

POLITECNICO DI TORINO

Master's Degree in Mechatronic Engineering



**Politecnico
di Torino**

Master's Degree Thesis

**Design of Thruster Control Allocation for
a Cubesat**

Supervisors

Prof. Elisa CAPELLO

Prof. Hyeongjun PARK

Candidate

Corrado DE VITA

July 2023

*To RoboTo team, whose presence
has not only nurtured my personal
growth but also provided unwavering
companionship throughout this journey.*

Abstract

Recently, Cubesats have gained prominence in the aerospace industry due to their cost-effectiveness, short development cycle, and compact size (typically 10 cm x 10 cm x 10 cm). Additionally, advancements in technology have facilitated the integration of various functionalities required for a wide range of scientific, commercial, and technological missions. Despite these advantages, their limited fuel-carrying capacity should be considered a challenge for research. This limitation raises a critical issue regarding fuel consumption, which must be optimized to maximize the life of CubeSat satellites. The optimization of fuel consumption can be explored in the context of an overactuated system, where the number of actuators exceeds the degrees of freedom. In such a scenario, it becomes essential to determine the optimal allocation of the control input among multiple actuators, which is commonly referred to as control allocation. Moreover, this thesis also assumes that the thrusters of the Cubesat can be deflected relatively to the main body using gimbal mechanisms. Consequently, the resulting optimization problem aims to identify the optimal thruster configuration and allocation of the control input. To evaluate the proposed allocation strategy, the entire system is tested in the MATLAB/Simulink environment. A mission scenario is chosen by considering a couple of different maneuvers, different types of thruster configuration and different types of CubeSats. Finally, the effectiveness of each control allocation strategy is evaluated by analyzing the delta-V required in such a mission scenario.

Acknowledgements

First and foremost, I would like to express my heartfelt gratitude to my supervisors, Prof. Elisa Capello and Prof. Hyeongjun Park, for their unwavering support and guidance throughout this significant milestone of my academic journey. Their valuable advice and suggestions have greatly contributed to the improvement of my work, and I am truly grateful for their assistance.

I would also like to extend my appreciation to the members of team RoboTo. Not only have you been wonderful companions, but you have also helped me face every challenge with a positive attitude, reminding me of the joys and rewards of being an engineer. Thank you for broadening my mindset and providing me with a new perspective on life. I would like to express my gratitude to Vittorio, Luca, Francesco P., Marco, Francesco G., Arcangelo, and Jiang.

I am deeply grateful to all those who have believed in me from the very beginning. Your unwavering faith and confidence have given me the strength to tackle every obstacle that came my way.

Above all, I want to dedicate my deepest gratitude to my family. Without their belief in me, none of this would have been possible. They have been my pillars of strength, constantly providing the encouragement and motivation needed to overcome every hurdle along this arduous journey. Their unwavering support has shaped the person I am today, and I am forever grateful for their presence during the most challenging moments of my life. Thank you for instilling in me the values of perseverance and determination.

Table of Contents

List of Tables	VIII
List of Figures	IX
Acronyms	XII
1 Introduction	1
1.1 Cubesat	1
1.2 Control Allocation	2
1.3 Thesis Overview	4
2 System Modeling	6
2.1 Spacecraft	6
2.2 Thrusters	6
2.2.1 Thruster Static Configuration	8
2.2.2 Thruster Dynamic Configuration	8
2.3 Reference Frames	11
2.3.1 Local Vertical Local Horizontal Frame	11
2.3.2 Body Frame	12
2.3.3 Reference Frame Transformation	13
2.4 Hill Equations of Motion	14
2.5 Attitude Dynamics and Kinematics	15
3 Control System	17
3.1 Position Control	17
3.2 Attitude Control	19
4 Thruster Allocation	22
4.1 Thruster Allocation Techniques	22
4.1.1 Pseudo-inverse	23
4.1.2 Mixed Optimization Problem	24

4.2	Mixed Optimization Problem Formulation	25
4.2.1	Static thruster configuration	25
4.2.2	Dynamic thruster configuration	29
4.3	Pulse Width Pulse Frequency Modulator	31
5	Simulation and Results	34
5.1	Simulation Environment	35
5.2	Radial Boost Maneuver	36
5.3	Cubesat Attachment	46
5.4	Straight Line Forced Motion Maneuver	48
6	Conclusion	65
6.1	Final Considerations	66
6.2	Future Work	66
	Bibliography	67

List of Tables

4.1	PWPF modulator tuning	33
5.1	Minimum ΔV required	45
5.2	Radial boost maneuver results	45
5.3	Minimum ΔV required	63
5.4	Straight line V-bar approach maneuver	63

List of Figures

1.1	Network2	3
1.2	Network2	3
1.3	Feedback control system scheme	4
1.4	Control allocation within feedback control system	4
2.1	3U Cubesat Model	7
2.2	Network2	7
2.3	Static thruster configuration x-z plane	9
2.4	Static thruster configuration y-z plane	9
2.5	Dynamic thruster configuration x-z plane	10
2.6	Dynamic thruster configuration y-z plane	10
2.7	Gimbal mount [4]	11
2.8	Gimbal mount with thruster mass [4]	11
2.9	LVLH frame [5]	12
2.10	Body frame	13
3.1	Position control scheme	18
3.2	Attitude control scheme	20
3.3	Reaction wheel working principle [8]	20
3.4	Reaction wheel in a Cubesat	21
4.1	CA Scheme	22
4.2	CA Scheme with PWPF modulator	32
4.3	PWPF modulator scheme	32
5.1	Simulink Scheme	35
5.2	Transfer along V-bar by radial impulses [5]	36
5.3	Radial boost maneuver reference	37
5.4	Attitude regulation	38
5.5	Attitude regulation spikes	38
5.6	Radial boost with CA	39
5.7	Radial boost control input using pseudo-inverse CA	39

5.8	Radial boost control input using static linear programming CA . . .	40
5.9	Radial boost control input using dynamic linear programming CA .	40
5.10	Pseudo-inverse thrust force	41
5.11	Static linear programming thrust force	42
5.12	Dynamic linear programming thrust force	43
5.13	Thruster deflection angles	44
5.14	6U Cubesat with fixed thrusters	46
5.15	6U Cubesat with fixed thrusters	47
5.16	6U Cubesat with fixed thrusters	48
5.17	6U Cubesat with moving thrusters	48
5.18	6U Cubesat with moving thrusters	49
5.19	6U Cubesat with moving thrusters	50
5.20	Straight line V-bar approach [5]	50
5.21	Straight line V-bar approach reference	51
5.22	Attitude regulation	52
5.23	Straight line V-bar approach with pseudo-inverse CA	52
5.24	Straight line V-bar approach with linear programming CA	53
5.25	V-bar approach control input using pseudo-inverse CA	53
5.26	V-bar approach control input using static linear programming CA .	54
5.27	V-bar approach control input using dynamic linear programming CA	54
5.28	Pseudo-inverse thrust force	55
5.29	Pseudo-inverse thrust force	56
5.30	Static linear programming thrust force	57
5.31	Static linear programming thrust force	58
5.32	Dynamic linear programming thrust force	59
5.33	Dynamic linear programming thrust force	60
5.34	Thruster deflection angles	61
5.35	Thruster deflection angles	62

Acronyms

CA

control allocation

LVLH

local vertical local horizontal

LQR

linear quadratic regulator

PWPF

pulse width pulse frequency

Chapter 1

Introduction

In recent decades, the aerospace industry has witnessed the growing significance of miniaturized satellites, specifically Cubesats. These compact satellites have become increasingly prominent due to advancements in technology, enabling the execution of various tasks within their limited dimensions. While their cost-effective production is advantageous, their small size poses a challenge in terms of fuel carrying capacity, thus necessitating a focus on fuel consumption optimization. This master thesis aims to address this concern by designing control allocation techniques for such satellites. The research project was conducted in collaboration with Politecnico di Torino, which provided their infrastructure, materials, and invaluable support to achieve the objectives of this study.

1.1 Cubesat

The seeds of Cubesats were planted in the late 1990s by innovative minds at Stanford University and California Polytechnic State University. Driven by a vision to make space exploration more accessible and educational, these institutions embarked on a project to develop small, standardized satellites that could be used as practical learning platforms. The initial concept was inspired by the need for a cost-effective and modular satellite design that could enable students to gain hands-on experience in designing, building, and operating spacecraft. The first breakthrough in Cubesat development came with the establishment of a standardized form factor. Dr. Jordi Puig-Suari of California Polytechnic State University and Dr. Bob Twiggs of Stanford University pioneered the Cubesat standard, defining a 10 cm x 10 cm x 10 cm unit size with a maximum mass of 1.33 kilograms. This uniformity allowed for compatibility and interchangeability of components, simplifying the manufacturing process and lowering costs. Early Cubesat missions focused primarily on educational objectives, providing students

with an opportunity to design and launch their own satellites. In 2003, the first successful university-led Cubesat mission, "Cubesat XI-V," was launched by Japan. This milestone demonstrated the feasibility of the Cubesat concept and paved the way for further exploration and development. As the educational potential of Cubesats became evident, their popularity quickly spread beyond academia. The turn of the 21st century witnessed a paradigm shift, as Cubesats emerged as viable platforms for scientific research, commercial applications, and technology demonstrations. Their small size, standardized design, and cost-effectiveness made them attractive options for organizations with limited budgets seeking to explore space and gather valuable data. In recent years, Cubesats have experienced a surge in popularity driven by technological advancements, reduced launch costs, and increased access to space. Universities, research institutions, startups, and even established space agencies have embraced Cubesats as versatile tools for a diverse range of missions. Cubesats have been employed for Earth observation, climate monitoring, communications, astronomy, and technology validation, to name just a few applications. Their success has inspired countless innovations, pushing the boundaries of what can be achieved within the confines of a Cubesat platform [1]. Looking towards the future, the trajectory of Cubesats appears promising. Continued advancements in miniaturization technology, electronics, propulsion systems, and communication capabilities will further enhance their capabilities. As previously mentioned, Cubesats offer numerous advantages. Since these satellites are made of off-shelf components, they possess some benefits including low production cost, short development time, and easy repairs. Additionally, their small size allows for a minimal and simple design. However, these benefits come with certain drawbacks. The primary disadvantage is their limited payload capacity, which restricts the inclusion of various instruments and sensors, potentially limiting the satellite's scientific and operational capabilities. This limitation also affects power consumption. Optimizing power consumption becomes crucial, particularly for longer-term missions. Consequently, Cubesats have a shorter lifespan due to limited fuel carrying capacity. Figure 1.2 and ?? depicts respectively a Norwegian Cubesat and a Cubesat and Rubik's cube comparison.

1.2 Control Allocation

Before delving into the details of control allocation, it is important to establish the working context. The Cubesat under study is designed to undertake various missions throughout its space journey, which may involve executing multiple maneuvers. These maneuvers are performed using a feedback control system. Figure 1.3 presents a simplified diagram of a typical feedback control system. As illustrated, the controller receives the control error as input, which is the difference between

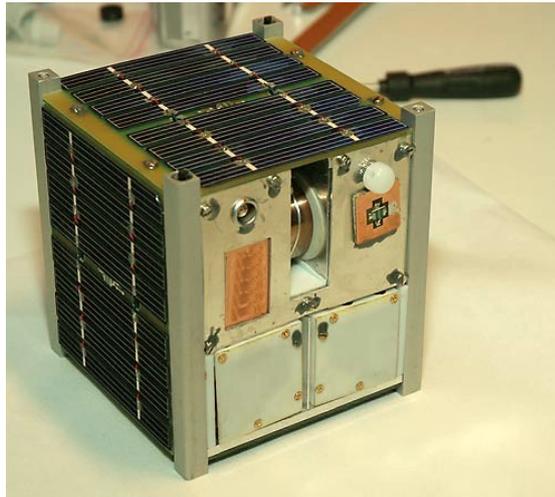


Figure 1.1: Ncube-2 Cubesat
<https://en.wikipedia.org/wiki/CubeSat>

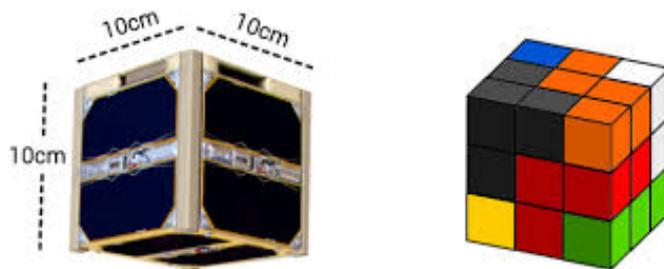


Figure 1.2: Rubik's cube vs Cubesat
<https://en.wikipedia.org/wiki/CubeSat>

the reference and the output of the plant. The controller processes the control error and generates the control input, which serves as the driving signal for the plant.

In this scenario, the reference typically represents the desired trajectory for a specific maneuver, while the plant generally represents the orbital dynamics.

With the working context established, the concept of control allocation can be introduced. In many cases, the specifics of how the plant is actuated are not readily apparent. Although it may appear from Figure 1.3 that the plant is directly actuated by the control input signal u , that is not entirely accurate. In reality, the control input signal is allocated into several actuators. The complete scheme is illustrated in Figure 1.4.

