22nd International Congress of Mechanical Engineering (COBEM 2013)
November 3-7, 2013, Ribeirão Preto, SP, Brazil

GREEDY STATISTICALLY CORRECT SIMULTANEOUS CALIBRATION OF MAGNETOMETERS AND ACCELEROMETERS

Conrado Silva Miranda
Janito Vaqueiro Ferreira
SCHOOL OF MECHANICAL ENGINEERING - UNICAMP, CAMPINAS - SP - BRASIL
{cmiranda,janito}@fem.unicamp.br

Abstract. *This work develops an algorithm that maximizes the likelihood of the sensor readings to correctly calibrate the gain matrix and bias of a magnetometer and an accelerometer. Most calibration algorithms rely on statistical approximations of the sensor noise, which may lead to inaccurate results. With the assumption of Gaussian noise with full covariance matrix, our algorithm uses a set of standing still sensors readings to iteratively optimize the sensors parameters, field intensity and current rotation. The results show that the reconstruction error median is less than 0.1 standard deviations from the correct value for reasonable number of samples and with almost constant time.*

Keywords: *sensor calibration, magnetometer, accelerometer*

1. INTRODUCTION

The sensors reading represents a fundamental step in the control loop of a robot, but this reading may provide values different from the correct ones due to sensors miscalibration. In the aerial vehicles setting, it's common to use accelerometers and magnetometers to estimate the system's attitude (Lizarraga *et al.*, 2013). Due to their similarities, most focus has been given to the magnetometer calibration, which has been extensively researched. While there are calibration algorithms using filters (Sabatini, 2006) or recursion (Pylvanainen, 2008), we'll focus here on two well established algorithms that resemble our own.

Alonso and Shuster (2002) developed an extension to the TWOSTEP algorithm that calibrates a magnetometer by maximizing the likelihood of a set of readings adjusting the bias and gain matrix estimates. This algorithm relies on minimizing the difference between the norm of each sensor reading with the known magnetic field's norm during each reading. Being independent of the current system's attitude, the readings may be captured during flight without prejudicing to the results. Crassidis *et al.* (2005) developed an extension to the algorithm that can be performed online.

Vasconcelos *et al.* (2008) proved that noiseless magnetometer readings fit perfectly to an ellipsis and, based on this result, developed a calibration algorithm for the sensor. Similarly to the previous one, this algorithm doesn't depend on the current system's attitude and may be used while the system is moving. However, the covariance model is more restrictive, allowing it to be only a product between a constant and the identity matrix.

Both algorithms correctly estimate the bias and gain matrix for the magnetometer and may be used for the accelerometer. However, as the calibrations must be performed independently, each sensor will be calibrated in its own sensor frame. Moreover, as the current attitude is unknown, the algorithms aren't able to estimate the direction of the magnetic field, which may be unknown and is essential to correctly estimate the attitude using a magnetometer (Sabatini, 2006; Gebre-Egziabher *et al.*, 2000; Madgwick *et al.*, 2011).

By creating sets of samples with the same attitude and assuming that the gravitational and the magnetic fields aren't collinear, we've developed an algorithm that simultaneously calibrates an accelerometer and a magnetometer without making any approximations. It's important to note that the attitudes' restriction only means that the algorithm may not be used when the system is operating in a non-stationary regime. The algorithm also computes the attitude at each set of readings and estimates the magnetic field's direction. To the best of our knowledge, this is the most complete and statistically correct calibration algorithm for an accelerometer and a magnetometer.

The paper is organized as follows. In Sections 2 and 3 we describe the sensor model used in this paper and how the gathered data is preprocessed, respectively. Using the sensor model, we develop a cost function and describe its optimization in Section 4. We describe the complete algorithm in Section 5, and we present the experimental study performed in this paper and its analysis in Section 6. Finally, we provide concluding remarks and future research directions in Section 7.

2. SENSOR MODEL

In this paper, we'll consider a linear sensor model affected by a Gaussian noise, as it's a common assumption (Alonso and Shuster, 2002; Vasconcelos *et al.*, 2008). Let $\mathbf{s}_s(t)$ be the sensor s reading at time t and $\mathbf{s}_s^R(t)$ the true value measured

by the sensor, such as gravity. Then their relationship may be expressed as

$$\mathbf{s}_s(t) = \mathbf{K}_s \mathbf{s}_s^R(t) + \mathbf{b}_s + \epsilon_s, \quad \epsilon_s \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_s) \tag{1}$$

where $\mathbf{K}_s$, $\mathbf{b}_s$ and $\mathbf{\Sigma}_s$ are the sensor's gain matrix, bias and covariance matrix, respectively. We'll consider here that the sensor index s can be a for the accelerometer or m for the magnetometer, but they may assume different values if more sensors are calibrated simultaneously.

Assuming there's no disturbance in the sensors other than the Gaussian noise, a correct reading $\mathbf{s}_s^R(t)$ is given by a rotation of the nominal field for each sensor from the inertial frame $\mathcal{I}$, where the fields are defined, to the sensor frame $\mathcal{S}$, where the calibration occurs. As the inertial frame $\mathcal{I}$ in which these fields are defined is arbitrary, we'll consider the one where the gravitational field $\mathbf{g}$ and the magnetic field $\mathbf{h}$ can be written as

$$\mathbf{g} = \begin{bmatrix} 0 & 0 & g_z \end{bmatrix}^\top, \quad \mathbf{h} = \begin{bmatrix} h_x & 0 & h_z \end{bmatrix}^\top, \quad g_z < 0,\ h_x > 0 \tag{2}$$

where g_z, h_x and h_z are the fields' components and we assume that $\mathbf{g}$ and $\mathbf{h}$ aren't collinear, that is, the magnetic field has a component in the x direction. If they are collinear, then the magnetometer doesn't provide enough new information for the calibration and doesn't allow a unique definition of $\mathcal{I}$.

