

# Modelling, Simulation and Testing of EIRSAT-1 Attitude Determination and Control System

J. Thompson, D. Sherwin, V. Ubeda, W. O'Connor

November 5, 2017

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Objectives</b>	<b>3</b>
<b>3</b>	<b>ADCS Modes and Functional Requirements</b>	<b>3</b>
3.1	Detumbling . . . . .	3
3.2	Sun/distant object pointing . . . . .	4
3.3	Nadir/zenith pointing . . . . .	4
3.4	Switching between modes . . . . .	4
<b>4</b>	<b>Mathematical Modelling</b>	<b>4</b>
4.1	Co-ordinate systems and Reference Frames . . . . .	4
4.2	Satellite Dynamics . . . . .	4
4.3	Rotational Kinematics . . . . .	5
4.3.1	Alternative representations of attitude and conversions . . . . .	5
4.4	Orbit Propagation . . . . .	6
4.5	Environment . . . . .	6
4.5.1	Magnetic Field . . . . .	9
4.5.2	Gravitational Field . . . . .	9
4.6	Sensor Models . . . . .	9
4.7	Actuator Models . . . . .	9

<b>5</b>	<b>Controller Design</b>	<b>9</b>
<b>6</b>	<b>Matlab-Simulink Model</b>	<b>9</b>
6.1	Dynamics . . . . .	10
6.1.1	Spacecraft Dynamics Block in Simulink . . . . .	10
6.1.2	Orbital Dynamics . . . . .	10
6.2	Environment block . . . . .	10
6.3	Disturbance block . . . . .	10
6.4	Sensor block . . . . .	10
6.5	Actuator block . . . . .	10
6.6	Controller block . . . . .	10
6.7	Reference block . . . . .	10
6.8	Mode Select block . . . . .	10
6.9	Dynamics block . . . . .	10
6.10	Test case . . . . .	10
6.11	Post processing . . . . .	11
<b>7</b>	<b>Results</b>	<b>11</b>
<b>A</b>	<b>Github</b>	<b>11</b>
A.1	Structure . . . . .	11
A.2	Typical Workflow . . . . .	11
<b>B</b>	<b>Latex</b>	<b>11</b>
<b>C</b>		<b>11</b>

## 1 Introduction

This document details a study of the modelling and simulation of the Attitude Determination and Control System (ADCS) for the EIRSAT-1 cubesat. This includes the satellite attitude dynamics, orbit propagation after release from the ISS; disturbances torques, position and attitude sensor models, magnetic actuator models and attitude feedback control system. The simulation environment was developed using MATLAB/Simulink.

Simulation results are presented for several different test cases. These test cases include demonstrations of different control modes, different control systems and different actuator and sensor configurations including failure of one or more devices.

## 2 Objectives

- Create a simulation environment for the EIRSAT-1 ADCS subsystem including:
  - Satellite attitude dynamics, modelled as a single rigid body with 6DOF
  - Orbit propagation of the satellite after release from the ISS
  - Disturbance torques including: gravity gradient, solar radiation pressure, aerodynamic, residual magnetic field
  - orbit propagation after release from ISS
- Define the different ADCS operational modes and functional requirements for each mode
- Design feedback controllers for each mode of the ADCS:
  - first assuming ideal 3-axis actuators
  - then with two magnetorquer actuators
- Design algorithm for safely switching between control modes
- Design controllers for failure modes including:
  - failure of one magnetorquer
  - failure of attitude sensors (needs to be more clearly defined)

## 3 ADCS Modes and Functional Requirements

- Detumbling
- Sun/object pointing
- Nadir/zenith pointing

### 3.1 Detumbling

Max rate to detumble? Time taken to detumble? Power?

### 3.2 Sun/distant object pointing

Required pointing accuracy? Settling time from one place to another? Power?

### 3.3 Nadir/zenith pointing

Required pointing accuracy? Time? Power?

### 3.4 Switching between modes

Which switches can happen? When can they happen ? (where in orbit, in eclipse?)  
Time taken?