Within the context of control allocation, the controller's output u_d is defined as

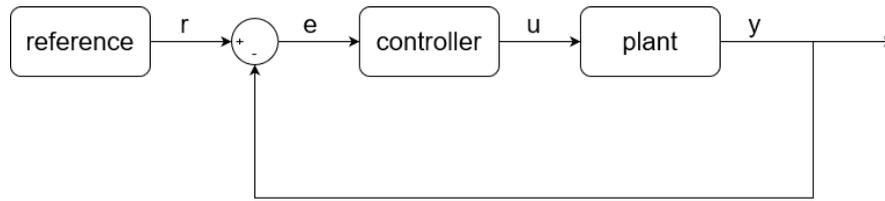


Figure 1.3: Feedback control system scheme

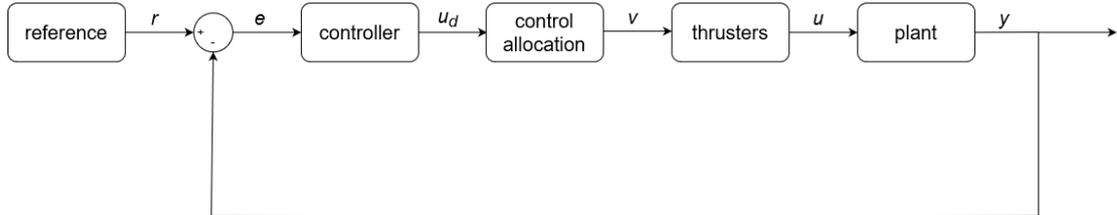


Figure 1.4: Control allocation within feedback control system

virtual control input. Such quantity represents the expected control input. The control allocation block's output v is defined as the actuation effort (or thrust force in the case of thruster actuators). The quantity v indicates the portion of the virtual control input assigned to each actuator. Finally, as the overall system is actuated, the sum of the several actuator efforts produces the control input u . The difference between the control input u and the virtual control input u_d should be as close as possible to 0. In other words, the actual control input u should be as similar as possible to the expected control input u_v .

The key assumption is that the satellite is an over-actuated system. An over-actuated system has more actuators than degrees of freedom. Consequently, the vector size of the virtual control input u_d is smaller than the vector size of the actuation effort v , indicating that there is not a unique mapping from u_d to v , therefore there are infinite ways to allocate a control input signal to multiple actuators. Given this fact, it is reasonable to consider which allocation method is optimal, and that depends on the specific goal. One common approach is to minimize the actuation effort v , thereby reducing fuel consumption. Another possible approach is to select the allocation that uses the fewest thrusters, reducing the risk of having faulty ones .

1.3 Thesis Overview

The main objective of this thesis is to investigate and evaluate various control allocation techniques for optimizing fuel consumption in miniaturized satellites,

specifically Cubesats. The thesis is divided into six chapters, each addressing different aspects of the research. Chapter 1 serves as an introduction, providing historical background on Cubesats, highlighting their advantages and disadvantages, and emphasizing the importance of control allocation techniques for these spacecraft. Additionally, a technical explanation of control allocation concepts is provided. Chapter 2 focuses on the system modeling of the Cubesat, providing a detailed description of the spacecraft and analyzing its actuation. The chapter also presents the static and dynamic thruster configurations of the Cubesat and discusses mathematical models for translational and rotational motion. Chapter 3 delves into the analysis of control systems used in Cubesats. The position controller and attitude controller, based on the Cubesat models developed in the previous chapter, are thoroughly discussed. Chapter 4 is dedicated to the development of various thruster control allocation techniques. The examined techniques include the pseudo-inverse and the mixed optimization problem. Furthermore, the advantages and disadvantages of these control allocation techniques are extensively explored. Chapter 5 introduces a simulation environment where the Cubesat is simulated in a mission scenario involving two maneuvers: the radial boost maneuver and the straight-line V-bar approach maneuver. Multiple simulations are conducted using the control allocation techniques developed in previous chapters. The results of these simulations are presented in graphs and tables, and a comprehensive comparison of the control allocation techniques is made, considering criteria such as feasibility, performance, and complexity. Finally, in Chapter 6, a summary of the work done is provided, and conclusions regarding the control allocation strategies are drawn.

Chapter 2

System Modeling

This chapter begins with an introduction to the Cubesat, providing a comprehensive overview of its essential features and characteristics. It subsequently explores the thrusters employed by the spacecraft in detail. Additionally, recognizing the necessity of effectively controlling the position and attitude of the Cubesat, the chapter delves into the mathematical aspects involved in modeling the translational and rotational motion of the Cubesat.

2.1 Spacecraft

The focus of this thesis revolves around a particular type of satellite known as a Cubesat. A Cubesat is characterized by its compact size, resembling a cube with dimensions of approximately 10 cm in length, width, and depth. This particular configuration is often referred to as a one-unit Cubesat or 1U Cubesat. In the context of this thesis, the Cubesat under study is a 3U Cubesat, consisting of three individual 1U Cubesats attached together along a common axis. Figure 2.1 provides a visual representation of a 3U Cubesat model.

A unit Cubesat is conventionally standardized to have a length (l) of 10 cm. Consequently, a 3U Cubesat would have the same length and height, measuring 10 cm each, while its total depth would be 30 cm due to the attachment of three unit Cubesats. The estimated weight of this 3U Cubesat is approximately 4 kg [2].

2.2 Thrusters

The propulsion systems for spacecraft offer a range of thruster options. Chemical thrusters rely on chemical reactions for generating thrust and are frequently used for launch vehicles and missions requiring high thrust. Electric thrusters, on the other hand, employ electric power to accelerate and expel propellant, offering enhanced

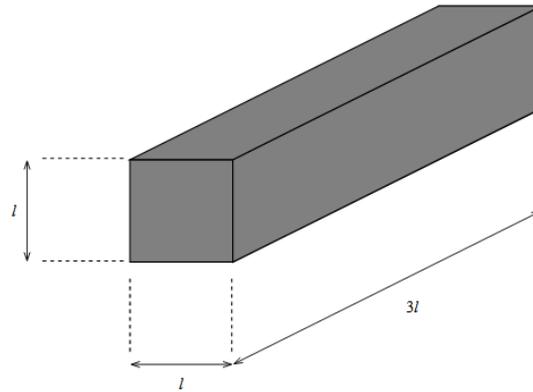


Figure 2.1: 3U Cubesat Model

efficiency and longer operational duration. Cold gas thrusters operate by utilizing compressed gas, such as nitrogen or helium, to generate thrust. While simple and dependable, they possess lower specific impulse compared to other thruster types. Ion thrusters, powered by electricity, ionize and accelerate propellant (e.g., xenon) to generate thrust, offering significantly high specific impulse and often employed in long-duration missions. Figure 2.2 shows an example of a Cubesat actuated by thrusters [3].



Figure 2.2: Propelled Cubesat
<https://spacenews.com/uwe-4-lowers-altitude>

Within the scope of this master's thesis, the specific categorization of actuators is of secondary importance. However, certain assumptions need to be established regarding the thruster's maximum force output and its characteristics, as these

factors hold significant relevance for the subsequent development of the project. Consequently, the following assumptions are made for the purpose of this study:

- Discontinuous thrusters (On-Off thrusters)
- Maximum Thruster Force: 0.05 [N]

This indicates that when the thruster is deactivated, it will not generate any propulsive force. Conversely, when the thruster is activated, it will generate a thrust force of 0.05 N. Moving on to the thruster configuration of the Cubesat, as mentioned earlier, the system being studied is classified as an over-actuated system. This means that the number of actuators (thrusters) exceeds the number of degrees of freedom of the Cubesat. Specifically, the Cubesat in question is equipped with a total of 7 thrusters. It is important to note the arrangement of these thrusters on the satellite. In this thesis, both a static configuration and a dynamic configuration of the thrusters will be considered. In the static configuration, it is assumed that the thrusters are fixed and cannot move relative to the main body of the satellite. On the other hand, in the dynamic configuration, the thrusters have the capability to be deflected or repositioned in relation to the satellite's main body. The chosen configuration of the thrusters will have a significant impact on the CA process, as it influences how the available thrust can be allocated and distributed.

2.2.1 Thruster Static Configuration

Figure 2.3 and 2.4 shows the thruster configuration in case of fixed thrusters. As observed, some thrusters on the satellite are not aligned perpendicular to the main body. In particular, thruster 1, 2, 3, 4 are inclined at a 75° angle in relation to the satellite's main body. This arrangement is necessary to generate thrust in the positive x-direction and in the positive and negative z-direction. The specific angle of 75 degrees does not have a particular significance; it could have been slightly lower or higher. The selection of such an angle depends on the requirements of the mission scenario. If the mission scenario necessitates a thrust force primarily oriented in the positive x-direction, the thrusters will be oriented accordingly. Conversely, if the mission scenario requires a thrust force primarily focused on the vertical direction, the thrusters will be adjusted accordingly.

2.2.2 Thruster Dynamic Configuration

Figure 2.5 and 2.6 shows the thruster configuration in case of moving thrusters. The improvement of this thruster configuration is that thrusters 1, 2, 3, and 4 are capable of deflection. These particular thrusters have the ability to adjust their angles within a range of $\pm 5^\circ$ around a nominal value of 75° by deflecting in

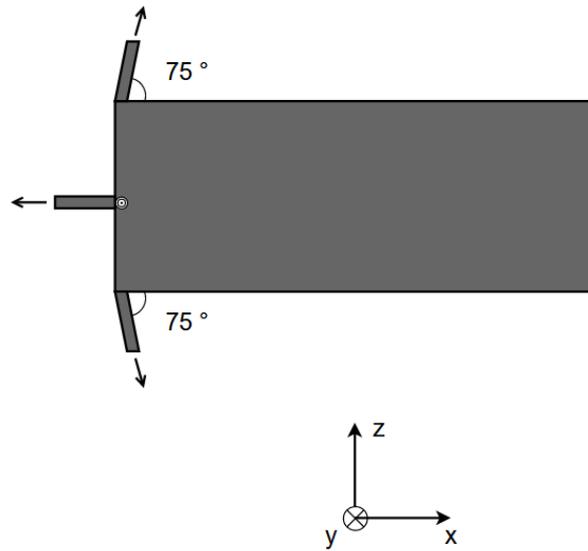


Figure 2.3: Static thruster configuration x-z plane

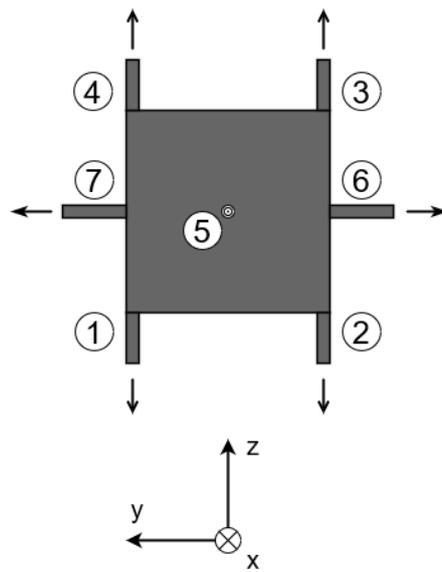


Figure 2.4: Static thruster configuration y-z plane

the x-z plane. It is worth mentioning the practical implementation of the thruster deflection mechanism. NASA has developed a solution involving a gimbal assembly for steering propelled Cubesats. This assembly incorporates a gimbal that serves as the seat for the thrusters. The gimbal is actuated by two piezoelectric motors, enabling control over the rotary motion (360°) and tilt angle ($\pm 12^\circ$) with an

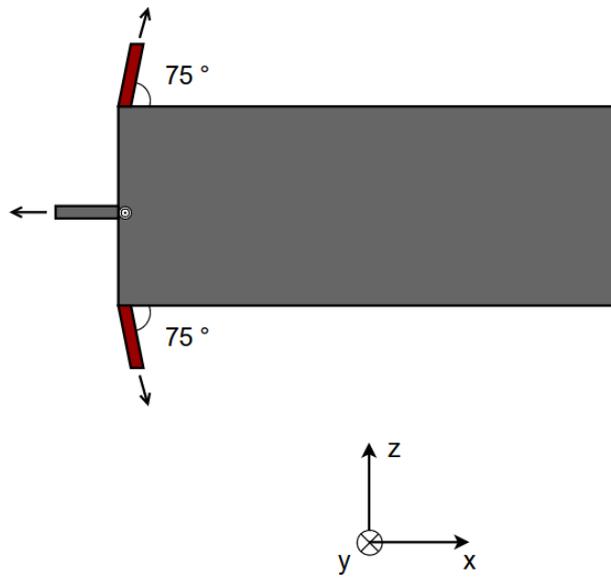


Figure 2.5: Dynamic thruster configuration x-z plane

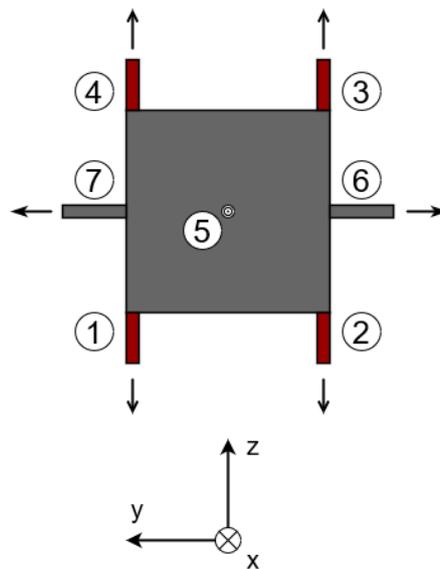


Figure 2.6: Dynamic thruster configuration y-z plane

impressive accuracy of 0.02° . 2.7 and 2.8 shows respectively the gimbal mount without and with the thruster mass [4].

