

Article

A Magnetometer-Only Attitude Determination Strategy for Small Satellites: Design of the Algorithm and Hardware-in-the-Loop Testing

Stefano Carletta ^{*,†}, Paolo Teofilatto [†] and M. Salim Farissi [†]

School of Aerospace Engineering, Sapienza University of Rome, 00138 Rome, Italy;
paolo.teofilatto@uniroma1.it (P.T.); salim.farys@gmail.com (M.S.F.)

* Correspondence: stefano.carletta@uniroma1.it

† These authors contributed equally to this work.

Received: 30 September 2019; Accepted: 2 January 2020; Published: 5 January 2020



Abstract: Attitude determination represents a fundamental task for spacecraft. Achieving this task on small satellites, and nanosatellites in particular, is further challenging, because the limited power and computational resources available on-board, together with the low development budget, set strict constraints on the selection of the sensors and the complexity of the algorithms. Attitude determination is obtained here from the only measurements of a three-axis magnetometer and a model of the Geomagnetic field, stored on the on-board computer. First, the angular rates are estimated and processed using a second-order low-pass Butterworth filter, then they are used as an input, along with Geomagnetic field data, to estimate the attitude matrix using an unsymmetrical TRIAD. The computational efficiency is enhanced by arranging complex matrix operations into a form of the Faddeev algorithm, which is implemented using systolic array architecture on the FPGA core of a CubeSat on-board computer. The performance and the robustness of the algorithm are evaluated by means of numerical analyses in MATLAB Simulink, showing pointing and angular rate accuracy below 10° and $0.2^\circ/\text{s}$. The algorithm implemented on FPGA is verified by Hardware-in-the-loop simulation, confirming the results from numerical analyses and efficiency.

Keywords: magnetometer-only; attitude determination; CubeSat; Faddeev algorithm; systolic array; HiL

1. Introduction

Mission operations often require the capability to orient the spacecraft in a specific direction. To perform these maneuvers, an adequately accurate knowledge of the spacecraft orientation in space, namely its attitude, is required, and this is the goal of attitude determination. The results from this process are input to attitude control, which produces, through the actuators, an adequate control torque driving the spacecraft to the desired attitude.

The problem of attitude determination has been extensively studied since the outset of space exploration, and consists of the determination of the rotation matrix A , which transforms any vector v_b , whose coordinates are expressed in a reference frame \mathcal{F}_b rotating rigidly with the spacecraft, to the corresponding vector v_i , expressed in a reference coordinate system \mathcal{F}_i [1]. Typically, the problem is solved using the Tri-axial Attitude Determination (TRIAD) algorithm which requires the knowledge of at least two couples of vectors in both \mathcal{F}_b and \mathcal{F}_i [2]. Similar methods, representing solutions to Wahba's problem [3], have been widely applied and include the q-method by Davenport [1,4], Schuster's QUaternion ESTimator (QUEST) algorithm [5,6] and the optimized TRIAD by Bar-Itzhack and Harman [7]. Other approaches produce attitude determination by direct quaternion estimation [8,9],

and can be significantly faster than the other algorithms outlined but can show singularities, which occur when the attitude rotation axis lays on the plane of the unit vectors on which the estimation is based, as detail by Markley [10].

The above-mentioned algorithms require two vector measurements in two-body frames \mathcal{F}_b and \mathcal{F}_i . In fact, the measurements in \mathcal{F}_i can be replaced by predictions of the vector behavior in time and space; for instance, the International Geomagnetic Reference Field (IGRF) model can be used to estimate the values of the Geomagnetic field vector at any orbital coordinates of the spacecraft [11]. These models can be rather complex and accurate, and their selection is a crucial part of the design process. A similar approach is not effective for vectors in \mathcal{F}_b because their prediction requires knowledge of the spacecraft attitude, which is instead the unknown of the attitude determination problem. As a result, on-board sensors must be used for vector measurements in \mathcal{F}_b , and their characteristics strongly determine the accuracy of the attitude determination algorithm. The use of star trackers can produce attitude determination with accuracy less than 0.01° . Nevertheless, these components are relatively expensive, power and memory consuming if compared to MEMS (Micro-Electro-Mechanical Systems) sensors, such as magnetometers, gyroscopes, and sun sensors, which, therefore, represent an attracting alternative for low-cost missions. MEMS gyroscopes are accurate angular rate sensors; nevertheless, because of technological limitations, they usually have structure defects which produce high drift and require compensation, thus opening to alternative methods of angular rate determination [12].

Gyroless attitude determination methods are of interest as backup solutions for small satellite missions that do not require high pointing accuracy for basic operations [13]. Such a solution can be effective when the above-mentioned measurement drift increases above a threshold level, in case of failure of the sensor and in the presence of an error in the algorithm which produces the interpretation of the signal. Several methods have been proposed based on the use of a three-axis magnetometer together either with other low-cost sensors or solar panels [14–17] or on their own.

Magnetometer-only attitude determination is a challenging task, mainly because the measurements from a three-axis magnetometer can provide information on only two axes of the spacecraft attitude. To resolve all three axes, either some constraints on the attitude motion or a filtering process is required. A first solution was proposed by Natanson et al. [18], whose DADMOD algorithm provided good results in the post-processing. A further implementation of the algorithm using DADMOD to initialize a Real-Time Sequential Filter (RTSF) extended it to real-time applications [19,20], with an accuracy of 2° on attitude and $0.01^\circ/\text{s}$ on angular rates.

The most popular methods for three-axis magnetometer-only attitude determination are based on the Extended Kalman Filter (EKF) and can reach an accuracy below 5° on attitude and $0.01^\circ/\text{s}$ on the angular rates [21–23]. Other solutions include the Unscented Kalman Filter (UKF) algorithm by Ma and Jiang [24], or the two-step EKF algorithm proposed by Searcy and Pernicka [25], which achieves accuracies of less than 1° , but is effective only if the angular rates along at least one axis exceeds $0.1^\circ/\text{s}$.

Despite their rather high accuracy, the mentioned algorithms either require sophisticated models for vector estimation, are effective only under some specific attitude control, or include complex vector and matrix operations. The latter can be rather complex to be implemented as the Field Programmable Gate Array (FPGA) core of an On-Board Computer (OBC). The use of FPGA-based OBC in small satellites, especially CubeSats, has grown in the last years. This trend is motivated by FPGA hardware and software flexibility, which allows on-ground and in-orbit risk-free reconfiguration, and by their capability to perform processes in parallel.

In this work, we propose a simple attitude determination strategy suitable to be implemented on an FPGA-based OBC. The solution does not require any Kalman Filter and is based only on the measurements of a three-axis magnetometer and the Geomagnetic field data stored on the OBC memory. Angular rates are determined independently from attitude, based on a method proposed previously by the same authors [26]. The Attitude matrix is then estimated by developing a form of the unsymmetrical TRIAD. All the complex matrix operations are arranged in the form of a Faddeev algorithm [27], which can be implemented on FPGA using one single compact systolic array architecture [28]. The use of the

same architecture for all the complex matrix operations leads to an increase in the system and algorithm efficiency. The performance and robustness of the attitude determination strategy are verified by means of numerical analyses in MATLAB Simulink, showing an accuracy of $0.2^\circ/\text{s}$ for angular rates and below 10° for the attitude. The results show how process delay, characterizing the algorithm, can be minimal when implemented on real hardware, as verified by Hardware-in-the-Loop (HiL) simulation.

The manuscript is organized as follows. In Section 2, the dynamical framework and the models used for the analysis are introduced. The design and implementation of the attitude determination algorithm is provided in Section 3, and its performance and robustness are verified by numerical analyses in Section 4. The analyses are performed simulating the detumbling and pointing of the satellite after deployment, assuming unknown initial conditions randomly selected. In Section 5 the algorithm is implemented on the FPGA core of a CubeSat OBC, and tested by HiL simulation as a preliminary verification of its suitability for implementation for real missions.

2. Dynamical Framework

The attitude determination strategy presented in this work is targeted to spacecraft orbiting in circular Low Earth Orbits (LEO) and does not require any a-priori knowledge of the Right Ascension of the Ascending Node (RAAN) Ω , orbit phase angle ϕ_0 , attitude and angular rates.

The satellite is modeled as a rigid body, and its attitude is described using Euler angles (φ, θ, ψ) and angular rates ($\omega = [\omega_x \ \omega_y \ \omega_z]^T$), representing the angular velocity components of the reference frame $\mathcal{F}_b = [\hat{x}_b \ \hat{y}_b \ \hat{z}_b]^T$, attached to the satellite, with respect to the inertial reference frame $\mathcal{F}_i = [\hat{x}_i \ \hat{y}_i \ \hat{z}_i]^T$, here selected as the Geo-Centric Inertial (GCI) frame. Assuming \mathcal{F}_b to be coincident with the principal axes of inertia of the satellite, then attitude dynamics can be expressed by the following set of equations [1]

$$\left\{ \begin{array}{l} \dot{\varphi} = \omega_x + \sin \varphi \tan \theta \omega_y + \cos \varphi \tan \theta \omega_z \\ \dot{\theta} = \cos \varphi \omega_y - \sin \varphi \omega_z \\ \dot{\psi} = \frac{(\sin \varphi \omega_y - \cos \varphi \omega_z)}{\cos \theta} \\ \dot{\omega}_x = \left(\frac{I_y - I_z}{I_x} \right) \omega_y \omega_z + \tau_x \\ \dot{\omega}_y = \left(\frac{I_z - I_x}{I_y} \right) \omega_x \omega_z + \tau_y \\ \dot{\omega}_z = \left(\frac{I_x - I_y}{I_z} \right) \omega_x \omega_y + \tau_z \end{array} \right. , \quad (1)$$

where I_i and τ_i are, respectively, the moment of inertia and the external torque in the i -th direction of \mathcal{F}_b . It can be noticed that the third equation in system (1) shows a singularity for $\theta = \frac{\pi}{2} + k\pi$ $k \in \mathbb{Z}$, and

this can be avoided using quaternions instead of Euler angles to represent the attitude [1]. The attitude dynamics using the quaternion representation can be expressed by the following system:

$$\begin{cases} \dot{q}_1 = \frac{1}{2}(q_4\omega_x + q_2\omega_z - q_3\omega_y) \\ \dot{q}_2 = \frac{1}{2}(q_4\omega_y - q_1\omega_z + q_3\omega_x) \\ \dot{q}_3 = \frac{1}{2}(q_4\omega_z + q_1\omega_y - q_2\omega_x) \\ \dot{q}_4 = -\frac{1}{2}(q_1\omega_x + q_2\omega_y + q_3\omega_z) \\ \dot{\omega}_x = \left(\frac{I_y - I_z}{I_x}\right)\omega_y\omega_z + \tau_x \\ \dot{\omega}_y = \left(\frac{I_z - I_x}{I_y}\right)\omega_x\omega_z + \tau_y \\ \dot{\omega}_z = \left(\frac{I_x - I_y}{I_z}\right)\omega_x\omega_y + \tau_z \end{cases}, \quad (2)$$

For the case examined here, the control torque is produced by three magnetorquers each one orthogonal to an axis of \mathcal{F}_b . Indicating with \mathbf{m} the magnetic dipole moment produced by the three magnetorquers, the control torque is given by:

$$\boldsymbol{\tau} = \mathbf{m} \times \mathbf{B}_b, \quad (3)$$

where $\mathbf{B}_b = [B_{b,x} \ B_{b,y} \ B_{b,z}]^T$ is the Geomagnetic field vector in \mathcal{F}_b .

