

# A Robust and Easy to Implement Method for IMU Calibration without External Equipments

David Tedaldi, Alberto Pretto and Emanuele Menegatti

**Abstract**—Motion sensors as inertial measurement units (IMU) are widely used in robotics, for instance in the navigation and mapping tasks. Nowadays, many low cost micro electro mechanical systems (MEMS) based IMU are available off the shelf, while smartphones and similar devices are almost always equipped with low-cost embedded IMU sensors. Nevertheless, low cost IMUs are affected by systematic error given by imprecise scaling factors and axes misalignments that decrease accuracy in the position and attitudes estimation.

In this paper, we propose a robust and easy to implement method to calibrate an IMU without any external equipment. The procedure is based on a multi-position scheme, providing scale and misalignments factors for both the accelerometers and gyroscopes triads, while estimating the sensor biases. Our method only requires the sensor to be moved by hand and placed in a set of different, static positions (attitudes). We describe a robust and quick calibration protocol that exploits an effective parameterless *static filter* to reliably detect the static intervals in the sensor measurements, where we assume local stability of the gravity's magnitude and stable temperature. We first calibrate the accelerometers triad taking measurement samples in the static intervals. We then exploit these results to calibrate the gyroscopes, employing a robust numerical integration technique.

The performances of the proposed calibration technique has been successfully evaluated via extensive simulations and real experiments with a commercial IMU provided with a calibration certificate as reference data.

## I. INTRODUCTION

IMUs (Inertial Measurement Units) are very popular sensors in robotics: among others, they have been exploited for inertial-only navigation [1], attitude estimation [2], and visual-inertial navigation [3], [4], also using a smartphone device [5]. IMUs used in robotics are usually based on MEMS (micro electro mechanical systems) technology. They are composed by a set of tri-axial clusters: an accelerometers, a gyros and often a magnetometer cluster. In an ideal IMU, the tri-axial clusters should share the same 3D orthogonal sensitivity axes that span a three dimensional space, while the scale factor should convert the digital quantity measured by each sensor into the real physical quantity (e.g, accelerations and gyro rates). Unfortunately, low cost MEMS based IMU are usually affected by non accurate scaling, sensor axis misalignments, cross-axis

This research has been partially supported by Consorzio Ethics with the grant "Rehabilitation Robotics", by Univ. of Padova with the grant "TIDY-UP: Enhanced Visual Exploration for Robot Navigation and Object Recognition", and by the European Commission under FP7-600890-ROVINA. The authors are with the Department of Information Engineering, University of Padova, Italy. Email: david.tedaldi@gmail.com, emg@dei.unipd.it. Pretto is also with the Department of Computer, Control, and Management Engineering "Antonio Ruberti", Sapienza University of Rome, Italy. Email: pretto@dis.uniroma1.it

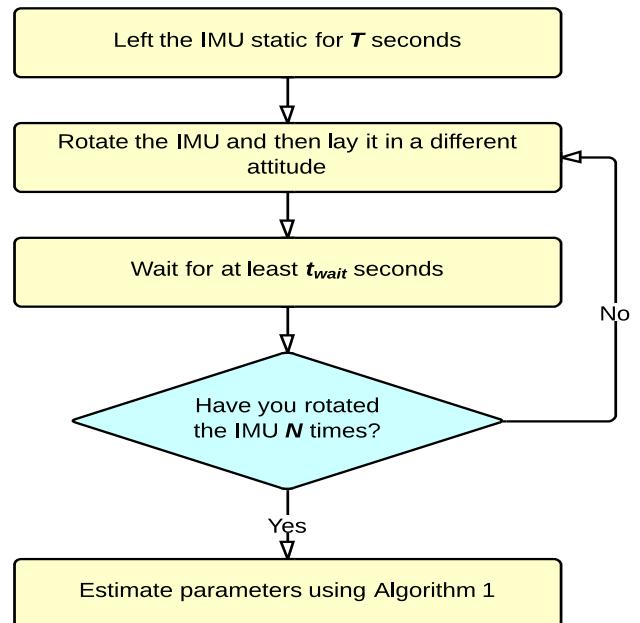


Fig. 1. Calibration protocol: (top) Some examples of the Xsens MTi IMU disposed in different attitudes as required by our method; (down) Diagram of our calibration protocol

sensitivities, and non zero biases. The IMU calibration refers to the process of identifying these quantities.

Many commercial IMU in the cost range from 1000 \\$ to 2000 \\$, such as the Xsens MTi [6] exploited in the experiments (Sec. V), are factory calibrated<sup>1</sup>. Each sensor is sold with its own calibration parameter set stored into the firmware or inside a non-volatile memory, providing accurate measurements off the shelf. Unfortunately, the overhead cost for the factory calibration is predominant: usually the sensor hardware (sensors, chips, embodiment, ...) is likely to be only a fraction of the final device cost. Actually, the factory calibration is usually performed using standard but effective methods, where the device outputs are compared with known references: this process requires time for each

<sup>1</sup>Often they are also compensated over temperature

sensor and a high cost equipment. On the other hand, low-cost IMUs (20-100 \$) and the IMU sensors that equip current smartphones are usually poorly calibrated, resulting in measurements coupled with not negligible systematic errors. For instance, state-of-the-art visual-inertial navigation systems such the one presented in [5], that exploits a smartphone as experimental platform, while performing so well in forward, almost regular, motion <sup>2</sup>, shows lower performances in more “exciting” motions, i.e. in motions that quickly change linear acceleration and rotational axes.

In this paper, we propose an effective and easy to implement calibration scheme, that only requires to collect IMU data with the simple procedure described in the flow chart reported in Fig. 1. After an initial initialization period with no motion, the operator should move the IMU in different positions, in order to generate a set of distinct, temporarily stable, rotations. The collected dataset is used to calibrate the scale and misalignments factors for both the accelerometers and gyroscopes triads, while estimating the sensor biases. As other calibration technique, we neglect the effect of the cross-axis sensitivities, since for minor misalignments and minor cross-axis sensitivities errors it is usually difficult to distinguish between them.

