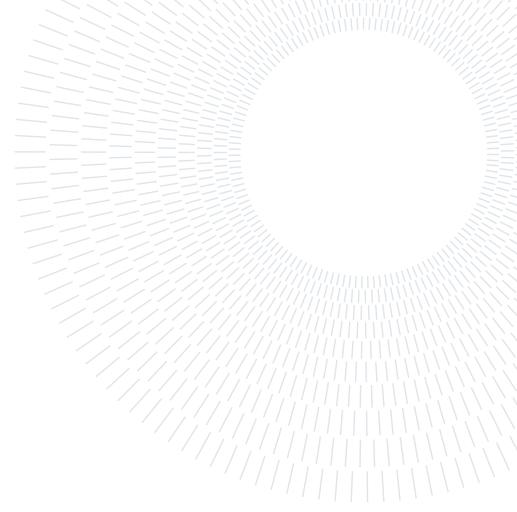




**POLITECNICO**  
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE  
E DELL'INFORMAZIONE



**Spacecraft Attitude Dynamics**  
**MSc Space Engineering**  
**Academic Year 2023/2024**

**Project Group n. 34**  
**Project n. 487**

**TEAM MEMBERS**

Person code	Surname	Name	Bachelor (type and university)
10729511	D'Annunzio	Valentina	Aerospace engineering Università di Bologna
10741197	Mascaretti	Mirko	Aerospace engineering Politecnico di Milano
10705062	Nicolucci Balocco	Edoardo	Aerospace engineering Politecnico di Milano
10735389	Orsenigo	Samuele	Aerospace engineering Politecnico di Milano

## PROJECT SPECIFICATIONS

	Assigned specification	Modification (if any)	Motivation for modification
<b>Platform</b>	Microsat (10 - 100 kg)		
<b>Attitude parameters</b>	Quaternions		
<b>Mandatory sensor</b>	Magnetic field sensor	Horizon sensor	Added due to low accuracy of magnetic field sensor
<b>Actuators</b>	3 reaction wheels		

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>Orbit and Spacecraft Characteristics</b>	<b>4</b>
2.1	Orbit Characterization . . . . .	4
2.2	Spacecraft Characteristics . . . . .	5
<b>3</b>	<b>Modeling</b>	<b>6</b>
3.1	Magnetic Field . . . . .	6
3.2	Gravity Gradient . . . . .	7
3.3	Solar Radiation Pressure - SRP . . . . .	7
3.4	Comparison between disturbances . . . . .	7
3.5	Sensoring and Attitude determination . . . . .	8
3.5.1	Triad-algebraic method . . . . .	8
3.5.2	Quest-statistical method . . . . .	8
3.6	Estimation of the angular velocity . . . . .	9
3.7	Our Simulink model . . . . .	9
<b>4</b>	<b>Control algorithms</b>	<b>10</b>
4.1	De-tumbling . . . . .	10
4.2	Slew manoeuvre . . . . .	11
4.3	Pointing with 3-axis stabilization . . . . .	11
<b>5</b>	<b>Results</b>	<b>12</b>
5.1	Comparison between Triad-algebraic and Quest-statistical methods	12
5.2	Montecarlo analysis . . . . .	13
5.3	Difference between q target and q estimated . . . . .	14
5.4	Reaction wheels behaviour . . . . .	15
5.5	Full orbit results . . . . .	16

## 1. Introduction

The purpose of this report is to present the results found by simulating the attitude dynamics and control of a satellite orbiting around the Earth, using the software *MatlabSimulink*. The mandatory specifications for our satellite made us decide to opt for a Low Earth Orbit (LEO) and in particular we chose to follow the same orbit of the ESEO mission, a satellite, which was launched in 2018, developed by the European Space Agency (ESA) in collaboration with various universities. The mandatory specifications that we were asked to model and then implement in our simulation were:

- magnetic field sensor to determine the attitude
- 3 reaction wheels as main actuators
- quaternions as attitude parameters

## 2. Orbit and Spacecraft Characteristics

### 2.1. Orbit Characterization

As said in the previous paragraph, the orbit we have chosen has the same orbital parameters of the ESEO satellite:

Orbit Altitude [Km]	Eccentricity	Inclination [deg]	Right ascension of the ascending node [deg]	Argument of the pericentre [deg]	Initial true anomaly [deg]
575	0.0011241	97.55	50.7464	206.3209	0

It is an almost circular Sun synchronous orbit (SSO) with an inclination close to 90 degrees. The period for a full revolution is around 96 minutes, due to its altitude of 575 km.

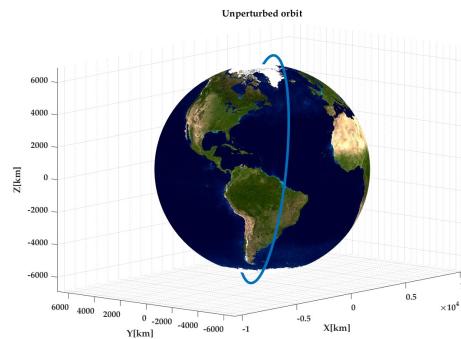


Figure 1: Unperturbed orbit

Because of the high inclination and the small distance from the centre of the Earth, in addition to the small surfaces of the satellite, the bigger disturbances are the magnetic field of the Earth and the gravity gradient, as we will later in the report. The air drag and the solar radiation pressure still have their influences on the undesired torque, but we decided to include in the simulation just the two greater as required.

