Iterative calibration method for inertial and magnetic sensors

Eric Dorveaux, David Vissière, Alain Pierre Martin, Nicolas Petit

HAL Id: hal-00473610

https://hal-mines-paristech.archives-ouvertes.fr/hal-00473610

Submitted on 12 May 2010

Joint 48th IEEE Conference on Decision and Control and
28th Chinese Control Conference
Shanghai, P.R. China, December 16-18, 2009

FrC09.3

Iterative calibration method for inertial and magnetic sensors

Eric Dorveaux, David Vissière, Alain-Pierre Martin, Nicolas Petit

Abstract— We address the problem of three-axis sensor calibration. Our focus is on magnetometers. Usual errors (misalignment, non-orthogonality, scale factors, biases) are accounted for. We consider a method where no specific calibration hardware is required. We solely use the fact that the norm of the sensed field must remain constant irrespective of the sensors orientation. The proposed algorithm is iterative. Its convergence is studied. Experiments conducted with MEMS sensors (magnetometers) stress the relevance of the approach.

INTRODUCTION

Numerous military and civilian control applications require high accuracy position, speed, and attitude estimations of a solid body. Examples range from Unmanned Air Vehicles (UAV), Unmanned Ground Vehicles (UGV), full-sized submarines, civil engineering positioning devices, to name a few. A widely considered solution is to use embedded Inertial Measurement Units (IMU). Accelerometers, gyroscopes signals can be used to derive position and orientation information through a double integration process [11], [7]. This approach requires very high precision IMUs and well calibrated sensors. An important challenge appears when cost, space or weight constraints become stringent.

The recent progress in very low cost (less than 300 USD), low weight (less than 100 g) and low size (less than 3 cm^2) IMUs have spurred a broad interest in the development of IMU-based positioning technologies. These Micro-Electro-Mechanical Systems (MEMS) IMUs appear to have quickly increasing capabilities. Several manufacturers are announcing new models under 4,000 USD capable of less than 20 deg/hr angular errors.

Yet, there still does not exist any reported experiment proposing to estimate the position from such a low cost IMU. In the literature, these IMUs are only used for orientation (attitude) estimation (see e.g. [4], [12] or [3] for an application to the control of mini-UAVs in closed loop). Some tentative work (using higher-end IMUs) address the problem of velocities estimation. In these cases, the speed information is obtained from a GPS receiver using the Doppler effect (see [11] for details on the quality of the obtained measurements information) or from odometers (in the case of ground vehicles). Model based approaches permit to reduce the dependence on GPS (see [15]).

For indoor pedestrian applications, GPS is almost totally unavailable. Recently, a new approach called distributed-magnetic-inertial navigation has been proposed [16]. It relies solely on MEMS magnetometers, accelerometers and gyroscopes and takes advantage of the unknown magnetic field disturbances usually observed indoors to estimate drift in velocities. This disturbances are evaluated using a three-dimensional array of magnetometers. In details, an experimental test bed was constructed that integrates an IMU consisting of one three-axis accelerometer, one three-axis gyroscope and one three-axis magnetometer and a set of eight spatially distributed three-axis magnetometers. This complex measurement system can produce meaningful data if and only if all the sensors are well-calibrated and all the sampling times are precisely known and accounted for (this last point is discussed in [6]). Calibration of such three axis sensors measuring a constant vector field is the subject of this paper.

In classic inertial navigation, there exist various methods for three-axis sensors calibration. Most of them have an important drawback in common. They require expansive tools to acquire the data and compare them against a fixed reference, and, quite often, a high degree of expertise to process the data. Usually (see e.g. [5]), IMUs calibration is achieved using a well-instrumented mechanical platform (called *calibration table*) whose varying orientation is precisely measured. The platform is rotated to various precisely controlled orientations which serve as comparisons against the orientations determined from the IMU sensors. The rotational velocities are precisely controlled as well. Magnetometers calibration is usually performed using a similar method in magnetically shielded facilities to provide a known uniform field (see e.g. [13]). Similarly, measurements are then performed with precise knowledge of sensors orientation.

The recent development of micro-electro mechanical systems (MEMS) and other low-cost sensors has led to a paradox. Due to their relatively low quality, these low-cost sensors are in great need of a calibration procedure (much more than higher-end sensors), but the cost of the traditional calibration procedures may exceeds by many times the cost of developing and constructing the sensors themselves, and calibration may change over time. Moreover, as the cost of the sensors are decreasing, their use is spreading. This yields a great interest in developing new "simple but effective" calibration procedures that do not require a high degree of expertise nor an expensive hardware to be put into practice. Lately, a new paradigm for such sensors calibration has

E. Dorveaux (corresponding author) is a PhD candidate in Mathematics and Control, Centre Automatique et Systèmes, Mines-ParisTech, 60, bd St Michel, 75272 Paris, France eric.dorveaux@mines-paristech.fr

D. Vissière is with SYSNAV, Zone industrielle B, 1 rue Jean de Becker Rémy, BP86, 27940 AUBEVOYE, France

A. P. Martin is "Ingénieur" in the Délégation Générale pour l'Armement (DGA) of the French Department of Defense.

N. Petit is with the Centre Automatique et Systèmes, Mines-ParisTech, 60, bd St Michel, 75272 Paris, France

emerged. Some procedures and algorithm have been proposed (see [14], [8]) and a few for magnetometers calibration (see [10], [9]). They all rely on the fact that the force field under consideration (the gravitational field for accelerometers and the Earth magnetic field for magnetometers) corresponds to a sensed vector having, in theory, a constant and known norm. The strategy is to identify and remove the measurement errors. No specific calibration hardware is required.