Our procedure exploits the basic idea of the multi-position method, firstly presented in [7] for accelerometers calibration: in a static position, the norms of the measured accelerations is equal to the magnitudes of the gravity plus a multi-source error factor (i.e., it includes biases, misalignment, noise,...). All these quantities can be estimated via minimization over a set of static attitudes. After the calibration of the accelerometer triad, we can use the gravity vector positions measured by the accelerometers as a reference to calibrate the gyroscope triad. Integrating the angular velocities between two consecutive static positions, we can estimate the gravity positions in the new orientation. The gyroscopes calibration is finally obtained minimizing the errors between these estimates and the gravity references given by the calibrated accelerometers.

In this procedure the gyroscopes calibration accuracy strongly depends on the accuracy of the accelerometers calibration, being used as a reference. Moreover, signal noise and biases should negatively affect both the calibration accuracy and the reliability of the algorithms used to detect the actual static intervals used in the calibration. Finally, a consistent numerical integration process is essential to mitigate the effect of the signal discretization, usually sampled at 100 Hz. In our approach, we face these problems introducing the following modifications to the standard multi-position method:

- The proposed calibration protocol exploits a larger number of static states with reduced periods, in order to increase the cardinality of the dataset while preserving the assumption of local stability of the sensors biases
- As proposed in [8], we characterize the gyroscope bias

<sup>2</sup>Actually, during an almost regular motion miscalibration errors may easily be assimilated by the biases included in the system state

drifts in a period estimated using the Allan variance

- We introduce a simple but effective *static detector* that exploits the sensor noise magnitude, a fixed-time sampling window and a cutting threshold, automatically estimated inside the optimization framework
- We employ the Runge-Kutta numerical integration method to improve the accuracy of the gyroscope calibration.

We have extensively tested our system using synthetic data affected by variable biases, misalignments, scale factor errors, and noise. In all of the cases, we have obtained stable and accurate results. Moreover, we have performed the calibration of a commercial, factory calibrated Xsens MTi IMU, using its raw, uncalibrated data as input. Our calibration results are comparable to the factory parameters reported by the device’s calibration certificate.

#### A. Related Works

Traditionally the calibration of an IMU has been done by using special mechanical platforms such as a robotic manipulator, moving the IMU with known rotational velocities in a set of precisely controlled orientations [9], [10], [11]. At each orientation, the output of the accelerometers are compared with the precomputed gravity vector while during the rotations the output of gyroscopes are compared with the precomputed rotational velocity. However, the mechanical platforms used for calibration are usually very expensive, resulting in a calibration cost that often exceeds the cost of the IMU’s hardware.

In [12] a calibration procedure that exploits a marker-based optical tracking system has been presented, while in [13], the GPS readings are used to calibrate initial biases and misalignments. Clearly, the accuracy of these methods strongly depends on the accuracy of the employed kinematic reference (i.e., the motion capture system or the GPS). The multi-position method was firstly introduced by Lötters *et al.* [7]: they proposed to calibrate the biases and the scale factor of the accelerometers using the fact that the magnitude of the static acceleration must equal to the gravity’s magnitude. This technique has been extended in [14] and [15] to include the accelerometer axis misalignment. The error model they proposed for the gyroscopes is similar to the one used for the accelerometers, but the calibration procedure in this case requires a single axis turntable to provide a strong rotation rate signal, providing high calibration accuracy. Unfortunately, these approaches not only require a mechanical equipment, but the two triads are independently calibrated, and the misalignment between them can’t be detected. In [8] and [16] authors presented two calibration schemes that do not require any external mechanical equipment. Similarly to our approach, in the first work the authors calibrate the accelerometers exploiting the high local stability of the gravity vector’s magnitude, and then gyroscopes calibration is obtained comparing the gravity vector sensed by the calibrated accelerometer with the gravity vector obtained by integrating the angular velocities. In the second work the authors also exploit the local stability of the magnetic field.

Hwangbo *et al.* [17] recently proposed a self-calibration technique based on an iterative matrix factorization. They use gravity as accelerometers reference, and a camera as gyroscopes reference.

## II. SENSOR ERROR MODEL

For an ideal IMU, the 3 axes of the accelerometers triad and the 3 axes of the gyroscopes triad define a single, shared, orthogonal 3D frame. Each accelerometer senses the acceleration along of one distinct axis, while each gyroscope measures the angular velocity around the same axis. Unfortunately in real IMUs, due to assembly inaccuracy, the two triads form two distinct (i.e., misaligned), non-orthogonal, frames. Also the single sensors are not perfect: typically the scaling factors used to convert the digital outputs of the sensors in real physical quantity are different for different instances of the same sensors, while only a default, nominal scaling factor is provided by manufacturers. Moreover, the output signals are almost always affected by non zero, variable biases.

As introduced above, both the accelerometers frame (AF) and the gyroscopes frame (GF) are usually non-orthogonal. We can define two associated orthogonal, ideal frames (AOF and GOF, respectively) in the following way:

- The x-axis of the AOF and the one of the AF coincide
- The y-axis of the AOF lies in the plan spanned by the x and y axes of the AF.

For the gyroscopes case, it is sufficient to substitute the AF and AOF acronyms with GF and GOF, respectively. Finally, we define a body frame (BF), which is an orthogonal frame that represents, for example, the coordinate frame of the IMU's chassis. The body frame usually differs from the AF and GF frames by small angles but, in general, there is no direct relation between them. For small angles, a measurement  $\mathbf{s}^S$  in a non-orthogonal frame (AF or GF) can be transformed in the orthogonal body frame as (for details of the derivation, see [18]):

$$\mathbf{s}^B = \mathbf{T}\mathbf{s}^S, \quad \mathbf{T} = \begin{bmatrix} 1 & -\beta_{yz} & \beta_{zy} \\ \beta_{xz} & 1 & -\beta_{zx} \\ -\beta_{xy} & \beta_{yx} & 1 \end{bmatrix} \quad (1)$$

where  $\mathbf{s}^B$  and  $\mathbf{s}^S$  denote the specific force (acceleration), or equivalently the rotational velocity, in the body frame coordinates and accelerometers (or gyroscopes) coordinates, respectively. Here  $\beta_{ij}$  is the rotation of the  $i$ -th accelerometer or gyroscope axis around the  $j$ -th BF axis, see Fig. 2.