Once the fields are defined, we may write the correct reading as $\mathbf{s}_s^R(t) = \mathbf{R}(t)\mathbf{s}_s^\mathcal{I}$, where $\mathbf{R}(t)$ is the current rotation between the inertial frame $\mathcal{I}$ and the sensor frame $\mathcal{S}$, and $\mathbf{s}_s^\mathcal{I}$ is the reference field for the sensor s expressed in $\mathcal{I}$, i.e., $\mathbf{g}$ and $\mathbf{h}$ for an accelerometer and a magnetometer, respectively.

With the sensor model given in Eq. (1), the probability of a certain reading $\mathbf{s}_s$ given the sensor's parameters is defined as

$$p(\mathbf{s}_s; \mathbf{\Sigma}_s, \mathbf{K}_s, \mathbf{b}_s, \mathbf{R}, \mathbf{s}_s^\mathcal{I}) = (2\pi)^{-\frac{3}{2}} |\mathbf{\Sigma}_s|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\left(\mathbf{K}_s\mathbf{R}\mathbf{s}_s^\mathcal{I} + \mathbf{b}_s - \mathbf{s}_s\right)^\top \mathbf{\Sigma}_s^{-1}\left(\mathbf{K}_s\mathbf{R}\mathbf{s}_s^\mathcal{I} + \mathbf{b}_s - \mathbf{s}_s\right)\right), \tag{3}$$

which will be used to estimate the sensor parameters.

3. DATA CAPTURE AND PREPROCESSING

For the data capture, consider that the sensors were held undisturbed for a total of N different orientations and, at each orientation $i \in \{1, 2, \ldots, N\}$, the sensor s captured $\Delta_s[i]$ samples. Using the sensor model defined in Eq. (1), we have

$$\begin{aligned} &\mu_s[i] = \mathbf{K}_s\mathbf{R}_i\mathbf{s}_s^\mathcal{I} + \mathbf{b}_s \\ &\mathbf{s}_s[i,j] \sim \mathcal{N}(\mu_s[i], \mathbf{\Sigma}_s), \quad j \in \{1, 2, \ldots, \Delta_s[i]\} \end{aligned} \tag{4}$$

where $\mathbf{s}_s[i,\cdot]$ are the samples collected at orientation i and $\mu_s[i]$ is the reference value for that orientation, which is constant for all $\Delta_s[i]$ samples collected.

As the amount of data collected this way may be huge and doesn't provide much information, we'll summarize them for further processing. The uniformly minimum variance unbiased estimator (Krishnamoorthy, 2010) for the reference value $\mu_s[i]$ is given by

$$\hat{\mu}_s[i] = \frac{1}{\Delta_s[i]} \sum_{j=1}^{\Delta_s[i]} \mathbf{s}_s[i,j] \tag{5}$$

and its value is a sample from a normal distribution with reduced covariance, so that

$$\hat{\mu}_s[i] \sim \mathcal{N}\left(\mu_s[i], \frac{\mathbf{\Sigma}_s}{\Delta_s[i]}\right) \tag{6}$$

Although we may iterate the estimation of $\mathbf{\Sigma}_s$ with the estimation of other sensor's parameters, we've noticed that it not only didn't improve performance significantly but also decreased it substantially most of the time. We've thus decided to use a single estimate during the whole process, which is given by

$$\hat{\mathbf{\Sigma}}_s = \frac{\sum_{i=1}^{N}\sum_{j=1}^{\Delta_s[i]} \left(\mathbf{s}_s[i,j] - \hat{\mu}_s[i]\right)\left(\mathbf{s}_s[i,j] - \hat{\mu}_s[i]\right)^\top}{\left(\sum_{i=1}^{N} \Delta_s[i]\right) - N} \tag{7}$$

and is the uniformly minimum variance unbiased estimator for the covariance matrix $\mathbf{\Sigma}_s$ when we generalize Bessel's correction (Kenney and Keeping, 1957).

As we don't have access to the correct matrix $\mathbf{\Sigma}_\mathbf{s}$, we'll use the estimate given by Eq. (7) as the true value during the other parameters' estimation.

4. COST FUNCTION OPTIMIZATION

Given a set of parameters

$$\Theta = \{\mathbf{K}_a, \mathbf{b}_a, \mathbf{g}, \mathbf{K}_m, \mathbf{b}_m, \mathbf{h}, \mathbf{R}_i\}$$

and the data $\hat{\mu}_s[i]$ and $\hat{\mathbf{\Sigma}}_s$ computed by Eq. (5) and Eq. (7), respectively, considering the model given by Eq. (6), and assuming that $\hat{\mathbf{\Sigma}}_s$ is the correct covariance, the parameters' likelihood is given by

$$L(\Theta; \hat{\mu}_a, \hat{\mu}_m, \hat{\mathbf{\Sigma}}_a, \hat{\mathbf{\Sigma}}_m) = \prod_{i=1}^{N} \prod_{s\in\{a,m\}} p\left(\hat{\mu}_s[i]; \frac{\hat{\mathbf{\Sigma}}_s}{\Delta_s[i]}, \mathbf{K}_s, \mathbf{b}_s, \mathbf{R}_i, \mathbf{s}_s^{\mathcal{I}}\right)$$

where $p(\cdot)$ is given by Eq. (3).