The measurement errors are represented by constant coefficients of a (vector) affine transformation. The calibration problem is to find inverse affine transformations to maximize a performance index. This index involves the norm of the reconstructed data and a comparison against its theoretical (scalar) constant value. In practice, the usual algorithm (see [10], [9]) proceeds in two steps. First, an exact linearization is performed by means of a change of variables. Then, an inverse transformation is analytically (or numerically) performed to obtain the desired variables. Interestingly, the linearizing change of coordinates is not unique. Several choices are possible and all yield some distorsion in the cost function. For cases where the sensed field is possibly substantially distorted (such as the above mentioned indoor applications), irrelevant solutions can appear. They may correspond to cases where the cost function is also substantially distorted.

In this paper we propose to go back to an original nonlinear formulation, close in spirit with those considered in the calibration-table free methods (see [10], [9], [14], [8]), and to treat it in an iterative way. This yields an algorithm with interesting mathematical properties, which is also very efficient in practice. More specifically, we address the case of magnetometers which is of particular interest for the distributed magnetometry applications mentioned above, but the method has also been applied successfully to accelerometers.

The article is organized as follows. In Section I, the calibration problem is defined along with the mathematical model used. The main magnetic disturbances (hard-iron, soft-iron) are briefly recalled and modeled along with the classic scale factors and misalignments. Notations required for the study of the calibration algorithm are presented. We also briefly present the distributed magnetometry test bed experiments are conducted on. In Section II, the state-of-the-art non-linear two-steps estimation algorithm and the proposed method are described for sake of further comparisons. In Section III, properties of the proposed algorithm are established. Experimental results are presented in Section IV. Finally, we conclude and sketch future directions.

I. CALIBRATION PROBLEM

In this section, we expose the calibration problem for a three-axis sensor. We present an error model and the notations employed throughout the paper.

A. Notations

Consider a three-axis sensor. Its sampled measurements are denoted y_i (3×1 vector), where i stands for the sampling index, whereas the actual value of the sensed field is denoted Y_i (3×1 vector). These measurements are collected in $y = (y_i)_{i=1,...N}$, where N is the total number of samples.

In order to improve the accuracy of raw sensor data y, especially when dealing with low-cost sensors, mathematical models must be built to take into account the various sources of errors. Some, such as scale-factors, misalignment and the resulting cross-coupling of axes, apply to all kinds of sensors (gyroscopes, accelerometers, ...) while some others only apply to a particular class of sensor (see below for the case of magnetometers). To cover most cases of interest, we might have simply considered a zero-bias vector β, scale-factors represented by a diagonal matrix α, and misalignments terms (accounting for harmonization errors only) represented by a skew-symmetric matrix R

$$Y_i = \alpha R y_i + \beta \quad \text{with } \beta = [\beta_1 \ \beta_2 \ \beta_3]^T$$
$$R = \begin{bmatrix} 1 & \psi & -\theta \\ -\psi & 1 & \phi \\ \theta & -\phi & 1 \end{bmatrix} \alpha = \begin{bmatrix} \alpha_1 & & \\ & \alpha_2 & \\ & & \alpha_3 \end{bmatrix}$$

However, due to the use of low cost sensors, substantial misalignments and errors must be considered, and, additionally, significant non-orthogonality between axes may arise. Therefore, no possibly simplifying assumption on the magnitude of the errors is made. All these factors are gathered into a general matrix A, and a zero-bias vector B. With these notations,

$$Y_i = A y_i + B \tag{1}$$

In the case of magnetometers which is of particular interest here, the specific sources of errors are mainly hard and soft iron errors (see [10], [13]). Hard iron errors are induced by permanent unwanted fields. They are generated by ferromagnetic materials attached to the magnetometer frame (typically the structure or the equipment installed near the magnetometer, or even non-varying current in close-by wires). They result in a bias. Soft iron errors are induced by materials that generate magnetic fields in response to externally applied magnetic fields. The model presented in this paper takes only into account the proportionality of this error to the applied external field. The constant of proportionality is referred to as the magnetic susceptibility of the material considered. Soft iron errors generally present a hysteresis which is often small enough to be neglected.

Without any disturbances on the sensed field, the only information available is that the actual norm of the sensed field is constant during all the measurement acquisition phase. This constant may be chosen equal to 1 without any loss of generality[1]. The calibration problem consists in determining A and B from the measurements y knowing that the actual norm of the sensed field Y_i is 1 for all samples $i = 1, ..., N$.

