

Article

Optimization of Gradient Descent Parameters in Attitude Estimation Algorithms

Karla Sever, Leonardo Max Golušin and Josip Lončar * 

Department of Communication and Space Technologies, Faculty of Electrical Engineering and Computing, University of Zagreb, Unska 3, 10000 Zagreb, Croatia

* Correspondence: josip.loncar@fer.hr

Abstract: Attitude estimation methods provide modern consumer, industrial, and space systems with an estimate of a body orientation based on noisy sensor measurements. The gradient descent algorithm is one of the most recent methods for optimal attitude estimation, whose iterative nature demands adequate adjustment of the algorithm parameters, which is often overlooked in the literature. Here, we present the effects of the step size, the maximum number of iterations, and the initial quaternion, as well as different propagation methods on the quality of the estimation in noiseless and noisy conditions. A novel figure of merit and termination criterion that defines the algorithm's accuracy is proposed. Furthermore, the guidelines for selecting the optimal set of parameters in order to achieve the highest accuracy of the estimate using the fewest iterations are proposed and verified in simulations and experimentally based on the measurements acquired from an in-house developed model of a satellite attitude determination and control system. The proposed attitude estimation method based on the gradient descent algorithm and complementary filter automatically adjusts the number of iterations with the average below 0.5, reducing the demand on the processing power and energy consumption and causing it to be suitable for low-power applications.

Keywords: attitude estimation; rotational quaternion; Euler angles; gradient descent algorithm; complementary filter; optimization



Citation: Sever, K.; Golušin, L.M.; Lončar, J. Optimization of Gradient Descent Parameters in Attitude Estimation Algorithms. *Sensors* **2023**, *23*, 2298. <https://doi.org/10.3390/s23042298>

Academic Editors: Daniele Mortari and Bruno Tiribilli

Received: 31 January 2023
Revised: 15 February 2023
Accepted: 16 February 2023
Published: 18 February 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The attitude estimation of an object has a significant role in many different areas of technology: from consumer electronics [1] and human motion analysis [2,3] to robotics [4] and aerospace technology [5–8]. For each of these technologies, precise and accurate attitude measurements are of great importance. Specifically, the choice of sensors and advanced processing techniques determines the precision and accuracy. In general, the attitude is expressed as the rotation between the Earth's inertial frame and the body's local frame. This rotation in 3D space can be expressed by using Euler angles [9], or more commonly by using quaternions [10,11]. Although less intuitive, quaternions avoid the singularity of Euler angles known as the gimbal lock [12], which is the main reason of quaternion supremacy over Euler angles. The attitude estimation algorithms are based on a mathematical description of a body frame rotation with respect to a global, usually inertial, frame. In the literature, various direct attitude estimation methods have been reported, such as the q-method [13], Estimator of Quaternion (ESOQ) [14], Quaternion Estimator (QUEST) [15], and Recursive Quaternion Estimator (REQUEST and Optimal-REQUEST) [16,17]. Since attitude estimation is necessarily based on multi-sensor observations, numerous sensor fusion techniques have been developed. These techniques combine measurements from various different sensors into a single, more accurate, and reliable attitude estimate. Often, an integrated inertial measurement unit (IMU) consisting of a tri-axis accelerometer, a tri-axis magnetometer, and a tri-axis gyroscope provides a low cost, lightweight, and easily accessible measurement system [18–21]. Although easily accessible, these sensors

have some major drawbacks that complicate the attitude estimation. Accelerometers are used as the devices for measuring gravitational acceleration, which points directly toward the center of Earth. However, other than gravitational acceleration, accelerometers measure any acceleration caused by forces acting on the sensor. On the other hand, magnetometers suffer from the influence of hard and soft iron effects caused by different magnetic materials present in the vicinity of the sensor interfering with the measurement and calibration. In terms of the noise level, gyroscopes show the highest quality of the three. They provide information about the angular velocity of an object and require numerical integration for attitude estimation that results in very low noise levels. Nevertheless, they suffer from an error accumulation caused by the ever-present bias. Although IMU sensors cannot provide reliable attitude estimation on their own, through sensor fusion techniques, adequate accuracy can be attained. By far the most widespread sensor fusion technique for attitude estimation used in practice is the Kalman filter [19–21] due to its high accuracy and efficiency. Despite the advantages of the Kalman filter, its implementation is more complex and requires higher computational power, which may not be suitable for low-power applications. In systems where the usage of a Kalman filter is neither possible nor preferable, less-complex sensor fusion techniques such as a complementary filter may be utilized [22–25]. Complementary filters offer an efficient, computationally inexpensive, and accurate attitude estimate by combining high-pass filtered gyroscope attitude estimate with the low-pass filtered accelerometer and a magnetometer attitude estimate. As mentioned above, there are many direct methods for attitude estimation that could be used for attitude estimation using an accelerometer and magnetometer. In this paper, we focus on one of the most recent iterative techniques based on the gradient descent algorithm [22,26–30].

The gradient descent algorithm combines accelerometer and magnetometer measurements to estimate the optimal rotational quaternion in terms of the least square error. Since the algorithm is iterative, the required number of iterations is quite an important parameter that ultimately affects the accuracy and processing time. Too few or too many iterations of the gradient descent algorithm indicate a suboptimal set of gradient descent parameters, which leads to either the inaccurate estimation of the rotational quaternion or wasting computational resources, as it will be detailed throughout the paper. Thus, the number of iterations is crucial in real time applications. However, this is often overlooked or completely ignored in the literature. Some reports state that a single iteration of the gradient descent algorithm is sufficient for accurate attitude estimation provided that the convergence rate of the estimated orientation is equal or greater than the rate of change in the object rotation [30]. However, the number of iterations depends on many parameters and is difficult to predict. Thus, in this paper, we propose an optimization method of the gradient descent parameters applied to the estimation of the rotational quaternion. We propose a novel figure of merit and the criterion based on which the iterative process is terminated. The termination criterion allows for an automatic determination of the required number of iterations. Using this approach, we show that the average number of iterations of less than one can be achieved. Furthermore, we present the analysis of three methods for the initial quaternion propagation, noise analysis, and experimental verification.

The paper is organized as follows. Section 2 describes the attitude representation based on the Euler angles and rotational quaternion, as well as the relation between them. Section 3 brings the mathematical background behind the attitude estimation based on the gradient descent algorithm and complementary filter. Section 4 provides a detailed description of the optimization of each gradient descent parameter. Section 5 describes the experimental results and observations, while Section 6 brings final conclusions and guidelines for future work.

2. Attitude Representation

The attitude of a body can be expressed as a rotation between the coordinate system of a body relative to the inertial, most often Earth's, coordinate system. A simple and intuitive representation of an object's orientation in three-dimensional (3D) space can be

obtained from Euler angles—yaw (ψ), pitch (θ), and roll (ϕ), which represent rotations of a body around three orthogonal axes: z -, y -, and x -axis, respectively, as shown in Figure 1. Each of the three mentioned rotations can be expressed in the form of a rotation matrix. Multiplying the three matrices yields a single rotation matrix known as the direction cosine matrix (DCM). Here, it should be noted that the Euler rotational sequence is important due to the fact that matrix multiplication is not commutative. In the following text, a conventional $z - y - x$ rotation sequence for aerospace applications will be used.

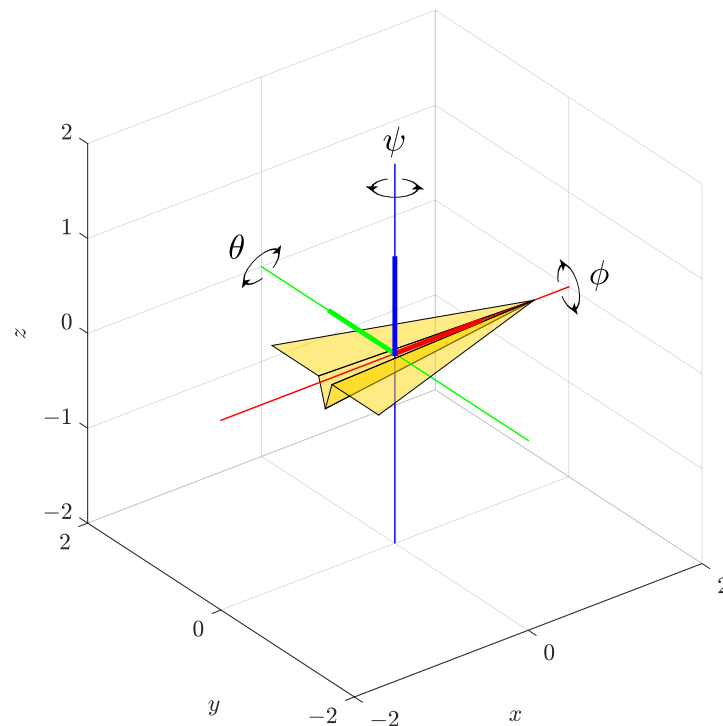


Figure 1. Euler angles ψ , θ , and ϕ representing the attitude of an object in 3D space.

The first rotation is applied around the z -axis of the inertial coordinate system for angle ψ . This rotation defines a new local coordinate system V_1 whose z -axis is aligned with the z -axis of the inertial coordinate system, while the x and y -axis of the coordinate system V_1 are rotated for angle ψ . This rotation is represented by the matrix $\mathbf{R}_i^{v_1}(\psi)$.

$$\mathbf{R}_i^{v_1}(\psi) = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Similarly, matrix $\mathbf{R}_{v_1}^{v_2}(\theta)$ represents the rotation around y -axis of the coordinate system V_1 (defined by the previous rotation) for angle θ . The rotation creates a new local coordinate system V_2 whose y -axis is aligned with the y -axis of the coordinate system V_1 , while the x and z -axis of the coordinate system V_2 are rotated for angle θ with respect to the coordinate system V_1 .

$$\mathbf{R}_{v_1}^{v_2}(\theta) = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \quad (2)$$

Finally, matrix $\mathbf{R}_{v_2}^b(\phi)$ defines a rotation around the x -axis of the coordinate system V_2 for angle ϕ . Again, the rotation creates a new, body coordinate system B whose x -axis is aligned with the x -axis of the V_2 coordinate system, while the y and z -axis of the coordinate system B are rotated for angle ϕ with respect to the coordinate system V_2 .

$$R_{v_2}^b(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \tag{3}$$

To obtain the DCM matrix, the rotation matrices defined by (1)–(3) are multiplied respecting the above-presented $z - y - x$ sequence.

$$R_i^b(\phi, \theta, \psi) = R_{v_2}^b(\phi) R_{v_1}^{v_2}(\theta) R_i^{v_1}(\psi) \tag{4}$$

The DCM elements can be expressed as functions of Euler angles $\psi, \theta,$ and ϕ as follows:

$$R_i^b(\phi, \theta, \psi) = \begin{bmatrix} \cos(\psi) \cos(\theta) & \cos(\theta) \sin(\psi) & -\sin(\theta) \\ \cos(\psi) \sin(\phi) \sin(\theta) - \cos(\phi) \sin(\psi) & \cos(\phi) \cos(\psi) + \sin(\phi) \sin(\psi) \sin(\theta) & \cos(\theta) \sin(\phi) \\ \sin(\phi) \sin(\psi) + \cos(\phi) \cos(\psi) \sin(\theta) & \cos(\phi) \sin(\psi) \sin(\theta) - \cos(\psi) \sin(\phi) & \cos(\phi) \cos(\theta) \end{bmatrix}. \tag{5}$$

Despite being simple and intuitive, the attitude representation based on Euler angles is not preferred in many applications due to the effect known as gimbal lock, a singularity of Euler angles that manifests itself in the loss of a rotational degree of freedom. When using Euler angles for attitude representation, the gimbal lock is present regardless of the chosen rotational sequence. For the $z - y - x$ rotation sequence, the gimbal lock occurs for $\theta = \pm 90^\circ$.