## 2.2. Spacecraft Characteristics

The ESEO spacecraft (Fig.2) is a microsatellite, term applied to a satellite with a mass between 10 and 100 kg. In our case the total weight is 50 kg and the size is 33x33x63 cm. The inertia matrix of the satellite is obtained with the hypothesis of a homogeneously distributed mass inside the satellite, and for this reason the inertia matrix is a diagonal matrix and the body reference frame and the principal inertia axis one are aligned.

$$\mathbf{I} = \begin{bmatrix} 1.8125 & 0 & 0 \\ 0 & 1.8125 & 0 \\ 0 & 0 & 1.5267 \end{bmatrix} \text{kg} \cdot \text{m}^2 \quad (1)$$

ESEO's aim was to take imagery of the planet's surface and also of other celestial bodies, measure the radiation environment and test technologies for future educational satellite missions, therefore it had several sensors, buses and actuators to guarantee the success of the mission.

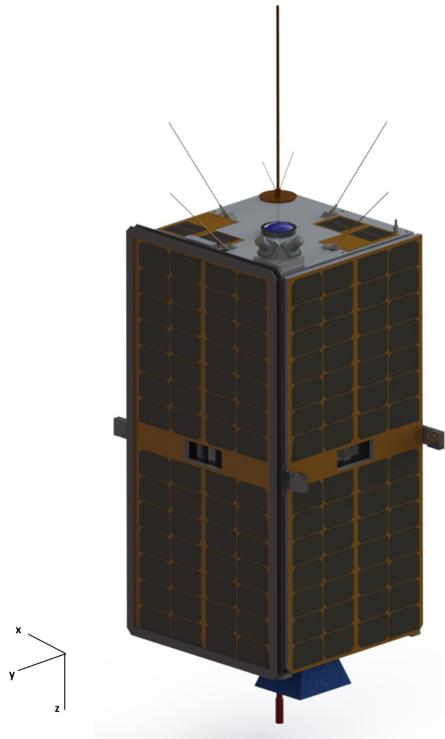


Figure 2: ESEO satellite with the body frame

Below is the list of sensors, actuators and on-board instrument chosen for this project:

- Solar Panels - they are located on all of the 4 lateral faces of 33x63 cm, producing all the needed electric power. The chosen ones are 1.5UCubesat solar panels, which have a mass of 65 g per unit, which makes a total mass of 3,120 kg, a specular reflectivity coefficient  $\rho_s$  of 0.8, while the two naked faces (the ones without solar panels) have  $\rho_s = 0.5$ . The diffusive reflection coefficient  $\rho_d$  is 0.1 for every surface;
- Magnetometer - the chosen one is MAG – 3, a 3-axis fluxgate magnetometer, with a mass of 100 g, accuracy of  $1.5 \cdot 10^{-4}$  and linearity of  $2 \cdot 10^{-3}$ . The Full Scale of 50000 nT was chosen based on the orbit characteristics, because the maximum value of the magnetic field was about 48000 nT. In Fig.3 we can see the components of

the magnetic field vector measured by the magnetometer. This sensor will be used, together with the model of the magnetic field, to compute the attitude determination;

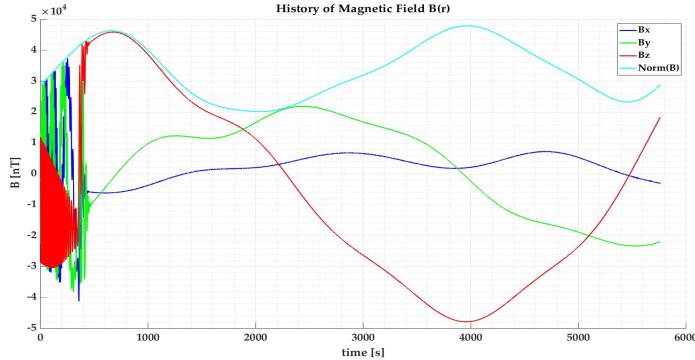


Figure 3: Measurement of the magnetic field with the magnetometer

- Horizon Sensor - we decided to add this sensor because the magnetic field sensor (as per project specifications) has low accuracy and it would have been difficult to have accurate results in the attitude determination. The reference sensor is *GEN1 : CubeSenseN*. It has a mass of  $30\text{ g}$  and a pointing accuracy of  $0.2^\circ$ . For this project, the assumption of a field of view (FOV) of  $360^\circ$  has been made, thinking that the sensor always knows where the Earth is, given that in real satellites there are more sensors to evaluate the attitude;
- Reaction Wheels (RW) - they are actuators consisting of spinning rotors. They work exchanging internal angular momentum between the satellite body and themselves. As per project specification, we implemented 3 RW in our model, in a typical configuration where their spin axes are aligned with the principal inertia axis of the satellite. The reference model is *Trillian – 1 Reaction Wheel*, with a mass of  $1,5\text{ kg}$  per unit and a total mass of  $4,5\text{ kg}$ . This wheel has a maximum rotation rate of  $6500\text{ rpm}$ , a maximum torque of  $47.1\text{ mNm}$  and an inertia of  $3,32 \cdot 10^{-3}\text{ kg} \cdot \text{m}^2$ . One thing to point out about the RW is that they have a saturation limit. This means that when they reach this value, no additional exchange of momentum with the satellite can happen in the same direction, so the reaction wheels must be de-saturated.

## 3. Modeling

### 3.1. Magnetic Field

The magnetic field has been implemented using the coefficients given by the International Geomagnetic Reference Field (IGRF) as it allows to have bigger accuracy over a model with a simple magnetic dipole, especially in the case of low orbits where the effects are stronger. The IGRF uses a model where the magnetic field is the negative gradient of the magnetic scalar potential.[2]

$$\vec{B}(r, \phi, \theta, t) = -\nabla V(r, \phi, \theta, t) = -\left(\frac{\partial V}{\partial r}, \frac{1}{r} \frac{\partial V}{\partial \theta}, \frac{1}{r \sin \theta} \frac{\partial V}{\partial \phi}\right) \quad (2)$$

The magnetic potential can be expressed in terms of a spherical harmonic expansion,

$$V(r, \phi, \theta, t) = a \sum_{n=1}^N \sum_{m=0}^n \left(\frac{a}{r}\right)^{n+1} (g_n^m(t) \cos m\phi + h_n^m(t) \sin m\phi) P_n^m(\cos \theta) \quad (3)$$

where  $g_n^m$  and  $h_n^m$  are the Gaussian coefficients. These coefficients are provided in 5-year intervals by the IAGA Working Group. The years for which the coefficients are provided are called model epochs, in our case the ones used are from 2020 with the secular variations related to those years. The components of the magnetic field are given in nanoTesla in the  $r - \phi - \theta$  frame.  $B_r$  is the component in the radial direction, positive outward,  $B_\theta$  is the coelevation component, positive towards south and  $B_\phi$  is the azimuth component, positive towards east. Once calculated, the components are converted, in sequence, to the geocentric inertial frame and to the body frame, in order to be used to calculate the momentum that they generate on the spacecraft.