---

[1]This standpoint is different from numerous approaches found in the literature [2] where the norm of the sensed field is obtained from dependable look-up tables. No such information is available indoor.
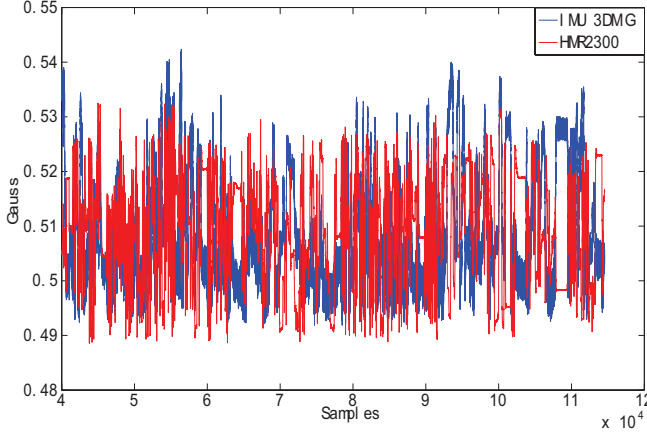
Fig. 1. Norm of the magnetic field before calibration for two distinct magnetometers (one HMR2300 from Honeywell® and one included in the Microstrain® IMU). The errors are due to the sensor ill-calibration.

B. Obtaining experimental data

Measurements are obtained while the three-axis sensor (magnetometer) is oriented in every possible direction. The motion is usually relatively slow, but this is not a strict requirement. Interestingly, no accurate measurements of the sensor orientation is made during this data acquisition. This makes the classically considered calibration table useless in this data collection step. The Earth magnetic field and the hard and soft iron errors are assumed constant during the data collection process. Similarly, for accelerometers calibration, the gravitational field would be also considered constant. To avoid any temperature drift, the sensors have to be warmed up before data acquisition. This warm-up phase last approximatively 1 min. The data acquisition phase lasts approximatively 25 min and a typical number of $N = 10^5$ samples are acquired.

C. Experimental testbed

The experimental testbed used here is described in [6]. In summary, nine magnetometers (e.g. HMR2300 from Honeywell®) are attached to a board which can simultaneously rotate and translate in 3D. A Microstrain® IMU is located in the center of the device. A power-PC microcontroller (MPC555 from Motorola®) is used to retrieve the data from all sensors and associate a time-stamp to each measurement for sake of synchronization. The measurement from all the sensors are gathered along with their timestamps in a single message sent through a serial port to a remote computer for post-treatment. Before the calibration process is achieved, the norm of the sensed magnetic fields varies significantly (see Figure 1 where the norms of the N samples are plotted).

II. CALIBRATION METHODS

A. Non-linear, two-step estimation algorithm

Here, we briefly recall the two-step calibration algorithm originally proposed in [10], [9]. As the norm of the sensed

field is assumed constant $\|Y_i\|^2 = 1$, for all $i = 1, ..., N$, the sensed field vectors y_i should all be located on the unit sphere. Due to errors, this is not the case, as already discussed. It is desired to find A and B such that the values $A(y_i - B)$ are as close as possible to this sphere. In details, A and B have to yield a fair approximation of $\|A(y_i - B)\|^2 = 1$ for every sample $i = 1, ..., N$. To this end, A and B are defined as the minimizers of the following cost function (2)

$$g(A, B, y) = \sum_{i=1}^{N} \left(\|A(y_i - B)\|^2 - 1 \right)^2 \qquad (2)$$

Once expanded, it becomes

$$g(A, B, y) =$$
$$\sum_{i=1}^{N} \left(y_i^T A^T A y_i - 2B^T A^T A y_i + (B^T A^T A B - 1) \right)^2$$

Here, A is sought after under the form of an upper triangular matrix, implicitly leaving out any rotation matrix which leaves the cost g invariant. The parameters to be determined through the minimization procedure are the components of A and B, i.e. 9 parameters. The cost function (2) is quartic in these parameters. Yet, as proposed in [10], [9], a two-step estimation using a least squares method can be performed. Note

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{pmatrix}, \quad B = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

First, the following change of variables is performed

$$\begin{cases} a = a_{11}^2 \\ b = a_{12}^2 + a_{22}^2 \\ c = a_{13}^2 + a_{23}^2 + a_{33}^2 \\ d = 2a_{11}a_{12} \\ e = 2a_{12}a_{13} + 2a_{22}a_{23} \\ f = 2a_{11}a_{13} \\ g = -2(ab_1 + db_2 + fb_3) \\ h = -2(db_1 + bb_2 + eb_3) \\ i = -2(fb_1 + eb_2 + cb_3) \\ j = ab_1^2 + bb_2^2 + cb_3^2 + 2db_1b_2 \\ \quad + 2eb_2b_3 + 2fb_1b_3 - 1 \end{cases}$$

These last 10 variables are normalized to 9 variables by considering the 9-dimensional vector of ratios $r = (a/b, c/b, d/b, e/b, f/b, g/b, h/b, i/b, j/b)$. Then, a new optimization problem (3) is formulated. This new (quadratic) problem is solved by a least-squares method. Finally, inverse algebraic transformations permit to recover the 9 coefficients of A and B from the 9 ratios. The optimization problem (3) is not equivalent to the minimization of the original cost function g in (2). The reason why is that, as appeared in the introduction of the vector or ratios r, the (not unique) normalization of the variables yielding the reduction to a