To avoid this problem, an alternative mathematical description of body orientation can be used. One of the most common and practical alternatives for Euler angles are rotational quaternions. The rotational quaternions are generalized complex numbers consisting of one real and three imaginary, mutually orthogonal units, which can likewise represent the rotation between the two coordinate systems. Unlike Euler angles, which depend on the rotation sequence, the rotational quaternions can be described by a single angle of rotation α around the unit vector $n = n_x \hat{i} + n_y \hat{j} + n_z \hat{k}$, representing the axis of rotation ($\hat{i}, \hat{j},$ and \hat{k} are basis vectors of the inertial Cartesian coordinate system).

$$n = \sqrt{n_x^2 + n_y^2 + n_z^2} = 1 \tag{6}$$

The rotational quaternion can be represented by the four-dimensional (4D) vector defined as follows:

$$q = \cos\left(\frac{\alpha}{2}\right) + \sin\left(\frac{\alpha}{2}\right)n = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} \cos(\alpha/2) \\ n_x \sin(\alpha/2) \\ n_y \sin(\alpha/2) \\ n_z \sin(\alpha/2) \end{bmatrix}. \tag{7}$$

Similar to (5), the elements of DCM can be expressed as functions of quaternion elements.

$$R_i^b(q) = \begin{bmatrix} q_1^2 + q_2^2 - q_3^2 - q_4^2 & 2(q_2q_3 + q_1q_4) & 2(q_2q_4 - q_1q_3) \\ 2(q_2q_3 - q_1q_4) & q_1^2 - q_2^2 + q_3^2 - q_4^2 & 2(q_3q_4 + q_1q_2) \\ 2(q_2q_4 + q_1q_3) & 2(q_3q_4 - q_1q_2) & q_1^2 - q_2^2 - q_3^2 + q_4^2 \end{bmatrix} \tag{8}$$

Using (5) and (8), a direct relation between Euler angles and the rotational quaternion can be found. Thus, the Euler angles expressed as functions of rotation quaternion can be written as:

$$\begin{bmatrix} \psi \\ \theta \\ \phi \end{bmatrix} = \begin{bmatrix} \arctan\left(\frac{r_{12}}{r_{11}}\right) \\ -\arcsin\left(\frac{r_{13}}{r_{11}}\right) \\ \arctan\left(\frac{r_{23}}{r_{33}}\right) \end{bmatrix} = \begin{bmatrix} \arctan\left(\frac{2(q_2q_3 + q_1q_4)}{q_1^2 + q_2^2 - q_3^2 - q_4^2}\right) \\ -\arcsin\left(\frac{2(q_2q_4 - q_1q_3)}{q_1^2 - q_2^2 - q_3^2 + q_4^2}\right) \\ \arctan\left(\frac{2(q_3q_4 + q_1q_2)}{q_1^2 - q_2^2 - q_3^2 + q_4^2}\right) \end{bmatrix}. \tag{9}$$

3. Attitude Estimation Methods Based on Gradient Descent Algorithm

3.1. Attitude Estimation Based on Gradient Descent Algorithm

The gradient descent algorithm is an iterative algorithm often used in optimization problems for finding the local minimum of an arbitrary cost function J . Applied to the problem of attitude estimation, the cost function is simply a function of the rotational quaternion. The gradient descent can be defined by a simple recurrence relation:

$$\mathbf{q}_{k+1} = \mathbf{q}_k - \mu \nabla J(\mathbf{q}_k), \quad (10)$$

where \mathbf{q} is the rotational quaternion and μ is a step size that defines the convergence rate. The algorithm starts by assuming an initial quaternion \mathbf{q}_0 that is updated for each k as the cost function $J(\mathbf{q}_k)$ approaches its minimum. The smaller the μ , the longer it requires for the algorithm to converge. On the other hand, if μ is too large, the algorithm may diverge and the minimum of the cost function may not be reached. Therefore, μ is often experimentally determined. Since our goal is to find the rotational quaternion that optimally represents the attitude of a body, the cost function being minimized is $J(\mathbf{q}) : \mathbb{R}^4 \rightarrow \mathbb{R}$. To define the cost function, let us first define the error vectors.

Let the vector \mathbf{a}^b be the body frame gravitational acceleration vector and let \mathbf{R}_i^b be the rotation matrix that describes the relation between the inertial and body reference frames. Ideally, if \mathbf{R}_i^b is a function of the correct rotation quaternion, the vector \mathbf{a}^b can be calculated as $\mathbf{a}^b = \mathbf{R}_i^b \mathbf{a}^i$, with \mathbf{a}^i being the referent vector of gravitational acceleration expressed in the inertial frame of reference. If the rotational quaternion defining the matrix \mathbf{R}_i^b does not correctly describe the attitude of a body, the error vector \mathbf{e}_a can be calculated as:

$$\mathbf{e}_a = \mathbf{R}_i^b \mathbf{a}^i - \mathbf{a}^b. \quad (11)$$

The error vector \mathbf{e}_b , which describes the rotational error of the magnetic field vector, is calculated in a similar manner.

$$\mathbf{e}_m = \mathbf{R}_i^b \mathbf{m}^i - \mathbf{m}^b \quad (12)$$

Vectors \mathbf{a}^i and \mathbf{m}^i represent known referent vectors of the gravitational acceleration and Earth's magnetic field, respectively, represented in the inertial frame of reference. Using (11) and (12), we define the cost function $J(\mathbf{q})$ as follows:

$$J(\mathbf{q}) = \mathbf{e}_a^T \mathbf{e}_a + \mathbf{e}_m^T \mathbf{e}_m. \quad (13)$$

Notice that, in ideal lossless conditions, the value of the cost function is equal to zero for the rotational quaternion representing the true attitude of an object, since both error vectors are equal to zero vectors. In reality, however, this is not the case due to unavoidable noise and sensor imperfections. Although these imperfections cannot be eliminated, the value of the cost function can be minimized using the gradient descent algorithm. In order to apply the gradient descent algorithm, one needs to derive the expression for the gradient of the cost function. To do so, the partial derivatives of the cost function (13) with respect to the i -th element of the rotation quaternion are derived.

$$\frac{\partial}{\partial q_i} J(\mathbf{q}) = 2\mathbf{e}_a^T \left(\frac{\partial}{\partial q_i} \mathbf{R}_i^b \right) \mathbf{a}^i + 2\mathbf{e}_m^T \left(\frac{\partial}{\partial q_i} \mathbf{R}_i^b \right) \mathbf{m}^i \quad (14)$$

Here, $\partial \mathbf{R}_i^b / \partial q_i$ represents the partial derivative of the rotation matrix \mathbf{R}_i^b (8), with respect to the i -th element of the rotation quaternion, as provided below.

$$\mathbf{M}_1 = \frac{\partial}{\partial q_1} \mathbf{R}_i^b = 2 \begin{bmatrix} q_1 & q_4 & -q_3 \\ -q_4 & q_1 & q_2 \\ q_3 & -q_2 & q_1 \end{bmatrix} \quad (15a)$$

$$\mathbf{M}_2 = \frac{\partial}{\partial q_2} \mathbf{R}_i^b = 2 \begin{bmatrix} q_2 & q_3 & q_4 \\ q_3 & -q_2 & q_1 \\ q_4 & -q_1 & -q_2 \end{bmatrix} \quad (15b)$$

$$\mathbf{M}_3 = \frac{\partial}{\partial q_3} \mathbf{R}_i^b = 2 \begin{bmatrix} -q_3 & q_2 & -q_1 \\ q_2 & q_3 & q_4 \\ q_1 & q_4 & -q_3 \end{bmatrix} \quad (15c)$$

$$\mathbf{M}_4 = \frac{\partial}{\partial q_4} \mathbf{R}_i^b = 2 \begin{bmatrix} -q_4 & q_1 & q_2 \\ -q_1 & -q_4 & q_3 \\ q_2 & q_3 & q_4 \end{bmatrix} \quad (15d)$$

Finally, using (14) and (15), one defines the gradient of the cost function at \mathbf{q}_k .

$$\nabla J(\mathbf{q}_k) = 2 \begin{bmatrix} \mathbf{e}_a^T \mathbf{M}_1 \mathbf{a}^i(\mathbf{q}_k) + \mathbf{e}_m^T \mathbf{M}_1 \mathbf{m}^i(\mathbf{q}_k) \\ \mathbf{e}_a^T \mathbf{M}_2 \mathbf{a}^i(\mathbf{q}_k) + \mathbf{e}_m^T \mathbf{M}_2 \mathbf{m}^i(\mathbf{q}_k) \\ \mathbf{e}_a^T \mathbf{M}_3 \mathbf{a}^i(\mathbf{q}_k) + \mathbf{e}_m^T \mathbf{M}_3 \mathbf{m}^i(\mathbf{q}_k) \\ \mathbf{e}_a^T \mathbf{M}_4 \mathbf{a}^i(\mathbf{q}_k) + \mathbf{e}_m^T \mathbf{M}_4 \mathbf{m}^i(\mathbf{q}_k) \end{bmatrix} \quad (16)$$

The performance of the gradient descent algorithm in attitude estimation depends on a number of parameters, such as the step size μ , maximum number of iterations N_{max} , and initial quaternion \mathbf{q}_0 , investigated in the following sections. Moreover, special care is devoted to defining the termination criterion and investigating different methods of initial quaternion propagation. To optimize the performance of the gradient descent algorithm, one should choose the optimal set of parameters that yields the highest accuracy of the estimates obtained in the fewest number of iterations.

