Trabajo Fin de Grado en Física

Cubesat Attitude Control System based on embedded magnetorquers in photovoltaic panels

Mario Castro Santiago

Junio de 2018

UNIVERSIDAD
DE GRANADA

Tutor: Andrés María Roldán Aranda
Departamento de Electrónica y Tecnología de Computadores
Universidad de Granada

Abstract

The use of magnetic actuation in order to stabilize and control a small 1U Cubesat is studied and analyzed, as a required step for the development of the future university satellite GranaSAT-I. This thesis gather crucial theoretical contents concerning the attitude control of a satellite in an orbit below 500 km, where the International Space Station is able to deploy nano-satellites. Moreover, these contents have been implemented in a MATLAB simulator. One stabilizing control law has been succesfully tested with this tool, and two different control algorithms have shown partial success when a 3-axis control has been required. In parallel, an autonomous Cubesat prototype has been manufactured in the GranaSAT Laboratory. The stabilizing algorithm has been implemented on the onboard computer. Telemetry data during tests reflect an adequate performance of the prototype.

Resumen

Se estudia el uso de actuadores magnéticos con el objetivo de estabilizar y controlar un pequeño Cubesat 1U, como paso necesario para el desarrollo futuro satélite universitario GranaSAT-I. Esta tesis recaba contenidos teóricos fundamentales respecto al control de orientación de un satélite en una órbita por debajo de 500 km, donde la Estación Espacial Internacional es capaz de lanzar nano-satélites. Además, esta teoría ha sido implementada en un simulador en MATLAB. Con esta herramienta, se ha probado con éxito un algoritmo de estabilización, y dos algoritmos de control han mostrado un éxito parcial cuando un control en los tres ejes ha sido necesario. Paralelamente, se ha desarrollado un prototipo autónomo de Cubesat en el Laboratorio de GranaSAT. El algoritmo de estabilización ha sido implementado en el ordenador de abordo. Los datos enviados por telemetría durante las pruebas reflejan un adecuado rendimiento del prototipo.

Contents

Agradecimientos

Dicen que si trabajas de lo que te gusta, no trabajarás nunca. Para un físico enamorado con el entorno aeroespacial, este dicho es una verdad a medias. Porque ciertamente, estas 40 páginas esconden asombro, entusiasmo y felicidad con los descubrimientos personales y los resultados obtenidos. Pero tambien esconden días y días leyendo el mismo texto hasta volver un mes después sobre él y finalmente entenderlo, horas de simulación y programación a la caza de errores de última hora, y finalmente n soluciones para los $n + 1$ problemas que encuentras al entrar en el mundo de la electrónica.

Y en gran parte de ese tiempo, siempre te he tenido a mi lado, Celia, y lo sabes bien. Tanto por tu apoyo como por tus consejos, este trabajo no habría salido adelante sin ti; pero sabes que no puedo ponerte como cotutora, así que te cedo el primer lugar de esta breve lista.

También quiero agradecer a mi tutor de este trabajo, Andrés, por acercar un pedazo del espacio a los chavales de Granada, por la inestimable ayuda prestada a deshoras y por la montaña de problemas propuestos que han conseguido sacar mi mejor potencial.

Agradecer también a los miembros del grupo GranaSAT; a Pablo y a José Carlos, y especialmente a Luis, por la gran ayuda prestada con la electrónica y por la cinta aislante. No olvido a la empresa española DHV, a David y a Lourdes, en la parte correspondiente al desarrollo del prototipo.

Y también me acuerdo de mis compañeros de carrera, con los que he compartido esta etapa de comenzar a trabajar e investigar. Destaca Tinaut, miembro también del grupo GranaSAT, que trabajó duramente con el simulador solar (algunos dirán que es un foco de conciertos) y con el que pasé unos buenos ratos pensando en matrices de rotación.

Agradecer a Sonsoles la ayuda de última hora con el maravilloso idioma que es el inglés y sus preposiciones.

Por último y no menos importante, gracias a mis hermanos, Pablo y David; y a mis padres, quienes siempre me han incentivado a ir más allá, a extender mi actividad fuera de los estudios habituales, aunque ello haya supuesto un piso de estudiante algo desordenado (todo no se puede pedir).

Notation

$\mathbf{v}$	3x1 vector
$\hat{\mathbf{v}}$	3x1 unit vector
$\mathbf{v}^T$	vector transpose
$\mathbf{v}^f$	vector expressed in frame f
$\mathbf{v}_F$	unit vector of frame F
$\mathbf{R}_o^i$	3x3 rotation matrix from frame o to i
$\mathbf{R}^T$	matrix transpose
$\mathbf{q}_o^i$	4x1 quaternion from frame o to i
R_{ij}	element ij of the rotation matrix $\mathbf{R}$
$\boldsymbol{\omega}_{oi}$	angular speed of frame i with respect to frame o
$\boldsymbol{I}^f$	3x3 inertia matrix in frame f
$\boldsymbol{I}$	3x3 inertia matrix in the body frame
$\boldsymbol{I}_{3x3}$	3x3 identity matrix
$\mathbf{S}(\mathbf{v})$	3x3 skew-symmetric form of a vector: $\mathbf{S}(\mathbf{v}) = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix}$
M_{Earth}	Mass of the Earth
G	Gravitational constant

1 Introduction

This Bachelor Thesis covers the study of the attitude determination and control system (ADCS) of a future student Cubesat [1], which is the main goal of GranaSAT [2], the Electronics Aerospace Group of the University of Granada. Cubesats are small satellites, made up with one ore more 10x10x10 cm cubes, and there are many universities and related institutions that have already designed and put a Cubesat in orbit (listed here [3]). These projects are unique opportunities for university students, allowing them to put in practice their recently achieved abilities, forcing them to explore new horizons, and finally integrating minds from very different disciplines. Thereby, these three qualities make projects like GranaSAT a crucial link between the academic and laboral world. Although reaching the Earth orbit could seem a distant achievement, a prototype of a Cubesat will be shown in this thesis. It consists on a 10x10x10 cm 3D printed Cubesat, with three embedded magnetic coils as actuators which, interacting with the Earth's magnetic field, will be responsible of controlling the attitude of the satellite.

By the moment, the GranaSAT-I satellite is suppossed to carry a camera as the payload, which should take pictures of the city of Granada. It will also have a very powerful LED matrix that must be seen from Earth surface. These two objectives can only be accomplished if the Cubesat is 3-axis controlled, i.e. it can be deliberately oriented in any direction. In Section 2 it is shown that magnetic actuators give only control on 2 axes at a time; however, the space-varing nature of the Earth's magnetic along the satellite's orbit could allow a 3-axis control.

This work includes theory and simulation of attitude control algorithms. Three important areas play an relevant role in the physical background: Rigid Solid Mechanics give us the equations which govern the motion of the satellite. Electromagnetism describes the interaction between the magnetorquers and the Earth's magnetic field. Finally, Geophysics and Orbital Mechanics knowledge is put in practice, since the magnetic field must be simulated in each point of the satellite orbit.

Additionally, some physical implementation is done as part of this work. This activity is practically inexistent in pure sciencies. Thus, very important skills were learnt. This implementation consisted on writing algorithms in Arduino code, calibrating sensors and basic learning on electronics. This resulted on an operative Cubesat protype, with its own battery and the ability of sending telemetry data.

1.1 Attitude determination

As its name suggests, ADCS involves two differents areas: attitude determination consists on the use of sensors onboard the satellite, and algorithms which read the measured information in order to properly estimate the orientation of the Cubesat with respect to some reference frame. In the case of GranaSAT-I, magnetometers and sun sensors are chosen. Their measurements give an estimation of the direction of the magnetic field and solar vector with respect to the satellite. Both measures are combined in order to estimate the attitude.

Although this issue is not investigated in this work, a functional attitude determination system is supposed to the Cubesat in the simulations.

1.2 Attitude control

On the other hand, attitude control consists not only on stabilizing the Cubesat by canceling its angular speed, but also on controlling its orientation with respect to a chosen reference frame. In this thesis, both objectives are successfully simulated, under certain restrictions.

1.3 ADCS requirements overview

In this subsection, a brief description of the ADCS objetives and phases are given.

1.3.1 Detumble

Once the satellite has been deployed into orbit (orbital insertion), all systems onboard must be desactivated for some minutes, for safety. After that, the satellite 'wakes up' with a random initial angular speed. Its first mission is to stabilize itself, and this can be achieved by the use of the so called B-dot algorithm. The function of this algorithm is to measure the magnetic field variation (that is why it is named B-dot) and using that information to genereate appropiated torque with the magnetorquers.

1.3.2 Nadir pointing

This is the nominal phase. As its name suggests, in this phase the satellite remains with one shape pointing to Nadir, i.e, towards the center of the Earth. A 3-axis control is needed.

1.3.3 Target tracking

The satellite enters on this phase when Granada or other target is in the line of sight. It should be able to point the camera in order to take pictures of the objective.

Summary

Figure 1 synthesizes all these phases, and it also shows information about the conditions which should be given so a phase change is done. These phases are called 'states', because they define the state of the satellite. A finite states machine must be implemented: the satellite onboard computer has to activate the required systems and use them properly according to the situation.

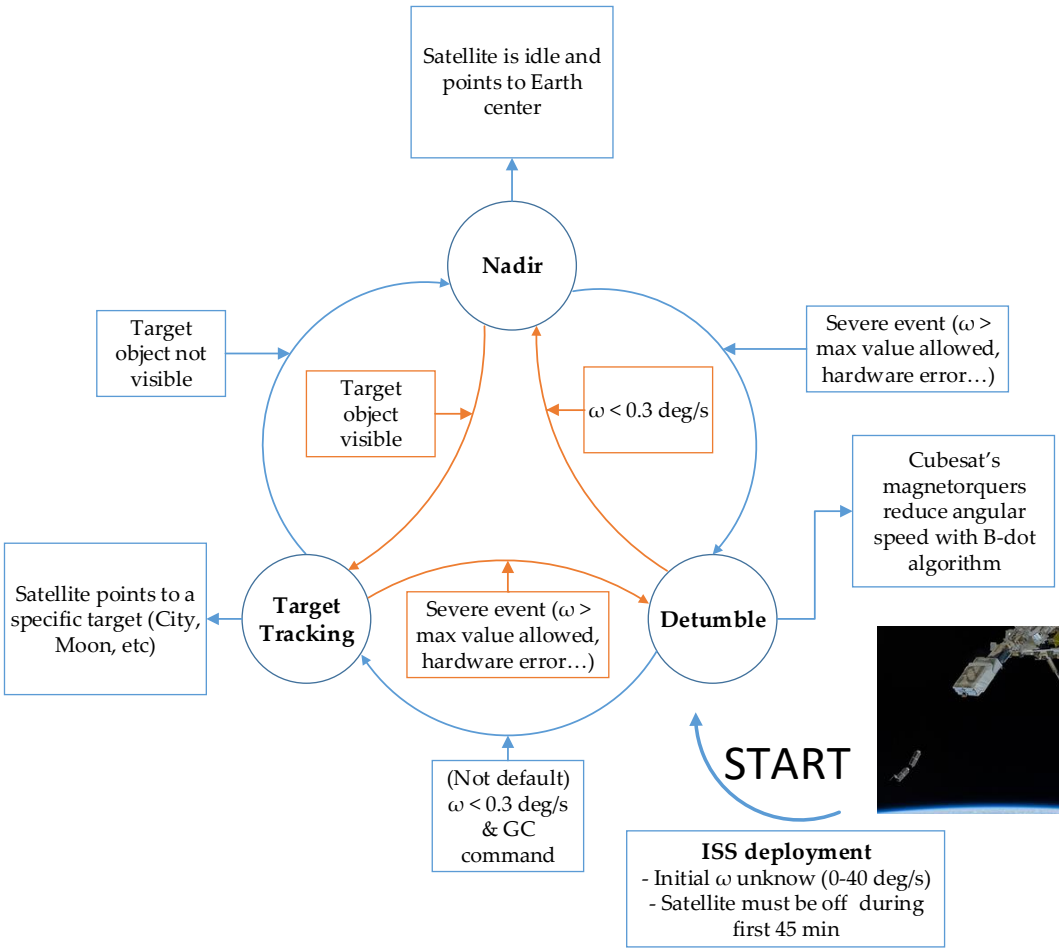

Figure 1: Mission states available for the satellite

2 Magnetorquers

In this section, magnetorquers are introduced. As their name indicates, magnetorquers are magnetic actuators that can generate torque, and thereby, change the attitude of a spacecraft. They are essentially rectangular coils, and we can find two types of them: wire coils and board printed magnetorquers, which are used in this work. If some intensity flows through them, they will generate a magnetic dipole moment which will interact with the magnetic field. Here, we approximate spiral magnetorquers by rectangular coils with an effective area and a number of turns. The magnetic moment produced by one magnetorquer is:

$$\mathbf{m} = iA\hat{\mathbf{n}} \tag{2.1}$$

Being i the intensity, A the area and $\hat{\mathbf{n}}$ the unit surface vector. The total magnetic moment produced by a satellite with N magnetorquers is the sum of the individual magnetic moments:

$$\mathbf{m} = \sum_{n=1}^{N} \mathbf{m}_i \tag{2.2}$$

It is suitable to have three orthogonal magnetorquers, thus the spacecraft can produce a magnetic moment in any direction. The torque produced by the interaction of the magnetic moment and the external magnetic field is [4]:

$$\boldsymbol{\tau} = \mathbf{m} \times \mathbf{B} \tag{2.3}$$

This equation shows the crucial issue concerning to magnetic actuation: the magnetic torque is confined in a plane which is orthogonal to **B**, a vectorial magnitude of which we do not have any control. Thus, we say that magnetic actuation has only two degrees of freedom.