Considering the worst-case scenario of a spacecraft deployed with unknown attitude and angular rates, equipped with only a three-axis magnetometer as an attitude sensor, then the only control actions that can be performed are stabilization and pointing with respect to the Geomagnetic field. A case of interest is that in which the satellite is stabilized to pure-spin around the i -th axis, with a desired angular rate $\bar{\omega}_i$ and, contemporarily, the i -th axis is redirected towards the Geomagnetic field vector. A suitable control law, leading to the mentioned result for \hat{x}_b , is given by the following equations:

$$\mathbf{m} = \mathbf{m}_d + \mathbf{m}_p \quad (4a)$$

$$\begin{cases} m_{d,x} = k_1 \dot{\beta} \\ m_{d,y} = -k_2(\omega_x - \bar{\omega}_x) \operatorname{sign}(B_{b,z}) & \text{if } |B_{b,z}| \geq |B_{b,y}| \\ 0 & \text{if } |B_{b,z}| < |B_{b,y}| \\ m_{d,z} = k_2(\omega_x - \bar{\omega}_x) \operatorname{sign}(B_{b,y}) & \text{if } |B_{b,z}| < |B_{b,x}| \\ 0 & \text{if } |B_{b,z}| \geq |B_{b,x}| \end{cases} \quad (4b)$$

$$\mathbf{m}_p = k_p \mathbf{B}_b \times (\hat{x}_b \times \hat{B}_b) \quad (4c)$$

where the hat “ $\hat{}$ ” indicates the direction of the vector \mathbf{a} , $\dot{\beta} = \cos(\hat{B}_{b,x})$, k_p is the gain for the pointing control, and k_i are the gains for the stabilization law. It can be noticed that Equation (4a) is similar to the nadir-pointing control in [29], in which the nadir direction in Equation (4c) is replaced by \hat{B}_b and the damping term, originally given by the B-dot, has been changed to Equation (4b). Equation (4b) corresponds to a rearrangement of the Y-Thompson spin, where the spin axis is \hat{x}_b instead of \hat{y}_b [30]. In this arrangement, a magnetic torque along \hat{x}_b produces damping, which is inversely proportional to the angular displacement between \hat{x}_b and the magnetic field vector, in virtue of $\beta = \cos(\hat{B}_{b,x})$. Differently, the magnetic torques along \hat{y}_b and \hat{z}_b decrease as ω_x reaches the desired value $\bar{\omega}_i$ allowing non-zero spin stabilization. It is worth noting that, once solved the attitude determination problem, the same control law Equations (4a)–(4c) can be used to point towards the desired direction \hat{r} , by simply replacing \hat{B}_b with \hat{r} in Equation (4c). The selection of the control gain represents a rather complex issue for both the control laws Equations (4b) and (4c). In fact, even though higher gains correspond to higher torques, these are typically limited by the threshold value of the magnetic dipole moments,

which can be generated by the attitude control system. Some criteria for an optimal selection of these gains, in the presence of different constraints, have been developed [31–33].

In Sections 4 and 5, numerical and HiL simulations are performed to verify the performance and robustness of the attitude determination algorithm.

3. Attitude Determination Algorithm

The design and implementation of the proposed magnetometer-only attitude determination algorithm are hereafter discussed. The spacecraft attitude and angular rates are determined in two different steps, thus limiting the size of the vectors and matrixes computed through the process to 3×1 and 3×3 . All those vector and matrix operations that can be rearranged into the following form, a matrix equation of size 3:

$$U = WX^{-1}Y + Z, \quad (5)$$

can be implemented on FPGA by means of the Faddeev algorithm [34], using a single systolic array architecture. It means that the FPGA area dedicated to the systolic array can be used to calculate the results for any vector or matrix operation expressed in the form of Equation (5), taking advantage of parallel computing and reducing the usage on FPGA, thus increasing the efficiency of the algorithm [35–37]. This concept is now applied to implement the attitude determination algorithm.

The three-axis magnetometer operates at a fixed sampling frequency f_k , producing the measurement $B_b^{(k)}$. The time derivatives of B_b can be approximated as $\dot{B}_b^{(k)} = f_k(B_b^{(k)} - B_b^{(k-1)})$ and, as proved in a previous work by the authors [26], the angular rates can be estimated by processing with a second-order low-pass filter the results from the following equation:

$$\omega^{(k)} \approx f_k \frac{\dot{B}_b^{(k)} \times \dot{B}_b^{(k-1)}}{\|\dot{B}_b^{(k)}\|^2}, \quad (6)$$

It is worth highlighting that Equation (6) requires three consecutive samplings $B_b^{(k)}$; therefore, it produces an estimation of the angular rates with a delay equal to three times the sampling time ($3/f_k$). The effect of this delay will be examined in Sections 4 and 5.

Equation (6) can be rearranged in a form suitable for systolic array implementation, which consists of the determination of the values of matrixes W , X , Y , and Z . A solution is given by the following set:

$$U = \omega \quad (7a)$$

$$W = (\dot{B}_b^{(k)} \times \dot{B}_b^{(k-1)}) \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \quad (7b)$$

$$X = \dot{B}_b^{(k)T} \dot{B}_b^{(k)} I \quad (7c)$$

$$Y = I \quad (7d)$$

$$Z = 0 \quad (7e)$$

where I and 0 indicate the 3×3 identities and the 3×3 zero matrixes. If the mass distribution of the spacecraft is not spherical, such as in CubeSats, Equation (6) is affected by an error, that resembles a high-frequency noise [26], associated with the nonlinear terms $-\omega \times I\omega$. This can be removed using a second-order Butterworth low-pass filter [35,36], characterized by the following transfer function:

$$H(s) = \frac{K_0}{\left(\frac{s}{f_{co}}\right)^2 + 1.4142\left(\frac{s}{f_{co}}\right) + 1} \quad (8)$$

where K_0 is the filter gain and f_{co} is the cut-off frequency.

The estimated values of the angular rates and a model of the Geomagnetic field are now used to implement the unsymmetrical TRIAD and estimate the attitude matrix A . As discussed in Section 1, the TRIAD requires knowledge of at least two vectors in \mathcal{F}_b and the corresponding values in \mathcal{F}_i . Each vector v_b can be converted to the corresponding v_i by applying the transformation $v_i = Av_b$, where the expression for the rotation matrix A is given below:

$$A = \begin{bmatrix} \cos \psi \cos \theta & \cos \varphi \sin \psi + \cos \psi \sin \varphi \sin \theta & \sin \varphi \sin \psi - \cos \varphi \cos \psi \sin \theta \\ -\cos \theta \sin \psi & \cos \varphi \cos \psi - \sin \varphi \sin \psi \sin \theta & \cos \psi \sin \varphi + \cos \varphi \sin \psi \sin \theta \\ \sin \theta & -\cos \theta \sin \varphi & \cos \varphi \cos \theta \end{bmatrix}, \quad (9)$$

According to Equation (9), A depends on the Euler angles and; therefore, it changes in time with the attitude of the spacecraft. The value of A is here estimated based on the values of $B_b^{(k)}$ which are provided at fixed sampling rates. The value of the attitude matrix is estimated corresponding to each sample k and indicated as $A^{(k)}$. The matrix is assumed constant in the interval between two consecutive samplings.

Based on these considerations, the Geomagnetic field vector and its derivative in the two reference frames can be related by the following equations:

$$B_i^{(k)} = A^{(k)} B_b^{(k)} \quad (10a)$$

$$\dot{B}_i^{(k)} = A^{(k)} \left(\dot{B}_b^{(k)} + \omega^{(k)} \times B_b^{(k)} \right) \quad (10b)$$

If the inclination of the orbital plane i and the altitude h of the spacecraft are known, the IGRF model can be used to calculate the Geomagnetic field data to be stored on the OBC [11]. These data are selected and arranged into tables whose first entry is the true anomaly (ϑ) of the satellite (i.e., $[\vartheta^{(k)} B_{i,x}^{(k)} B_{i,y}^{(k)} B_{i,z}^{(k)}]$). When the satellite passes over the ground station, the value of ϑ at the passage is sent to the satellite and stored by the FPGA, which then propagates it in time. The value of ϑ triggers the reading of the table, producing the vector $B_i^{(k)}$. The derivative in time of $B_i^{(k)}$ can then be computed as $\dot{B}_i^{(k)} = f_k(B_i^{(k)} - B_i^{(k-1)})$.