$$\min_r \sum_{i=1}^{N} \left\| \frac{y_{i1}^2}{y_{i2}^2} \cdot \frac{a}{b} + \frac{y_{i3}^2}{y_{i2}^2} \cdot \frac{c}{b} + \frac{y_{i1} \cdot y_{i2}}{y_{i2}^2} \cdot \frac{d}{b} + \frac{y_{i2} \cdot y_{i3}}{y_{i2}^2} \cdot \frac{e}{b} + \frac{y_{i3} \cdot y_{i1}}{y_{i2}^2} \cdot \frac{f}{b} + \frac{y_{i1}}{y_{i2}^2} \cdot \frac{g}{b} + \frac{y_{i2}}{y_{i2}^2} \cdot \frac{h}{b} + \frac{y_{i3}}{y_{i2}^2} \cdot \frac{i}{b} + \frac{1}{y_{i2}^2} \cdot \frac{j}{b} - 1 \right\|^2 \tag{3}$$
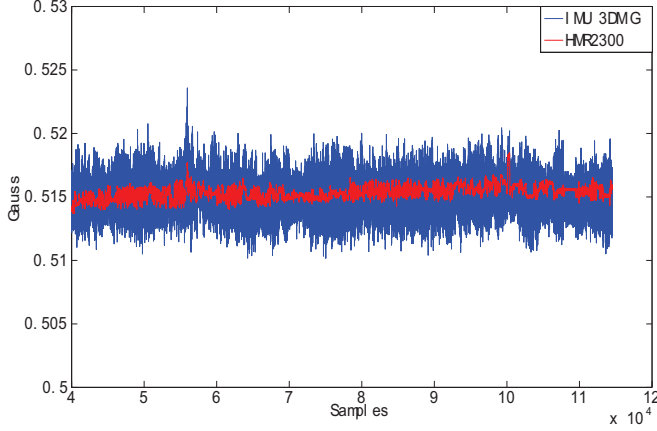

Fig. 2. Norm of the magnetic field calibrated by the non-linear, two-step algorithm presented in Section II-A.

Magneto IMU	Raw data	Calibrated data
Mean	$5.099327e-1$	$5.139927e-1$
Deviation	$1.077045e-2$	$1.223969e-3$
Samples	120000	120000
Time	-	$approx.1.2\ s$
HMR2300	Raw data	Calibrated data
Mean	$5.090592e-1$	$5.139977e-1$
Deviation	$9.846039e-3$	$6.878812e-4$
Samples	120000	120000
Time	-	$approx.1.2\ s$

Fig. 3. Raw data and calibrated data (through the 2-step algorithm) for two magnetometers.

classic least-squares problem, has introduced a non uniform weighting of the various measurements (see Equation 3). In theory, it would have been possible to consider any of the 9 variables a, b, ..., j to normalize the problem. Similarly, this would have led to nonequivalent optimization problems though. The original optimization problem with cost function (2) has thus been "distorted". In some cases this can be a problem. This point is illustrated later in this paper (see Section IV).

Results: Figure 2 presents results obtained with this algorithm applied on the raw data shown in Figure 1. The statistics of both raw and calibrated data sets of a magnetometer HMR2300 and of the magnetometer of the IMU are presented in Figure 3. The reported time is the time elapsed during the computation of the algorithm on a Intel Core 2 Duo 2.6GHz.

B. Proposed algorithm

Instead of considering linearizing changes of variables, we propose to solve an optimization problem by means of iterations of least square problems and successive partial calibration of data.

Consider a step in the iterations, say the kth. Following the idea of [10], [9], we wish to account for the fact that the sensed field is constant. Consider the N data $y_{i,k}$, $i = 1, ..., N$, which are initialized at step $k = 0$ with the measurements y_i. First, we formulate the following cost to be minimized

$$h(A, B, k) = \sum_{i=1}^{N} \left\| (A y_{i,k} + B) - \frac{y_{i,k}}{\|y_{i,k}\|} \right\|^2 \tag{4}$$

This function is quadratic with respect to the coefficients of A and B. In view of algorithmic minimization, this is an advantage over the cost in (2) which is quartic with respect to these same variables. We note the uniquely defined solution

$$(A_{k+1}, B_{k+1}) = \arg\min_{A,B} h(A, B, k)$$

which can be obtained by a classic least-squares approach. Then, we use these matrices to update the data as follows

$$y_{i,k+1} = A_{k+1} y_{i,k} + B_{k+1} \tag{5}$$

After k such iterations, a matrix $\tilde{A}_k$ and a vector $\tilde{B}_k$ are obtained recursively by

$$\tilde{A}_k = A_k \tilde{A}_{k-1}$$
$$\tilde{B}_k = A_k \tilde{B}_{k-1} + B_k$$

They relate $y_{i,k}$ to the raw measurements y_i. In details,

$$y_{i,k} = \tilde{A}_k y_i + \tilde{B}_k \tag{6}$$

We can now summarize the method
Algorithm 1 (Proposed algorithm):
1) Initialize $k = 0$, $y_{i,0} = y_i$ for all $i = 1, ..., N$
2) Compute $(A_{k+1}, B_{k+1}) = \arg\min_{A,B} h(A, B, k)$ by a least-squares method (where h is given in Equation 4)
3) Update the data $y_{i,k+1} = A_{k+1} y_{i,k} + B_{k+1}$
4) Increase k by 1 and return to step 2

A limited number K of iterations is usually considered. Then the data $y_{i,K}$, $i = 1, ..., N$ are the "calibrated data".

In words, this algorithm solves a sequence of least square problems in which the input data are iteratively calibrated using the successively determined calibration matrices and vectors.

In this method, a very natural cost function (4) is formulated. It is close in spirit to (2), but can be handled differently. As will now appear, this also brings some interesting properties.