### 3.2. Gravity Gradient

The effect of gravity gradient perturbation (GG), consequence of the non uniform mass distribution of our oblate planet, is still relevant despite our small satellite size. To model it, we calculated the spacecraft position vector in the principal axis reference system to get  $c_1, c_2$  and  $c_3$ , which take part into the following GG torque equation:

$$\mathbf{M} = \frac{3Gm_t}{R^3} \begin{Bmatrix} (I_z - I_y)c_2c_3 \\ (I_x - I_z)c_1c_3 \\ (I_y - I_x)c_1c_2 \end{Bmatrix} \quad (4)$$

where  $G$  is the gravitational constant,  $m_t$  the mass of the Earth,  $R$  the radius of the orbit and  $I_x, I_y, I_z$  the principal inertia moments of the satellite.

### 3.3. Solar Radiation Pressure - SRP

The Solar Radiation Pressure, or SRP, is the effect the Sun radiation has on our satellite, to which has to be added the radiation directly emitted by the Earth. We also considered when the spacecraft goes in an orbit region where the Sun is covered by the planet, causing obviously a zero torque. The expression we used to model it is the following:

$$\mathbf{F}_i = -PA_i(\hat{\mathbf{S}}_B \cdot \hat{\mathbf{N}}_{Bi}) \left[ (1 - \rho_s)\hat{\mathbf{S}}_B + \left( 2\rho_s(\hat{\mathbf{S}}_B \cdot \hat{\mathbf{N}}_{Bi}) + \frac{2}{3}\rho_d \right) \hat{\mathbf{N}}_{Bi} \right] \quad (5)$$

that contemplates  $\rho_s$  and  $\rho_d$  coefficients, the area  $A_i$  of the 6 surfaces, the angle  $\theta$  between the normal to the surface  $\hat{\mathbf{N}}_B$  and the radiation  $\hat{\mathbf{S}}_B$ , and the power of the radiation  $P = F_e/c$ , where  $F_e$  is  $1358 \text{ W/m}^2$  for the Sun and  $150 \text{ W/m}^2$  for the Earth, while  $c$  is the speed of light.

### 3.4. Comparison between disturbances

One of the project recommendation was to implement two among the many perturbations that affect the attitude of the satellite. The criterion of choice was to take the two disturbances with the major influence on the attitude of the spacecraft, based on the orbital region and the structure of the satellite. As mentioned before, we modeled the perturbations given by the magnetic field, the Gravity Gradient and the Solar Radiation Pressure. In Fig.4 (in logarithmic scale) we can observe the norm of the torques applied on the satellite caused by the three disturbances. Considering their maximum value, it is clear that the two that mainly affect the satellite are the Gravity Gradient and the magnetic perturbation, while the Solar Radiation Pressure is smaller by one order of magnitude.

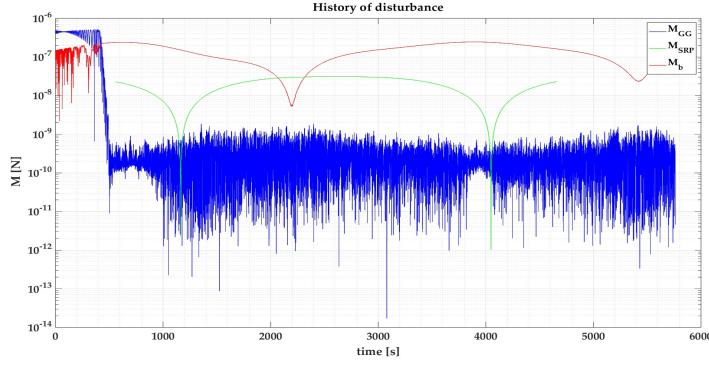


Figure 4: Norm of the disturbances

### 3.5. Sensoring and Attitude determination

In order to determine the satellite attitude, we need, first of all, the dynamics and kinematics blocks. These two subsystems rely on the real position and angular velocity of the spacecraft, giving us the actual attitude we use later on to compare it with the estimated one. The angular velocity  $\omega$  resulting from the dynamics is utilized, with the initial quaternion  $q_0$ , to compute for every time instant the quaternion  $q$ , which is our attitude parameter as required by specifications.

The sensors are modeled giving an error related to their precision and a discretization in time, respectively with a *white noise* and a *zero order hold* blocks. This way what we get is an on-board estimated attitude, with a little difference from the real one. The sensor we have chosen are the magnetometer, which is not so much precise but still quite reliable in our case, and the horizon sensor, which provides a more accurate pointing.

There are two principal methods to do it, both starting from 2 measurements: the Triad-algebraic method and the Quest-statistical method.