It is essential to emphasize that the movement capability of the thrusters in this thesis project differs from the aforementioned NASA implementation. The thesis

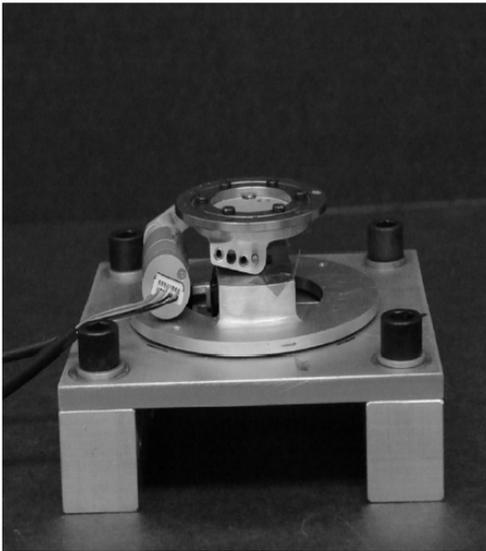


Figure 2.7: Gimbal mount [4]

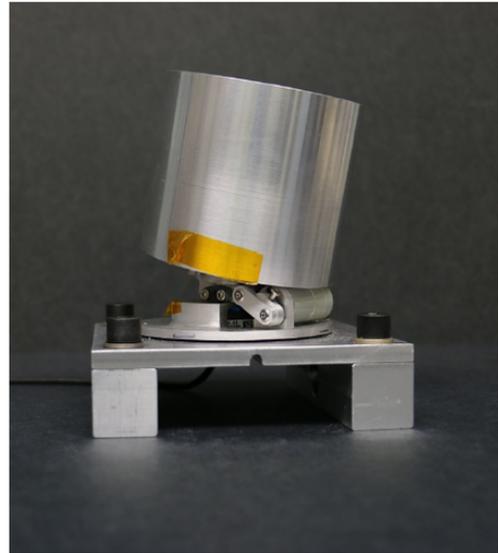


Figure 2.8: Gimbal mount with thruster mass [4]

assumes simpler thrusters capable of deflecting within a plane by $\pm 5^\circ$. This explanation serves to highlight the feasibility of incorporating dynamic thrusters in a Cubesat satellite.

2.3 Reference Frames

2.3.1 Local Vertical Local Horizontal Frame

The LVLH Frame serves as a local reference system that simplifies the analysis and control of satellite motion relative to its own orientation and position. This frame provides a convenient means of describing and understanding the behavior of a satellite in its immediate vicinity, particularly when analyzing the trajectory of a chaser satellite with respect to a target satellite. The LVLH Frame is centered at the target satellite's center of mass and is oriented relative to the satellite's local vertical and local horizontal directions. The local vertical direction points towards the Earth's center of mass, and the local horizontal direction lies in the plane perpendicular to the local vertical, pointing in the direction of the satellite's velocity vector [5]. Figure 2.9 illustrates an example of LVLH frame.

By analyzing the chaser's trajectory with respect to the target within the LVLH Frame, valuable insights about the relative positioning and motion of the two satellites can be obtained. This understanding is particularly advantageous for tasks such as rendezvous and docking maneuvers, where precise control of the

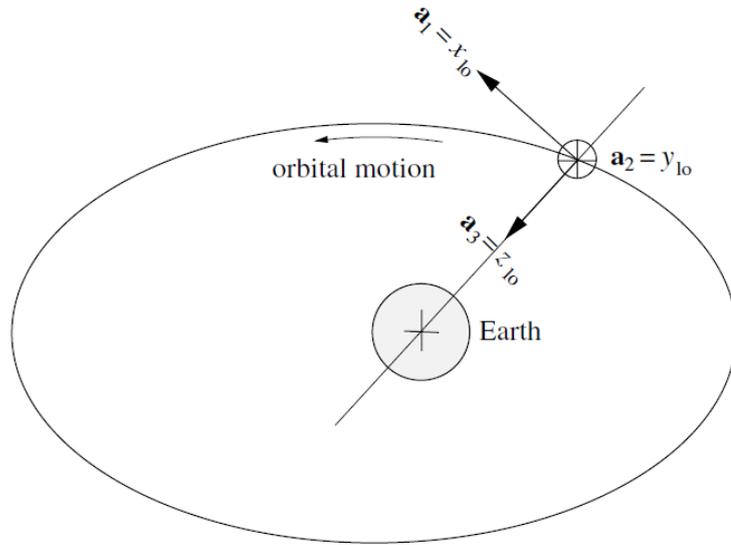


Figure 2.9: LVLH frame [5]

chaser's trajectory relative to the target is crucial for a successful approach.

2.3.2 Body Frame

The body frame is a reference frame that is fixed to the spacecraft or satellite itself. The body frame is typically aligned with the spacecraft's principal axes of inertia, which are determined by its shape and mass distribution. In this frame, the x-axis is aligned with the spacecraft's longitudinal axis, the y-axis corresponds to the lateral axis, and the z-axis represents the vertical axis. The body frame provides a convenient framework for analyzing the satellite properties such as inertial properties as well as the thruster configuration or any other satellite's features related to its geometry [5]. Figure 2.10 illustrates the body frame of the Cubesat under analysis.

The main distinction between the LVLH frame and the body frame lies in their respective attitudes. Although both frames are body-centered, the LVLH frame's attitude is determined relative to the celestial body being orbited, while the body frame's attitude is fixed to the spacecraft itself. In this thesis project, the LVLH frame will be centered in the target satellite while the body frame will be centered in the chaser satellite.

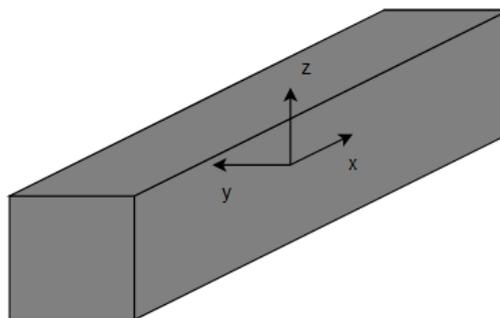


Figure 2.10: Body frame

2.3.3 Reference Frame Transformation

The understanding of the relationship between the Local Vertical Local Horizontal (LVLH) frame and the Body frame is crucial in this study. The most straightforward mathematical approach to determine the correlation between two reference frames is through the use of rotation matrices. A rotation matrix from frame A to frame B consists of columns representing the unit base vectors of frame B expressed in frame A . Each rotation matrix can be broken down into three elementary rotations. An elementary rotation refers to the rotation of one frame relative to another around a base unit vector. Thus, to fully characterize a transformation between two reference frames, it is sufficient to determine the values the three angles representing these elementary rotations. However, the use of rotation matrices as a mathematical tool to describe frame transformations presents the singularity problem. This problem arises when a complex frame transformation cannot be uniquely decoupled into three elementary rotations. Consequently, in the field of aerospace engineering, it is common practice to employ quaternions. Quaternions, similar to rotation matrices, are mathematical tools used to express frame transformations. However, quaternions depend not on three but on four variables. This additional information overcomes the singularity problem and enables the mapping of each frame transformation to its corresponding quaternion. Equation 2.1 represents the quaternion equation considering a rotation about an axis u of an angle ω .

$$q = \left(\cos \left(\frac{\beta}{2} \right), u_1 \sin \left(\frac{\beta}{2} \right), u_2 \sin \left(\frac{\beta}{2} \right), u_3 \sin \left(\frac{\beta}{2} \right) \right) \quad (2.1)$$

It is also possible to switch from the quaternion representation to the rotation matrix representation. Equation 2.2 represents the change of representation from quaternion to rotation matrix.

$$T = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (2.2)$$

2.4 Hill Equations of Motion

The Hill equations, or Clohessy-Wiltshire equations, are linearized differential equations that describe the relative motion between a chaser and target satellite in a circular orbit expressed in the LVLH frame. They are important in chaser and target scenarios, such as rendezvous and docking missions, as they provide a simplified mathematical model for analyzing and controlling the relative dynamics [6]. Hill equations are reported in Equation 2.3.

$$\begin{cases} \ddot{x} = 2\omega\dot{z} + \frac{1}{m_c}F_x \\ \ddot{y} = -\omega^2y + \frac{1}{m_c}F_y \\ \ddot{z} = -2\omega\dot{x} + 3\omega^2z + \frac{1}{m_c}F_z \end{cases} \quad (2.3)$$

The coordinates x, y and z indicate the chaser position with respect to the target in the LVLH frame. The variable x is the coordinate along the 'V-bar', the variable y is the coordinate along the 'H-bar' and finally the variable z is the coordinate along the 'R-bar'. The parameter ω expresses the orbital angular velocity of the target. Lastly, F_x , F_y , and F_z represent the forces exerted on the chaser. These equations can be easily converted into state space equation given their linearity. The state space form is reported in Equation 2.4.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -2w \\ 0 & -w^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3w^2 & -2w & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} + \begin{bmatrix} \frac{1}{m_c} & 0 & 0 \\ 0 & \frac{1}{m_c} & 0 \\ 0 & 0 & \frac{1}{m_c} \end{bmatrix} \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} \quad (2.4)$$

The state space form of the Hill equations will be utilized as a linear model in the feedback control system for tracking the satellite's position.

2.5 Attitude Dynamics and Kinematics

The attitude dynamics and kinematics modelling is an essential part to build a complete attitude regulation system. The attitude dynamics can be described by means of the so called Euler moment equation reported in Equation 2.5 [6].

$$J\dot{\omega} = M - \omega \times (J\omega) \quad (2.5)$$

The state variable ω represent the satellite's angular velocity, the vector M is defined as the moment acting on the satellite, and finally the matrix J represent the satellite's inertial properties. Equation 2.6 reports the Euler moment equation written in matrix form.

$$\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \end{bmatrix} = \begin{bmatrix} 0 & \sigma_1\omega_3 & 0 \\ \sigma_2\omega_3 & 0 & 0 \\ \sigma_3\omega_2 & 0 & 0 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} + \begin{bmatrix} \frac{M_1}{J_1} \\ \frac{M_2}{J_2} \\ \frac{M_3}{J_3} \end{bmatrix} \quad (2.6)$$

The parameters σ_1 , σ_2 , and σ_3 are defined as:

$$\begin{aligned} \sigma_1 &= \frac{J_2 - J_3}{J_1}, \\ \sigma_2 &= \frac{J_3 - J_1}{J_2}, \\ \sigma_3 &= \frac{J_1 - J_2}{J_3}. \end{aligned}$$

The attitude dynamic equations alone do not provide sufficient information for achieving attitude regulation. To address this, a further step is necessary, which involves establishing a relationship between the angular velocity ω and the quaternion q . These equations, known as the attitude kinematic equations, are presented in Equation 2.7.

$$\dot{q} = \frac{1}{2} \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \quad (2.7)$$

By integrating Equation 2.7, it is possible to compute the quaternion q . Both the satellite angular velocity and its corresponding quaternion will be used to regulate the satellite's attitude.

Chapter 3

Control System

Control techniques play a pivotal role in the field of aerospace, serving as a crucial element in ensuring safe and efficient operations of aircraft and spacecraft. These techniques encompass a wide range of methods and technologies designed to manipulate the motion, stability, and attitude of aerospace vehicles. This section focuses on the utilization of control techniques for managing position and attitude in aerospace applications. The control of position is paramount when satellites engage in common maneuvers like rendezvous or orbit adjustments. Likewise, attitude control holds equal significance to position control, as satellites may need to align with other satellites or maintain a fixed orientation to collect data from Earth.

3.1 Position Control

Before going into the detail about the control technique employed for the position control, it is necessary to present the general setup. The overall feedback control system is shown in Figure 3.1.

