(\mathbf{m})$ in the subspace of the parameters $\langle r, \alpha, \mathbf{\Gamma}, o_z\rangle$.

Even considering now the model (6) in its generality, that is, with a generic misalignment $\mathbf{R}$ between sensor and bench reference frames, the rotation just introduces a linear combination of the components of $\mathbf{a}(t)$ over $\mathbf{v}(t)$ and therefore has no influence on the existence of the null space. In synthesis, the null space is due to the constancy of the acceleration component $a_z(t)$, which cannot be distinguished from a constant offset.

Elimination of the indeterminacy: According to the above discussion, a way to solve for this indeterminacy is to eliminate at least one unknown from the null space $\langle r, \alpha, \mathbf{\Gamma}, o_z\rangle$. This could be achieved, for instance, by predetermining one of these parameters and introduce it in the model (6) as a-priori known. Since $\mathbf{\Gamma}$ and o_z belong to the sensor calibration parameters, it makes no sense to predetermine them. Therefore the choice has to fall on r or α. Having the goal in mind to keep the bench construction and the overall setup as simple as possible, it is reasonable to choose the parameter which is easiest to be measured on the calibration bench. For this setup we chose to predetermine r, easier to be accurately measured than α. In fact, r corresponds to the distance between the sensor (the center of the accelerometer chip) and the rotation axis (the center of the wheel hub); both points can be easily and precisely located on the bench, so their distance can be measured with sufficient accuracy. For different constructions of the calibration bench, however, it might be simpler to measure the angle α instead of r. The proposed algorithm could be simply adapted to work with any of the two parameters as predetermined.

III. The Calibration Algorithm

The proposed algorithm consists of four steps. A schematic flowchart of the algorithm is shown in Fig. 5.

The first step estimates roughly the misalignment of the sensor reference $\langle xyz\rangle$ with respect to the bench reference $\langle XYZ\rangle$ (see Fig. 4), in form of a *re-alignment matrix* $\mathbf{T}$.

The second step estimates the *motion parameters* $\langle\vartheta_0, \omega_0 \ldots \omega_3\rangle$ modeling the sensor rotation $\vartheta(t)$ according to (2). By exploiting the sinusoidal components in the acquired signal, it is possible to estimate $\vartheta(t)$ without any knowledge of the sensor parameters.

The third step exploits the knowledge of the just estimated motion parameters and r (known by measurement), to estimate the sensor misalignment matrix $\mathbf{R}$, the tilt angle α, and the

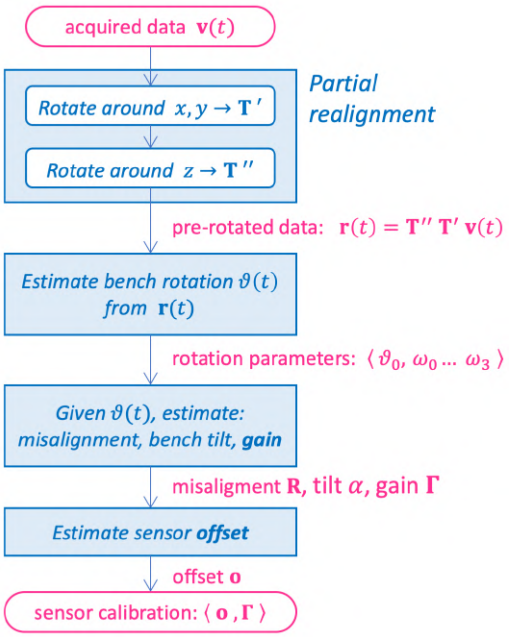

Fig. 5. Flowchart of the proposed calibration. Each of the blocks corresponds to one of the four steps, described in Sections III-A–III-D

sensor gain $\mathbf{\Gamma}$. This estimation is based on the property that the matrix product $\mathbf{W} = \mathbf{T}\mathbf{\Gamma}\mathbf{R}$ (being $\mathbf{T}$ the re-alignment matrix) results to be upper-triangular (explained in Section III-A).

The fourth step estimates the last unknown parameter, the sensor offset $\mathbf{o}$, as the additive constant vector that best fits to the acquired data $\mathbf{v}(t)$. Each of the four steps are described in details in the following subsections.

A. Partial re-alignment

Referring to Fig. 4, the goal of this step is the partial re-alignment of the acceleration data, defined in the sensor reference frame $\langle xyz \rangle$, to the bench reference frame $\langle XYZ \rangle$. This is achieved by applying a 3D rotation $\mathbf{T}$ to the acquired data $\mathbf{v}(t)$, that compensates for the misalignment $\mathbf{R}$ as much as possible. As shown in the following, this re-alignment would correspond to the inverse rotation of the misalignment $\mathbf{R}$ if the gain $\mathbf{\Gamma}$ were isotropic, that is, $\mathbf{\Gamma} = \gamma\,\mathbf{I}_3$.

According to the proposed model (6), the relationship between the acquired data $\mathbf{v}(t)$ and the applied acceleration $\mathbf{a}(t)$ is:

$$\mathbf{v}(t) = \mathbf{o} + \mathbf{\Gamma}\,\mathbf{R}\,\mathbf{a}(t) \tag{14}$$

The application of a rotation $\mathbf{T}$ to the acquired data $\mathbf{v}(t)$ can be expressed as:

$$\mathbf{r}(t) = \mathbf{T}\,\mathbf{v}(t) = \mathbf{T}\,\mathbf{o} + \mathbf{T}\,\mathbf{\Gamma}\,\mathbf{R}\,\mathbf{a}(t) = \mathbf{T}\,\mathbf{o} + \mathbf{W}\,\mathbf{a}(t) \tag{15}$$

with

$$\mathbf{W} = \mathbf{T}\,\mathbf{\Gamma}\,\mathbf{R} \tag{16}$$

where $\mathbf{\Gamma}$ is diagonal, and $\mathbf{T}$ and $\mathbf{R}$ are both rotation matrices.