On the other hand, the two orthogonal frames BF and AOF (and, equivalently, BF and GOF) are relate by a pure rotation.

In the presented calibration method, we assume that the body frame BF coincides with the accelerometers orthogonal frame AOF: in such case, the angles  $\beta_{xz}, \beta_{xy}, \beta_{yx}$  become zero, so in the accelerometers case Eq. 1 becomes:

$$\mathbf{a}^O = \mathbf{T}^a \mathbf{a}^S, \quad \mathbf{T}^a = \begin{bmatrix} 1 & -\alpha_{yz} & \alpha_{zy} \\ 0 & 1 & -\alpha_{zx} \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

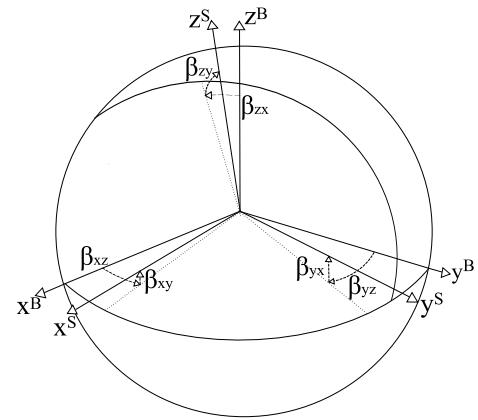


Fig. 2. Non-orthogonal sensor (accelerometers or gyroscopes) axes ( $x^S, y^S, z^S$ ), and body frame axes ( $x^B, y^B, z^B$ ).

where we have changed letter  $\beta$ , referring to the general case, with the letter  $\alpha$ , referring to the accelerometer case, while  $\mathbf{a}^O$  and  $\mathbf{a}^S$  denote the specific acceleration in AOF and AF, respectively<sup>3</sup>.

As mentioned before, gyroscope and accelerometer measurements should refer to the same reference frame, in our case AOF. Then, using Eq. 1, for the gyroscopes, we have:

$$\omega^O = \mathbf{T}^g \omega^S, \quad \mathbf{T}^g = \begin{bmatrix} 1 & -\gamma_{yz} & \gamma_{zy} \\ \gamma_{xz} & 1 & -\gamma_{zx} \\ -\gamma_{xy} & \gamma_{yx} & 1 \end{bmatrix} \quad (3)$$

where  $\omega^O$  and  $\omega^S$  denote the specific angular velocities in AOF and in GF, respectively.

Both the accelerometers and the gyroscopes are affected by biases and scale errors. Two scaling matrix are introduced

$$\mathbf{K}^a = \begin{bmatrix} s_x^a & 0 & 0 \\ 0 & s_y^a & 0 \\ 0 & 0 & s_z^a \end{bmatrix}, \quad \mathbf{K}^g = \begin{bmatrix} s_x^g & 0 & 0 \\ 0 & s_y^g & 0 \\ 0 & 0 & s_z^g \end{bmatrix} \quad (4)$$

We introduce also two bias vector

$$\mathbf{b}^a = \begin{bmatrix} b_x^a \\ b_y^a \\ b_z^a \end{bmatrix}, \quad \mathbf{b}^g = \begin{bmatrix} b_x^g \\ b_y^g \\ b_z^g \end{bmatrix} \quad (5)$$

The complete sensor error model is

$$\mathbf{a}^O = \mathbf{T}^a \mathbf{K}^a (\mathbf{a}^S + \mathbf{b}^a + \boldsymbol{\nu}^a) \quad (6)$$

for the accelerometers, and

$$\omega^O = \mathbf{T}^g \mathbf{K}^g (\omega^S + \mathbf{b}^g + \boldsymbol{\nu}^g) \quad (7)$$

for the gyroscope, where  $\boldsymbol{\nu}^a$  and  $\boldsymbol{\nu}^g$  are the accelerometer measurement noise and the gyroscope measurement noise, respectively.

<sup>3</sup>To relate the obtained calibration with a different body frame (e.g. BF'), it is sufficient to estimate the rotation matrix that relate AOF to BF', for instance using the accelerometers outputs in three different orthogonal orientations.

### III. BASIC CALIBRATION FRAMEWORK

In order to calibrate the accelerometers triad, we need to estimate the following unknown parameter vector:

$$\boldsymbol{\theta}^{acc} = [\alpha_{yz}, \alpha_{zy}, \alpha_{zx}, s_x^a, s_y^a, s_z^a, b_x^a, b_y^a, b_z^a] \quad (8)$$

We define the following function:

$$\mathbf{a}^O = h(\mathbf{a}^S, \boldsymbol{\theta}^{acc}) = \mathbf{T}^a \mathbf{K}^a (\mathbf{a}^S + \mathbf{b}^a) \quad (9)$$

Here we can neglect the measurements noise thanks to the fact that in our calibration procedure we apply signal averaging in each static interval.

As in the conventional multi-position scheme, we move the IMU in a set of  $M$  distinct, temporarily stable, rotations. We can extract  $M$  acceleration vectors  $\mathbf{a}_k^S$  (measured in the non-orthogonal AF), averaging the accelerometers readings in a temporal window inside each static interval. The cost function we use to estimate accelerometers' parameters is:

$$\mathbf{L}(\boldsymbol{\theta}^{acc}) = \sum_{k=1}^M (||\mathbf{g}||^2 - ||h(\mathbf{a}_k^S, \boldsymbol{\theta}^{acc})||^2)^2 \quad (10)$$

where  $||\mathbf{g}||$  is the actual magnitude of the local gravity vector that can easily recovered from specific public tables (e.g., knowing latitude, longitude and altitude of the location where we are performing the calibration). In order to minimize Eq. 10, we employ the *Levenberg-Marquardt* (LM) algorithm.

In order to calibrate the gyroscope triad, we can assume the system as bias-free simply averaging the static gyroscope signals over a suitable initial period of no motion. This is justified by the following discussion about the Allan variance (see Sec. IV-C). Moreover, since we need to use the accelerometers as known references, we use the calibration parameters  $\boldsymbol{\theta}^{acc}$  computed above, correcting the accelerometers readings with Eq. 9.