3 Theory

3.1 Attitude parametrization

Attitude or orientation can be parametrized in many ways. Here, we use the notion of rotation matrix and quaternion.

3.1.1 The rotation matrix

A rotation matrix is a $3x3$ matrix which contains information of how a coordinate system is orientated with respect to other. In spite of the fact that it involves 9 terms, it has only 3 degrees of freedom. In fact, a rotation matrix contains unit vectors of one frame expressed in the other frame. They are, then, orthonormal, limitating the matrix to have 3 degrees of freedom.

This rotation matrix, applied to one vector in some frame F_o (noted as $\mathbf{v}^o$), results in the same vector but expressed in other reference frame, F_i. The matrix is noted as $\mathbf{R}_o^i$ (from o to i).

$$\mathbf{v}^i = \mathbf{R}_o^i \mathbf{v}^o \tag{3.1}$$

The inverse of a rotation matrix is its transpose. Thus, it is clear that we can do the inverse transformation in equation 3.1 by using the transpose of $\mathbf{R}_o^i$ ($\mathbf{R}_o^{iT}$).

Another important relation is the composition of rotation matrices. The rotation of a vector expressed in F_c frame to a F_a frame can be written as:

$$\mathbf{v}^a = \mathbf{R}_b^a \mathbf{R}_c^b \mathbf{v}^c \tag{3.2}$$

3.1.2 Quaternions

Quaternions are defined as unit complex numbers with one real part (η); called the scalar part, and three imaginary parts ($\boldsymbol{\epsilon}$); called the vectorial part. They are usally expressed as a column vector:

$$\mathbf{q_o^i} = \begin{bmatrix} \eta \\ \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{bmatrix} \tag{3.3}$$

This quaternion expresses the orientation of F_i with respect to F_o.

Since they have norm one, it only takes three numbers to determine them completely, in the same way as with the rotation matrix. They are widely used because of their singularity-free nature, regarding to kinematic equations. They can be obtained from the rotation matrix elements:

$$\eta = \pm\frac{1}{2}\sqrt{1 + \text{trace}(\mathbf{R})} \tag{3.4}$$

$$\boldsymbol{\epsilon} = \frac{1}{4\eta}\begin{bmatrix} R_{23} & R_{32} \\ R_{31} & R_{13} \\ R_{12} & R_{21} \end{bmatrix} \tag{3.5}$$

If two frames are aligned, the quaternion results to be $\mathbf{q} = [1000]^T$. More properties of quaternions can be found in Appendix D of [5].

3.2 Coordinate frames

Different kinds of reference frames are used in this thesis. It is crucial to understand how they are defined and related. These coordinate frames are:

3.2.1 ECI: Earth-centered inertial frame

The Earth-centered inertial, or the ECI frame, is a non-rotational frame which has its origin in the center of the Earth. It is defined by the unit vectors $\mathbf{x}_{ECI}$, $\mathbf{y}_{ECI}$ and $\mathbf{z}_{ECI}$. The $\mathbf{z}_I$ vector has the same direction as the Earth rotation axis, and it points towards the North Pole. The $\mathbf{x}_{ECI}$ vector points towards the vernal equinox direction. Finally, the $\mathbf{y}_{ECI}$ completes the orthogonal triad, following the righ-hand rule. Vectors expressed in this frame are noted as $\mathbf{v}^i$.

3.2.2 ECEF: Earth-centered fixed frame

The Earth-centered fixed frame, or the ECEF frame, is similar to the ECI frame. The $\mathbf{z}_{ECEF}$ axis is the same as $\mathbf{z}_{ECI}$. However, $\mathbf{x}_{ECEF}$ points towards the 0° latitude 0° longitude Earth surface point. Again, the $\mathbf{y}_{ECEF}$ completes the orthogonal triad. Taking into account this definition, this is not an inertial frame; it rotates with the Earth around the $\mathbf{z}_{ECEF}$ with angular speed $\boldsymbol{\omega}_{ECEF} = 7.2921 \cdot 10^{-5}\ rad/s$. Vectors expressed in this frame are noted as $\mathbf{v}^e$.

3.2.3 Orbit frame

This reference frame has its origin located in the center of mass (CoM) of the satellite, thus it is non-inertial. The $\mathbf{z}_O$ axis points towards the Earth origin (Nadir), the $\mathbf{y}_O$ has the direction opposite to the orbit normal, (i.e. opposite to the angular orbital moment). The vector $\mathbf{x}_O$ complets the set; this vector is parallel to the velocity vector in case of a circular orbit. Vectors expressed in this frame are noted as $\mathbf{v}^o$.

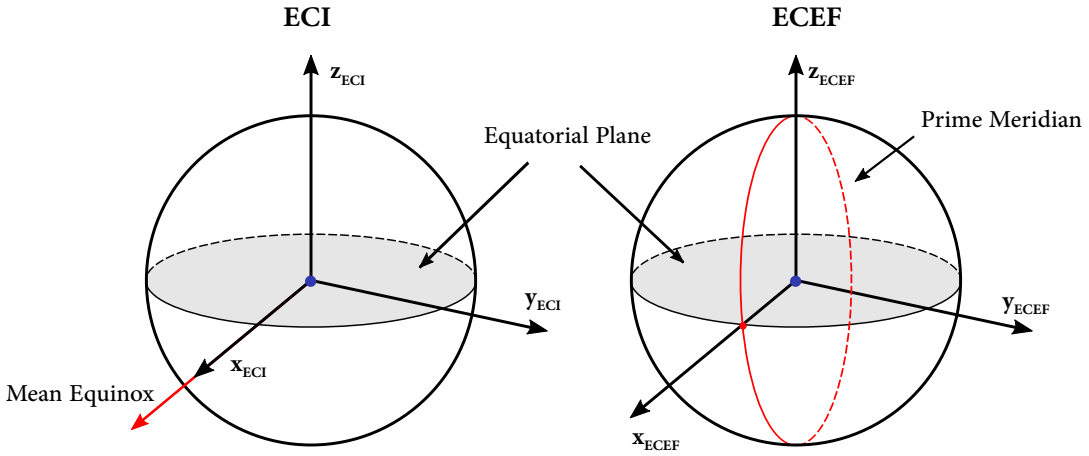

Figure 2: ECI and ECEF frame [6]

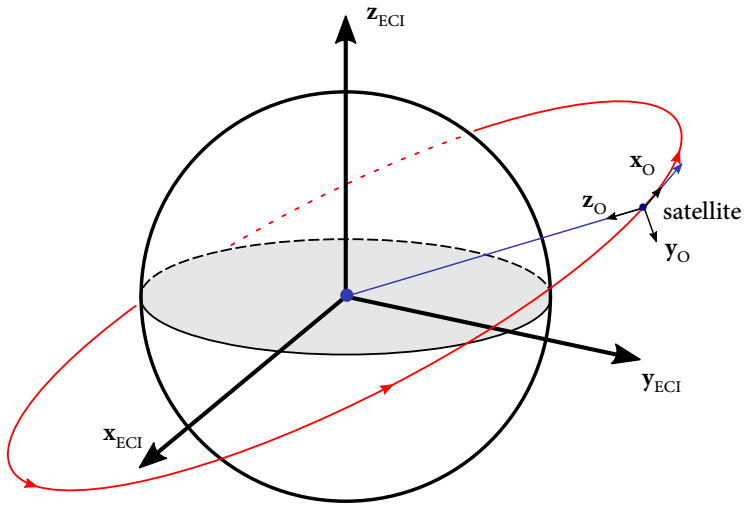

Figure 3: Orbit frame [6]

3.2.4 Body frame

The body frame has the same origin as the orbit frame. However, this coordinate system rotates with the satellite. Their unit vectors, $\mathbf{x}_B$, $\mathbf{y}_B$ and $\mathbf{z}_B$, are chosen in a way that they coincide with the satellite's principal inertia axes. Supposing a homogeneous Cubesat, these vectors will be perpendicular to its faces. Vectors expressed in this frame are noted as $\mathbf{v}^b$.

3.2.5 Target frame

This frame is similar to orbit frame. However, its $\mathbf{z}_T$ vector is aligned with the target. The $\mathbf{y}_T$ vector is normal both to $\mathbf{z}_T$ and the projection of $\mathbf{v}_T$ on $\mathbf{z}_T$. Again, $\mathbf{x}_T$ completes the set as usual. Vectors expressed in this frame are noted as $\mathbf{v}^t$.

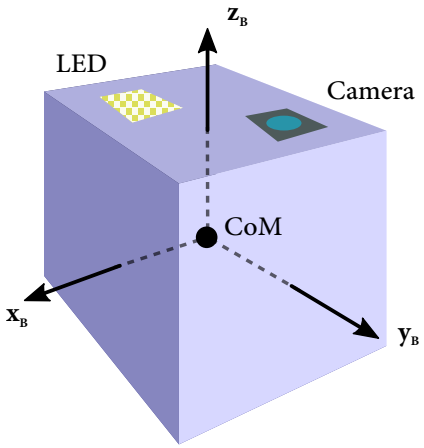

Figure 4: Body frame. Camera and LED matrix are shown on the top face as a posible payload

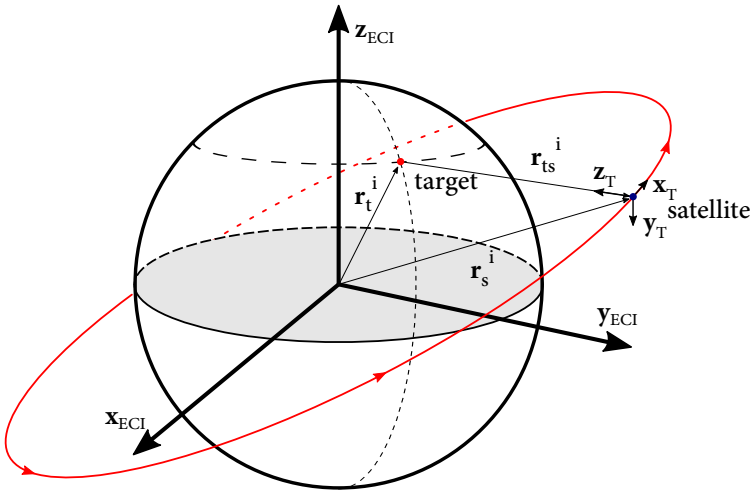

Figure 5: Target frame [6]

3.3 Frame transformations

The transformation between two frames is defined by a rotation matrix, which converts vector from one frame to another. Some important transformations are:

3.3.1 ECI to ECEF

We can get the ECEF frame by rotating the ECI frame around Earth's rotation axis. The angle rotated depends on the time, and the rotation matrix is [6]:

$$\mathbf{R}_I^{ECEF} = \begin{bmatrix} cos(\theta) & -sin(\theta) & 0 \\ sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.6}$$

Where θ is the angle between $\mathbf{x}_{ECI}$ and $\mathbf{x}_{ECEF}$. Since ECEF rotates around $\mathbf{z}_{ECI}$, following Earth's rotation, θ has a periodicity of one day. It can be written as:

$$\theta(t) = \frac{2\pi t}{\mathrm{t_d}} \tag{3.7}$$

Where t is the time since ECI and ECEF had the same orientation, and $\mathrm{t_d}$ is the duration of a siderial day.

