

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,000

Open access books available

116,000

International authors and editors

120M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



# Modeling and Attitude Control of Satellites in Elliptical Orbits

*Espen Oland*

## Abstract

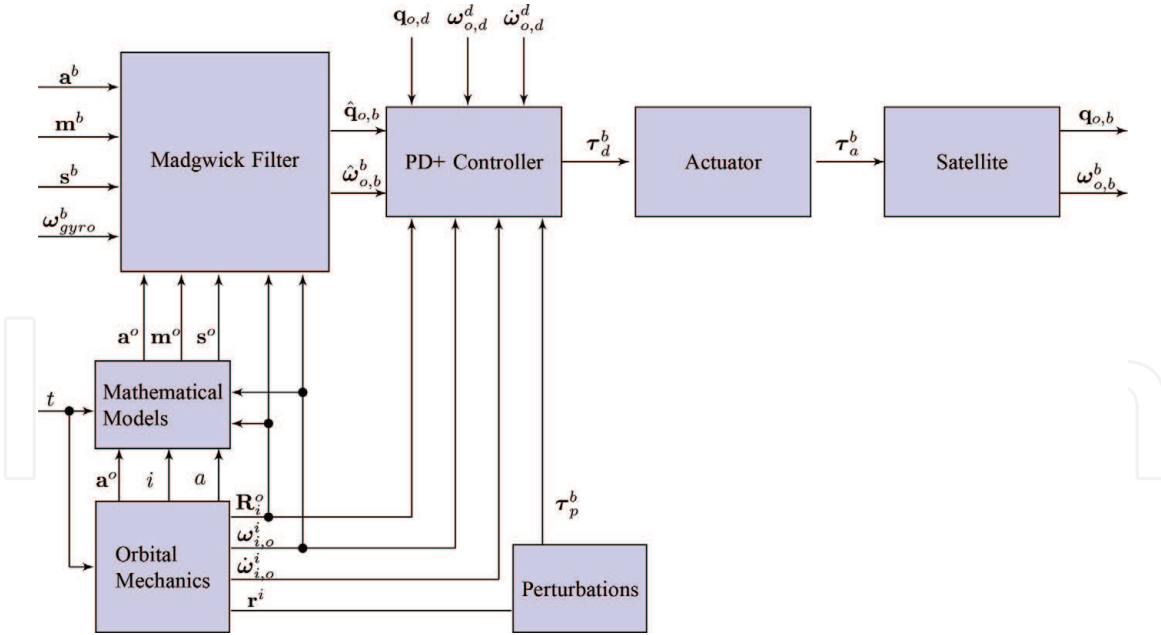
The attitude determination and control system (ADCS) for spacecraft is responsible for determining its orientation using sensor measurements and then applying actuation forces to change the orientation. This chapter details the different components required for a complete attitude determination and control system for satellites moving in elliptical orbits. Specifically, the chapter details the orbital mechanics; perturbations; controller design; actuation methods such as thrusters, reaction wheels, and magnetic torquers; actuation modulation methods such as bang-bang, pulse-width modulation, and pulse-width pulse-frequency; as well as attitude determination using vector measurements combined with mathematical models. In sum, the work describes in a tutorial manner how to put everything together to enable the design of a complete satellite simulator.

**Keywords:** ADCS, attitude control, attitude estimation, thrusters, reaction wheels, magnetic torquers, elliptical orbits, PWM, PWPF, bang-bang, Madgwick filter, Sun vector model, magnetic field model, sliding surface control, quaternions, angular velocity

## 1. Introduction

The problem of developing attitude determination and control systems (ADCS) has received much attention in the last century with general books such as [1–4], as well as description of individual ADCS designs for different satellites in works such as [5–8]. While Refs. [1, 2] can be considered excellent foundation books within the topic of spacecraft modeling and control, there is a need for a more concise presentation of the attitude control problem and how this can be modeled in a simple manner, both not only as a tutorial for new researchers but also to give insight into the different components required for ADCS design drawing on ideas and results from previous works as mentioned above.

This chapter is an extension of [9] and builds on much of the previous work, as well as the research done through the HiNCube project as presented in [10–12]. This work considers the problem of designing a complete ADCS system comprising all the required components. **Figure 1** shows the control structure and the required signal paths, giving an overview of the contents in this chapter, as each block is described in detail to put the reader in position to design their own ADCS system. The required sensors for this system are a magnetometer to measure the Earth's magnetic field, a gyro to measure the angular velocity of the satellite, as well as Sun sensors for measuring the direction toward the Sun. Further, the mathematical



**Figure 1.** This figure shows the different components required for modeling and controlling a satellite in an elliptical orbit and shows the main components required for creating a satellite simulator.

models used together with sensor measurements to determine the attitude of the satellite requires a real-time clock, as the time and date are required to know the direction toward the Sun as well as what the magnetic field looks like at a given day and time. Comparing the sensor measurements with the mathematical models allows for the determination of the attitude of the satellite, something that is done using the Madgwick filter as presented in [13]. With an estimated attitude obtained using the Madgwick filter, the attitude can be controlled to point a sensor onboard the satellite in a desired direction, something that is solved in this chapter using a PD+ attitude controller, calculating the desired torques required in order to make the attitude and angular velocity errors go to zero. In order to create the desired moments, this chapter presents how this can be achieved using a number of different actuators, namely, magnetic torquers, reaction wheels, and thrusters. The orbital mechanics block describes how the satellite moves in its elliptical orbit, while the perturbing forces and moments block describe how the different perturbations affect the satellite. Simulations show the performance of the different methods and should put the reader in a position to simulate and design new attitude determination and control systems for satellites in elliptical orbits.

## 2. Mathematical modeling

### 2.1 Notation

This subsection is similar to the author's previous works, e.g., [9, 14]. Let  $\dot{\mathbf{x}} = d\mathbf{x}/dt$  denote the time derivative of a vector, while the Euclidean length is defined as  $\|\mathbf{x}\| = \sqrt{\mathbf{x}^T \mathbf{x}}$ . Superscript denotes frame of reference for a given vector. The rotation matrix is denoted  $\mathbf{R}_a^c \in \mathcal{SO}(3) = \{\mathbf{R} \in \mathbb{R}^{3 \times 3} : \mathbf{R}^T \mathbf{R} = \mathbf{I}, \det(\mathbf{R}) = 1\}$ , which rotates a vector from frame  $a$  to frame  $c$  and where  $\mathbf{I}$  denotes the identity matrix. The inverse rotation is found by taking its transpose, such that  $\mathbf{R}_c^a = (\mathbf{R}_a^c)^T$ . The angular velocity of frame  $c$  relative to frame  $a$  referenced in frame  $e$  is denoted  $\omega_{a,c}^e$ , and angular velocities can be added together as  $\omega_{a,f}^e = \omega_{a,c}^e + \omega_{c,f}^e$  (cf. [15]).

The derivative of the rotation matrix is defined as  $\dot{\mathbf{R}}_a^c = \mathbf{R}_a^c \mathbf{S}(\boldsymbol{\omega}_{c,a}^a)$  where  $\mathbf{S}(\cdot)$  denotes the cross product operator, which is defined such that for two vectors  $\mathbf{v}_1, \mathbf{v}_2 \in \mathbb{R}^3$ ,  $\mathbf{S}(\mathbf{v}_1)\mathbf{v}_2 = \mathbf{v}_1 \times \mathbf{v}_2$ ,  $\mathbf{S}(\mathbf{v}_1)\mathbf{v}_2 = -\mathbf{S}(\mathbf{v}_2)\mathbf{v}_1$ ,  $\mathbf{S}(\mathbf{v}_1)\mathbf{v}_1 = \mathbf{0}$ , and  $\mathbf{v}_1^T \mathbf{S}(\mathbf{v}_2)\mathbf{v}_1 = 0$ .

Quaternions can be used to parameterize the rotation matrix, where  $\mathbf{q}_{c,a} \in \mathcal{S}^3 = \{\mathbf{q} \in \mathbb{R}^4 : \mathbf{q}^T \mathbf{q} = 1\}$  denotes the quaternion representing a rotation from frame  $a$  to frame  $c$  through the angle of rotation  $\vartheta_{c,a}$  around the axis of rotation  $\mathbf{k}_{c,a}$ . The inverse quaternion is defined as  $\mathbf{q}_{a,c} = [\eta_{c,a} \quad -\boldsymbol{\epsilon}_{c,a}^T]^T$ , also sometimes denoted as  $\mathbf{q}^*$ . A quaternion comprises a scalar and a vector part, where  $\eta_{c,a}$  denotes the scalar part, while  $\boldsymbol{\epsilon}_{c,a} \in \mathbb{R}^3$  denotes the vector part. This allows the rotation matrix to be constructed using quaternions as  $\mathbf{R}_a^c = \mathbf{I} + 2\eta_{c,a} \mathbf{S}(\boldsymbol{\epsilon}_{c,a}) + 2\mathbf{S}^2(\boldsymbol{\epsilon}_{c,a})$ . Composite quaternions can be found through quaternion multiplication as

$$\mathbf{q}_{c,e} = \mathbf{q}_{c,a} \otimes \mathbf{q}_{a,e} = \mathbf{T}(\mathbf{q}_{c,a}) \mathbf{q}_{a,e} \text{ with}$$

$$\mathbf{T}(\mathbf{q}_{c,a}) = \begin{bmatrix} \eta_{c,a} & -\boldsymbol{\epsilon}_{c,a}^T \\ \boldsymbol{\epsilon}_{c,a} & \eta_{c,a} \mathbf{I} + \mathbf{S}(\boldsymbol{\epsilon}_{c,a}) \end{bmatrix}. \quad (1)$$

The use of the quaternion product ensures that the resulting quaternion maintains the unit length property. The quaternion kinematics is defined as

$$\dot{\mathbf{q}}_{c,a} = \frac{1}{2} \mathbf{q}_{c,a} \otimes \begin{bmatrix} 0 \\ \boldsymbol{\omega}_{c,a}^a \end{bmatrix} = \frac{1}{2} \mathbf{T}(\mathbf{q}_{c,a}) \begin{bmatrix} 0 \\ \boldsymbol{\omega}_{c,a}^a \end{bmatrix}. \quad (2)$$

For attitude control, several different frames are needed:

**Inertial:** The Earth-centered inertial (ECI) has its origin in the center of the Earth, where the  $\mathbf{x}^i$  axis points toward the vernal equinox and the  $\mathbf{z}^i$  points through the North Pole, while  $\mathbf{y}^i$  completes the right-handed orthonormal frame. The inertial frame is denoted by  $\mathcal{F}^i$ .

**Orbit:** The orbit frame has its origin in the center of mass of the satellite (cf. [16], p. 479). The  $\mathbf{e}_r$  axis coincides with the radius vector  $\mathbf{r}^i \in \mathbb{R}^3$ , which goes from the center of the Earth to the center of mass in the satellite. The  $\mathbf{e}_h$  axis is parallel to the orbital angular momentum vector, pointing in the normal direction of the orbit. The  $\mathbf{e}_\theta$  completes the right-handed orthonormal frame where the vectors can be described as  $\mathbf{e}_r = \frac{\mathbf{r}^i}{\|\mathbf{r}^i\|}$ ,  $\mathbf{e}_\theta = \mathbf{e}_h \times \mathbf{e}_r$ , and  $\mathbf{e}_h = \frac{\mathbf{h}}{\|\mathbf{h}\|}$  where  $\mathbf{h} = \mathbf{r}^i \times \dot{\mathbf{r}}^i$ . The orbit frame is denoted by  $\mathcal{F}^o$ .

**Body:** The body frame has its origin in the center of mass of the satellite, where its axes coincide with the principal axes of inertia. The body frame is denoted by  $\mathcal{F}^b$ .

**Desired:** The desired frame can be defined arbitrarily to achieve any given objective (cf. [17]). The desired frame is denoted by  $\mathcal{F}^d$ .

## 2.2 Orbital mechanics

This section describes how the orbit frame can be related to the inertial frame through the six classical orbital parameters, and for more details, the reader is referred to [1]. Specifically, the objective with this subsection is to find the radius, velocity, and acceleration vector of the orbit, as well as its angular velocity and acceleration. From well-known orbital mechanics, the six classical parameters can be defined as the semimajor axis  $a$ , the eccentricity  $e$ , the inclination  $i$ , the right ascension of the ascending node  $\Omega$ , the argument of the perigee  $\omega$ , and the mean anomaly  $M$ . The distance to the apogee and perigee from the center of the Earth can

be defined, respectively, as  $r_a$  and  $r_p$ , allowing the semimajor axis to be found as  $a = \frac{r_a+r_p}{2}$  and the eccentricity of the orbit as  $e = \frac{r_a-r_p}{r_a+r_p}$ , while the mean motion can be found from  $n = \sqrt{\frac{\mu}{a^3}}$ . Here,  $\mu = GM_{Earth}$ , where  $G$  is the gravitational constant, while  $M_{Earth}$  is the mass of the Earth. With knowledge of the mean motion, the mean anomaly can be found as  $M = n(t - t_0) = \psi - e \sin(\psi)$  where  $\psi$  is the eccentric anomaly,  $t$  is the time, and  $t_0$  is the time when passing the perigee. To properly describe where in the orbit the satellite is located, the true anomaly can be found as  $\theta = \cos^{-1}\left(\frac{\cos(\psi)-e}{1-e \cos(\psi)}\right)$ , while its derivative can be found as  $\dot{\theta} = \frac{n(1+e \cos(\theta))^2}{(1-e^2)^{\frac{3}{2}}}$  ([18], p. 42). When running a simulation, it is desirable to have a continuously increasing true anomaly, while the direct method will map the angle between 0 and 180°. Instead, by integrating the derivative overtime, a smooth true anomaly can be found that increases continuously. The eccentric anomaly, however, cannot be found analytically, but can be found through an iterative algorithm as described in ([1], p. 26)  $\psi_k(t) = M(t) + e \sin(\psi_{k-1}(t))$ , where  $k$  is the iteration number. This algorithm is valid as long as  $0 < e < 1$ , which holds for elliptical orbits. From these calculations, the rotation matrix from inertial frame to orbit frame can now be constructed as

$$\mathbf{R}_i^o = \begin{bmatrix} \cos(\omega + \theta) \cos(\Omega) - \cos(i) \sin(\omega + \theta) \sin(\Omega) & \cos(\omega + \theta) \sin(\Omega) + \sin(\omega + \theta) \cos(i) \cos(\Omega) & \sin(\omega + \theta) \sin(i) \\ -\sin(\omega + \theta) \cos(\Omega) - \cos(i) \sin(\Omega) \cos(\omega + \theta) & -\sin(\omega + \theta) \sin(\Omega) + \cos(\omega + \theta) \cos(i) \cos(\Omega) & \cos(\omega + \theta) \sin(i) \\ \sin(i) \sin(\Omega) & -\sin(i) \cos(\Omega) & \cos(i) \end{bmatrix}. \quad (3)$$