We define the operator  $\Psi$ , that takes as input a sequence of  $n$  gyroscopes readings  $\omega_i^S$  and an initial gravity verson  $\mathbf{u}_{a,k-1}$  (i.e., a unit vector representing the gravity direction) given by the calibrated accelerometers, and return the final gravity verson  $\mathbf{u}_{g,k}$ , computed using the gyroscopes measurements between the  $k-1$ -th and the  $k$ -th static intervals:

$$\mathbf{u}_{g,k} = \Psi [\omega_i^S, \mathbf{u}_{a,k-1}] \quad (11)$$

$\Psi$  can be any integration algorithm that computes the final orientation through integrating the input angular velocities. The unknown parameter vector we need to estimate to calibrate the gyroscope is:

$$\boldsymbol{\theta}^{gyro} = [\gamma_{yz}, \gamma_{zy}, \gamma_{xz}, \gamma_{zx}, \gamma_{xy}, \gamma_{yx}, s_x^g, s_y^g, s_z^g] \quad (12)$$

In this case, we can define the cost function as:

$$\mathbf{L}(\boldsymbol{\theta}^{gyro}) = \sum_{k=2}^M ||\mathbf{u}_{a,k} - \mathbf{u}_{g,k}||^2 \quad (13)$$

where  $M$  is the number of static intervals,  $\mathbf{u}_{a,k}$  is the acceleration verson measured averaging in a temporal window the

calibrated accelerometer readings in the  $k$ -th static interval, and  $\mathbf{u}_{g,k}$  is the acceleration verson computed using Eq. 11 (i.e., integrating the angular velocities between the  $k-1$ -th and the  $k$ -th static intervals). We obtain  $\boldsymbol{\theta}^{gyro}$  minimizing Eq. 13 with LM.

### IV. CALIBRATION PROCEDURE

As introduced in Sec. I, the proposed calibration framework requires to collect a dataset with the stream of the raw accelerometers and gyroscopes readings, taken while the operator moves the IMU in different static positions, in order to generate a set of distinct, temporarily stable, rotations. A simple diagram of our calibration protocol is reported in Fig. 1. As mentioned in Sec. III, to mitigate the noise effect in the minimization of Eq. 10 and Eq. 13, we need to average the signals over a suitable time interval. This imposes a lower bound in the length of the static interval ( $t_{wait}$  in Fig. 1). A initialization period ( $T_{init}$  in Fig. 1) with no motion is essential as well: this will be exploited to characterize the gyroscopes biases (Sec. IV-C) and the static detector operator (Sec. IV-A).

#### A. Static Detector

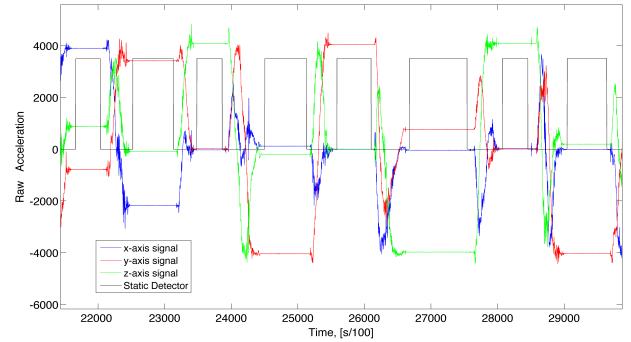


Fig. 3. An example of the static detector applied to the accelerometers data: the static detector is represented by the black square wave, its high level classify an interval as static.

The accuracy of the calibration strongly depends on the reliability in the classification between static and motion intervals: to calibrate the accelerometers we use static intervals, while for gyroscopes calibration we also include the motion intervals between two consecutive static intervals. In our experience, band-pass filter based operators, like the *quasi-static detector* used in [8], perform poorly with real datasets: detected static intervals frequently includes some small portion of motion. Moreover, they require a fine tuning, since they depend on three parameters.

We propose instead to use a variance based static detector operator, that exploits the lower bound in the lengths of the static interval introduced above. We base our detector on the accelerometer signals: given a time interval of length  $t_{wait}$  seconds (see Fig. 1), for each accelerometers sample  $(\mathbf{a}_x^t, \mathbf{a}_y^t, \mathbf{a}_z^t)$  at time  $t$ , we compute the variance magnitude, i.e. the magnitude of the variance, as:

$$s(t) = \sqrt{[var_{t_w}(\mathbf{a}_x^t)]^2 + [var_{t_w}(\mathbf{a}_y^t)]^2 + [var_{t_w}(\mathbf{a}_z^t)]^2} \quad (14)$$

where  $\text{var}_{t_w}(\mathbf{a}^t)$  is an operator that compute the variance of a general signal  $\mathbf{a}^t$  in a time interval of length  $t_w$  seconds centered in  $t$ . We classify between static and motion intervals simply checking if the square of the  $\zeta(t)$  is lower or greater than a threshold. As a threshold, we consider an integer multiple of the square of the variance magnitude  $\zeta_{\text{init}}$ , computed over all the initialization period  $T_{\text{init}}$ . In all the experiments, we use  $t_w = 2 \text{ sec}$ , while  $T_{\text{init}}$  is estimated using the Allan variance (see Sec. IV-C). It is important to note that our static detector does not require any parameter tuning: the integer multiplier used in the classification is automatically estimated by our calibration algorithm (see Sec. IV-D). Fig. 3 reports an example of how our static filter works on real data: in this case the estimated integer multiplier is 6.