Using this likelihood, we define the cost function as $J = -\log L$, and the calibration objective becomes minimizing J, which is a method as maximum likelihood estimation. Ignoring the constants and irrelevant multiplicative terms, we can write J as

$$J(\Theta; \hat{\mu}_a, \hat{\mu}_m, \hat{\mathbf{\Sigma}}_a, \hat{\mathbf{\Sigma}}_m) = \sum_{i=1}^{N} \sum_{s\in\{a,m\}} \Delta_s[i] \left(\mathbf{K}_s\mathbf{R}_i\mathbf{s}_s^{\mathcal{I}} + \mathbf{b}_s - \hat{\mu}_s[i]\right)^\top \hat{\mathbf{\Sigma}}_s^{-1} \left(\mathbf{K}_s\mathbf{R}_i\mathbf{s}_s^{\mathcal{I}} + \mathbf{b}_s - \hat{\mu}_s[i]\right) \tag{8}$$

Note that the accelerometer and magnetometer calibrations are related only by the rotations $\mathbf{R}_i$ and each rotation depends on a single set of measurements.

4.1 Rotation estimation

Consider that every parameter in Θ is known, except for the rotations $\mathbf{R}_i$. The cost function, given by Eq. (8), may be decomposed in N isolated equations, one for each rotation. These equations are given by

$$J_i(\mathbf{R}_i) = \sum_{s\in\{a,m\}} \Delta_s[i] \left(\mathbf{K}_s\mathbf{R}_i\mathbf{s}_s^{\mathcal{I}} + \mathbf{b}_s - \hat{\mu}_s[j]\right)^\top \hat{\mathbf{\Sigma}}_s^{-1} \left(\mathbf{K}_s\mathbf{R}_i\mathbf{s}_s^{\mathcal{I}} + \mathbf{b}_s - \hat{\mu}_s[j]\right) \tag{9}$$

which are nonlinear in $\mathbf{R}_i$ and thus subject to local minima. As we currently lack a better proposal to solve this, we express $\mathbf{R}_i$ in quaternion form (Chou, 1992) and optimize its values directly through gradient descent.

4.2 Bias and fields estimation

Consider now that only the biases $\mathbf{b}_a$ and $\mathbf{b}_m$ and the field parameters g_z, h_x and h_z are unknown among the parameters in Θ. We'll use the generalized least square method (Kariya and Kurata, 2004) to perform this optimization.

The generalized least squares (GLS) problem may be described as follows: given two sets of observations $\mathbf{y}$ and $\mathbf{X}$ and a matrix $\mathbf{\Omega}$ such that $\mathbf{y}$ may be written as

$$\mathbf{y} = \mathbf{X}\beta + \epsilon, \quad \mathrm{E}[\epsilon|\mathbf{X}] = \mathbf{0}, \quad \mathrm{Var}[\epsilon|\mathbf{X}] = \mathbf{\Omega}, \tag{10}$$

find the set of parameters β that best fits the observations. The best estimator for β is given by

$$\hat{\beta} = \left(\mathbf{X}^\top\mathbf{\Omega}^{-1}\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{\Omega}^{-1}\mathbf{y} \tag{11}$$

Using the models in Eq. (4) and Eq. (6), we see that each accelerometer summarized reading $\hat{\mu}_a[i]$ may be written as

$$\hat{\mu}_a[i] = \mathbf{K}_a\mathbf{R}_i\vec{\mathbf{z}}g_z + \mathbf{b}_a + \epsilon_a[i], \quad \epsilon_a[i] \sim \mathcal{N}\left(\mathbf{0}, \frac{\hat{\mathbf{\Sigma}}_a}{\Delta_a[i]}\right) \tag{12}$$

where $\vec{\mathbf{z}}$ is the unitary vector in the z direction.

Writing the set of equations given by Eq. (12) in the form used by the GLS problem in Eq. (10), we have

$$\mathbf{y}_a = \begin{bmatrix} \hat{\mu}_a[1] \\ \hat{\mu}_a[2] \\ \vdots \\ \hat{\mu}_a[N] \end{bmatrix}, \; \mathbf{X}_a = \begin{bmatrix} \mathbf{K}_a\mathbf{R}_1\vec{\mathbf{z}} & \mathbf{I}_3 \\ \mathbf{K}_a\mathbf{R}_2\vec{\mathbf{z}} & \mathbf{I}_3 \\ \vdots & \vdots \\ \mathbf{K}_a\mathbf{R}_N\vec{\mathbf{z}} & \mathbf{I}_3 \end{bmatrix}, \; \mathbf{\Omega}_a = \begin{bmatrix} \frac{\hat{\mathbf{\Sigma}}_a}{\Delta_a[1]} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \frac{\hat{\mathbf{\Sigma}}_a}{\Delta_a[2]} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \frac{\hat{\mathbf{\Sigma}}_a}{\Delta_a[N]} \end{bmatrix}, \; \beta_a = \begin{bmatrix} g_z \\ \mathbf{b}_a \end{bmatrix} \tag{13}$$

where $\mathbf{I}_3$ is a 3×3 identity matrix and $\mathbf{0}$ is a null matrix of the appropriate size.

It's important to highlight that the GLS problem as stated in Eq. (13) and Eq. (10) minimizes the cost function defined in Eq. (8). To see that, we may re-write the GLS problem as trying to minimize a cost J' given by

$$\begin{aligned} J' &= \|\mathbf{X}_a\beta_a - \mathbf{y}_a\|^2_{\mathbf{\Omega}_a^{-1}} \\ &= \sum_{i=1}^{N} \|\mathbf{K}_a\mathbf{R}_i\vec{\mathbf{z}}g_z + \mathbf{b}_a - \hat{\mu}_a[i]\|^2_{\left(\frac{\hat{\mathbf{\Sigma}}_a}{\Delta_a[i]}\right)^{-1}} \\ &= \sum_{i=1}^{N} \|\mathbf{K}_a\mathbf{R}_i\mathbf{g} + \mathbf{b}_a - \hat{\mu}_a[i]\|^2_{\left(\frac{\hat{\mathbf{\Sigma}}_a}{\Delta_a[i]}\right)^{-1}} \\ &= \sum_{i=1}^{N} \Delta_a[i]\left(\mathbf{K}_a\mathbf{R}_i\mathbf{g} + \mathbf{b}_a - \hat{\mu}_a[i]\right)^\top \hat{\mathbf{\Sigma}}_a^{-1}\left(\mathbf{K}_a\mathbf{R}_i\mathbf{g} + \mathbf{b}_a - \hat{\mu}_a[i]\right) \end{aligned} \tag{14}$$

where $\|\mathbf{x}\|_\mathbf{M} = \sqrt{\mathbf{x}^\top\mathbf{M}\mathbf{x}}$ is the weighted Euclidean vector norm. It's straightforward to notice that Eq. (14) is the same as Eq. (8) when we consider only $s = a$. Therefore the GLS solution for Eq. (13) minimizes the original cost function J.

