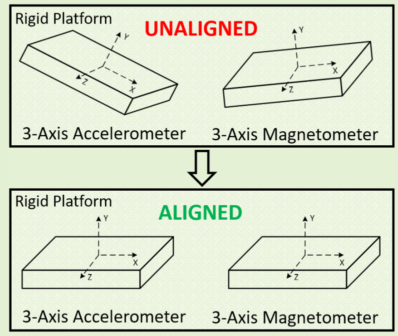
IEEE Sensors Council IEEE SENSORS JOURNAL, VOL. 20, NO. 24, DECEMBER 15, 2020 15061

A Fast and Accurate Accelerometer and Magnetometer Alignment Algorithm

Konstantinos Papafotis, *Student Member, IEEE*, and Paul P. Sotiriadis, *Senior Member, IEEE*

***Abstract*—A computationally efficient algorithm for axes alignment of a 3-axis accelerometer and a 3-axis magnetometer is introduced. The proposed algorithm applies when the two sensors are fixed on the same rigid platform and individually calibrated. It exploits the magnetic inclination phenomenon to derive both the inclination angle, and, the axes alignment rotation matrix. It is more than 50 times faster than the existing algorithms based on the same principle and using the Newton-Raphson or the gradient descent methods. The algorithm's accuracy and computational efficiency are demonstrated using multiple measurement sets as well as simulated data sets.**



***Index Terms*—Accelerometer, axes alignment, inertial sensors, magnetic inclination, magnetometer.**

I. Introduction

Accelerometers and magnetometers are typically used in combination in a broad variety of applications. Popular application fields include, and are not limited to, navigation [1], attitude estimation [2] and image stabilization [3]. Their wide use was enabled by the advancement of micro-electro-mechanical (MEMS) technology over the past years. Small-size, low-cost MEMS accelerometers and magnetometers are nowadays embedded in many commercial devices (smartphones, activity trackers, etc) further broadening their application span.

When the two sensors are used together, the alignment of their sensitivity axes is of major importance. Thus, except from the individual calibration of each sensor, an extra axes alignment procedure is mandatory. When the two sensors are in different packages, the axes misalignment is more evident and it is mainly caused by the placement of the sensors' packages on the common rigid platform. Even when the two sensors are built into the same package, the soft-iron distortion, often caused by the surrounding electronic components and the enclosure, significantly distorts the measured magnetic field vector causing a misalignment between the two sensors.

Manuscript received May 28, 2020; revised July 2, 2020; accepted July 11, 2020. Date of publication July 23, 2020; date of current version November 18, 2020. This work was supported by the Greece and the European Union [European Social Fund (ESF)] through the Operational Program Human Resources Development, Education and Lifelong Learning in the context of the Project Strengthening Human Resources Research Potential via Doctorate Research, implemented by the State Scholarships Foundation (IKY) under Grant MIS-5000432. The associate editor coordinating the review of this article and approving it for publication was Dr. Jürgen Kosel. *(Corresponding author: Konstantinos Papafotis.)*

The authors are with the Department of Electrical and Computer Engineering, National Technical University of Athens, 15780 Athens, Greece (e-mail: k.papafotis@gmail.com).

Digital Object Identifier 10.1109/JSEN.2020.3011579

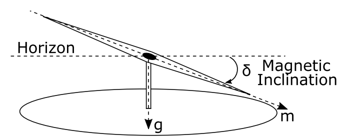


Fig. 1. Magnetic inclination [4].

Existing works deal with the axes alignment either as an independent problem [4]–[6] or as part of a calibration procedure [7]–[11]. The authors in [7], [8], [10], [12] propose a magnetometer calibration algorithm incorporating the axes alignment step. In [4], [5], [13] the axes alignment is part of an inertial (accelerometer and gyroscope) and magnetic sensors' calibration algorithm. An accelerometer - magnetometer calibration algorithm, also accounting for the axes misalignment is proposed in [6], [9], [11]. Note that in [7]–[11], [13] the calibration and axes alignment are fused together while in [4]–[6] the alignment step is a discrete part of the calibration algorithm.

While their algorithmic implementation and computational efficiency vary significantly, all the aforementioned algorithms are based on two fundamental approaches; they either use a calibrated inertial sensor (accelerometer or gyroscope) as reference (e.g. [5], [7], [10], [12]), or, exploit the magnetic inclination to align the axes of the two sensors (e.g. [4], [8]).

Magnetic inclination (or magnetic dip) is the angle between the horizon and the Earth's magnetic field lines as shown in Figure 1. It varies with location and time and the $\sin(\cdot)$ of it equals the normalized inner product of the gravity vector g (in m/s^2) and the magnetic field vector m (in $Tesla$).[1]

$$sin(\delta) = \frac{g^T m}{\|g\|\|m\|} \tag{1}$$

[1] All norms in this paper are Euclidean norms unless it is indicated otherwise.

When a gyroscope is used as a reference, the calibration and the axes alignment are usually formulated as an estimation problem which is typically solved using Kalman filter [7]. When the magnetic inclination is exploited, the axes alignment is formulated as an optimization problem which is usually solved using the gradient descent [4] or the Newton-Raphson [6] methods implying increased computational burden and potential convergence issues.