III. PROPERTIES OF THE PROPOSED ALGORITHM

A. Calibration improvement and alignment

We now prove two properties of the proposed algorithm 1. For sake of conciseness, we note $y_k = (y_{i,k})_{i=1,\ldots,N}$.

Proposition 1 (calibration improvement): The data generated by the Algorithm 1 satisfy the following decreasingness property:

$$f(y_{k+1}) \leq f(y_k)$$

where

$$f(y_k) = \sum_{i=1}^{N} (1 - \|y_{i,k}\|)^2$$

Further, $(f(y_k))_{k\in\mathbb{N}}$ decreases and is positive, so it converges to a limit $\ell \geq 0$.

This property shows that, in the sense detailed by the function f, the data are better and better calibrated as the iterations are pursued.

Proof: By construction, A_k and B_k are such that $h(A_k, B_k, k)$ is minimal. In particular, one can compare them against the identity matrix and the zero vector

$$h(A_{k+1}, B_{k+1}, k) \leq h(I_3, 0, k)$$

Replacing $A_{k+1}y_{i,k} + B_{k+1}$ by $y_{i,k+1}$ yields

$$\sum_{i=1}^{N} \left\| y_{i,k} - \frac{y_{i,k}}{\|y_{i,k}\|} \right\|^2 \geq \sum_{i=1}^{N} \left\| y_{i,k+1} - \frac{y_{i,k}}{\|y_{i,k}\|} \right\|^2 \quad (7)$$

$$\sum_{i=1}^{N} \left(y_{i,k} \cdot \left(1 - \frac{1}{\|y_{i,k}\|} \right) \right)^2 \geq \sum_{i=1}^{N} \left\| y_{i,k+1} - \frac{y_{i,k}}{\|y_{i,k}\|} \right\|^2$$

which gives, by triangle inequality on the right term,

$$\sum_{i=1}^{N} (1 - \|y_{i,k}\|)^2 \geq \sum_{i=1}^{N} \left(\|y_{i,k+1}\| - \left\| \frac{y_{i,k}}{\|y_{i,k}\|} \right\| \right)^2 \quad (8)$$

Finally,

$$f(y_k) \geq f(y_{k+1})$$

which concludes the proof. ■

Proposition 2 (alignment): The data generated by Algorithm 1 satisfy the following alignment property

$$\lim_{k \to +\infty} \left(\frac{y_{i,k+1}}{\|y_{i,k+1}\|} - \frac{y_{i,k}}{\|y_{i,k}\|} \right) = 0$$

This property shows that, as the iterations are pursued, the calibrated data make little progress in orientation.

Proof: First, we perform a preliminary decomposition of the objective function h

$$h(k) \triangleq h(A_{k+1}, B_{k+1}, k)$$
$$= \sum_{i=1}^{N} \left(\|y_{i,k+1}\|^2 + 1 - 2\frac{\langle y_{i,k+1} | y_{i,k} \rangle}{\|y_{i,k}\|} \right)$$

which can also be written under the form

$$h(k) = f(y_{k+1}) + \sum_{i=1}^{N} \left(2\|y_{i,k+1}\| - 2\frac{\langle y_{i,k+1}|y_{i,k} \rangle}{\|y_{i,k}\|} \right) \quad (9)$$

$$= f(y_{k+1}) + 2\sum_{i=1}^{N} \|y_{i,k+1}\| \left(1 - \left\langle \frac{y_{i,k+1}}{\|y_{i,k+1}\|} \Big| \frac{y_{i,k}}{\|y_{i,k}\|} \right\rangle \right) \quad (10)$$

Consider again (7) and (8), one obtains

$$f(y_k) \geq h(k) \geq f(y_{k+1})$$

From Proposition 1, we know that $f(y_k)$ converges to a limit ℓ as $k \to +\infty$. Therefore, from the preceding inequalities, we conclude that $h(k)$ converges to the same limit. Then, from Equation 9, we deduce

$$\lim_{k \to +\infty} \left(\sum_{i=1}^{N} \|y_{i,k+1}\| \left(1 - \left\langle \frac{y_{i,k+1}}{\|y_{i,k+1}\|} \Big| \frac{y_{i,k}}{\|y_{i,k}\|} \right\rangle \right) \right) = 0 \quad (11)$$

All the terms under the $\sum$ sign are positive or zero, therefore we conclude that, $\forall i = 1, \ldots, N$,

$$\lim_{k \to +\infty} \left(\|y_{i,k+1}\| \left(1 - \left\langle \frac{y_{i,k+1}}{\|y_{i,k+1}\|} \Big| \frac{y_{i,k}}{\|y_{i,k}\|} \right\rangle \right) \right) = 0 \quad (12)$$

$$\lim_{k \to +\infty} \left(\|y_{i,k+1}\| \left\langle \frac{y_{i,k+1}}{\|y_{i,k+1}\|} \Big| \left(\frac{y_{i,k+1}}{\|y_{i,k+1}\|} - \frac{y_{i,k}}{\|y_{i,k}\|} \right) \right\rangle \right) = 0 \quad (13)$$