The radius, velocity, and acceleration vector can be defined in the orbit frame, respectively, as ([1], pp. 26–27)

$$\mathbf{r}^o = \begin{bmatrix} a \cos(\psi) - ae & a \sin(\psi) \sqrt{1 - e^2} & 0 \end{bmatrix}^T, \quad (4)$$

$$\mathbf{v}^o = \begin{bmatrix} -\frac{a^2 n}{r} \sin(\psi) & \frac{a^2 n}{r} \sqrt{1 - e^2} \cos(\psi) & 0 \end{bmatrix}^T, \quad (5)$$

$$\mathbf{a}^o = \begin{bmatrix} -\frac{a^3 n^2}{r^2} \cos(\psi) & -\frac{a^3 n^2}{r^2} \sqrt{1 - e^2} \sin(\psi) & 0 \end{bmatrix}^T, \quad (6)$$

where  $r = \|\mathbf{r}^i\|$  is the length of the radius vector. Each of these vectors can be rotated to the inertial frame using Eq. (3), such that  $\mathbf{r}^i = \mathbf{R}_i^o \mathbf{r}^o$ ,  $\mathbf{v}^i = \mathbf{R}_i^o \mathbf{v}^o$ , and  $\mathbf{a}^i = \mathbf{R}_i^o \mathbf{a}^o$ . The angular velocity of the orbit frame relative to the inertial frame can be found as  $\boldsymbol{\omega}_{i,o}^i = \frac{\mathbf{r}^i \times \mathbf{v}^i}{(\mathbf{r}^i)^T \mathbf{r}^i}$ , while the angular acceleration can be found through differentiation as

$$\dot{\boldsymbol{\omega}}_{i,o}^i = \frac{(\mathbf{r}^i \times \mathbf{a}^i)(\mathbf{r}^i)^T \mathbf{r}^i - 2(\mathbf{r}^i \times \mathbf{v}^i)(\mathbf{v}^i)^T \mathbf{r}^i}{((\mathbf{r}^i)^T \mathbf{r}^i)^2}. \quad (7)$$

In order to implement the orbital mechanics in, e.g., a Simulink framework, the required input to the subsystem would be the time ( $t$ ). Further, the orbital parameters must be defined as given in **Table 1** and can be changed depending on the orbit. These constants allow for the calculations of the eccentricity ( $e$ ), the semimajor axis ( $a$ ), and mean motion ( $n$ ). With the mean motion, the mean anomaly ( $M$ ) can be found and used to approximate the eccentric anomaly ( $\psi$ ) using the iterative algorithm presented above. The rate of change of the true anomaly ( $\dot{\theta}$ ) can also be found and by integration enables the calculation of true anomaly ( $\theta$ ). All

Parameter	Value	Unit
$G$	$6.67408 \cdot 10^{-11}$	$\text{m}^3 \text{kg}^{-1} \text{s}^{-2}$
$M_{Earth}$	$5.9742 \cdot 10^{24}$	kg
$r_a$	$R_e + 1200$	km
$r_p$	$R_e + 800$	km
$R_e$	6378	km
$i$	75	°
$\Omega$	0	°
$\omega$	0	°

**Table 1.**  
 Parameters required for calculation of the orbital dynamics.

these values allow for calculating the rotation matrix from the orbit frame to the inertial frame ( $\mathbf{R}_o^i$ ), the radius vector ( $\mathbf{r}^o$ ), the velocity vector ( $\mathbf{v}^o$ ), the acceleration vector ( $\mathbf{a}^o$ ), angular velocity ( $\boldsymbol{\omega}_{i,o}^i$ ), and angular acceleration vector ( $\dot{\boldsymbol{\omega}}_{i,o}^i$ ). Hence, all the outputs from this subsystem can easily be found following this procedure and will be used in several other subsystems.

### 3. Attitude dynamics and control

#### 3.1 Attitude dynamics

The attitude dynamics can be derived with the basis in Euler's moment Equation ([1], p. 95). The angular momentum of a rigid body in the body frame is given as

$$\mathbf{h}^b = \mathbf{J}\boldsymbol{\omega}_{i,b}^b, \quad (8)$$

where  $\mathbf{J} \in \mathbb{R}^{3 \times 3}$  is the inertia matrix, while  $\boldsymbol{\omega}_{i,b}^b \in \mathbb{R}^3$  is the angular velocity of the body frame relative to the inertial frame. The angular momentum can be found in the inertial frame as

$$\mathbf{h}^i = \mathbf{R}_b^i \mathbf{h}^b. \quad (9)$$

The rate of change of angular momentum is equal to the total torque, such that  $\dot{\boldsymbol{\tau}}^i = \dot{\mathbf{h}}^i$ . Hence, by differentiating Eq. (9), it is obtained that

$$\dot{\boldsymbol{\tau}}^i = \dot{\mathbf{h}}^i = \mathbf{R}_b^i \mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{h}^b + \mathbf{R}_b^i \dot{\mathbf{h}}^b, \quad (10)$$

which can be written in the body frame by using Eq. (8) as  $\dot{\boldsymbol{\tau}}^b = \mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{J}\boldsymbol{\omega}_{i,b}^b + \mathbf{J}\dot{\boldsymbol{\omega}}_{i,b}^b$ , where the inertia matrix is assumed to be constant. Decomposing the total torque into an actuation component and a perturbing component,  $\boldsymbol{\tau}^b = \boldsymbol{\tau}_a^b + \boldsymbol{\tau}_p^b$ , allows the rotational dynamics to be written as

$$\mathbf{J}\dot{\boldsymbol{\omega}}_{i,b}^b = -\mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{J}\boldsymbol{\omega}_{i,b}^b + \boldsymbol{\tau}_a^b + \boldsymbol{\tau}_p^b, \quad (11)$$



where  $\boldsymbol{\tau}_a^b \in \mathbb{R}^3$  denotes the actuation torques (e.g., output from reaction wheels), while  $\boldsymbol{\tau}_p^b \in \mathbb{R}^3$  denotes the perturbing torques (e.g., gravity torque). Further, by using quaternion representation, the update law for the quaternion representing the attitude of the body frame relative to the inertial frame can be written as

$$\dot{\mathbf{q}}_{i,b} = \frac{1}{2} \mathbf{T}(\mathbf{q}_{i,b}) \begin{bmatrix} 0 \\ \boldsymbol{\omega}_{i,b}^b \end{bmatrix}. \quad (12)$$

Hence, Eqs. (11) and (12) serve as governing equations describing the attitude and angular velocity of the satellite. The inputs that affect these values are the perturbation and actuation torques, where the latter will be found in the following sections.

### 3.2 Error dynamics

From Euler's moment equation, the angular acceleration is defined relative to the inertial frame. For attitude control, it is often more interesting controlling the attitude and angular velocity relative to the orbit frame. For that reason, the angular velocity of the body frame relative to the orbit frame can be found as  $\boldsymbol{\omega}_{o,b}^b = \boldsymbol{\omega}_{i,b}^b - \mathbf{R}_i^b \boldsymbol{\omega}_{i,o}^i$ , which can be differentiated as

$$\mathbf{J} \dot{\boldsymbol{\omega}}_{o,b}^b = -\mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{J} \boldsymbol{\omega}_{i,b}^b + \boldsymbol{\tau}_a^b + \boldsymbol{\tau}_p^b + \mathbf{J} \mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{R}_i^b \boldsymbol{\omega}_{o,i}^i - \mathbf{J} \mathbf{R}_i^b \dot{\boldsymbol{\omega}}_{i,o}^i \quad (13)$$

giving a description of the attitude dynamics relative to the orbit frame. It is also possible to find the error dynamics, to enable tracking of a desired attitude and angular velocity. Let  $\mathbf{q}_{o,d}^d, \boldsymbol{\omega}_{o,d}^d, \dot{\boldsymbol{\omega}}_{o,d}^d \in \mathcal{L}_\infty$  denote a desired quaternion, angular velocity, and acceleration; then, the quaternion and angular velocity error can be found as

$$\mathbf{q}_{d,b} = \mathbf{q}_{d,o} \otimes \mathbf{q}_{o,b}, \quad (14)$$

$$\boldsymbol{\omega}_{d,b}^b = \boldsymbol{\omega}_{o,b}^b - \mathbf{R}_d^b \boldsymbol{\omega}_{o,d}^d, \quad (15)$$

with the kinematics as

$$\dot{\eta}_{d,b} = -\frac{1}{2} \boldsymbol{\varepsilon}_{d,b}^\top \boldsymbol{\omega}_{d,b}^b, \quad (16)$$

$$\dot{\boldsymbol{\varepsilon}}_{d,b} = (\eta_{d,b} \mathbf{I} + \mathbf{S}(\boldsymbol{\varepsilon}_{d,b})) \boldsymbol{\omega}_{d,b}^b. \quad (17)$$

The angular acceleration error can be found by differentiating Eq. (15) as

$$\mathbf{J} \dot{\boldsymbol{\omega}}_{d,b}^b = -\mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{J} \boldsymbol{\omega}_{i,b}^b + \boldsymbol{\tau}_a^b + \boldsymbol{\tau}_p^b + \mathbf{J} \mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{R}_i^b \boldsymbol{\omega}_{o,i}^i - \mathbf{J} \mathbf{R}_i^b \dot{\boldsymbol{\omega}}_{i,o}^i + \mathbf{J} \mathbf{S}(\boldsymbol{\omega}_{o,b}^b) \mathbf{R}_d^b \boldsymbol{\omega}_{o,d}^d - \mathbf{J} \mathbf{R}_d^b \dot{\boldsymbol{\omega}}_{o,d}^d. \quad (18)$$

Hence, the control objective can be defined as that of making  $(\mathbf{q}_{d,b}, \boldsymbol{\omega}_{d,b}^b) \rightarrow (\mathbf{0}, \mathbf{0})$ , which will make the satellite point in a desired direction and move with a desired angular velocity.

### 3.3 PD+ attitude controller

Takegaki and Arimoto [19] proposed in 1981 a simple method for position control of robots, something that was extended by [20] to enable trajectory tracking. The so-called PD+ controller has been applied for spacecraft by [21, 22] showing good results. The author has applied this method in previous works such as [23, 24].

In order to design a PD+ attitude controller, let a Lyapunov function candidate be chosen as  $V = \frac{1}{2} (\boldsymbol{\omega}_{d,b}^b)^T \mathbf{J} \boldsymbol{\omega}_{d,b}^b + k_p (1 - \eta_{d,b})^2 + k_p \boldsymbol{\epsilon}_{d,b}^T \boldsymbol{\epsilon}_{d,b}$  where  $k_p$  is a positive scalar gain. The derivative is found by using Eqs. (16)–(18) as

$$\begin{aligned} \dot{V} = & k_p \boldsymbol{\epsilon}_{d,b}^T \dot{\boldsymbol{\omega}}_{d,b}^b + (\boldsymbol{\omega}_{d,b}^b)^T \left( -\mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{J} \boldsymbol{\omega}_{i,b}^b + \boldsymbol{\tau}_a^b + \boldsymbol{\tau}_p^b + \mathbf{J} \mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{R}_i^b \boldsymbol{\omega}_{o,i}^i \right. \\ & \left. - \mathbf{J} \mathbf{R}_i^b \dot{\boldsymbol{\omega}}_{i,o}^i + \mathbf{J} \mathbf{S}(\boldsymbol{\omega}_{o,b}^b) \mathbf{R}_d^b \boldsymbol{\omega}_{o,d}^d - \mathbf{J} \mathbf{R}_d^b \dot{\boldsymbol{\omega}}_{o,d}^d \right). \end{aligned} \quad (19)$$

A PD+ attitude control law can now be chosen as

$$\begin{aligned} \boldsymbol{\tau}_d^b = & \mathbf{J} \mathbf{R}_d^b \dot{\boldsymbol{\omega}}_{o,d}^d - \mathbf{J} \mathbf{S}(\boldsymbol{\omega}_{o,b}^b) \mathbf{R}_d^b \boldsymbol{\omega}_{o,d}^d + \mathbf{J} \mathbf{R}_i^b \dot{\boldsymbol{\omega}}_{i,o}^i - \mathbf{J} \mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{R}_i^b \boldsymbol{\omega}_{o,i}^i - \boldsymbol{\tau}_p^b + \mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{J} \boldsymbol{\omega}_{i,b}^b \\ & - k_p \boldsymbol{\epsilon}_{d,b} - k_d \boldsymbol{\omega}_{d,b}^b, \end{aligned} \quad (20)$$

where  $k_d$  is another positive scalar gain and  $\boldsymbol{\tau}_d^b$  denotes the desired torque required to make the attitude and angular velocity errors go to zero. Assuming no actuator dynamics, i.e.,  $\boldsymbol{\tau}_a^b = \boldsymbol{\tau}_d^b$ , and then by inserting Eq. (20) into Eq. (19), it is obtained that  $\dot{V} \leq -k_d \|\boldsymbol{\omega}_{d,b}^b\|^2$ , which is negative semidefinite. By applying the Matrosov theorem (cf. [24]), it can be shown that the origin  $(\boldsymbol{\epsilon}_{d,b}, \boldsymbol{\omega}_{d,b}^b) = (\mathbf{0}, \mathbf{0})$  is uniformly asymptotically stable.