Accelerometers and magnetometers are widely used in embedded low-cost devices with very limited hardware resources. In such cases, the computational efficiency of the calibration and axes alignment algorithms is of major importance as they have to be executed quickly on a microcontroller or on a basic microprocessor. To that purpose, we propose a two-phases computationally efficient, iterative algorithm for aligning the axes of a 3-axis accelerometer and a 3-axis magnetometer. It is more than 50 times faster than similar algorithms in the literature and requires no special piece of measurement equipment.

Using a set of measurements of the two sensors, and exploiting the magnetic inclination, the proposed algorithm calculates both the axes alignment rotation matrix and the magnetic inclination angle by solving a computationally-light least-squares problem and then using a single iteration of the Newton-Raphson method. The proposed algorithm applies when the two sensors are individually calibrated and fixed on the same rigid platform.

The algorithm is tested using both experimental measurements and simulated data. Finally, its convergence and efficiency are compared to existing algorithms using the standard iterative methods i.e. gradient descent and Newton-Raphson.

The rest of the paper is organized as follows. In Section II, the axes alignment is formulated as an optimization problem. The proposed algorithm is presented in Section III. The evaluation of the algorithm using both experimental measurements and simulated data is done in Section IV. Finally, the conclusions are drawn in Section V.

II. Problem Formulation

Consider a 3-axis accelerometer and a 3-axis magnetometer, both fixed on the same rigid platform, and denote their coordinate frames as $\{A\}$ and $\{M\}$ respectively.[2] Note that in the following analysis, the two sensors are assumed to be individually calibrated.

Now consider a set of K measurements of each sensor, captured while the sensors' platform is still in K different orientations respectively and away from magnetic disturbances. The measurements of the accelerometers are naturally expressed in the $\{A\}$-frame while those of the magnetometer are expressed in the $\{M\}$-frame. They are denoted as g_k^A and m_k^M respectively, for $k = 1, 2, \ldots, K$.

Let the gravity vector g and magnetic field vector m be expressed in a fixed inertial coordinate frame $\{I\}$. We express both sensors' measurements in the $\{A\}$-frame as

$$g_k^A = Q_k g \quad \text{and} \quad m_k^A = Q_k m, \quad k = 1, 2, \ldots, K \tag{2}$$

[2]Both coordinate frames are assumed to be right-handed.

where $Q_k \in SO(3)$, $k = 1, 2, \ldots, K$ is the frame transformation matrix from the $\{I\}$-frame to the $\{A\}$-frame corresponding to the k^{th} orientation. Combining (1) and (2), we can calculate the inclination angle using the sensors' measurements

$$\sin(\delta) = \frac{(g_k^A)^T Q_k Q_k^T m_k^A}{\|Q_k^T g_k^A\| \|Q_k^T m_k^A\|} = \frac{(g_k^A)^T m_k^A}{\|g_k^A\| \|m_k^A\|}, \quad k = 1, 2, \ldots, K \tag{3}$$

where we exploited the orthogonality of Q_k and the rotational invariance of the Euclidean norm.

Let $R_M^A \in SO(3)$ be the frame transformation matrix from the $\{M\}$-frame to the $\{A\}$-frame and m_k^M be the magnetometer's measurement in the $\{M\}$-frame. Then, the magnetometer's measurement, expressed in the $\{A\}$-frame, is written as

$$m_k^A = R_M^A m_k^M, \quad k = 1, 2, \ldots, K \tag{4}$$

Substituting (4) in (3) and using again the rotational invariance of the Euclidean norm, we get

$$\sin(\delta) = \frac{(g_k^A)^T R_M^A m_k^M}{\|g_k^A\| \|m_k^M\|}, \quad k = 1, 2, \ldots, K \tag{5}$$

In (5), g_k^A and m_k^M are the known accelerometer's and magnetometer's measurements, respectively, while both R_M^A and inclination angle, δ, are unknown. A standard approach to derive them is to form and solve the following optimization problem [4], [6]

$$\begin{aligned} \underset{R_M^A, \delta}{\text{minimize}} \quad & \sum_{k=1}^{K} \left(\sin(\delta) - \frac{(g_k^A)^T R_M^A m_k^M}{\|g_k^A\| \|m_k^M\|} \right)^2 \\ \text{subject to} \quad & R_M^A \in SO(3) \\ & \delta \in \left[-\frac{\pi}{2}, \frac{\pi}{2} \right] \end{aligned} \tag{6}$$

Solving (6) is typically done using either the gradient descent or the Newton-Raphson methods. Both are computational costly; gradient descent typically requires hundreds of iterations to converge while Newton-Raphson, although it converges faster, it requires the calculation of the Hessian matrix in every step.

III. The Proposed Algorithm

The proposed algorithm is executed in two phases. In the first phase, an estimate of the axes alignment rotation matrix R_M^A and the sine of the inclination angle, $\sin(\delta)$, is derived in a closed-form. Then, in the second phase, the derived R_M^A and $\sin(\delta)$ are used as a starting point for the Newton-Raphson method which rapidly (in 1 iteration) converges to more accurate estimates.