For the magnetometer, we have a set of matrices given by

$$\mathbf{X}_m = \begin{bmatrix} \mathbf{K}_m\mathbf{R}_1\vec{\mathbf{x}} & \mathbf{K}_m\mathbf{R}_1\vec{\mathbf{z}} & \mathbf{I}_3 \\ \mathbf{K}_m\mathbf{R}_2\vec{\mathbf{x}} & \mathbf{K}_m\mathbf{R}_2\vec{\mathbf{z}} & \mathbf{I}_3 \\ \vdots & \vdots & \vdots \\ \mathbf{K}_m\mathbf{R}_N\vec{\mathbf{x}} & \mathbf{K}_m\mathbf{R}_N\vec{\mathbf{z}} & \mathbf{I}_3 \end{bmatrix}, \ \beta_m = \begin{bmatrix} h_x \\ h_z \\ \mathbf{b}_m \end{bmatrix} \tag{15}$$

where $\vec{\mathbf{x}}$ is the unitary vector in the x direction and the matrices $\mathbf{y}_m$ and $\mathbf{\Omega}_m$ are defined similarly to the accelerometer matrices in Eq. (13).

By solving the GLS problems given by Eq. (13) and Eq. (15) using Eq. (11), we have the best estimates for the biases and fields. However, the fields have restrictions on their values due to the definitions in Eq. (2). Through tests, we found out that simply changing the signs after solving the GLSs when they're incorrect worked better than performing a constrained optimization. This occurs because the sign change may be caused by other reasons, such as wrong rotation or gain estimate, and only occurs in the first few iterations, when the parameters are starting to fit into place.

4.3 Problem with parameter redundancy

When we are trying to estimate all parameters in Θ, we notice that two problems occur when trying to estimate the gain matrices $\mathbf{K}_a$ and $\mathbf{K}_m$.

The first one is due to the fact that there are infinite values such that $\mathbf{K}_a\mathbf{R}_i\mathbf{g}$ is constant for every $\mathbf{R}_i$, as we can scale $\mathbf{K_a}$ and $\mathbf{g}$ by inverse constants and keep the product the same. Therefore, we need to impose some restriction to allow only one combination to be feasible or the algorithm may suffer from convergence issues. We've decided to fix $g_z = -1$ so that it satisfies the restriction given by Eq. (2). To do this, we multiply the current estimation of $\mathbf{K}_a$ by $|g_z|$ and set $g_z = -1$ after solving the GLS problem given by the Eq. (13). Although we theoretically should multiply the estimate by $-g_z$, we noticed that just scaling the matrix was more robust and eased the optimization. Similarly, we set $h_x = 1$ and adjust $\mathbf{K}_m$ and h_z accordingly to keep the product constant.

The second problem occurs because, due to the polar decomposition theorem (Alonso and Shuster, 2002), we may write $\mathbf{K}_a$ as $\mathbf{K}_a = \mathbf{P}\mathbf{U}$ where $\mathbf{P}$ is a symmetric semi-positive matrix and $\mathbf{U}$ is a unitary matrix. As any rotation $\mathbf{R}_i$ is also a unitary matrix, the term $\mathbf{U}\mathbf{R}_i$ in $\mathbf{K}_a\mathbf{R}_i = \mathbf{P}\mathbf{U}\mathbf{R}_i$ doesn't allow us to distinguish between the rotation $\mathbf{R}_i$ and the matrix $\mathbf{U}$ that makes $\mathbf{K}_a$ become asymmetric. Therefore, we have again an infinite set of possible values. To solve this, we restrict $\mathbf{U}$ to be $\mathbf{U} = \text{diag}(\pm1, \pm1, \pm1)$, where $\text{diag}(\cdot)$ represents a diagonal matrix, thus making $\mathbf{K}_a$ symmetric.

We could be more restrictive in the definition of $\mathbf{U}$, but that would make the optimization harder without providing any benefit. If another format for $\mathbf{U}$ is desired, such as the identity, the polar decomposition may be performed in the final gain matrix estimate. Also, this restriction may be applied to the magnetometer instead of the accelerometer, as it doesn't matter which sensor has its gain matrix restricted as long as only one has.

4.4 Gain matrices estimation

Considering now that only the gain matrices $\mathbf{K}_a$ and $\mathbf{K}_m$ are unknown amongst the parameters in Θ, we may use GLS to find the best values again. To do so, we need to define two sets of auxiliary variables:

$$\begin{bmatrix} g^\mathcal{S}_{i,1} & g^\mathcal{S}_{i,2} & g^\mathcal{S}_{i,3} \end{bmatrix}^\top = \mathbf{R}_i\mathbf{g}$$
$$\left(\mathbf{h}^\mathcal{S}_i\right)^\top = \mathbf{R}_i\mathbf{h}$$

and define $K_{s,ij}$ to be the value in line i and column j of the matrix $\mathbf{K}_s$.