As depicted in Figure 3.1, the feedback control system includes the controller, plant, and reference components. The plant is modeled by the Hill equations discussed in Chapter 2. The outputs of the plant, which coincides with the state variables, namely the satellite's position x, y, z and satellite's linear velocity $\dot{x}, \dot{y}, \dot{z}$ are fed back into the controller. It is assumed that the Cubesat is equipped with appropriate sensors capable of accurately measuring the satellite's position and linear velocity. Additionally, it is assumed that these sensors are free from any noise or corruption. The reference is the trajectory to follow and it includes the position reference, namely x_r, y_r, z_r and the velocity reference $\dot{x}_r, \dot{y}_r, \dot{z}_r$. It is important to notice that both the state variables and the reference are expressed in the LVLH frame discussed in Chapter 2. Therefore the position and velocity variable

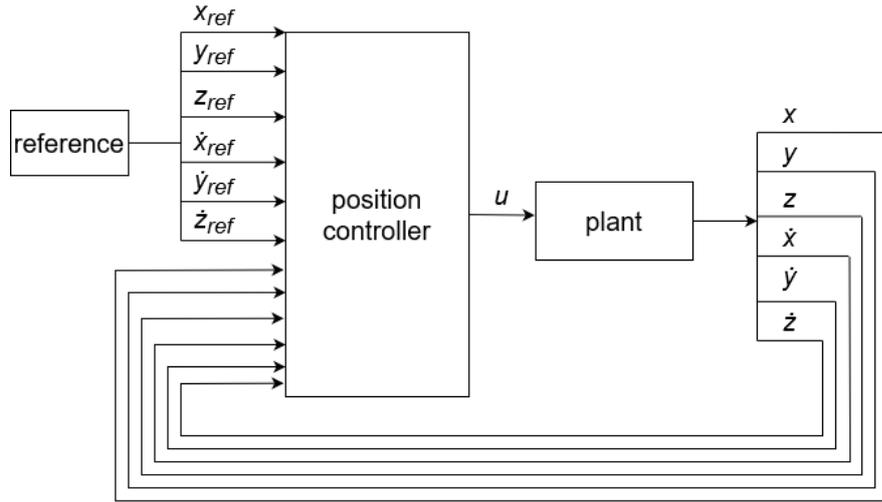


Figure 3.1: Position control scheme

expresses the relative position and velocity of the chaser satellite with respect to a target satellite. The controller takes the state variables and the reference as inputs and generates the control input signal, denoted as u , which drives the satellite's behavior. Moving to the heart of the discussion, it is now time to delve into the control technique employed in this particular scenario. When it comes to aerospace applications, a common approach is to utilize a model predictive control (MPC) due to its various advantages. One notable advantage is its ability to handle nonlinear systems and consider constraints. However, this method requires substantial computational resources. In this thesis, a different controller called the linear quadratic regulator (LQR) is chosen. Unlike the MPC controller, the LQR is primarily designed for linear time invariant system and does not account for constraints, however it is computationally less demanding. The absence of constraint consideration is not a major issue in this case. Typically, constraints on the actuators, such as the maximum thruster force, can be accounted for at a later stage, specifically in the CA development section. The implementation of an LQR controller assumes that the system dynamics can be accurately modeled by a set of linear differential equations. The primary objective of the LQR controller is to minimize a quadratic cost function that captures both the system's state and control input. The cost function is reported in Equation 3.1.

$$J = \int [x(t)^\top Q x(t) + u(t)^\top R u(t)] dt \quad (3.1)$$

The variables x and u represents respectively the state of the system and the control

input. As previously mentioned, the state of the system is the vector including position and linear velocity of the chaser while the control input is the thrust force vector applied on the chaser. By tuning the diagonal entries of the matrices Q and R in the equation, the primary objective of the minimization problem can be modified. If the diagonal entries of matrix Q are greater than those of matrix R , the state tracking performance is prioritised, emphasizing accurate tracking of the desired trajectory. Conversely, if the diagonal entries of matrix R are greater than those of matrix Q , the minimization of the control input required to achieve the desired tracking performance is prioritised. The control input is determined by calculating the optimal state feedback gain matrix. This matrix maps the system's current state to the control input. The control law is reported in Equation 3.2.

$$u(t) = -Kx(t) \tag{3.2}$$

The calculation of the optimal state-feedback gain matrix involves solving the Riccati equation. By solving the Riccati equation, the controller determines the optimal gain matrix that minimizes the quadratic cost function. The Riccati equation and the equation to calculate the state-feedback gain matrix are reported respectively in Equation 3.3 and Equation 3.4.

$$P = A^T P A - A^T P B (R + B^T P B)^{-1} B^T P A + Q \tag{3.3}$$

$$K = (R + B^T P B)^{-1} B^T P A \tag{3.4}$$

It is important to notice the computational efficiency and ease of implementation of the LQR controller. The calculations involved in solving the Riccati equation and computing the control input are relatively straightforward, making the LQR controller suitable for real-time control applications. However, it's important to note that the LQR controller is specifically designed for linear systems without constraints. It may not perform optimally in the presence of non-linearities, time-varying dynamics, or when there are constraints on the system's inputs, states, or outputs [7].

3.2 Attitude Control

Similarly to the previous section, it is appropriate to present the general setup of the attitude control. The attitude control scheme is reported in Figure 3.2.

Figure 3.2 illustrates the components involved in the system: the rotational dynamics, the rotational kinematics and the attitude controller. The rotational

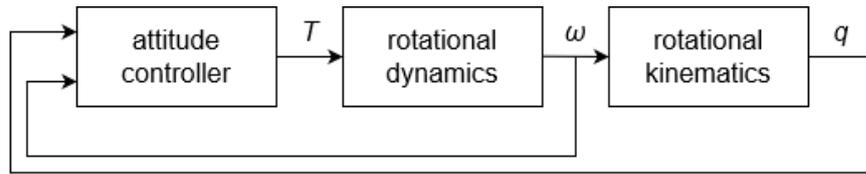


Figure 3.2: Attitude control scheme

dynamics and rotational kinematics can be seen as the plant driven by the attitude controller. The rotational dynamics is described by the Euler equations presented in Chapter 2 while the rotational kinematics is described by the quaternion kinematics also discussed in Chapter 2. The attitude controller operates based on feedback signals as it receives in feedback the satellite's angular velocity ω and the satellite's attitude q . It is assumed that the spacecraft is equipped with sensors able to measure these physical quantities. Furthermore, for the purpose of this analysis, it is assumed that these measurements are free from any noise or disturbances. Additionally, the output of the controller, denoted as T , corresponds to a torque signal that directly actuates the spacecraft. In the context of this thesis project, it is assumed that this torque signal is provided by reaction wheels. Reaction wheels are electric actuators utilized for controlling the attitude of a satellite. They generate angular momentum through the application of reaction torque to the flywheel [8]. Figure 3.3 and 3.4 illustrate the working principle of reaction wheels and their placement within a Cubesat, respectively.

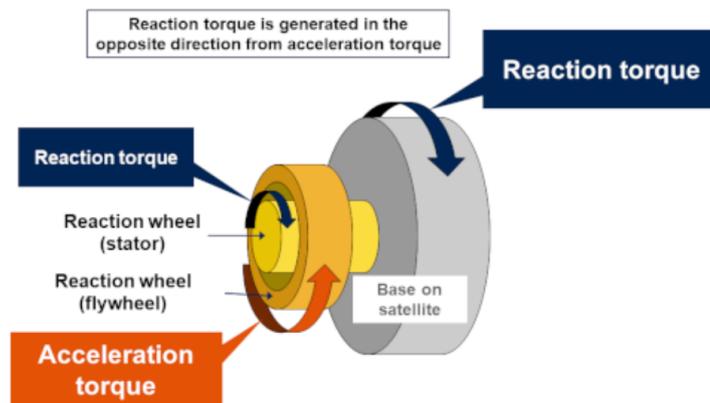


Figure 3.3: Reaction wheel working principle [8]

Unlike a position control system, the attitude control system's objective is to regulate the spacecraft's attitude rather than track a specific attitude over time. The distinction between attitude regulation and attitude tracking lies in the fact that, in the former, the satellite aims to achieve a fixed attitude, while in the latter,

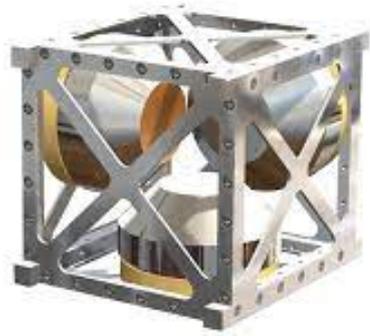


Figure 3.4: Reaction wheel in a Cubesat

it must follow a time-varying attitude reference signal. Next, the control law for the attitude control system will be presented, followed by several considerations. The control law is represented by Equation 3.5.

$$u = k_p \cdot \tilde{q} - k_d \cdot \omega \quad (3.5)$$

The variables ω and \tilde{q} respectively represent the satellite's angular velocity and the quaternion tracking error. The control law behaves as a Proportional-Derivative (PD) control law, which consists of a proportional part with gain K_p and a derivative part with gain K_d . The proportional part is directly proportional to the error between the desired quaternion set-point and the measured quaternion. It generates a control action that is proportional to the magnitude of the error. Larger errors result in larger control actions. The proportional part contributes to system stability and responsiveness. It helps steer the system towards the desired attitude by providing a corrective action that is proportional to the deviation from the set-point. The derivative part, on the other hand, predicts the future trend of the error by calculating the rate of change of the error signal. It is proportional to the rate of change of the error. The derivative part anticipates future changes in the error and provides additional control action to counteract them. This helps dampen the response of the system, reducing overshoot and oscillations. The derivative part enhances system stability and improves overall performance. By appropriately tuning the constants K_p and K_d , a balance between stability and responsiveness can be achieved [6].

Chapter 4

Thruster Allocation

CA plays a critical role in the overall satellite system. The CA strategy involves determining how the control input should be distributed among the various actuators, which in this case are the thrusters. This strategy is particularly valuable in the context of over-actuated systems, as there are multiple possible ways for distributing the control input, and a well-designed CA technique can identify the optimal distribution. As explained in Chapter 2, the Cubesat is an over-actuated system due to its 7 unidirectional on-off thrusters. Each thruster is capable of producing a maximum thrust force of 0.05 N. These thrusters can be arranged in both a static configuration and a dynamic configuration, offering additional flexibility in the CA process.

4.1 Thruster Allocation Techniques

In this section, a general overview of CA techniques is provided. Few specific techniques are presented as well as their advantages and disadvantages. It's also convenient to present the general setup of CA techniques. In figure 4.1 it is shown the location of the CA within the feedback control system.

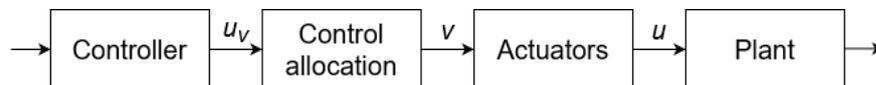


Figure 4.1: CA Scheme

It is helpful to establish the nomenclature that will be used throughout the remainder of the thesis project. The signal u_v represents the virtual control input, which is the signal generated by the control law. The term "virtual" indicates that this signal does not directly drive the behavior of the plant. The signal v represents

the thrust force, which specifies the desired magnitude of thrust that each thruster should produce. It contains the individual thrust force values for each thruster, determined by the CA process. Finally, the signal u denotes the control input, which is the signal generated by the actuators and directly influences the behavior of the plant. The actuator block is typically represented by a linear model given by

$$u = Bv \tag{4.1}$$

where B is the actuator model matrix. The matrix B represents the mapping between the thrust force v and the control input u . This matrix clearly depends on the thruster configuration of the satellite. It is not a square matrix due to the mismatch between the dimension of the vector v and u . As already mentioned in chapter 1, CA aims to identify an appropriate vector v that reduces the discrepancy between the desired control input u_v and the actual control input u . Additionally, it seeks to maintain the thrust force v at a minimum level in order to minimize fuel consumption.

4.1.1 Pseudo-inverse

The most straightforward idea in order to solve the CA problem is to invert the actuator model matrix B , therefore the thrust force v is computed according to equation 4.2.

$$v = B^{-1}u_v \tag{4.2}$$

Unfortunately, the actuator model matrix B is not a square matrix. In the case of an over-actuated system, it is typical for B to have full row rank and a non-trivial null space. Consequently, there exists an infinite number of vectors v that satisfy Equation 4.1 for any given u . To address this extra freedom, it is common to employ generalized inverses (or pseudo-inverses). In the following section, we outline this approach within the framework of minimizing a least-squares cost function [9]. The expression for the cost function to be minimized is presented as Equation 4.2.

$$\min_{v \in \mathbb{R}^p} \frac{1}{2} (v - v_p)^T W (v - v_p) \quad \text{subject to} \quad u_v = Bv \tag{4.3}$$

Through the minimization of this cost function, the obtained thrust force vector v generates a control input u identical to the virtual control input u_v . Simultaneously, the thrust force vector v is optimized to closely approximate the desired thrust force vector v_p . For the special case where $W = I$ and $v_p = 0$, the solution is defined by the Moore-Penrose pseudo-inverse given by

$$v = B^+ u_v \tag{4.4}$$

where

$$B^+ = (B^T B)^{-1} B^T \tag{4.5}$$

The primary benefit of this approach is its minimal computational cost due to its concise closed-form solution. However, its main limitation is the inability to incorporate thruster constraints, such as maximum thrust force and maximum thrust force variation. Consequently, it is suitable for applications where there is high confidence that the actuators will not surpass their physical limitations.

4.1.2 Mixed Optimization Problem

The mixed optimization problem is a conventional problem that involves minimizing an objective function. This objective function is a combination of multiple terms, each weighted by appropriate coefficients. Each of these terms represents a specific goal, and their importance and priority are determined by the assigned weighting coefficients. In the general form of a mixed optimization problem, constraints can also be incorporated [10]. An illustration of a mixed optimization problem within the context of CA is reported in expression 4.6.

$$\min_v \|Bv - u_v\| + \gamma \|v - v_p\| \tag{4.6}$$

Expression 4.6 demonstrates that this minimization problem can be divided into two parts for analysis. In the first part, the objective is to minimize the disparity between the virtual control input and the actual control input. In the second part, the aim is to acquire a desired actuation vector. These two tasks can be referred to as an error minimization problem and a preferential actuation vector problem, respectively. The effectiveness of this optimization problem depends on the tuning of the relative weight, denoted as gamma.