Equation (6) could be used to solve the determination problem; nevertheless, a more accurate expression was developed based on the following result in [26]:

$$\omega_{\perp} = \frac{\dot{B}_b \times B_b}{\|B_b\|^2} \quad (11)$$

The vector product $\omega \times B_b$ in Equation (10b) can be rearranged as follows:

$$\omega \times B_b = \|\omega\| \|B_b\| \sin \alpha \hat{n}$$

where α is the angle between the vectors and \hat{n} is the direction orthogonal to the plane, which they define. Noting that $\|\omega\| \sin \alpha$ is the projection of ω orthogonal to B_b , then $\omega_{\perp} = \|\omega\| \sin \alpha$ and considering Equation (11) it follows that:

$$\omega \times B_b = \|\omega\| \|B_b\| \sin \alpha \hat{n} = \omega_{\perp} \times B_b = \frac{\dot{B}_b \times B_b}{\|B_b\|^2} \times B_b \quad (12)$$

Using the triple vector product properties $\frac{\dot{B}_b \times B_b}{\|B_b\|^2} \times B_b = \frac{[-(B_b^T B_b) \dot{B}_b + (B_b^T \dot{B}_b) B_b]}{\|B_b\|^2} = -\dot{B}_b + \frac{(B_b^T \dot{B}_b) B_b}{\|B_b\|^2}$, then Equation (10b) can be written as:

$$\begin{aligned}\dot{B}_i^{(k)} &= A^{(k)} \left(\dot{B}_b^{(k)} + \omega^{(k)} \times B_b^{(k)} \right) = A^{(k)} \left(\dot{B}_b^{(k)} - \dot{B}_b^{(k)} + \frac{B_b^{(k)} \dot{B}_b^{(k)T} B_b^{(k)}}{\|B_b^{(k)}\|^2} \right) = \\ &= A^{(k)} \left(\hat{B}_b^{(k)} \dot{B}_b^{(k)T} \hat{B}_b^{(k)} \right)\end{aligned}\quad (13)$$

For the sake of clearness, the following compact notation is introduced:

$$\left\{ \begin{array}{l} \hat{b}_1 = \hat{B}_b^{(k)} \\ \hat{b}_2 = \hat{B}_b^{(k)} \hat{B}_b^{(k)T} \hat{B}_b^{(k)} \\ \hat{b}_3 = \hat{b}_1 \times \hat{b}_2 \\ \hat{b}_4 = \hat{b}_1 \times \hat{b}_3 \end{array} \right. \quad \left\{ \begin{array}{l} \hat{r}_1 = \hat{B}_i^{(k)} \\ \hat{r}_2 = \hat{B}_i^{(k)} \\ \hat{r}_3 = \hat{r}_1 \times \hat{r}_2 \\ \hat{r}_4 = \hat{r}_1 \times \hat{r}_3 \end{array} \right. \quad (14)$$

Based on Equation (14) the attitude matrix at time k can be estimated as follows:

$$A^{(k)} = [b_1 \cdot b_3 \cdot b_4] [r_1 \cdot r_3 \cdot r_4]^T \quad (15)$$

Equation (15) is equivalent to the unsymmetrical form of the TRIAD algorithm [10], in which the relative weight of the measurement b_1 is higher than that of b_2 in the estimation of $A^{(k)}$. This form was preferred because the accuracy on $B_b^{(k)}$, measured by on-board sensors, is reasonably higher than that of ω_\perp , estimated during the process. It can be noticed that the matrix $[r_1 \cdot r_3 \cdot r_4]$ has a unit norm, and its transpose corresponds to the inverse. Therefore, Equation (15) can be set in the form Equation (6) of the Faddeev algorithm and solved through the systolic array architecture:

$$U = A^k \quad (16a)$$

$$W = [b_1 \cdot b_3 \cdot b_4] \quad (16b)$$

$$X = [r_1 \cdot r_3 \cdot r_4] \quad (16c)$$

$$Y = I \quad (16d)$$

$$Z = 0 \quad (16e)$$

4. Performance and Robustness Analysis

A preliminary analysis of the attitude determination algorithm is performed by numerical simulation on two Test Cases (TC), TC1 and TC2, which differ by the angular rates at the deployment. The simulations are performed considering the properties of the 3U CubeSat Tigrisat, launched by the School of Aerospace Engineering in 2014, with volume $340 \times 100 \times 100 \text{ mm}^3$ and the total mass $m = 4 \text{ kg}$. The dynamic attitude equations are integrated using Matlab Simulink *ode8* fixed-step solver, considering a time step of 1 s and for a total time corresponding to three orbital periods. For each step of the integration, the Geomagnetic field vector B_i is calculated using the IGRF model and converted to B_b as $B_b = A^{-1} B_i$ [11].

The inertial properties of the satellite, the orbital elements and the filter and control parameters common to the two test cases are listed in Table 1, where the total time of the simulation corresponds to three orbital periods. The angular rates at the deployment for the two test cases are shown in Table 2.

Table 1. Parameters for the simulations of Test Case 1 and Test Case2.

Moments of Inertia		Orbital Parameters	
I_x (kgm ²)	0.0065	h (km)	600
I_y (kgm ²)	0.0409	i (°)	87
I_z (kgm ²)	0.0409	Ω (°)	0
Coil number of Turns		Coil Area	
N_x	320	A_x (m ²)	0.0032
N_y	220	A_y (m ²)	0.0149
N_z	220	A_z (m ²)	0.0149
Control Gains		Filter Cut-off Frequencies	
k_1	1.8	f_{cox} (Hz)	0.0218
k_2	1	f_{coy} (Hz)	0.0017
k_p	500	f_{coz} (Hz)	0.0017
Filter Gains		Euler Angles at Deployment	
$K_{0,x}$	1	φ	0
$K_{0,y}$	1	θ	0
$K_{0,z}$	1	ψ	0
Simulation Parameters			
f_k (Hz)	1		
Solver	ODE8		
Total time (s)	17386		
$\bar{\omega}_x$ (°/s)	2.5		

Table 2. Angular rates at deployment for Test Case 1 and Test Case 2.

TC1	TC2		
ω_x (°/s)	11	ω_x (°/s)	5
ω_x (°/s)	11	ω_x (°/s)	3
ω_x (°/s)	10	ω_x (°/s)	-3

First, TC1 is examined. To investigate the accuracy of the attitude determination algorithm, the Mean Squared Error (MSE) between the actual A and the estimated A^k is calculated. The actual A is calculated based on the Euler angles calculated during the numerical integration by means of Equation (9). Figure 1 shows that the MSE assumes a final value of 1.172×10^{-2} and enters the $\pm 10\%$ error band (with respect to the final value) in 16680 s. The figure also reports the MSE calculated for A considering a delay of 5 (dashed line) and 6 (dotted line) seconds. These are calculated comparing, for each sample, the actual A with the same matrix at, respectively, the sample k-5 and k-6. The MSE for the 5 s and 6 s “delayed” A reaches the final value of, respectively, 1.065×10^{-2} and 1.533×10^{-2} , 10% lower and 23% higher than the MSE for TC1. This result indicates that the process suffers a delay, which is in the range 5–6 s, thus five to six times higher than the simulation time step. Such a process delay arises because both the estimation processes require the same systolic array architecture, and the attitude matrix is calculated after the angular rates, which require three consecutive samples ($3/f_k = 3$ s), and filtering to produce the result. The delay is considerably reduced when the algorithm is implemented on real hardware with higher operating and sampling frequency.

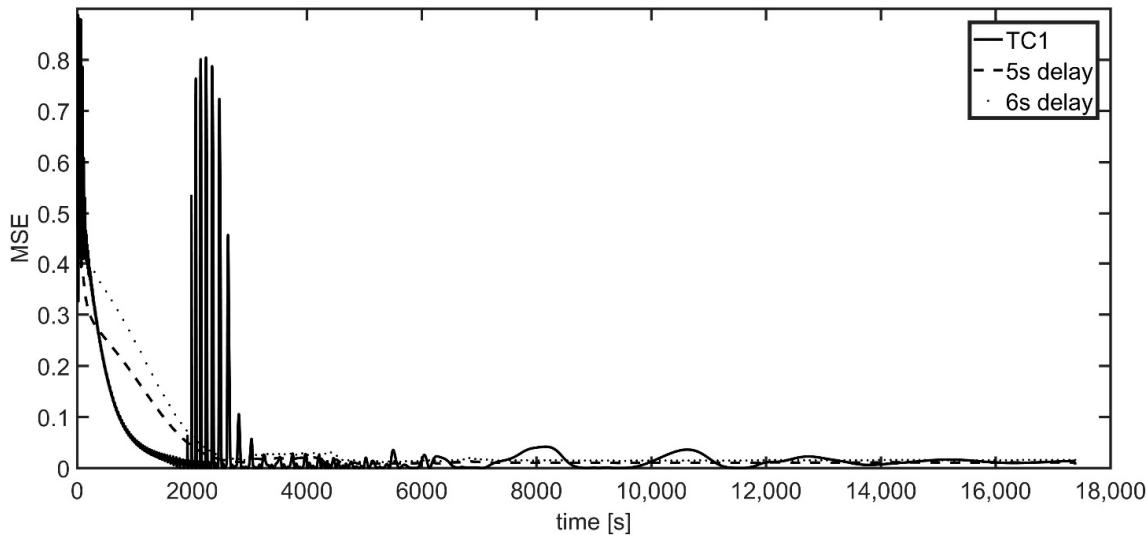


Figure 1. Mean Square Error calculated on the estimated A^k and on the actual A with 5 s and 6 s delay for Test Case 1.

For the sake of clarity, the accuracy of the attitude determination is also evaluated by comparing the “actual” Euler angles, resulting from the ODE8 solver to the estimated ones, computed from A^k . The difference between the estimated and the actual (from the integration) Euler angles is shown in Figure 2.