With these definitions and considering that $\mathbf{K}_a$ is symmetric, we may write the problem's matrices for the accelerometer as

$$\mathbf{y}_a = \begin{bmatrix} \hat{\mu}_a[1] - \mathbf{b}_a \\ \hat{\mu}_a[2] - \mathbf{b}_a \\ \vdots \\ \hat{\mu}_a[N] - \mathbf{b}_a \end{bmatrix}, \mathbf{X}_a = \begin{bmatrix} g^{\mathcal{S}}_{1,1} & g^{\mathcal{S}}_{1,2} & g^{\mathcal{S}}_{1,3} & 0 & 0 & 0 \\ 0 & g^{\mathcal{S}}_{1,1} & 0 & g^{\mathcal{S}}_{1,2} & g^{\mathcal{S}}_{1,3} & 0 \\ 0 & 0 & g^{\mathcal{S}}_{1,1} & 0 & g^{\mathcal{S}}_{1,2} & g^{\mathcal{S}}_{1,3} \\ g^{\mathcal{S}}_{2,1} & g^{\mathcal{S}}_{2,2} & g^{\mathcal{S}}_{2,3} & 0 & 0 & 0 \\ 0 & g^{\mathcal{S}}_{2,1} & 0 & g^{\mathcal{S}}_{2,2} & g^{\mathcal{S}}_{2,3} & 0 \\ 0 & 0 & g^{\mathcal{S}}_{2,1} & 0 & g^{\mathcal{S}}_{2,2} & g^{\mathcal{S}}_{2,3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ g^{\mathcal{S}}_{N,1} & g^{\mathcal{S}}_{N,2} & g^{\mathcal{S}}_{N,3} & 0 & 0 & 0 \\ 0 & g^{\mathcal{S}}_{N,1} & 0 & g^{\mathcal{S}}_{N,2} & g^{\mathcal{S}}_{N,3} & 0 \\ 0 & 0 & g^{\mathcal{S}}_{N,1} & 0 & g^{\mathcal{S}}_{N,2} & g^{\mathcal{S}}_{N,3} \end{bmatrix}, \beta_a = \begin{bmatrix} K_{a,11} \\ K_{a,12} \\ K_{a,13} \\ K_{a,22} \\ K_{a,23} \\ K_{a,33} \end{bmatrix} \tag{16}$$

and $\mathbf{\Omega}_a$ is the same written in Eq. (13). For the magnetometer, we have

$$\mathbf{X}_m = \begin{bmatrix} h^{\mathcal{S}}_1 & 0 & 0 \\ 0 & h^{\mathcal{S}}_1 & 0 \\ 0 & 0 & h^{\mathcal{S}}_1 \\ h^{\mathcal{S}}_2 & 0 & 0 \\ 0 & h^{\mathcal{S}}_2 & 0 \\ 0 & 0 & h^{\mathcal{S}}_2 \\ \vdots & \vdots & \vdots \\ h^{\mathcal{S}}_N & 0 & 0 \\ 0 & h^{\mathcal{S}}_N & 0 \\ 0 & 0 & h^{\mathcal{S}}_N \end{bmatrix}, \beta_m = \begin{bmatrix} K_{m,11} \\ K_{m,12} \\ K_{m,13} \\ K_{m,21} \\ K_{m,22} \\ K_{m,23} \\ K_{m,31} \\ K_{m,32} \\ K_{m,33} \end{bmatrix} \tag{17}$$

and $\mathbf{y}_m$ and $\mathbf{\Omega}_m$ defined similarly to Eq. (16) and Eq. (13), respectively.

Equations (16) and (17) are solved using Eq. (11) to obtain the gain matrices' best estimations.

5. CALIBRATION ALGORITHM

Using the estimators developed in the last section, we can now describe a greedy algorithm that always minimizes the cost function given by Eq. (8). The algorithm is:

1. Pre-process the data using Eq. (5) and Eq. (7).
2. Create the first parameters' estimates, set $k = 1$ and compute the cost $J[0]$ using Eq. (8). The first estimates are given by null biases, gain and rotation matrices as the identity, $h_x = 1$, $h_z = 1$ and $g_z = -1$.
3. Optimize the rotations as described in Sec. 4.1.
4. Optimize the bias and fields as described in Seq. 4.2 and 4.3.
5. If there was little change in the fields, optimize the gains as described in Sec. 4.4. We added this condition because we noticed that optimizing the gain matrices every iteration didn't provide enough benefit for its cost. Therefore, we only perform this optimization when the change in h_z is less than 10^{-3} between two steps, which means that the other estimates are stabilized.
6. Compute the $J[k]$ using Eq. (8).
7. If $J[k-1] - J[k] > \gamma$, go to step 3. Otherwise, end calibration.

As can be seen, every step in this algorithm is greedy, computing only the best solution at each iteration. As we'll see, this may lead to convergence problems when a step isn't convex, which in our case is the attitude estimation.

6. EXPERIMENTS

6.1 Experimental setup

The experiment was run in a AMD Phenom II x4 955 with 6GB of RAM memory available. The algorithm was implemented using SciPy (Jones *et al.*, 2001–) and NumPy. However, due to implementation details, we were unable

to measure the processor time for each simulation, which would be a more precise time measurement. To avoid huge differences in the wall time due to background programs, only 3 of the CPU's cores where used and the program received high priority.

6.2 Synthetic data

During our simulations, we considered a total of $N = 15$ sets of measurements for the accelerometer and magnetometer, both having the same number $\Delta[i] \sim \mathcal{U}(\{400, 401, \ldots, 600\})$ of collected samples at each rotation. The rotations where randomly created using the algorithm described in (Kuffner, 2004), and the algorithm's stop condition was set to $\gamma = 10^{-4}$.