#### 3.5.1 Triad-algebraic method

The idea is simple: calling  $s_i$  the unit vectors measured by the sensors in body fixed frame,  $v_i$  the unit vectors of the same celestial bodies but referred to an inertial frame, we know that  $s_i = A_{B/N} * v_i$ , so, if we merge  $s_i$  and  $v_i$  into S and V matrices, we can calculate  $A_{B/N} = SV^{-1}$ . In our case we have just two measurements so we have to come up with a way to define the three main axis: we call the vectors measured in body frame  $p$  and  $q$  and we call  $a$  and  $b$  their corresponding directions in inertial space. This way, we can create two orthogonal frames with

$$s_1 = p, \quad s_2 = \frac{p \times q}{|p \times q|}, \quad s_3 = p \times s_2 \quad (6)$$

$$v_1 = a, \quad v_2 = \frac{a \times b}{|a \times b|}, \quad v_3 = a \times v_2 \quad (7)$$

to achieve the attitude matrix  $A_{B/N}$  that we can later use to compute  $q_{B/N}$ . This method works better if vector  $p$  is well measured and  $q$  is as orthogonal to  $p$  as possible.

#### 3.5.2 Quest-statistical method

It is a statistical method that consider the relative precision of the sensors in order to give importance to a measurement based on their reliability and find an attitude matrix  $A_{B/N}$

which minimize a suitable weighted error function

$$J = \frac{1}{2} \sum_{i=1}^n a_i \| \mathbf{s}_i - \mathbf{A} \mathbf{v}_i \|^2 \quad (8)$$

To find the optimal solution at least two measurements are needed and a simpler way to do that is using quaternions. The equation now becomes

$$\hat{J} = \mathbf{q}^\top \mathbf{K} \mathbf{q} - (\lambda \mathbf{q}^\top \mathbf{q} - 1) \quad (9)$$

where  $\mathbf{K}$  is a square matrix of order 4 made from a specific combination of  $s_i, a_i$  and  $v_i$ . In particular

$$\mathbf{K} = \begin{bmatrix} \mathbf{S} - \sigma \mathbf{I} & \mathbf{z} \\ \mathbf{z}^\top & \sigma \end{bmatrix} \quad (10)$$

$$\mathbf{B} = \sum_i a_i \mathbf{s}_i \mathbf{v}_i^\top \quad (11)$$

$$\begin{cases} \mathbf{S} = \mathbf{B} + \mathbf{B}^\top \\ \mathbf{z} = \sum_i a_i (\mathbf{s}_i \times \mathbf{v}_i) \\ \sigma = \text{tr}(\mathbf{B}). \end{cases} \quad (12)$$

The desired final quaternion is the eigenvector associated to the maximum eigenvalue of  $\mathbf{K}$ . Once we have in our project the dynamics and kinematics blocks and the outputs from the sensors, we can proceed to set a desired orientation and calculate the relative error and the estimated angular velocity, essential as inputs in control block.

### 3.6. Estimation of the angular velocity

The lack of a gyroscope on board of the satellite prevents the direct measurement of the angular velocity. In this case, it is possible to make an estimation of  $\omega$  using the magnetometer and horizon sensor's outputs. In this simulation, the angular velocity is expressed as function of the quaternions, through the following equations:

$$\frac{d\mathbf{q}}{dt} = \frac{1}{2} \Omega \mathbf{q} \quad (13)$$

$$\Omega \mathbf{q} = \begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & -q_1 \\ -q_2 & q_1 & q_4 \\ -q_1 & -q_2 & -q_3 \end{bmatrix} \begin{Bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{Bmatrix} = Q \boldsymbol{\omega} \quad (14)$$

$$\boldsymbol{\omega} = 2Q^* \frac{d\mathbf{q}}{dt} \quad (15)$$

where  $Q^*$  is the pseudo-inverse of the  $Q$  matrix.

This formulation has the disadvantage to have high-frequency noise and low-frequency estimation errors, and this will be evident in paragraph 5.1, in the plots of the angular velocity error.

### 3.7. Our Simulink model

In Fig.5 it is shown the whole Simulink model designed for this project. The dashboard at the top left of the model is necessary to select the parameters for the simulation, including the order of the model of the magnetic field (from 1 to 13) and the choice between the Triad-algebraic and the Quest-statistical methods.

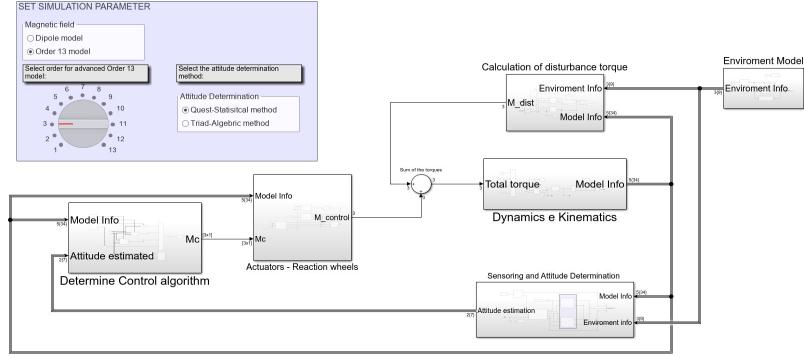


Figure 5: Our Simulink model

## 4. Control algorithms

In this section, we tackle the issue of control and modeling choices. To make the simulation as realistic as possible, we have opted to divide the mission into 4 phases:

- Waiting phase to facilitate movement to a safety location away from bus module
- De-tumbling
- Slew manoeuvre to target position
- Pointing 3-axis stabilization

For this mission we made the assumption that our spacecraft will follow the unperturbed orbit calculated through propagation. For our mission, we have designated Earth as our target, intending to align  $\hat{z}_{body}$  with the nadir to optimize observations of Earth.[6]

The control employs a non-linear control formulation, employing a specific Lyapunov function for each distinct problem. The control algorithm focuses primarily on determining a control moment that is then transmitted to the actuator system to achieve the desired objective.