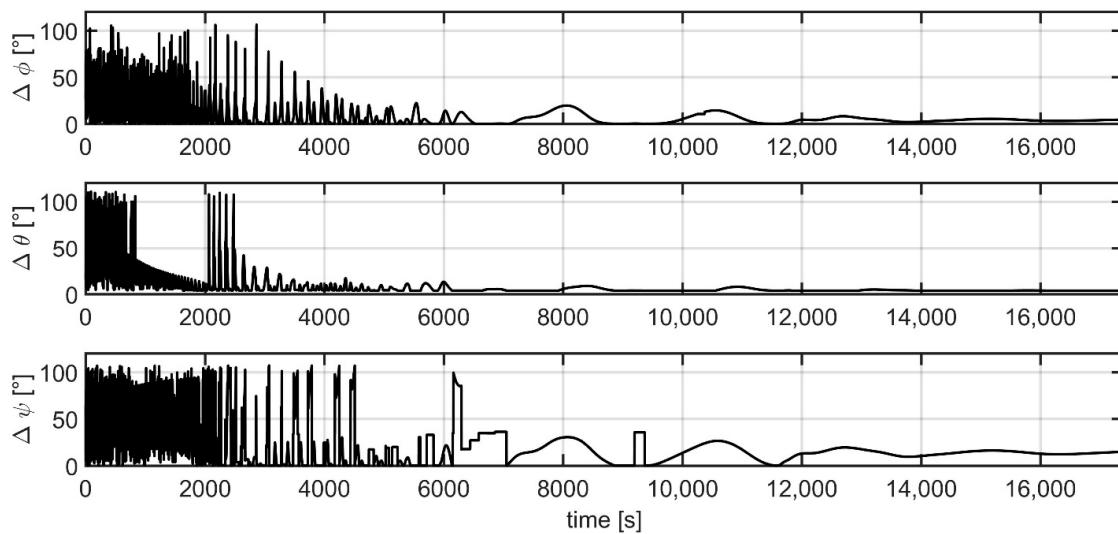


Figure 2. Difference between the actual and estimated Euler angles for Test Case 1.

In particular, a highlight of Figure 2 from time 6000 s to the end of the simulation is reported in Figure 3, showing that the estimated φ and θ enter the $\pm 10^\circ$ error band but the same is not verified for ψ . The Root Mean Square (RMS) values of the estimation errors for three different time windows are reported in Table 3.

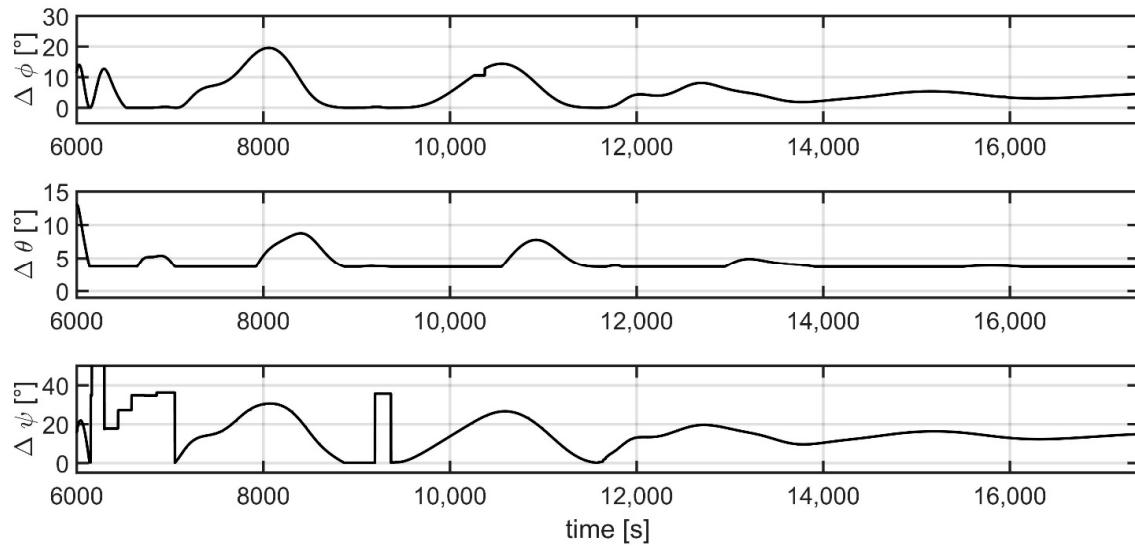


Figure 3. Detail of Figure 2 in the range 0–30° from 6000 to 17,386 s.

Table 3. Root Mean Square of attitude estimation errors for Test Case 1.

TC1			
Time Window (s)	RMS φ (°)	RMS θ (°)	RMS ψ (°)
0–6000	17.99	22.80	42.62
6000–12,000	7.99	1.95	23.72
12,000–17,386 (end)	4.42	3.97	14.23

Figures 4 and 5 show the comparison and the estimation error between the actual angular rates and the estimated ones.

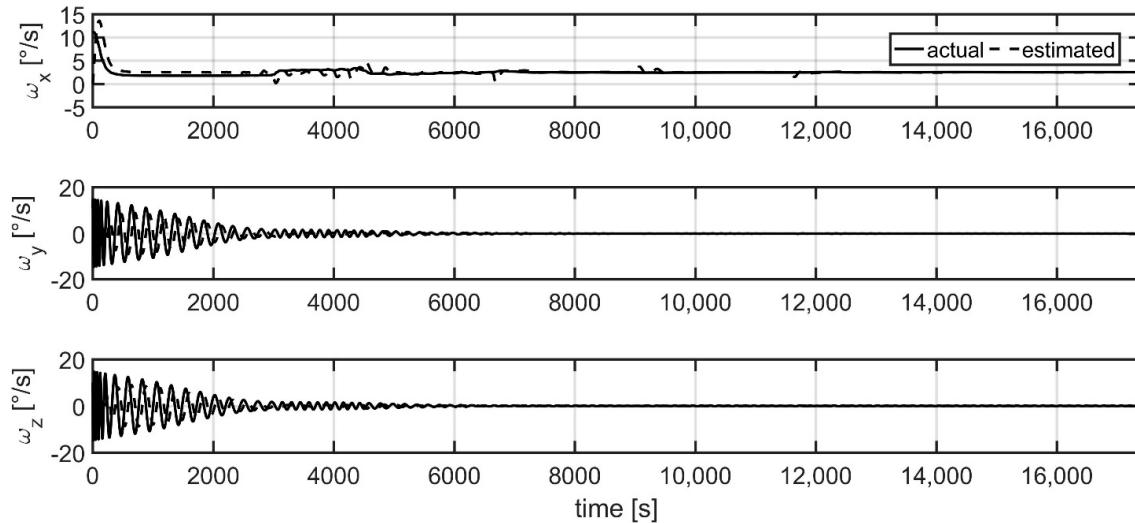


Figure 4. Actual and estimated angular rates for Test Case 1.

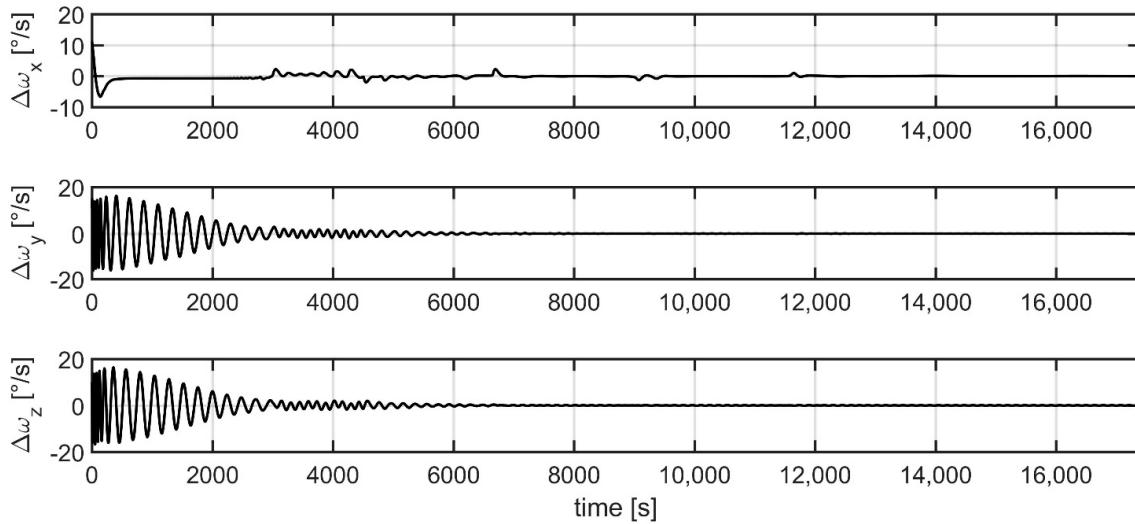


Figure 5. Difference between the actual and estimated angular rates for Test Case 1.

It can be noticed that the angular rate ω_x reaches the target value of 2.5° , while ω_y and ω_z approach zero. The angular rates enter the $\pm 0.2^{\circ}/s$ error band in the times and with the RMS error indicated in Table 4.

Table 4. Root Mean Square of the angular rates estimation error and settling times for Test Case 1.

TC1			
RMS ω_x ($^{\circ}/s$)	0.034	$t_{s,x}$ (s)	11707
RMS ω_y ($^{\circ}/s$)	0.004	$t_{s,y}$ (s)	6639
RMS ω_z ($^{\circ}/s$)	0.003	$t_{s,z}$ (s)	6697

Simulations results indicate that the control law Equations (4a)–(4c) is effective in producing the desired attitude, with the x-axis aligning to the Geomagnetic field vector, as shown in Figure 6.

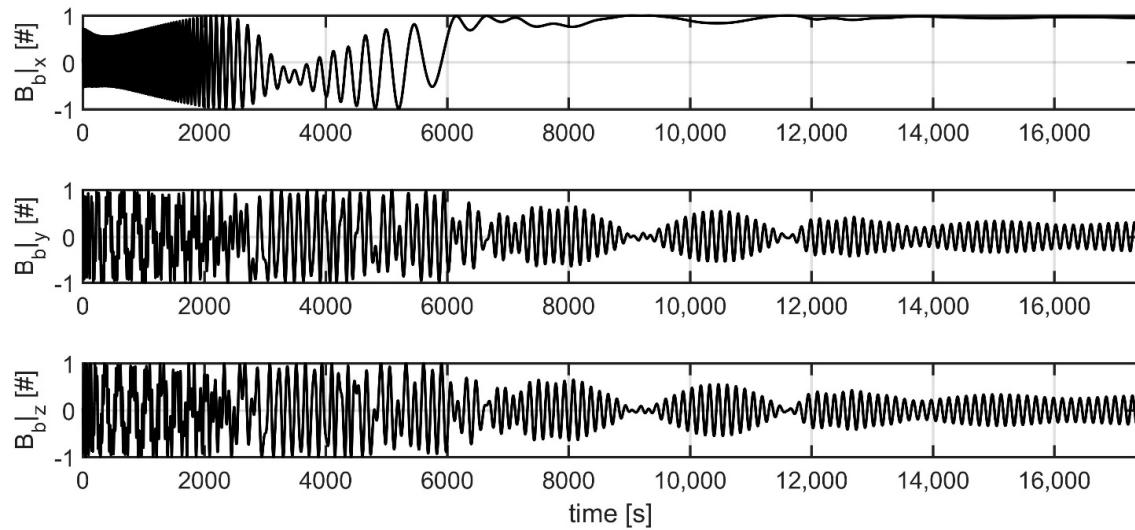


Figure 6. Attitude behavior with respect to the Geomagnetic field of Test Case 1.