3.3.2 Orbit to ECI

Orientation with respect to orbit system can be determined only if the orbital position of the satellite is known. Thus, the rotation matrix required will be function of some kleperian elements. Here, we suppose a circular orbit (although the actual orbit would be an ellipse, it can be approximated by a circle). Then, the rotation matrix is [7]:

$$\mathbf{R}_I^O = \begin{bmatrix} -sucΩ - cucisΩ & -susΩ + cucicΩ & cusi \\ -sisΩ & sicΩ & -ci \\ -cucΩ + sucisΩ & -cusΩ - sucicΩ & -susi \end{bmatrix} \tag{3.8}$$

Where s and c are abbreviations for sin and cos. For example, $-sucΩ$ means $-sin(u)cos(Ω)$. $Ω$ is the right ascension of the ascending node, i is the inclination of the orbit and u is the argument of latitude. The last term is the sum of ν (true anomaly) and ω (argument of periapsis).

3.3.3 Target to ECI

This rotation matrix can be derived from the position vectors of the satellite and the target. Figure 5 shows that:

$$\mathbf{z}_T^i = -\frac{\mathbf{r}_{ts}^i}{\|\mathbf{r}_{ts}^i\|} = -\frac{\mathbf{r}_s^i - \mathbf{r}_t^i}{\|\mathbf{r}_s^i - \mathbf{r}_t^i\|} \tag{3.9}$$

Vector $\mathbf{y}_T$ was defined as normal to $\mathbf{z}_T$ and the normal projection of $\mathbf{v}$ on $\mathbf{z}_T$, so its expression is:

$$\mathbf{y}_T^i = \mathbf{z}_T^i \times \frac{\mathbf{v}^i - (\mathbf{v}^i \cdot \mathbf{z}_T^i)\mathbf{z}_T^i}{\|\mathbf{v}^i - (\mathbf{v}^i \cdot \mathbf{z}_T^i)\mathbf{z}_T^i\|} \tag{3.10}$$

Notice that we are applying Gram-Schmidt to orthogonalize the velocity vector with respect to $\mathbf{z}_T^i$. Finally, completing the triad:

$$\mathbf{x}_T^i = \mathbf{y}_T^i \times \mathbf{z}_T^i \tag{3.11}$$

As mentioned above, the rotation matrix between two frames consists on expressing the unit vector of one frame in other, so the rotation matrix is:

$$\mathbf{R}_t^i = [\mathbf{x}_T^i \ \mathbf{y}_T^i \ \mathbf{z}_T^i] \tag{3.12}$$

3.3.4 Body to any other frame

This transformation is in definitive the notion of attitude determination. Obviously, this rotation matrix depends on the orientation of the satellite at a time given. Sensors onboard combined with physical models of magnetic field and sun position will give an estimation of this matrix.

3.4 Satellite kinematic and dynamics

In this section, equations of kinematic and dynamic of a rigid body are described.

3.4.1 Satellite dynamic

A satellite can be modelled as a rigid solid. Earlier, one relevant non-inertial frame was defined: the body frame. We can describe the relation between external torques, the inertia matrix and the angular speed of the body frame with respect to the ECI frame (which is an inertial frame) thanks to Euler's equations of motion [8]:

$$\boldsymbol{\tau}^b = \mathbf{I}\dot{\boldsymbol{\omega}}_{ib}^b + \boldsymbol{\omega}_{ib}^b \times (\mathbf{I}\boldsymbol{\omega}_{ib}^b) \tag{3.13}$$

Where $\mathbf{I}$ is the inertia matrix in the body frame (constant). $\boldsymbol{\tau}^b$ is the total torque acting on the satellite. It is not only the sum of the control torque provided by magnetorquers ($\boldsymbol{\tau}_m$), since there exist other environmental torques (disturbances). Later, we will analyse two important disturbances: gravity gradient torque ($\boldsymbol{\tau}_g$) and aerodynamic torque ($\boldsymbol{\tau}_a$). Total torque results to be:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_m + \boldsymbol{\tau}_g + \boldsymbol{\tau}_a \tag{3.14}$$

3.4.2 Satellite kinematic

Kinematic equations give us the relation between angular speed and attitude rate change. Expressing attitude by quaternions, they are [8]:

$$\dot{\eta} = -\frac{1}{2}\boldsymbol{\epsilon}^T\boldsymbol{\omega}_{ib}^b \tag{3.15}$$

$$\dot{\boldsymbol{\epsilon}} = \frac{1}{2}(\eta\mathbf{I}_{3x3} + \mathbf{S}(\boldsymbol{\epsilon}))\boldsymbol{\omega}_{ib}^b \tag{3.16}$$

These equations give the relation for angular speed of the body frame with respect to the ECI frame. However, the satellite should be aligned with the orbit or target frames sometimes. In these cases, we are more interested in the angular speed with respect to those frames: $\boldsymbol{\omega}_{ob}^b$ and $\boldsymbol{\omega}_{tb}^b$. The former is easy to describe, since the angular speed of the orbit frame with respect to the ECI frame is the orbital angular speed (ω_0), around -$\boldsymbol{y}_O$ axis. If we express that angular speed in the body frame, we can express $\boldsymbol{\omega}_{ob}^b$ as:

$$\boldsymbol{\omega}_{ob}^b = \boldsymbol{\omega}_{ib}^b - \boldsymbol{\omega}_{io}^b = \boldsymbol{\omega}_{ib}^b + \omega_0\mathbf{y}_O^b \tag{3.17}$$

Source	Dependence on altitude	Dominant in
Aerodynamic	$e^{-\alpha x}$	Altitudes below 500
Magnetic	$1/r^3$	500 km to 35000 km
Gravity gradient	$1/r^3$	500 km to 35000 km
Solar radiation	Independent	Interplanetary space
Micrometeorietes	Depends on region	Normaly negligible

Table 1: Disturbance torques

The angular speed with respect to the ECI frame is given by gyroscopes. The orbital speed ω_0 is derived from Newton's second law and his universal gravitational law:

$$\omega_0 = \sqrt{\frac{GM_{Earth}}{r^3}} \tag{3.18}$$

The latter could be described analytacally, but it is simplier to use the definition of angular speed in terms of the rotation matrix [8]:

$$\mathbf{S}(\boldsymbol{\omega}_{it}^i) = \dot{\boldsymbol{R}}_t^i (\boldsymbol{R}_t^i)^T \tag{3.19}$$

Where $\boldsymbol{\omega}_{it}$ is the angular velocity of frame t relative to frame i

Kinematic equations expressed with angular variables (Euler angles, rotation matrices...) contains trigonometric expressions that become infinite for some rotations (they are called *kinematic singularities*). Their quaternion representation has no associated kinematic singularities. This is the main reason for the usage of quaternions.

3.5 Environment

3.5.1 Disturbance torques

Disturbance torques are external torques produced by environmental forces that act on a spacecraft. Their sources are summarized in Table 1.

The Cubesat modeled here will be in a ISS-like orbit, i.e. below 500 km. Hence, predominant torques are of type aerodynamic, magnetic and gravitational. Magnetic torques are due to interaction between Earth's magnetic field and the satellite's residual magnetization. Since our satellite will produce its own magnetic field (stronger than any residual one), this torque is discarded. Solar radiation torque and torque due to micrometeorites are ignored too.

3.5.2 Gravity gradient torque

Earth's gravitational force varies over any spacecraft, since it is a function of distance. This results on a non-zero torque, which is stronger for low altitudes and nonsymmetrical satellites. The expression for this torque is derived in [5]:

$$\boldsymbol{\tau}_g^b = 3\frac{GM_{Earth}}{r}[\mathbf{z}_O^b \times (\mathbf{I} \cdot \mathbf{z}_O^b)] \tag{3.20}$$

Where r is the distance from Earth center and $\mathbf{I}$ is the inertia matrix in the body frame. This torque tends to align elongated bodies in the direction of an Earth radius, i.e, the direction of the minimun moment of inertia.

3.5.3 Aerodynamic torque

As its name suggests, aerodynamic torque is caused by interaction between upper atmospheric particles and the spacecraft. The aerodynamic force (in the body frame) acting on a surface can be found in [9]:

$$\mathbf{f}_a^b = -\frac{1}{2}\rho \mathrm{v}^2 C_D S(\hat{\mathbf{v}}^b \cdot \hat{\mathbf{n}}^b) \cdot \hat{\mathbf{v}}^b \tag{3.21}$$

Here, ρ is the atmospheric density, $\mathbf{v}^b$ is the satellite's velocity, $\hat{\mathbf{n}}$ is the unit vector normal to the surface, C_D is the drag coefficient (set to 2.2 based on [10]), and S is the area.

The aerodynamic torque can be modeled as the vectorial sum of forces acting on the Cubesat's surfaces which are facing the 'wind'. The torque results to be:

$$\boldsymbol{\tau}_a^b = \sum_{n=1}^{6} \mathbf{r}_{pf}^b \times \mathbf{f}_{a,n}^b \cdot max(\hat{\mathbf{v}} \cdot \hat{\mathbf{n}}, 0) \tag{3.22}$$

The last term ensures that only the faces which are faced to the atmospheric flow are taking into account.

3.5.4 Earth magnetic field

The Earth's magnetic field must be modeled in order to simulate the actuation of the magnetorquers. One important global model is the International Geomagnetic Reference Field (IGRF) [11]. This model computes the magnetic field as the negative gradient of the scalar potential V (in spherical coordinates):

$$\mathbf{B} = -\nabla V \tag{3.23}$$

$$V(r,\theta,\phi,t) = a\sum_{n=1}^{N}\sum_{m=0}^{n}(\frac{a}{r})^{n+1}[g_n^m(t)cos(m\phi) + h_n^m(t)sin(m\phi)]P_n^m(cos(\theta)) \tag{3.24}$$

Where $a = 6371.2\ km$ and $N = 13$. Time-varing coefficients $g_n^m(t)$ and $h_n^m(t)$ are updated each 5 years. Code for this model is provided in [11] in some programming languages, and a MATLAB subroutine is found in the program forum [12].

3.6 Lyapunov stability

In this subsection, a brief study of stability based on Lyapunov theory is done, which is necessary to study the effectiveness of control algorithms. This mathematical background is further described in [13] and [14], and based on [15] and [16]. Next paragraphs are extracted from these two books.

3.6.1 Stability (non-autonomous systems)

Consider the non-autonomous system

$$\dot{x} = f(x,t) \tag{3.25}$$

where $f : [0,\infty) \times D \to R^n$ is piecewise continous in t and locally Lipschitz [1] in x on $[0,\infty) \times D$, and $D \subset R^n$ is a domain that contains the origin $x = 0$. The origin is an equilibrium point for 3.25 at $t = 0$ if

$$f(t,0) = 0, \forall t \geq 0 \tag{3.26}$$

Following definitions are extracted from [15], and they are mathematical definitions for intuitive concepts as stability:

Definition 1. The equilibrium point $x = 0$ of 3.25 is

1. *Stable* if, for each $\epsilon > 0$, there exists $\delta = \delta(\epsilon, t_0) > 0$ such that

$$\|x(t_0)\| < \delta \implies \|x(t)\| < \epsilon, \forall t \geq t_0 \geq 0 \tag{3.27}$$

2. *Uniformly stable* if, for each $\epsilon > 0$, there is $\delta = \delta(\epsilon) > 0$, independent of t_0, such that 3.27 is satisfied.

3. *Unstable* if it is not stable.

4. *Asymptotically* stable if it is stable and there is a positive constant $c = c(t_0)$ such that $x(t) \to 0$ as $t \to \infty$, for all $\|x(t_0)\| < c$.

5. *Uniformly asymptotically stable* if it is *uniformly stable* and there is a positive constant c, independent of t_0, such that for all $\|x(t_0)\| < c$, $x(t) \to 0$ as $t \to \infty$, *uniformly* in t_0; that is, for each $\eta > 0$, there is $T = T(\eta) > 0$ such that

$$\|x(t)\| < \eta, \forall t \geq t_0 + T(\eta), \forall \|x(t_0)\| < c \tag{3.28}$$

6. *Globally uniformly asymptotically stable* if it is *uniformly stable*, $\delta(\epsilon)$ can be chosen to satisfy $\lim_{\epsilon \to \infty} \delta(\epsilon) = \infty$, and, for each pair of positive numbers η and c, there is $T = T(\eta, c) > 0$ such that

$$\|x(t)\| < \eta, \forall t \geq t_0 + T(\eta, c), \forall \|x(t_0)\| < c \tag{3.29}$$

Figure 6 give an idea about a stable and asymptotically stable point. Regarding to the former, the system starts inside a ball of radius ϵ, and then lives forever inside another ball of radius δ. Asymptotically stability means a stronger concept of stability, since the system converges exactly to the equilibrium point, provided that it started inside a ball of radius c.