The inputs to the control law (Eq. 20) are the desired states  $\mathbf{q}_{o,d}$ ,  $\boldsymbol{\omega}_{o,d}^d$ , and  $\dot{\boldsymbol{\omega}}_{o,d}^d$ , which are to be defined by the reader, e.g., as part as a guidance block depending on the mission objective. The inertia matrix ( $\mathbf{J}$ ) is assumed to be known, while the angular velocity vector between the body frame and orbit frame ( $\boldsymbol{\omega}_{o,b}^b$ ) can be found as described above. The other angular velocities are found from the orbital mechanics, while the rotation matrices are found as composite rotations, e.g.,  $\mathbf{R}_i^b = \mathbf{R}_o^b \mathbf{R}_i^o$ , or by using the relationship between the quaternions and rotation matrices directly (cf. Section 2A). The error quaternion and angular velocity are found from Eqs. (14) and (15), while the perturbing torques will be described in the following section.

## 4. Perturbing torques

There are different kinds of perturbing torques, such as gravity torque, aerodynamic torque, magnetic field due to the electronics inside the satellite, as well as solar radiation torque. This section only considers the gravity torque. In [16], p. 147, the gravity torque is defined as

$$\boldsymbol{\tau}_g^b = \frac{3GM_{Earth}}{r^5} \mathbf{r}^i \times \mathbf{J} \mathbf{r}^i, \quad (21)$$



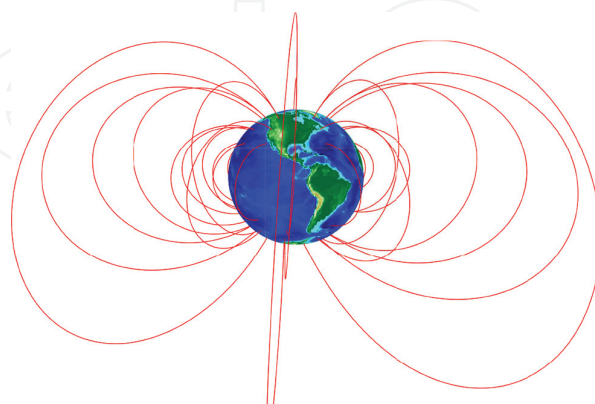
where the terms have previously been defined. As can be seen from this equation, non-diagonal inertia matrices will induce gravitational torques to align the satellite with the gravity field. This is also sometimes used for passive control, using e.g., a gravity boom to ensure that one side of the satellite is always facing the Earth. For this chapter, the perturbing torque is set equal to the gravity torque such that  $\tau_p^b = \tau_g^b$ .

## 5. Actuators

The control signal must be mapped to an actuator that must generate the desired torque. With limitations in actuation, the saturation must be modeled in order to obtain realistic results when simulating attitude control. This section considers three types of actuators commonly used for spacecraft attitude control: magnetic torquers, reaction wheels, and thrusters.

### 5.1 Magnetic torquers

Magnetic torquers operate by creating a local magnetic field that interacts with the Earth's magnetic field. In simple terms, magnetic torques can be explained as that of a compass needle. By applying current through a coil, a local magnetic field is created, which will try to align itself with the Earth's magnetic field. This allows the attitude of a spacecraft to be changed and is a very popular approach for small satellites, e.g., cubesats. One of the drawbacks or challenges with magnetic actuation lies in the fact that the Earth's magnetic field goes from the North Pole to the South Pole as shown in **Figure 2**.<sup>1</sup> As can be seen, when the satellite crosses the North Pole, there will be mainly a downward magnetic field component, reducing the possibility of actuation to only two axes, and similarly along the equator. This subsection is based on [12] and will describe how to model magnetic torquers and how it can be applied for attitude control. A magnetic torquer produces a magnetic torque by applying a current through a coil, which can be expressed as [2].



**Figure 2.**

*Magnetic field of the Earth visualized using the IGRF model. The control torque using magnetic torquers is always perpendicular to the magnetic field, such that at the poles, only roll, and pitch control are available, while at the equator, only pitch and yaw control is available [25].*

<sup>1</sup> Figure created using the MATLAB script “international geomagnetic reference field (IGRF) model” by Drew Compston.

$$\boldsymbol{\tau}_m^b = \mathbf{S}(\mathbf{m}^b)\mathbf{b}^b, \quad (22)$$

where  $\mathbf{m}^b = [m_x \ m_y \ m_z]^T = NA [i_x \ i_y \ i_z]^T$  is the induced magnetic field,  $N$  is the number of turns of the coils,  $i_{(\cdot)}$  is the current around a given axis, and  $A$  is the area of the coils. The Earth's magnetic field is represented through the vector  $\mathbf{b}^b$ , meaning that to use magnetic actuation for attitude control, a good model of the Earth's magnetic field is needed.

From a control point of view, the physical parameters  $A$  and  $N$  are defined when the spacecraft is designed, such that the controller needs to dictate which currents that must be sent to the torquers in order to get a desired torque. This means that Eq. (22) must be inverted with regard to  $\mathbf{m}^b$ , which is not straightforward due to the cross product, meaning that you obtain rank 2 when inverting the right-hand side, losing information about one of the axes. To that end, an approximation to inverting this equation is given in [25].

$$\mathbf{m}^b = \frac{\mathbf{S}(\mathbf{b}^b)\boldsymbol{\tau}_d^b}{\|\mathbf{b}^b\|^2}, \quad (23)$$

enabling the currents to be found as

$$[i_x \ i_y \ i_z]^T = \frac{1}{NA} [m_x \ m_y \ m_z]^T. \quad (24)$$

It is here assumed that all three torquers are identical, but depending on satellite configuration, there might be differences in the number of turns and areas. Hence, the desired torque  $\boldsymbol{\tau}_d^b$  can be used to find the magnetic moment  $\mathbf{m}^b$  in Eq. (23) and solved for the currents and applied resulting in the actuation torque in Eq. (22). Hence, the control law (Eq. 20) can be mapped to a desired magnetic moment (Eq. 23), which then can be used to find the desired current to each of the three coils. Then, the limits in current will dictate the maximum magnetic moment that can be generated.

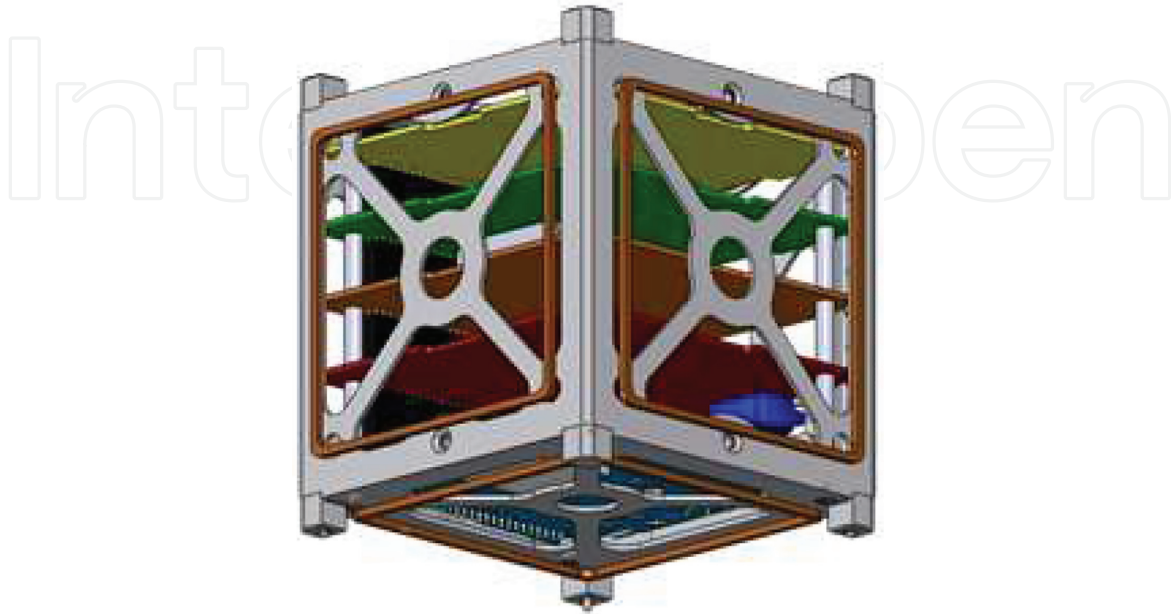
Consider the HiNCube satellite as shown in **Figure 3**. The cubesat comprises three orthonormal magnetic torquers with an area  $A = 0.00757 \text{ m}^2$  and with a maximum current of 47.27 mA and  $N = 100$  turns. This gives a maximum magnetic moment of  $m_{max} = 0.03578 \text{ mA}^2$ . Hence, an implementation of using magnetic torquers for attitude control would encompass mapping the output from the control law to a desired magnetic moment using Eq. (23) and then imposing the maximum magnetic moment on each axis, before finding the resulting actuation torque using Eq. (22). Note that to ensure sign correctness due to the projection, the actuation torque can be found as

$$[\tau_{x,a} \ \tau_{y,a} \ \tau_{z,a}]^T = [\text{sign}(\tau_{x,d})\tau_{x,m} \ \text{sign}(\tau_{y,d})\tau_{y,m} \ \text{sign}(\tau_{z,d})\tau_{z,m}]^T, \quad (25)$$

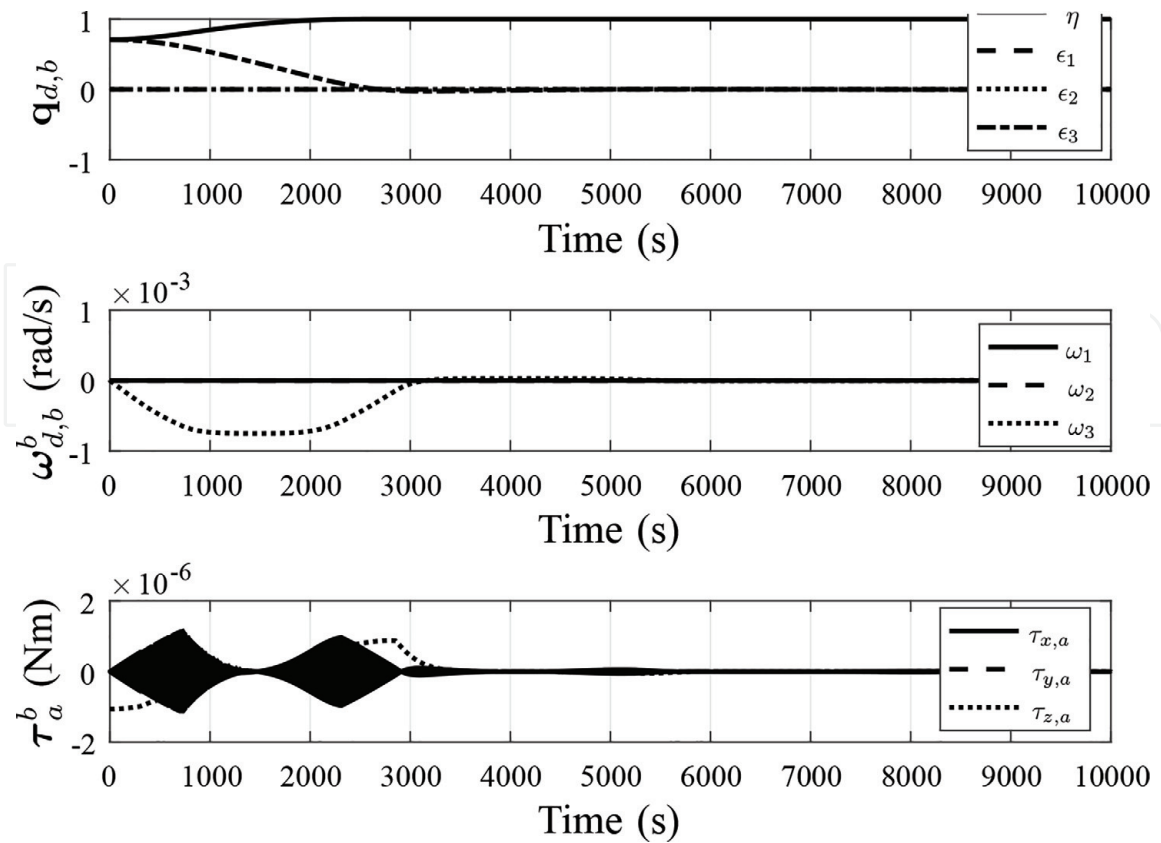
where  $\boldsymbol{\tau}_a^b = [\tau_{x,a} \ \tau_{y,a} \ \tau_{z,a}]^T$ ,  $\boldsymbol{\tau}_d^b = [\tau_{x,d} \ \tau_{y,d} \ \tau_{z,d}]^T$ , and  $\boldsymbol{\tau}_m^b = [\tau_{x,m} \ \tau_{y,m} \ \tau_{z,m}]^T$ .

To show the performance of magnetic torquers, consider again the HiNCube satellite, which had an inertia matrix of  $\mathbf{J} = 1.67 \cdot 10^{-3} \mathbf{I} \text{ kg m}^2$ . Consider the problem of making rotating 90° from an initial quaternion  $\mathbf{q}_{o,b} = [0 \ 0 \ 0 \ 1]^T$  to  $\mathbf{q}_{o,d} = [1 \ 0 \ 0 \ 0]^T$ . The gains for the PD+ controller are set  $k_p = 1 \cdot 10^{-5}$  and

$k_d = 5 \cdot 10^{-3}$ , and the satellite is assumed to have an orbit of  $r_p = 500$  km and  $r_a = 600$  km, with inclination of  $75^\circ$ . **Figure 4** shows the attitude, angular velocity, and actuation torque. It is evident that magnetic torquers produce very low torque, such that it takes a very long time to change the attitude of the spacecraft (about 1 h). To some extents, this can be improved by being in a lower orbit where the magnetic



**Figure 3.**  
Magnetic torquers on the HiNCube satellite (shown in brown).



**Figure 4.**  
Quaternion error, angular velocity error, and actuation torque using magnetic torquers.

field is stronger or by using larger coils with higher currents. Also, note that the actuation signal varies in strength as a function of time, depending on the orbital position of the satellite.