The same analysis is repeated for the TC2, in which the satellite is supposed to be deployed at lower angular rates. The MSE between the actual A and the estimated $A^{(k)}$ are plotted in Figure 7. In TC2, the MSE enters the $\pm 10\%$ error band with respect to the final value of 1.158×10^{-2} in 12,111 s. The MSE for 5 s and 6 s delays are 1.071×10^{-2} and 1.539×10^{-2} , respectively, the 8% lower and the 25%

higher than the final MSE for TC2. Similar to TC1, this result indicates which the delay process is in the interval 5–6 s.

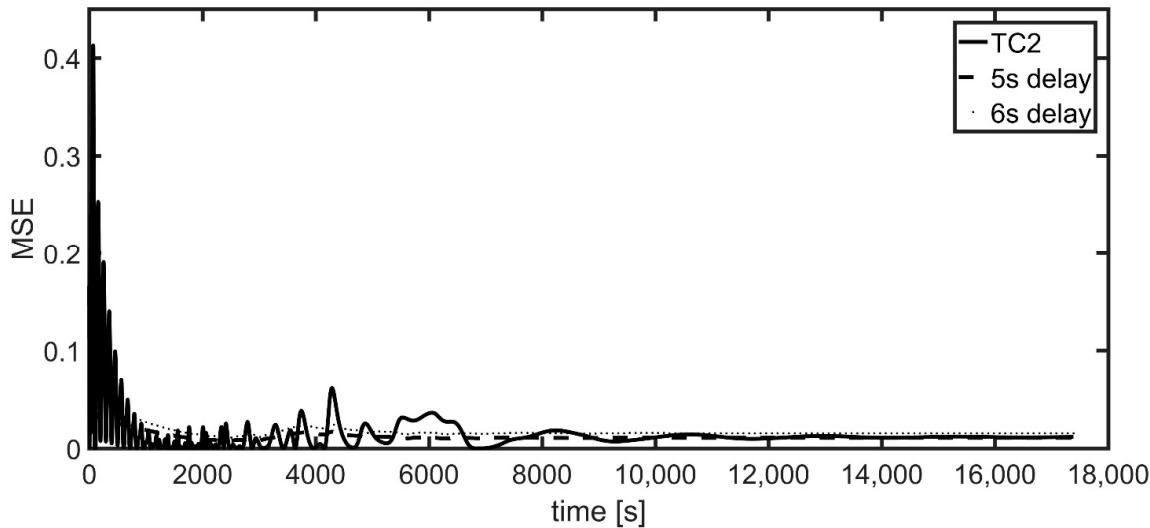


Figure 7. Mean Squared Error calculated on the estimated A^k and on the actual A with the 5 s and 6 s delay for Test Case 2.

The estimation errors on the Euler angles show similar behavior in time, as reported in Figure 8, with φ and θ entering the $\pm 10^\circ$ error band in 4740 s and 5367 s, and ψ settling between 10° and 15° . The RMS values for TC2 are reported in Table 5.

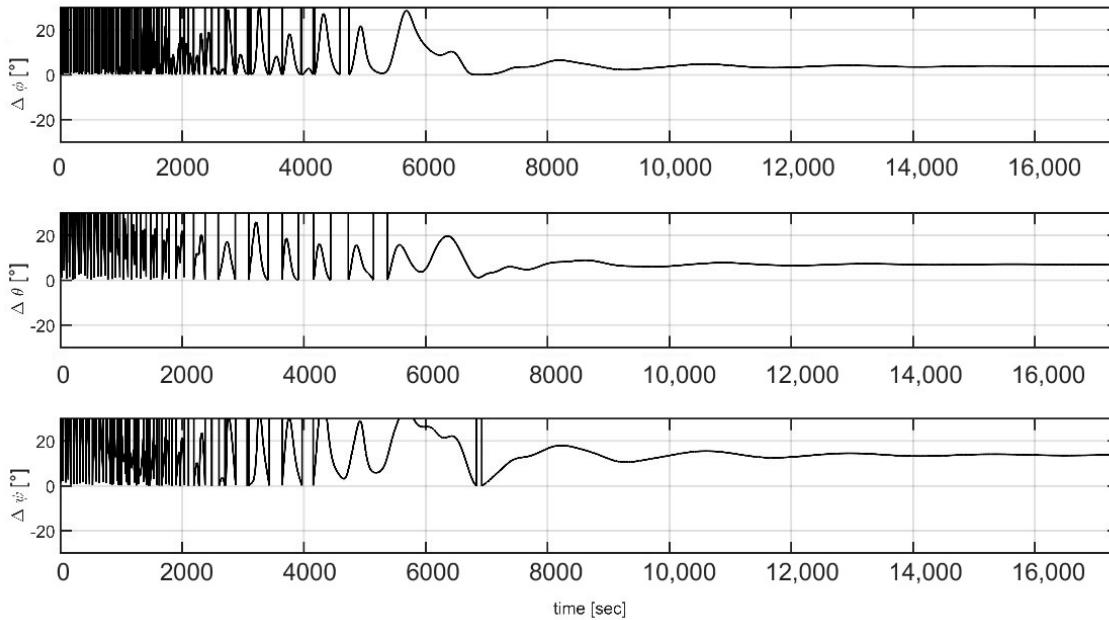


Figure 8. Difference between the actual and estimated Euler angles for Test Case 2 in the range $\pm 30^\circ$.

Table 5. Root Mean Square of attitude estimation errors for Test Case 2.

TC2			
Time Window (s)	RMS φ ($^\circ$)	RMS θ ($^\circ$)	RMS ψ ($^\circ$)
0–6000	16.70	11.15	26.69
6000–12,000	4.69	5.61	14.41
12,000–17,386 (end)	3.77	4.03	13.76

With respect to the angular rates, the target values are reached, and the error band of $\pm 0.2^\circ/\text{s}$ is entered for all the axes in a time shorter than that for TC1. The results are shown in Figure 9 and in Table 6, including the RMS of angular rates estimation errors after settling time.

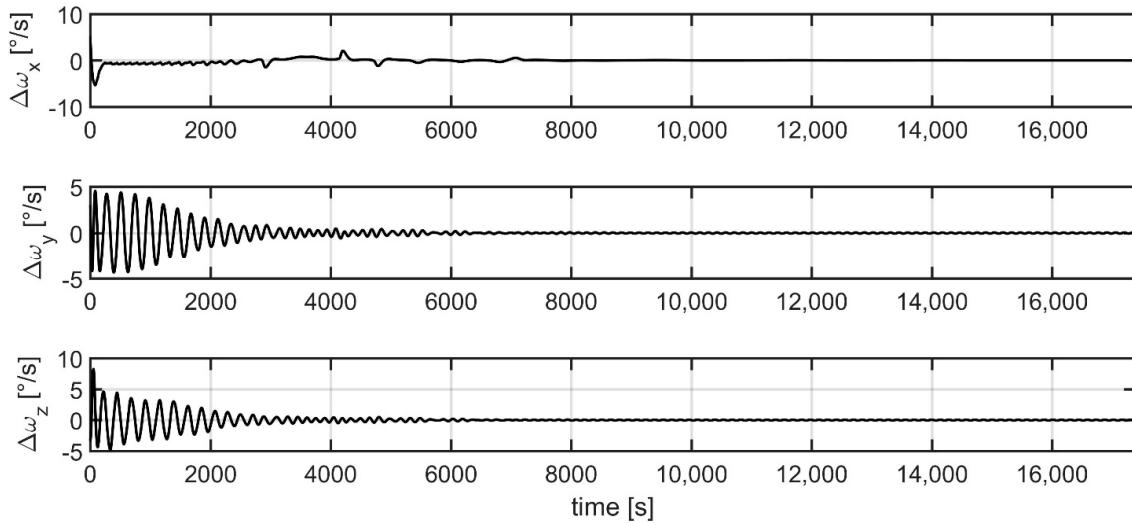


Figure 9. Difference between the actual and estimated angular rates for Test Case 2.

Table 6. Root Mean Square of the angular rates estimation error and settling times for Test Case 2.

TC2			
RMS ω_x ($^\circ/\text{s}$)	0.014	$t_{s,x}$ (s)	7175
RMS ω_y ($^\circ/\text{s}$)	0.001	$t_{s,y}$ (s)	5550
RMS ω_z ($^\circ/\text{s}$)	0.002	$t_{s,z}$ (s)	5584

To evaluate the robustness of the algorithm, a total of 100 Test Cases (TC) is simulated, selecting random initial conditions for the orbital parameters, initial attitude, and initial angular rates. The values are selected within the following ranges:

$$\begin{aligned} \Omega, \phi_0 &\in [-180^\circ, 180^\circ] \\ \varphi, \theta, \psi &\in [-180^\circ, 180^\circ] \\ \omega_x, \omega_y, \omega_z &\in [-10^\circ/\text{s}, 10^\circ/\text{s}] \\ h &\in [400 \text{ km}, 700 \text{ km}] \\ i &\in [80^\circ, 100^\circ] \end{aligned}$$

There are different altitudes; therefore, the orbital period of the 100 TC will be different, therefore a unique total time for the simulation is selected, and it is set equal to 18,000 s, to be comparable with TC1 and TC2. To ensure that all the simulated TC converge within the mentioned time, the control gains were set to the following values $k_p = 400$, $k_1 = 10$, and $k_2 = 1$. The filter parameters are the same for all the TC and equal to the values reported in Table 1.