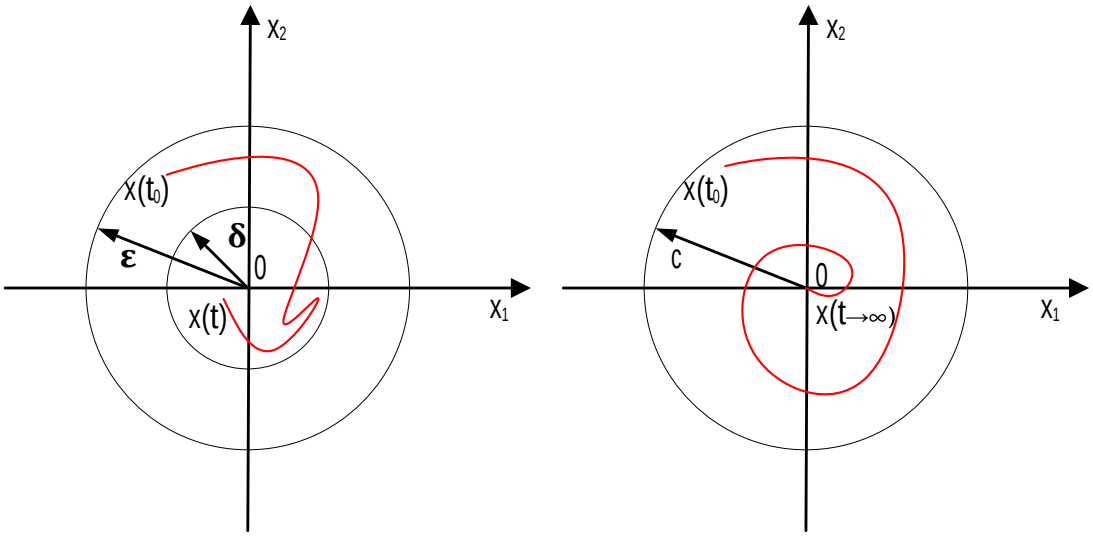

Figure 6: Stable equilibrium (left) and Asymptotically Stable equilibrium (right)

The word 'uniformly' implies that ϵ and c are independent of initial time t_0. Finally, for a 'globally uniformly asymptotically stable' point, it does not matter where the state stars inside the whole domain D: it will converge to equilibrium point $x = 0$.

Next important theorem is found in [16]:

Theorem 1. The equilibrium $x = 0$ of the system 3.25 is uniformly stable if there exist a C^1, decrescent, locally positive definite function $V : R_+ \times R^n \to R$ and a constant $r > 0$ such that

$$\dot{V}(t, x) \leq 0, \forall t \geq 0, \forall x \in B_r \text{ (ball of radius } r) \tag{3.30}$$

The function V is called a *Lyapunov function*. It is also called *Lyapunov function candidate* when only mets conditions required for Theorem 1, but not equation 3.30.

4 Control

In this section, three control laws are investigated. Their main goal is to achive requirements stablished in Section 1.3. The first of them is the so-called B-dot algorithm, which makes use of the measured magnetic field variation onboard the satellite to produce a magnetic moment whose resulting torque stabilizes the satellite. This way, magnetometers are the only sensors needed. We will see that not only its formulation will convice us of its effectiveness, but also a Lyapunov analysis based on Section 3.6. A extensive analysis of this control law can be found in [18].

[1] From [17] A function $f : A \subset R^n \to R^m$ is locally Lipschitz if for each $x_0 \in A$, there exist constants $M > 0$ and $\delta_0 > 0$ such that $\|x - x_0\| < \delta_0 - \implies \|f(x) - f(x_0)\| \leq M\|x - x_0\|$

Regarding to Nadir pointing and target tracking, linear and non-linear controllers can be implemented. Linear controllers are based on a linearization of equations of motion, disturbances and actuators; this is not done here. These controller are useful, for example, when using linear Kalman filters, as in [19]. On the other hand, non-linear controllers do not do linear assumptions. This fact made them more suitable when the satellite's dynamic is far from being linear.

4.1 B-dot algorithm

Magnetic moment produced under B-dot actuation is:

$$\mathbf{m} = -k\frac{1}{\|\mathbf{B}\|^2}\dot{\mathbf{B}} \tag{4.1}$$

Where k is a positive constant gain. Torque produced by its interaction with the external field is, accordingly to Section 2:

$$\boldsymbol{\tau}_m^b = \mathbf{m}^b \times \mathbf{B}^b \tag{4.2}$$

The magnetic field variation in the body frame provides information about angular speed. In [18] is shown that, for relative high angular speeds (with respect to orbital angular speed, i.e. after orbit insertion), $\dot{\mathbf{B}}$ can be written as:

$$\dot{\mathbf{B}}^b \approx \mathbf{B}^b \times \boldsymbol{\omega}_{ib}^b \tag{4.3}$$

Equation 4.3 implies that $\dot{\boldsymbol{B}}$ and $\mathbf{B}$ are orthogonal, so $\mathbf{m}$ defined by equation 4.1 is orthogonal to magnetic field too, and then the produced torque (equation 4.2) is maximum. Moreover, Figure 7 shows that this torque has a component with opposite sign than angular speed. Thus, the satellite is being stabilized while this algorithm is acting.

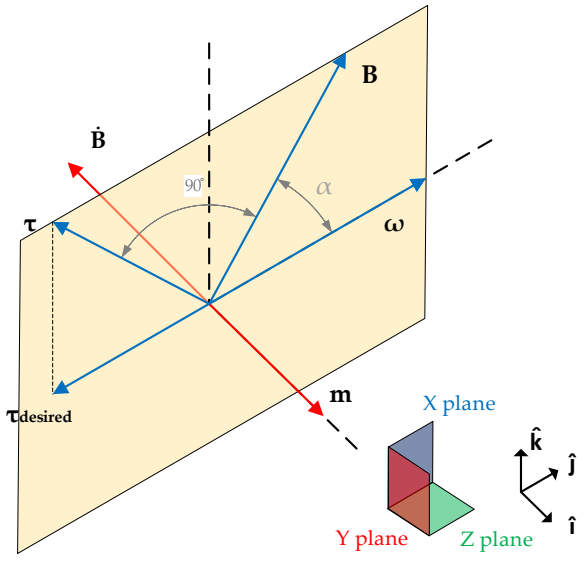

Figure 7: Vectors involved in B-dot algorithm (vector with the same color are coplanar)

In [18], a Lyapunov function (remember Theorem 1) involving ECI-body quaternion is found, with $\mathbf{q}_i^b = [1000]^T$ as a stable point.

4.2 Nadir pointing: non-linear controller

Nadir pointing means that the body frame must be aligned with the orbit frame. In this situation, it is clear that $\boldsymbol{\omega}_{ob}^b = [0\ 0\ 0]^T$ and $\mathbf{q}_b^o = [1000]^T$. The controller should then reduce this angular speed and have information about the orientation with respect to the orbit frame; this means that attitude determination is required.

A very intuitive controller is investigated in [14], [20]. It proposes an expression for magnetic moment as follows:

$$\mathbf{m}^b = \frac{1}{\|\mathbf{B}^b\|}[\alpha(\mathbf{B}^b \times \boldsymbol{\epsilon}) - \beta(\mathbf{B}^b \times \boldsymbol{\omega}_{ob}^b)] \tag{4.4}$$

Where α and β are positive constant gains. This magnetic moment is cross-multplied by magnetic field in order to project it to a plane in which is perpendicular to it, since only the component of $\mathbf{m}$ parallel to $\mathbf{B}$ produces non zero torque (this can be deduced from equation 2.3).

The second term of this magnetic moment will produce a torque which is proportional to angular speed, and with opposite sign. This term is analysed in [14], and it is based on energy considerations, which are summarized here.

The main goal is to find a Lyapunov function that could describe the system, and one classical candidate is the total energy of the satellite. Its expression is derived in Section 2 of [14], and it is the sum of three terms:

$$E = E_{kin} + E_{gg} + E_{gyro} \tag{4.5}$$

The first term is the kinetic energy, the second one is the potential energy due to the gravity gradient and the last one is another potential energy due to revolution of the satellite around the Earth. This equation results to be (Section 7 of [14]):

$$E = \frac{1}{2}\boldsymbol{\omega}_{ob}^{bT}\mathbf{I}\boldsymbol{\omega}_{ob} + \frac{3}{2}\omega_0^2(\mathbf{z}_O^{bT}\mathbf{I}\mathbf{z}_O^b - I_z) + \frac{1}{2}\omega_0^2(I_x - \mathbf{x}_O^{bT}\mathbf{I}\mathbf{x}_O^b) \tag{4.6}$$

This function is positive definite if:

$$I_x > I_y > I_z \tag{4.7}$$

Since this is our Lyapunov candidate function, we need to investigate its derivative too, which tacking into account Euler's equations 3.13, results to be:

$$\dot{E} = \boldsymbol{\omega}_{ob}^{bT}\boldsymbol{\tau}_m^b \tag{4.8}$$

Inserting the second term of controller 4.4 in equation 4.8 gives:

$$\dot{E} = -\beta(\mathbf{B}^b \times \boldsymbol{\omega}_{ob}^b)^T(\mathbf{B}^b \times \boldsymbol{\omega}_{ob}^b) \tag{4.9}$$

This means that the derivative of the energy is negative semidefinite, so the total energy is a Lyapunov function. This is a sufficient condition for the existence of stable equilibrium points (theorem 1). However, by the use of an extension of Lyapunov stability theory based on periodic systems, (which is not explained in this thesis), in [14] is found that there exist four equilibria points that are locally uniformly asymptotically stable:

$$(\boldsymbol{\omega}_{ob}^{b}, \mathbf{z}_{O}^{b}, \mathbf{x}_{O}^{b}) \ : \ (\mathbf{0}, \pm\mathbf{z}_{O}^{o}, \pm\mathbf{x}_{O}^{o}) \tag{4.10}$$

This equation shows that we can align the body and orbit frames in four ways. Attending to the sign preceding $\mathbf{z}_O^o$ and $\mathbf{x}_O^o$, we have the combinations: $++$, $+-$, $-+$ and $--$. In the first case, $\mathbf{z}_O^o$ and $\mathbf{x}_O^o$ are the same in the body and orbit frames; in the second case, $\mathbf{z}_O^o$ is the same but $\mathbf{x}_O^o$ is rotated 180 degrees... etc.

If our satellite has only one camera in one face, we need to ensure that this face will be pointing to the Earth. For this reason, controller 4.4 has another term which involves attitude information. With this term, [14] proves that the equilibrium point:

$$(\boldsymbol{\omega}_{ob}^{b}, \mathbf{z}_{O}^{b}, \mathbf{x}_{O}^{b}) \ : \ (\mathbf{0}, \mathbf{z}_{O}^{o}, \mathbf{x}_{O}^{o}) \tag{4.11}$$

is globally asymptotically stable, with a time-varing gain $\alpha(t)$. Here we use a constant gain α, and simulations prove that this equilibrium point is reached under some conditions.

4.3 Target pointing: non-linear controller

It have been discussed that there exist stable equilibrium points that allow a Nadir pointing controller. This is in fact due to the 'assistance' of potential energy caused by the Earth's gravitational field. In this case, however, the satellite must point to a desired location on Earth surface. Here, we propose the same control law 4.4, but substituting angular speed body-orbit with body-target, and the same with the attitude term. Remember that this angular speed was estimated by equation 3.19, and the rotation matrix $\mathbf{R}_b^t$ can be obtained as a composite rotation:

$$\mathbf{R}_b^t = \mathbf{R}_i^t \mathbf{R}_b^i \tag{4.12}$$

The first matrix is derived in section 3.3.3, and the second one is obtained by the Attitude Determination System.

Finally, the controller law is:

$$\mathbf{m}^b = \frac{1}{\|\mathbf{B}^b\|}[\alpha(\mathbf{B}^b \times \boldsymbol{\epsilon}) - \beta(\mathbf{B}^b \times \boldsymbol{\omega}_{tb}^b)] \tag{4.13}$$

Where $\boldsymbol{\epsilon}$ is the vectorial part of $\mathbf{q}_t^b$.