3.2. Complementary Filter Based on Gradient Descent Algorithm

A complementary filter is a common approach for the fusion of different sensor types. It uses a low-pass and a high-pass filter with the same cutoff frequency to maintain the advantages and reject the disadvantages of each sensor type. The implementation of the complementary filter is very straightforward and lightweight, which is why it is often preferred in the low-power applications. The complementary filter combines the estimates obtained using different sensors and techniques using a simple weighted sum. For example, the rotation quaternion can be estimated by using an accelerometer and a magnetometer through the gradient descent algorithm as described above. Let us denote this estimated quaternion with \mathbf{q}_{gd} . Likewise, it can also be estimated based on the angular velocity measured using a gyroscope. This estimate is denoted by \mathbf{q}_ω . The weighted sum, representing the implementation of the complementary filter, can be written simply as:

$$\mathbf{q} = K\mathbf{q}_\omega + (1 - K)\mathbf{q}_{gd}. \quad (17)$$

Here, K represents scaling factor (also known as the complementary filter gain) and \mathbf{q} the final estimate of the rotation quaternion. Since a gyroscope is a rate sensor (i.e., it provides the information about the angular velocity $\boldsymbol{\omega}$ of an object), \mathbf{q}_ω is estimated through the process of numerical integration as follows:

$$\mathbf{q}_k = \mathbf{q}_{k-1} + \Delta T \dot{\mathbf{q}}_k, \quad (18)$$

where ΔT is the time between two samples and $\dot{\mathbf{q}}_k$ is the rate of change (i.e., the time derivative) of the rotation quaternion in step k . $\dot{\mathbf{q}}$ can be calculated from the angular velocity as follows [31]:

$$\dot{\mathbf{q}} = \frac{d}{dt} \mathbf{q} = \frac{1}{2} \boldsymbol{\Omega}(\boldsymbol{\omega}) \mathbf{q}. \quad (19)$$

Here, the matrix $\Omega(\omega)$ is defined as [31]:

$$\Omega(\omega) = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix}. \quad (20)$$

While the process of numerical integration highly reduces the measured noise, integrating the bias present in gyroscopes leads to the inevitable drift of the estimated rotational quaternion. Thus, q_ω is high-pass filtered through the complementary filter. Moreover, gyroscopes as the rate sensors do not provide information about the absolute attitude of an object. Here, the absolute attitude is provided by the accelerometer and magnetometer. These sensors are much more affected by noise. Thus, q_{gd} is low-pass filtered through the complementary filter. Consequently, the final estimate inherits the advantages of each estimation method, while the summation of the rotational quaternion in general does not maintain the unit length of the quaternion, it is applicable to the quaternions close to each other in the quaternion space. Nevertheless, it is recommended to normalize the final estimated quaternion to enforce the unit length and preserve the convexity of the estimation problem [32].

4. Simulations and Optimization of Gradient Descent Parameters

As mentioned in the previous sections, there are several parameters that determine the performance of the gradient descent algorithm in the estimation of the rotational quaternion. These parameters are the step size μ , maximum allowed number of iterations N_{max} , and initial rotational quaternion q_0 . In this section, we analyze the effect of these parameters on the number of iterations and principal angle error (PAE) of the estimate. Moreover, we present and evaluate different methods of initial quaternion propagation and introduce yet another parameter (G_{max}) as the figure of merit and termination criterion. The goal is to present the methodology and guidelines for selecting the optimal set of parameters that provides the highest accuracy of the estimate using the fewest iterations, which represent the opposing requirements in iterative algorithms. Finding the optimal parameter set leads to lower power consumption and lowers the required computational power, which may be of crucial importance, especially in low-power applications.

The simulations and results presented in this section are based on the software generated three-axes gravitational acceleration, three-axes magnetic field, and three-axes angular velocity characteristic for IMUs. In these simulations, the referent vector of gravitational acceleration is chosen to be $a^i = [0 \ 0 \ 9.81]^T$ m/s², and the referent vector of the magnetic field $m^i = [22.2 \ 1.7 \ 42.7]^T$ μ T, according to the World Magnetic Model [33] at the location 45°48'26" N, 15°58'3" E, representing the coordinates of Zagreb, Croatia. The software generated data are based on the ground truth rotational quaternions known for every sample, which allow for the calculation of the PAE. The PAE is extracted from the corrective rotational matrix defined as $R^T \hat{R}$, R , and \hat{R} being the true and estimated rotational matrices, respectively. For each sample, the PAE is calculated as:

$$PAE = \arccos\left(\frac{1}{2}\left(\text{Tr}(R^T \hat{R}) - 1\right)\right). \quad (21)$$

The simulations are performed in an ideal, noiseless condition as well as with the Gaussian white noise added to the software generated data. The noise levels of all three sensors (accelerometer, magnetometer, and gyroscope) correspond to the noise levels of the low cost IMU MPU9250 used as a reference. Before the noise analysis, the magnetometer calibration procedure was performed. To eliminate the unwanted artifacts, the noise levels were extracted from the measurement of more than 3500 samples with the IMU in static position. The noise variance applied to the software generated data is the highest

variance extracted from the measurement for each sensor, namely, $0.2 \times 10^{-3} \text{ (m/s}^2\text{)}^2$ for the accelerometer, $0.65 \text{ (}\mu\text{T)}^2$ for the meagnetometer, and $0.35 \times 10^{-6} \text{ (rad/s)}^2$ for the gyroscope.

4.1. Selection of Step Size

The step size, denoted by μ in (10), is one of the most important parameters in the gradient descent algorithm. It is a positive real number that determines how fast the estimate converges toward the ground truth. If not chosen carefully, it may cause divergence of the algorithm. The step size is often empirically determined. In order to select the optimal step size, let us assume that the attitude of an object is represented by the rotational quaternion q defined by the arbitrarily chosen vector of rotation $n = [1 \ -1 \ 0]^T$ and the angle of rotation α . Based on (7), for the given angle α , the true rotational quaternion q can be calculated. Note that the vector of rotation n should be normalized before calculating the rotational quaternion. Using (8), the rotation matrix $R_i^b(q)$ can be calculated, which, together with the known reference vectors of the gravitational acceleration a^i and the magnetic field m^i expressed in the inertial frame of reference, were used to calculate the representation of the same physical quantities in the object's body frame ($a^b = R_i^b a^i$ and $m^b = R_i^b m^i$).

In this simulation, no noise was added to the calculated body frame vectors. The body frame vectors were passed as the inputs to the gradient descent algorithm implemented in MathWorks Matlab as described in the previous section. The initial rotational quaternion was chosen to be $q_0 = [1 \ 0 \ 0 \ 0]^T$. The value of the cost function $J(q)$ for each iteration of the gradient descent algorithm is plotted in Figure 2 for different values of step size μ and angle α .

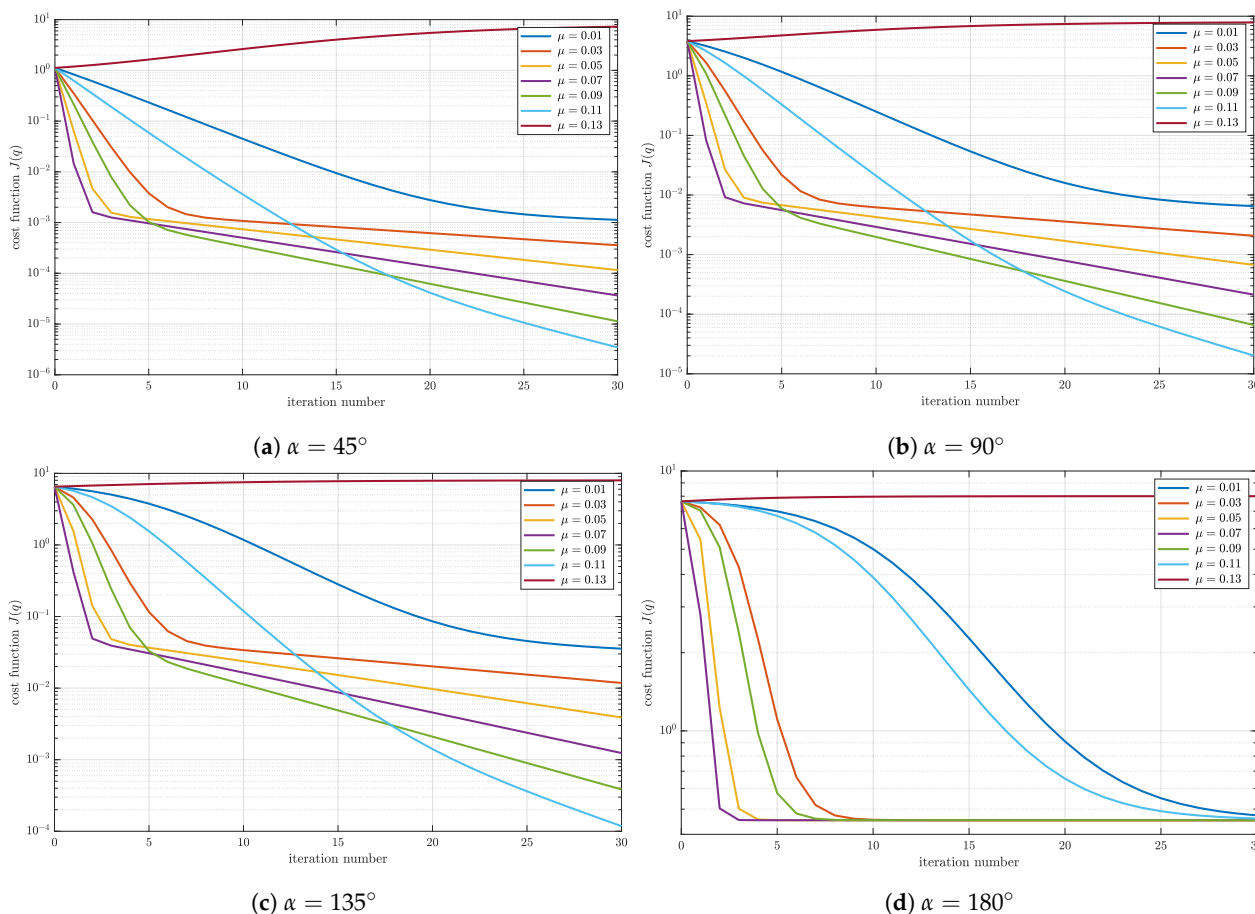


Figure 2. Cost function $J(q)$ of each iteration for different step size μ and angle of rotation α .

Focusing on Figure 2a plotted for the rotation angle $\alpha = 45^\circ$, one can observe that the convergence rate of the cost function highly depends on the step size value. The values of tested step sizes are chosen in the range from 0.01 to 0.13. The initial value of the cost function is equal for each step size. As expected, the value of the cost function drops with each iteration indicating the drop of error in estimated rotational quaternion (13), except for $\mu = 0.13$. This is the largest step size in the range and causes the divergence of the gradient descent, which manifests in monotonic growth of the cost function with each iteration. For the other values of the step size, a clear difference in steepness is observable. For all step sizes (except $\mu = 0.13$), the knee in each graph that represents a boundary between the regions of higher (for lower iteration numbers) and lower (for the higher iteration numbers) steepness of the cost function can be detected. The difference in steepness causes certain values of the step size to be a better choice for certain applications. For example, if the expected number of iterations is lower than 5, one should choose the step size with the steepest graph in that region (i.e., $\mu = 0.07$) since this step size yields the estimate in lowest number of iterations. On the other hand, different step size may be more suitable if an application requires larger number of iterations. If the number of iterations exceeds 15, the optimal choice represents the step size $\mu = 0.11$.