## 5.2 Reaction wheels

Another way of changing the attitude of a satellite is through reaction wheels. Reaction wheels are based on the principle of Newton's third law: *When one body exerts a force on a second body, the second body simultaneously exerts a force equal in magnitude and opposite in direction on the first body.* This means that by spinning a reaction wheel in one direction, the satellite will rotate in the other direction. Mounting three reaction wheels in an orthogonal configuration enables three-axis attitude control of spacecraft. From Newton's third law, the momentum generated by the reaction wheels will have opposite sign of the momentum of the satellite, such that  $\dot{\mathbf{h}}^i = -\dot{\mathbf{h}}_w^i$  where  $\dot{\mathbf{h}}_w^i$  is the momentum production by the reaction wheels,  $\dot{\mathbf{h}}^i$  is the momentum acting on the satellite, and  $\boldsymbol{\tau}_w^b$ . By employing Euler's moment equation similarly as in Section 3, the torque generated by the reaction wheels can be found by differentiating  $\mathbf{h}_w^i = \mathbf{R}_b^i \mathbf{h}_w^b$  with  $\mathbf{h}_w^b = \mathbf{J}_w \boldsymbol{\omega}_w^b$  where  $\boldsymbol{\omega}_w^b$  is the angular velocity of the reaction wheels and  $\mathbf{J}_w$  denotes their inertia. This gives

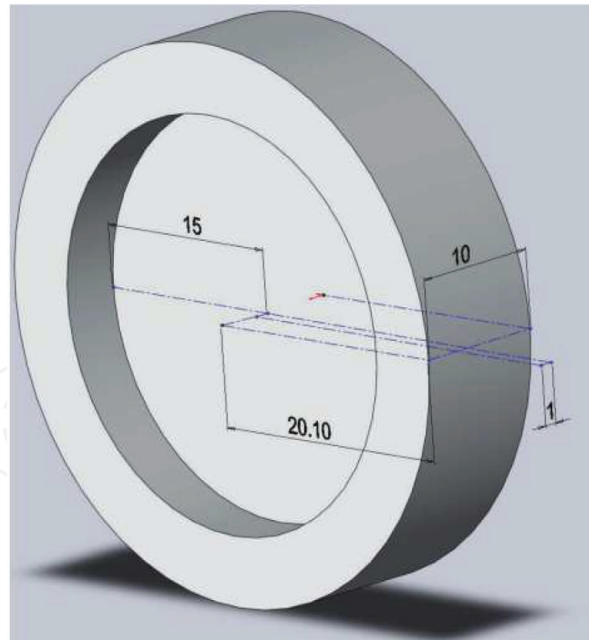
$$\boldsymbol{\tau}_a^b = -\dot{\mathbf{h}}_w^b - \mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{h}_w^b, \quad (26)$$

where  $\boldsymbol{\tau}_w^b = \dot{\mathbf{h}}_w^b$  is the torque generated by the reaction wheels. Now, consider a set of three orthonormal reaction wheels, where one produces torques around the  $x$ -axis, one around the  $y$ -axis, and one around the  $z$ -axis of the body frame. Then, the PD+ control law dictates a desired torque,  $\boldsymbol{\tau}_d^b$ , which shall be achieved by the reaction wheels. To that end, the torque by the reaction wheel can be rewritten as  $\boldsymbol{\tau}_w^b = \boldsymbol{\tau}_d^b - \mathbf{S}(\boldsymbol{\omega}_{i,b}^b) \mathbf{h}_w^b$ , where  $\boldsymbol{\tau}_w^b$  must be bounded by the motor torque limit, while the angular momentum will be bounded as a function of maximum rotational speed. After imposing the torque and speed constraints, the angular momentum of the reaction wheels is found by integrating  $\dot{\mathbf{h}}_w^b$  allowing the actuation torque to be calculated using Eq. (26).

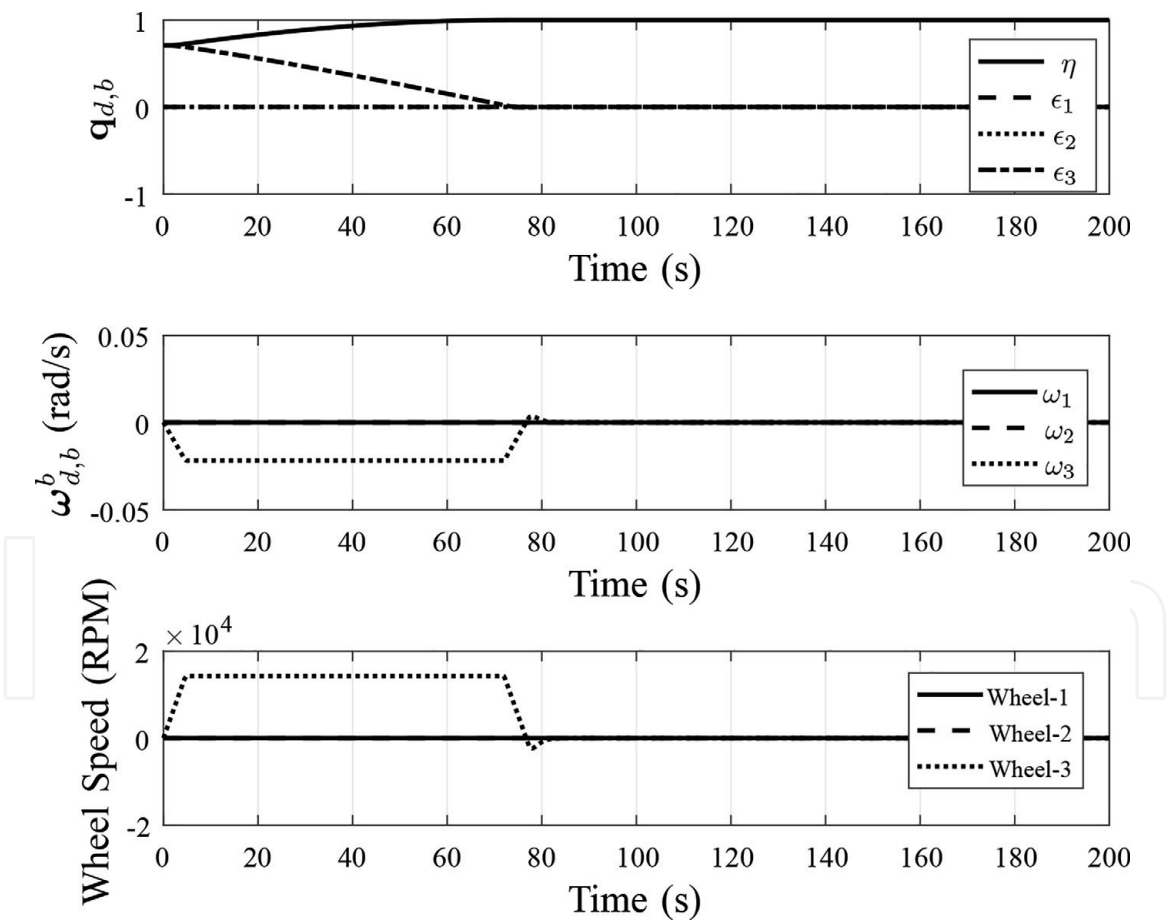
Consider the HiNCube satellite again, where it is possible to use three small reaction wheels as described in [11] where the main idea is to place most of the mass away from the center as shown in **Figure 5**. The inertia of an individual reaction wheel was found to be  $J_w = 1.46 \cdot 10^{-5} \text{ kg m}^2$ , and by assuming a maximum rotation speed of 13,700 rpm with maximum torque of  $\tau_{max} = 0.0047 \text{ Nm}$ , the maximum momentum generated by the reaction wheels is found as

$$h_{max} = J_w \omega_w = 1.46 \cdot 10^{-5} \cdot 13700 \cdot \frac{2\pi}{60} = 1.52389 \cdot 10^{-6}.$$

Now, consider the same simulation as when using the magnetic torquers, where the gains for the PD+ controller is changed to  $k_p = k_d = 2$  and the reaction wheels has the limits as defined above. **Figure 6** shows the simulation results, where it is obvious that by using reaction wheels, the satellite is able to change its orientation after about 80 s. To some extent, this can be credited to the higher gains, but it lies mainly with the better actuation system that is able to produce higher torque than the reaction wheels. From the figure, the reaction wheels quickly go into saturation of 13,700 RPM, where the angular velocity also goes into saturation. As the quaternion error goes toward zero, the reaction wheel despin, reducing the angular velocity and the control objective, is completed.



**Figure 5.**  
Example design of a reaction wheel for cubesats (dimensions are in mm) [11].



**Figure 6.**  
Quaternion error, angular velocity error, and wheel speeds when using reaction wheels for attitude control.

### 5.3 Thrusters

The third kind of actuator that will be studied is using reaction control thrusters. This section presents how to map the control signal (Eq. 20) to four thrusters used for attitude control. Let the location of each thruster be denoted by



$\mathbf{r}_i^b = [r_x \ r_y \ r_z]^T$ , and let them have an azimuth and an elevation angle described by  $\chi$  and  $\gamma$ . Then, the torque produced by a given thruster can be found as [1], p. 262

$$\boldsymbol{\tau}_i^b = \mathbf{r}_i^b \times \mathbf{f}_i^b = \begin{bmatrix} r_y \sin(\chi) \cos(\gamma) - r_z \sin(\gamma) \\ r_z \cos(\gamma) \cos(\chi) - r_x \cos(\gamma) \sin(\chi) \\ r_x \sin(\gamma) - r_y \cos(\gamma) \cos(\chi) \end{bmatrix} f_i, \quad (27)$$

where  $f_i$  denotes the total thrust from the  $i$ th thruster. Given the thruster configuration defined in **Table 2**, let the vector of thruster signals be denoted  $\mathbf{u} = [f_1 \ f_2 \ f_3 \ f_4]^T$ , and then the torque can be found as  $\boldsymbol{\tau}_a^b = \mathbf{B}\mathbf{u}$  with

$$\mathbf{B} = \begin{bmatrix} -\frac{\sqrt{2}}{5} & \frac{\sqrt{2}}{5} & \frac{\sqrt{2}}{5} & -\frac{\sqrt{2}}{5} \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{4} \\ -\frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} \end{bmatrix}. \quad (28)$$

Given a desired torque from the PD+ control law, it must be mapped to the desired thruster firings, such that the combination of thrusters produces the desired torque. To that end, there are several different modulation methods that can be applied, ranging from a simple bang-bang modulation to more sophisticated pulse-width pulse-frequency modulation. This section will give an introduction to the different methods and detail how they can be implemented. In general the desired torque can be mapped to the desired thruster firings as  $\mathbf{u}_d = \mathbf{B}^\dagger \boldsymbol{\tau}_d^b$  where  $\dagger$  denotes the Moore-Penrose pseudoinverse and  $\mathbf{u}_d = [u_1 \ u_2 \ u_3 \ u_4]^T$  denotes the magnitude of each of the thrusters.

1. *Bang-bang modulation*: The easiest approach to thruster firings is bang-bang modulation, where the thruster is fully actuated as long as the  $i$ th signal of  $\mathbf{u}_d$  is above zero, such that

$$f_i = \begin{cases} f_{max} & \text{if } u_i > 0 \\ 0 & \text{if } u_i \leq 0 \end{cases}, \quad (29)$$

where  $f_{max}$  denotes the maximum available thrust from the  $i$ th thruster. After applying bang-bang modulation, the vector  $\mathbf{u}$  can be constructed allowing the actuator torque to be found as  $\boldsymbol{\tau}_a^b = \mathbf{B}\mathbf{u}$ .