5 Results

Controllers defined in previous section are tested in a MATLAB simulator. This simulator comprises three main parts:

1. **Orbit propagation**: the trajectory of the satellite is propagated via SGP4 algorithm [21]. This algorithm is one of a total of five mathematical models used to calculate orbital positions and velocities of spacecrafts or debris, called *Simplified Perturbations Model*. A MATLAB subroutine have been created, which receives a TLE [2] and a time interval as inputs. The output consists on the position and velocity of the satellite during the time specified, as well as some keplerian elements. The core of this program is the SGP4 source code found in [22]. All the vectors generated in this step are in the ECI frame. The temporal resolution of the calculated parameters is 1 second.

2. **Magnetic field calculation**: the magnetic field is calculated with a MATLAB subroutine [11] and then transformated to ECI coordinates. These magnetic fields values are used in the next step.

3. **Attitude simulation**: this is the most important part of the simulator. The equations of rotation of the Cubesat are integrated (the integrator used is part of a larger open source MATLAB toolbox [23], which code can be download at [24]). This integrator has been modified in order to use the appropiate quaternion when integrating the kinematic equations. In each iteration, the gravity gradient and aerodynamic torque are calculated, as well as the magnetic moment generated by magnetorquers, which depends on the algorithm chosen. The maximum magnetic moment producible is a function of the magnetorquers design (equation 2.2) . The control torque is finally computed with equation 4.2. Moreover, power and energy consumed are computed. Finally, it has to be added that an artificial gaussian error is applied to attitude parameters (the body-ECI quaternion) and to measured angular velocities, since they are estimated with sensors in real life. The temporal resolution of the attitude simulation is 0.1 seconds (orbital and magnetic field values are updated each 10 iterations, due to their lower resolution).

Physical properties of the simulated Cubesat are summarized in Table 2.

Parameter	Value
Mass (kg)	1
Size (cm)	1U (10x10x10)
Coil resistances (Ω)	[50 50 50]
Intensity limits (mA)	[100 100 100]
Coil areas (cm^2)	[88 88 88]
Inertia matrix (m^2kg)	diag([0.0018 0.0017 0.0015])

Table 2: Cubesat properties

The inertia matrix meet the condition 4.7. In real life, the satellite should be manufactured taking into account this requirement. This is straight forward for 2U or 3U

[2] TLE: Two-Line Element sets are text files which contain information about a spacecraft and its orbit, used by SGP4 to propagate the orbit over time

satellites, but in this case, internal masses inside the 1U must be non-uniformly disposed.

The orbit was propagated with a ISS TLE from March 16, 2018, with an orbital inclination of 51 degrees and an altitude of approximately 420 km of altitude.

5.1 Detumble

B-dot algorithm is tested over time. In this case, equations of kinematic (3.15) are integrated in the form body-ECI. Figures 8 and 9 show the angular speed and the energy consumption over time. Initial conditions are summarized in Table 3.

Parameter	Value
$\boldsymbol{\omega}_{ib}^{b}$ (deg/s)	$[20 \ -7 \ 15]^T$
Attitude ($\mathbf{q}_i^b$)	$[0.78 \ -0.42 \ 0.07 \ 0.45]^T$
k $(\mathrm{A} \cdot \mathrm{m}^2 \cdot \mathrm{T})$	1E-5

Table 3: B-dot initial conditions

5.2 Nadir pointing

In this case, simulations were performed first with the gravity gradient torque as the only disturbance, and then the aerodynamic torque contribution were added. The vectorial part of the attitude quaternion of the rotation between the body and orbit frames is plotted, as well as the energy consumption. Kinematic equations are expressed in the body-orbit form.

5.2.1 Gravity gradient torque only

Figures 10 and 11 show the vectorial part of the quaternion attitude and the energy consumption, with only gravity gradient torque as disturbance. Initial conditions are:

Parameter	Value
$\boldsymbol{\omega}_{ob}^{b}$ (deg/s)	$[1 \ -0.5 \ 0.3]^T$
Attitude ($\mathbf{q}_o^b$)	$[0.96 \ -0.09 \ 0.01 \ 0.26]^T$
α $(\mathrm{A} \cdot \mathrm{m})$	0.0001
β $(\mathrm{A} \cdot \mathrm{m} \cdot \mathrm{s})$	0.1

Table 4: Nadir pointing initial conditions (only gravity gradient torque)

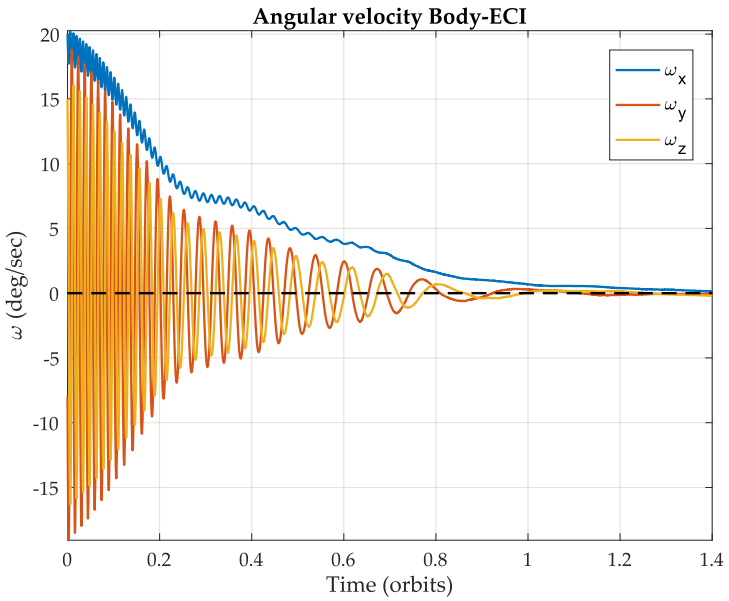

Figure 8: Cubesat angular speed with B-dot.

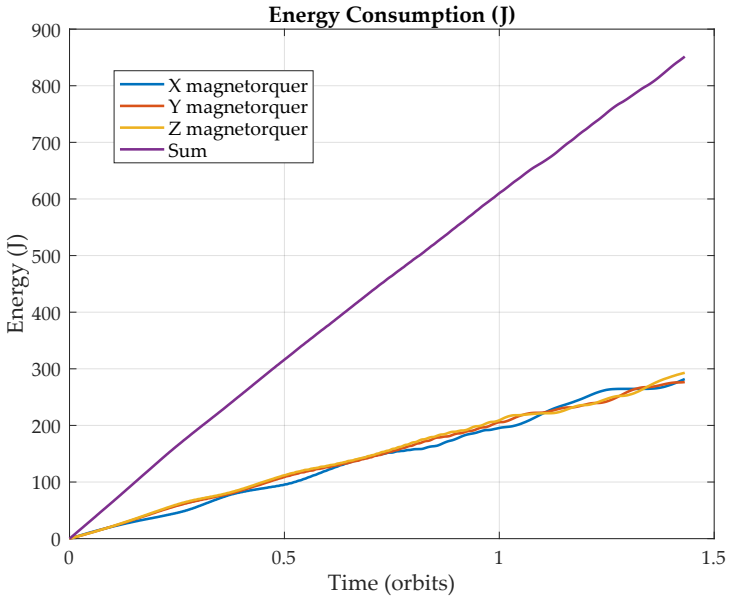

Figure 9: Cubest energy consumption (accumulated) with B-dot

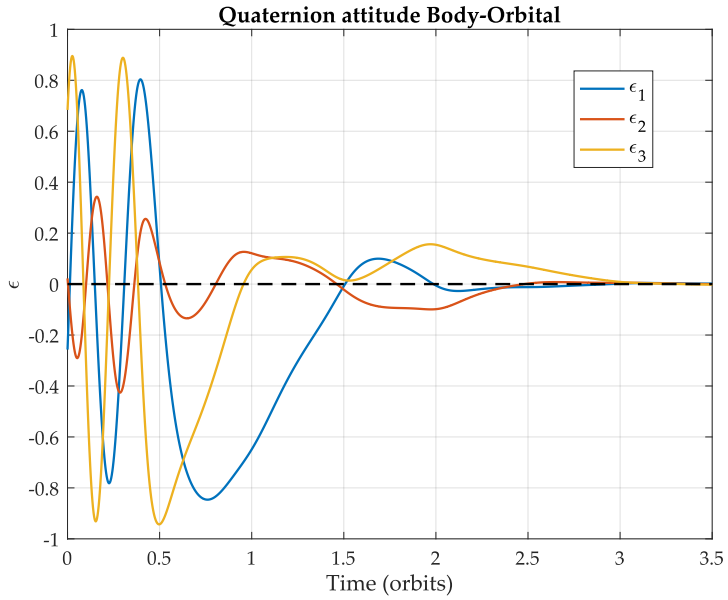

Figure 10: Vectorial part of the quaternion Body-Orbital (only gravity gradient torque)

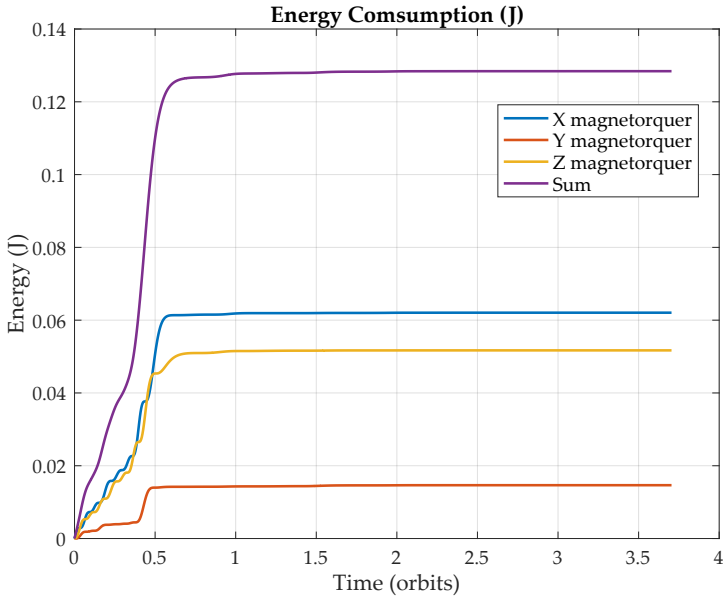

Figure 11: Cubest energy consumption (accumulated) with Nadir pointing (only gravity gradient torque)

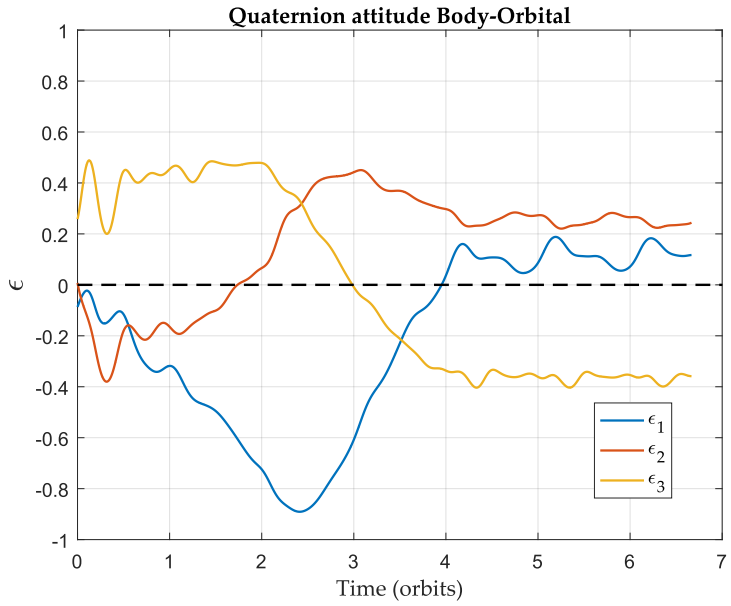

Figure 12: Vectorial part of the quaternion Body-Orbital (with aerodynamic torque)

5.2.2 Gravity gradient and aerodynamic torque

Figures 12 and 13 shows same simulations but adding aerodynamic torque. In this case, the angle between a camera placed on the +Z face of the Cubesat and the Nadir direction is plotted too (Figure 14). Initial conditions are:

Parameter	Value
$\boldsymbol{\omega}_{ob}^{b}$ (deg/s)	$[0.01 \ -0.05 \ 0.03]^T$
Attitude ($\mathbf{q}_{o}^{b}$)	$[0.96 \ -0.09 \ 0.01 \ 0.26]^T$
α $(\mathrm{A} \cdot \mathrm{m})$	0.00008
β $(\mathrm{A} \cdot \mathrm{m} \cdot \mathrm{s})$	0.12

Table 5: Nadir pointing initial conditions (with aerodynamic torque)

5.3 Target tracking

In this case, simulations were performed first with gravity gradient torque as the only disturbance, and then the aerodynamic torque contribution were added. The vectorial part of the attitude quaternion between body and target frame is plotted, as well as the energy consumption. kinematic equations are expressed in the body-target form. The target coordinates are longitude 45 degrees East, latitude 30 degrees North.