The MSE was calculated on $A^{(k)}$ for all the 100 TC, and it settles to a value ranging from 1.856×10^{-2} to 2.505×10^{-2} , proving that the algorithm is robust with respect to the uncertain initial. For a clearer evaluation of the performance, the mean value for the RMS of the attitude estimation errors over the 100 TC was calculated and are reported in Table 7. The results in the table are in agreement with those evaluated for TC1 and TC2, with the accuracy of attitude determination exceeding the threshold level only for ψ . Angular rates determination is verified as well, registering that all the TC enter the $\pm 0.2^\circ/\text{s}$

error band within 14,902 s. The mean value for the RMS of the angular rates estimation errors was calculated and are reported in Table 8.

Table 7. Mean Root Mean Square of the attitude estimation errors for the 100 Test Cases.

100 TC			
Time Window (s)	RMS φ ($^{\circ}$)	RMS θ ($^{\circ}$)	RMS ψ ($^{\circ}$)
0–6000	19.79	17.99	34.39
6000–12,000	11.35	9.52	25.07
12,000–18,000 (end)	5.65	6.55	18.06

Table 8. Mean Root Mean Square of the angular rates estimation errors for the 100 Test Cases.

100 TC			
Time Window (s)	RMS ω_x ($^{\circ}/s$)	RMS ω_y ($^{\circ}/s$)	RMS ω_z ($^{\circ}/s$)
0–6000	1.21	1.49	1.49
6000–12,000	0.16	0.02	0.02
12,000–18,000 (end)	0.14	0.01	0.01

Numerical analyses here discussed represent only a preliminary result. The final validation of the algorithm is performed by HiL simulations, discussed in the following section.

5. Implementation of FPGA and Hardware-in-the-Loop Simulations

Once validated, the global behavior of the algorithm was implemented on an Artix-7 xc7a50t FPGA, which is a core of the OBC currently under development for STECCO (Space Traveling Egg-Controlled Catadioptric Object), the first PocketQube of the School of Aerospace Engineering. The main blocks of the algorithm are shown in Figure 10.

The Geomagnetic field is simulated using Matlab Simulink and sent to the FPGA for processing them. The Geomagnetic field data were processed to include the quantization error characterizing the 12-bit MEMS three-axis magnetometer and random white noise, considering a standard deviation of 2×10^{-7} T. The same systolic array architecture is used for both angular rates and attitude estimation, using Equations (7a)–(7d) and (16a)–(16d). The second-order Butterworth low-pass filter was implemented according to [38,39].

When implementing an algorithm on FPGA, its mathematical formulation is converted into several logical operations, which are stored into organized structures called Look Up Tables (LUT) and executed with desired time scheduling, under the control of Flip Flops (FF) [40]. The number of operators that can be stored in the LUT, as well as the total number of LUT and FF, is limited. Therefore, minimizing their utilization is a crucial task, and for this aim, systolic array architecture was considered [28]. It is worth highlighting that, as mentioned before, the use of the same architecture for both attitude matrix and angular rates estimation further increases the efficiency in terms of usage. Similar considerations apply for the total available memory (RAM) and the usage of the Digital Signal Processor, leading the logical process.

The resource utilization for the algorithm implemented on the FGPA is reported in Table 9. One run of the algorithm requires 53 clock cycles, 40 of which are addressed to the systolic array architecture, which, therefore, represents the most computationally demanding section.

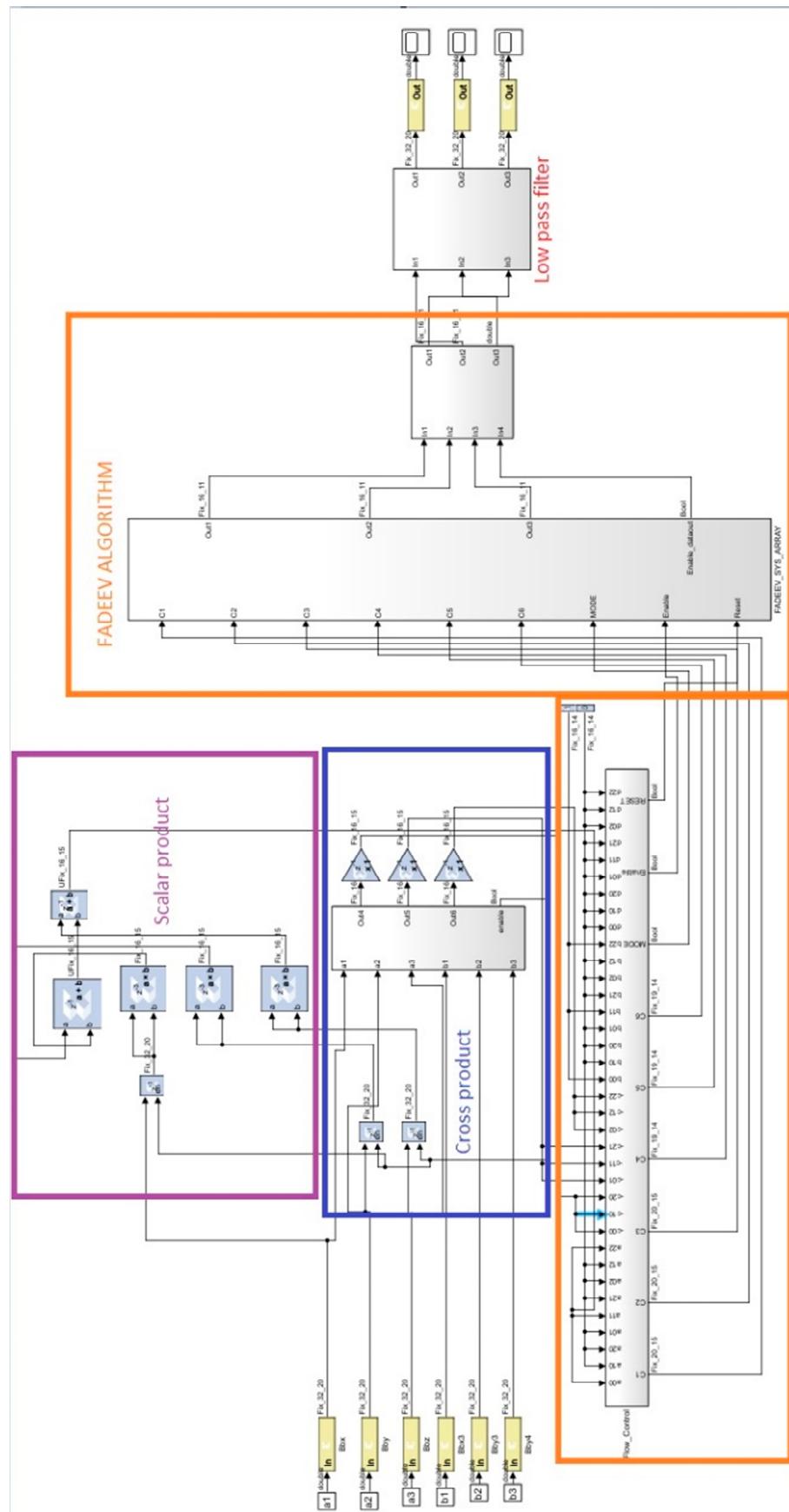


Figure 10. Main blocks of the attitude determination algorithm implemented on Field Programmable Gate Array.

Table 9. Indicative values of the utilization and power required for the attitude determination algorithm.

Resource	Utilization	% Utilization
Look Up Table	4738	14.53
Flip Flop	2576	3.95
RAM	4.50	6
DSP	30	25
Power [W]	0.277	
Digit representation	32-bit Fixed-point	
No. clock cycles	53	

The HiL simulations are performed based on the same data used for TC1, reported in Tables 1 and 2. The high working frequency of the FPGA, equal to 50 MHz, allows reducing both the delay due to the filtering process, corresponding to a few clock cycles (2×10^{-8} s) and that related to the sampling frequency of the magnetometer, which is here increased to $f_k = 100$ Hz, with benefit on the accuracy of the algorithm. In the selection of the sampling frequency, the Nyquist criterion must be considered to avoid aliasing and, therefore, loss of information. In particular, according to Fonod and Gill [33], the sampling frequency can be related to the maximum angular rate expected for the satellite as $f_k \geq \frac{\omega_{max}}{\pi}$, a condition which is verified for TC1 and $f_k = 100$ Hz. The MSE for TC1 at 100 Hz sampling frequency is represented in Figure 11, finally showing that the impact of process delay is a major cause of the error and that this delay can be reduced to negligible values in real hardware implementation.

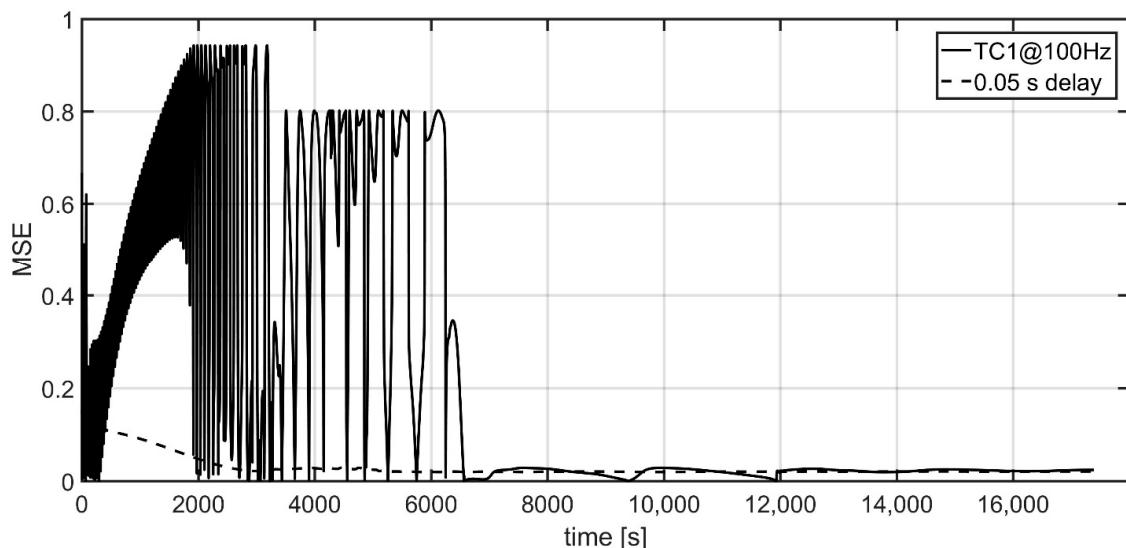


Figure 11. MSE calculated on the estimated A^k and on the actual A with 5×10^{-2} s delay for Test Case 1 at 100 Hz.