If the gain were the same on the three axes, then $\mathbf{\Gamma}$ could be written as $\mathbf{\Gamma} = \gamma\,\mathbf{I}_3$, and (16) would become:

$$\mathbf{W} = \gamma\,\mathbf{T}\,\mathbf{R} \tag{17}$$

with $\mathbf{TR}$ also being a rotation matrix (as composition of rotations). In this case, the rotation matrix $\mathbf{T}$, aligning z to Z and making y orthogonal to X, would exactly correspond to the rotation that aligns the sensor reference $\langle xyz \rangle$ to the bench reference $\langle XYZ \rangle$. In other words, $\mathbf{T}$ would be the inverse (and transpose) of $\mathbf{R}$:

$$\mathbf{T} = \mathbf{R}^T \quad \Rightarrow \quad \mathbf{W} = \mathbf{\Gamma} = \gamma\,\mathbf{I}_3; \tag{18}$$

in this case, the re-alignment would exactly compensate for the sensor misalignment $\mathbf{R}$.

However, real triaxial accelerometers are generally anisotropic: the gains along distinct axes are slightly different. The fact that $\mathbf{\Gamma} \neq \gamma\,\mathbf{I}_3$ makes $\mathbf{W}$ no longer orthogonal.

The product $\mathbf{W}\,\mathbf{a}(t)$ in (15) can be seen as a change of reference of the 3D vector $\mathbf{a}$ from its Euclidean reference $\langle XYZ \rangle$ to a new reference $\langle x'y'z' \rangle$, as follows:

$$\mathbf{a}' = \begin{bmatrix} a'_{x'} \\ a'_{y'} \\ a'_{z'} \end{bmatrix} = \mathbf{W}\,\mathbf{a} = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} a_X \\ a_Y \\ a_Z \end{bmatrix} \tag{19}$$

If $\mathbf{W}$ is not orthogonal, $\langle x'y'z' \rangle$ is a non-orthogonal reference frame; in this case, no rotation, applied to the reference $\langle x'y'z' \rangle$, could bring it to overlap onto the orthogonal reference $\langle XYZ \rangle$. This situation is shown in Fig. 6: in order to align $\langle x'y'z' \rangle$ to $\langle XYZ \rangle$ as good as possible, we could just apply a rotation that aligns z' to Z, which makes $z' \perp \langle XY \rangle$, and then a further rotation that causes y' to lie in the YZ plane, thus making $y' \perp X$.

Each coefficient w_{ij} in (19) represents the component of a'_i (where i denotes the i-th axis of $\langle x'y'z' \rangle$) along a_j (being j the j-th axis of $\langle XYZ \rangle$). If the axes of a'_i and a_j are orthogonal, then $w_{ij} = 0$.

Thus, the above described orthogonalities caused by the rotations shown in Fig. 6 (applied through matrix $\mathbf{T}$) lead to corresponding zero-coefficients in $\mathbf{W}$, namely:

$$\begin{array}{lll} z' \perp \langle XY \rangle & \rightarrow & w_{31} = w_{32} = 0 \\ y' \perp X & \rightarrow & w_{21} = 0 \end{array} \tag{20}$$

which makes $\mathbf{W}$ upper-triangular:

$$\mathbf{W} = \mathbf{T}\,\mathbf{\Gamma}\,\mathbf{R} = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ 0 & w_{22} & w_{23} \\ 0 & 0 & w_{33} \end{bmatrix}. \tag{21}$$

The determination of the *re-alignment matrix* $\mathbf{T}$ is the task of this step. $\mathbf{T}$ is calculated as the composition of the two rotations $\mathbf{T}'$ and $\mathbf{T}''$ described in Fig. 6, which are subsequently estimated and then combined to obtain $\mathbf{T} = \mathbf{T}''\,\mathbf{T}'$.

It is noteworthy that $\mathbf{T}$ can be determined with $\mathbf{\Gamma}$ and $\mathbf{R}$ still unknown. The upper-triangularity of $\mathbf{W}$ will be exploited later in the calibration procedure, for the estimation of the misalignment matrix $\mathbf{R}$ (Section III-C).

Estimation of $\mathbf{T}'$: Referring to Fig. 6, the first rotation $\mathbf{T}'$ is the rotation around the axes X and Y that brings z' to be coincident with Z. Applied to the acquired data $\mathbf{v}(t)$, $\mathbf{T}'$ rotates them in such a way to bring their z-component to be parallel to Z. Being the rotated data:

$$\mathbf{r}'(t) = \mathbf{T}'\mathbf{v}(t) = \mathbf{T}'\mathbf{o} + \mathbf{W}'\mathbf{a}(t), \tag{22}$$

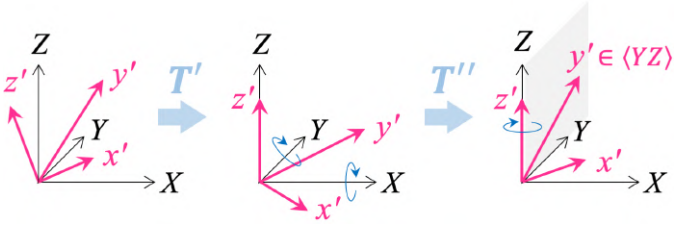

Fig. 6. Schematization of the re-alignment step. The reference frame $\langle \boldsymbol{x'y'z'} \rangle$ is not orthogonal, therefore it cannot be aligned to $\langle \boldsymbol{XYZ} \rangle$ by rotation. However, $\langle \boldsymbol{x'y'z'} \rangle$ can be rotated so that a) $\boldsymbol{z'} \equiv \boldsymbol{Z}$ (rotation $\mathbf{T'}$) and then b) $\boldsymbol{y'} \in \langle \boldsymbol{YZ} \rangle$ (rotation $\mathbf{T''}$). The re-alignment matrix corresponds to the composition of these rotations: $\mathbf{T} = \mathbf{T''T'}$.

where $\mathbf{W'} = \mathbf{T'}\,\boldsymbol{\Gamma}\mathbf{R}$, its z-component is

$$r'_z(t) = \mathbf{t}'_3\mathbf{o} + w'_{31}a_X(t) + w'_{32}a_Y(t) + w'_{33}a_Z(t) \qquad (23)$$

where $\mathbf{t}'_3$ is the third row of $\mathbf{T'}$. As shown in Fig. 6, if z' is parallel to Z, it is necessarily orthogonal to X and Y; this means that $w'_{31} = w'_{32} = 0$ and, according to (23), that the components $a_X(t)$ and $a_Y(t)$ are absent in $r'_z(t)$.