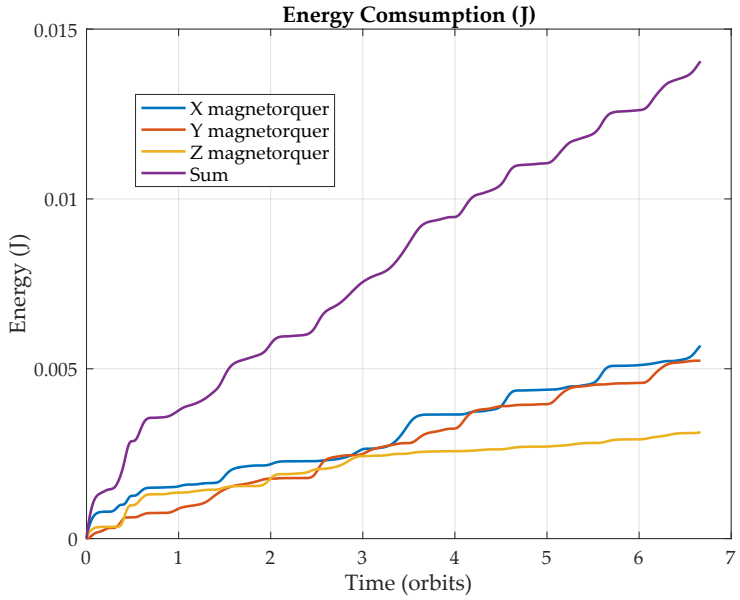

Figure 13: Cubest energy consumption (accumulated) with Nadir pointing (with aerodynamic torque)

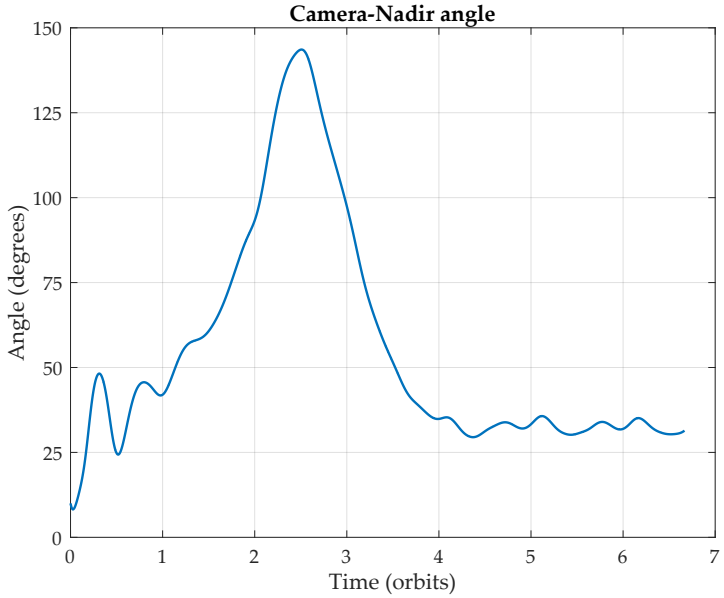

Figure 14: Angle between $\mathbf{z}_B^b$ and $\mathbf{z}_O^b$ (with aerodynamic torque)

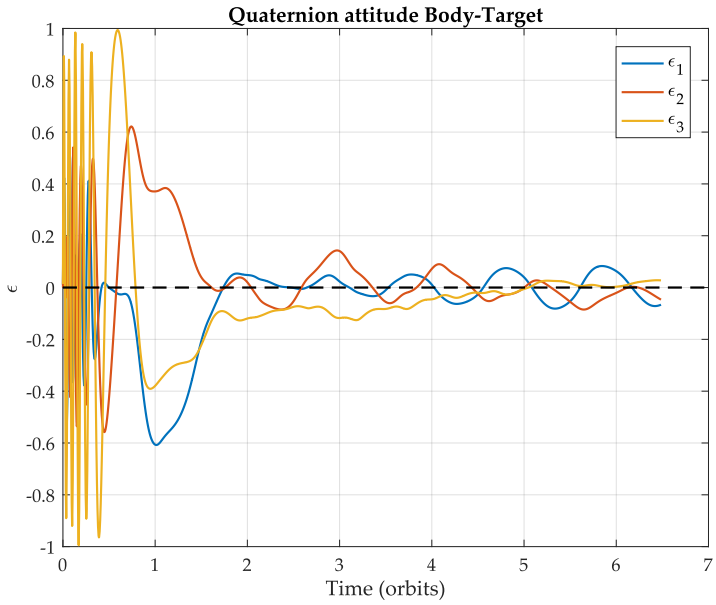

Figure 15: Vectorial part of the quaternion Body-Target (only gravity gradient torque)

5.3.1 Gravity gradient torque only

Figures 15 and 16 show the vectorial part of the quaternion attitude and the energy consumption, with only gravity gradient torque as disturbance. Initial conditions are:

Parameter	Value
$\boldsymbol{\omega}_{tb}^{b}$ (deg/s)	$[0.01 \ -0.05 \ 0.03]^T$
Attitude ($\mathbf{q}_t^b$)	$[0.96 \ -0.09 \ 0.01 \ 0.26]^T$
α $(\mathrm{A}\cdot\mathrm{m})$	0.0001
β $(\mathrm{A}\cdot\mathrm{m}\cdot\mathrm{s})$	0.1

Table 6: Target tracking initial conditions (only gravity gradient torque)

5.3.2 Gravity gradient and aerodynamic torque

Figures 17 and 18 show the vectorial part of the quaternion attitude and the energy consumption, with only gravity gradient torque as disturbance. The intensity in each magnetorquers is also plotted versus time in Figure 19. Initial conditions are:

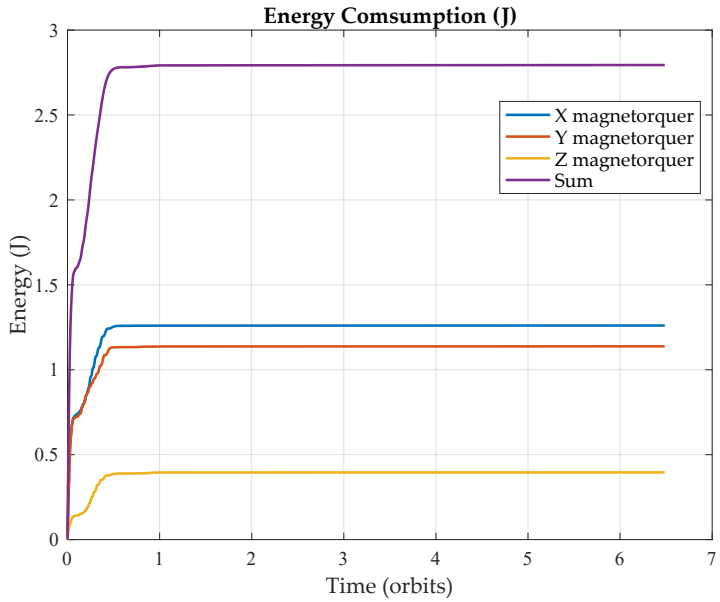

Figure 16: Cubest energy consumption (accumulated) with Target tracking (with aerodynamic torque)

Parameter	Value
$\boldsymbol{\omega}_{tb}^{b}$ (deg/s)	$[0.01 \ -0.05 \ 0.03]^T$
Attitude ($\mathbf{q}_{t}^{b}$)	$[0.96 \ -0.09 \ 0.01 \ 0.26]^T$
α $(\mathrm{A} \cdot \mathrm{m})$	0.00008
β $(\mathrm{A} \cdot \mathrm{m} \cdot \mathrm{s})$	0.12

Table 7: Target tracking initial conditions (with aerodynamic torque)

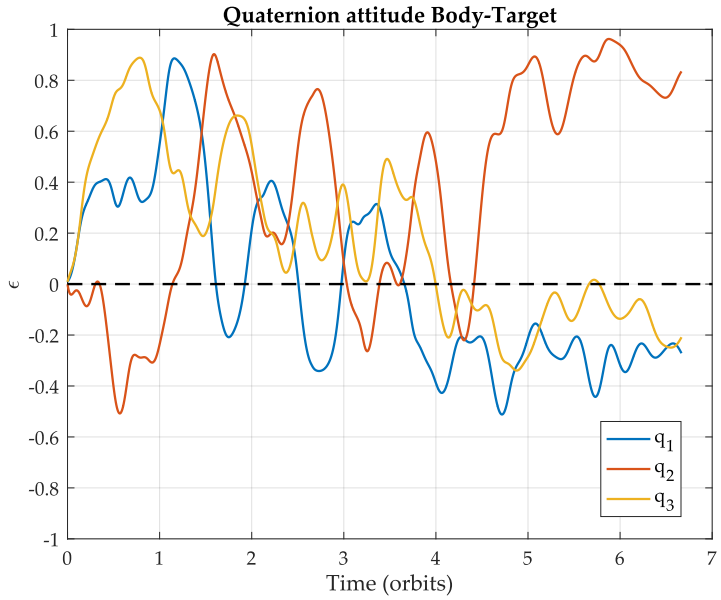

Figure 17: Vectorial part of the quaternion Body-Target (with aerodynamic torque)

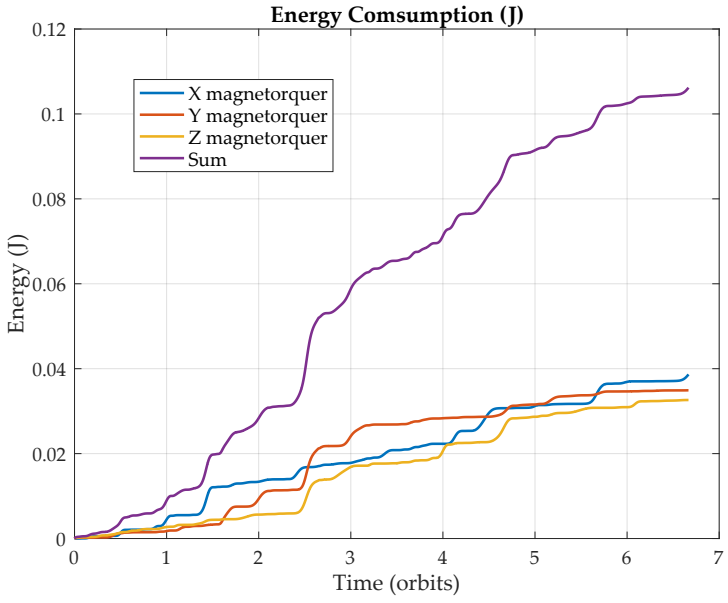

Figure 18: Cubest energy (accumulated) consumption with Target tracking (with aerodynamic torque)

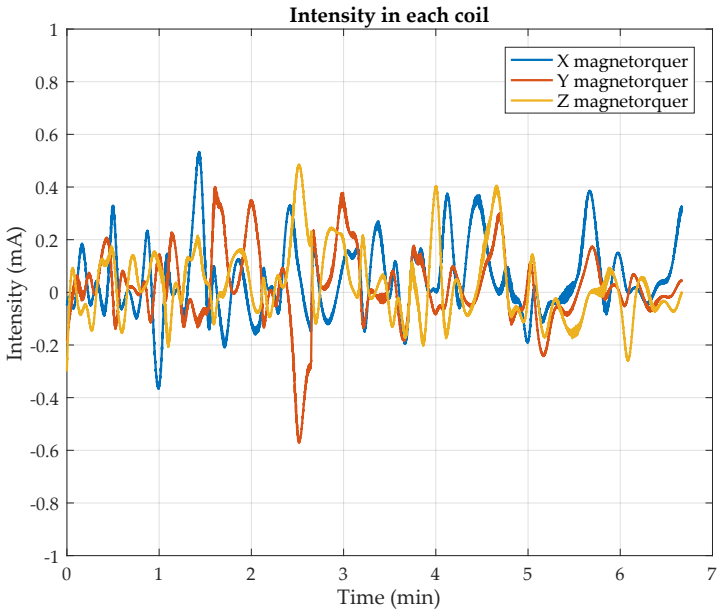

Figure 19: Intensity in each magnetorquer (with aerodynamic torque)

6 Experimental setup

Parallel to the investigation and summarization of attitude control theory and the MATLAB simulator implementation, a physical prototype of a 1U-Cubesat has been developed and partially tested, with the aid of GranaSAT team and resources from its laboratory, as well as the company DHV.