## 4 Mathematical Modelling

### 4.1 Co-ordinate systems and Reference Frames

The spacecraft moves in an inertial reference frame  $N$  with associated Cartesian coordinate system  $O_{XYZ}$  where  $\{\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3\}$  is a set of unit vectors along the  $X$ ,  $Y$  and  $Z$  axes respectively. This coordinate system is centred at the centre of the Earth,  $\mathbf{n}_3$  is normal to the equatorial plane (hence parallel to the earth rotational axis), and points towards north pole,  $\mathbf{n}_1$  points towards the vernal equinox, and  $\mathbf{n}_2$  completes the right-handed orthonormal frame.

Recall that  $N$  is considered an inertial reference, neglecting the rotation of the Earth around the sun, and that of the sun around bigger objects.

A second reference frame  $A$ , is attached to the spacecraft rigid body with coordinate system  $C_{xyz}$  where  $C$  is the mass centre of the rigid body and  $\{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3\}$  are unit vectors along the  $x$ ,  $y$  and  $z$  axes respectively. It will be assumed that the three axes of the body fixed frame are aligned with the principal axes of inertia of the satellite.

A third reference frame, named orbital frame, and here denoted  $O$ , is spanned by the unit vectors  $\{\mathbf{o}_1, \mathbf{o}_2, \mathbf{o}_3\}$ , and centred at the satellite's centre of mass. The vector  $\mathbf{o}_3$  constantly points to the Earth's centre, whereas  $\mathbf{o}_2$  aligns with the orbit angular momentum.  $\mathbf{o}_2$  is the cross product of  $\mathbf{o}_3$  and  $\mathbf{o}_1$ , completing the three axis orthonormal frame.

### 4.2 Satellite Dynamics

d

EIRSAT-1 is modelled as a single rigid body with six degrees of freedom: three rotational and three translational.

The satellite has mass  $m$  and inertia tensor  $\mathbf{I}$  in the body fixed coordinate system  $O$ . From the assumption that  $O$  is aligned with the principal axes of inertia of the satellite, it follows that is a  $3 \times 3$  diagonal matrix.

### 4.3 Rotational Kinematics

The attitude of the spacecraft is described using unit quaternions. These avoid singularities that may arise when other approaches such as DCM (Direction Cosine Matrix) or Euler Angle - Euler Axis are used. Due to the lack of singularities, unit quaternions are widely employed through the literature for computational purposes in ADCS design and simulations. See, for instance ([???]).

#### 4.3.1 Alternative representations of attitude and conversions

Rotations in three-dimensional Euclidean space may be described in a number of ways. For instance, let  $\mathbf{r}^X$  be a vector expressed as a linear combination of the unit vectors that span the right - handed orthonormal frame  $X = \{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ . That is

$$\mathbf{r} = \sum_{i=1}^3 r_i \mathbf{e}_i \quad (1)$$

The same vector can be represented by a linear combination of unit vectors spanning an alternative orthonormal frame  $X' = \{\mathbf{e}'_1, \mathbf{e}'_2, \mathbf{e}'_3\}$

$$\mathbf{r} = \sum_{i=1}^3 r'_i \mathbf{e}'_i \quad (2)$$

In vector form:

$$\mathbf{r} = \begin{pmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} = \begin{pmatrix} \mathbf{e}'_1 & \mathbf{e}'_2 & \mathbf{e}'_3 \end{pmatrix} \begin{pmatrix} r'_1 \\ r'_2 \\ r'_3 \end{pmatrix} = \quad (3)$$