### B. Runge-Kutta Integration

As reported in Eq. 11, in the gyroscopes calibration we need to perform a discrete time angular velocity integration: a robust and stable numerical integration method is desirable since it can improve the calibration accuracy. Given a common instruments rate of 100 Hz (like the Xsens MTi IMU used in the experiments) and since we represent rotations using quaternion arithmetic, with this setup a proper integration algorithm choice [19] is the *Runge-Kutta 4<sup>th</sup> order normalized method* (RK4n). In our experience (experiments not reported for space constraints) RK4n outperforms the standard linear integration procedure providing higher accuracy results. Let Eq. 15 be the differential equation describing the quaternion kinematics:

$$\mathbf{f}(\mathbf{q}, t) = \dot{\mathbf{q}} = \frac{1}{2} \Omega(\boldsymbol{\omega}(t)) \mathbf{q}, \quad (15)$$

where  $\Omega(\boldsymbol{\omega})$  is the operator which turns the considered tri-dimensional angular velocity into the real skew symmetric matrix representation, that is:

$$\Omega(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix}. \quad (16)$$

The RK4n integration algorithm is:

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \Delta t \frac{1}{6} (\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4), \quad (17)$$

$$\mathbf{k}_i = \mathbf{f}(\mathbf{q}^{(i)}, t_k + c_i \Delta t), \quad (18)$$

$$\mathbf{q}^{(i)} = \mathbf{q}_k, \quad \text{for } i = 1, \quad (19)$$

$$\mathbf{q}^{(i)} = \mathbf{q}_k + \Delta t \sum_{j=1}^{i-1} a_{ij} \mathbf{k}_j, \quad \text{for } i > 1. \quad (20)$$

where all the coefficients needed,  $c_i$  and  $a_{ij}$ , are

$$\begin{aligned} c_1 &= 0, & c_2 &= \frac{1}{2}, & c_3 &= \frac{1}{2}, & c_4 &= 1, \\ a_{21} &= \frac{1}{2}, & a_{31} &= 0, & a_{41} &= 0, \\ a_{32} &= \frac{1}{2}, & a_{42} &= 0, & a_{43} &= 1. \end{aligned}$$

Finally, for each step, we also need to normalize the  $(k+1)$ -th quaternion:

$$\mathbf{q}_{k+1} \rightarrow \frac{\mathbf{q}_{k+1}}{\|\mathbf{q}_{k+1}\|}. \quad (21)$$

### C. Allan Variance

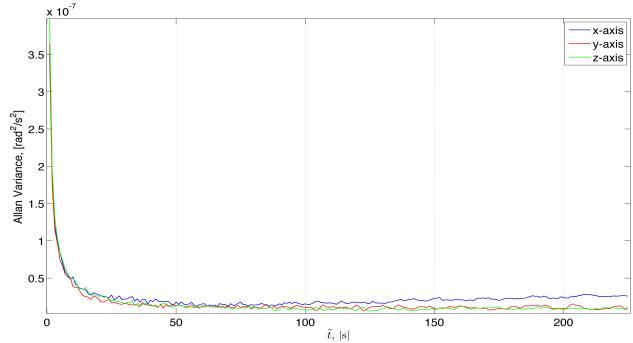


Fig. 4. Allan Variance computed for the Xsens MTi gyroscopes triad.

We characterize the random gyroscope bias drifts using the Allan variance ([20], [8]), which measures the variance of the difference between consecutive interval averages. The Allan variance  $\sigma_a^2$  is defined as:

$$\begin{aligned} \sigma_a^2 &= \frac{1}{2} \langle (x(\tilde{t}, k) - x(\tilde{t}, k-1))^2 \rangle = \\ &= \frac{1}{2K} \sum_{k=1}^K (x(\tilde{t}, k) - x(\tilde{t}, k-1))^2 \end{aligned} \quad (22)$$

where  $x(t, k)$  is the  $k$ -th interval average which spans  $\tilde{t}$  seconds, and  $K$  is the number of interval which the total considered time is segmented in. We compute the Allan variance for each gyroscope axis, with  $t_0 \leq \tilde{t} \leq t_n$ . We fix  $t_0 = 1\text{s}$ ,  $t_n = 225\text{s}$ . The time in which the Allan variances of the three axis converge to a small value represents a good choice for initialization period  $T_{\text{init}}$  (Fig. 1). In this initialization period, we compute the average of the static gyroscope signals to correctly determine the gyroscopes biases used in the calibration. In the case of the Xsens MTi IMU, a good value for  $T_{\text{init}}$  is 50 seconds (see Fig. 4).

### D. Complete Procedure

To avoid unobservability in the calibration parameters estimation, a minimum of nine different attitudes [15] has to be collected (e.g., Fig. 1). In our experience, a higher number  $N$  of distinct attitudes are required to get better calibration results, while keeping reduced the duration of each static interval in order to preserve the assumption of temporal stability of the gyroscopes biases. With  $36 \leq N \leq 50$  and  $1\text{ sec} \leq t_{\text{wait}} \leq 4\text{ sec}$ , we obtain a good trade-off between calibration accuracy, biases stability, and noise reduction. The duration of the initialization period  $T_{\text{init}}$  is given by the Allan variance analysis (see Sec. IV-C). The calibration protocol is summarized in Fig. 1, while in Algorithm 1 the pseudo-code of the calibration algorithm is reported.

## V. EXPERIMENTS

We have tested our method with both synthetic and real data. In the simulations, we can compare the results with a perfect ground truth, i.e., a noise-free, undistorted

---

**Algorithm 1** IMU Calibration

---

**Require:**  $T_{init}$ ,  $t_{wait}$ ;  $\mathbf{a}^S$  and  $\boldsymbol{\omega}^S$  (accelerometer's and gyroscope's dataset collected according to Fig. 1).

---

```

 $\mathbf{b}^g \leftarrow$  average gyroscope signals over  $T_{init}$ ;
 $\boldsymbol{\omega}_{biasfree}^S \leftarrow \boldsymbol{\omega}^S - \mathbf{b}^g$ ;
 $M_{inf} \leftarrow$  empty matrix;
 $s_{init} \leftarrow$  Eq. 14, with  $t_w = T_{init}$ ;
for  $i = 1 : k$ 
     $threshold \leftarrow i * s_{init}^2$ ;
     $s\_intervals \leftarrow$  static detector computed
        using  $t_{wait}$  and  $threshold$ ;
     $[Residual, Params_{acc}] \leftarrow$  optimize Eq. 10 using
         $s\_intervals$  and  $\mathbf{a}^S$ , averaging with  $t_{wait}$ ;
     $M_{inf}(i) \leftarrow [Residual, Params_{acc}, threshold,$ 
         $s\_intervals]$ ;
end
 $index_{opt} \leftarrow$  index of the minimum residual in  $M_{inf}$ ;
 $Params_{acc} \leftarrow$  from  $M_{inf}$  using  $index_{opt}$ ;
 $s\_intervals_{opt} \leftarrow$  from  $M_{inf}$  using  $index_{opt}$ ;
 $\mathbf{a}^O \leftarrow$  calibrate  $\mathbf{a}^S$  using  $Params_{acc}$ ;
 $Params_{gyro} \leftarrow$  optimize Eq. 13 using  $s\_intervals_{opt}$ ,
 $\boldsymbol{\omega}_{biasfree}^S$  and  $\mathbf{a}^O$ , averaging with  $t_{wait}$ .