When we vary the parameters N and γ to test the algorithm's performance, we need to do so in a way that allows us to correctly compare the results. For γ, we used the same set of samples for every value tested. For N, we made it so that $N = n + 1$ contains all the samples of $N = n$ plus a new random set of samples. We thus link the values generated in a way that simulates how the data would be captured, that is, if someone wants to evaluate the algorithm, he will capture the maximum number of sets and use subsets of the samples for smaller values of N. The calibrations for each value of N are performed separately, simulating a user that collected data only once, to avoid convergence issues to propagate.

The fields where created from random numbers in the intervals $[-1.5, -0.5]$, $[0.5, 1.5]$ and $[-1.5, 1.5]$ for g_z, h_x and h_z, respectively. It is important to note that these parameters won't be estimated, as explained in Sec. 4.3, but the ratio between h_x and h_z will be estimated.

The accelerometer and the magnetometer have their parameters created in the same way. Each value $b_{s,i}$ of the bias $\mathbf{b}_s$ has a random value between $[-1, 1]$, the gain matrix is given by $\mathbf{K}_s = \mathbf{I}_3 + \delta\mathbf{K}_s$ where $\delta K_{s,ij} \sim \mathcal{U}([-0.1, 0.1])$, and the covariances are given by

$$\mathbf{\Sigma}_s = \alpha_s \begin{cases} \Sigma_{s,ij} \sim \mathcal{U}([0.5, 2]), & \text{if } i = j \\ \Sigma_{s,ij} \sim \mathcal{U}([-0.2, 0.2]), & \text{otherwise} \end{cases} \tag{18}$$

where $\alpha_s \sim \mathcal{U}([10^{-2}, 10^{-4}])$.

We performed a Monte Carlo simulation with a total of 100 runs and new parameters for each run. Table 1 presents a summary of the parameters used in the experiment.

Table 1: Experiment parameters

Parameter	Description	Value
N	Number of measurements sets	15
$\Delta[i]$	Number of samples for each set i	$\mathcal{U}(\{400, 401, \ldots, 600\})$
$\mathbf{R}_i$	Rotation associated with set i	See (Kuffner, 2004)
γ	Stop condition	10^{-4}
g_z	Gravity's z component	$\mathcal{U}([-1.5, -0.5])$
h_x	Magnetic field's x component	$\mathcal{U}([0.5, 1.5])$
h_z	Magnetic field's z component	$\mathcal{U}([-1.5, 1.5])$
$b_{s,i}$	Bias' component for sensor s	$\mathcal{U}([-1, 1])$
$\mathbf{K}_s$	Gain matrix for sensor s	$I + \delta K_s,\ \delta K_{s,ij} \sim \mathcal{U}([-0.1, 0.1])$
$\mathbf{\Sigma}_s$	Covariance matrix for sensor s	See Eq. (18)
	Total of Monte Carlo runs	100

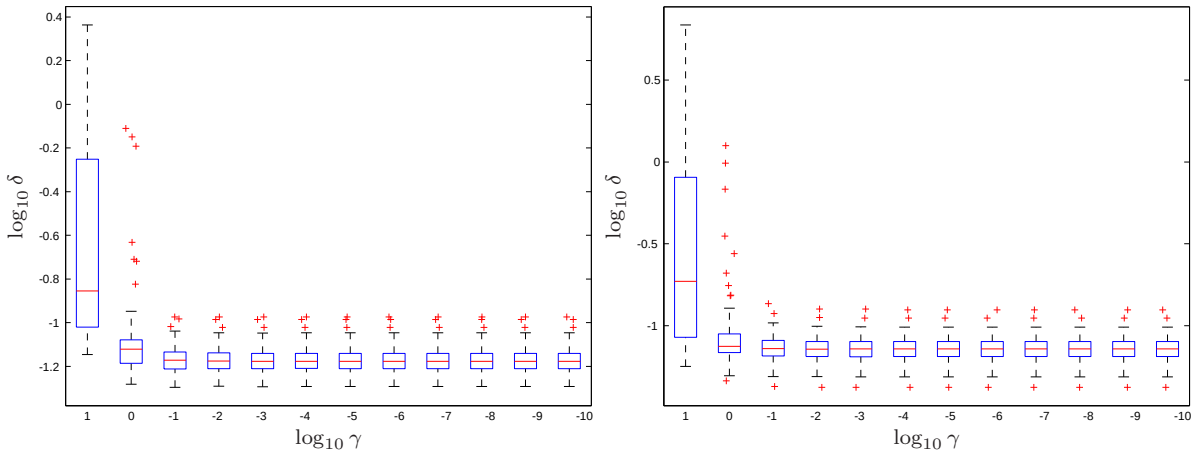
6.3 Experimental results

After the calibration, we'll use the estimated parameters and rotations to rebuild estimates $\tilde{\mu}_s[i]$ for $\mu_s[i]$. To compare the values' reconstructions quality, we'll use the following measurement:

$$\delta_s = \sqrt{\frac{\sum_{i=1}^{N} \Delta[i] \left(\mu_s[i] - \tilde{\mu}_s[i]\right)^\top \mathbf{\Sigma}_s^{-1} \left(\mu_s[i] - \tilde{\mu}_s[i]\right)}{\sum_{i=1}^{N} \Delta[i]}} \tag{19}$$

which is derived from the Mahalanobis distance (Mahalanobis, 1936).