Each one of the components  $r'_i$  is the projection (ie. vector product) of the vector  $\mathbf{r}$  into  $\mathbf{e}'_i$ . That is  $r'_i = \mathbf{r} \cdot \mathbf{e}'_i = \sum_{j=1}^3 (r_j \mathbf{e}_j) \cdot \mathbf{e}'_i = \sum_{j=1}^3 (\mathbf{e}'_i \cdot \mathbf{e}_j) r_j$ . In matrix form:

$$\begin{pmatrix} r'_1 \\ r'_2 \\ r'_3 \end{pmatrix} = \begin{pmatrix} \mathbf{e}'_1 \cdot \mathbf{e}_1 & \mathbf{e}'_1 \cdot \mathbf{e}_2 & \mathbf{e}'_1 \cdot \mathbf{e}_3 \\ \mathbf{e}'_2 \cdot \mathbf{e}_1 & \mathbf{e}'_2 \cdot \mathbf{e}_2 & \mathbf{e}'_2 \cdot \mathbf{e}_3 \\ \mathbf{e}'_3 \cdot \mathbf{e}_1 & \mathbf{e}'_3 \cdot \mathbf{e}_2 & \mathbf{e}'_3 \cdot \mathbf{e}_3 \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} \quad (4)$$

Figure 1: Direction Cosines

$$\mathbf{A}_X^{X'} = \begin{pmatrix} \mathbf{e}'_1 \cdot \mathbf{e}_1 & \mathbf{e}'_1 \cdot \mathbf{e}_2 & \mathbf{e}'_1 \cdot \mathbf{e}_3 \\ \mathbf{e}'_2 \cdot \mathbf{e}_1 & \mathbf{e}'_2 \cdot \mathbf{e}_2 & \mathbf{e}'_2 \cdot \mathbf{e}_3 \\ \mathbf{e}'_3 \cdot \mathbf{e}_1 & \mathbf{e}'_3 \cdot \mathbf{e}_2 & \mathbf{e}'_3 \cdot \mathbf{e}_3 \end{pmatrix} \quad (5)$$

There exists a matrix  $\mathbf{A}_X^{X'} \in SO(3)$  that allows the transformation to the frame  $X' = \{\mathbf{e}'_1, \mathbf{e}'_2, \mathbf{e}'_3\}$ . That is to say

$$\mathbf{r}^{X'} = \mathbf{A}_X^{X'} \mathbf{r}^X \quad (6)$$

$$\mathbf{r}^{X'} = \sum_{i=1}^3 a_{ij} r_j^X \mathbf{e}'_i \quad (7)$$

Where  $SO(3)$  is the group of rotations in three-dimensional euclidean space. If rotations are described using an orthonormal basis in  $\mathbb{R}^3$ , then all matrices in  $SO(3)$  are  $3 \times 3$  orthogonal matrices: real matrices whose transpose equals its left and right inverses.

The same matrix  $\mathbf{A}_X^{X'}$  that produces the change of coordinate system from  $X$  to  $X'$  when applied to a vector expressed in terms of  $X$ , may be used to rotate a vector  $\mathbf{r}$ , forcing it to perform the inverse rotation that  $X'$  did with respect to  $X$ , to obtain another vector  $\mathbf{u}$ . Intuitively, if one can place an observer fixed to  $\mathbf{r}$  while such a rotation takes place, it would *see* the reference frame  $X$  transform to  $X'$ . Recall that when the rotation matrix is used for the latter purpose, no frame transformation takes place, and only one coordinate frame is used to represent each vector.

The matrix  $\mathbf{A}_X^{X'}$  is called the **Direction Cosine Matrix** as it is formed by the dot products  $\mathbf{A}_X^{X'} = \{a_{ij}\} = \{\mathbf{e}'_i \cdot \mathbf{e}_j\}$  which are, in fact, the cosines of the angles that the unit vectors  $\mathbf{e}'_i$  form with respect to  $\mathbf{e}_j$ . (see figure 1)

The orientation of the spacecraft may also be described using Euler angles. Starting with the body-fixed *oxyz* axes aligned with the *OXYZ* axes, the body undergoes a sequence of rotations by angles  $\theta_3$ ,  $\theta_2$  and  $\theta_1$  about the body-fixed  $z$  (yaw),  $y$  (pitch) and  $x$  (roll) axes respectively to reach its final orientation.