Recalling the time-dependency of $a_X(t)$, $a_Y(t)$, and $a_Z(t)$, defined in (1):

- $a_X(t)$ is a sinusoid plus a sixth-degree polynomial of t, due to the cubic fitting of $\dot{\vartheta}(t)$ in (2);
- $a_Y(t)$ is essentially a zero-mean sinusoid (the tangential acceleration $r\,\ddot{\vartheta}(t)$, almost negligible in the acquired data, can be neglected at this step);
- $a_Z(t)$ is constant,

we can exploit the fact that $a_Z(t)$ is constant whereas $a_X(t)$ and $a_Y(t)$ are both time-varying: the constancy of $r'_z(t)$ would mean that both $a_X(t)$ and $a_Y(t)$ are absent, therefore z' is orthogonal to X and Y. Consequently, $\mathbf{T'}$ can be determined as the rotation matrix that makes $r'_z(t)$ constant.

In practice, considering that real data are noisy, $\mathbf{T'}$ is found as the rotation that minimizes the variance of $r'_z(t)$. Being $\mathbf{T'}$ a rotation around X and Y, the problem can be solved as a minimum search in the space of its Euler angles $\langle \varepsilon_x, \varepsilon_y \rangle$:

$$\mathbf{T'}(\hat{\varepsilon}_x, \hat{\varepsilon}_y) = \underset{\varepsilon_x, \varepsilon_y}{\arg\min}\ \sigma^2\left[r'_z(t)\right]. \qquad (24)$$

Estimation of $\mathbf{T''}$*:* The second rotation $\mathbf{T''}$ is applied around the Z axis (now coincident with z), in such a way to make the y axis orthogonal to the radial axis X (and therefore lying in the tangential plane YZ, as shown in Fig. 6). Considering now the completely rotated data $\mathbf{r}(t) = \mathbf{T}\,\mathbf{v}(t)$ defined in (15), where $\mathbf{T} = \mathbf{T''T'}$, let us focus on the y component, $r_y(t)$:

$$r_y(t) = \mathbf{t}_2\mathbf{o} + w_{21}\,a_X(t) + w_{22}\,a_Y(t) + w_{23}\,a_Z(t) \qquad (25)$$

where $\mathbf{t}_2$ is the second row of $\mathbf{T}$. The orthogonality between y and X would correspond to the absence of the component $a_X(t)$ in $r_y(t)$, meaning that $w_{21} = 0$. A consequence of this condition would be the absence, in $r_y(t)$, of the sixth-degree polynomial component $r\,\dot{\vartheta}^2(t)$ contained in $a_X(t)$.

The component $r\,\dot{\vartheta}^2(t)$ makes $a_X(t)$ monotonically decreasing, whereas $a_Y(t)$ is basically a constant-amplitude sinusoid (the tangential deceleration $r\ddot{\vartheta}(t)$ can be neglected here), and $a_Z(t)$ is constant. Hence, the presence of a contribution proportional to $a_X(t)$ necessarily widens the *excursion* of $r_y(t)$, which can be simply estimated as $\max\left[r_y(t)\right] - \min\left[r_y(t)\right]$. As a consequence, $w_{21} = 0$ in (25) corresponds to the condition of minimum excursion for $r_y(t)$.

We can exploit this condition to determine $\mathbf{T''}$: we search for the matrix defining the rotation around Z which minimizes the excursion of $r_y(t)$. Also this problem can be solved as a minimum search, finding the optimum Euler angle $\hat{\varepsilon}_z$:

$$\mathbf{T''}(\hat{\varepsilon}_z) = \underset{\varepsilon_z}{\arg\min}\ \left\{\max\left[r_y(t)\right] - \min\left[r_y(t)\right]\right\}. \qquad (26)$$

Since $\mathbf{T''}$ is a rotation around Z, its third row $\mathbf{t}''_3 = [0\ 0\ 1]$. Consequently, since $\mathbf{W} = \mathbf{T''W'}$, the third rows of $\mathbf{W}$ and $\mathbf{W'}$ result to be equal, therefore $w_{31} = w'_{31} = 0$ and $w_{32} = w'_{32} = 0$, which, together with $w_{21} = 0$ imposed in determining $\mathbf{T''}$, makes $\mathbf{W}$ upper-triangular, as desired.

B. Estimation of the bench rotation

Considering again the y component of the rotated data $r_y(t)$ expressed in (25), and considering that $w_{21} = 0$ after the re-alignment step (meaning that $r_y(t)$ is orthogonal to the radial axis X, as shown in Fig. 6), $r_y(t)$ results to be:

$$r_y(t) = \mathbf{t}_2\mathbf{o} + w_{22}\,g\sin\alpha\sin\vartheta(t) + w_{23}\,g\cos\alpha, \qquad (27)$$

which can be synthetically expressed as a constant plus a sinusoid with time-varying pulsation, as follows:

$$r_y(t) = k + A\sin\vartheta(t)\ :\quad \begin{array}{l} A = w_{22}\,g\sin\alpha \\ k\ = \mathbf{t}_2\mathbf{o} + w_{23}\,g\cos\alpha \end{array} \qquad (28)$$

where the phase $\vartheta(t)$ is modeled as a fourth-degree polynomial of t, as expressed in (2), by means of the five *motion parameters* $\langle \vartheta_0, \omega_0 \ldots \omega_3 \rangle$.