### 4.1. De-tumbling

Typically, the de-tumbling phase is the first one of the satellite life, due to the fact that the detachment of the spacecraft from the deployer triggers an uncontrolled rotation around its three axis (tumbling indeed). For this reason, the aim of this phase is to slow down the satellite and bring all the components of angular velocity down to zero, so as to allow it to perform its mission. Although the reaction wheels are not the best actuators to perform the de-tumbling task, in this project we use them, changing their inertia value to  $5 \cdot 10^{-3} \text{ kg} \cdot \text{m}^2$  to avoid the saturation, to be able to use the same actuators in all the phases of the mission. The de-tumbling phase algorithm is quite simple. We can set  $\mathbf{u} = -K_1\boldsymbol{\omega}$  to achieve global asymptotic stability, where  $K_1$  is a tuning parameter. To choose the  $K_1$  we use the lqr matlab function, that solve the Riccati's equation. In order to do so, we need to linearize our problem in the proximity of an equilibrium state. We can describe our generic non-linear system considering only the input  $M_c$  as

$$I \frac{d\boldsymbol{\omega}}{dt} = I\boldsymbol{\omega} \times \boldsymbol{\omega} + \mathbf{M}_c \quad (16)$$

Considering the equilibrium around  $\bar{\omega}_z = n$  and  $\bar{\omega}_x = \bar{\omega}_y = 0$ , and considering  $\mathbf{M}_c = \mathbf{u} = -K_1\boldsymbol{\omega}$  we obtain the linearized matrix:

$$A = \begin{bmatrix} 0 & \frac{I_y - I_z}{I_x}n & 0 \\ \frac{I_z - I_x}{I_y}n & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \text{diag}\left(\frac{1}{I_i}\right) \quad (17)$$

The choice of weights Q and R is based on the limitation of their maximum absolute of state. This can be achieved by defining the matrices Q and R as

$$Q = \text{diag}\left(\frac{1}{x_i^2}\right) \quad R = \text{diag}\left(\frac{1}{u_i^2}\right) \quad (18)$$

For instance, we assumed a maximum angular velocity of  $x_i = 1 \text{ rad/s}$ , while for the input  $u_i$ , the maximum torque that the reaction wheels can provide. This helps avoid saturation issues.

## 4.2. Slew manoeuvre

A slew manoeuvre consists in a rotation of a certain angle with the aim to align the spacecraft with a target direction, that can be the ground station or any object of observation given by the mission. As aforementioned, the slew will be performed by the on-board actuators, the reaction wheels, thanks to the exchange of internal angular momentum with the satellite.

In this case we will use a slew that corresponds to an eigenaxis rotation. This control is particularly useful since it can be optimally tuned using the tuning law  $k_1^2 = 2k_2$ . We have thus selected the following control function and its respective Lyapunov function based on quaternions

$$\begin{cases} V = \frac{1}{2}\boldsymbol{\omega} \cdot \boldsymbol{\omega} + 2k_2H(q_{4e}) \\ \mathbf{u} = \boldsymbol{\omega} \times I\boldsymbol{\omega} - k_1I\boldsymbol{\omega} + k_2I\frac{dH(q_{4e})}{dq_{4e}}\mathbf{q}_e \end{cases} \quad (19)$$

where  $q_{4e}$  is the scalar component of the quaternion error  $\mathbf{q}_e$ , that represents the deviation of the actual orientation from the desired one  $\mathbf{q}_d$ .

$$\begin{bmatrix} q_{1e} \\ q_{2e} \\ q_{3e} \\ q_{4e} \end{bmatrix} = \begin{bmatrix} q_{4d} & q_{3d} & -q_{2d} & -q_{1d} \\ -q_{3d} & q_{4d} & q_{1d} & -q_{2d} \\ q_{2d} & -q_{1d} & q_{4d} & -q_{3d} \\ q_{1d} & q_{2d} & q_{3d} & q_{4d} \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \quad (20)$$

To satisfy the conditions of Lyapunov function we must have  $H(\pm 1) = 0$  and  $H(q_{4e}) > 0$  for  $q_{4e} \neq \pm 1$ , also, the choice of  $H(q_{4e})$  impacts how the quaternion error  $q_{4e}$  contributes to the overall stability of the system. In our case we choose  $H(q_{4e}) = 1 - q_{4e}$  in order to emphasize the cases where the scalar component and the error are close to each other.

The gain coefficient are tuned manually to obtain the best performance. We have selected  $k_1 = 0.8$  and  $k_2 = 0.32$ .

## 4.3. Pointing with 3-axis stabilization

Once the target direction has been reached, the purpose of the pointing with 3-axis stabilization is to keep this attitude along the orbit, controlling the satellite to avoid undesired angular velocities.

Considering this manoeuvre following a slew manoeuvre that positioned the satellite at the target and, in our case, at equilibrium, we implemented a linearized version of the function based on quaternions

$$u_i = M_{ic} = 2K_{pi}q_{ie}q_{4e} + k_{di}(\omega_i - \bar{\omega}_i) \quad (21)$$

where the equilibrium angular velocity is  $\omega = [0, 0, n]^T$ . Also in this case we implemented *lqr matlab function* to define the optimal  $k_1$  gain, and then calculated  $k_2$  with the already used tuning law  $k_1^2 = 2k_2$ . The matrices A and B are the same already used (equation 17), but this time, we changed the weight matrices Q and R. To gain more control over the state variables  $\omega$ , we set a value of  $Q = diag(10, 10, 1)$  and  $R = I$ . The choice of Q is not random; it's based on previous simulations. We found that during attitude determination, errors occur at certain points in our orbit due to approach to the poles. Since the  $w_z$  component has a larger error, by penalizing  $w_z$ , we restrict the error propagation from attitude to control.

## 5. Results

The first thing we can do is to analyse the passive stability of our spacecraft by looking at the coefficients:

$$\begin{cases} K_{yaw} = \frac{I_z - I_y}{I_x} = -0.1577 \\ K_{roll} = \frac{I_z - I_x}{I_y} = -0.1577 \\ K_{pitch} = \frac{I_y - I_x}{I_z} = 0 \end{cases} \quad (22)$$

As we can see the three coefficients all have negative values, so our case falls in a region where there is instability.