```

---

signal, while the calibration matrices are known. With the real dataset, we compare the estimated calibration parameters with the calibration parameters reported in the IMU datasheet.

#### A. Synthetic Data

We first generate a set of ideal, noise-free signals. The accelerometers readings are generated starting from a three-dimensional signal based on three different-pulsation sinusoids randomly modulated. At the beginning we add 5000 zero samples (the initialization period) and every time the three signals are simultaneously zero we introduce 400 zero samples (the static intervals). The three dimensional gravity vector projected onto the three axis has been added as well. For the angular velocities sensed by the gyroscope, the idea is to consider a tri-dimensional angular velocity vector  $\boldsymbol{\omega}$ , which describes the perceived rotation of the aforementioned gravity vector, and then project  $\boldsymbol{\omega}$  onto the three axis of the gyroscope. In this way we correlate the measurement of the two different clusters of sensors. For each motion interval different zenith and azimuth velocities are randomly chosen, while for the rest of the time these velocities are considered to be equal to zero. The sampling frequency of the whole synthetic data has been fixed to 100 Hz.

For each ideal signal, we add a white gaussian noise and finally we distort the data with random generated distortion parameters, i.e.:

$$\mathbf{a}_{synth}^S = (\mathbf{T}^a \mathbf{K}^a)^{-1} \mathbf{a}_{synth}^O - \mathbf{b}^a \quad (23)$$

for the accelerations, and:

$$\boldsymbol{\omega}_{synth}^S = (\mathbf{T}^g \mathbf{K}^g)^{-1} \boldsymbol{\omega}_{synth}^O - \mathbf{b}^g \quad (24)$$

for the angular velocities.  $\mathbf{a}_{synth}^S$  and  $\mathbf{a}_{synth}^O$  are the synthetic acceleration in the non-orthogonal sensor frame and in the associated orthogonal frame, respectively.  $\boldsymbol{\omega}_{synth}^S$  and  $\boldsymbol{\omega}_{synth}^O$  are the synthetic angular velocities in the non-orthogonal sensor frame and in the associated orthogonal frame, respectively. Eq. 23 and Eq. 24 are obtained from models proposed in Eq. 6 and in Eq. 7. The metrics we use to evaluate the quality of the results are:

- 1) Comparing the estimated values to the real ones;
- 2) Comparing the average difference between the perfect, noise-free and undistorted signal with the noisy signal before and after calibration;
- 3) For the accelerometers only: we consider the magnitude and the angular error between the sensed acceleration and the applied one during each *static intervals*. Since the magnitude of the gravity vector is assumed to be the only quantity known, the angular error here is calculated for the worst case, where the full error appears on a single accelerometer axis which is perfectly horizontal, *i.e.* perpendicular to the gravity vector. An error of  $\mathbf{g} \cdot \sin(\theta_{div}^{acc})$  will result in the pitch or roll angle being measured as  $\theta_{div}^{acc}$  radians instead of zero;
- 4) For the gyroscopes only: we consider the magnitude and the angular error between the acceleration sensed by the calibrated accelerometer and the acceleration computed integrating the angular velocities given by the gyroscope.

We have generated 1200 different signals, combining 40 different distortion parameter sets with a set of 30, randomly generated, ideal signals. Due to reduced space, we report average and worst case results for one distortion parameter set, applied to the whole set of ideal signals. Results for the other distortion parameter sets are very similar<sup>4</sup>.

TABLE I  
First metric: ACCELEROMETERS PARAMETERS

	Real Value	Mean value	RMS x10 <sup>-3</sup>	Mean Error x10 <sup>-3</sup>	RMS x10 <sup>-3</sup>
$\alpha_{yz}$	0.0049	0.0049	0.0481	0.0398	0.0275
$\alpha_{zy}$	-0.0055	-0.0055	0.0401	0.0334	0.0214
$\alpha_{zx}$	0.0079	0.0079	0.0296	0.0248	0.0190
$s_x^a$	0.9908	0.9908	0.0327	0.0265	0.0191
$s_y^a$	1.0068	1.0068	0.0304	0.0258	0.0199
$s_z^a$	1.0066	1.0066	0.0215	0.0178	0.0151
$b_x^a$	0.0793	0.0793	0.1369	0.1163	0.0819
$b_y^a$	-0.0024	-0.0024	0.2138	0.1760	0.1178
$b_z^a$	0.0636	0.0636	0.1332	0.0953	0.0919

From Tab. I and Tab. II, we report a negligible average error, with an order of magnitude always equal or less than 10<sup>-4</sup>. From Tab. IV and Tab. V, we report significant

<sup>4</sup> All synthetic results, the software for real calibration and a real dataset are available following the link: robotics.dei.unipd.it/~pretto/software/tpm\_icra2014.zip

TABLE II  
First metric: GYROSCOPES PARAMETERS

	Real value	Mean value	RMS x10 <sup>-3</sup>	Mean Error x10 <sup>-3</sup>	RMS x10 <sup>-3</sup>
$\gamma_{yz}$	0.0112	0.0110	0.8547	0.6392	0.5920
$\gamma_{zy}$	-0.0211	-0.0210	0.4419	0.3468	0.2669
$\gamma_{xz}$	0.0040	0.0039	1.0630	0.9080	0.5266
$\gamma_{zx}$	-0.0010	-0.0011	0.4102	0.3386	0.2302
$\gamma_{xy}$	0.0270	0.0270	0.8154	0.6375	0.4944
$\gamma_{yx}$	0.0151	0.0155	0.7250	0.7315	0.3958
$s_x^g$	0.8786	0.8785	0.4121	0.3366	0.2299
$s_y^g$	0.9703	0.9704	0.4059	0.3353	0.2237
$s_z^g$	1.0460	1.0460	0.4216	0.3410	0.2397