6.1 Cubesat prototype

The prototype's structure is formed by 3D-printed components. It has three side panels which are basically PCBs (printed circuit boards) with the printed spiral magnetorquers. Photovoltaic cells should be placed on these panels, but they are not needed for ADCS tests. (Figure 20).

Inside the Cubesat, there is an Arduino board connected to an EPS (electronic power system) designed by a GranaSAT member [25] (Figure 21). This electronic device is the interface between the Arduino and the rest of the satellite, which consists on (Figures 21 and 22):

1. **Inertial Measurement Unit LSM9DS0 (IMU)**: this electronic device includes a 3-axis magnetometer, a 3-axis accelerometer and a 3-axis gyroscope. It has also a thermometer.
2. **Tiny RTC DS1307 clock**: this module provides the hour to the OBC.
3. **Wireless serial module**: a serial transmitter which uses an antenna to connect via radiofrequency. Since the Cubesat will be tested in a 0-friction ambient, it can be connected to any wire. This antenna is used to send telemetry to a computer.
4. **Rechargable battery**.

The OBC is programmed with B-dot. It computes the derivative of the magnetic field onboard, and set a PWM (pulse-width modulation) voltage in the magnetorquers. Since the signal in the magnetorquers can only be high (5V) or low (0V), this technique switchs on and off the current with a high frequency. In this way, average values of voltage between 0 and 5V are possible, varying the relative duration of high-tension interval and low-tension interval.

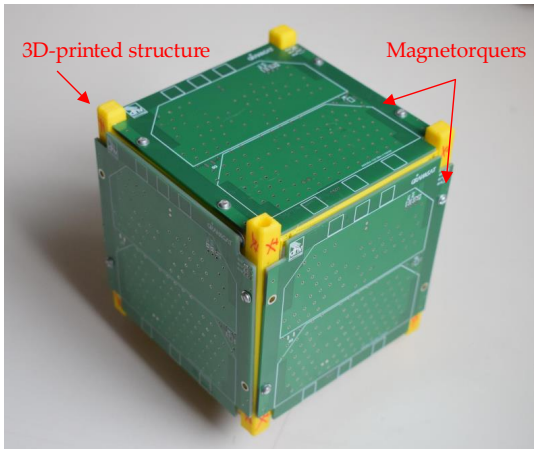

Figure 20: Cubesat prototype

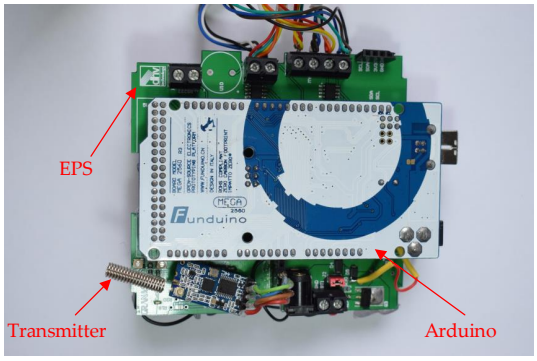

Figure 21: Cubesat internal components (downside)

6.2 Testbed

In order to emulate the zero-gravity environment (required for attitude control tests), a testbed with an air-bearing system has been designed by GranaSAT members (firstly in [26], and finally an improved one in [27]). The one used is shown in Figure 23.

Unfortunately, the performance of the testbed was far from expected. A simple experiment consisted on switch on the testbed with the Cubesat on it, with no initial perturbation. It was seen that the air flow induced a rotation on the Cubesat, reaching more than 200 degrees per second in a few minutes. Figure 24 shows the telemetry sent during the experience.

This angular acceleration was too strong to test any control algorithm with small magnetorquers, thus another experiment was designed in order to partially test the B-dot implementation. Essentially, it was the same experiment but in this case, the OBC

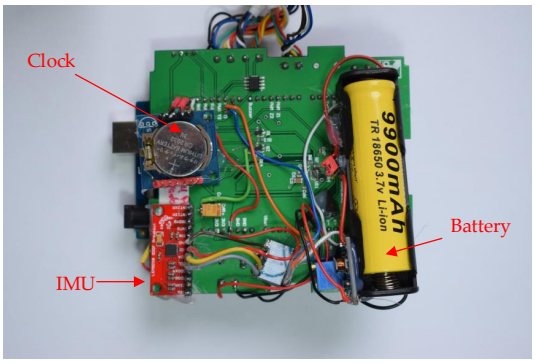

Figure 22: Cubesat internal components (upside)

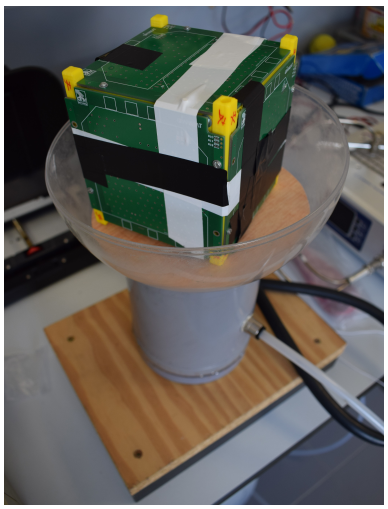

Figure 23: Cubesat on the testbed

also sent the information about the intensity of magnetorquers, expressed as the PWM value written to each magnetorquer.

6.3 Magnetic simulator

A magnetic simulator made with six orthogonal Helmholtz coils is located in GranaSAT's laboratory, desgined by a GranaSAT member [28]. Its mission would be to simulate the magnetic field values in a specified orbit. This device is not used in this thesis, but will be crucial to properly test attitude control algorithm in the future.

6.4 B-dot implementation and results

The B-dot algorithm has been written in C++ and run in the OBC. In order to test its performance, the Cubesat was placed on the testbed while sending angular speed data, as well as the intensity of the magnetorquers, in the form of the corresponding PWM value (from -255 to 255, depending on the sense of the current). The plane XY of the body frame is parallel to the floor, so $\mathbf{z}_B$ is normal to it. In Figure 25, PWM values of intensities are ploted over a short period of time, when the vertical angular speed was close to 115 deg/sec. On the other hand, Figure 26 shows the same situation but

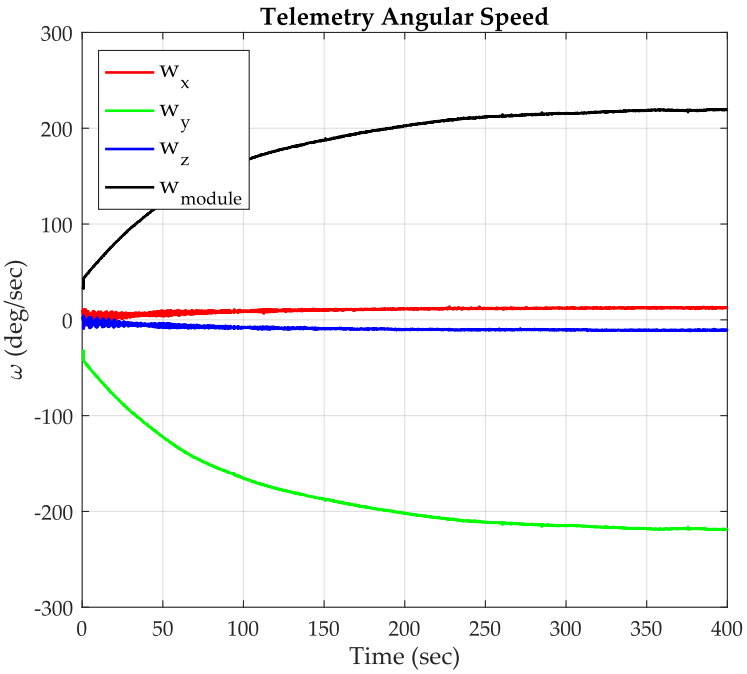

Figure 24: Angular speed during testbed testing

calculated with the attitude simulator.

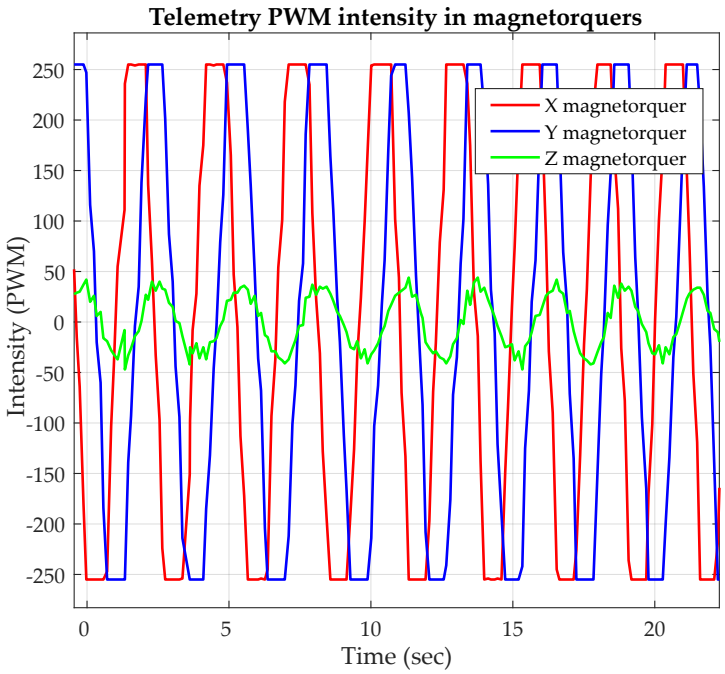

Figure 25: Intensity values in each magnetorquer in PWM units over time (Telemetry data)

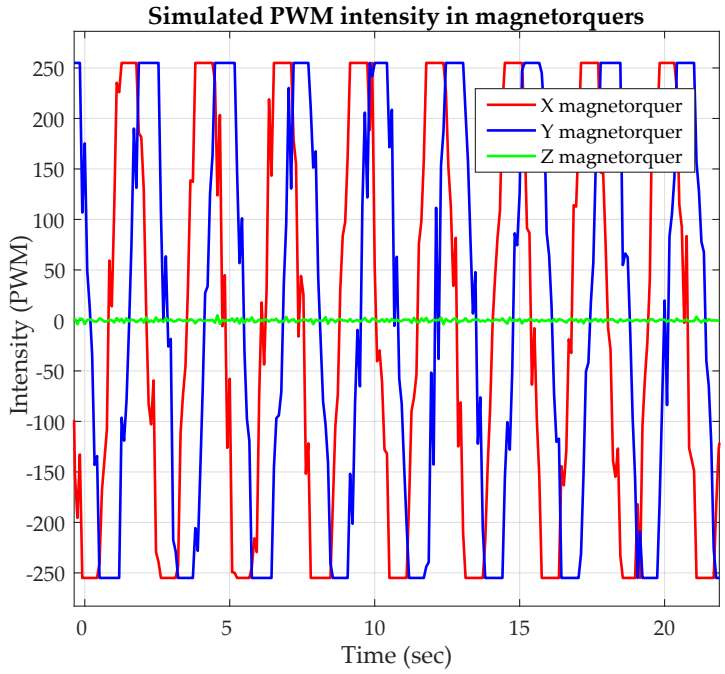

Figure 26: Intensity values in each magnetorquer in PWM units over time (Simulated data)

7 Discussion

7.1 Attitude simulator

One of the goals of this thesis is to design an operative MATLAB simulator for attitude control purposes. It involves orbit propagation, magnetic field calculation and the attitude propagation. The latter is the main part of this simulator, and it is the result of months of implementation and testing. Although there are free source Euler integrators available on the internet, the code written here goes beyond general objectives; it is specifically designed for ADCS simulations, for any orbit or Cubesat size, actuated by magnetorquers or other actuators if defined. Henceforth, GranaSAT team will have a powerfull simulation platform, an essential requirement to put the first satellite of the University of Granada in orbit.

Results obtained here are similar to ones available on many Master's Theses referenced throughout this text.

7.2 Control algorithms testing

Three control algorithms have been tested under certain conditions.