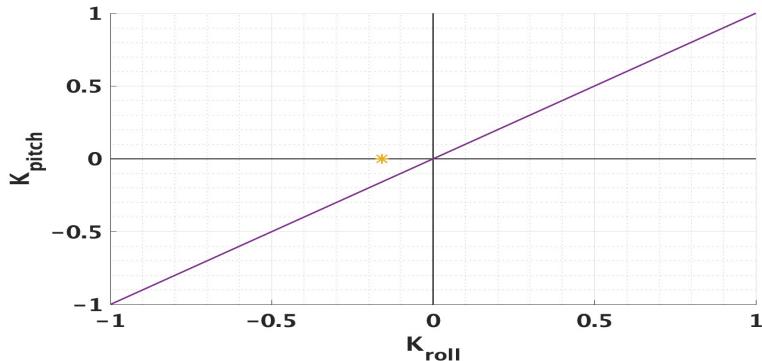


Figure 6: Stability diagram

This means that in the case of small perturbations the angular velocity, without control, will start to oscillate.

### 5.1. Comparison between Triad-algebraic and Quest-statistical methods

As mentioned in Chapter 3, to compute the attitude determination, two methods were implemented: the Triad-algebraic method and the Quest-statistical one. In Fig.7 we can see the errors between the components of the true angular velocity of the satellite and the estimated one, for both methods, in a limited time span between 1000 s and 3000 s.

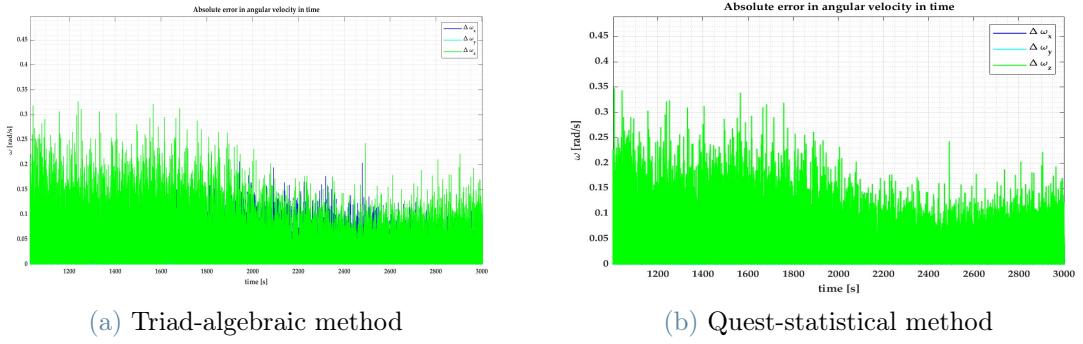


Figure 7: Error between the components of the true and estimated angular velocity

It can be seen that in Fig.7(a) the error of the component  $\omega_x$  (the blue one) of the angular velocity fluctuates more than the one in the Quest method, where the component cannot be seen. This shows that the Quest-statistical method is more accurate than the Triad-algebraic one. The greater precision of this method is due to the use of the weights to give more or less importance to one sensor or the other. In this simulation we chose to give weights of  $\frac{1}{6}$  and  $\frac{5}{6}$  respectively to the magnetometer and the horizon sensor, because the last one is more precise in its measurements. Following this result, we chose to use the Quest-statistical method to do the other analysis. The error in the angular velocities are not zero because the simulation computes them from the quaternions, as aforementioned in paragraph 3.5.

It is important to point out a peak of the error around  $4000\text{ s}$  (Fig.8), due to the passage of the satellite above one of the magnetic poles. Here the direction outputs of the magnetometer and the horizon sensor are close, so it is difficult to have an accurate attitude determination.

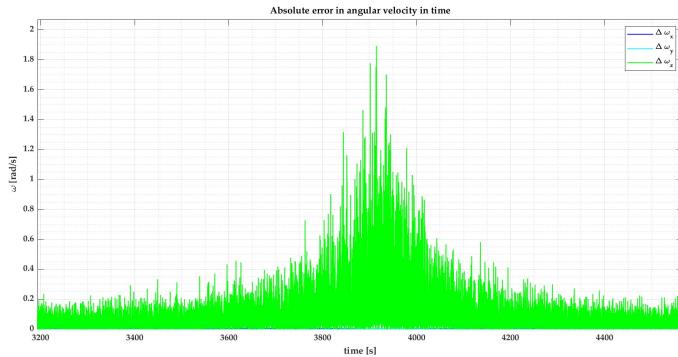


Figure 8: Peak of the error due to passage above one of the magnetic poles

## 5.2. Montecarlo analysis

In this paragraph, we have examined the convergence of our de-tumble and slew manoeuvre to target position algorithms. To systematically test the problem, we used the Monte Carlo method. This involves observing the outcome of numerous simulations starting from different initial conditions. We conducted 100 simulations by randomly selecting initial values for angular velocity (between  $-1$  and  $1\text{ rad/s}$ ) and  $q_0$ . For simulations, we employed a third-order magnetic field model and the Quest method for attitude determination. We set the  $t_{step} = 0.1\text{s}$  and  $t_{sensor} = 0.5\text{s}$ . In this analysis we don't consider the initial waiting time. As we can see in the figure, the angular velocity and the quaternions always converge to