It can be noticed that when the algorithm converges, the MSE settles in the range from 1.056×10^{-2} to 1.1805×10^{-2} , a value included between the MSE calculated for A with 5×10^{-2} s and 6×10^{-2} s delay, respectively equal to 1.038×10^{-2} and 1.2455×10^{-2} . The estimation errors on the Euler angles are shown in Figure 12.

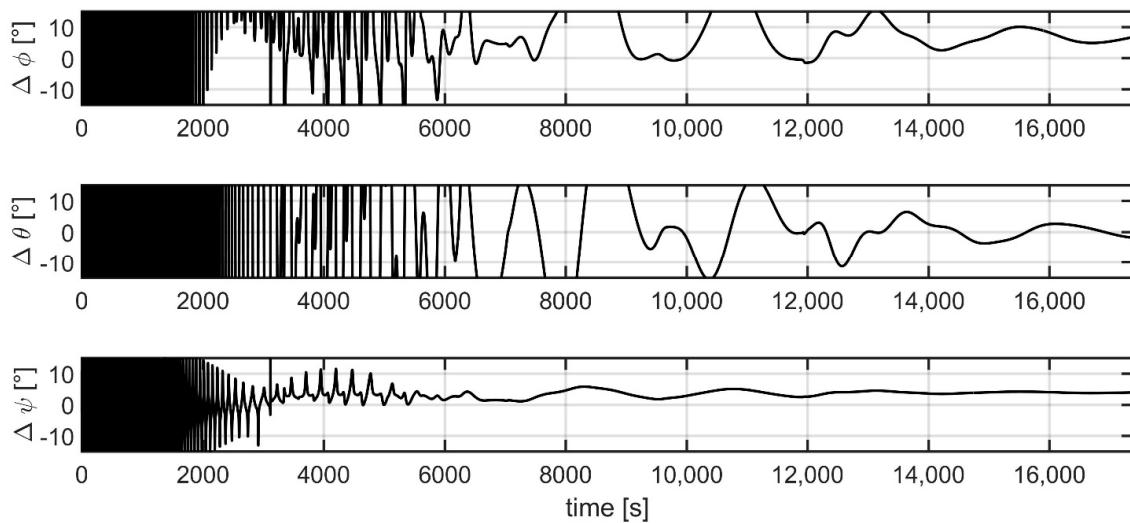


Figure 12. Difference between the actual and estimated Euler angles for Hardware-in-the-Loop simulation of Test Case 1 in the range $\pm 15^\circ$.

The results obtained from the HiL simulations match those from numerical analysis in Section 4, with φ , θ , and ψ entering the $\pm 10^\circ$ error band in 13,431 s, 12,655 s, and 4775 s. The RMS values of the attitude estimation error are reported in Table 10.

Table 10. Root Mean Square of the attitude estimation errors for Hardware-in-the-Loop simulation of Test Case 1.

TC1			
Time Window (s)	RMS φ ($^\circ$)	RMS θ ($^\circ$)	RMS ψ ($^\circ$)
0–6000	17.99	22.80	23.72
6000–12,000	13.99	11.96	4.23
12,000–17,386 (end)	5.43	3.15	3.71

The time behavior of the estimation errors on the angular rates is reported in Figure 13, showing that ω_x reaches the target value of $2.5^\circ/\text{s}$, while ω_y and ω_z approach zero. The settling times and the RMS values of the estimation of the angular rates after settling are reported in Table 11.

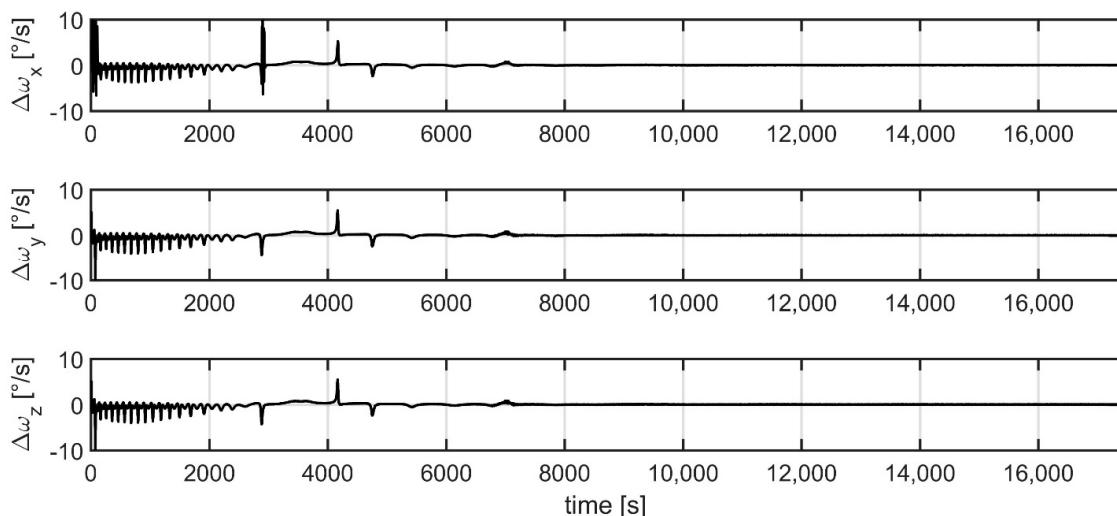


Figure 13. Difference between the actual and estimated angular rates for Hardware-in-the-Loop simulation of Test Case 1 in the range $\pm 10^\circ/\text{s}$.

Table 11. Root Mean Square of angular rates estimation error and settling times for Hardware-in-the-Loop simulation of Test Case 1.

TC2			
RMS ω_x ($^{\circ}/\text{s}$)	0.034	$t_{s,x}$ (s)	7104
RMS ω_y ($^{\circ}/\text{s}$)	0.004	$t_{s,y}$ (s)	7140
RMS ω_z ($^{\circ}/\text{s}$)	0.003	$t_{s,z}$ (s)	7155

6. Discussion

After having verified the performance of the algorithm, it is worth to compare it to other magnetometer-only solutions available in the literature [19–25]. As mentioned in the introduction, these are typically based on the use of a Kalman filter (EKF or UKF) and can provide accuracy as good as 1° and $0.01^{\circ}/\text{s}$ in the estimation of, respectively, the Euler angles and the angular rates. By comparison with the values reported in Tables 10 and 11, it can be noticed that the accuracy of KF based algorithms is higher than that of the solution proposed. This result indicates that further investigation should be performed to improve the accuracy of the algorithm, but the output from HiL simulations is promising.

Nevertheless, the attitude matrix and angular rate estimation algorithm discussed here is proposed to be a backup solution to be activated in case of failure of the primary attitude determination system. Therefore, also other features, besides accuracy, should be taken into account to evaluate its suitability. In particular, two aspects are worth to be examined, which are power and area usage. The former should be limited to be compliant with realistic power limitations occurring when a backup mode is required. The latter represents a constraint for the implementation on the FPGA, as discussed in Section 5, and should be minimized. Using systolic array architecture represents our attempt to reduce both power and area usage. This task was achieved by designing the attitude and angular rate estimation algorithm to be implemented in the Faddeev form.

A detailed comparison in terms of power usage between the solution proposed and the KF based algorithms available in the literature would require the implementation of these onto microprocessors, which is beyond the scope of this work. Nevertheless, an estimation of power usage can be provided from the analysis by Mohd et al. [41], evaluating $N \times N$ matrix multiplication performance on FPGA, using a systolic array, and on two microprocessors. A comparison between the power usage by the two devices in terms of the size N of the matrix is reported in Table 12, considering the lowest value between the two microprocessors.

Table 12. Comparison of the power usage between FPGA and microprocessor [41].

Matrix Size	Power Usage (mW)	
	FPGA	Microprocessor
2×2	237.05	794.24
4×4	287.70	791.43
8×8	290.27	788.36
16×16	293.53	840.42

It can be noticed that the estimated power usage from the algorithm proposed, implementing 3×3 matrix operations, is equal to 277 mW, a value compatible with those in Table 12. If referring to the methods [20–25], the matrix size ranges from 6×6 to 9×9 ; therefore, their power usage can be estimated to approximately 2.7 times higher.

Recalling that the comparative analysis presented in this section represents only a preliminary result, it indicates that the attitude matrix and angular rate estimation strategy proposed can be a suitable option when the use of FPGA based OBC is preferred and lower accuracy can be accepted for the benefit of lower power usage.

7. Conclusions

We proposed a magnetometer-only attitude determination strategy tailored for implementation on the FPGA core of an OBC, using systolic array architecture to enhance the efficiency of the algorithm. The determination algorithm is aimed to be a backup solution in case of failure of the primary determination system.

The algorithm provides results with accuracy of $\pm 0.2^\circ/\text{s}$ on the angular rates and of $\pm 10^\circ$ on the Euler angles. The numerical analyses and HiL simulations indicate that the algorithm suffers some processing delay, introduced by the filtering process, and by the sampling frequency of the magnetometer. This delay reduces as the operating frequency of the FPGA and, especially, the sampling frequency of the magnetometer increase. Simulations performed considering a MEMS three-axis magnetometer with sampling frequency as high as 100 Hz and an FPGA running at 50 MHz, indicate that the processing delay can be reduced to 50–60 milliseconds.

Despite being compatible with several small satellite missions and applications, the algorithm proposed is less accurate than other magnetometer-only methods available in the literature. Nevertheless, its lower power usage can represent an advantage for some critical scenarios. Furthermore, the low resources required and the use of systolic array architecture allows implementing the algorithm on the OBC at minimal area usage, resulting in a suitable and versatile solution for the use in small satellite missions.