Although the analysis presented above is based on the specific case of the rotational quaternion defined by the specific angle of rotation $\alpha = 45^\circ$, the same behavior is observable for different angles of rotation presented in Figure 2b–d. By increasing the angle of rotation, the distance between the true and initial rotational quaternion in quaternion space increases. Because of that, the initial value of the cost function increases with the angle α . Moreover, in the 30 iterations presented in Figure 2, different final values of the cost function are achieved indicating the difference in accuracy of the estimate. This behavior is characteristic for iterative algorithms. However, notice that the step sizes yielding the fastest convergence for the lower and the higher iteration numbers remain $\mu = 0.07$ and $\mu = 0.11$, respectively. Since the number of iterations in attitude estimation based on the presented algorithm is expected to be small, the step size is set to $\mu = 0.07$.

4.2. Termination of Iterative Process

In the previous examples, the step size and the cost function are analyzed with the fixed number of iterations set to 30, although, in the steady state, the number of iterations used for attitude estimation is expected to be much lower. In some publications, a single iteration is assumed to be sufficient [30,34]. However, this hypothesis is difficult to prove without setting a figure of merit. Here, we analyze different methods and approaches for the termination of the gradient descent iterative process. In general, the process can be terminated based on the number of iterations or desired accuracy. These two criteria represent opposing requirements.

The first and the simplest approach for the termination of the iterative process is setting the maximum number of iterations N_{max} . In perfect noiseless conditions, as the number of iterations approaches infinity, an estimate approaches the ground truth. Thus, setting the finite number of iterations leads to the loss of accuracy. On the other hand, aiming for the highest accuracy requires large (theoretically infinite) number of iterations, which significantly increases the processing time. If the time constraint is the limiting factor of the application, one can reduce the duration of the gradient descent algorithm by limiting the maximum number of iterations by setting N_{max} . The value of N_{max} highly depends on the processing power of the targeted processing unit.

If, on the other hand, the desired accuracy of the estimate is reached, the iterative process should be terminated. However, a figure of merit is needed to properly set the termination criterion. In ideal lossless conditions, the termination criterion can be set based on the value of the cost function. Recall that, if the step size is set correctly, the value of the cost function monotonically decreases with each iteration. If the value of the cost function $J(q)$ drops below some a priori defined value J_{max} , the desired accuracy is reached and the iterative process can be terminated. Thus, J_{max} represents the user defined positive real

number that sets the highest acceptable error and thus the desired accuracy of the estimate. The lower the J_{max} , the higher the accuracy. Unfortunately, such a criterion is impractical in real world applications. The reason for that lies in the fact that the measurements of physical quantities based on which the rotational quaternion is estimated are always affected by noise. In the presence of noise, the gradient descent algorithm is still able to find the rotational quaternion that minimizes the cost function, however, the value of the cost function at its minimum is no longer equal to zero, causing it to be practically impossible to reach the desired accuracy by setting J_{max} . However, as the estimate approaches the ground truth, the difference between the values of the cost function in current and previous iteration becomes smaller. Eventually, at the minimum of the cost function, the difference becomes equal to zero. Thus, the termination criterion can be set based on parameter ΔJ_{max} . Such a criterion is reported in [26]. Here, we proposed yet another criterion based on the gradient of the cost function $\nabla J(\mathbf{q}_k)$ defined by (16). Indeed, calculating the difference between two neighboring points of the cost function is very similar to finding its derivative. This information is also contained in the gradient itself, which is calculated in each iteration of the gradient descent algorithm. If the minimum of the cost function is reached, $\nabla J(\mathbf{q}_k)$ becomes a zero vector. Thus, to determine the accuracy of the gradient descent algorithm, we introduce the scalar representation of the gradient G defined as:

$$G = (\nabla J(\mathbf{q}))^T \nabla J(\mathbf{q}). \quad (22)$$

The iterative process can be terminated in the k -th iteration if $G_k < G_{max}$, here G_{max} is an arbitrarily chosen user-defined positive real number. Similar to ΔJ_{max} , by decreasing the value of G_{max} , the accuracy of the quaternion estimation increases at the cost of the number of iterations.

To gain a better understanding of how N_{max} and G_{max} affect the quality of the rotational quaternion estimation, another simulation is performed. In this simulation, it is assumed that an object rotates with the constant angular speed of 20 deg/s around an arbitrary chosen vector of rotation $\mathbf{n} = [1 \ -1 \ 0]^T$. The initial angle of rotation around the vector \mathbf{n} is chosen to be $\alpha_0 = 45^\circ$. Based on the provided data, the true rotational quaternion is calculated for each time step. The time period between the calculated quaternion is chosen to be 0.1 s. The true quaternion is then used to calculate the body frame gravitational acceleration and magnetic field that are passed to the gradient descent algorithm as the input arguments at each time step. The initial quaternion for the very first iterative process was chosen to be $\mathbf{q}_0 = [1 \ 0 \ 0 \ 0]^T$. However, the rotational quaternion estimated in every other iterative process is passed as the initial quaternion for the next iterative process. This is one of the initial quaternion propagation methods that will be discussed in detail further on. In this simulation, the number of iterations and PAE were monitored with G_{max} fixed to 1×10^{-5} for different N_{max} in the range from 5 to 80. The step size is chosen to be $\mu = 0.07$. The simulation results are shown in Figure 3.

In Figure 3a, one can observe that N_{max} has a dominant effect on the duration of the transient, i.e., the number of samples of the gravitational acceleration and the magnetic field needed for the algorithm to converge. Indeed, the higher the N_{max} , the quicker the steady state is reached. In a steady state, the number of iterations required for reaching the accuracy defined by $G_{max} = 1 \times 10^{-5}$ is equal to 9 for every N_{max} , except for $N_{max} = 5$. Notice that, for $N_{max} = 5$, the number of iterations is maxed out during the whole simulation period. This indicates that five iterations are simply not enough for reaching the required accuracy. This is also evident in Figure 3b. Notice that, in a steady state, the same PAE is reached for each N_{max} except for $N_{max} = 5$. The PAE is greater for the $N_{max} = 5$, therefore indicating an insufficient number of iterations. Furthermore, one can notice that, during the transient, the number of iterations is maxed out for each N_{max} except for $N_{max} = 80$ for which the steady state is reached in only 0.1 s, indicating a single sample of gravitational acceleration and magnetic field.

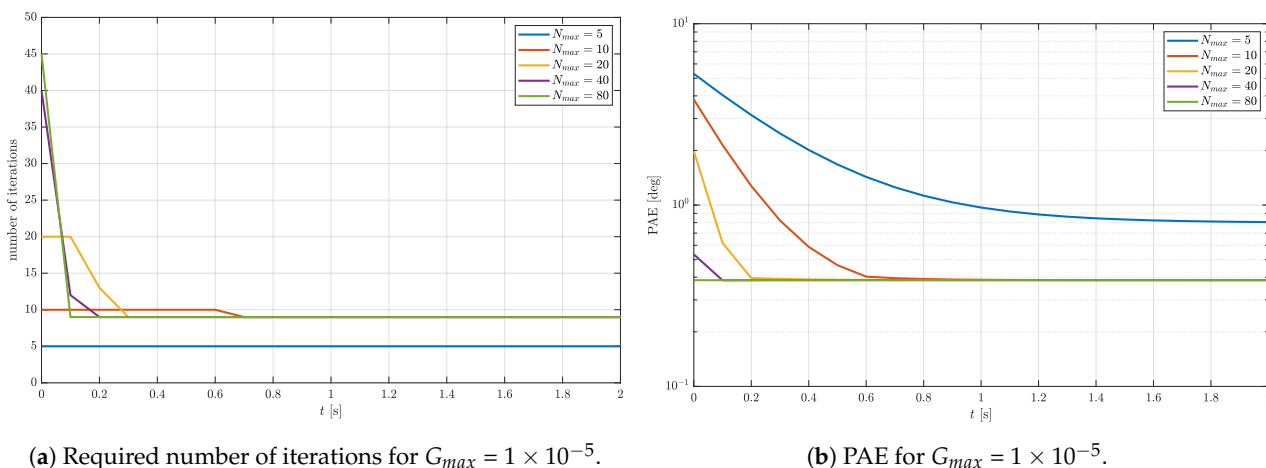


Figure 3. Required number of iterations and PAE of the estimated rotational quaternion ($\mu = 0.7$).

The same simulation is repeated with N_{max} set to the fixed value of 100, while G_{max} is varied in the range from 1×10^{-9} to 1×10^{-5} . Since the number of iterations is set to a relatively high value, a quick convergence is expected for each G_{max} . This is confirmed by the results shown in Figure 4a. However, the number of iterations in a steady state is not equal for every G_{max} . The lower the G_{max} , the higher the number of iterations required for reaching the desired accuracy. This is evident from Figure 4b as well. If G_{max} is smaller, the small PAE is reached in a steady state. This is a clear confirmation that parameters G defined in (22) and G_{max} represent a valid figure of merit and termination criterion.

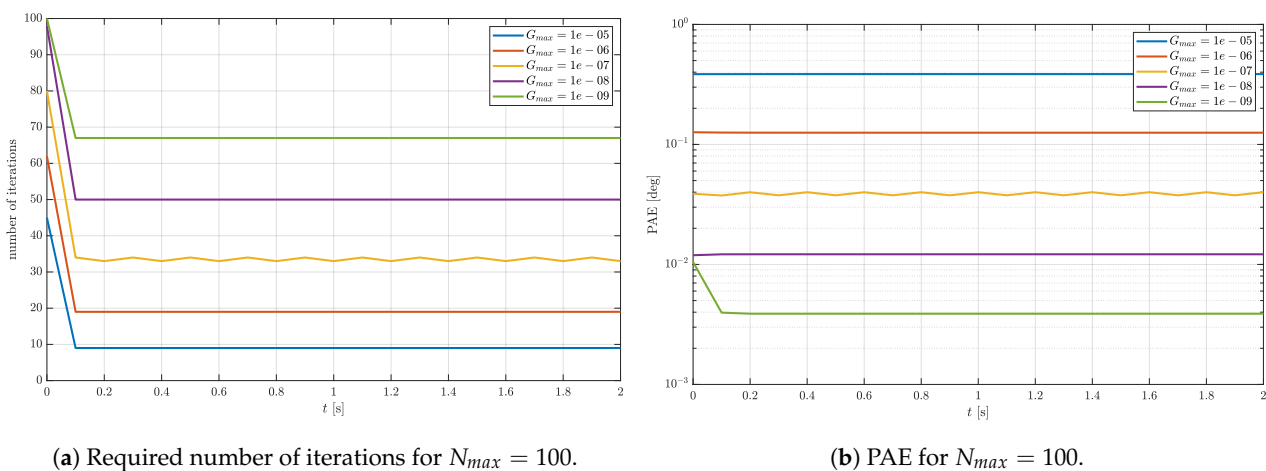


Figure 4. Required number of iterations and PAE of the estimated rotational quaternion $\mu = 0.7$.