For the following analysis it is assumed that both the accelerometer and the magnetometer have already been individually calibrated. For notational convenience, in the rest of this paper we drop the superscripts from the field vectors and both the subscript and superscript from the axes alignment matrix i.e. $g_k \triangleq g_k^A$, $m_k \triangleq m_k^M$ and $R \triangleq R_M^A$.

In addition, without loss of generality, we assume that the gravity and the magnetic field measurement vectors are normalized i.e. $\|g_k\| = \|m_k\| = 1$, $k = 1, 2, \ldots, K$. Finally we define $s \triangleq \sin(\delta)$.

A. Derivation of R and s Estimates in Closed-Form

Given the aforementioned assumptions, (5) is written as

$$s = g_k^T R m_k, \;\; k = 1, 2, \ldots, K \tag{7}$$

Using the identity $\text{vec}(AXB) = (B^T \otimes A)\text{vec}(X)$, where $\otimes$ denotes the Kronecker's product [14], from (7) we get

$$s = (m_k^T \otimes g_k^T)\text{vec}(R), \;\; k = 1, 2, \ldots, K \tag{8}$$

Equation (8) is written in the more convenient matrix form

$$s\,\mathbf{1} = A\ \text{vec}(R) \tag{9}$$

where $\mathbf{1}$ is the $K \times 1$ vector of ones and

$$A \triangleq \begin{bmatrix} (m_1 \otimes g_1)^T \\ (m_2 \otimes g_2)^T \\ \vdots \\ (m_K \otimes g_K)^T \end{bmatrix}. \tag{10}$$

In (9), the $K \times 9$ matrix A, which is formed using the measurements of the two sensors is known, whereas both the sine of the inclination angle, s, and the alignment matrix in vector form, $\text{vec}(R)$, are unknown. To proceed, it is assumed that the sine of the inclination angle, s, is nonzero, and furthermore, that matrix A is of full rank.

We solve (9) for $\text{vec}(R)$ in a least square sense

$$\text{vec}(R) = s\left(A^T A\right)^{-1} A^T \mathbf{1}. \tag{11}$$

Splitting $\left(A^T A\right)^{-1} A^T \mathbf{1}$ into three 3×1 vectors h_1, h_2 and h_3, such that $[h_1^T\ h_2^T\ h_3^T]^T = \left(A^T A\right)^{-1} A^T \mathbf{1}$, we form the following matrix

$$H = [h_1\ h_2\ h_3]. \tag{12}$$

Then, from (11) and (12) we have

$$R = sH \tag{13}$$

Note that in (13), H is formed using the sensors' measurements and it is known but s is unknown.

Consider a Singular Value Decomposition (SVD) of H i.e. $H = U\Sigma V^T$ where $U, V \in O(3)$ and $\Sigma = \text{diag}(\sigma_1, \sigma_2, \sigma_3)$ with $\sigma_1 \geq \sigma_2 \geq \sigma_3 > 0$.[3] Then (13) gives

$$R = sU\Sigma V^T \tag{14}$$

With ideal (noiseless) measurements it is $R \in SO(3)$ implying $\det(R) = 1$. In the more realistic case, when real noisy measurements are used, R is expected to be close to but not necessarily in $SO(3)$. However it is reasonable to assume that even with noisy measurements, it is $\det(R) > 0$ and thus (13) implies that $\text{sign}(s) = \text{sign}(\det(H))$. Using the last one and (13) we derive the following alternative SVD of R,

$$R = \hat{U}\hat{\Sigma}V^T \tag{15}$$

with

$$\hat{U} = \text{sign}\left(\det(H)\right)U \;\text{ and }\; \hat{\Sigma} = |s|\Sigma, \tag{16}$$

[3]From (13) it is $1 = \det(R) = s^3 \det(H)$ implying $\sigma_3 > 0$. Due to continuity of the functions involved, this is also true for real measurements with acceptable noise levels.

for which

$$\det(\hat{U}V^T) = 1. \tag{17}$$

Since, with real data, R in (15) is close to but not necessarily in $SO(3)$, we derive matrix $\hat{R}$ which is the closest orthogonal matrix to R in the following sense

$$\hat{R} = \underset{\hat{R} \in SO(3)}{\text{argmin}} \|R - \hat{R}\|_F. \tag{18}$$

This is a variation of the Orthogonal Procrustes Problem [15], [16], the unique solution of which is given by

$$\hat{R} = \hat{U}V^T \tag{19}$$

where $\det(\hat{R}) = 1$ is implied by (17).

An estimate of the inclination angle, can now be derived by replacing R by $\hat{R}$ in (7) and averaging, i.e.,

$$\hat{s} = \frac{1}{K}\sum_{k=1}^{K} g_k \hat{R} m_k \tag{20}$$

This concludes the first phase of the proposed algorithm.