The DCM (Direction Cosine Matrix) is then described by equation 8.

$$C = \quad (8)$$

#### 4.4 Orbit Propagation

#### 4.5 Environment

#### Spherical Harmonics

The set of spherical harmonics is an orthonormal set of functions defined in the unit sphere. They are given as the angular part of the solution to the Laplace Equation in spherical coordinates. Analogously to how sines and cosines are used in harmonic analysis to represent periodic functions through Fourier Series, Spherical Harmonics may be used to define functions in the surface of a sphere.

### Spherical Coordinates

It is important to identify the coordinates to be used, as, for instance, spherical coordinates have often two different interpretations: one, which is mainly used by mathematicians, uses the latitude, being the angle between the position vector and the equatorial plane, as one of the angular coordinates; while the other, broadly employed by physicists, uses the co-latitude, defined as the angle between the position vector and the  $z$  axis, thus being the complementary of the latitude.

Here, the spherical coordinates are taken to be

$r$ , the radial distance from Earth's center

$\phi$ , the East longitude, bounded between  $-\pi$  and  $\pi$

$\theta$ , the co-latitude, defined to be  $\theta = \frac{\pi}{2} - \theta'$ , with  $\theta'$  the latitude. Bounded between 0 and  $\pi$

The magnetic field is given by the negative gradient of the potential  $V$  defined for  $r \geq R_e$ , and given by the spherical harmonic approximation

$$V = r \sum_{n=1}^L \left( \frac{R_e}{r} \right)^{n+1} \sum_{m=0}^n (g_n^m \cos(m\phi) + h_n^m \sin(m\phi)) P_n^m(\cos\theta) \quad (9)$$

That is

$$\boldsymbol{\beta} = -\nabla V \quad (10)$$

Where  $g_n^m$  and  $h_n^m$  are the set of IGRF gaussian coefficients, published and revised every five years by the participating members of the IAGA (International Association of Geomagnetism and Aeronomy). The 12<sup>th</sup> generation is here used, as the last revision by the time of this work. These coefficient also include the secular variation (SV), which bestow the model on the proper time dependence. Keeping track of the change in the coefficients in  $nT$  per year. The coefficients for some epoch year are referred to as IGRF. When real data about the geomagnetic field becomes available so that some adjustments can be made, the model becomes definitive and changes its name to DGRF (Definitive geomagnetic reference field).

$P_n^m(\cos\theta)$  are the Schmidt normalized associated Legendre polynomials of degree  $n$  and order  $m$

One can find very effective, recursive ways of computing the Schmidt quasi-normalization factors, the Associated Legendre polynomials and its derivatives, as can be found on [J.Davis - Mathematical Modeling of Earth's Magnetic Field]

The three componentens of the magnetic field in  $r$ ,  $\phi$  and  $\theta$  directions can be found in terms of the partial derivatives of 9

$$\beta_r = -\frac{\partial V}{\partial r} = \sum_{n=1}^N (n+1) \left(\frac{R_e}{r}\right)^{n+2} \sum_{m=0}^n (g_n^m \cos(m\phi) + h_n^m \sin(m\phi)) P_n^m(\cos\theta) \quad (11)$$

$$\beta_\theta = -\frac{1}{r} \frac{\partial V}{\partial \theta} = -\sum_{n=1}^L \left(\frac{R_e}{r}\right)^{n+2} \sum_{m=0}^n (g_n^m \cos(m\phi) + h_n^m \sin(m\phi)) \frac{\partial P_n^m(\cos\theta)}{\partial \theta} \quad (12)$$

$$\beta_\phi = -\frac{1}{r \sin\theta} \frac{\partial V}{\partial \phi} = \frac{1}{\sin\phi} \sum_{n=1}^L \left(\frac{R_e}{r}\right)^{n+2} \sum_{m=0}^n m (g_n^m \sin(m\phi) - h_n^m \cos(m\phi)) P_n^m(\cos\theta) \quad (13)$$