Thruster	Elevation ( $\gamma$ )	Azimuth ( $\chi$ )	$r_x$	$r_y$	$r_z$
$f_1$	45	90	-0.5	-0.45	-0.05
$f_2$	135	90	-0.5	-0.45	0.05
$f_3$	-45	90	-0.5	0.45	-0.05
$f_4$	-135	90	-0.5	0.45	0.05

**Table 2.**  
 Thruster configuration.



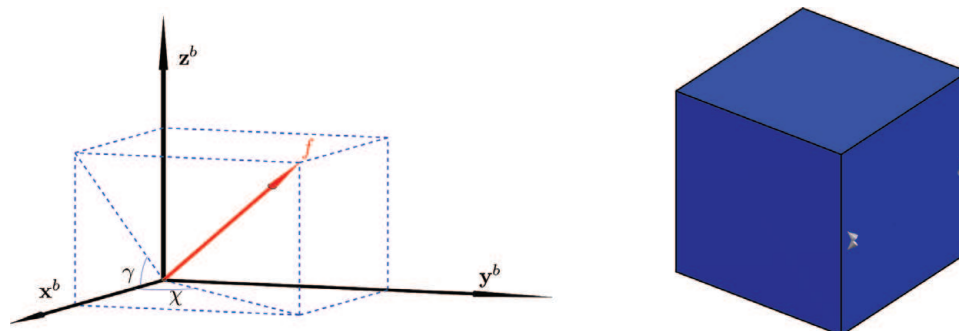
2. *Bang-bang modulation with dead zone*: One of the major drawbacks by using simple bang-bang modulation is when the tracking error has converged to zero, where the thruster firings will continue to maintain the desired attitude. Sensor noise is another source that leads to continuous firings, quickly spending all the propellant. To that end, bang-bang modulation can be augmented with a dead zone, giving

$$f_i = \begin{cases} f_{max} & \text{if } u_i > D \\ 0 & \text{if } u_i \leq D \end{cases} \quad (30)$$

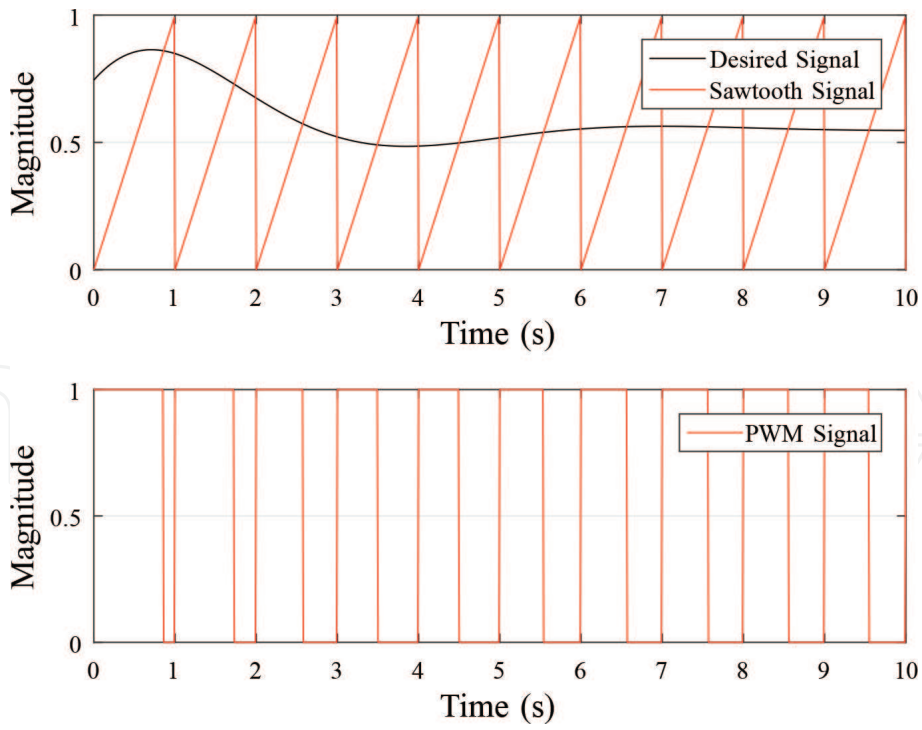
where  $D > 0$  denotes the dead zone. By properly selecting a suitable dead zone enables the thrusters to avoid firing when close to the equilibrium point.

3. *Pulse-width modulation*: Another approach that is often used for thruster firings is by using pulse-width modulation (PWM), where an analogue signal (desired torque) can be mapped to discrete signals using PWM. Instead of changing the thrust level, the duration of the pulses can be changed, leading to a pulse that is proportional to the torque command from the PD+ controller. A simple way of achieving this is by using the intersective technique, which uses a sawtooth signal that is compared to the control signal. When the sawtooth is less than the control signal, the PWM signal is in a high state and otherwise in a low state. This makes it possible to go from continuous control signal to a discrete representation which can be used for thruster firings. **Figure 7** shows how to achieve the PWM signal, enabled through a simple comparison of the two signals.

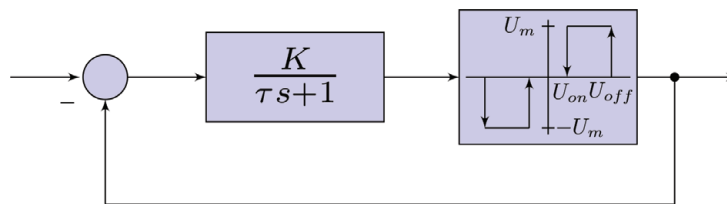
4. *Pulse-width pulse-frequency modulation*: In addition to controlling the width of the pulse as in PWM, it is also possible to control the frequency of the pulse—something that is done through pulse-width pulse-frequency (PWPF) modulation ([1], p. 265) (**Figure 8**). The modulation approach comprises a lag filter and a Schmitt trigger as shown in **Figure 9**. As long as the input to the Schmitt trigger is below  $U_{on}$ , the output is kept at zero and must be larger than  $\frac{U_{on}}{K}$  to produce an output, where  $K$  is a DC gain,  $\tau$  is the time constant,  $U_{on}$  and  $U_{off}$  are the on and off limits for the Schmitt trigger, while  $U_m$  is the maximum output. Much research has been performed on improving PWPF modulation, and in [26], the authors propose the following settings (cf. **Figure 9**):  $2 < K < 6$ ,  $0.1 < \tau < 0.5$ ,  $U_{on} > 0.3$ ,  $U_{off} < 0.8U_{on}$ , and  $U_m = 1$ .



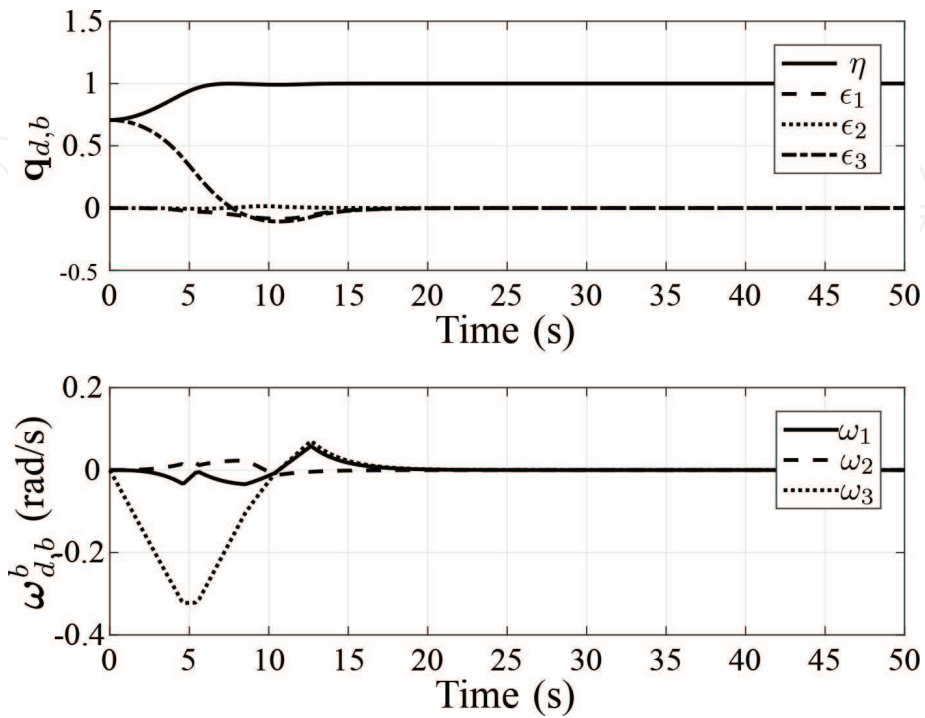
**Figure 7.** Thruster configuration. The left subfigure shows the definition of azimuth and elevation angles used to dictate the orientation of the thruster, while the right subfigure shows a satellite with thrusters placed and oriented as given in **Table 2**.



**Figure 8.**  
 Achieving pulse-width modulated signals for thruster firings.



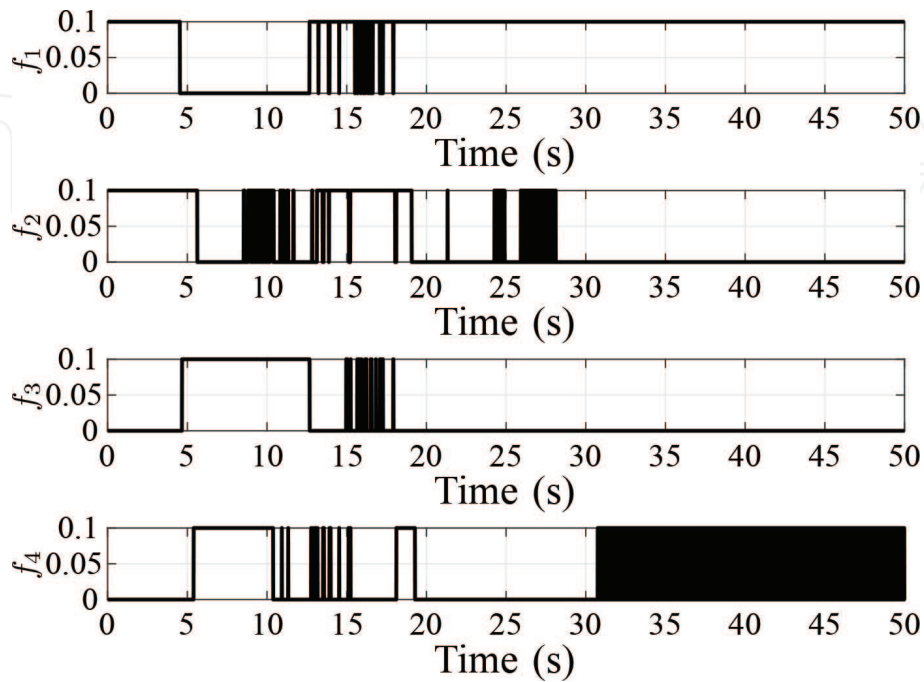
**Figure 9.**  
 Pulse-width pulse-frequency modulation.



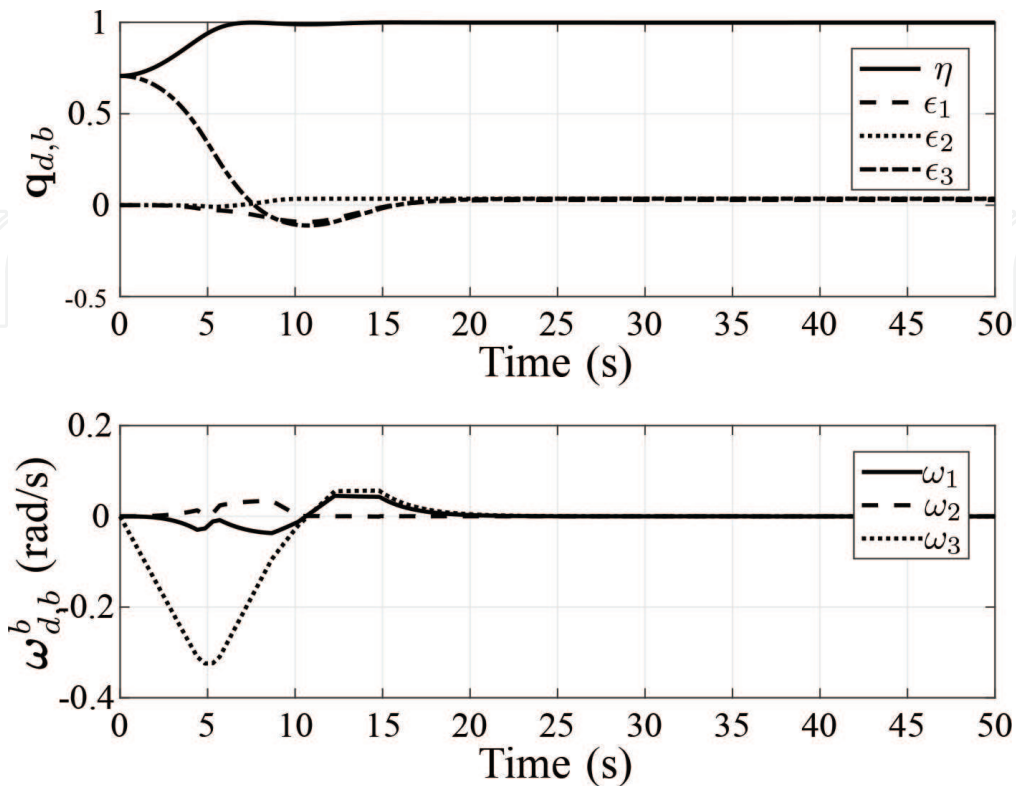
**Figure 10.**  
 Attitude control using thrusters with bang-bang modulation.