B. Improving Accuracy With Newton-Raphson Method

In order to improve the estimation's accuracy, we use the derived $\hat{R} \in SO(3)$ and $\hat{s}$ from (19) and (20) respectively as starting point for the Newton-Raphson method. To apply the Newton-Raphson method, we write the following cost-plus-penalty function, associated with (6) which captures both the axes alignment error and the divergence of R from orthogonality as the algorithm progresses.

$$J(R, s) = \lambda_1 \|RR^T - I\|_F^2 + \lambda_2 \sum_{k=1}^{K}\left(s - g_k^T R m_k\right)^2 \tag{21}$$

Note that the weights $\lambda_1, \lambda_2 > 0$ are typically set to one.

To ensure that $R \in SO(3)$ we should also include $(\det(R) - 1)^2$ as an extra term to (21), forcing the determinant of R to be equal to one. However, since the starting point $\hat{R}$ of the algorithm is typically close to the final solution, this is not required, assuming data of sufficient quality.

The implementation of the Newton-Raphson method is presented in Appendix I where the required Hessian matrix is provided. The proposed method is summarized in Algorithm 1.

IV. Algorithm's Evaluation

In order to evaluate the convergence and the computational efficiency of the proposed algorithm, we performed a series of experimental measurements. In addition to that, multiple simulated measurements were used to demonstrate the algorithm's accuracy.

A. Measurement Acquisition Procedure

The 12-step measurement acquisition procedure introduced in [4] was used to record four datasets. More specifically we placed the sensors' platform in the 12 still orientations proposed in [4] recording $K = 12$ accelerometer's and magnetometer's measurements respectively. As mentioned in [4], no special piece of equipment was used and the required sensors' placement was done by hand. In order to increase the

Algorithm 1 The Proposed Algorithm

Phase 1

Step 1: Use the K accelerometer's, g_k, and magnetometer's, m_k, measurements to form the $K \times 9$ matrix A in (10).
Step 2: Use (11) and (12) to form the 3×3 matrix H.
Step 3: Assume an SVD of H i.e. $H = U\Sigma V^T$ where $U, V \in O(3)$ and $\Sigma = \text{diag}(\sigma_1, \sigma_2, \sigma_3)$.
Step 4: Use (16) and (19) to calculate $\hat{R}$.
Step 5: Use (20) to calculate $\hat{s}$.

Phase 2

Step 6: Minimize (21) using Newton-Raphson method with $(\hat{R}, \hat{s})$ as the starting point.

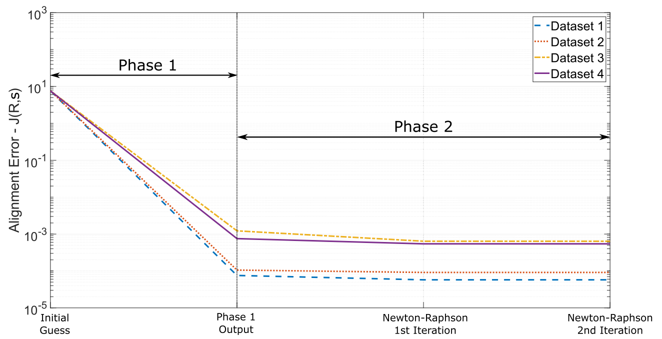


Fig. 2. Convergence of the proposed algorithm for the four recorded datasets.

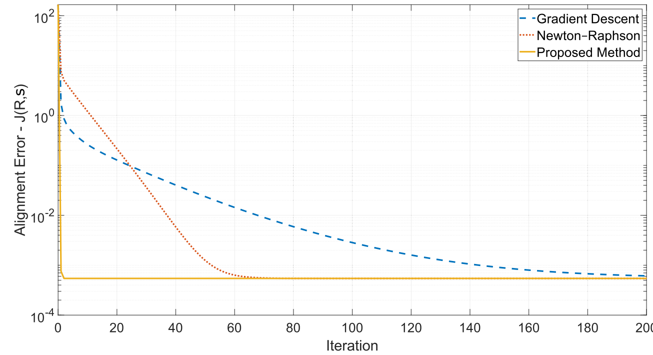
measurements' accuracy, we used multiple measurements and averaging. We repeated the measurement acquisition procedure four times and recorded four different datasets. Note that all the measurements took place on the same day, inside the campus of the National Technical University of Athens, Greece, away from magnetic disturbances.

B. Algorithm's Convergence and Computational Efficiency

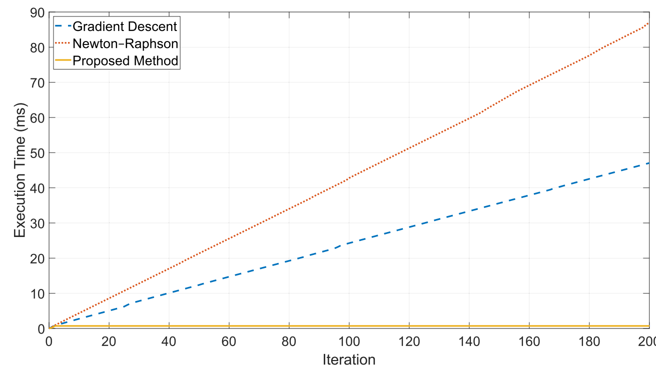
The convergence of the proposed algorithm for the four recorded datasets is presented in Figure 2. Using $\hat{R}$ and $\hat{s}$ from phase 1 as the starting point of the algorithm, the error converges after just one iteration of the Newton-Raphson method for all four datasets.