Consequently, it is possible to determine the motion parameters, together with the parameters $\langle \hat{A}, \hat{k} \rangle$, as the values that, applied in (2) and (28) respectively, yield the best approximation of $r_y(t)$, defined as that giving the least-squares difference between the rotated data $r_y(t)$ and the modeled data provided by (28). The estimation can be formalized as a minimization problem defined on the seven-dimensional search space of the parameters:

$$\begin{aligned} &\langle \hat{\vartheta}_0, \hat{\omega}_0 \ldots \hat{\omega}_3, \hat{A}, \hat{k} \rangle = \\ &= \underset{\vartheta_0, \omega_0 \ldots \omega_3, A, k}{\arg\min}\ \| r_y(t) - [k + A\sin\vartheta(t)] \|^2. \end{aligned} \qquad (29)$$

where $\vartheta(t)$ is obtained using the motion parameters in (2).

C. Estimation of misalignment, tilt, and gain

Once the bench rotation $\vartheta(t)$ is known, the following step performs the combined estimation of the sensor misalignment $\mathbf{R}$, the tilt angle α and the gain $\boldsymbol{\Gamma}$. Referring to (15) and (16), we exploit now the constraint that $\mathbf{W}$ is upper-triangular, considering the following coefficients of $\mathbf{W}$:

$$\begin{cases} w_{21} = t_{21}\gamma_x r_{11} + t_{22}\gamma_y r_{21} + t_{23}\gamma_z r_{31} = 0 \\ w_{31} = t_{31}\gamma_x r_{11} + t_{32}\gamma_y r_{21} + t_{33}\gamma_z r_{31} = 0 \\ w_{32} = t_{31}\gamma_x r_{12} + t_{32}\gamma_y r_{22} + t_{33}\gamma_z r_{32} = 0 \\ w_{22} = t_{21}\gamma_x r_{12} + t_{22}\gamma_y r_{22} + t_{23}\gamma_z r_{32} = \overline{w_{22}} \end{cases} \qquad (30)$$

where t_{ij} and r_{ij} are the coefficients of $\mathbf{T}$ and $\mathbf{R}$, $\langle \gamma_x, \gamma_y, \gamma_z \rangle = \mathrm{diag}(\boldsymbol{\Gamma})$ are the components of the sensor gain,

and $\overline{w_{22}}$ is an estimation of w_{22}: since $r_y(t)$, according to (28), is a sinusoid of amplitude $A = w_{22}\, g \sin\alpha$, its standard deviation $\sigma[r_y(t)] = A/\sqrt{2}$. Thus, we can estimate w_{22} as:

$$\overline{w_{22}} = \frac{A}{g\, \sin\alpha} = \frac{\sqrt{2}\ \sigma[r_y(t)]}{g\, \sin\alpha}; \tag{31}$$

as $\sigma[r_y(t)]$ is computed over the whole acquisition period, the estimation $\overline{w_{22}}$ is quite robust to the noise affecting $r_y(t)$.

The four equations in (30) form a non-homogeneous overdetermined linear system, with the gain components $\boldsymbol{\gamma} = \langle \gamma_x, \gamma_y, \gamma_z \rangle$ as unknowns. This system can be exploited to compute, for any given value of $\mathbf{R}$ and α ($\mathbf{T}$ is known), the corresponding gain $\boldsymbol{\gamma}$. Since the system is overdetermined, it can be solved in closed form, in the least-square sense, by solving the associated pseudo-inverse system [16]: expressing (30) in matrix form, naming $\mathbf{U}$ the coefficient matrix and $\mathbf{b} = [0\ 0\ 0\ \overline{w_{22}}]^T$, $\boldsymbol{\gamma}$ can be derived as:

$$\mathbf{U}\,\boldsymbol{\gamma} = \mathbf{b} \quad \Rightarrow \quad \boldsymbol{\gamma} = \left(\mathbf{U}^T\mathbf{U}\right)^{-1}\mathbf{U}\,\mathbf{b}. \tag{32}$$

Hence, the combined estimation of $\mathbf{R}$, α and $\boldsymbol{\Gamma}$ can be carried out as a nonlinear minimum search in the four-dimensional space $\langle \mathbf{R}(\theta,\phi,\psi), \alpha \rangle$ of the three Euler angles plus the tilt angle α. In the search, for each candidate $\langle \mathbf{R}, \alpha \rangle$ the corresponding gain $\boldsymbol{\gamma}$ is computed using (32). The objective function is the norm of the difference between the acquired and realigned data $\mathbf{r}(t) = \mathbf{T}\mathbf{v}(t)$ and the corresponding modeled data, $\mathbf{r}^m(t)$, after having subtracted from both their own mean value, thus zeroing their means:

$$\begin{aligned} \mathbf{r}_0(t) &= \mathbf{r}(t) - \text{mean}\left[\mathbf{r}(t)\right] \\ \mathbf{r}_0^m(t) &= \mathbf{r}^m(t) - \text{mean}\left[\mathbf{r}^m(t)\right] \end{aligned} \tag{33}$$

The minimization problem can be therefore expressed as:

$$\langle \hat{\mathbf{R}}, \hat{\alpha} \rangle = \underset{\mathbf{R},\,\alpha}{\arg\min} \left\| \mathbf{r}_0(t) - \mathbf{r}_0^m(t) \right\|^2. \tag{34}$$

Using the zero-mean components $\mathbf{r}_0(t)$ and $\mathbf{r}_0^m(t)$ is necessary to make the objective function in (34) independent from the offset $\mathbf{o}$, which is still unknown. In fact, as expressed in (15), the offset $\mathbf{o}$ in $\mathbf{r}(t)$ is just an additive constant. In this way, the estimation of $\mathbf{R}$, α, and $\boldsymbol{\Gamma}$ is decoupled from the estimation of $\mathbf{o}$, thus simplifying the search for the minimum, due to the reduced dimension of the search space.