In summary, the mixed optimization problem is a highly versatile problem that allows for the consideration of various constraints and the inclusion of secondary goals. While its main drawback is the computational cost and running time, which can be critical for miniaturized spacecraft like Cubesats, it is possible to mitigate the computational complexity issue by reformulating the optimization problem as a linear programming problem. By assuming linearity in the objective function and constraints, the mixed optimization problem can be transformed into

a linear programming formulation [11]. The advantage of this approach is that there are existing algorithms that can solve linear programming problems quickly and efficiently.

4.2 Mixed Optimization Problem Formulation

In this section, the goal is to establish a mixed optimization problem formulation for the CA of the Cubesat under analysis. Initially, the formulation of the mixed optimization problem will be presented, followed by its re-formulation into a linear programming problem. This process will be carried out for both a Cubesat with a static thruster configuration and a Cubesat with a dynamic thruster configuration. The static and dynamic thruster configuration were discussed in Chapter 2.

4.2.1 Static thruster configuration

For the static thruster configuration case, the actuator model can be described by a linear relation similar to Equation 4.1. The specific B matrix used for the Cubesat under study is given by

$$B = \begin{bmatrix} -\sin(\theta) & -\sin(\theta) & -\sin(\theta) & -\sin(\theta) & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ \cos(\theta) & \cos(\theta) & -\cos(\theta) & -\cos(\theta) & 0 & 0 & 0 \end{bmatrix} \quad (4.7)$$

where $\theta = 15^\circ$. A first formulation of the mixed optimization problem where no thruster constraints are taken into account is given by

$$\begin{aligned} \min_{e,v} \quad & \gamma \|e\|_1 + \delta \|v\|_1 \\ \text{subject to} \quad & \\ & e = u_v - Bv \end{aligned} \quad (4.8)$$

Once again, this minimization problem can be broken down in 2 sub-tasks. The first task is the error minimization between the virtual control input uv and the control input u , while the second task involves the minimization of the thrust force vector v . The first part must be addressed to replicate the expected control input computed by the controller while the second part concerns the minimization the fuel consumption. The importance of these two task is given by their weighting factors. Since the error minimization task is the primary task, the weighting factor γ is going to be much greater than the weighting factor δ . Unfortunately

this formulation present an issue. The norm selected for the minimization of the actuation is the 1-norm; at first this seems the best choice since the 1-norm is directly related to the fuel consumption optimization. However, it may also cause the algorithm to prioritize few actuators. This may cause few thrusters to be too stressed and overloaded and may cause the premature failure of such thrusters. To fix this, a new formulation of the mixed optimization problem is needed and it is given by

$$\begin{aligned}
 & \min_{e,v} && \gamma\|e\|_1 + \delta\|v\|_1 + \epsilon\|v\|_\infty \\
 & \text{subject to} && \\
 & && e = u_v - Bv
 \end{aligned} \tag{4.9}$$

In this revised formulation, an additional task is introduced along with its corresponding weighting factor ϵ . This new task involves the minimization of the thrust force, represented by the vector v , in the infinity norm. The infinity norm of the thrust force vector v is not directly linked to fuel consumption but instead promotes a more balanced distribution of the control input among the actuators. By definition, the infinity norm of a vector minimizes the maximum entry of that vector. Moreover, the importance assigned to this task is lower compared to the other two tasks. As previously discussed, the primary goal is to minimize the error between the virtual control input u_v and the actual control input u , followed by the secondary goal of minimizing fuel consumption, and finally, the tertiary goal of achieving an even distribution of the thrust force among the actuators. The next step is to refine the previous formulation by taking into account the constraints of the physical actuators. The constraints to be considered are the following:

- Maximum thrust force: The thruster is capable of producing a maximum thrust force of 0.05 N, which is an upper limit that cannot be surpassed.
- Thrust force variation: The frequency of the thrust force vector is limited, meaning that the magnitude of the thrust force cannot undergo significant changes between consecutive time intervals.

The final mixed optimization problem formulation is reported in Expression 4.10.

$$\begin{aligned}
 & \min_{e,v} && \gamma\|e\|_1 + \delta\|v\|_1 + \epsilon\|v\|_\infty \\
 & \text{subject to} && \\
 & && e = u_v - Bv \\
 & && \|v\|_1 \leq F_{\max} \\
 & && \|\dot{v}\|_1 \leq \dot{F}_{\max}
 \end{aligned} \tag{4.10}$$

The subsequent step involves transforming the problem into a linear programming minimization problem. Given that the problem does not show any non-linearities, it is possible to introduce appropriate slack variables and rearrange the constraints to obtain the problem in a linear programming form. The most general representation of a linear programming problem is depicted in Expression 4.11.

$$\begin{aligned}
 & \min_x && c^T x + d \\
 & \text{subject to} && \\
 & && A_{\text{eq}} x = b_{\text{eq}} \\
 & && Ax \leq b
 \end{aligned} \tag{4.11}$$

Initially, the objective function is modified by introducing slack variables. The new objective function is given by

$$\gamma * \lambda + \delta * \mu + \epsilon * \nu \tag{4.12}$$

where λ , μ and ν are the new introduced slack variables. The constraints deriving from the introduction of these slack variables are reported in Equation 4.13, 4.14 and 4.15.

$$\begin{cases} e - \lambda < 0 \\ -e - \lambda < 0 \end{cases} \tag{4.13}$$

$$\begin{cases} v - \mu < 0 \\ -v - \mu < 0 \end{cases} \tag{4.14}$$

$$\begin{cases} v - \nu < 0 \\ -v - \nu < 0 \end{cases} \tag{4.15}$$

Moreover, with the introduction of slack variables, the number of optimization variables increases. In particular, the total optimization variables are e , v , λ , μ , and ν . Regarding the constraints related to the thrust force and thrust force variation, they also need to be re-formulated for the linear programming re-formulation. The re-formulation of the thrust force constraint is reported in Equation 4.16.

$$\begin{cases} v < F_{\text{max}} \\ -v < -F_{\text{min}} \end{cases} \tag{4.16}$$

The constraint about the thrust force is re-written as follows

$$\frac{|v - v_0|}{\Delta t} \leq \dot{F}_{\max} \quad (4.17)$$

where Δt represents the difference between two consecutive time instants (depending on the sample time of the feedback control system), and v_0 is the actuation vector at the previous time instant. The final formulation of the thrust force variation constraint, reported in Equation 4.18 is obtained by moving the optimization variable on the left and the remaining variables on the right.

$$\begin{cases} v < \dot{F}_{\max}\Delta t + v_0 \\ -v < \dot{F}_{\max}\Delta t - v_0 \end{cases} \quad (4.18)$$

Finally, the resulting linear programming formulation of the mixed optimization problem is reported in Expression 4.19.

$$\begin{aligned} & \min_{e, v, \lambda, \mu, \nu} \quad \gamma\lambda + \delta\mu + \epsilon\nu \\ & \text{subject to} \quad \begin{cases} e + Bv = u_v \\ e - \lambda < 0 \\ -e - \lambda < 0 \\ v - \mu < 0 \\ -v - \mu < 0 \\ v - \nu < 0 \\ -v - \nu < 0 \\ v < F_{\max} \\ -v < -F_{\min} \\ v < \dot{F}_{\max}\Delta t + v_0 \\ -v < \dot{F}_{\max}\Delta t - v_0 \end{cases} \end{aligned} \quad (4.19)$$

Throughout the rest of this thesis project, the linear programming formulation of the mixed optimization problem with fixed thrusters will be addressed as "static linear programming".

4.2.2 Dynamic thruster configuration

When assuming moving thruster, the relation between the control input u and the thrust force v becomes more complicated as shown in Equation 4.20 and 4.21.

$$u = f(v, \theta) \tag{4.20}$$

$$+ \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} = \begin{bmatrix} -\sin(\theta) & -\sin(\theta) & -\sin(\theta) & -\sin(\theta) & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ \cos(\theta) & \cos(\theta) & -\cos(\theta) & -\cos(\theta) & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{bmatrix} \tag{4.21}$$

The control input u is influenced by both the thrust force vector V and the deflection angle θ of the thruster, as demonstrated in Equation 4.20 and 4.21. This relationship exhibits a strong nonlinearity. An initial formulation of the mixed optimization problem is given by

$$\begin{aligned} \min_{e,v} \quad & \gamma \|e\|_1 + \delta \|v\|_1 + \epsilon \|v\|_\infty \\ \text{subject to} \quad & \\ & e = u_v - f(v, \theta) \end{aligned} \tag{4.22}$$

where the deflection angle θ appears as an additional optimization variable. The subsequent stage involves refining the mixed optimization problem by incorporating the physical constraints specific to the thrusters:

- Maximum thrust force: The thruster is capable of producing a maximum thrust force of 0.05 N, which is an upper limit that cannot be surpassed.
- Thrust force variation: The frequency of the thrust force vector is limited, meaning that the magnitude of the thrust force cannot undergo significant changes between consecutive time intervals.
- Deflection angle range: The movable thrusters have a range of +/- 5° within their initial configuration.
- Deflection angle variation: The thrusters are not allowed to change their deflection angle significantly between consecutive time intervals.

The final mixed optimization problem formulation is illustrated in Expression 4.23.

$$\begin{aligned}
 & \min_{e,v,\theta} && \gamma \|e\|_1 + \delta \|v\|_1 + \epsilon \|v\|_\infty \\
 & \text{subject to} && \\
 & && e = u_v - f(v, \theta) \\
 & && \|v\|_1 \leq F_{\max} \\
 & && \|\dot{v}\|_1 \leq \dot{F}_{\max} \\
 & && \|\theta\|_1 \leq \theta_{\max} \\
 & && \|\dot{\theta}\|_1 \leq \dot{\theta}_{\max}
 \end{aligned} \tag{4.23}$$

The following step involves transforming the problem into a minimization problem in linear programming form. However, the constraint related to the error between the virtual control input u_v and the actual control input u is expressed through a highly non-linear relationship. Hence, it is necessary to linearize the actuator model. This issue is tackled by employing the first-order Taylor approximation. The linearized actuator model is depicted in Equation 4.24.

$$f(v, \theta) \approx f(v_0, \theta_0) + \left. \frac{\partial f}{\partial v} \right|_{(v_0, \theta_0)} (v - v_0) + \left. \frac{\partial f}{\partial \theta} \right|_{(v_0, \theta_0)} (\theta - \theta_0) \tag{4.24}$$

By equating

$$\begin{aligned}
 f(v_0, \theta_0) &= f_0, \\
 \left. \frac{\partial f}{\partial v} \right|_{(v_0, \theta_0)} &= F_v, \\
 \left. \frac{\partial f}{\partial \theta} \right|_{(v_0, \theta_0)} &= F_\theta,
 \end{aligned}$$

the linearized actuator model present a more compact form, illustrated in Equation 4.25.

$$f(v, \theta) \approx f_0 + F_v(v - v_0) + F_\theta(\theta - \theta_0) \tag{4.25}$$

The validity of the linearized model is maintained as long as the variables (v, θ) remain in the neighbourhood of the linearization point (v_0, θ_0) . To prevent them from deviating beyond this range, the model is updated during each iteration of the CA application. At every iteration a new linearized actuator model is produced

by considering an updated linearization point (v_0, θ_0) . Such a linearization point indicates the value of v and θ from the previous time step. The constant update of the linearized actuator model ensures that the performance of the CA allocation remains unaffected by potential inaccuracies in the model.

The linear programming formulation of the mixed optimization problem is reported in Expression 4.26.

$$\begin{aligned}
 & \min_{e, v, \lambda, \mu, \nu} \quad \gamma\lambda + \delta\mu + \epsilon\nu \\
 & \text{subject to} \quad \left\{ \begin{array}{l}
 e + F_v v + F_\theta \theta = u_v - f_0 + F_v v_0 + F_\theta \theta_0 \\
 e - \lambda < 0 \\
 -e - \lambda < 0 \\
 v - \mu < 0 \\
 -v - \mu < 0 \\
 v - \nu < 0 \\
 -v - \nu < 0 \\
 v < F_{\max} \\
 -v < -F_{\min} \\
 v < \dot{F}_{\max} \Delta t + v_0 \\
 -v < \dot{F}_{\max} \Delta t - v_0 \\
 \theta < F_{\max} \\
 -\theta < -F_{\min} \\
 \theta < \dot{\theta}_{\max} \Delta t + \theta_0 \\
 -\theta < \dot{\theta}_{\max} \Delta t - \theta_0
 \end{array} \right. \tag{4.26}
 \end{aligned}$$

Throughout the rest of this thesis project, the linear programming formulation of the mixed optimization problem with moving thrusters will be addressed as "dynamic linear programming".

4.3 Pulse Width Pulse Frequency Modulator

The CA method developed in the preceding sections is based on the assumption of continuous thrusters, which implies that the thrusters can deliver any desired thrust force within their physical limitations. However, in the current scenario, the thrusters are assumed to be on-off, meaning they can only provide either the maximum thrust force or no thrust force at all. To bridge this gap, a pulse width pulse frequency modulator is employed to convert the continuous signal generated by the CA into the on-off signal required by the actuators. Figure 4.2 illustrates

the role of the PWPF modulator.