We compare the proposed algorithm's computational efficiency, to that of the standard approaches in the literature using the gradient descent or the Newton-Raphson method to minimize (6). In Figures 3a and 3b the convergence and the execution time of the three methods, are presented. The proposed algorithm converges after just 2 iterations and about 0.5 ms. Newton-Raphson requires 75 iterations and about 32 ms, and, gradient descent requires 200 iterations and 47 ms to achieve the same accuracy.

In Figure 3, the convergence of the error and the execution time of the three algorithms are presented for a single dataset and a single run. To provide a better indication about the algorithms' performance, we used all four recorded datasets and a total of 400 runs of each algorithm (100 runs per dataset).



(a) Alignment error of the proposed algorithm compared to the alignment error of existing algorithms based on the same principle and using gradient descent or Newton-Raphson to minimize (6) parametrized on their iteration step.



(b) Execution time of the proposed algorithm compared to the the execution time of existing algorithms based on the same principle and using gradient descent or Newton-Raphson to minimize (6) parametrized on their iteration step.

Fig. 3. Performance comparison between the proposed algorithm and the standard methods used in the literature.

TABLE I
MEAN EXECUTION TIME OF THE PROPOSED ALGORITHM FOR 400 RUNS COMPARED TO THE MEAN EXECUTION TIME OF NEWTON-RAPHSON AND GRADIENT DESCENT BASED APPROACHES (ALL ACHIEVING THE SAME FINAL ERROR)

Method	**M.E.T. (ms)**	**Relative Time**
P	0.61	$\times 1$
NR	33.12	$\times 54$
GD	46.20	$\times 75$

The mean execution time (M.E.T.) of each algorithm is presented in Table I. As seen in Table I, the proposed algorithm (P) is 54 times faster than the Newton-Raphson (NR) and 75 times faster than the gradient descent (GD).

C. Algorithm's Accuracy

Since the recorded datasets are real measurements, they incorporate the non-idealities of a real system, e.g., noise, quantization and residual calibration errors of the accelerometer and the magnetometer. Therefore, they are appropriate to demonstrate the convergence and the computational efficiency of the proposed algorithm in real world applications.

On the other hand, since the misalignment of the two sensors is not known, the algorithm's accuracy cannot be properly

TABLE II

MEAN VALUE AND VARIANCE OF THE EULER ANGLE REPRESENTATION OF THE RESIDUAL ALIGNMENT ERROR E FOR FIVE DIFFERENT TEST-CASES USING THE PROPOSED ALGORITHM

R_A^M $[\psi \quad \theta \quad \phi]$	Sensors' Noise σ_g^2, σ_m^2	μ_E $[\mu_{E\psi} \quad \mu_{E\theta} \quad \mu_{E\phi}]$	σ_E^2 $[\sigma_{E\psi}^2 \quad \sigma_{E\theta}^2 \quad \sigma_{E\phi}^2]$
$[1^\circ \quad 1^\circ \quad 1^\circ]$	$5.2 \cdot 10^{-4}$, $5.5 \cdot 10^{-6}$	$[-0.001^\circ \quad 0.002^\circ \quad -0.001^\circ]$	$[0.002^\circ \quad 0.012^\circ \quad 0.024^\circ]$
$[3^\circ \quad 4^\circ \quad 5^\circ]$	$5.2 \cdot 10^{-4}$, $5.5 \cdot 10^{-6}$	$[-0.001^\circ \quad 0.001^\circ \quad 0.002^\circ]$	$[0.024^\circ \quad 0.012^\circ \quad 0.016^\circ]$
$[15^\circ \quad 20^\circ \quad 25^\circ]$	$5.2 \cdot 10^{-4}$, $5.5 \cdot 10^{-6}$	$[0.004^\circ \quad -0.004^\circ \quad -0.004^\circ]$	$[0.018^\circ \quad 0.009^\circ \quad 0.014^\circ]$
$[45^\circ \quad 45^\circ \quad 45^\circ]$	$5.2 \cdot 10^{-4}$, $5.5 \cdot 10^{-6}$	$[0.002^\circ \quad -0.003^\circ \quad -0.003^\circ]$	$[0.008^\circ \quad 0.015^\circ \quad 0.020^\circ]$
$[90^\circ \quad 0^\circ \quad 0^\circ]$	$5.2 \cdot 10^{-4}$, $5.5 \cdot 10^{-6}$	$[0.001^\circ \quad -0.002^\circ \quad 0.001^\circ]$	$[0.012^\circ \quad 0.018^\circ \quad 0.007^\circ]$

In all five cases, noisy simulated measurements were used. The algorithm was run 1000 times for every test-case using different (random) initial gravity, g_1^A, and magnetic field, m_1^A, vectors, as well as, different sequences of additive noise. The variance of the additive accelerometer's, σ_g^2, and magnetometer's noise, σ_m^2, was the same for all test-cases.

evaluated using experimental data. To this purpose, we used simulated accelerometer's and magnetometer's measurements.