The simulation results presented above show that, if the G_{max} is too high, the accuracy of the estimate is too low. On the other hand, if the G_{max} is too low, the desired accuracy may be too high with respect to N_{max} . Increasing N_{max} leads to a larger number of iterations. If a targeted processing unit is capable of processing a large number of iterations in a given amount of time, this is a perfectly viable option in lossless conditions. However, in the presence of noise, a higher number of iterations does not necessarily mean higher accuracy. Thus, setting G_{max} too low may lead to wasting computational resources. Naturally, there is an optimal range of values of G_{max} , which should be determined using simulations or experiments. The optimal G_{max} minimizes the number of iterations required in a steady state. Here, we propose the method to determine the optimal G_{max} . The method is based on static simulation with noise added to the gravitational acceleration and magnetic field. The assumed attitude is represented by the rotational quaternion defined by the vector and angle of rotation, $n = [1 \ -1 \ 0]^T$ and $\alpha_0 = 45^\circ$, respectively. The initial quaternion is again chosen to be $q_0 = [1 \ 0 \ 0 \ 0]^T$. The rotation quaternion estimated for each sample

is passed as the initial quaternion in the next iterative process. In a steady state, without the presence of noise, the previously estimated quaternion is the same as the current quaternion since the simulation is static. Consequently, the number of iterations per iterative process is expected to drop to zero in a steady state, regardless of G_{max} . However, in the presence of noise, zero iterations per iterative process means that the selected value of G_{max} is not optimal. In that case, higher accuracy can be achieved by decreasing the value of G_{max} . G_{max} should be decreased until the number of iterations starts to increase. Decreasing the G_{max} even further leads to the increase in the number of iterations without gaining accuracy. Since the object is static, the additional iterations appear solely due to the sensor noise. This conclusion is supported by the simulation results shown in Figure 5 representing the number of iterations for G_{max} in a range from 1×10^{-1} to 5×10^{-4} .

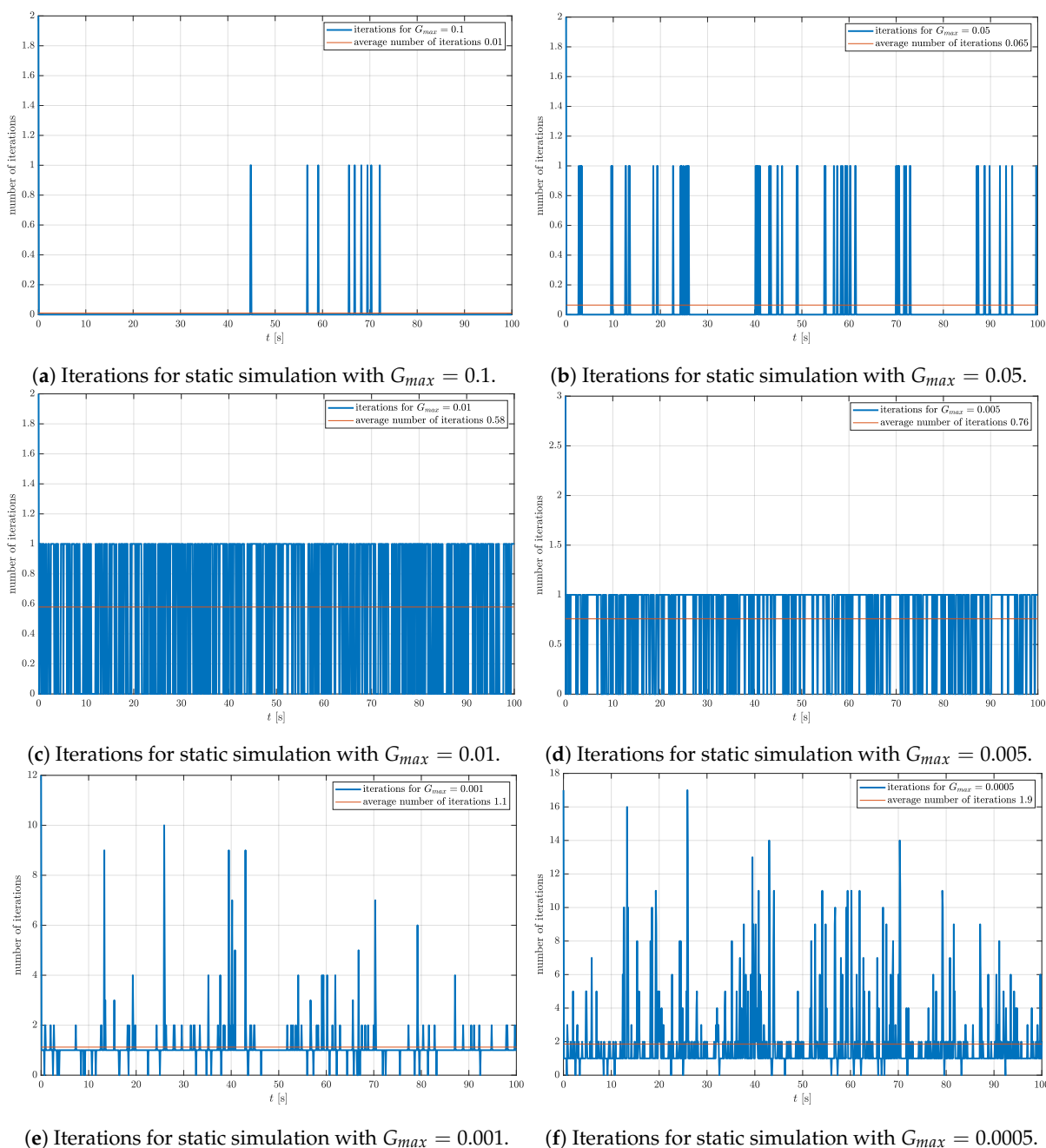


Figure 5. Number of gradient descent iterations required for estimation of rotation quaternion based on a previous estimate in static condition for six values of G_{max} in range from 1×10^{-1} to 5×10^{-4} and for $\mu = 0.7$.

All the graphs in Figure 5 have the highest number of iterations around $t = 0$ s caused by the difference between the true and the initial quaternion, which quickly vanishes. This effect is detailed in the previous section. Figure 5a,b represent results with the highest G_{max} . The average number of iterations over the simulation period of 100 s is below 0.1. Based on the presented results, one can conclude that the chosen values of G_{max} are too high and that the accuracy of the estimate can be further improved by decreasing the value of G_{max} . On the other hand, Figure 5e,f represent the opposite scenario. Although the average number of iterations is not unreasonably high (below 2), at some time steps, a significantly larger number of iterations is needed to achieve the desired accuracy. This is an indication that the selected G_{max} is too low, which results in a noisy estimate. Thus, one concludes that optimal G_{max} yields the average number of iterations around the midpoint of the 0–1 range. Figure 5c,d represent the results obtained for G_{max} equal to 1×10^{-2} and 5×10^{-3} , respectively. In these cases, the number of iterations does not exceed 1 at any time step. Furthermore, notice that, in both cases, the average number of iterations is closer to 0.5. Thus, the G_{max} is fixed to 1×10^{-2} for the next simulations in which we examine different methods of the initial quaternion propagation.

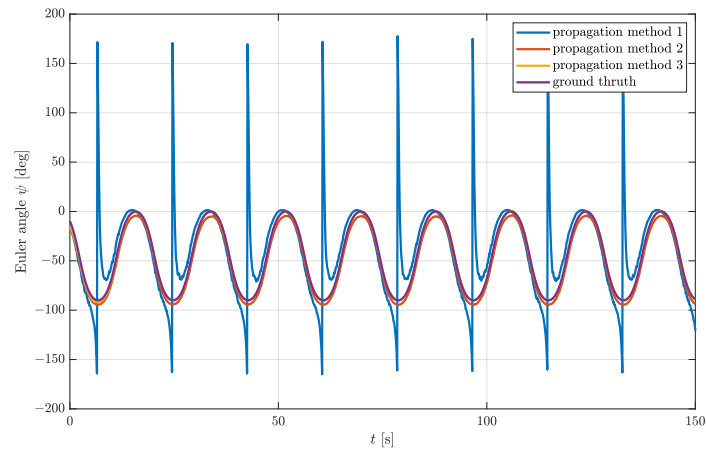
4.3. Propagation of Initial Quaternion

There are several approaches to selecting the initial quaternion q_0 for the gradient descent algorithm. Here, we present and examine three different propagation methods. In propagation method 1, a fixed arbitrarily selected rotation quaternion is used for each sample/iterative process. Since it is not possible to know the attitude of an object a priori, each rotational quaternion selected as an initial quaternion represents a reasonable guess. In propagation method 2, the rotational quaternion estimated in current iterative process is used as an initial quaternion of the next iterative process. This propagation method was briefly introduced in the sections above. Here, we use the assumption that the attitude of an object cannot change instantaneously (or cannot change much in a short period of time). Thus, the previously estimated quaternion is expected to be closer to the current quaternion, which reduces the number of iterations in comparison with the previous method. Finally, propagation method 3 uses an attitude estimate and information about the angular rate at the current time step to predict the rotational quaternion at the next time step. This prediction is then used as an initial quaternion in the next iterative process.

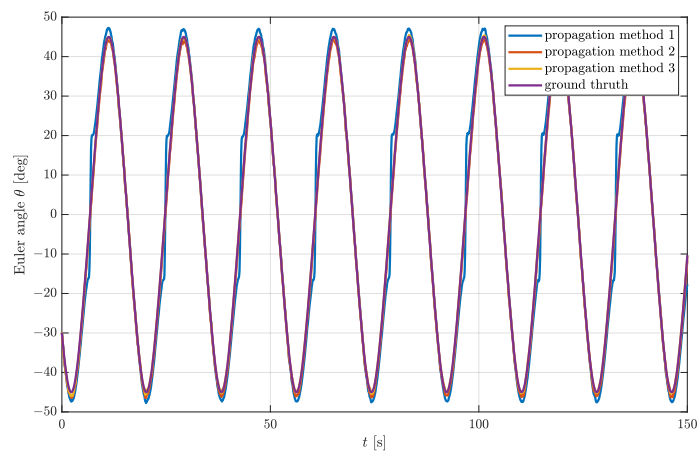
Unlike the other gradient descent based attitude estimation methods reported in the literature [30,34], which fix the number of iterations to 1, adaptively changing the step size with respect to the measured angular velocity, the iterative process here is performed with an initial quaternion closest to the true quaternion until the desired accuracy defined by $G_{max} = 1 \times 10^{-2}$ is reached. This method is expected to yield the lowest number of iterations. All three propagation methods are implemented as a part of the complementary filter. The complementary filter fuses the quaternions estimated using the gradient descent algorithm (based on the gravitational acceleration and magnetic field data) and quaternions estimated from the angular velocity. Here, the complementary filter gain is set to $K = 0.5$, providing the same weight to each of the quaternion estimation methods. The value of the parameter K is not optimized, as it is not part of the gradient descent algorithm per se.