Now, to conclude, note

$$u_{i,k} = \frac{y_{i,k+1}}{\|y_{i,k+1}\|} - \frac{y_{i,k}}{\|y_{i,k}\|}$$

and decompose it under the form

$$u_{i,k} = (1 - \alpha_{i,k})\frac{y_{i,k+1}}{\|y_{i,k+1}\|} - \beta_{i,k}\frac{y_{i,k+1}^{\perp}}{\|y_{i,k+1}^{\perp}\|}$$

where $y_{i,k+1}^{\perp}$ is directly orthogonal to $y_{i,k+1}$, and with

$$\alpha_{i,k}^2 + \beta_{i,k}^2 = 1$$

We deduce from (13) that

$$\lim_{k \to +\infty} \alpha_{i,k} = 1, \qquad \lim_{k \to +\infty} \beta_{i,k} = 0$$

Finally, this gives

$$\lim_{k \to +\infty} \left(u_{i,k} = \frac{y_{i,k+1}}{\|y_{i,k+1}\|} - \frac{y_{i,k}}{\|y_{i,k}\|} \right) = 0$$

which concludes the proof. ■

B. Illustrative example

To illustrate the presented algorithm, we use it on a very small set of 12 planar data. Initially, they are all lying on a sharp ellipse (plotted in red in Figure 4). This scenario corresponds to strong scale factors, moderate bias, and misalignment. The algorithm is run over 4 iterations. Results are reported on top of each other in Figure 4. As can be observed, the data are quickly calibrated, i. e. they all get close to the unit circle within this small number of iterations.
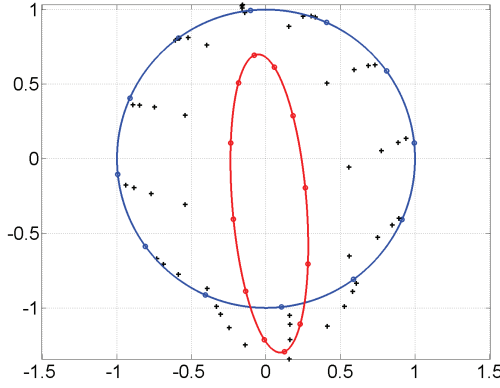
Fig. 4. Evolution of the calibration over 4 iterations of the proposed algorithm. Measurements are in red and the unit circle in blue. The four iterations of the calibration algorithm are represented by black crosses.

C. A special case: 2 dimensional unbiased calibration problem

The previously presented propositions provide theoretical insight into the progression of the algorithm along the iterations: the data are better and better calibrated (Proposition 1), and make few progress in rotation (Proposition 2). It is in fact possible to go a little bit further in the analysis. Let us now focus on a specific 2 dimensional case (i. e. where the measurements are done in 2D), and further, let us make the following simplifying assumptions. We consider that the measurements constitute a closed and centered ellipse (i.e. that there is a continuum of data, and that a preliminary debiasing has been performed). The measurements are noted $y(\theta)$, $\theta \in [0, 2\pi[$, and the iteratively calibrated data are $y_k(\theta)$, $\theta \in [0, 2\pi[$. Then, we assume that no misalignment is present and consider only scale factors. Therefore, the measurements satisfy

$$Y(\theta) = Ay(\theta)$$

where A is an unknown diagonal matrix. By analogy to equation (4), the cost to be minimized at iteration k is

$$h(A, k) = \int_0^{2\pi} \left\| Ay_k(\theta) - \frac{y_k(\theta)}{\|y_k(\theta)\|} \right\| d\theta$$

Note A_{k+1} the solution of this minimization problem, i.e.

$$A_{k+1} = \arg \min_{A \text{ diagonal}} h(A, k)$$

Iteratively, the data are calibrated using $y_{k+1}(\theta) = A_{k+1}y_k(\theta)$. At step k, let us note the inverse calibration equation

$$y_k(\theta) = \begin{pmatrix} \alpha_k & 0 \\ 0 & \beta_k \end{pmatrix} Y(\theta)$$

With the notation

$$f(y_k) = \int_0^{2\pi} \left| 1 - \|y_k(\theta)\|^2 \right| d\theta$$

a reasoning similar to the one in the proof of Proposition 2 directly yields

$$f(y_k) \geq h(A_{k+1}, k) \geq f(y_{k+1})$$

Further,

$$f(y_k) = 2\pi \left(1 + \frac{\alpha_k^2}{2} + \frac{\beta_k^2}{2} \right) - 2 \int_0^{2\pi} \|y(\theta)\| d\theta$$
$$= 2\pi \left(1 + \frac{\alpha_k^2}{2} + \frac{\beta_k^2}{2} \right) - 2P(\alpha, \beta) \tag{14}$$

where $P(\alpha, \beta)$ is the perimeter of an ellipse having $1/\alpha_k$ and $1/\beta_k$ as semi-axis. We wish to show that the inverse calibration matrix $\begin{pmatrix} \alpha_k & 0 \\ 0 & \beta_k \end{pmatrix}$ tends to the identity as k tends to infinity, i.e. that the proposed algorithm converges to exact calibration of the data. Similarly to Proposition 1, one can readily show that f decreases along the iterations and goes to a limit $\ell \geq 0$. Using an estimate for (14), it is possible to deduce convergence information on α_k and β_k. To simplify the exposition, let us first consider that this limit is $\ell = 0$.