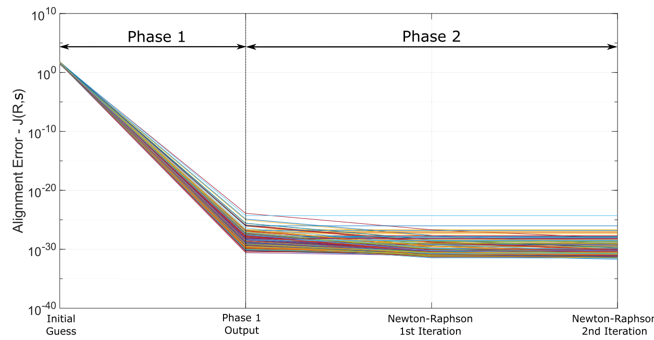
Simulated measurements were generated following the measurement acquisition procedure of [4]. More specifically, two random unit vectors representing the gravity, g_1^A, and the magnetic field, m_1^A, vectors, expressed in the $\{A\}$-frame, were first generated. Then they were both rotated 11 times according to [4] resulting in the $\{A\}$-frame measurement vectors g_k^A and m_k^A, $k = 2, 3, \ldots, 12$. Finally the magnetometers' measurements were simultaneously rotated once more using a selected rotation matrix R_A^M to get the corresponding $\{M\}$-frame measurement vectors m_k^M, $k = 1, 2, \ldots, 12$.

The accuracy of the proposed algorithm was evaluated for five different cases of selected rotation matrices R_A^M, corresponding to a misalignment of a few degrees, to the extreme case of a 90° misalignment between the sensors' axes. For every case, two different simulations were performed; one using ideal, noiseless measurements, and one using noisy measurements resulting by adding band-limited white noise to the noiseless ones. The noise characteristics were chosen to match those of the measurement platform used for the experimental measurements. Every simulation consisted of 1000 runs of the proposed algorithm. In each run we used different (random) initial gravity, g_1^A, and magnetic field, m_1^A, vectors, as well as different sequences of additive noise samples.

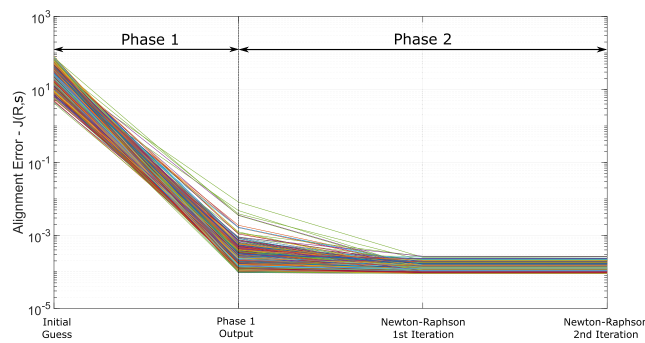
The convergence of the proposed algorithm for 1000 runs using noiseless and noisy simulated measurements is presented in Figure 4. In the case of the noiseless measurements, the algorithm converges to an extremely small value within one step (phase 1) indicating perfect alignment between the sensors' coordinate frames. When the more realistic, noisy measurements, are used, the error converges to a much larger value, indicating a residual alignment error. We define the residual alignment error matrix, E, as the product between the original rotation of the $\{M\}$-frame, R_A^M, and the derived rotation matrix R,

$$E = R_A^M R \tag{22}$$

In the ideal case, where $R = R_M^A$, the error E is equal to the identity matrix, representing zero rotation between the two frames. The mean value, μ_E, and the variance, σ_E^2, of the Euler angle representation of E for every test-case is presented in Table II.



(a) Alignment error of the proposed algorithm using 1000 different datasets of noiseless simulated measurements. In dataset, different (random) initial gravity, g_1^A, and magnetic field, m_1^A, vectors were used.



(b) Alignment error of the proposed algorithm using 1000 different datasets of noisy simulated measurements. Different (random) initial gravity, g_1^A, and magnetic field, m_1^A, vectors were used in the datasets.

Fig. 4. Convergence of the proposed algorithm using noiseless (a) and noisy (b) simulated data.

V. CONCLUSION

A computationally efficient algorithm for aligning a 3-axis accelerometer and a 3-axis magnetometer was presented. The proposed method exploits the magnetic inclination to form a least-squares based iterative algorithm for the calculation of both the axes alignment matrix and the magnetic inclination angle. Experimental results were used to demonstrated its computational efficiency; the proposed algorithm is about 55 times faster than the existing ones. Its accuracy was proved using several different datasets of simulated measurements.