TABLE III  
First metric: WORST CASE - PARAMETERS

	Real value	Est value		Real value	Est Value
$\alpha_{yz}$	0.0049	0.0049	$\gamma_{yz}$	0.0112	0.0099
$\alpha_{zy}$	-0.0055	-0.0055	$\gamma_{zy}$	-0.0211	-0.0207
$\alpha_{xz}$	0.0079	0.0079	$\gamma_{xz}$	0.0040	0.0030
$s_x^a$	0.9908	0.9908	$\gamma_{zx}$	-0.0010	-0.0011
$s_y^a$	1.0068	1.0068	$\gamma_{xy}$	0.0270	0.0252
$s_z^a$	1.0066	1.0066	$\gamma_{yx}$	0.0151	0.0166
$b_x^a$	0.0793	0.0792	$s_x^g$	0.8786	0.8790
$b_y^a$	-0.0024	-0.0026	$s_y^g$	0.9703	0.9701
$b_z^a$	0.0636	0.0636	$s_z^g$	1.0460	1.0460

TABLE IV  
Second metric: ABSOLUTE ERRORS ALONG THE AXIS - ACCELEROMETERS

	x-axis m/s <sup>2</sup>	y-axis m/s <sup>2</sup>	z-axis m/s <sup>2</sup>
Uncalib	0.0842	0.0564	0.0635
Calib	0.0055	0.0056	0.0056

TABLE V  
Second metric: ABSOLUTE ERRORS ALONG THE AXIS - GYROSCOPES

	x-axis (rad/s)	y-axis (rad/s)	z-axis (rad/s)
Uncalib	0.1043	0.1097	0.0345
Calib	0.0035	0.0039	0.0042

TABLE VI  
Third metric: ACCELEROMETERS DIVERGENCE ERROR

	Average error m/s <sup>2</sup> (rad)	Max observed error m/s <sup>2</sup> (rad)	Worst case average error m/s <sup>2</sup> (rad)	Worst case max error m/s <sup>2</sup> (rad)
Uncalib	0.0665 ( 0.0114)	0.2133 ( 0.0226)	0.0623 ( 0.0115)	0.2098 ( 0.0240)
Calib	0.0056 ( 0.0009)	0.0299 ( 0.0035)	0.0056 ( 0.0009)	0.0298 ( 0.0038)

TABLE VII  
Fourth metric: GYROSCOPES DIVERGENCE ERROR

	Average error m/s <sup>2</sup> (rad)	Max observed error m/s <sup>2</sup> (rad)	Worst case average error m/s <sup>2</sup> (rad)	Worst case max error m/s <sup>2</sup> (rad)
Uncalib	4.7125 ( 0.5494)	9.2930 ( 0.5494)	5.2859 ( 0.6029)	8.5822 ( 0.6029)
Calib	0.2208 ( 0.0256)	0.4469 ( 0.0256)	0.5102 ( 0.0569)	0.8597 ( 0.0569)

reductions in the absolute errors. Finally, from Tab. VI we report that the divergence's magnitude is reduced by a factor of 11.9 and the angular error by a factor of 12.7, while in Tab. VII the magnitude of the considered divergence is reduced by a factor of 21.4 and the angular error by a factor of 21.9. In Fig. 5, we show some examples of the calibrated signal compared with the ideal one.

### B. Real Data

As introduced, we have exploited an Xsens MTi IMU (Fig. 1) as experimental platform. The device datasheet provides the factory calibrated misalignment matrices that align the accelerometers (AF) and gyroscopes (GF) frames to the body frame BF, while we estimate the matrices that align that AF and GF to AOF. In order to compare our results with the results of the factory calibration, we need to know the matrix  $R_b$  that relates AOF to BF. Given  $R_b$ , we can express our calibration vectors in BF. Given the factory calibration,  $R_b$  can be easily estimated using data contained in misalignment matrix (see discussion about Eq. 1) and following a simple method proposed in [14]<sup>5</sup>.

We acquired a dataset as described in Fig. 1, with an initial static period of about 50 seconds, followed by a set of 37 rotations separated by static intervals of 2-4 seconds. We uses as initial guess for the optimization the ideal values for the accelerometer, that are (see Eq. 8):

$$[1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1]. \quad (25)$$

While for the gyroscope we used (see Eq. 12):

$$\left[ \frac{1}{r} \ 0 \ 0 \ 0 \ \frac{1}{r} \ 0 \ 0 \ 0 \ \frac{1}{r} \right], \ r = \frac{(2^n - 1)}{2y} \quad (26)$$

where  $n$  is the numbers of bit of the A/D converter, and the gyroscope full-scale from datasheet is  $[-y, +y] \text{ rad/s}$ .

The achieved calibration is comparable to the factory calibration parameters given by the datasheet, see tables starting from VIII. For each table, we report the factory calibration parameters (on the left), and our calibration parameters (on the right). It is important to point out that in our results we are implicitly including an error that can't be attributed to our calibration method. This is the propagated error caused by the IMU's datasheet rounded values (see table) when we compute  $R_p$ . In spite of this problem, the results we obtain are very close to the parameters reported in the datasheet<sup>4</sup>.

TABLE VIII  
SCALING - ACCELEROMETER

415	0.00	0.00
0.00	413	0.00
0.00	0.00	415

414.41	0	0
0	412.05	0
0	0	414.61

<sup>5</sup>Due to reduced space, we can't report here other details.