To obtain $\mathbf{r}^m(t)$, we compute $\mathbf{v}^m(t)$ according to the model (6) using an arbitrary value for $\mathbf{o}$ (its value is irrelevant, since $\mathbf{r}_0^m(t)$ is forced to zero mean), and using the measured value of the radius r (as described in Section II-C). From now on, we consider the complete model (1) for $\mathbf{a}(t)$, no longer neglecting the tangential acceleration component $r\ddot{\vartheta}(t)$. Once $\mathbf{v}^m(t)$ is computed, then $\mathbf{r}^m(t) = \mathbf{T}\,\mathbf{v}^m(t)$.

D. Estimation of the offset

The estimation of the last unknown parameter, the offset $\mathbf{o}$, can be carried out as a minimum search in the three-dimensional space of its components $\langle o_x, o_y, o_z \rangle$, minimizing the norm of the difference between the originally acquired data $\mathbf{v}(t)$ and the synthesized data $\mathbf{v}^m(t)$, computed according to (6) using the already determined parameters. The minimization problem can be then formalized as:

$$\langle \hat{\mathbf{o}} \rangle = \underset{\mathbf{o}}{\arg\min} \left\| \mathbf{v}(t) - \mathbf{v}^m(t) \right\|^2. \tag{35}$$

All the search problems defined in this Section can be solved using a nonlinear minimum search algorithm like [17].

IV. Experimental Results

Two kinds of experiments have been carried out to assess the capabilities of the proposed calibration. To evaluate the robustness of the estimated parameters against noisy data, the calibration has been tested on synthetically generated data, for which the exact value of the model parameters is a-priori known. Then, to verify the practical usability of the proposed calibration procedure and to evaluate the actual precision of the estimations, the procedure has been then tested experimentally, calibrating a common triaxial MEMS accelerometer.

A. Accuracy evaluation with synthetic data

Aim of this test is to evaluate the robustness of the calibration algorithm and the accuracy of its estimations, in presence of noisy acceleration data. This has been achieved by running the calibration repeated times, each time with a different realization of noise added to the same synthetic data, and then analyzing the dispersion of the resulting calibration parameters.

Each realization of the synthetic data has been obtained by adding zero-mean Gaussian noise (simulating the various sources of noise affecting real accelerometers) to the 'ideal' acceleration data computed according to the bench/sensor model (6). Different levels of Gaussian noise have been considered, with standard deviations distributed logarithmically in the range $0.01\,\text{m/s}^2 \div 1\,\text{m/s}^2$. This range includes the typical noise levels of consumer-grade MEMS accelerometers. For each considered noise level, we repeated the calibration for 60 realizations of random noise, to get acceptable statistical confidence. For each noise level, we obtain therefore 60 estimations of the same calibration parameters. The mean and standard deviation of these estimations are good indicators of the algorithm capabilities: the mean values evidence the presence of possible biases in the estimations and give an idea of the numerical robustness of the algorithm; the standard deviations provide a good estimate of the achievable precision on each parameter, as a function of the noise level.

Table I synthesizes the obtained results. The table reports the mean values of the estimated parameter and their standard deviations, for two noise levels in the typical range of MEMS accelerometers. The shown results evidence that, in spite of the simplicity of the proposed setup, this calibration is able to reach high levels of accuracy (closeness between estimated and true values) and precision (low standard deviations).

The mean values of the estimated parameters, as *normalized deviation* from their true value, are plotted in Fig. 7, as a function of the standard deviation of the added noise. The normalized deviation is defined as difference between mean and true values v_{true}, divided by the true value and expressed

TABLE I
RESULTS OF CALIBRATIONS WITH SYNTHETIC DATA: MEAN AND STANDARD DEVIATION OF THE ESTIMATED PARAMETERS, FOR TWO DIFFERENT NOISE LEVELS

PARAMETER		TRUE	ESTIMATED ($\pm\sigma$) $\sigma_n = 0.08\ \mathrm{m/s^2}$	ESTIMATED ($\pm\sigma$) $\sigma_n = 0.2\ \mathrm{m/s^2}$
rotation prm.	ω_3	2	$2.000 \pm .002$	$2.000 \pm .004$
	ω_2	500	500.0 ± 0.3	499.9 ± 0.5
	ω_1	-90	-90.00 ± 0.01	-90.00 ± 0.02
	ω_0	9	$9.000 \pm .0001$	$9.000 \pm .0002$
	ϑ_0	30°	$29.99° \pm 0.03°$	$29.94° \pm 0.07°$
bench tilt	α	45°	$45.00° \pm 0.01°$	$45.01° \pm 0.03°$
misalignment	θ	30°	$29.99° \pm 0.03°$	$29.93° \pm 0.07°$
	ϕ	20°	$19.99° \pm 0.02°$	$19.97° \pm 0.03°$
	ψ	25°	$25.00° \pm 0.08°$	$25.04° \pm 0.16°$
Gain	γ_x	0.9	$0.8999 \pm .0001$	$0.8998 \pm .0003$
	γ_y	1.0	$1.0000 \pm .0004$	$0.9999 \pm .0009$
	γ_z	1.1	$1.1005 \pm .0007$	$1.1026 \pm .0017$
Offset	o_x	1	1.001 ± 0.005	1.006 ± 0.010
	o_y	-1	$-1.001 \pm .014$	$-1.008 \pm .027$
	o_z	1	0.996 ± 0.009	1.006 ± 0.019

UNITS: ω_3: $\mu\mathrm{rad/s^4}$, ω_2: $\mu\mathrm{rad/s^3}$, ω_1: $\mathrm{mrad/s^2}$, ω_0: $\mathrm{rad/s}$, Offset: $\mathrm{m/s^2}$

as percentage: $\mu_{dev} = 100\,(\mu_{cal}/v_{true} - 1)$. For homogeneity of representation, the deviations of similar parameters have been aggregated by averaging; the plotted lines represent, respectively:

- offset (average of x, y, z components);
- gain (average of x, y, z components);
- bench tilt angle;
- misalignment (average of the three Euler angles);
- motions parameters (average of the five parameters).