In this simulation, it is assumed again that an object rotates with the constant angular speed of 20 deg/s around an arbitrary chosen vector of rotation $n = [1 \quad -1 \quad 0]^T$. The initial quaternion and angle of rotation around the vector n are chosen to be $q_0 = [1 \quad 0 \quad 0 \quad 0]^T$ and $\alpha_0 = 45^\circ$, respectively. For each time step, the true rotational quaternion is calculated, which is then used to calculate the body frame gravitational acceleration and magnetic field. The angular velocity vector remains the same in both frames since it lies on the axis of rotation. Corresponding noise is added to each physical quantity and passed to the complementary filter implemented in MathWorks Matlab with the sample time of 0.1 s. The rotational quaternions estimated in each time step using all three initial quaternion propagation methods are converted in Euler angles and shown in Figure 6. Propagation methods 2 and 3 show excellent results with respect to the ground truth. Propagation

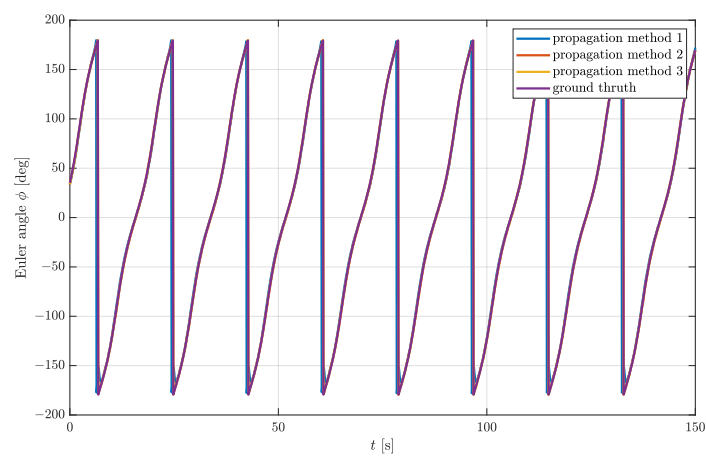
method 1, however, had difficulties estimating the rotational quaternion even with the maximum iteration number $N_{max} = 20$, which is especially manifested in the plot of Euler angle ψ in Figure 6a.



(a) Euler angle ψ .



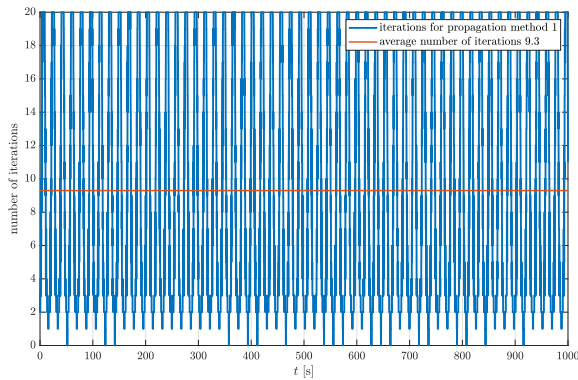
(b) Euler angle θ .



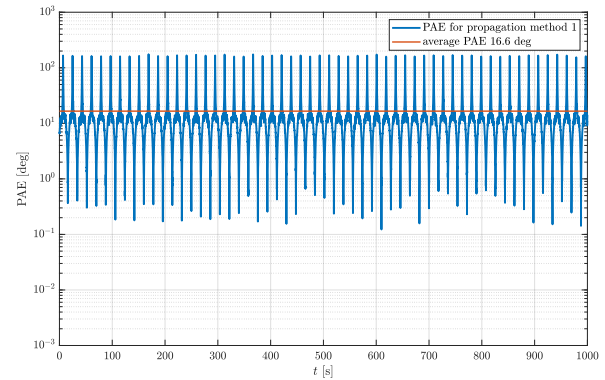
(c) Euler angle ϕ .

Figure 6. Comparison of estimated Euler angles using different propagation methods and complementary filter with parameters $\mu = 0.7$, $N_{max} = 20$, $G_{max} = 1 \times 10^{-2}$, and $K = 0.5$.

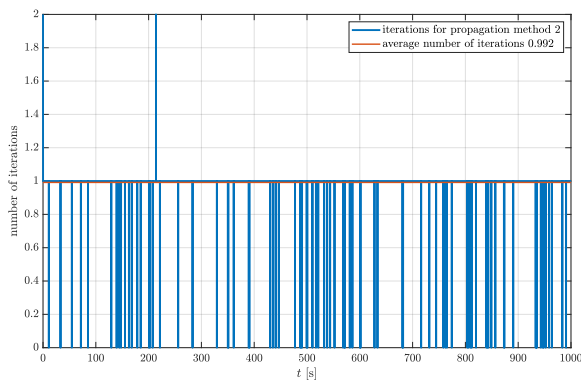
At some specific time steps, the initial quaternion is simply not good enough for the gradient descent to converge, which manifests in periodic spikes. On the other hand, Euler angles θ and ϕ are estimated reasonably well even with propagation method 1. The number of required iterations and PAE for the same simulation are shown in Figure 7.



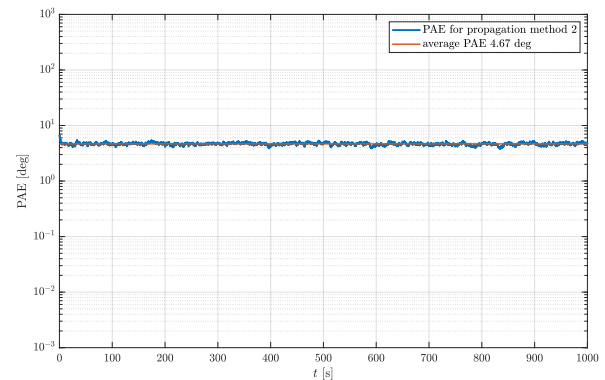
(a) Number of iterations required for propagation method 1.



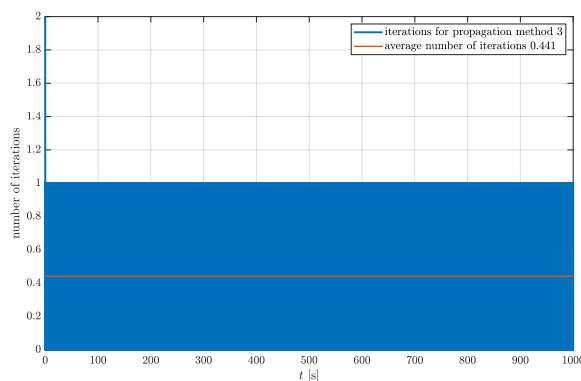
(b) PAE for propagation method 1.



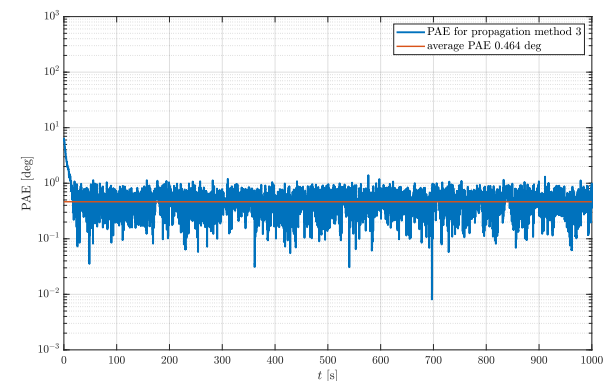
(c) Number of iterations required for propagation method 2.



(d) PAE for propagation method 2.



(e) Number of iterations required for propagation method 3.



(f) PAE for propagation method 3.

Figure 7. Comparison of number of iterations and PAE for different propagation methods and complementary filters with parameters $\mu = 0.7$, $N_{max} = 20$, $G_{max} = 1 \times 10^{-2}$, and $K = 0.5$.

Figure 7 clearly shows the difference in performance between the three propagation methods. It is clear that the iteration number is maxed out at exactly the same time steps at which the Euler angle ψ is incorrectly estimated using propagation method 1 (Figure 7a). This is evident even from the PAE in Figure 7b. This indicates that propagation method 1 is non-optimal, which is also confirmed by a relatively large average number of iterations of 9.3 and PAE of 16.6° . Propagation method 2 yields much more reasonable results with respect to the number of iterations and the PAE shown in Figure 7c,d, respectively. Notice

that the average number of iterations is very close but is still below 1. This means that the iterative process is not always needed for the rotational quaternion estimation. This can also be observed at the time steps at which the number of iterations is equal to zero. It is interesting that the noise added to the simulated data actually helps to bring the initial quaternion closer to the optimal estimate, which leads to the desired accuracy without applying a single iteration of a gradient descent algorithm. This effect is very similar to the process of intentionally applying noise known as dithering [35]. Furthermore, notice that the average PAE of 4.67° is drastically improved with respect to propagation method 1. As for propagation method 3, the results of which are shown in Figure 7e,f, it evidently yields the best results. The average PAE is over an order of magnitude lower with respect to propagation method 2, i.e., 0.464° . Moreover, notice that the noise floor of the PAE is reached. This confirms that G_{max} defining the accuracy of the estimate is correctly set. The improvement is clearly visible in the average number of iterations, which is dropped to 0.441. Thus, one can conclude that the attitude estimation algorithm mostly relies upon the angular velocity data, which for MPU9250 have the lowest noise level of the three. This noise level is decreased even further through the numerical integration characteristic for the estimation of the rotational quaternion from angular velocity. Only now and then is the estimate automatically corrected using the gradient descent algorithm if $G > G_{max}$.

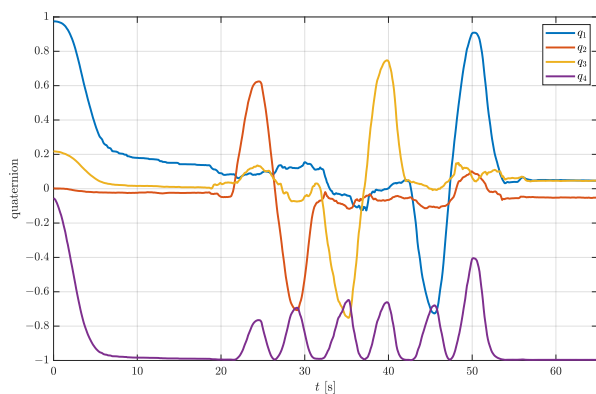
5. Experimental Results

The presented attitude estimation method is based on the gradient descent algorithm, complementary filter, and propagation method 3 and is experimentally tested using the model of the satellite attitude determination and control system (ADCS) shown in Figure 8. The model was developed for educational purposes at the University of Zagreb, Faculty of Electrical Engineering and Computing, Department of Communication and Space Technologies.

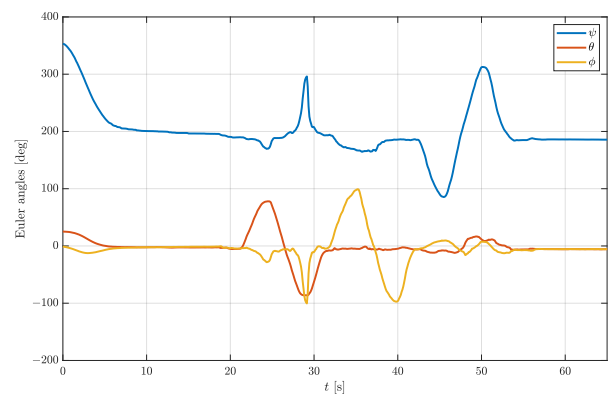


Figure 8. Photograph of the ADCS model used for experimental verification of the presented attitude estimation algorithm based on gradient descent and complementary filter.