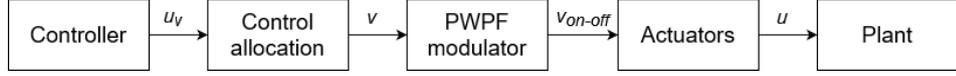


Figure 4.2: CA Scheme with PWPF modulator

The Pulse Width Pulse Frequency Modulator (PWPF Modulator) is a modulation technique used in electronic systems to convert an analog signal into a digital signal. The PWPF modulator takes in an analog signal as its input. This signal represents the desired amplitude or value to be transmitted. The first step in the modulation process is pulse width modulation (PWM). PWM involves comparing the amplitude of the analog signal with a reference voltage or a fixed threshold. The duration of the pulse (pulse width) is determined based on this comparison. If the analog signal is higher than the reference voltage, the pulse width will be longer, and if it is lower, the pulse width will be shorter. After pulse width modulation, the resulting pulse train is further processed using pulse frequency modulation (PFM). PFM involves varying the frequency of the pulses based on the value of the analog signal. Higher amplitudes will result in a higher pulse frequency, while lower amplitudes will lead to a lower pulse frequency. The final output of the PWPF modulator is a digital signal that consists of pulses with varying widths and frequencies. The PWPF modulator scheme is shown in Figure 4.3.

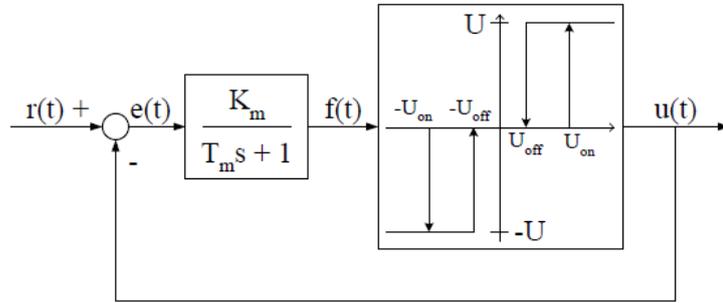


Figure 4.3: PWPF modulator scheme

Figure 4.3 shows that the PWPF modulator comprises a Schmitt trigger and a first-order filter integrated within a feedback loop. The tuning of the PWPF modulator plays a critical role in maximizing its performance. To achieve this, several tunable parameters need to be identified. Specifically, K_m and T_m represent the gain and time constant of the first-order filter, while U_{on} and U_{off} correspond

to the two thresholds of the Schmitt trigger. These parameters, K_m , T_m , U_{on} , and U_{off} , significantly influence the behavior and output of the PWPF modulator, and their proper adjustment is crucial to optimize its performance. Krovel provides a detailed explanation of the tuning process for the PWPF modulator [12]. However, due to the scope limitations of this thesis, a comprehensive analysis of the PWPF modulator tuning is not included. Instead, only the final results of the tuning process are reported, and these results are summarized in Table 4.1.

PWPF modulator tuning	
Parameter	Value
K_m	7.5
T_m	0.65
U_{on}	0.35
h	0.2

Table 4.1: PWPF modulator tuning

Chapter 5

Simulation and Results

In this chapter, a mission scenario is introduced to evaluate the performance of the CA algorithm developed in Chapter 4. The mission scenario is outlined as follows: initially, a 3U Cubesat executes a radial boost maneuver. Subsequently, the 3U Cubesat docks with another 3U Cubesat, resulting in the formation of a 6U Cubesat. Finally, the 6U Cubesat performs a straight-line V-bar approach maneuver. It is important to note that the docking maneuver of the two 3U Cubesats will not be extensively, as it falls outside the scope of the thesis. In this chapter, the mission scenario is simulated multiple times using different CA techniques. Specifically, three simulations are conducted. The first simulation utilizes the pseudo-inverse CA, the second simulation employs the static linear programming CA and the third simulation adopts the dynamic linear programming CA. In the context of aerospace application, fuel consumption is typically quantified using the ΔV concept. ΔV represents the change in velocity that a spacecraft must undergo to execute a specific maneuver. The ΔV quantity is strictly related to the control input u : the greater is the signal norm of the control input, the greater is the ΔV and the greater is the fuel consumption. It is important to clarify the radial boost maneuver and the straight line V-bar approach maneuver can be initially approximated as impulsive maneuvers. Impulsive maneuvers involve instantaneous changes in velocity, requiring infinite amount of thrust force. Real-world constraints such as limited thrust levels make the achievement of purely impulsive maneuvers unattainable. Instead, constant thrust forces need to be applied over a certain duration to successfully execute the maneuver. Nevertheless, the case of impulsive maneuvers are the most ideal types of maneuver, requiring the least amount of ΔV , therefore, spending the least amount of fuel. The studying of impulsive maneuver is crucial to determine the minimum ΔV required to perform a specific maneuver which can be used as a benchmark to compare the same maneuvers adopting feasible CA techniques.

5.1 Simulation Environment

The simulation environment employed for this thesis project is Matlab & Simulink. The complete Simulink scheme is reported in Figure 5.1.

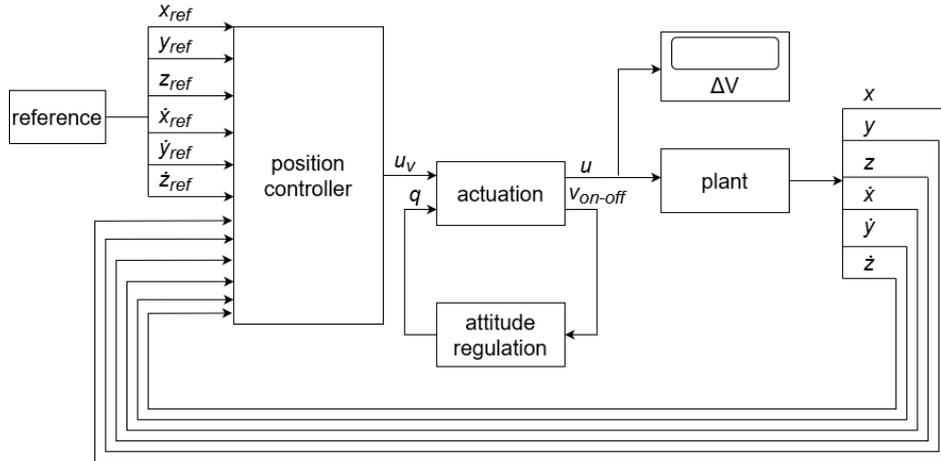


Figure 5.1: Simulink Scheme

The scheme includes six identifiable blocks: the reference block, the position controller block, the actuation block, the plant block, the attitude controller block, and the ΔV block. The plant block describes the spacecraft model using the Clohessy-Wiltshire (CW) equations of motion discussed in Chapter 2. The position controller block represents the LQR controller presented in Chapter 3, while the attitude controller block represents the controller for attitude regulation, also discussed in Chapter 3. Additionally, the Simulink scheme includes the reference block, which provides the reference trajectory based on the chosen maneuver. Next, the actuation block incorporates the CA algorithm, PWPF modulator, and the actuator model. Finally, the ΔV block computes the ΔV for the current maneuver from the control input signal u . As previously mentioned, the presented scheme shows a feedback control system, as the signal from the plant is fed back into the controller. It is noteworthy to observe the interaction between the attitude controller block and the actuation block. Depending on the satellite's orientation, the thrusters need to be fired accordingly. Conversely, the spacecraft's attitude is affected by the actuation: indeed, any time a thruster fires, the orientation of the satellite may be disturbed.

5.2 Radial Boost Maneuver

Before discussing the simulation of the system for this particular maneuver, it is essential to provide an introduction to the maneuver itself. Radial maneuvers involve firing the thrust in the radial direction and are employed for transfers along the target orbit. Figure 5.2 illustrates an example of a radial boost maneuver.

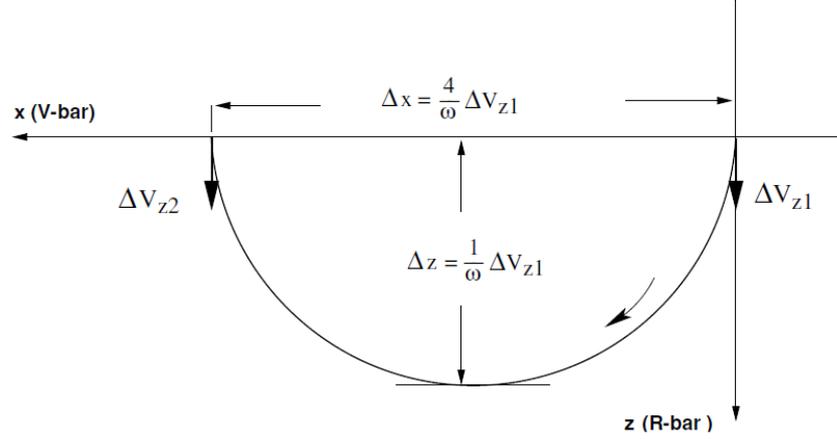


Figure 5.2: Transfer along V-bar by radial impulses [5]

Figure 5.2 shows the maneuver trajectory in the LVLH frame. The V-bar indicates the tangential axis while the R-bar represents the radial axis. By applying a first radial thrust, ΔV_{z1} , along the R-bar, followed by a second radial thrust, ΔV_{z2} , after half an orbital period $T/2$, the spacecraft progresses along the V-bar axis by a distance of δx given by Equation 5.1.

$$\Delta x = \frac{4}{\omega} \Delta V_{z1} \quad (5.1)$$

The required ΔV is given by Equation 5.2.

$$\Delta V_{z1} = \Delta V_{z2} = \frac{\omega}{4} \Delta x \quad (5.2)$$

The total ΔV required for such a two-pulse manoeuvre is reported in 5.3.

$$\Delta V_{\text{tot}} = \frac{\omega}{4} \Delta x \quad (5.3)$$

Equation 5.3 represents the minimum ΔV required for this maneuver. This value will be compared with the ΔV provided by the simulated system. Regarding the first part of the mission scenario, the spacecraft will undergo both a position change through a radial boost maneuver and an attitude regulation. Regarding the radial maneuver, the chaser spacecraft, specifically the 3U Cubesat, is required to approach a target located 150 meters ahead along the V-bar axis. The chaser needs to advance by 100 meters in total. Hence, the initial position of the chaser is given by $x_i = [-150, 0, 0]$, and the final position is $x_f = [-50, 0, 0]$. The Cubesat needs to follow the trajectory illustrated in Figure 5.3.

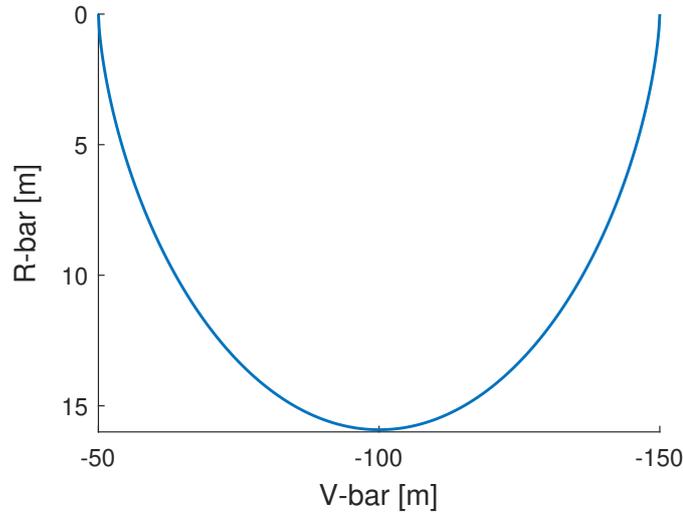


Figure 5.3: Radial boost maneuver reference

In terms of attitude regulation, the Cubesat is initially assumed to have a random orientation relative to the target. The objective is to align the Cubesat with the target, in a way that there is no relative rotation between the body frame of the Cubesat and the LVLH frame of the target. The chaser initial attitude is $q_0 = [0.688, 0.577, -0.413, -0.145]$ and $q_f = [1, 0, 0, 0]$. The system is simulated using the three CA techniques analyzed in Chapter 4: the pseudo-inverse CA, the static linear programming CA and the dynamic linear programming. The simulation time corresponds to the time needed to perform the radial boost maneuver, which corresponds to one orbital period $T = 5572$ seconds. The simulation result are reported below.