The diagram evidences that, for low noise levels, $\mu_{dev} \approx 0$, meaning that $\mu_{cal} = v_{true}$, for all parameters. This proves that the estimation is unbiased. The estimation remains essentially unbiased ($|\mu_{dev}| < 0.25\%$) up to noise levels of $\sigma_n = 0.2\,\mathrm{m/s^2}$, significantly more than the typical noise levels of consumer-grade low-g accelerometers [18], [19].

Fig. 8 shows the standard deviations of the estimated parameters, normalized to their mean value[1], as a function of the standard deviation of the data noise, with homogeneous variables aggregated by averaging as in Fig. 7. From this diagram we can derive the following:

- the standard deviation of all parameters grows approximately with the same rate as the data noise over the whole considered noise range, with no tendency to diverge. This proves the robustness of the estimation even with noise levels well over those of common MEMS accelerometers.
- The estimation of the motion parameters is characterized by the lowest deviation. This fact supports the proposed idea of a very simple calibration setup, with no motion control: the motion of the calibration bench is estimated with enough precision to be considered exactly known, as if a high-precision rotating bench were used.
- Offset and gain, the actual aim of calibration, are estimated with substantially different precision, with $\sigma_{\mathbf{o}} \approx 20\,\sigma_{\boldsymbol{\gamma}}$ over the whole noise range; the offset results to be the parameter estimated with the worst precision.

[1]for gain and offset we used $\mu_{\mathbf{o}} = \mu_{\boldsymbol{\gamma}} = 1$, therefore their normalized deviations in Fig. 8 are numerically coincident with the standard deviations.

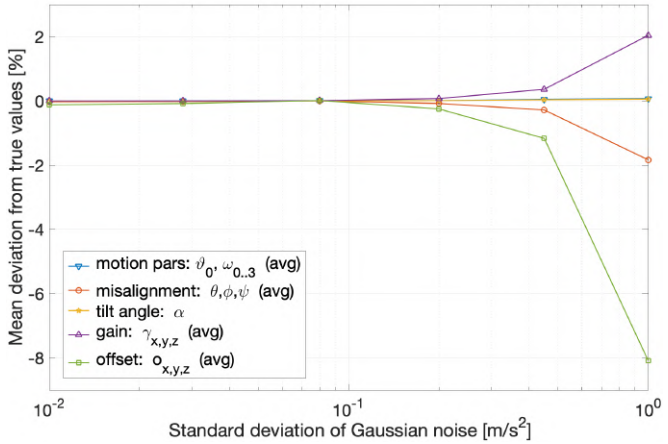

Fig. 7. Mean value of the estimated parameters, expressed as normalized difference from the true value $\boldsymbol{v_{true}}$, for different levels (standard deviation) of the Gaussian noise added to the ideal acceleration data.

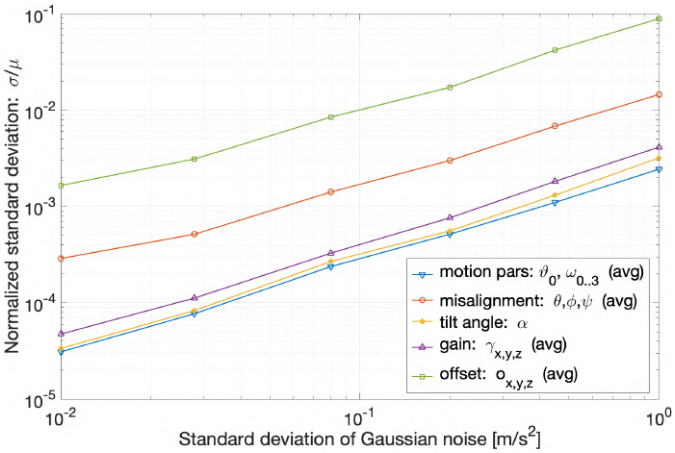

Fig. 8. Standard deviation of the estimated parameters, normalized to their mean value ($\boldsymbol{\sigma/\mu}$), for different levels (standard deviation) of the Gaussian noise added to the ideal acceleration data.

The poorer accuracy in the estimation of the offset is reasonably due to the kinematics of the proposed setup: depending on sensor misalignment and bench tilt, one or more sensing axes acquire a unipolar acceleration signal, whereas it is a bipolar signal that makes offset evident. Notwithstanding, the achieved offset accuracy is more than satisfying: for noise levels up to $\sigma_n = 0.1\,\mathrm{m/s^2}$, the normalized standard deviation is less than 10^{-2}, corresponding to an accuracy in the offset estimation better than 1%. Considering that the noise levels of consumer-grade MEMS accelerometers lies below $0.1\,\mathrm{m/s^2}$ [18], [19], the achievable accuracy is significantly high, despite the simplicity of the proposed setup.

B. Performance evaluation on a real sensor

Although the tests carried out with synthetic data give a valid indication of the potential accuracy of this method, the suitability of the proposed model for the calibration of a real accelerometer can be only verified by performing real experiments on that sensor.

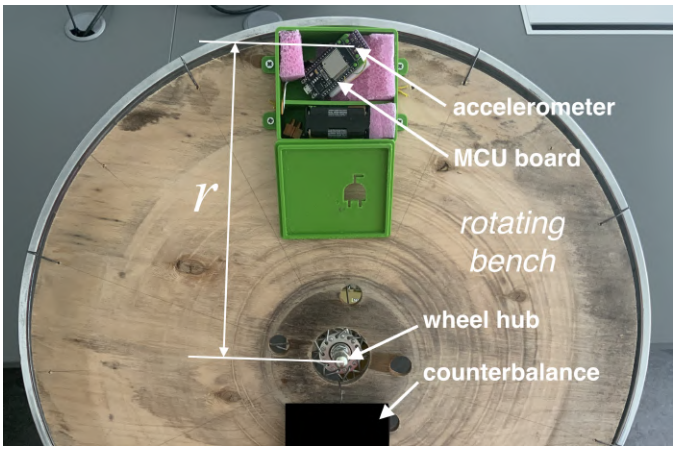

Fig. 9. Prototype of the calibration setup. The sensor is deliberately mounted in a tilted position with respect to the bench, to obtain a significant misalignment with respect to the bench reference. This guarantees the presence of all acceleration contributes (gravity, centripetal acceleration, angular deceleration) along all three sensing axes.