### Legendre Polynomials

The set of regular Legendre Polynomials is given by the solutions  $P_n(\nu)$  to the equation

$$\frac{1}{\sqrt{1-2\nu x+x^2}} = \sum_{n=0}^{\infty} x^n P_n(\nu) \quad (14)$$

Which gives

$$P_n(\nu) = \frac{1}{n!2^n} \frac{d^n}{d\nu^n} (\nu^2 - 1)^n \quad (15)$$

### Associated Legendre Polynomials

The associated Legendre polynomials  $P_n^m(\nu)$  of degree  $n$  and order  $m$  relate to  $P_n(\nu)$  by

$$P_{n,m}(\nu) = (1-\nu^2)^{\frac{1}{2}m} \frac{d^m}{d\nu^m} (P_n(\nu)) \quad (16)$$

These polynomias are not normalized in any way.

### Gaussian normalization and Schmidt quasi-normalization

To compute the geomagnetic field, the Schmidt quasi-normalization is used, and is given by

$$P_n^m(\nu) = \sqrt{\frac{2(n-m)!}{(n+m)!}} P_{n,m}(\nu) \quad (17)$$



The Gaussian normalized Legendre polynomials, whose interest lies in the existence of a recursive formula for their effective computation are given by

$$P^{n,m}(\nu) = \frac{2^n!(n-m)!}{(2n)!} P_{n,m}(\nu) \quad (18)$$

One can relate the Gaussian normalization and the Schmidt quasi-normalization by using

$$P_n^m(\nu) = \Gamma_{n,m} P^{n,m}(\nu) \quad (19)$$

Where

$$\begin{aligned} \Gamma_{n,m} &= \sqrt{\frac{(n-m)!}{(n+m)!} \frac{(2n-1)!!}{(n-m)!}}, \quad m = 0 \\ \Gamma_{n,m} &= \sqrt{\frac{2(n-m)!}{(n+m)!} \frac{(2n-1)!!}{(n-m)!}}, \quad m \neq 0 \end{aligned} \quad (20)$$

It is more convinient to modify the coefficients, as

$$\begin{aligned} g_n^m &= \Gamma_{n,m} g_{n,m} \\ h_n^m &= \Gamma_{n,m} h_{n,m} \end{aligned} \quad (21)$$

1

#### 4.5.1 Magnetic Field

#### 4.5.2 Gravitational Field

#### 4.6 Sensor Models

#### 4.7 Actuator Models

### 5 Controller Design

### 6 Matlab-Simulink Model

Components:

Table 1: Spacecraft Dynamics Block Signals and Parameters

	Name	Symbol	Dimension	Unit
Input	Total Torque	$\boldsymbol{\tau}$	$3 \times 1$	
Output	Spacecraft angular velocity (Body frame)	$\boldsymbol{\omega}_b$	$3 \times 1$	
Output	Attitude quaternion (Body relative to Inertial)	$q_{b/i}$	$3 \times 1$	
Parameter	Spacecraft inertia tensor (Body frame)	$\boldsymbol{I}$	$3 \times 3$	$\text{kg m}^2$

## 6.1 Dynamics

### 6.1.1 Spacecraft Dynamics Block in Simulink

Outputs

### 6.1.2 Orbital Dynamics

### 6.2 Environment block

### 6.3 Disturbance block

### 6.4 Sensor block

### 6.5 Actuator block

### 6.6 Controller block

### 6.7 Reference block

### 6.8 Mode Select block

### 6.9 Dynamics block

### 6.10 Test case

Model Setup File Simulation Setup File Output Text file

---

<sup>1</sup>Note the use of notation here to distinguish between Regular, Associated, and Schmidt quasi-normalized Legendre Polynomials

### **6.11 Post processing**

## **7 Results**

### **A Github**

#### **A.1 Structure**

#### **A.2 Typical Workflow**

Branch for feature Pull Request

### **B Latex**

### **C**