Consider in equation (14), f as a function of (α_k, β_k). It can be proved that f has $(1, 1)$ as unique local minimum (in a rather large neighborhood) and that its value there is 0. To prove that, one can simply use Peano's approximation of the perimeter of an ellipse [1] reproduced in (17). Accounting for the approximation implied by this formula, one can compute the following exact decomposition of f under the form

$$f(\alpha_k, \beta_k) = f_1(\alpha_k, \beta_k) + f_2(\alpha_k, \beta_k)$$

where $f_2(\alpha_k, \beta_k)$ is strictly positive away from $\alpha_k = \beta_k$ and zero there, while f_1, given in (18), is convex on the considered domain $[3/4 \ 4/3]^2$ (its Hessian is given in (19)), strictly positive away from $(1, 1)$ and zero there. Therefore, $(1, 1)$ is the only zeroing point of f, and one can conclude that (α_k, β_k) converges to $(1, 1)$. This estimate reveals handy in experimental results where ℓ can be evaluated numerically.

Now, let us extend the analysis to the case $\ell > 0$. A local expansion of f for (α, β) about $(1, 1)$ yields the following inequalities

$$\frac{\begin{pmatrix} \alpha - 1 & \beta - 1 \end{pmatrix} \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} \alpha - 1 \\ \beta - 1 \end{pmatrix}}{16} \leq \ell$$
$$\ell \leq \frac{\begin{pmatrix} \alpha - 1 & \beta - 1 \end{pmatrix} \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} \alpha - 1 \\ \beta - 1 \end{pmatrix}}{4}$$

and we deduce the estimation, where d is the distance of (α, β) to $(1, 1)$,

$$\frac{\sqrt{2}}{4}d \leq \sqrt{\ell} \leq d \tag{15}$$

meaning that both α_k and β_k approach 1 as the square root of ℓ approaches 0.
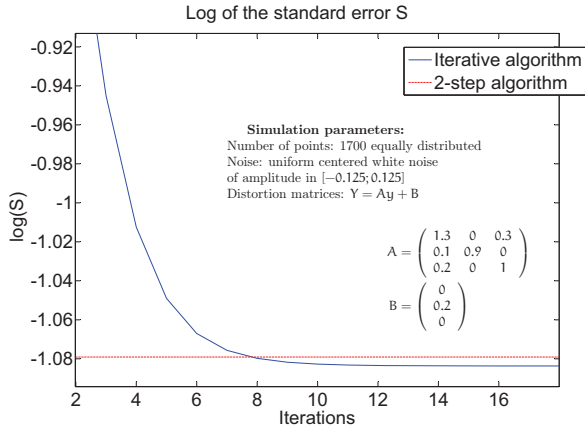
Fig. 6. Evolution of the standard error S over a few iterations of the iterative algorithm proposed and comparison with the results obtained with the 2-step algorithm

Fig. 7. Erratic behavior. A set of 1700 perfectly scaled unbiased 3D data is considered, and a strong misalignment which is varies from 0 to 40 is introduced. Mean and standard error S are given for both the 2-step algorithm (in red) and the proposed one (in blue).

IV. EXPERIMENTAL RESULTS

Several experiments have been conducted using the hardware presented in Section I-C. To evaluate the performance of the algorithms, the standard error S is used.

$$S(y) = \frac{1}{N-1} \cdot \sum_{i=1}^{N} (\|y_i\| - 1)^2 \qquad (16)$$

Table 5 shows some results obtained with simulated data and with less disturbed real data from a magnetometer (HMR2300 from Honeywell®). It appears that the calibration process allows significant improvement compared to the raw data. The standard error S used to evaluate the various algorithms is better for the proposed iterative one. When cross-coupling terms are small, on the one hand, improvement brought by this algorithm get smaller, but, on the other hand, fewer iterations are needed and computation time is faster.

To underline the possibly erratic behavior of the 2-step algorithm in extreme cases (see Section II-A), we consider a set of 1700 perfectly scaled unbiased 3D data, and introduce a strong misalignment which is varied from 0 to 40. As can be seen in Figure 7, when the misalignment term (one of the upper triangular part of matrix A) reaches the values of 20 (approximately), the 2-step algorithm ends up with ill-calibrated data. Both the standard error and the mean value are inconsistent with reality. Interestingly, the proposed algorithm keeps working properly.

Finally, accuracy of the proposed method can be investigated. By increasing the number of iterations, the accuracy is enhanced. This point is visible in Figure 6. In this plot, the log of the standard error is plotted for various numbers of iterations, giving an idea of the rate of convergence. It appears in the considered scenario (1700 simulated data with matrix and bias reported in the figure), that the proposed algorithm outperforms the 2-step method as soon as the number of iterations is larger than 10.
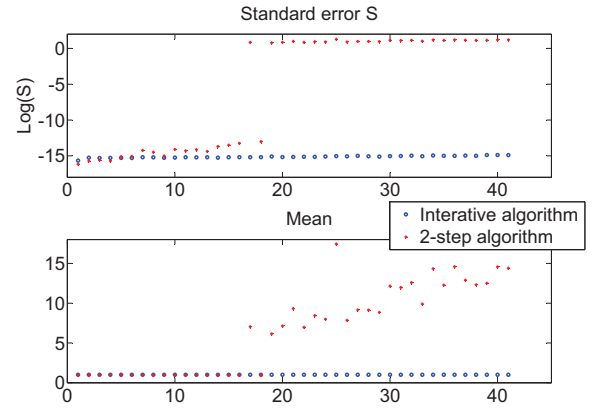
V. CONCLUSION

In this paper, we have introduced a new algorithm for calibrating three-axis sensors. This iterative method makes a repeating use of least-squares algorithm. As has been demonstrated, it proves very effective when treating large sets of uncalibrated data. Certainly, the numerical efficiency of the method can be improved upon, by reusing key information from one iteration to the next for example. We also believe that more theoretical results can be obtained. Several points remains to be explored, in particular the magnitude of the residual error seems to be possible to estimate. Finally, an extension of the proposed algorithm is currently considered. It aims at calibrating *at once* an array of sensors by including corrections of the *misalignment between the sensors* measuring the same force field. These are current directions of future work.