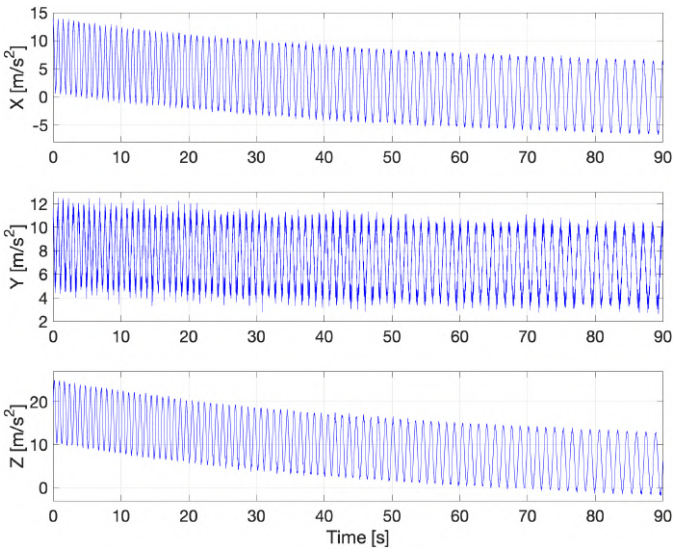

Fig. 10. One of the acquisitions of acceleration data used for the calibration tests with a real MEMS sensor (Invensense™ MPU-9250 [18]).

The most popular calibration methods in literature assess their accuracy by comparing the fitness of the sensor output to a known model, like the closeness of the absolute value to g in static conditions, as in [4], [5], [10]–[12], or the fitness to other known model constraints, as in [2], [3], [13], before and after the calibration. This approach only proves that the calibrated acceleration signal is more accurate than the uncalibrated output, but no information is provided about the actual precision of the estimated calibration parameters.

Conversely, we decided to evaluate the performance following a metrological approach, by estimating the precision of the calibrated parameters through repeated measurements [20]. We carried out repeated calibrations of the same MEMS sensor, obtaining each time a measurement sample of the calibrated parameters. The statistical analysis (mean and standard deviation) of the set of samples provides a reasonable estimation of the achieved precision for each parameter.

We built a very-low-cost calibration setup (we used a common bicycle wheel as base of the rotating bench) as schematized in Fig. 1. An image of the calibration bench we built is shown in Fig. 9. We designed and built a wireless device, composed of the sensor to calibrate, a MEMS digital accelerometer (Invensense™ MPU9250 [18]), which was connected to a battery-operated MCU board (ESP32-Wroom [21]). We exploited the on-board provided Bluetooth interface for wireless data acquisition and remote device control.

The setup was arranged as follows: the rotating bench was fixed to the basement with an arbitrary tilt angle; the sensing device was mounted on the calibration bench, and the position of the counterbalance was correspondingly adjusted to balance the wheel. The rotation radius r was then measured as the distance between the sensor and the center of the wheel hub (see Fig. 9). To acquire the calibration data, the wheel was spun by hand and the sensor data were collected for several turns of the bench.

In the presented experiments we measured a rotation radius $r = 280\,\mathrm{mm}$. The bench was tilted of about $45°$. The sensor was oriented on purpose with a significant misalignment with respect to the bench's reference, in order to better distribute the acceleration components on the three sensing axes. Data were acquired at a sampling rate of $200\,\mathrm{Hz}$ for approximately 2 minutes; from this acquisition, a $90\,\mathrm{s}$-long sequence (approximately corresponding to 100 wheel turns) was extracted for calibration. Fig. 10 shows the signals acquired in one of these sessions.

TABLE II
RESULTS OF REPEATED CALIBRATIONS OF A MEMS ACCELEROMETER: MEAN AND STANDARD DEVIATION OF THE CALIBRATED PARAMETERS.

PARAMETER		mean (μ)	std. deviation (σ)
Bench tilt	α	$46.111°$	$0.154°$
Misalignment	θ	$24.760°$	$0.088°$
	ϕ	$7.311°$	$0.089°$
	ψ	$31.974°$	$0.214°$
Gain	γ_x	1.00952	0.00072
	γ_y	0.99420	0.00143
	γ_z	1.02752	0.00316
Offset $[\mathrm{m/s^2}]$	o_x	0.1413	0.0307
	o_y	-0.0679	0.0371
	o_z	-0.3186	0.0312

This calibration process has been repeated 10 times with the same bench geometry (but, of course, each time with different motion, due to the manual bench spinning). Table II reports the mean value and the standard deviation for each calibrated parameter.

To assess the ability of this calibration to reach the best possible accuracy, it is reasonable to compare the experimental standard deviations reported in Table II to the corresponding deviations obtained with synthetic data with the same noise level. Indeed, since the synthetic data are generated by adding Gaussian noise to *ideal* acceleration data, the standard deviations in Fig. 8 can be considered the minimum achievable deviation, for each noise level. These deviations can be then

reasonably taken as reference for the best achievable precision.

Information about the level of noise affecting the output of real accelerometers is generally provided by the manufacturer: for the sensor used in this experiment, the standard deviation of the acceleration noise, as reported in its data sheet [18], is $\sigma_n \approx 0.08\,\mathrm{m/s^2}$. For such a noise level, the plots in Fig. 8 indicate, for gain and offset (the actual parameters of interest in calibration), the following best achievable standard deviations:

$$\sigma_{\text{gain}}^{\text{best}} \approx 2 \times 10^{-3}, \quad \sigma_{\text{offset}}^{\text{best}} \approx 9 \times 10^{-3}. \tag{36}$$

Compared to them, the standard deviations on the parameters estimated with real data, obtained by averaging the three components of offset and gain in Table II, are

$$\sigma_{\text{gain}}^{\text{meas}} = 1.8 \times 10^{-3}, \quad \sigma_{\text{offset}}^{\text{meas}} = 3.3 \times 10^{-2}. \tag{37}$$

The measured deviations indicate optimal precision for the gain, whose deviation is even lower than the expected value, whereas, for the offset, the deviation is ≈ 3.5 times larger than the expected one, indicating a slightly sub-optimal precision. Nonetheless, the essential proximity of the experimental deviations to the best achievable ones is a proof of the accurate conformity of the proposed model (6) to the real calibration setup and, therefore, of the reliability of the calibration results yielded by this procedure.