Algorithm 2 Gradient Descent Based Algorithm

Step 1: Initialize R as I_3 and $s = 0$ and form $x = \left[\mathrm{vec}(R)^T \ s\right]^T$.
Step 2: Initialize t, a and β
Step 3: Calculate the gradient: Δx=-∇J(x)
Step 4: Choose step size: While $J(x + t\Delta x) > J(x) + at\nabla J(x)^T \Delta x$ t:= βt
Step 5: Update $x = x + t\Delta x$
Step 6: Calculate $J(x)$
Step 7: Repeat steps 3-6 until $J(x)$ is sufficiently small

Algorithm 3 Newton-Raphson Based Algorithm

Step 1: Initialize R as I_3 and $s = 0$ and form $x = \left[\mathrm{vec}(R)^T \ s\right]^T$.
Step 2: Initialize t, a and β
Step 3: Calculate Δx: $\Delta x = -\left(\nabla^2 J(x)\right)^{-1} \nabla$J(x)
Step 4: Choose step size: While $J(x + t\Delta x) > J(x) + at\nabla J(x)^T \Delta x$ t:= βt
Step 5: Update $x = x + t\Delta x$
Step 6: Calculate $J(x)$
Step 7: Repeat steps 3-6 until $J(x)$ is sufficiently small

Appendix I
Gradient Descent and Newton-Raphson Based Algorithms

In this section, the implementation of the gradient descent and Newton-Raphson based algorithms is described. In addition, the required gradient vector and Hessian matrix for applying the two methods are provided. To do so, for notational purposes, we write the cost-plus-penalty function of (21) as

$$J(x) = \|RR^T - I\|_F^2 + \sum_{k=1}^{K} \left(s - g_k^T R m_k\right)^2 \tag{23}$$

i.e., instead of $J(R, s)$ we write it equivalently as $J(x)$ where vector x is a different formation of the pair (R, s),

$$x = \left[\mathrm{vec}(R)^T \ s\right]^T \tag{24}$$

A. Gradient Descent Based Algorithm

To solve (6) using the gradient descent method we implemented the gradient descent method incorporating the backtracking line search according to [17] as shown in Algorithm 2. The 10×1gradient vector is derived in a closed-form as follows:

$$\nabla J(x) \triangleq \left[\left(\frac{\partial J(x)}{\partial \mathrm{vec}(R)}\right)^T \ \frac{\partial J(x)}{\partial s}\right]^T$$

where

$$\frac{\partial J(x)}{\partial \mathrm{vec(R)}} = -2\sum_{k=1}^{K}\left[\left(s - g_k^T R m_k\right)(m_k \otimes g_k)\right] + 4\mathrm{vec}\left(RR^T R - R\right)$$

$$\frac{\partial J(x)}{\partial s} = 2\sum_{k=1}^{K}\left(s - g_k^T R m_k\right)$$

B. Newton-Raphson Based Algorithm

To solve (6) using the Newton-Raphson method, we implemented theNewton-Raphson method, incorporating the backtracking line search according to [17] as shown in Algorithm 3. The required 10×10 Hessian matrix is derived in a closed-form as follows:

$$\nabla^2 J(x) \triangleq \frac{\partial^2 J(x)}{\partial x \partial x^T} = \begin{bmatrix} \frac{\partial^2 J(x)}{\partial \mathrm{vec}(R) \partial \mathrm{vec}(R)^T} & \frac{\partial^2 J(x)}{\partial \mathrm{vec}(R) \partial s} \\ \frac{\partial^2 J(x)}{\partial s \partial \mathrm{vec}(R)^T} & \frac{\partial^2 J(x)}{\partial s^2} \end{bmatrix}$$

where

$$\frac{\partial^2 J(x)}{\partial \mathrm{vec}(R)\, \partial \mathrm{vec}(R)^T} = 4\left[(R^T R) \otimes I_3 + I_3 \otimes (RR^T) + (R^T \otimes R)A\right] - 4I_9 + 2\sum_{k=1}^{K}\left[(m_k \otimes g_k)(m_k \otimes g_k)^T\right]$$

$$\frac{\partial^2 J(x)}{\partial \mathrm{vec}(R)\, \partial s} = -2\sum_{k=1}^{K}(m_k \otimes g_k)$$

$$\frac{\partial^2 J(x)}{\partial s\, \partial \mathrm{vec}(R)^T} = -2\sum_{k=1}^{K}(m_k \otimes g_k)^T$$

$$\frac{\partial^2 J(x)}{\partial s^2} = 2K$$

and

$$A = \left[e_1 \ e_4 \ e_7 \ e_2 \ e_5 \ e_8 \ e_3 \ e_6 \ e_9\right]^T$$

where e_k is the k^{th} normal column vector in $\mathbb{R}^9$.

References

[1] C. Fischer, P. Talkad Sukumar, and M. Hazas, "Tutorial: Implementing a pedestrian tracker using inertial sensors," *IEEE Pervas. Comput.*, vol. 12, no. 2, pp. 17–27, Apr. 2013.

[2] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," in *Proc. IEEE Int. Conf. Rehabil. Robot.*, Jun. 2011, pp. 1–7.

[3] C.-H. Chu, "Video stabilization for stereoscopic 3D on 3D mobile devices," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2014, pp. 1–6.