Author Contributions: Conceptualization—S.C. and P.T.; methodology—S.C.; software—M.S.F.; “validation” S.C. and M.S.F.; data curation—S.C. and M.S.F.; writing—original draft preparation—S.C.; writing—review and editing—P.T. and M.S.F.; supervision—P.T. 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. Wertz, J.R. (Ed.) *Spacecraft Attitude Determination and Control*, 3rd ed.; Kluwer Academic Publisher: Dordrecht, The Netherlands, 1978; pp. 343–429.
2. Black, H. A Passive System for Determining the Attitude of a Satellite. *AIAA J.* **1964**, *2*, 1350–1351. [[CrossRef](#)]
3. Wahba, G. A Least Squares Estimate of Spacecraft Attitude. *SIAM Rev.* **1965**, *7*, 409. [[CrossRef](#)]
4. Davenport, P.B. *A Vector Approach to the Algebra of Rotations with Applications*; NASA TR D-4696; CFSTI: Springfield, VI, USA, 1968.
5. Schuster, M.D.; Oh, S.D. Three-Axis Determination from Vector Observations. *J. Guid. Control Dyn.* **1981**, *4*, 70–77. [[CrossRef](#)]
6. Schuster, M.D. The Quest for Better Attitudes. *J. Astronaut. Sci.* **2006**, *54*, 657–683. [[CrossRef](#)]
7. Bar-Itzhack, T.Y.; Harman, R.R. Optimized TRIAD Algorithm for Attitude Determination. *J. Guid. Control Dyn.* **1997**, *20*, 208–211. [[CrossRef](#)]
8. Reynolds, R.G. Quaternion Parametrization and a Simple Algorithm for Global Attitude Estimation. *J. Guid. Control Dyn.* **1998**, *21*, 669–671. [[CrossRef](#)]
9. Valenti, R.G.; Dryanovski, I.; Xiao, J. Keeping a Good Attitude: A Quaternion-Based Orientation Filter for IMUs and MARG. *Sensors* **2015**, *15*, 19302–19330. [[CrossRef](#)]
10. Markley, F.L. *Attitude Determination Using Two Vector Measurements*; NASA Conference Publications; Flight Mechanics Symposium Goddard Space Flight Center: Greenbelt, MD, USA, 1999.
11. Thébault, E.; Finlay, C.C.; Zvereva, T. International Geomagnetic Reference Field: The 12th generation. *Earth Planets Space* **2015**, *67*, 79. [[CrossRef](#)]
12. Diao, Z.; Quan, H.; Lan, L.; Han, Y. Analysis and compensation of MEMS gyroscope drift. In Proceedings of the 2013 Seventh International Conference on Sensing Technology (ICST), Wellington, New Zealand, 3–5 December 2013. [[CrossRef](#)]
13. Hajiyev, C.; Demet, C.G. Review of gyroless attitude determination methods for small satellites. *Prog. Aerosp. Sci.* **2017**, *90*, 54–66. [[CrossRef](#)]

14. Bar-Itzhack, I.Y.; Oshman, Y. Attitude Determination from Vector Observations. *IEEE Trans. Aerosp. Electron. Syst.* **1985**, *AES-21*, 128–136. [[CrossRef](#)]
15. Gebre-Egziabher, D.; Elkaim, G.G.; Powell, J.D.; Parkinson, B.W. A gyro-free quaternion-based attitude determination system suitable for implementation using low cost sensors. In Proceedings of the IEEE Position Location and Navigation Symposium, San Diego, CA, USA, 13–16 March 2000. [[CrossRef](#)]
16. Hyo-Sung, A. Gyroless Attitude Estimation of Sun-Pointing Satellites Using Magnetometers. *IEEE Geosci. Remote Sens. Lett.* **2005**, *2*, 8–12. [[CrossRef](#)]
17. Santoni, F.; Bolotti, F. Attitude determination of small spinning spacecraft using three axis magnetometer and solar panels data. In Proceedings of the 2000 IEEE Aerospace Conference, Big Sky, MT, USA, 25 March 2000; pp. 127–133. [[CrossRef](#)]
18. Natanson, G.A.; McLaughlin, S.F.; Nicklas, R.C. *A Method of Determining Attitude from Magnetometer Data Only*; NASA Goddard Space Flight Center, Flight Mechanics/Estimation Theory Symposium: Silver Spring, MD, USA, 1990; pp. 359–380.
19. Natanson, G.A.; Challacombe, J.; Deutschmann, J.; Baker, D.F. Magnetometer only attitude and rate determination for a gyro-less spacecraft. In Proceedings of the Third International Symposium on Space Mission Operations and Ground Data Systems, Greenbelt, MD, USA, 15–18 November 1994; pp. 791–798.
20. Ma, H.; Xu, S. Magnetometer-only attitude and angular velocity filtering estimation for attitude changing spacecraft. *Acta Astronaut.* **2014**, *102*, 89–102. [[CrossRef](#)]
21. Psiaki, M.L. Global Magnetometer-Based Spacecraft Attitude and Rate Estimation. *J. Guid. Control Dyn.* **2004**, *27*, 240–250. [[CrossRef](#)]
22. Hart, C. Satellite Attitude Determination Using Magnetometer Data Only. In Proceedings of the AIAA Aerospace Science Meeting Including the New Horizons Forum and Aerospace Exposition, Orlando, FL, USA, 5–9 January 2009. [[CrossRef](#)]
23. Sugimura, N.; Kuwahara, T.; Yoshida, K. Attitude Determination and Control System for Nadir Pointing Using Magnetorquer and Magnetometer. In Proceedings of the 2016 IEEE Aerospace Conference, Big Sky, MT, USA, 5–12 March 2016. [[CrossRef](#)]
24. Ma, G.F.; Jiang, X.Y. Unscented Kalman filter for spacecraft attitude estimation and calibration using magnetometer measurements. In Proceedings of the 2005 International Conference on Machine Learning and Cybernetics, Guangzhou, China, 18–21 August 2005. [[CrossRef](#)]
25. Searcy, J.D.; Pernicka, H.J. Magnetometer-Only Attitude Determination Using Novel Two-Step Kalman Filter Approach. *J. Guid. Control Dyn.* **2012**, *35*, 1639–1701. [[CrossRef](#)]
26. Carletta, S.; Teofilatto, P. Design and Numerical Validation of an Algorithm for the Detumbling and Angular Rate Determination of a CubeSat Using Only Three-Axis Magnetometer Data. *Int. J. Aerosp. Eng.* **2018**, *2018*, 9768475. [[CrossRef](#)]
27. Nash, J.G.; Hansen, S. Modified Faddeev Algorithm for Matrix Manipulation. In Proceedings of the SPIE 0495, Real-Time Signal Processing VII, San Diego, CA, USA, 28 November 1984. [[CrossRef](#)]
28. Kung, H.T. Why systolic architectures? *Computer* **1982**, *15*, 37–46. [[CrossRef](#)]
29. Teofilatto, P.; Testani, P.; Celani, F.; Nascetti, A.; Truglio, M. A nadir-pointing magnetic attitude control system for Tigrisat nanosatellite. In Proceedings of the International Astronautical Congress, Beijing, China, 23–27 September 2013; Volume 6, pp. 4818–4829.
30. Thompson, W.T. Spin Stabilization of Attitude against Gravity Torque. *J. Astronaut. Sci.* **1962**, *9*, 31–33.
31. Avanzini, G.; Fiulietti, F. Magnetic Detumbling of a Rigid Spacecraft. *J. Guid. Control Dyn.* **2012**, *35*, 1326–1334. [[CrossRef](#)]
32. Bruni, R.; Celani, F. Combining Global and Local Strategies to Optimize Parameters in Magnetic Spacecraft Control via Attitude Feedback. *J. Optim. Theory Appl.* **2019**, *181*, 997–1014. [[CrossRef](#)]
33. Fonod, R.; Gill, E. Magnetic Detumbling of Fast-tumbling Picosatellites. In Proceedings of the 69th International Astronautical Congress, Bremen, Germany, 1–5 October 2018.
34. Faddeev, D.K.; Sominsky, I. *Problems in Higher Algebra*, from the 1968 Russian edition; Yankovsky, G., Translator; Mir Publishers: Moscow, Russia, 1968.
35. Bigdeli, A.; Biglari-Abhari, M.; Salcic, Z.; Lai, Y.T. A New Pipelined Systolic Array-Based Architecture for Matrix Inversion in FPGAs with Kalman Filter Case Study. *EURASIP J. Appl. Signal Process.* **2006**, 089186. [[CrossRef](#)]

36. Vucha, M.; Rajawat, A. Design and FPGA Implementation of Systolic Array Architecture for Matrix Multiplication. *Int. J. Comput. Appl.* **2011**, *26*, 18–22. [[CrossRef](#)]
37. Nurdin, D.S.; Md.Isa, M.N.; Ismail, R.C.; Ahmad, M.I. High Performance Systolic Array Core Architecture Design for DNA Sequence. *MATEC Web Conf.* **2018**, *150*, 06009. [[CrossRef](#)]
38. Butterworth, S. On the Theory of Filter Amplifiers. *Exp. Wirel. Wirel. Eng.* **1930**, *7*, 536–541.
39. Bianchi, G.; Sorrentino, R. *Electric Filter Simulation and Design*, 1st ed.; McGraw-Hill Professional: New York, NY, USA, 2007.
40. Sulaiman, N.; Obaid, Z.A.; Marhaban, M.H.; Hamidon, M.N. Design and Implementation of FPGA-Based Systems—A review. *Aust. J. Basic Appl. Sci.* **2009**, *3*, 3575–3596.
41. Mohd, I.I.; Fatt, C.C.; Marsono, M.N. 32-Bit NxN Matrix Multiplication: Performance Evaluation for Altera FPGA, i5 Clarkdale, and Atom Pineview-D Intel General Purpose Processors. *Int. J. Comput. Appl.* **2012**, *52*, 11.



© 2020 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 (<http://creativecommons.org/licenses/by/4.0/>).