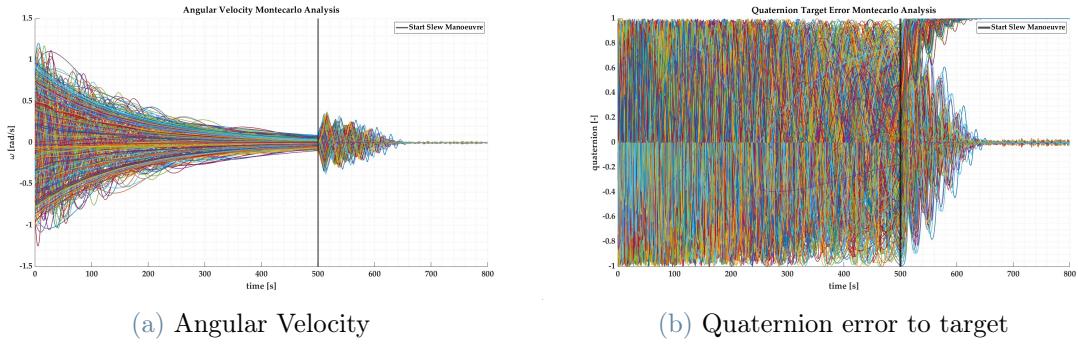


Figure 9: Montecarlo results

the target condition. Even in the worst case the angular velocity decreases exponentially, showing that the control algorithm used in the de-tumble manoeuvre and slew manoeuvre works for every starting condition.

### 5.3. Difference between $q$ target and $q$ estimated

After the slew manoeuvre we want the satellite to keep maintaining the orientation in the direction given by  $q$ -target. From the results we can see that the pointing is satisfying since all the components of the  $q$  estimated follow their respective component of  $q$ -target. We can see a slight oscillations around 1000 s and 4000 s, that is because at these specific times the satellite is orbiting close to the poles, reducing the accuracy of the measurements taken by the magnetometer and the horizon sensor since they measure direction that are close to be aligned.

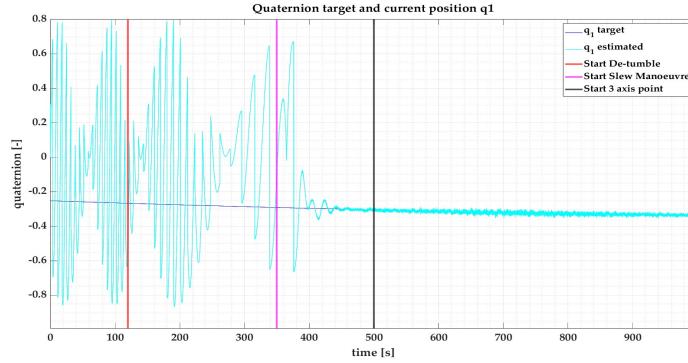


Figure 10: Comparison between  $q_1$  target and  $q_1$  estimated

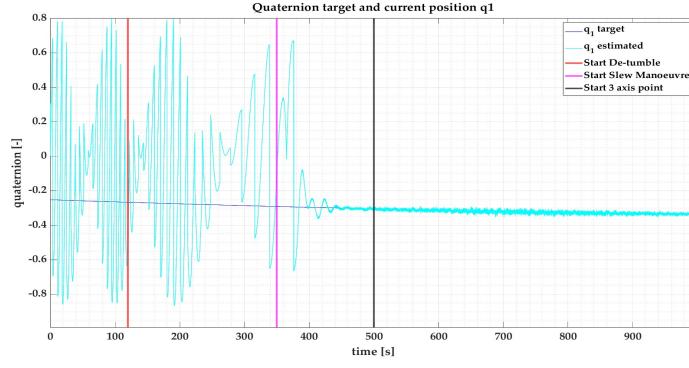


Figure 11: Comparison between  $q_2$  target and  $q_2$  estimated

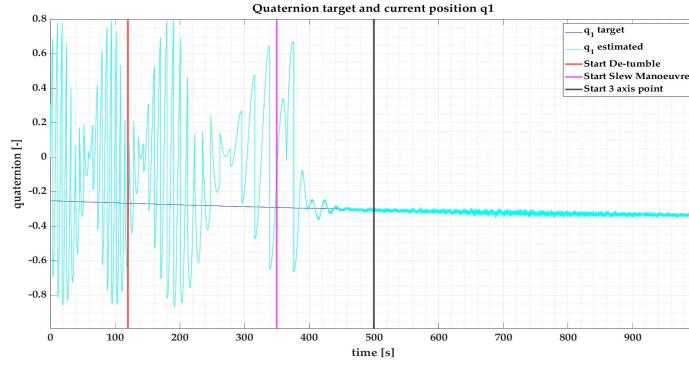


Figure 12: Comparison between  $q_3$  target and  $q_3$  estimated

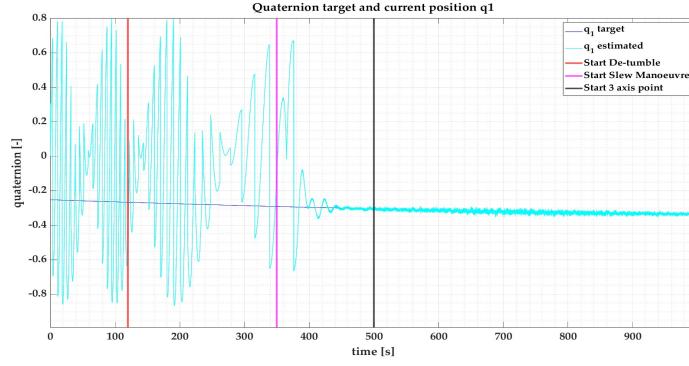


Figure 13: Comparison between  $q_4$  target and  $q_4$  estimated

## 5.4. Reaction wheels behaviour

The reaction wheels, with the changed value of the Inertia, bring acceptable results as they provide the necessary angular momentum during all three phases of the mission and they never pass the saturation limit of 6500 rpm (Fig.14), thanks to a saturation block on the possible torques that the reaction wheels could output. This result can be reached if the time the satellite takes to complete the manoeuvres is longer than usual, especially for the de-tumbling phase, and that is the reason for which the reaction wheels are not used to de-tumble the satellite in reality.