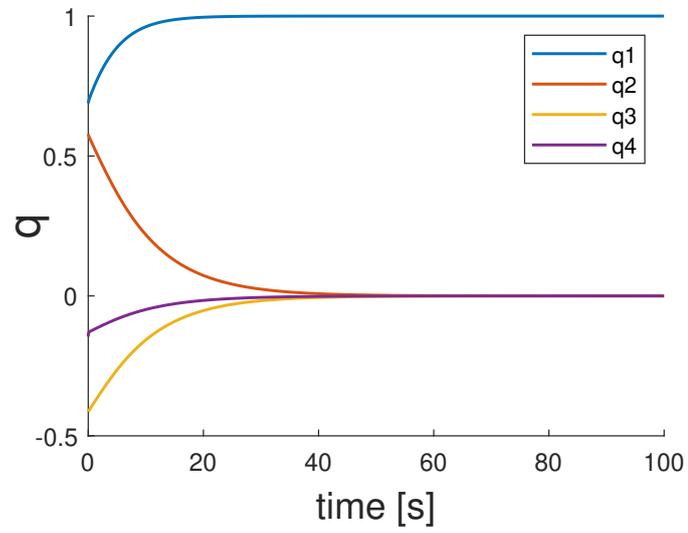


Figure 5.4: Attitude regulation

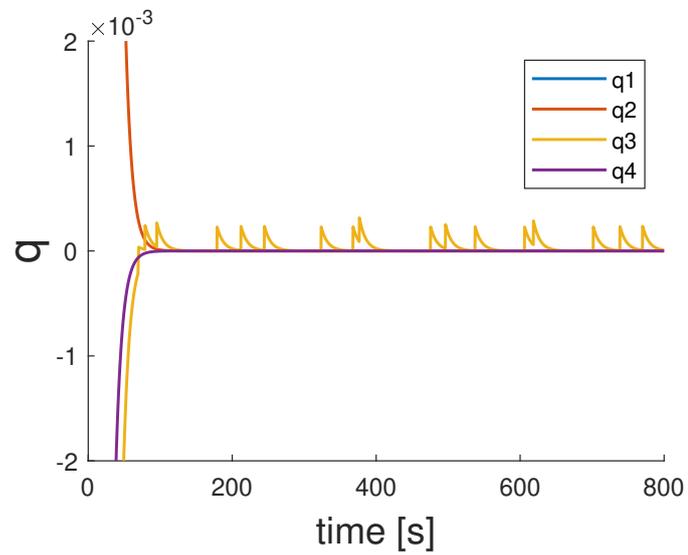


Figure 5.5: Attitude regulation spikes

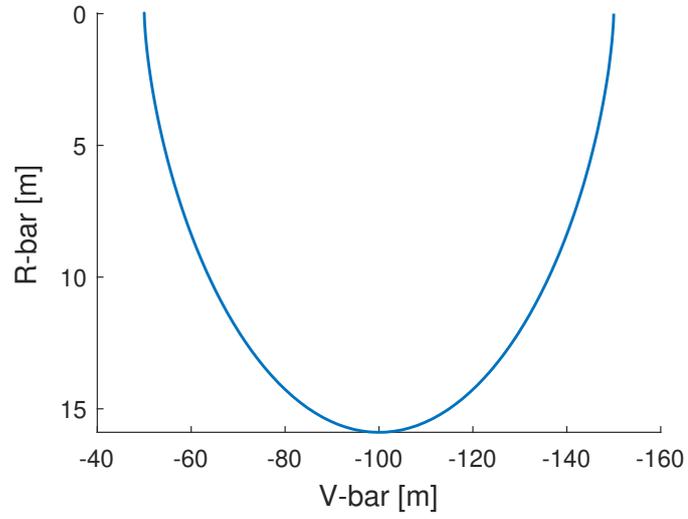


Figure 5.6: Radial boost with CA

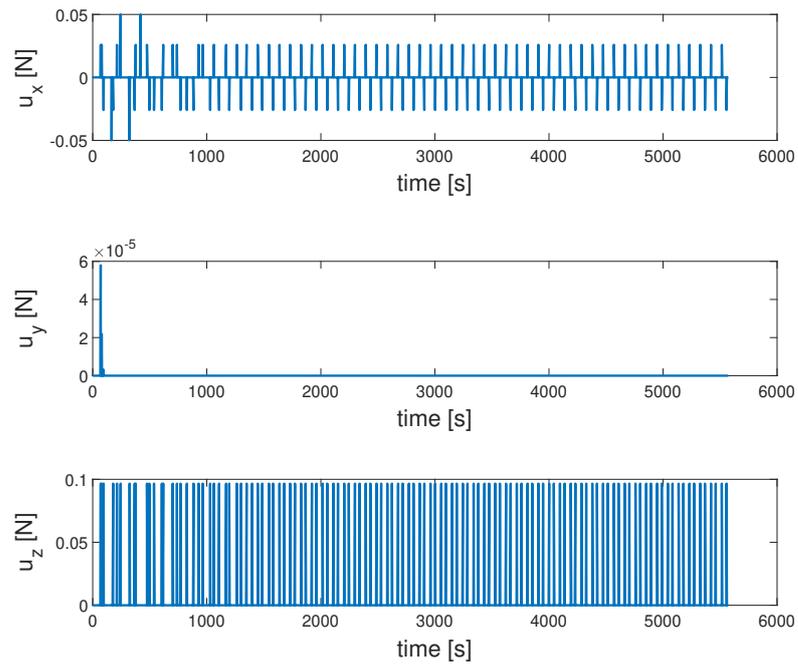


Figure 5.7: Radial boost control input using pseudo-inverse CA

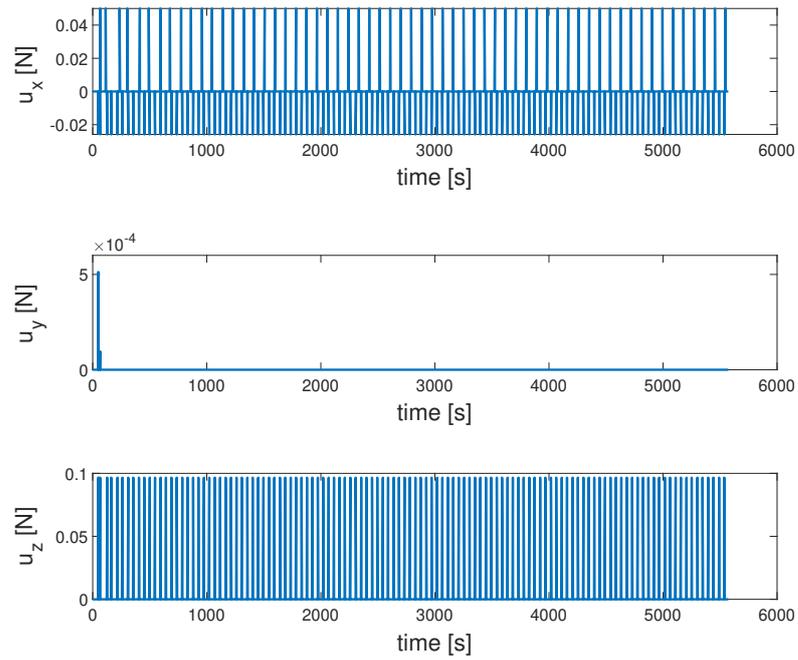


Figure 5.8: Radial boost control input using static linear programming CA

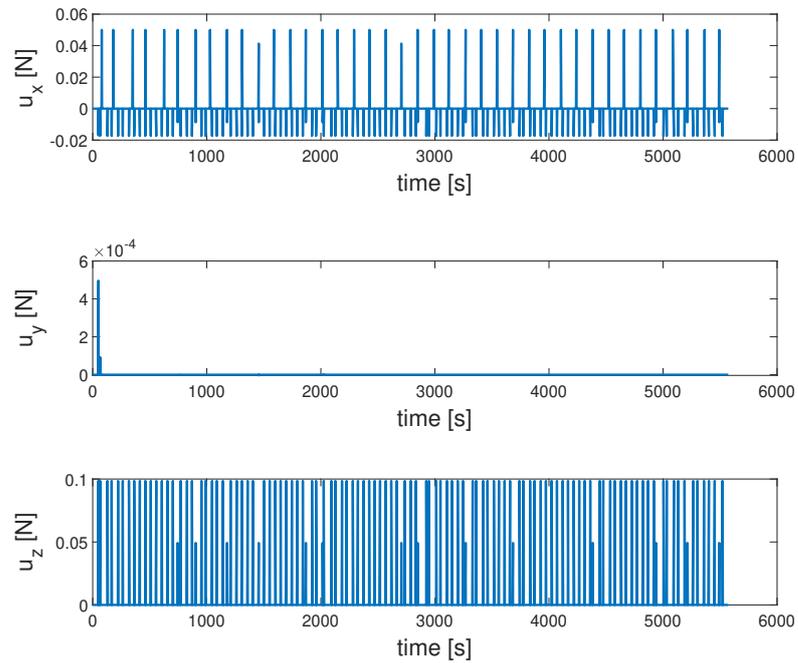


Figure 5.9: Radial boost control input using dynamic linear programming CA

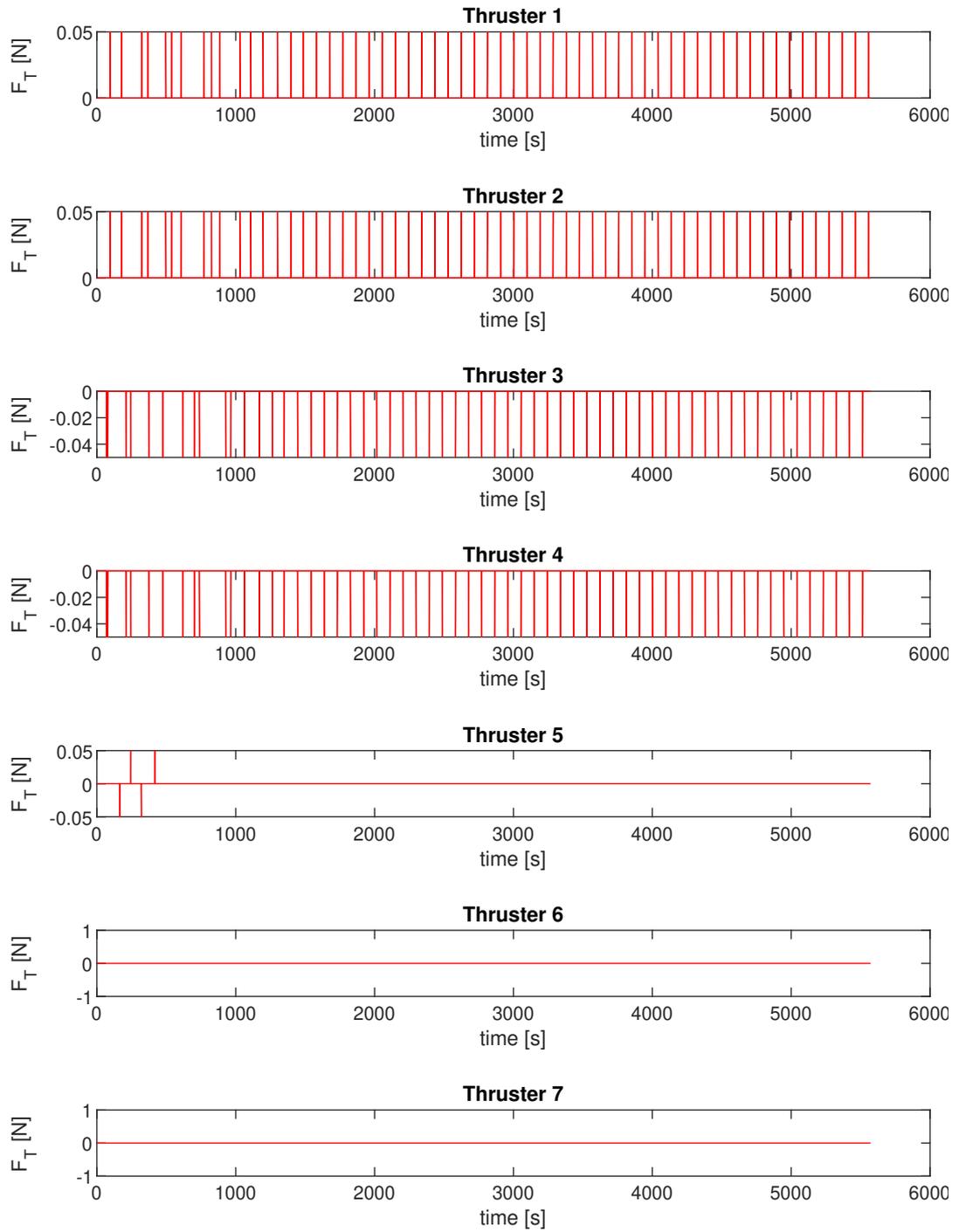


Figure 5.10: Pseudo-inverse thrust force

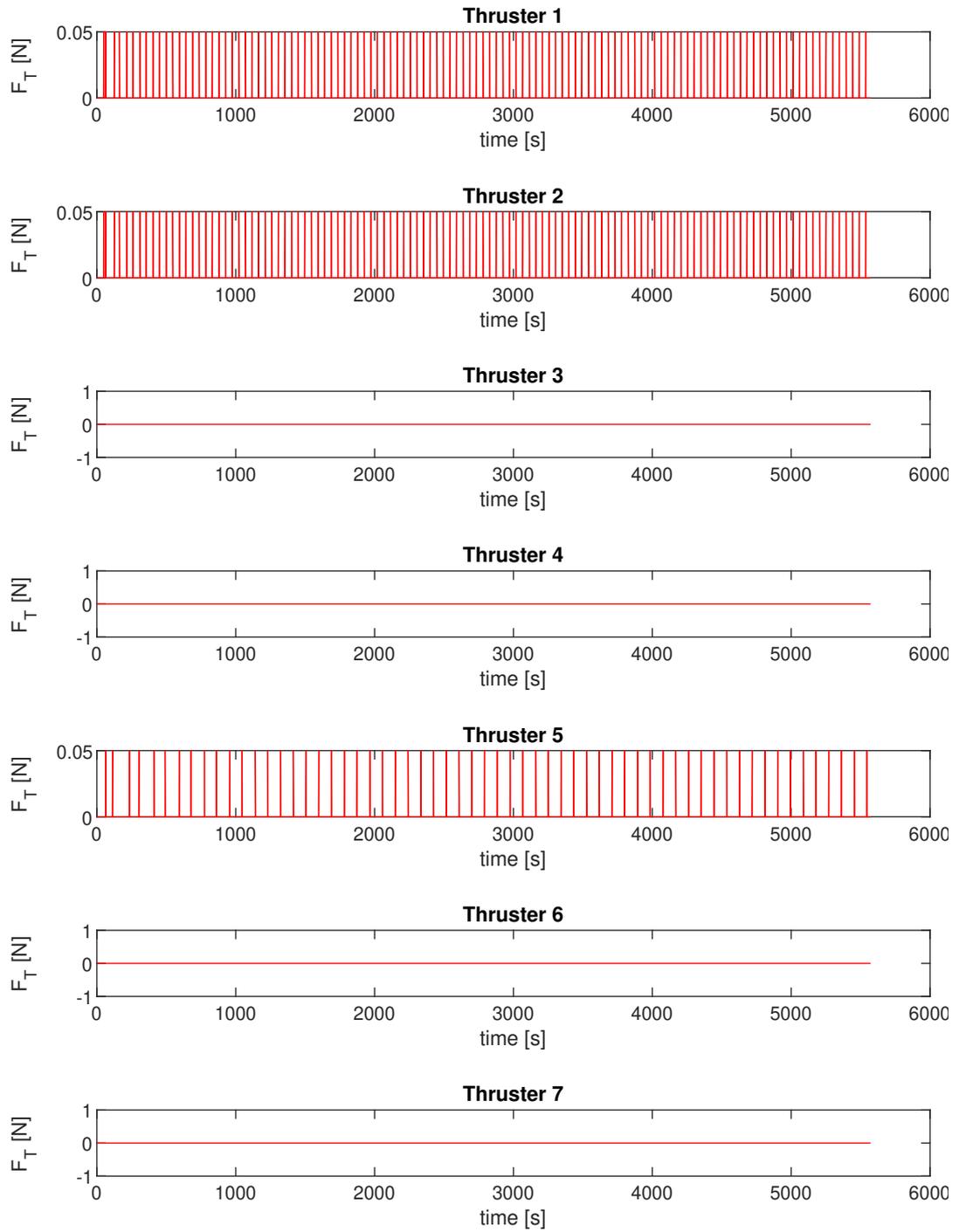


Figure 5.11: Static linear programming thrust force

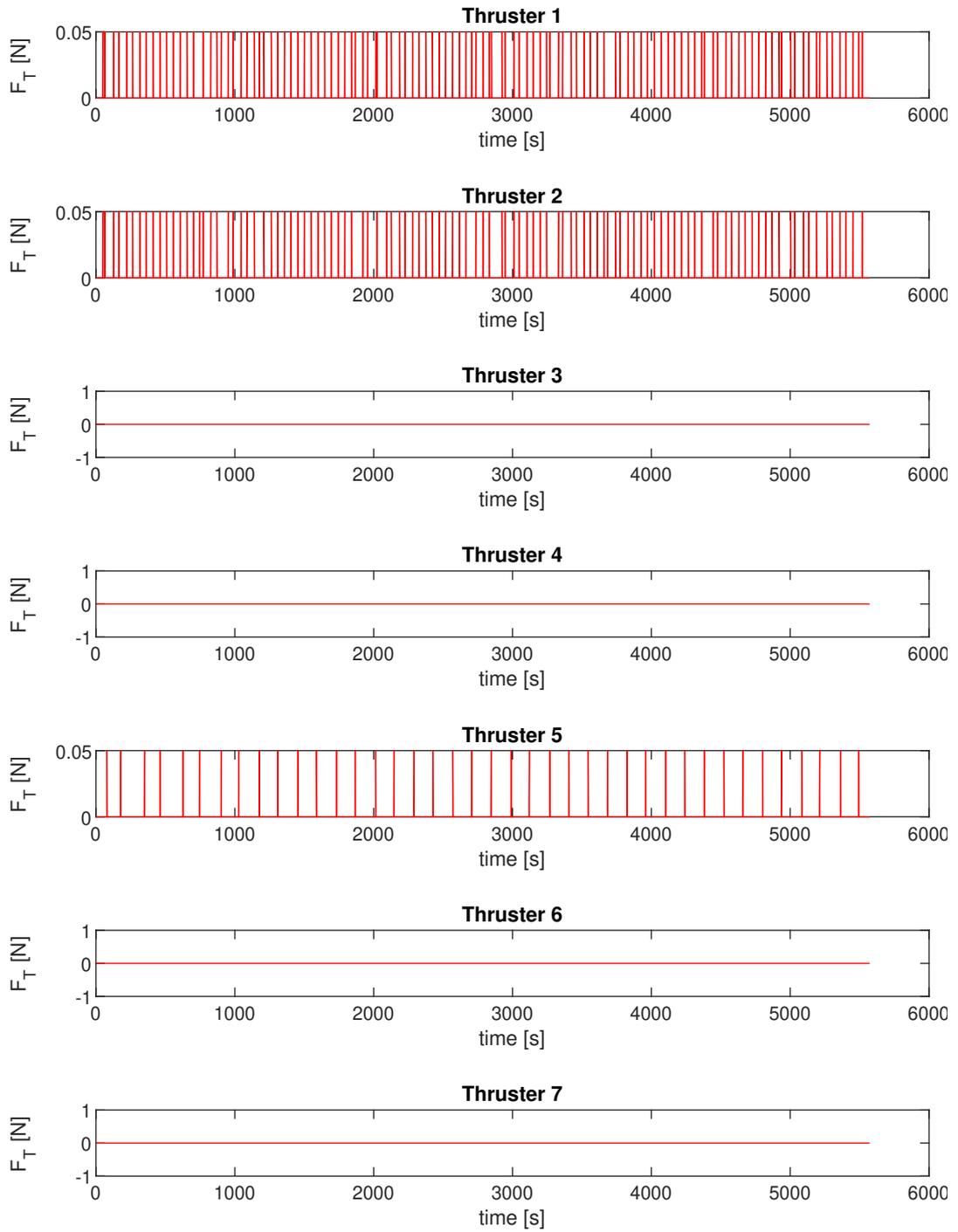


Figure 5.12: Dynamic linear programming thrust force

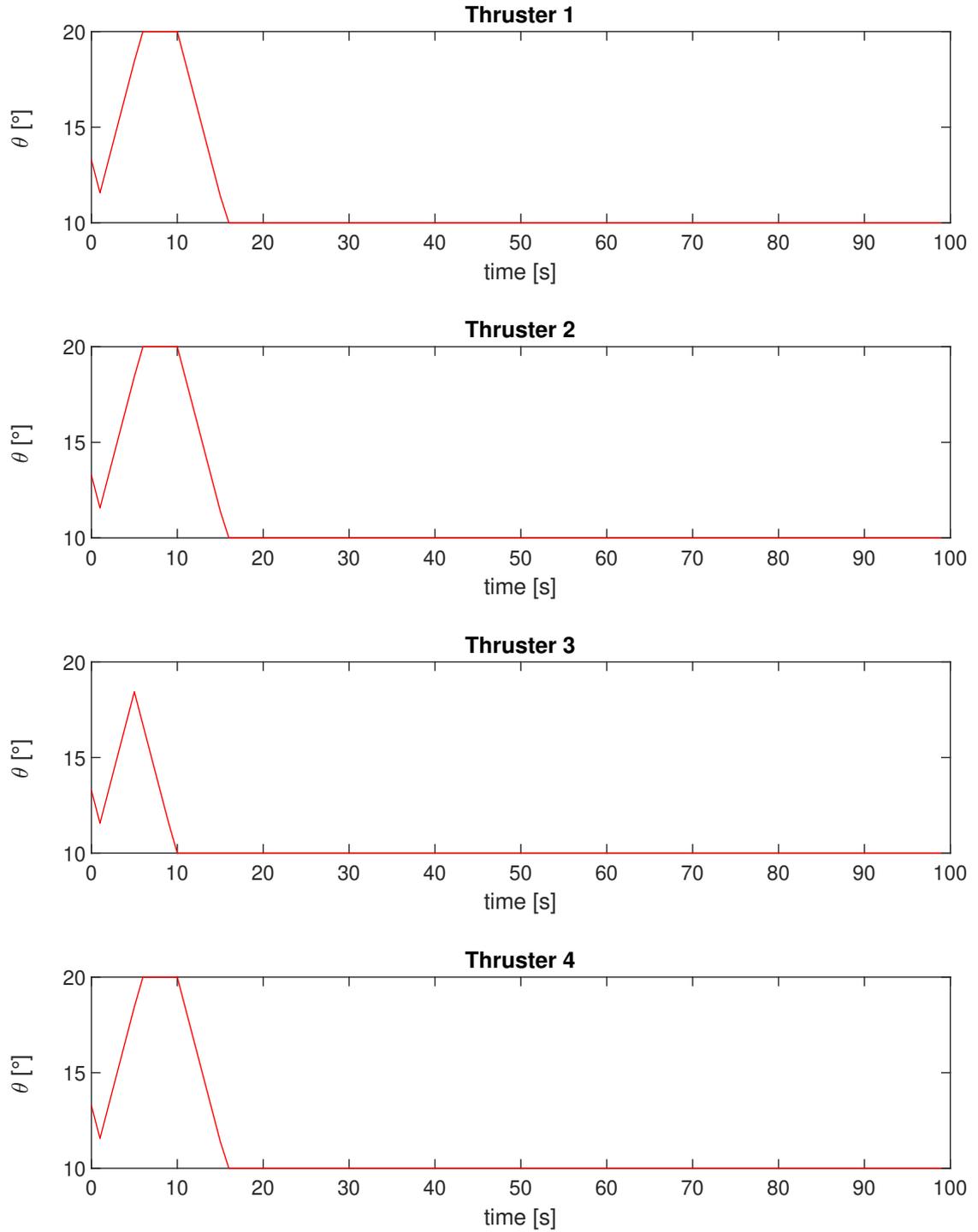


Figure 5.13: Thruster deflection angles

Maneuver	Minimum ΔV
Radial Boost	0.028 [m/s]

Table 5.1: Minimum ΔV required

Radial Boost Maneuver	
3U Cubesat	
Control Allocation	ΔV [m/s]
pseudo-inverse	0.073
static linear programming	0.087
dynamic linear programming	0.076

Table 5.2: Radial boost maneuver results

Figure 5.4 presents the attitude regulation carried out by the 3U Cubesat utilizing all three previously mentioned CA techniques. Only one figure is reported due to the similar performance of the CA methods in achieving the desired attitude. Figure 5.4 shows that the desired attitude was successfully attained within a duration of 50 seconds. However, it should be noted that the CA task and the attitude regulation task are not completely independent. In Figure 5.5, a zoomed-in graph of the attitude regulation is presented. It is observed that spikes appear on the line representing the quaternions. These spikes correspond to the disturbances caused by the thrusters upon firing. Fortunately, the magnitude of these disturbances is relatively small, resulting in no significant impact on the spacecraft. Figure 5.6 illustrates the positional tracking of the radial maneuver for the three CA methods. Similarly to the attitude regulation, only one