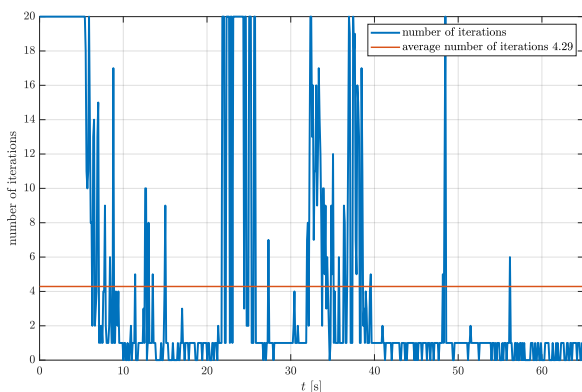
The ADCS model consists of a microcontroller, Bluetooth interface, reaction wheel (not used in this experiment), and the MPU9250 IMU with a 3-axis accelerometer, magnetometer, and gyroscope. The nine measurements were sampled with the sampling frequency of 10 Hz, wirelessly transferred to a computer and then used as the input arguments of the implemented attitude estimation algorithm. For the experimental purposes, the model is rotated for approximately $\pm 90^\circ$ around three orthogonal axes by hand. The initial quaternion for the first sample was chosen to be $q_0 = [1 \ 0 \ 0 \ 0]^T$. However, the ADCS model was initially rotated for approximately 180° around the z-axis, which represents the farthest point in quaternion space with respect to the initial quaternion and the worst case scenario. Please recall the results shown in Figure 2. Notice that the final value of the cost function shown in Figure 2d is indeed the highest compared to the other angles α for the same step size and number of iterations. The estimated quaternions and corresponding Euler angles are shown in Figure 9 together with the required number of iterations per sample and initial value of the cost function at each time step.



(a) Estimated rotational quaternions.



(b) Estimated Euler angles.



(c) Required number of iterations for each measured sample.

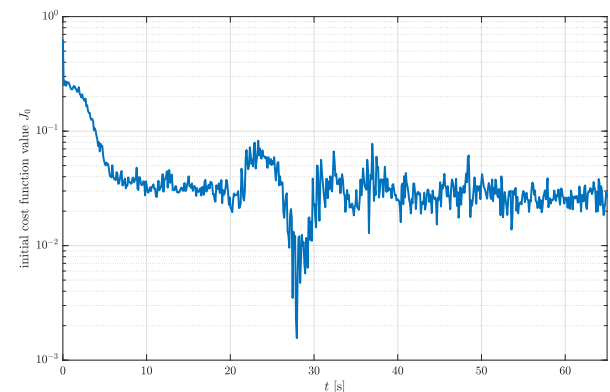
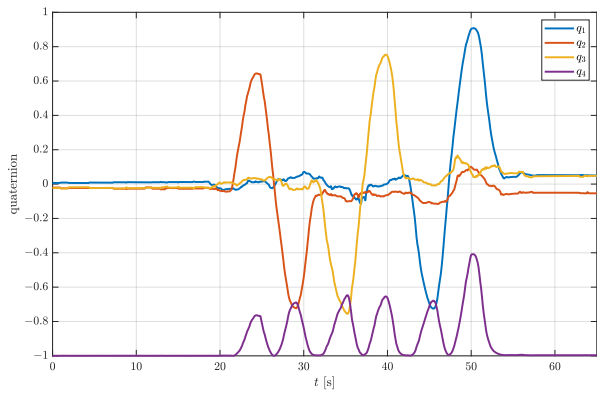
(d) Initial value of the cost function J_0 at each time step.

Figure 9. Experimental results obtained with parameters $\mu = 0.7$, $q_0 = [1 \ 0 \ 0 \ 0]^T$, $N_{max} = 20$, $G_{max} = 1 \times 10^{-2}$, and $K = 0.5$.

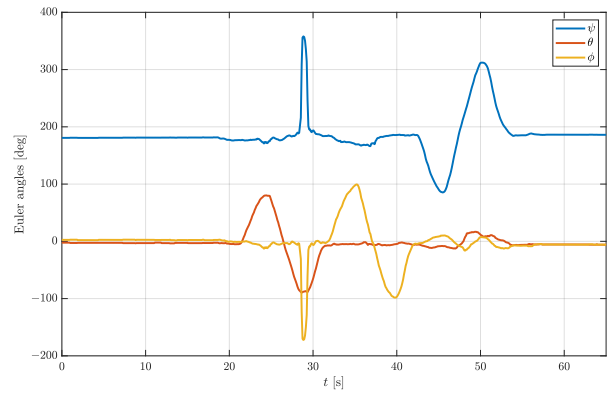
Figure 9 shows the perfect example of a real time scenario. In Figure 9b, the rotations for approximately $\pm 90^\circ$ around all three orthogonal axes, representing the Euler angles ψ , θ , and ϕ , are clearly visible. Notice that the average number of iterations is significantly higher than simulated (Figure 9c). The reason for that is the presence of other imperfections and influences on the IMU sensors. Recall that, for example, the accelerometers measure not only the gravitational acceleration. Rotating the model by itself may introduce additional acceleration and may interfere with the measurement of gravitational acceleration. Likewise, the magnetometers suffer from hard and soft iron effects caused by different

magnetic materials present in the vicinity of the sensor. Although gyroscopes show the best performances of the three in terms of noise, they are not immune to issues such as bias stability and angle random walk, which may significantly affect attitude estimation. All these effects are difficult to model and account for in simulations. Thus, the experimental validation of the chosen gradient descent parameter set is of the highest importance. It is fairly straightforward to show that these secondary effects cause a larger number of iterations by plotting the initial value of the cost function at each time step, as shown in Figure 9d. Since propagation method 3 is used to propagate the initial quaternion, it is expected that the initial value of the cost function at each time step monotonically decreases down to its minimum. For the first sample of the simulation, the initial value of the cost function is expected to be at its maximum due to the difference between the true and guessed initial quaternion. This is indeed the case. However, in the time interval of 20–40 s, a sudden increase in the initial cost function value can be observed. This increase is caused by the secondary effects, which manifest as an undesired signal superposed to the measurements. The consequence of the secondary effect is the lower accuracy of the estimated rotational quaternion. Since the accuracy of the attitude estimation is lowered, it is justified to slightly increase the value of the parameter G_{max} .

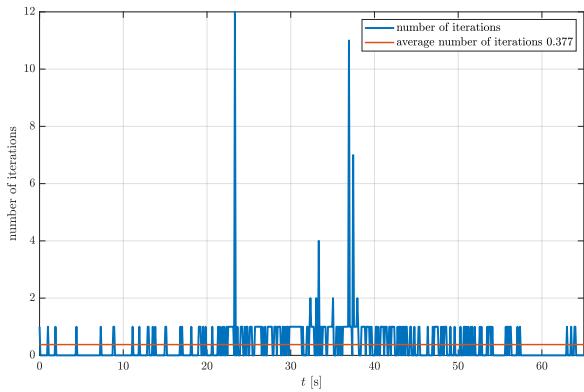
To prove that the initial value of the cost function is highest at the beginning of the measurement ($t \in [0, 10]$ s) due to the difference between the initial and true quaternion, the estimation is repeated with the initial quaternion set to $q_0 = [0 \ 0 \ 0 \ 1]^T$. This new initial quaternion represents the rotation for 180° around z-axis, which approximately describes the true initial attitude of the ADCS model. Likewise, the value of G_{max} was increased to 3×10^{-2} . The results are presented in Figure 10. Since the initial quaternion is very close to the true rotational quaternion, it is clear that the transient is fully eliminated, which manifests through constant Euler angles (Figure 10b), a low number of iterations (Figure 10c), and a constant initial value of the cost function (Figure 10d) for $t \in [0, 10]$ s. As expected, the sudden increase in the initial cost function value is still present in the time interval of 20–40 s. Nevertheless, the number of iterations in the same period is not maxed out due to the higher G_{max} . Furthermore, the average number of iterations is significantly lower (only 0.377) compared to the previous estimation (Figure 9c). This significantly unloads the targeted processing unit and lowers the processing power requirement. Most of the time not a single iteration of the gradient descent algorithm is needed. During these periods, the estimation of the rotational quaternion relies upon the gyroscope only. Sometimes, however, if the measurement is corrupted or the gyroscope bias accumulated, the value of the cost function increases, which automatically activates the gradient descent algorithm and corrects the estimate. As shown in Figure 9c, the number of gradient descent iterations may be significantly higher than one. This demonstrates the importance of the valid termination criterion. It also leads to the conclusion that it is not sufficient to limit the number of iterations to one. Increasing G_{max} from 1×10^{-2} to 3×10^{-2} does not substantially affect the quality of the attitude estimation, as shown in Figure 11. The graph shown in Figure 11 represents the principal angle deviation (PAD) between the estimated rotational quaternions for both parameter sets. It starts at higher values, caused by the transient of the first estimation, but eventually drops to very low values with a minimum below 0.2° .



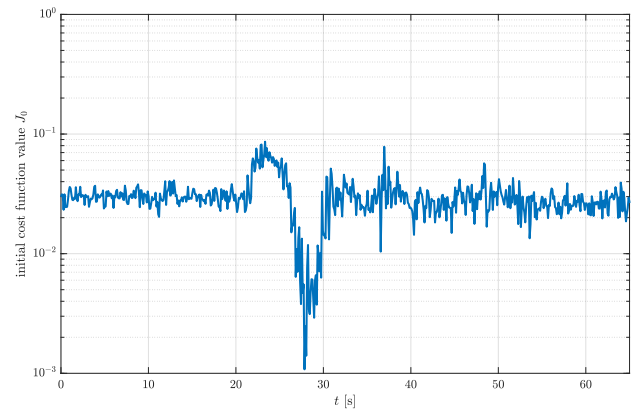
(a) Estimated rotational quaternions.



(b) Estimated Euler angles.



(c) Required number of iterations for each measured sample.



(d) Initial value of the cost function J_0 at each time step.

Figure 10. Experimental results obtained with parameters $\mu = 0.7$, $q_0 = [0 \ 0 \ 0 \ 1]^T$, $N_{max} = 20$, $G_{max} = 3 \times 10^{-2}$, and $K = 0.5$.