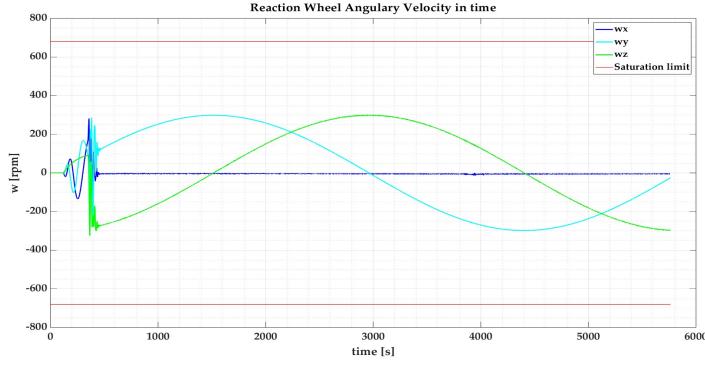


Figure 14: Angular velocity of the reaction wheels

## 5.5. Full orbit results

Finally, we can test the results of the control algorithm throughout the entire time period and conduct some comparisons. For this paragraph, the initial conditions are set as  $\omega_0 = [0.5, 0.5, 0.5]^T$  and  $q_0 = [0, 0, 0, 1]^T$ . The sample time of the sensor is  $ts_{sensor} = 0.05s$ , five time bigger than the sample time of the simulation  $t = 0.01s$ . We use third-order of the magnetic field model. It is interesting to see, in Fig.15, how the components of the true angular velocity vary over the course of the de-tumbling and the slew manoeuvre in the first phases of the mission, nearly reaching the value of zero with the beginning of the pointing with 3-axis stabilization.

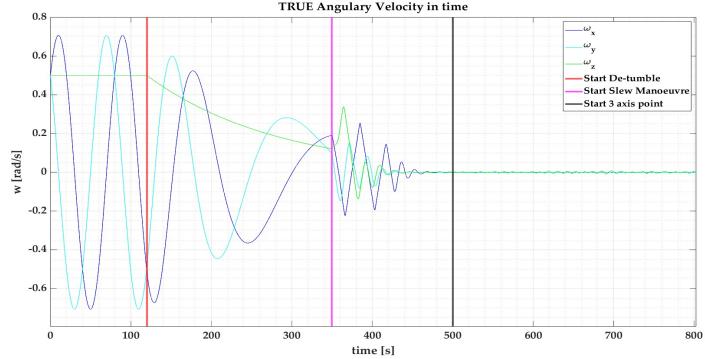


Figure 15: Components of true angular velocity in limited time span

Fig.16 and Fig.17 show the error between the target and the current attitude of the satellite. The two plots show the same results, but the physical meaning of the errors is more understandable in Fig.17, because it is simpler to visualize the rotations using the attitude angles in degrees. The zoom box highlights the greater oscillation of the pitch angle  $\psi$  compared to the yaw and roll angles, and this result confirms the study of the stability carried out at the beginning of this chapter.

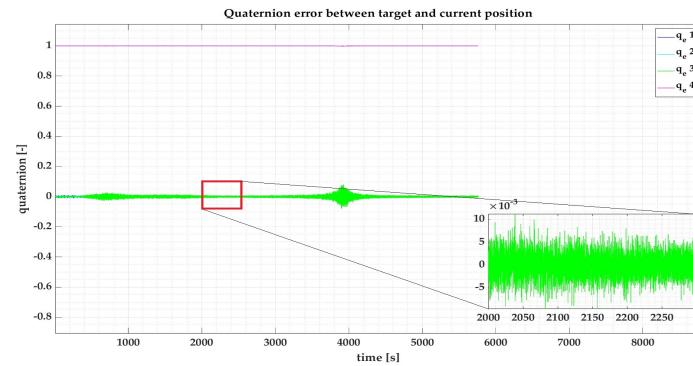


Figure 16: Quaternions error along the orbit

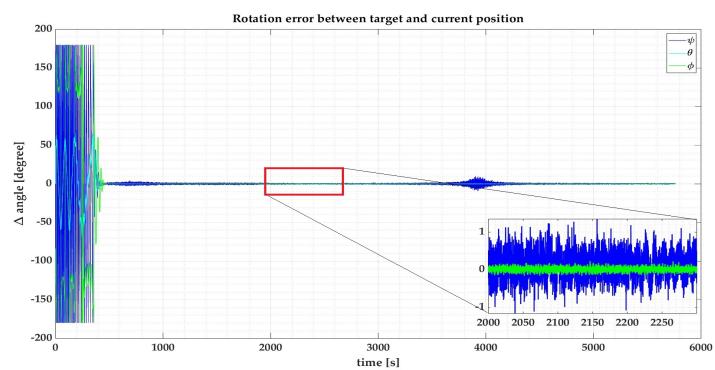


Figure 17: Angles error along the orbit

## References

- [1] [https://satsearch.co/.](https://satsearch.co/)
- [2] Jeremy Davis. Mathematical Modeling of Earth's Magnetic Field. Technical report, Virginia Tech, Blacksburg, VA 24061, 2004.
- [3] ESA. [https://www.esa.int/Education/ESEO/The\\_ESEO\\_Mission](https://www.esa.int/Education/ESEO/The_ESEO_Mission).
- [4] Prof. Franco Bernelli Zazzera. Course Notes. *Part 1: Attitude dynamics and kinematics*.
- [5] Prof. Franco Bernelli Zazzera. Course Notes. *Part 2: Attitude sensors and actuators, attitude determination*.
- [6] Prof. Franco Bernelli Zazzera. Course Notes. *Part 3: Introduction to state-space control and attitude control*.
- [7] Andrea Colagrossi Vincenzo Pesce and Stefano Silvestrini. *Modern Spacecraft Guidance, Navigation, and Control: From System Modeling to AI and Innovative Applications*. Elsevier, 2022.