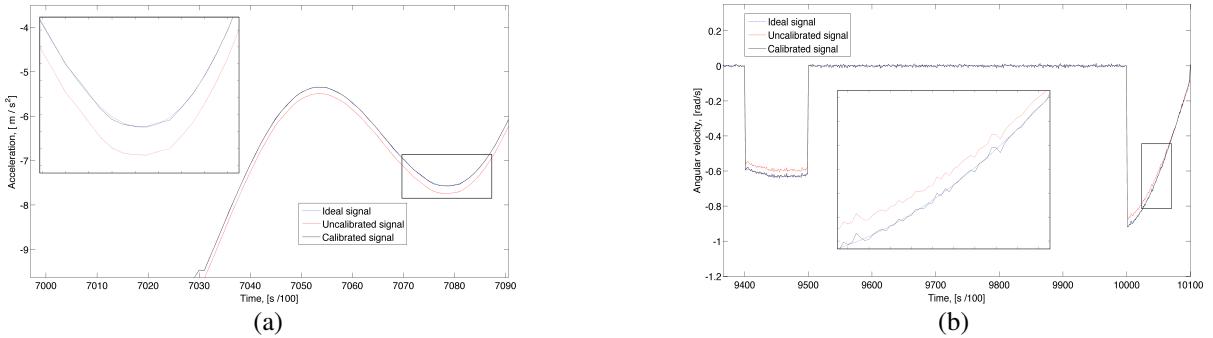


Fig. 5. Calibration Improvement: (a) accelerometers, (b) gyroscopes (the small box is zoomed in the larger box).

TABLE IX  
SCALING - GYROSCOPE

4778	0.00	0.00
0.00	4758	0.00
0.00	0.00	4766

4778.0	0	0
0	4764.8	0
0	0	4772.6

TABLE X  
MISALIGNMENT - ACCELEROMETER

1.00	0.00	-0.01
0.01	1.00	0.01
0.02	0.01	1.00

1.0000	-0.0066	-0.0110
0.0102	1.0001	0.0114
0.0201	0.0098	0.9998

## VI. CONCLUSIONS AND FUTURE WORKS

In this paper, we have presented an effective and semi-automatic IMU calibration method. The calibration protocol is quick, reliable and it does not need any external equipment. The proposed algorithms are easy to implement and they do not require any parameters tuning. Results achieved with both synthetic and real data show the real possibility to improve at no cost the performances of low cost, poor calibrated, IMUs, such the ones that equip the current smartphones. Actually, we are currently testing our approach on smartphone-grade IMU sensors. We also plan to integrate the proposed procedure in a calibration framework for camera-IMU systems, that includes IMU, camera-IMU frame and temporal calibrations.

## REFERENCES

- [1] V. Kubelka and M. Reinstein, "Complementary filtering approach to orientation estimation using inertial sensors only," in *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 599–605.
- [2] T. Hamel and R. E. Mahony, "Attitude estimation on  $\text{so}(3)$  based on direct inertial measurements," in *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, 2006, pp. 2170–2175.
- [3] K. Tsotsos, A. Pretto, and S. Soatto, "Visual-inertial ego-motion estimation for humanoid platforms," in *Proc. of IEEE-RAS International Conference on Humanoid Robots*, 2012, pp. 704–711.
- [4] K. Konolige and M. Agrawal, "Frameslam: From bundle adjustment to real-time visual mapping," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1066–1077, Oct. 2008.
- [5] M. Li, B. Kim, and A. I. Mourikis, "Real-time motion estimation on a cellphone using inertial sensing and a rolling-shutter camera," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, May 2013, pp. 4697–4704.
- [6] XSens, <http://www.xsens.com>.
- [7] J. Lotters, J. Schipper, P. Veltink, W. Olthuis, and P. Bergveld, "Procedure for in-use calibration of triaxial accelerometers in medical applications," *Sensors and Actuators A: Physical*, vol. 68, no. 1-3, pp. 221–228, 1998.
- [8] W. Fong, S. Ong, and A. 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.
- [9] R. Rogers, *Applied Mathematics in Integrated Navigation Systems*, ser. AIAA education series. American Institute of Aeronautics and Astronautics, 2003.
- [10] A. B. Chatfield, *Fundamentals of high accuracy inertial navigation*, ser. Progress in astronautics and aeronautics. Reston, VA. American Institute of Aeronautics and Astronautics, Inc., 1997.
- [11] J. J. Hall and R. L. Williams II, "Inertial measurement unit calibration platform," *Journal of Robotic Systems*, vol. 17, no. 11, pp. 623–632, 1998.
- [12] A. Kim and M. Golnaraghi, "Initial calibration of an inertial measurement unit using an optical position tracking system," in *Position Location and Navigation Symposium, 2004. PLANS 2004*, 2004, pp. 96–101.
- [13] E. M. Nebot and H. F. Durrant-Whyte, "Initial calibration and alignment of low-cost inertial navigation units for land vehicle applications," *Journal of Robotic Systems*, vol. 16, no. 2, pp. 81–92, 1999.
- [14] I. Skog and P. Hndel, "Calibration of a mems inertial measurement unit," in *Proc. of XVII IMEKO WORLD CONGRESS*, 2006.
- [15] Z. F. Syed, P. Aggarwal, C. Goodall, X. Niu, and N. El-Sheimy, "A new multi-position calibration method for mems inertial navigation systems," *Measurement Science and Technology*, vol. 18, pp. 1897–1907, 2007.
- [16] C. M. Cheuk, T. K. Lau, K. W. Lin, and Y. Liu, "Automatic calibration for inertial measurement unit," in *Proc. of International Conference on Control, Automation, Robotics and Vision*, 2012.
- [17] M. Hwangbo, J.-S. Kim, and T. Kanade, "Imu self-calibration using factorization," *IEEE Transactions on Robotics*, vol. 29, no. 2, pp. 493–507, 2013.
- [18] C. Jekeli, *Inertial Navigation Systems with Geodetic Applications*. De Gruyter, 2001.
- [19] M. Andrie and J. Crassidis, "Geometric integration of quaternions," in *Proc. of AIAA/AAS Astrodynamics Specialist Conference*, 2012.
- [20] A. M. Sabatini, "A wavelet-based bootstrap method applied to inertial sensor stochastic error modelling using the allan variance," *Measurement Science and Technology*, vol. 17, no. 11, pp. 2980–2988, 2006.