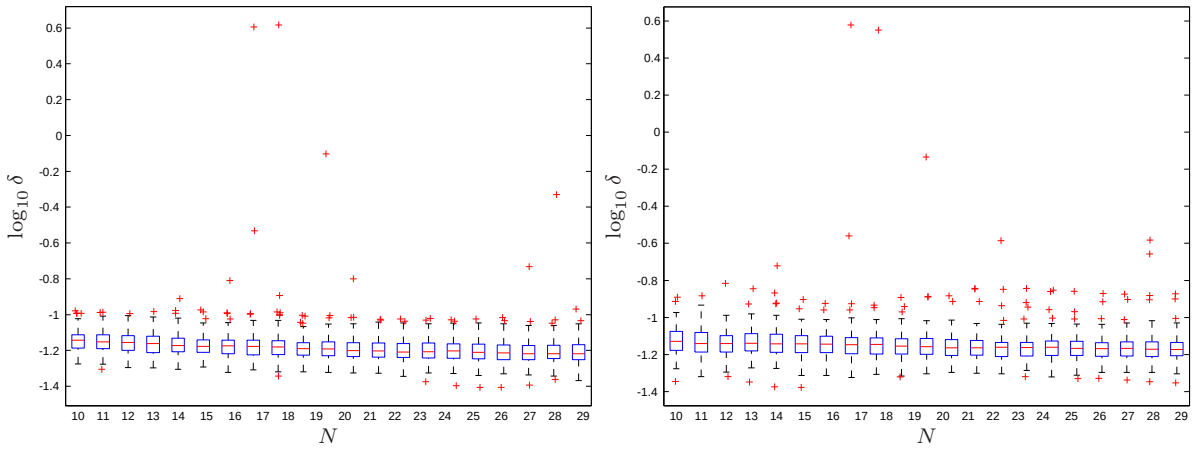
As can be seen from Fig. 1, there's little difference in performance for $\gamma \leq 10^{-2}$ and both the accelerometer and magnetometer reach similar error levels. The magnetometer errors are in general a little higher than the accelerometer's, but that seems to occur only because the magnetic field has more values to be estimated, leaving more room for error. It's worth noting that the change from $\gamma = 10$ to $\gamma = 1$ reduces significantly the reconstruction error due to more gain



(a) Accelerometer (b) Magnetometer

Figure 1: Reconstruction error varying the stop condition.

matrices estimations. With $\gamma = 10$, many runs stopped before the change in h_z was small enough to trigger a gain matrix estimation, as described in the algorithm's step 5. While performing making this step do improve performance, after a few estimations it doesn't change enough to lower the error levels significantly.



(a) Accelerometer (b) Magnetometer

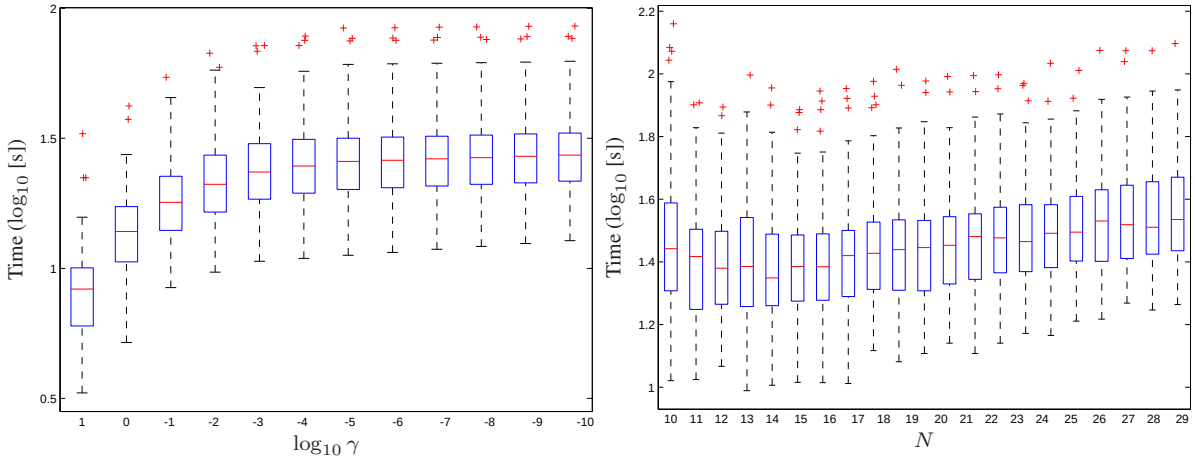
Figure 2: Reconstruction error varying the number of samples sets.

From Fig. 2, we note again that the error levels for the accelerometer and magnetometer are close, and that the number of samples doesn't affect the error levels significantly in most cases. However, sometimes the rotation estimation provides poor results, reaching low quality local minima. As the gradient descent is incapable of avoiding these minima and the algorithm is greedy, i.e., it only uses the best current estimates, all other parameters may be incorrectly estimated due to that step's non-convexity. It's curious to note that this problem presented itself when more samples where collected despite not occurring with fewer samples. We usually consider that more samples allow better estimations, but that isn't the case even when the data collected perfectly fits the model, as it's artificially created. Our suggestion is to perform the calibration with different subsets of the collected data and discard the outliers, which probably are caused by poor minima.

The computing time for the calibration process is shown in Fig. 3. We note in Fig. 3a that the time is almost constant for $\gamma \leq 10^{-5}$, which indicates that the cost function is close to a minimum and only little changes in the parameters are performed. However, as previously noted, the reconstruction error doesn't change significantly for $\gamma \leq 10^{-2}$, leading us to believe that only fine tunning is performed after that point and that they may be unnecessary. A more surprising result is observed when we vary the number of intervals.

Fig. 3b shows that the computing time is almost constant for all values of N. There are 2 main reasons why the time should be higher. First, each optimization step is more expensive as additional rotations need to be estimated and the GLS's matrices become bigger with higher number of samples. Second, the cost is a direct sum of terms that increase linearly with N, so the difference between the cost of two sequential iterations $J[k-1]$ and $J[k]$ also increases linearly with N, which should mean that more iterations are needed to reach a difference of γ.

Despite these two reasonable observations, the computing time is about the same and even has a minimum around



(a) Varying stop condition

(b) Varying number of samples sets

Figure 3: Calibration time.