REFERENCES

[1] R. W. Barnard, K. Pearce, and L. Schovanec. Inequalities for the perimeter of an ellipse. *Journal of Mathematical Analysis and Applications*, 260(2):295 – 306, 2001.
[2] C. Barton. Revision of International Geomagnetic Reference Field Release. *EOS Transactions*, 77, April 1996.
[3] P. Castillo, A. Dzul, and R. Lozano. Real-time stabilization and tracking of a four rotor mini rotorcraft. *IEEE Trans. Control Systems Technology*, 12(4):510–516, 2004.
[4] S. Changey, D. Beauvois, and V. Fleck. A mixed extended-unscented filter for attitude estimation with magnetometer sensor. In *Proc. of the 2006 American Control Conference*, 2006.
[5] A. B. Chatfield. *Fundamentals of High Accuracy Inertial Navigation*, volume 174 of *Progress in Astronautics and Aeronautics Series*. AIAA, 1997.
[6] E. Dorveaux, D. Vissière, A. P. Martin, and N. Petit. Time-stamping for an array of low-cost sensors. In *Proc. of the PDES 2009, IFAC workshop on Programmable Devices and Embedded Systems*, 2009.
[7] P. Faurre. *Navigation inertielle et filtrage stochastique*. Méthodes mathématiques de l'informatique. Dunod, 1971.
[8] W. T. Fong, S. K. Ong, and A. Y. C. Nee. Methods for in-field user calibration of an inertial measurement unit without external equipment. *Measurement Science and Technology*, 19(8):085202–+, Aug. 2008.

Simulated data	(see Figure 6)		
	Mean	Standard error	Samples
Raw data	1.01	$2.01e-1$	1700
2-step algorithm	0.983	$8.34e-2$	1700
Proposed algorithm (K=20)	0.993	$8.25e-2$	1700
Real magnetometer data	(HMR2300)		
	Mean	Standard error	Samples
Raw data	1.016391	$2.040928e-2$	80000
2-step algorithm	0.9999951	$1.397928e-3$	80000
Proposed algorithm (K=40)	0.9999980	$1.397917e-3$	80000

Fig. 5.   Numerical calibration results for a set of simulated data and a set of experimental measurements.

[9] C. Foster and G. Elkaim.   Extension of a non-linear, two-step calibration methodology to include non-orthogonal sensor axes. In *IEEE Journal of Aerospace Electronic Systems*, volume 44, July 2008.

[10] D. Gebre-Egziabher, G. Elkaim, J. Powell, and B. Parkinson. A non-linear, two-step estimation algorithm for calibrating solid-state strapdown magnetometers. In *8th International St. Petersburg Conference on Navigation Systems (IEEE/AIAA)*, May 2001.

[11] M. S. Grewal, L. R. Weill, and A. P. Andrews. *Global positioning systems, inertial navigation, and integration*.   Wiley Inter-science, 2001.

[12] R. Mahony, T. Hamel, and J.-M. Pflimlin. Complementary filter design on the special orthogonal group SO(3). In *Proc. of the 44th IEEE Conf. on Decision and Control, and the European Control Conference 2005*, 2005.

[13] P. Ripka, editor. *Magnetic Sensors and Magnetometers*. ed. Boston: ARTECH HOUSE, 1. edition, 2001.

[14] I. Skog and P. Händel.   Calibration of a mems inertial measurement unit.   In *XVII IMEKO World Congress Metrology for a Sustainable Development*, Sept. 2006.

[15] D. Vissière, P.-J. Bristeau, A. P. Martin, and N. Petit. Experimental autonomous flight of a small-scaled helicopter using accurate dynamics model and low-cost sensors. 2008.

[16] D. Vissière, A. P. Martin, and N. Petit. Using magnetic disturbances to improve IMU-based position estimation. In *Proc. of the 9th European Control Conf.*, 2007.

APPENDIX

Peano's approximation of the perimeter of the ellipse is always over-estimating the true value. Here is the approximation:

$$P(\alpha, \beta) \approx \pi \left(\frac{3}{2} (\alpha + \beta) - \sqrt{\alpha\beta} \right) \tag{17}$$

which yields to

$$f_1(\alpha, \beta) = 2\pi \left(1 + \frac{\alpha^2}{2} + \frac{\beta^2}{2} \right) - 2\pi \left(\frac{3}{2} (\alpha + \beta) - \sqrt{\alpha\beta} \right) \tag{18}$$

The hessian of f_1 is the following one

$$\frac{\nabla^2 f_1(\alpha, \beta)}{2\pi} = \begin{pmatrix} 1 - \frac{\beta}{4\alpha\sqrt{\alpha\beta}} & \frac{1}{4\sqrt{\alpha\beta}} \\ \frac{1}{4\sqrt{\alpha\beta}} & 1 - \frac{\alpha}{4\beta\sqrt{\alpha\beta}} \end{pmatrix} \tag{19}$$