5. *Simulations of thruster modulations*: Consider a satellite with an inertia matrix as

$$J = \begin{bmatrix} 0.5 & -0.2 & -0.1 \\ -0.2 & 0.5 & -0.2 \\ -0.1 & -0.2 & 0.5 \end{bmatrix}, \quad (31)$$

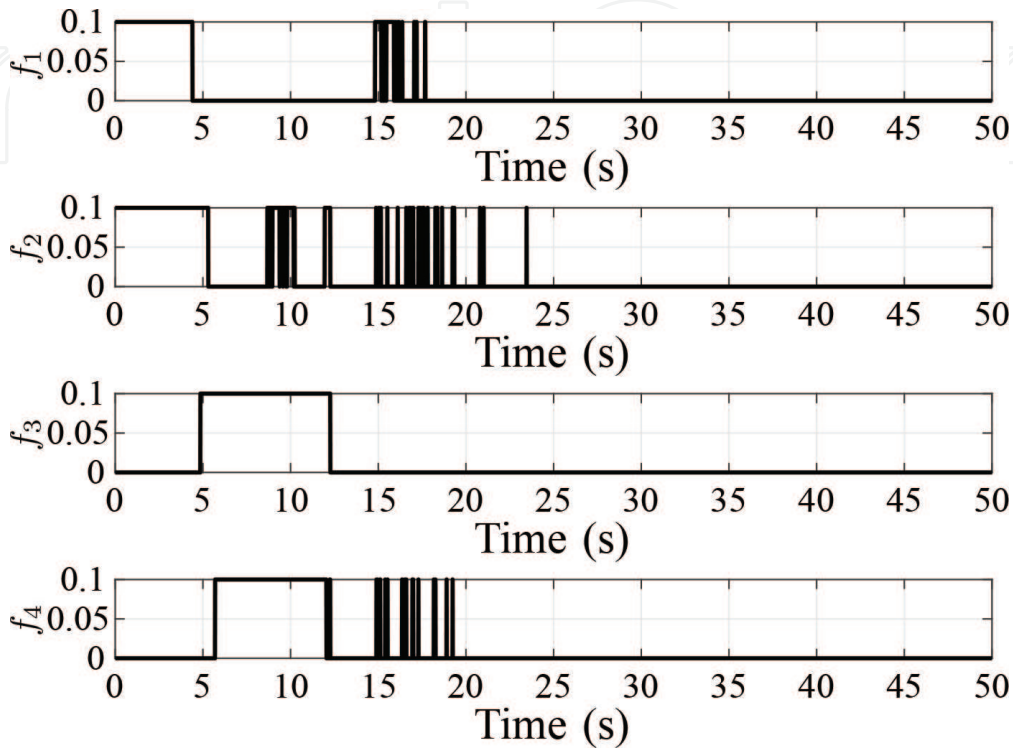


**Figure 11.**  
*Thruster firings when using bang-bang modulation.*

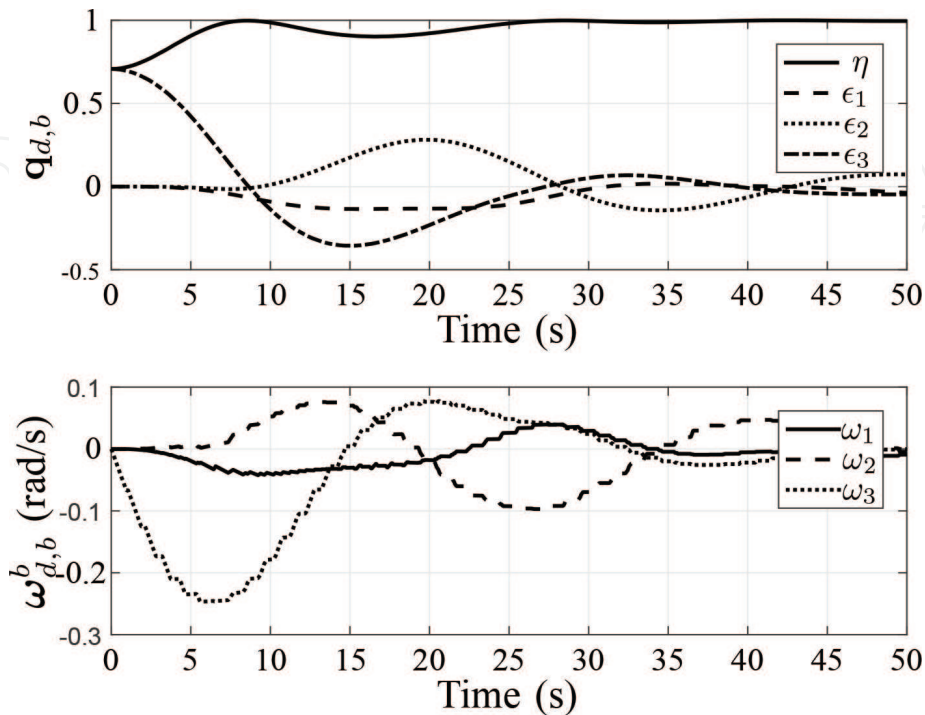


**Figure 12.**  
*Attitude control using thrusters with bang-bang modulation with dead zone.*

where the objective is to perform a yaw maneuver of  $90^\circ$  using four thrusters with 0.1 N force, with a specific impulse of 200 s. **Figure 10** shows the attitude and angular velocity vectors when using bang-bang modulation, where the satellite is able to make the errors go to zero. However, due to the modulation, the thrusters will continue firing as shown in **Figure 11**. To that end, consider the bang-bang modulation with dead zone. Let the dead zone be chosen as  $D = 0.05$ , and then the satellite obtains an accuracy as shown in **Figure 12** where there is a small



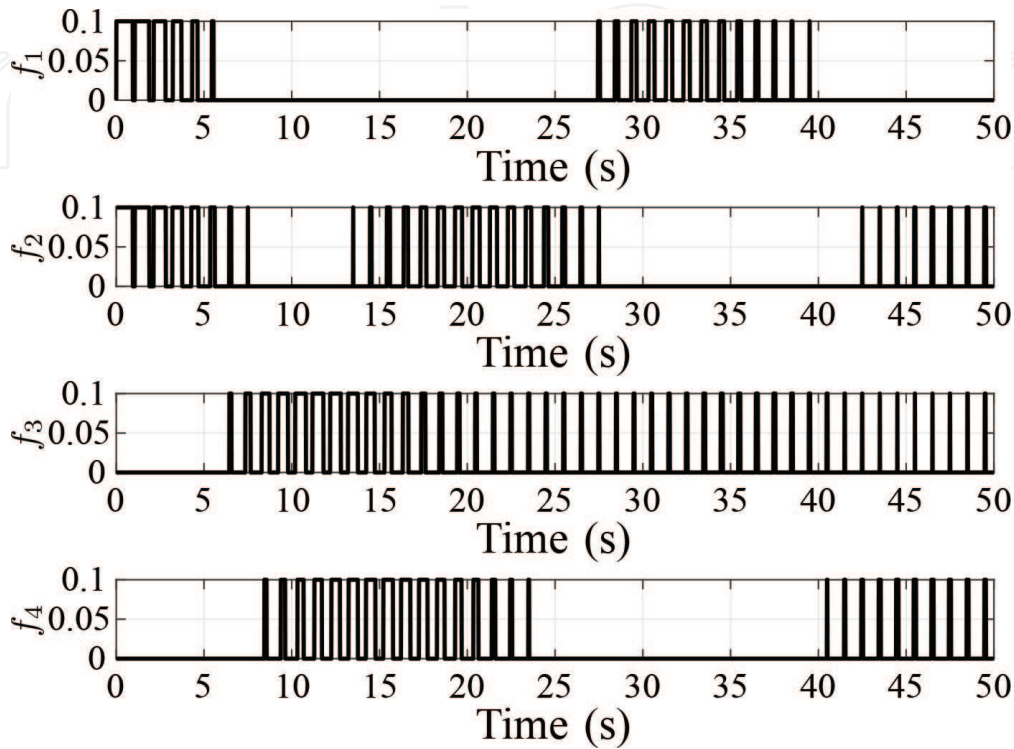
**Figure 13.**  
 Thruster firings when using bang-bang modulation with dead zone.



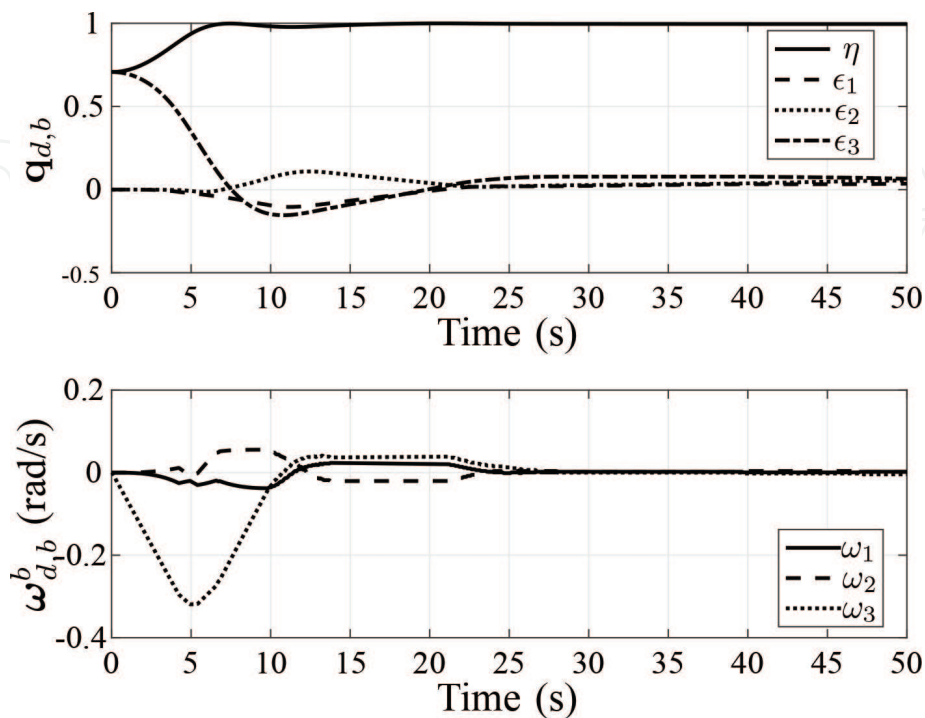
**Figure 14.**  
 Attitude control using thrusters with PWM modulation.

offset from the origin which is proportional to the dead zone. On the other hand, the thruster firings are much less prone to do continuous firings as shown in **Figure 13**.

Now, consider pulse-width modulation. Let the sawtooth signal have an amplitude of 1 and a frequency of 1 Hz. Then, the attitude and angular velocity error is obtained as shown in **Figure 14**, while the thruster firings are shown in **Figure 15**. It is possible to tune on sawtooth frequency to improve the performance.



**Figure 15.**  
Thruster firings when using PWM modulation.

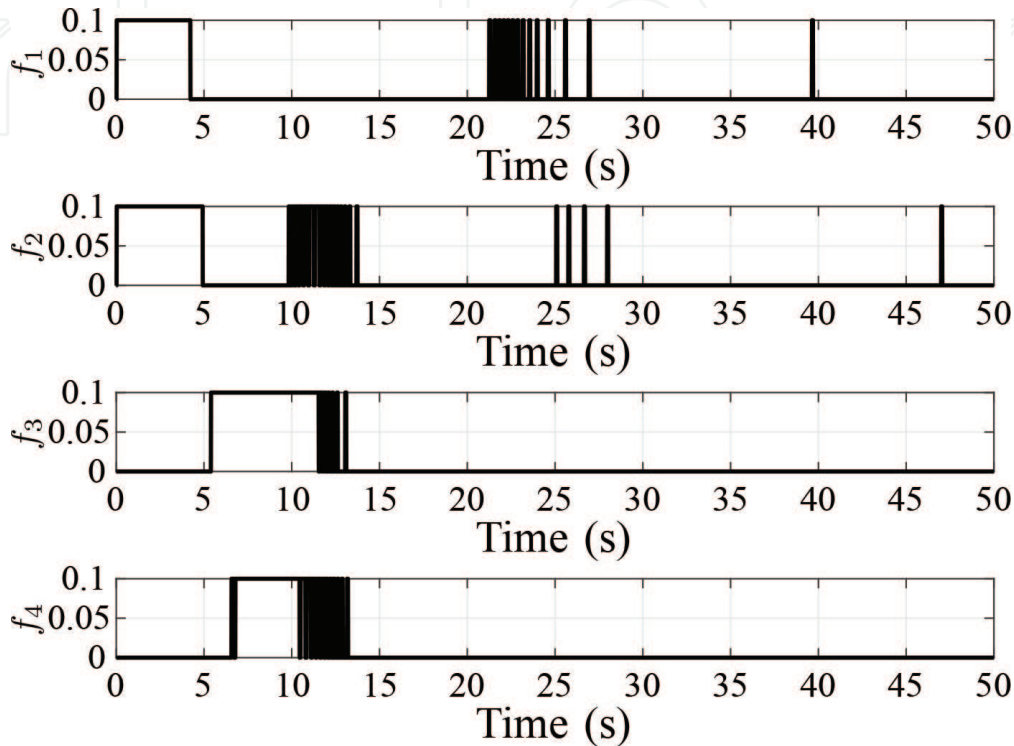


**Figure 16.**  
Attitude control using thrusters with PWPF modulation.

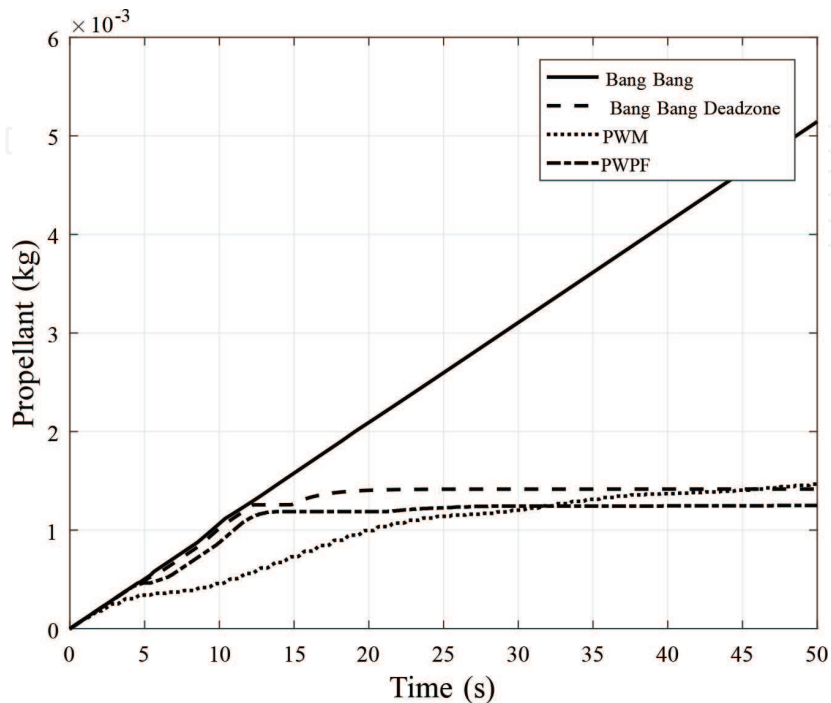


The final scenario is using PWPF modulation, where the parameters are chosen as  $K = 3$ ,  $\tau = 0.2$ ,  $U_{on} = 0.35$ , and  $U_{off} = 0.28$ . **Figure 15** shows the attitude and angular velocity, which go close to zero, while the thruster firings are shown in **Figure 16**, which is able to constrain the amount of thruster firings, and therefore propellant.