$N = 14$. Although it's true that each step has higher computational cost and the cost function is higher for higher values of N if we fix the parameters, the additional data allows for faster convergence of the parameters' estimates, thus requiring significantly fewer steps to reach the stop condition.

7. CONCLUSION

In this paper, we presented a calibration process that uses maximum likelihood estimation to estimate gain and covariance matrices and bias for a magnetometer and an accelerometer, while also estimating the rotation of each set of measurements and the magnetic field direction. To the best of the authors' knowledge, this is the only algorithm that uses complete statistical model and estimates the magnetic field direction.

The proposed greedy algorithm has only one non-convex step, while the convex steps make use of the generalized least squares problem, allowing them to use the full covariance matrix. The calibration is performed in an arbitrary sensor frame, like all work in this area we know, thus needing to be integrated with an algorithm to calibrate this frame.

Using artificial data, we showed that the reconstruction error median is lower than 0.1 for all cases with reasonable stop conditions, with the error being measured by a generalization of the Mahalanobis distance given by Eq. (19). We also showed that the algorithm has almost constant computing time for a reasonable number of samples sets. Increasing number of samples doesn't improve the estimates significantly, leading to a trade-off between time spent collecting samples and calibration quality.

The rotation estimation step has the higher cost amongst all steps and is also the cause of poor parameter estimates in some cases. To solve the first issue, we envision using an approximation for the cost function described in Eq. (9) which allows faster solution computation, be it through a closed solution or better optimization behaviour. The second issue may be solved using a non-greedy algorithm, which may have higher cost but escapes poor local minima, or by an approximation that leads the rotation to a region near a good minima, allowing direct optimization to work well. We are currently working on a single approximation to solve both these problems.

Another intended future work is to analyze the prediction error on samples not used on training. Although it's not a common practice in this field, checking an algorithm's performance on a test set is common among machine learning researchers due to the possibility of overfitting. Therefore, we would like to know how well the estimates are able to fit unseen data. We haven't done this yet because we need to fit only the rotation for these new samples, but the non-convexity of our algorithm leads to poor local minima.

8. ACKNOWLEDGEMENTS

The authors would like to thank FAPESP for its support through the process 2012/01511-6.

9. REFERENCES

Alonso, R. and Shuster, M.D., 2002. "Complete linear attitude-independent magnetometer calibration". *Journal of the Astronautical Sciences*, Vol. 50, No. 4, pp. 477–490.

Chou, J.C.K., 1992. "Quaternion kinematic and dynamic differential equations". *Robotics and Automation, IEEE Transactions on*, Vol. 8, No. 1, pp. 53–64. ISSN 1042296X. doi:10.1109/70.127239.

Crassidis, J.L., Lai, K. and Harman, R.R., 2005. "Real-time attitude-independent three-axis magnetometer calibration".

Journal of Guidance Control and Dynamics, Vol. 28, No. 1, pp. 115–120. ISSN 0731-5090. doi:10.2514/1.6278.

Gebre-Egziabher, D., Elkaim, G.H., Powell, J.D. and Parkinson, B.W., 2000. "A gyro-free quaternion-based attitude determination system suitable for implementation using low cost sensors". In *Position Location and Navigation Symposium, IEEE 2000*. Ieee, pp. 185–192. ISBN 0-7803-5872-4. doi:10.1109/PLANS.2000.838301.

Jones, E., Oliphant, T., Peterson, P. *et al.*, 2001–. "SciPy: Open source scientific tools for Python". URL "`http://www.scipy.org/`".

Kariya, T. and Kurata, H., 2004. *Generalized least squares*. Wiley.

Kenney, J. and Keeping, E., 1957. *Mathematics of statistics*. Number v. 2 in Mathematics of Statistics. Van Nostrand.

Krishnamoorthy, K., 2010. *Handbook of statistical distributions with applications*, Vol. 188. Chapman and Hall/CRC.

Kuffner, J.J., 2004. "Effective sampling and distance metrics for 3D rigid body path planning". In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*. April, pp. 3993–3998, vol. 4. ISBN 0780382323.

Lizarraga, M.I., Elkaim, G.H. and Curry, R., 2013. "SLUGS UAV: A Flexible and Versatile Hardware/Software Platform for Guidance Navigation and Control Research". *Airborne Experimental Test Platforms: From Theory to Flight, Washington DC*.

Madgwick, S.O.H., Harrison, A.J.L. and Vaidyanathan, R., 2011. "Estimation of IMU and MARG orientation using a gradient descent algorithm". In *Rehabilitation Robotics (ICORR), 2011 IEEE International Conference on*. Vol. 2011, pp. 1–7. ISBN 9781424498628. ISSN 1945-7901. doi:10.1109/ICORR.2011.5975346.

Mahalanobis, P.C., 1936. "On the generalized distance in statistics". In *Proceedings of the National Institute of Sciences of India*. New Delhi, Vol. 2, pp. 49–55.

Pylvanainen, T., 2008. "Automatic and adaptive calibration of 3D field sensors". *Applied Mathematical Modelling*, Vol. 32, No. 4, pp. 575–587. ISSN 0307904X. doi:10.1016/j.apm.2007.02.004.

Sabatini, A.M., 2006. "Quaternion-based extended Kalman filter for determining orientation by inertial and magnetic sensing". *Biomedical Engineering, IEEE Transactions on*, Vol. 53, No. 7, pp. 1346–1356. ISSN 0018-9294. doi:10.1109/TBME.2006.875664.

Vasconcelos, J.F., Elkaim, G.H., Silvestre, C., Oliveira, P. and Cardeira, B., 2008. "A geometric approach to strapdown magnetometer calibration in sensor frame". In *Navigation, Guidance and Control of Underwater Vehicles*. pp. 172–177, vol. 2.

10. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.