[4] K. Papafotis and P. P. Sotiriadis, "MAG.I.C.AL.–A unified methodology for magnetic and inertial sensors calibration and alignment," *IEEE Sensors J.*, vol. 19, no. 18, pp. 8241–8251, Sep. 2019.

[5] J. Metge, R. Mégret, A. Giremus, Y. Berthoumieu, and T. Décamps, "Calibration of an inertial-magnetic measurement unit without external equipment, in the presence of dynamic magnetic disturbances," *Meas. Sci. Technol.*, vol. 25, no. 12, Dec. 2014, Art. no. 125106. [Online]. Available: http://stacks.iop.org/0957-0233/25/i=12/a=125106

[6] M. V. Gheorghe, "Calibration for tilt-compensated electronic compasses with alignment between the magnetometer and accelerometer sensor reference frames," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (I2MTC)*, May 2017, pp. 1–6.

[7] Y. Wu, D. Zou, P. Liu, and W. Yu, "Dynamic magnetometer calibration and alignment to inertial sensors by Kalman filtering," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 2, pp. 716–723, Mar. 2018.

[8] M. Kok, J. D. Hol, T. B. Schön, F. Gustafsson, and H. Luinge, "Calibration of a magnetometer in combination with inertial sensors," in *Proc. 15th Int. Conf. Inf. Fusion*, Jul. 2012, pp. 787–793.

[9] X. Li and Z. Li, "A new calibration method for tri-axial field sensors in strap-down navigation systems," *Meas. Sci. Technol.*, vol. 23, no. 10, Sep. 2012, Art. no. 105105.

[10] M. Zhu, Y. Wu, and W. Yu, "An efficient method for gyroscope-aided full magnetometer calibration," *IEEE Sensors J.*, vol. 19, no. 15, pp. 6355–6361, Apr. 2019.
[11] K. Papafotis and P. P. Sotiriadis, "Accelerometer and magnetometer joint calibration and axes alignment," *Technologies*, vol. 8, no. 1, p. 11, Jan. 2020.
[12] M. Kok and T. B. Schön, "Maximum likelihood calibration of a magnetometer using inertial sensors," *IFAC Proc. Volumes*, vol. 47, no. 3, pp. 92–97, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1474667016415983
[13] J. Wu, "Real-time magnetometer disturbance estimation via online nonlinear programming," *IEEE Sensors J.*, vol. 19, no. 12, pp. 4405–4411, Jun. 2019.
[14] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2013.
[15] P. H. Schönemann, "A generalized solution of the orthogonal procrustes problem," *Psychometrika*, vol. 31, no. 1, pp. 1–10, Mar. 1966.
[16] N. J. Higham, *Functions of Matrices: Theory and Computation*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2008.
[17] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2013.

Konstantinos Papafotis (Student Member, IEEE) received the Diploma degree in electrical and computer engineering from the National Technical University of Athens, Greece, in 2015, where he is currently pursuing the Ph.D. degree, under the supervision of Prof. Paul P. Sotiriadis. He is a Teaching Assistant in several undergraduate courses and acts as an Advisor for several diploma theses. He is the author of several conference papers and journal articles. His main research interests include inertial navigation, embedded systems, and wireless sensors' systems. He has received the Best Paper Award in the IEEE International Conference on Modern Circuits and Systems Technologies, in 2019. He is a regular reviewer for many IEEE publications.

Paul P. Sotiriadis (Senior Member, IEEE) received the Diploma degree in electrical and computer engineering from the National Technical University of Athens, Greece, the M.S. degree in electrical engineering from Stanford University, CA, USA, and the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, USA, in 2002.

He is currently a Faculty Member with the Electrical and Computer Engineering Department, National Technical University of Athens, the Director of the Electronics Laboratory of the University, and a member of the Governing Board of the Hellenic Space Center and the National Space Center of Greece. In 2002, he joined Johns Hopkins University as an Assistant Professor of Electrical and Computer Engineering. In 2012, he joined the faculty of the Electrical and Computer Engineering Department, National Technical University of Athens. He has authored and coauthored more than 150 research publications, most of them in IEEE journals and conferences, holds one patent, and has contributed chapters to technical books. His research interests include design, optimization, and mathematical modeling of analog and mixed-signal circuits, RF and microwave circuits, advanced frequency synthesis, biomedical instrumentation, and interconnect networks in deep-submicrometer technologies. He has led several projects in these fields funded by the U.S. organizations and has collaborations with industry and national labs.

Dr. Sotiriadis has received several awards, including the 2012 Guillemin-Cauer Award from the IEEE Circuits and Systems Society, the Best Paper Award in the IEEE International Symposium on Circuits and Systems 2007, the Best Paper Award in the IEEE International Frequency Control Symposium 2012, and the Best Paper Award in the IEEE International Conference on Modern Circuits and Systems Technologies 2019. He is an Associate Editor of the IEEE SENSORS JOURNAL. He has served as an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS from 2016 to 2020 and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS from 2005 to 2010. He has been a member of technical committees of many conferences. He regularly reviews for many IEEE transactions and conferences and serves on proposal review panels.