For satellite control using thrusters, propellant is a critical resource that must not be wasted. To that end it is desirable to limit the amount of propellant while at the same time obtain good pointing accuracy (**Figure 17**). With the basis in



**Figure 17.**  
 Thruster firings when using PWPF modulation.



**Figure 18.**  
 Propellant consumption of the different modulation methods.



Tsiolkovsky's rocket equation, the propellant consumption during thruster firings can be found as  $m_{propellant} = \int_0^t \frac{f}{I_{sp}g} dt$  where  $f$  is the force from one of the thrusters and  $I_{sp}$  is the specific impulse, while  $g = 9.81 \text{ m/s}^2$  is the acceleration due to gravity. **Figure 18** shows a comparison between the different modulation methods, where it is evident that the PWPF method allows for the least amount of propellant while obtaining close to acceptable performance. The bang-bang modulation will continue spending propellant until running out of fuel but on the other hand obtains the best tracking performance.

## 6. Attitude determination

As a preliminary step before trying to estimate the attitude of the satellite, some knowledge of measurement vectors must be known, i.e., what is the direction toward the Sun and how does the magnetic field vector look like at a given position. There are several other quantities that can be measured to obtain the attitude, where star trackers are known to be the most accurate. For the reader to obtain insight into using multiple measurements and combine it to find the attitude, this work presents a Sun vector model and a simplified magnetic field model that can be used for simulation purposes.

### 6.1 Sun vector model

To find the direction toward the Sun, there are several models that can be applied. The simplest would be to divide a circle into 365 days and have a vector always point toward the Sun. Then, by knowing which day it is, it is straightforward to find the direction toward the Sun. This approach would be coarse, such that more accurate models exist. For example, the Sun vector model in [3], pp. 281–282, has an accuracy of  $0.01^\circ$  and is valid until 2050. First, the time and date must be converted into the Julian date as [3], p. 189.

$$JD = 367(yr) - INT\left(\frac{7(yr + INT(\frac{mo+9}{12}))}{4}\right) + INT\left(\frac{275mo}{9}\right) + d$$

$$+ 1,721,013.5 + \frac{(\frac{s}{60} + \min)}{60} + \frac{h}{24}, \quad (32)$$

where a real truncation is denoted by  $INT()$  and the year, month, day, hour, minute, and second are denoted by  $yr, mo, d, h, \min, s$ . If the day contains a leap second, 61 s should be used instead of  $60^*$ . This gives the Sun vector model as

$$T_{UT1} = \frac{JD - 2,451,545.0}{36,525}, \quad (33)$$

$$\lambda_{M_\odot} = 280.460^\circ + 36,000.771T_{UT1}, \quad (34)$$

$$M_\odot = 357.5277233^\circ + 35,999.05034T_{UT1}, \quad (35)$$

$$\lambda_{ecliptic} = \lambda_{M_\odot} + 1.914666471^\circ \sin(M_\odot) + 0.019994643 \sin(2M_\odot), \quad (36)$$

$$\varepsilon = 23.439291^\circ - 0.0130042T_{UT1}, \quad (37)$$

$$\mathbf{s}^o = \mathbf{R}_i^o \begin{bmatrix} \cos(\lambda_{ecliptic}) \\ \cos(\varepsilon) \sin(\lambda_{ecliptic}) \\ \sin(\varepsilon) \sin(\lambda_{ecliptic}) \end{bmatrix}. \quad (38)$$

Here, the number of Julian centuries is denoted by  $T_{UT1}$ , the mean longitude of the Sun is denoted by  $\lambda_{M_\odot}$ , the mean anomaly for the Sun is denoted by  $M_\odot$ , the ecliptic longitude is denoted by  $\lambda_{ecliptic}$ , and the obliquity of the ecliptic is denoted by  $\varepsilon$ , while the Sun vector in orbit frame is denoted by  $\mathbf{s}^o$ .

## 6.2 Magnetic field model

Several different geomagnetic models can be applied for attitude determination in conjunction with a magnetometer. The most basic are simple dipole models [27], while more advanced are, e.g., the chaos model or the 12th generation IGRF model [28], which is the most commonly used model for attitude determination. This section presents the simple dipole model by [27], which can be described by the magnetic field vector in orbit frame as

$$\mathbf{m}^o = \frac{\mu_f}{a^3} [\cos(\omega_0 t) \sin(i) \quad -\cos(i) \quad 2 \sin(\omega_0 t) \sin(i)]^T, \quad (39)$$

where the time measured from passing the ascending node of the orbit relative to the geomagnetic equator is denoted by  $t$  and the dipole strength is denoted  $\mu_f = 7.9 \cdot 10^{-15}$  Wb-m, while  $\omega_0 = \|\boldsymbol{\omega}_{i,o}^i\|$  denotes the angular speed of the orbit. For a real application, the reader is recommended to apply the IGRF model, which is available in Simulink inside the Aerospace Toolbox, as C++ implementation<sup>2</sup> or using Python.<sup>3</sup>

## 6.3 Attitude determination using the Madgwick filter

The objective of attitude determination is to find what direction the satellite is pointing. In its core, it mainly requires two vector measurements and two mathematical models that can be compared and used to find the attitude. There are several different kinds of filters applied for attitude estimation, such as the Triad method [29], the Kalman filter [30], or the Mahony filter [31]. The Madgwick filter by [13] has shown good results in attitude estimation based on IMU measurements and is commonly applied in drone applications. The main idea by the filter is to use gradient descent in combination with the complementary filter to fuse sensor data together to produce an estimated quaternion. This section presents an application of the Madgwick filter by using measurements of the Sun vector and the magnetic field vector as well as the acceleration vector (gravity) and shows how to fuse that data together to estimate the attitude of a satellite in an elliptical orbit.

Let the quaternion estimate be denoted by  $\hat{\mathbf{q}} = [q_1 \quad q_2 \quad q_3 \quad q_4]^T$ . The measured acceleration, Sun vector, and magnetic field vectors can be defined, respectively, as  $\mathbf{a}^b$ ,  $\mathbf{s}^b$ , and  $\mathbf{m}^b$  and can be combined with the mathematical models of the acceleration, Sun vector, and magnetic field vector given in Eqs. (6), (38), and (39) to estimate the attitude. Here, the current estimate is denoted by subscript  $k$ , while the previous estimate is denoted by  $k - 1$ . Let the objective function be

<sup>2</sup> <https://github.com/JDeeth/MagDec>

<sup>3</sup> <https://github.com/scivision/pyigrf12>

$$\mathcal{F} = \begin{bmatrix} \mathbf{f}(\hat{\mathbf{q}}_{k-1}, \mathbf{a}^o, \mathbf{a}^b) \\ \mathbf{f}(\hat{\mathbf{q}}_{k-1}, \mathbf{s}^o, \mathbf{s}^b) \\ \mathbf{f}(\hat{\mathbf{q}}_{k-1}, \mathbf{m}^o, \mathbf{m}^b) \end{bmatrix}, \quad (40)$$

where the objective is found in an estimated quaternion that minimizes this function, something that can be achieved by using gradient descent. The Jacobian matrix can be found as

$$\mathcal{J} = \begin{bmatrix} \mathbf{J}_q(\hat{\mathbf{q}}_{k-1}, \mathbf{a}^o) \\ \mathbf{J}_q(\hat{\mathbf{q}}_{k-1}, \mathbf{s}^o) \\ \mathbf{J}_q(\hat{\mathbf{q}}_{k-1}, \mathbf{m}^o) \end{bmatrix} \quad (41)$$

and allows the gradient to be found as  $\nabla \mathbf{f} = \mathcal{J}^T \mathcal{F}$ . Now, let a vector in the orbit frame obtained from a mathematical model be denoted by  $\mathbf{z}^o = [o_x \ o_y \ o_z]^T$  and a vector in the body frame obtained through measurement be denoted by  $\mathbf{z}^b = [b_x \ b_y \ b_z]^T$ . Then, the functions  $\mathbf{f}(\hat{\mathbf{q}}_{k-1}, \mathbf{z}^o, \mathbf{z}^b)$  and  $\mathbf{J}_q(\hat{\mathbf{q}}_{k-1}, \mathbf{z}^o)$  are given by

$$\mathbf{f}(\hat{\mathbf{q}}_{k-1}, \mathbf{z}^o, \mathbf{z}^b) = \begin{bmatrix} 2o_x(0.5 - q_3^2 - q_4^2) + 2o_y(q_1q_4 + q_2q_3) + 2o_z(q_2q_4 - q_1q_3) - b_x \\ 2o_x(q_2q_3 - q_1q_4) + 2o_y(0.5 - q_2^2 - q_4^2) + 2o_z(q_1q_2 + q_3q_4) - b_y \\ 2o_x(q_1q_3 + q_2q_4) + 2o_y(q_3q_4 - q_1q_2) + 2o_z(0.5 - q_2^2 - q_3^2) - b_z \end{bmatrix}, \quad (42)$$

$$\mathbf{J}_q(\hat{\mathbf{q}}_{k-1}, \mathbf{z}^o) = \begin{bmatrix} 2o_yq_4 - 2o_zq_3 & 2o_yq_3 + 2o_zq_4 & -4o_xq_3 + 2o_yq_2 - 2o_zq_1 & -4o_xq_4 + 2o_yq_1 + 2o_zq_2 \\ -2o_xq_4 + 2o_zq_2 & 2o_xq_3 - 4o_yq_2 + 2o_zq_1 & 2o_xq_2 + 2o_zq_4 & -2o_xq_1 - 4o_yq_4 + 2o_zq_3 \\ 2o_xq_3 - 2o_yq_2 & 2o_xq_4 - 2o_yq_1 - 4o_zq_2 & 2o_xq_1 + 2o_yq_4 - 4o_zq_3 & 2o_xq_2 + 2o_yq_3 \end{bmatrix}. \quad (43)$$

Given the gyro measurement  $\boldsymbol{\omega}_{gyro}^b$  (relative to inertial frame), the angular velocity relative to orbit frame can be found as

$$\boldsymbol{\omega}_{o,gyro}^b = \begin{bmatrix} 0 \\ \boldsymbol{\omega}_{gyro}^b - \mathbf{R}_o^b(\hat{\mathbf{q}}_{k-1})\mathbf{R}_i^o \boldsymbol{\omega}_{i,o}^i \end{bmatrix} \in \mathbb{R}^4, \quad (44)$$

where the rotation matrix from orbit to body frame is constructed using the estimated quaternion and denoted by  $\mathbf{R}_o^b(\hat{\mathbf{q}}_{k-1})$ . The Madgwick filter can now be presented as

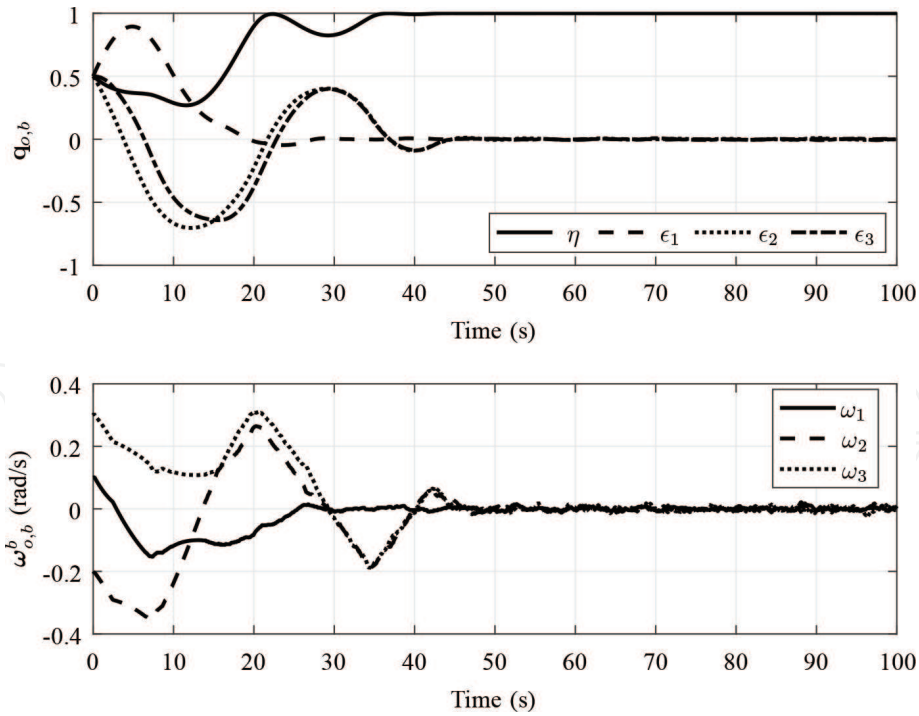
$$\hat{\boldsymbol{\omega}}_k^b = 2\mathbf{T}(\hat{\mathbf{q}}_{k-1}^*) \frac{\nabla \mathbf{f}}{\|\nabla \mathbf{f}\|}, \quad (45)$$

$$\boldsymbol{\omega}_{bias,k}^b = \boldsymbol{\omega}_{bias,k-1}^b + \zeta \hat{\boldsymbol{\omega}}_k^b \Delta T, \quad (46)$$

$$\boldsymbol{\omega}_{o,b}^b = \mathbf{H}(\boldsymbol{\omega}_{o,gyro}^b - \boldsymbol{\omega}_{bias,k}^b), \quad (47)$$

$$\dot{\hat{\mathbf{q}}}_k = \frac{1}{2}\mathbf{T}(\hat{\mathbf{q}}_{k-1}) \begin{bmatrix} 0 \\ \boldsymbol{\omega}_{o,b}^b \end{bmatrix} - \beta \frac{\nabla \mathbf{f}}{\|\nabla \mathbf{f}\|}, \quad (48)$$

$$\hat{\mathbf{q}}_k = \hat{\mathbf{q}}_{k-1} + \dot{\hat{\mathbf{q}}}_k \Delta T, \quad (49)$$



**Figure 19.** Attitude and angular velocity during the maneuver [9].

$$\hat{\mathbf{q}}_k = \frac{\hat{\mathbf{q}}_k}{\|\hat{\mathbf{q}}_k\|}, \quad (50)$$

where  $\beta$  and  $\zeta$  are gains, the time step is denoted by  $\Delta T$ , the estimated angular velocity based on vector measurements is denoted  $\hat{\omega}_k^b \in \mathbb{R}^4$ , and the estimated gyro bias is denoted by  $\omega_{bias,k}^b \in \mathbb{R}^4$ , while the angular velocity of the body frame relative to the orbit frame is denoted  $\omega_{o,b}^b \in \mathbb{R}^3$  (expected output) and the estimated quaternion is denoted  $\hat{\mathbf{q}}_k$  describing the body frame relative to the orbit frame. Note that the quaternion must be normalized to ensure unit length and that the first elements of  $\hat{\omega}_k^b$ ,  $\omega_{bias,k}^b \in \mathbb{R}^4$  are enforced to zero. To map the angular velocity from four to three elements, the projection matrix is defined as  $\mathbf{H} = [\mathbf{0} \quad \mathbf{I}] \in \mathbb{R}^{3 \times 4}$ , which has a column vector of zeros followed by the identity matrix such that  $\omega_{o,b}^b \in \mathbb{R}^3$ .