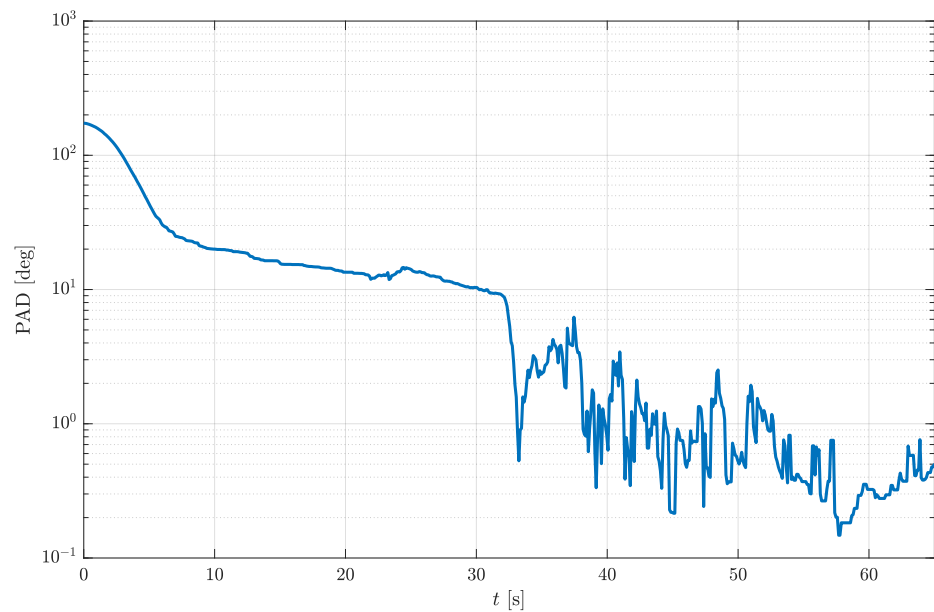


Figure 11. PAD between the estimated rotational quaternions for both sets of parameters.

6. Conclusions

In this paper, a comprehensive approach to the optimization of the gradient descent parameters with an application in the estimation of rotational quaternions is presented. The goal of the optimization was to reduce the computational complexity of the algorithm by minimizing the required number of iterations while maintaining the accuracy of the estimation. Most of the gradient descent attitude estimation methods limit the number of iterations to one. Here, however, we showed that such a low number of iterations may not be sufficient for achieving the desired accuracy under the influence of different sensor imperfections and external factors. Nevertheless, using the proposed approach, the average number of iterations may be much lower than one. The number of iterations is determined automatically based on the presented termination criterion. The optimized gradient descent algorithm was used as a part of complementary filter fusing the measurements acquired from low cost MPU9250 IMU sensors. The low average number of iterations and simplicity of implementation causes the proposed method to be ideal for low-power applications. The same approach to attitude estimation based on the gradient descent algorithm and its optimization are suitable and may be used as a part of more demanding attitude estimation algorithms employing Kalman filtering, which will be the subject of our future research efforts.

Author Contributions: Conceptualization, J.L.; methodology, J.L.; software, K.S., L.M.G. and J.L.; validation, J.L.; formal analysis, K.S. and J.L.; investigation, K.S., L.M.G. and J.L.; resources, J.L.; writing—original draft preparation, K.S., L.M.G. and J.L.; writing—review and editing, K.S., L.M.G. and J.L.; visualization, K.S. and J.L.; supervision, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhou, P.; Li, M.; Shen, G. Use it free: Instantly knowing your phone attitude. In Proceedings of the 20th Annual International Conference on Mobile Computing and Networking, Maui, HI, USA, 7–11 September 2014; pp. 605–616.
2. Luinge, H.J.; Veltink, P.H. Inclination measurement of human movement using a 3-D accelerometer with autocalibration. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2004**, *12*, 112–121. [[CrossRef](#)]
3. Aminian, K.; Najafi, B. Capturing human motion using body-fixed sensors: outdoor measurement and clinical applications. *Comput. Animat. Virtual Worlds* **2004**, *15*, 79–94. [[CrossRef](#)]
4. Vaganay, J.; Aldon, M.J.; Fournier, A. Mobile robot attitude estimation by fusion of inertial data. In Proceedings of the IEEE International Conference on Robotics and Automation, Atlanta, GA, USA, 2–6 May 1993; pp. 277–282.
5. Chatterji, G.B.; Menon, P.K.; Sridhar, B. Vision-based position and attitude determination for aircraft night landing. *J. Guid. Control. Dyn.* **1998**, *21*, 84–92. [[CrossRef](#)]
6. Bijker, J. Development of an Attitude Heading Reference System for an Airship. Ph.D. Thesis, University of Stellenbosch, Stellenbosch, South Africa, 2006.
7. Lefferts, E.J.; Markley, F.L.; Shuster, M.D. Kalman filtering for spacecraft attitude estimation. *J. Guid. Control. Dyn.* **1982**, *5*, 417–429. [[CrossRef](#)]
8. Cilden-Guler, D.; Raitoharju, M.; Piche, R.; Hajiyev, C. Nanosatellite attitude estimation using Kalman-type filters with non-Gaussian noise. *Aerosp. Sci. Technol.* **2019**, *92*, 66–76. [[CrossRef](#)]
9. Gonçalves, F.; Ribeiro, T.; Ribeiro, A.F.; Lopes, G.; Flores, P. A recursive algorithm for the forward kinematic analysis of robotic systems using euler angles. *Robotics* **2022**, *11*, 15. [[CrossRef](#)]
10. Chou, J.C.; Kamel, M. Finding the position and orientation of a sensor on a robot manipulator using quaternions. *Int. J. Robot. Res.* **1991**, *10*, 240–254. [[CrossRef](#)]
11. Marins, J.L.; Yun, X.; Bachmann, E.R.; McGhee, R.B.; Zyda, M.J. An extended Kalman filter for quaternion-based orientation estimation using MARG sensors. In Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems, Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180), Maui, HI, USA, 29 October–3 November 2001; Volume 4, pp. 2003–2011.
12. Hemingway, E.G.; O'Reilly, O.M. Perspectives on Euler angle singularities, gimbal lock, and the orthogonality of applied forces and applied moments. *Multibody Syst. Dyn.* **2018**, *44*, 31–56. [[CrossRef](#)]
13. Davenport, P.B. *A Vector Approach to the Algebra of Rotations with Applications*; National Aeronautics and Space Administration: Washington, DC, USA, 1968; Volume 4696.

14. Mortari, D. ESOQ: A closed-form solution to the Wahba problem. *J. Astronaut. Sci.* **1997**, *45*, 195–204. [[CrossRef](#)]
15. Shuster, M.D.; Oh, S.D. Three-axis attitude determination from vector observations. *J. Guid. Control.* **1981**, *4*, 70–77. [[CrossRef](#)]
16. Bar-Itzhack, I.Y. REQUEST-A recursive QUEST algorithm for sequential attitude determination. *J. Guid. Control. Dyn.* **1996**, *19*, 1034–1038. [[CrossRef](#)]
17. Choukroun, D.; Bar-Itzhack, I.Y.; Oshman, Y. Optimal-REQUEST algorithm for attitude determination. *J. Guid. Control. Dyn.* **2004**, *27*, 418–425. [[CrossRef](#)]
18. Sabatini, A.M. Kalman-filter-based orientation determination using inertial/magnetic sensors: Observability analysis and performance evaluation. *Sensors* **2011**, *11*, 9182–9206. [[CrossRef](#)]
19. Yuan, X.; Yu, S.; Zhang, S.; Wang, G.; Liu, S. Quaternion-based unscented Kalman filter for accurate indoor heading estimation using wearable multi-sensor system. *Sensors* **2015**, *15*, 10872–10890. [[CrossRef](#)]
20. Guo, H.; Hong, H. Research on filtering algorithm of MEMS gyroscope based on information fusion. *Sensors* **2019**, *19*, 3552. [[CrossRef](#)]
21. Farahan, S.B.; Machado, J.J.; de Almeida, F.G.; Tavares, J.M.R. 9-DOF IMU-Based Attitude and Heading Estimation Using an Extended Kalman Filter with Bias Consideration. *Sensors* **2022**, *22*, 3416. [[CrossRef](#)]
22. Cavallo, A.; Cirillo, A.; Cirillo, P.; De Maria, G.; Falco, P.; Natale, C.; Pirozzi, S. Experimental comparison of sensor fusion algorithms for attitude estimation. *IFAC Proc. Vol.* **2014**, *47*, 7585–7591. [[CrossRef](#)]
23. Quoc, D.D.; Sun, J. Attitude estimation algorithms using low cost IMU. *Int. J. Control. Autom.* **2015**, *8*, 113–126. [[CrossRef](#)]
24. Liu, M.; Cai, Y.; Zhang, L.; Wang, Y. Attitude Estimation Algorithm of Portable Mobile Robot Based on Complementary Filter. *Micromachines* **2021**, *12*, 1373. [[CrossRef](#)]
25. Marantos, P.; Koveos, Y.; Kyriakopoulos, K.J. UAV state estimation using adaptive complementary filters. *IEEE Trans. Control. Syst. Technol.* **2015**, *24*, 1214–1226. [[CrossRef](#)]
26. Sever, K.; Indir, I.; Vnučec, I.; Lončar, J. Evaluation of Gradient Descent Algorithm for Attitude Estimation. In Proceedings of the 2021 International Symposium ELMAR, Zagreb, Croatia, 13–15 September 2021; pp. 131–134.
27. Cheguini, M.; Ruiz, F. Real-time attitude estimation based on Gradient Descent algorithm. In Proceedings of the 2012 IEEE 4th Colombian Workshop on Circuits and Systems (CWCAS), Barranquilla, Colombia, 1–2 November 2012; pp. 1–6.
28. Wu, J. MARG attitude estimation using gradient-descent linear Kalman filter. *IEEE Trans. Autom. Sci. Eng.* **2020**, *17*, 1777–1790. [[CrossRef](#)]
29. Admiraal, M.; Wilson, S.; Vaidyanathan, R. Improved formulation of the IMU and MARG orientation gradient descent algorithm for motion tracking in human–machine interfaces. In Proceedings of the 2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Daegu, Republic of Korea, 16–18 November 2017; pp. 403–410.
30. Madgwick, S.O.; Harrison, A.J.; Vaidyanathan, R. Estimation of IMU and MARG orientation using a gradient descent algorithm. In Proceedings of the 2011 IEEE International Conference on Rehabilitation Robotics, Zurich, Switzerland, 29 June–1 July 2011; pp. 1–7.
31. Sola, J. Quaternion kinematics for the error-state Kalman filter. *arXiv* **2017**, arXiv:1711.02508.
32. Wu, J.; Zhou, Z.; Song, M.; Fourati, H.; Liu, M. Convexity analysis of optimization framework of attitude determination from vector observations. In Proceedings of the 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT), Paris, France, 23–26 April 2019; pp. 440–445.
33. National Centers for Environmental Information. Geomagnetic Models and Software. 2022. Available online: <https://www.ngdc.noaa.gov/geomag> (accessed on 28 November 2022).
34. Liu, N.; Qi, W.; Su, Z.; Feng, Q.; Yuan, C. Research on Gradient-Descent Extended Kalman Attitude Estimation Method for Low-Cost MARG. *Micromachines* **2022**, *13*, 1283. [[CrossRef](#)]
35. Ulichney, R. *Digital Halftoning*; MIT Press: Cambridge, MA, USA, 1987.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.