V. Conclusion

We propose a dynamic calibration method for three-axes accelerometers working with a very simple and inexpensive setup. Testing the calibration on both synthetic and real data has shown that a) the method provides satisfying accuracy, and b) the accuracy varies essentially with the same rate as the noise level on the data. The latter fact encourages the author to further investigate this approach, considering that more accurate sensors would allow to adopt more accurate models (e.g. considering nonlinearities, nonorthogonal axes, or hysteresis), whose estimation is in any case unfeasible using static calibration approaches.

Acknowledgment

I would like to thank Luca Marangi for his work. For his B.S. project, Luca worked at the construction of the calibration bench and at the firmware/software for sensor data acquisition.

References

[1] S. Zihajehzadeh, D. Loh, T. J. Lee, R. Hoskinson, and E. J. Park, "A cascaded kalman filter-based gps/mems-imu integration for sports applications," *Measurement*, vol. 73, pp. 200–210, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0263224115002821

[2] D. Lee, S. Lee, S. Park, and S. Ko, "Test and error parameter estimation for mems based low cost imu calibration," *International Journal of Precision Engineering and Manufacturing*, vol. 12, no. 4, pp. 597–603, 2011.

[3] T. Nieminen, J. Kangas, S. Suuriniemi, and L. Kettunen, "An enhanced multi-position calibration method for consumer-grade inertial measurement units applied and tested," *Measurement Science and Technology*, vol. 21, no. 10, pp. 105 204 (1–11), 2010.

[4] O. Srkk, T. Nieminen, S. Suuriniemi, and L. Kettunen, "A multi-position calibration method for consumer-grade accelerometers, gyroscopes, and magnetometers to field conditions," *IEEE Sensors Journal*, vol. 17, no. 11, pp. 3470–3481, 2017.

[5] I. Frosio, F. Pedersini, and N. A. Borghese, "Autocalibration of mems accelerometers," *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 6, pp. 2034–2041, 2009.

[6] W. T. Ang, P. K. Khosla, and C. N. Riviere, "Nonlinear regression model of a low-*g* mems accelerometer," *IEEE Sensors Journal*, vol. 7, no. 1, pp. 81–88, 2007.

[7] I. Frosio, F. Pedersini, and N. A. Borghese, "Autocalibration of triaxial mems accelerometers with automatic sensor model selection," *IEEE Sensors Journal*, vol. 12, no. 6, pp. 2100–2108, 2012.

[8] Chin-Woo Tan and Sungsu Park, "Design of accelerometer-based inertial navigation systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 54, no. 6, pp. 2520–2530, 2005.

[9] M. Zhang, S. Yan, Z. Deng, P. Chen, Z. Li, J. Fan, H. Liu, J. Liu, and L. Tu, "Cross-coupling coefficient estimation of a nano-g accelerometer by continuous rotation modulation on a tilted rate table," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2021.

[10] J. A. Garca, E. Lara, and L. Aguilar, "A low-cost calibration method for low-cost mems accelerometers based on 3d printing," *Sensors*, vol. 20, no. 22, 2020.

[11] S.-h. P. Won and F. Golnaraghi, "A triaxial accelerometer calibration method using a mathematical model," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 8, pp. 2144–2153, 2010.

[12] M. Glueck, D. Oshinubi, P. Schopp, and Y. Manoli, "Real-time autocalibration of mems accelerometers," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 1, pp. 96–105, 2014.

[13] A. A. Asadi, M. R. Homaeinezhad, S. Arefnezhad, A. Safaeifar, and M. Zoghi, "Designing a 2-dof passive mechanism for dynamical calibration of mems-based motion sensors," in *2014 Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM)*, 2014, pp. 256–261.

[14] R. Zhang, F. Hoflinger, and L. M. Reind, "Calibration of an imu using 3-d rotation platform," *IEEE Sensors Journal*, vol. 14, no. 6, pp. 1778–1787, 2014.

[15] Y. A. Cengel and J. M. Cimbala, *Fluid Mechanics: Fundamentals and Applications*, 3rd ed. McGraw-Hill Higher Education, 2013.

[16] R. Penrose, "On best approximate solution of linear matrix equations," *Proceedings of the Cambridge Philosophical Society*, vol. 52, no. 1, pp. 17–19, 1956.

[17] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence properties of the nelder-mead simplex method in low dimensions," *SIAM Journal of Optimization*, vol. 9, no. 1, pp. 112–147, 1998.

[18] "MPU-9250 Product Specification," Invensense Inc., 2016, PS-MPU-9250A-01, Rev. 1.1. [Online]. Available: https://invensense.tdk.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf

[19] "ADXL317 Data Sheet," Analog Devices, 2019, Rev. 0. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ADXL317.pdf

[20] F. Pavese, "Replicated observations in metrology and testing: modelling repeated and non-repeated measurements," *Accreditation and Quality Assurance*, vol. 12, no. 10, pp. 525–534, 2007.

[21] "Esp32 wroom32ue datasheet," Espressif Systems Co., Ltd., 2021, version 1.2. [Online]. Available: https://www.espressif.com/

Federico Pedersini received the M.S. in Electronic Engineering (Summa cum Laude) in 1991 and the Ph.D. in Electronics and Communication Engineering in 1994 from Politecnico di Milano, Italy. Since 2005 he is Associate Professor at the University of Milan, Department of Computer Science. His main research interests include: digital signal/image processing, 3D reconstruction from images, high-performance computing on special architectures, sensors and sensor systems.