## 6.4 Simulation

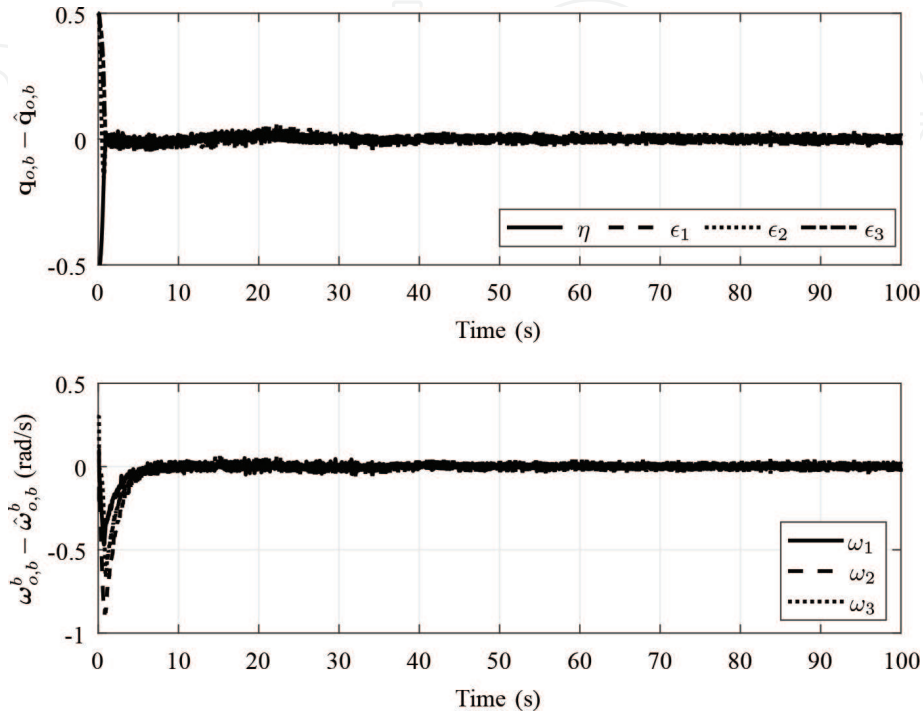
Let a satellite have the inertia matrix as given in Eq. (43), which contains non-diagonal terms which therefore will create perturbing moments due to the gravity. Furthermore, let the satellite have the following initial conditions:

$\mathbf{q}_{o,b}(0) = [0.5 \quad 0.5 \quad 0.5 \quad 0.5]^T$  and  $\omega_{o,b}^b = [0.1 \quad -0.2 \quad 0.3]^T$  with  $\hat{\mathbf{q}}_{o,b}(0) = [1 \quad 0 \quad 0 \quad 0]^T$  and  $\hat{\omega}_{o,b}^b = [0 \quad 0 \quad 0]^T$ . The desired quaternion can be chosen as  $\mathbf{q}_{o,d} = [1 \quad 0 \quad 0 \quad 0]^T$ , while  $\omega_{o,d}^d = \hat{\omega}_{o,d}^d = \mathbf{0}$ . To model noise in the sensor measurements, the quaternion is converted into Euler angles, where noise is added to the different sensors. Then, creating the rotation matrix from the noisy Euler angles allows the Sun vector, acceleration, and magnetometer models in the orbit frame to be rotated to the body frame, where the measurements now contain noise. The step size of the simulation is 0.01, while the sensors are sampled every 0.1 s.

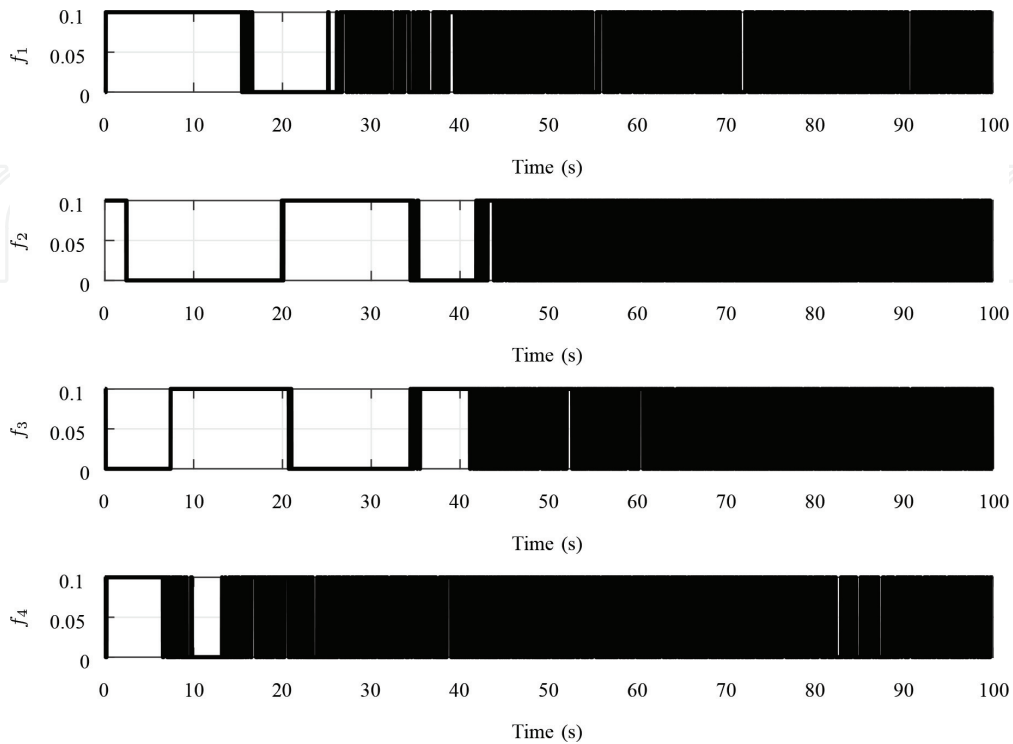
The quaternion and angular velocity error of the satellite are shown in **Figure 18**. After about 50 s, the objective of making the attitude error and angular velocity

error go to zero is achieved. Since the attitude is not measured directly, the Madgwick filter is used for attitude estimation as shown in **Figure 19**. Both the quaternion error (estimated truth) and angular velocity error converge close to zero.

From the PD+ controller, the desired torque is mapped to the desired thrust firings using bang-bang modulation (**Figure 20**). **Figure 21** shows the thruster firings required to maintain the attitude error close to zero.



**Figure 20.**  
Estimation error [9].



**Figure 21.**  
Thruster firings to control the attitude [9].

## 7. Conclusion

This chapter has presented all the components required to create an attitude determination and control system for satellites in elliptical orbits. With this as basis, it is the hope by the author that the work can help in developing new results within attitude determination and control systems, ranging from nonlinear controllers to new understanding in orbital mechanics, attitude determination, new sensors, and new actuation methods and strategies.

IntechOpen

IntechOpen

### Author details

Espen Oland  
Department of Electrical Engineering, UiT - The Arctic University of Norway,  
Narvik, Norway

\*Address all correspondence to: [espen.oland@uit.no](mailto:espen.oland@uit.no)

### IntechOpen

---

© 2018 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 



## References

- [1] Sidi MJ. *Spacecraft Dynamics & Control*. New York: Cambridge University Press; 1997
- [2] Wertz J, editor. *Spacecraft Attitude Determination and Control*. D. Dordrecht, Holland: Reidel Publishing Company; 1978
- [3] Vallado DA. *Fundamentals of Astrodynamics and Applications*. 3rd ed. El Segundo, California: Microcosm Press and Dordrecht, Boston, London: Kluwer Academic Publishers; 2007
- [4] Markley FL, Crassidis JL. *Fundamentals of Spacecraft Attitude Determination and Control*. El Segundo, California: Microcosm Press and New York, Heidelberg, Dordrecht London: Springer; 2014
- [5] Alarcon J, Örgen N, Kim S, Soon L, Cho M. Aoba VELOX-IV attitude and orbit control system design for a LEO mission applicable to a future lunar mission. In: *Proceedings of the 67th International Astronautical Congress*; Guadalajara City, Mexico; 2016
- [6] Dechao R, Tao S, Lu C, Xiaoqian C, Yong Z. Attitude control system design and on-orbit performance analysis of nano-satellite - tian Tuo 1. *Chinese Journal of Aeronautics*. 2014;27(3): 593-601
- [7] Nakajima Y, Murakami N, Ohtani T, Nakamura Y, Hirako K, Inoue K. SDS-4 attitude control system: In-flight results of three axis attitude control for small satellites. *IFAC Proceedings Volumes*. 2013;16(19):283-288
- [8] Fritz M, Shoer J, Singh L, Henderson T, McGee J, Rose R, Ruf C. Attitude determination and control system design for the CYGNSS microsatellite. In: *Proceedings of the IEEE Aerospace Conference*; Big Sky, Montana; 2015
- [9] Oland E. Attitude determination and control system for satellites in elliptical orbits—A complete solution. In: *Proceedings of the 8th International Conference on Recent Advances in Space Technology (RAST)*; Istanbul, Turkey; 2017
- [10] Oland E, Houge T, Vedal F. Norwegian student satellite program—HiNCube. In: *Proceedings of the 22nd Annual AIAA/USU Conference on Small Satellites*; Utah, USA; 2008
- [11] Oland E, Schlanbusch R. Reaction wheel design for cubesats. In: *Proceedings of the 4th International Conference on Recent Advances in Space Technologies (RAST)*; Istanbul, Turkey; 2009
- [12] Schlanbusch R, Oland E, Nicklasson PJ. Modeling and simulation of a cubesat using nonlinear control in an elliptic orbit. In: *Proceedings of the 4th International Conference on Recent Advances in Space Technologies (RAST)*; Istanbul, Turkey; 2009
- [13] Madgwick S, Harrison A, Vaidyanathan R. Estimation of IMU and MARG orientation using a gradient descent algorithm. In: *Proceedings of the 2011 IEEE International Conference on Rehabilitation Robotics*; Zurich, Switzerland; 2011
- [14] Oland E. Autonomous inspection of the International Space Station. In: *Proceedings of the 8th International Conference on Mechanical and Aerospace Engineering*; Prague, Czech; 2017
- [15] Egeland O, Gravdahl JT. *Modeling and Simulation for Automatic Control*. Trondheim, Norway: Marine Cybernetics; 2002. ISBN: 82-92356-01-0
- [16] Schaub H, Junkins JL. *Analytical Mechanics of Space Systems*. AIAA

American Institute of Aeronautics & Ast; 2003. pp. 1-19

[17] Oland E, Andersen TS, Kristiansen R. Subsumption architecture applied to flight control using composite rotations. *Automatica*. 2016;**69**:195-200

[18] Kristiansen R. Dynamic synchronization of spacecraft [PhD thesis]. Trondheim, Norway: Norwegian University of Science and Technology; 2008

[19] Takegaki M, Arimoto S. A new feedback method for dynamic control of manipulators. *ASME Journal of Dynamic Systems, Measurement, and Control*. 1981;**103**(2):119-125

[20] Paden B, Panja R. Globally asymptotically stable 'PD+' controller for robot manipulators. *International Journal of Control*. 1988;**47**(6):1697-1712

[21] Kristiansen R, Nicklasson PJ, Gravdahl JT. Spacecraft coordination control in 6DOF: Integrator backstepping vs. passivity-based control. *Automatica*. 2008;**44**(11):2896-2901

[22] Schlanbusch R, Loría A, Kristiansen R, Nicklasson PJ. PD+ based output feedback attitude control of rigid bodies. *IEEE Transactions on Automatic Control*. 2012;**57**(8):2146-2152

[23] Oland E. A command-filtered backstepping approach to autonomous inspections using a quadrotor. In: *Proceedings of the 24th Mediterranean Conference on Control and Automation*; Athens, Greece; 2016

[24] Hahn W. *Stability of Motion*. Berlin/Heidelberg/New York: Springer-Verlag; 1967. ISBN: 978-3-642-50087-9

[25] Wiśniewski R. Satellite attitude control using only electromagnetic

actuation [PhD thesis]. Aalborg University; 1996

[26] Hu Q, Ma G. Flexible spacecraft vibration suppression using PWWF modulated input component command and sliding mode control. *Asian Journal of Control*. 2007;**9**(1):20-29

[27] Psiaki M. Magnetic torquer attitude control via asymptotic periodic linear quadratic regulation. *Journal of Guidance, Control, and Dynamics*. 2001;**24**(2):386-394

[28] Thébault E et al. International geomagnetic reference field: The 12th generation. *Earth, Planets and Space*. 2015;**67**(79):1-19

[29] Shuster MD, Oh SD. Three-axis attitude determination from vector observations. *Journal of Guidance and Control*. 1981;**4**(1):70-77

[30] Farrell JL. Attitude determination by kalman filtering. *Automatica*. 1970;**6**(3):419-430

[31] Mahony R, Hamel T, Pflimlin J. Nonlinear complementary filters on the special orthogonal group. *IEEE Transactions on Automatic Control*. 2008;**54**(5):1203-1217