The **B-dot algorithm** has shown an excelent performance (Figure 8), widely cited before in the literature. Regarding to the gain used, it has been setted manually, after a few simulations testing different values. These simulations show that if its value is too low, the satellite lasts a long time before stabilizing, which is expected. However, very high values resulted in a fast angular speed reduction to some value followed by a low convergence to zero. This phenomenon is less obvious to understand; if the magnetic moment produced is too strong, the satellite rapidly aligns itself with the magnetic field vector, a situation in wich the low angle between $\boldsymbol{\omega}$ and $\mathbf{B}$ results in a practically zero torque. Finally, Figure 9 shows a high energy consumption, in comparison with the other control laws. This is due to the high kinetic energy disipation that is done.

Results concerning **Nadir pointing** have shown that the alignment of the body-orbit frames is an asymptotically stable equilibrium when there exists only gravity gradient torque, i.e. the quaternion $\mathbf{q}_o^b = [1000]^T$ is achieved (Figure 10). After a relative intense energy consumption phase, the satellite just needs a minimun amount of energy to keep itself in the equilibrium point, as Figure 11 reflects. However, the situation changed when the aerodynamic torque was added, even with lower inital angular speeds. In Figure 12, the attitude appears to tend to a stable equilibrium, but this equilibrium is not the required one. The energy consumption keeps growing after that equilibrium is reached (Figure 13). Nevertheless, the angle between a camera located on the face +Z and the Nadir remains constant after a few orbits. With an approximated value of 30 degrees (Figure 14), pictures of Earth could be taken.

Finally, the **Target tracking** control law has been tested in a similar way as Nadir pointing. For the target selected, it has shown a remarkable performance (even with a relative high initial angular speed) when the only disturbance was the gravity gradient torque. Attending to the results (Figure 15), it could be said that $\mathbf{q}_t^b = [1000]^T$ is a stable equilibrium, since the attitude quaternion remains always close to the desired value. However, after the inclusion of the aerodynamic torque, that equilibrium is far from being reached with the tested values of α and β (Figure 17). However, some stabilization

is achieved. Concerning to the energy consumed, Figures 16 and 18 exhibit similar results to the ones in Nadir pointing.

The comparison between simulating with and without aerodynamic torque has two purposes, (1) to verificate the performance of the simulator under theoretically studied conditions (in the case of Nadir pointing) and (2) to serve as a prime order aproximation of the studied control laws. Additionaly, for higher altitudes (this study is based on a 400 km orbit), the aerodynamic torque is less important than the gravity gradient torque. For instance, Moon's Cubesats, which are not experiencing aerodynamic torque, could use these controllers.

In the case of the Target tracking, the intensity used by magnetorquers has been plotted to show an apparent contradiction. (Figure 19): while the satellite is fighting against the air molecules, it is using only a tiny fraction of its power limit (each magnetorquer uses less than 0.6 mA, when the maximum allowed current is 100 mA). However, higher values of the attitude term of the magnetic moment do not guide the satellite to the equilibrium point.

7.3 Experimental results

Figures 25 and 26 shows practically identical results: magnetorquers performance is the same as the expected one. Only a few difference is appreciated; in Figure 25 it is shown that the intensity of the magnetorquer Z oscillates over time. This is due to a non-zero angular speed around the body axes which are on the plane parallel to the ground (this is observed in Figure 24).

8 Conclusions and future recommendations

Three attitude control laws have been succesfully simulated in MATLAB. Although B-dot works pretty well under aerodynamic disturbance, some improvements could be done. In [29], an extensive analysis of the control gain is done, and [30] proposes non-constant gains. Further steps could been done in this direction by the GranaSAT team.

Nadir pointing and target tracking exhibited a bad performance under aerodynamic torque. However, results suggest that it is posible to achieve a 3-axis control for Cubesats in orbit below 500 km. Values of constants α and β were setted manually. Nevertheless, time-varing or attitude-depending forms of these coeficients could be implemented, as in [14], for instance. This is one interesting way of research for future students.

Concerning to the physical implementations, a functional Cubesat prototype has been created. In order to properly test its performance, the testbed must be fixed. This could be achieved by remaking the air vents of the air-bearing with the aid of a computerised system.

Solar sensors should be purchased or created by future students to create an attitude determination system based on solar and magnetic measurements.

Finally, some way of testing Nadir pointing and target tracking algorithms (as well as implement them on the OBC) in the magnetic simulator should be designed.

9 Gantt chart of the project

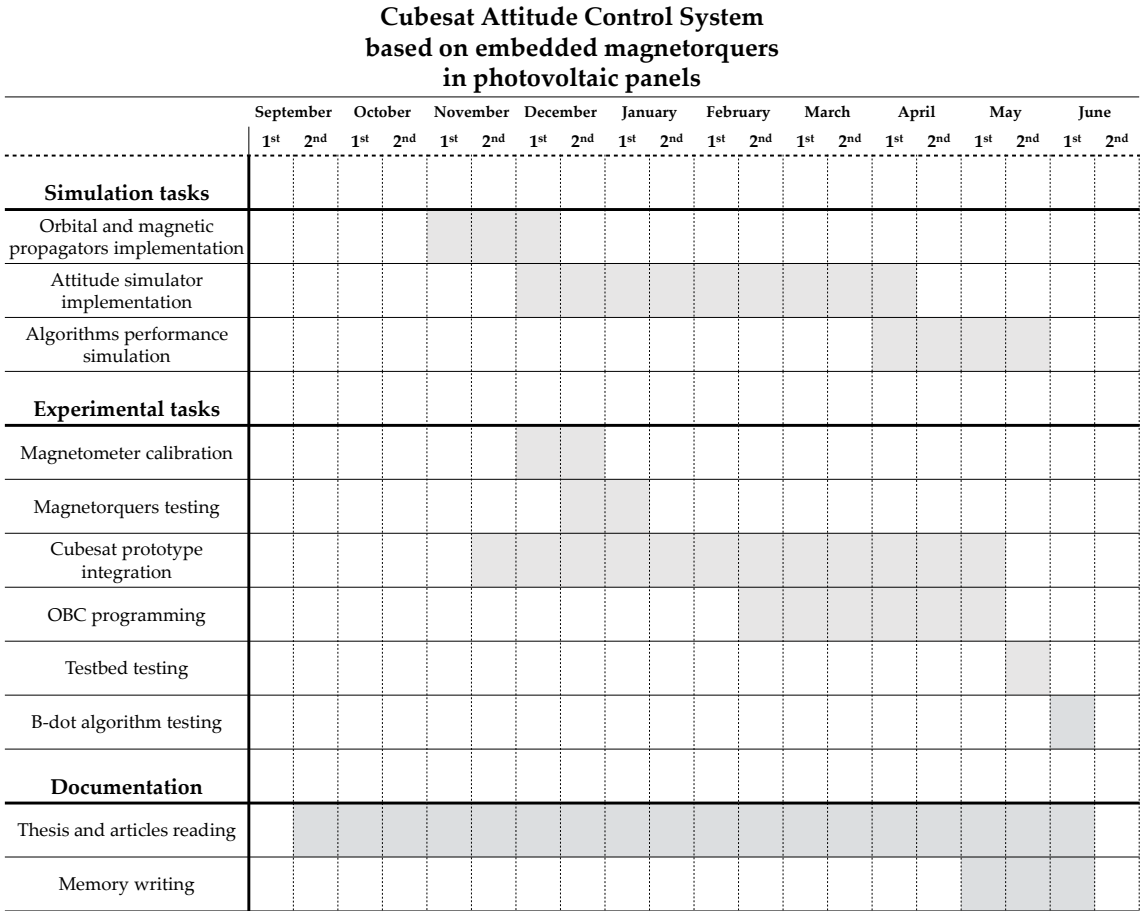

Figure 27: Gantt chart of the project

References

[1] Cubesat standard. http://www.cubesat.org/.

[2] Granasat main page. http://granasat.ugr.es/.

[3] List of Cubesats and other nano-satellites. http://nanosats.eu/.

[4] Bernardo García Olmedo. *Fundamentos de Electromagnetismo*. Dpto. de Electromagnetismo y Física de la Materia (UGR), 2006.

[5] James R. Wertz. *Spacecraft Attitude Determination and Control*. Kluwer Academic Publishers, 2002.

[6] Brian Gasberg Thomsen and Jens Nielsen. CubeSat Sliding Mode Attitude Control. Master's thesis, Aalborg University, Aalborg, June 2016.

[7] Christopher D.Hall. Spacecraft Attitude Dynamics and Control, 2003.

[8] Olav Egeland. *Modeling and simulation for automatic control*. Marine cybernetics, 2002.

[9] F. Landis Markley. *Fundamentals of Spacecraft Attitude Determination and Control*. Space Technology Library, 2014.

[10] Li Qiao, Chris Rizos, and Andrew G. Dempster. Analysis and comparison of cubesat lifetime, 2012. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.588.3019&rep=rep1&type=pdf.

[11] IGRF model. https://www.ngdc.noaa.gov/IAGA/vmod/igrf.html/.

[12] IGRF matlab code. https://www.mathworks.com/matlabcentral/fileexchange/34388-international-geomagnetic-reference-field--igrf--model/.

[13] Eli Jerpseth. Attitude control for the Norwegian student satellite nCube. Master's thesis, Norwegian University of Science and Technology, Trondheim, May 2004.

[14] Rafal Wisniewski. Satellite Attitude Control Using Only Electromagnetic Actuation. Phd thesis, Aalborg University, Fredrik Bajers, Aalborg, December 1996.

[15] Hassan K. Khalil. *Nonlinear Systems*. Prentice Hall, 2002.

[16] M. Vidyasagar. *Nonlinear Systems Analysis*. Prentice Hall, 1993.

[17] Jerrold E. Marsden. *Elementary Classical Analysis 2nd Ed*. WH Freeman, 1993.

[18] Marco Lovera. Magnetic satellite detumbling: the b-dot algorithm revisited. *2015 American Control Conference*, 2015. https://ieeexplore.ieee.org/document/7171005/.

[19] Stian Sondersrod Ose. Attitude determination for the Norwegian student satellite nCube. Master's thesis, Norwegian University of Science and Technology, Trondheim, Norway, June 2004.

[20] Gaute Brathen. Design of Attitude Control System of a Double Cubesat. Master's thesis, Norwegian University of Science and Technology, Trondheim, January 2013.

[21] David A. Vallado and Paul Crawford. SGP4 Orbit Determination. *AIAA/AAS Astrodynamics Specialist Conference and Exhibit. Honolulu, Hawaii* , 2008. https://arc.aiaa.org/doi/abs/10.2514/6.2008-6770.

[22] SGP4 source code. https://celestrak.com/software/vallado-sw.asp.

[23] Valdemir Carrara. An open source satellite attitude and orbit simulator toolbox for Matlab. *DINAME 2015 - Proceedings of the XVII International Symposium on Dynamic Problems of Mechanics, Brazil* , 2015. http://www.dem.inpe.br/~val/publicacoes/DIN-2015-0055.pdf.

[24] PROPAT Matlab Toolbox. http://www.dem.inpe.br/~val/projetos/propat/default.htm.

[25] L.M. Nieto. Plataforma de test para paneles fotovoltaicos aeroespaciales. Master's thesis, University of Granada, Granada, September 2017. This Master's Thesis is a contribution to the GranaSAT project.

[26] V. Burgos. Testbed for a 1u cubesat. Bachelor thesis, University of Granada, Granada, September 2016. This B.S Thesis is a contribution to the GranaSAT project.

[27] A. J. Ortiz. An improved testbed for a 1u cubesat. Master's thesis, University of Granada, Granada, September 2016. This Master's Thesis is a contribution to the GranaSAT project.

[28] R. Lendínez. Simulador de campo magnético terrestre. Master's thesis, University of Granada, Granada, September 2015. This Master's Thesis is a contribution to the GranaSAT project.

[29] Giulio Avanzini. Magnetic Detumbling of a Rigid Spacecraft. *Journal of Guidance Control and Dynamics*, 2012. https://www.researchgate.net/publication/271035898_Magnetic_Detumbling_of_a_Rigid_Spacecraft?el=1_x_8&enrichId=rgreq-9669a7e20173dae92dd9bf97b7d434a4-XXX&enrichSource=Y292ZXJQYWdlOzI2MzAwODQwNztBUzoxMDY4OTY5NjcwODE5OTFAMTQwMjQ5NzYzMTk1Nw==.

[30] Mirko Leomanni. Comparison of Control Laws for Magnetic Detumbling. *Technical Report*, 2012. https://www.researchgate.net/publication/263008407_Comparison_of_Control_Laws_for_Magnetic_Detumbling.