figure is provided as the performance of the CA techniques in executing the required maneuver is very similar. Figure 5.6 demonstrates the successful execution of the radial boost maneuver. Figure 5.10, Figure 5.11, and Figure 5.12 display the thrust force v , exerted by the thrusters for each CA technique. It is clearly evident that the thrust force generated by the pseudo-inverse CA method differs significantly from that provided by the static and dynamic linear programming CA. The pseudo-inverse CA utilizes mainly thrusters 1, 2, 3, and 4, while the static and dynamic linear programming CA utilize thrusters 1, 2, and 5. Figure 5.7, 5.8 and 5.9 illustrates the control input for each CA techniques. Additionally, Figure 5.13 depicts the changes in the thruster angles in the case of dynamic linear programming. It is worth noting that only the graphs of Thruster 1 and Thruster 2 are meaningful, as Thruster 3 and Thruster 4 are not actuated, as shown in Figure 5.12. Within a duration of 20 seconds, both thruster 1 and thruster 2 reach their most vertically oriented positions relative to the main body of the chaser. This positioning ensures a more effective radial thrust

during the maneuver.

Since the trajectory was accurately tracked by all CA techniques, the remaining criterion for evaluating the performance of these CA strategy is the fuel consumption. By referring to Table 5.1 and 5.2 it is possible to make some considerations. Table 5.1 describes the minimum ΔV required for the maneuver, serving as a benchmark for understanding the deviation of the CA techniques from the ideal impulsive maneuver. Table 5.2 represents the ΔV required for performing the radial maneuver with each CA method. It is evident that the pseudo-inverse CA technique exhibits the lowest fuel consumption, followed by the dynamic linear programming. Unfortunately, the pseudo-inverse method does not consider any constraints. Figure 5.10 shows that the least squares approach assigns negative thrust force to thruster 3 and 4, which is not feasible in practice due to the unidirectional nature of the thrusters mounted on the spacecraft. Consequently, the pseudo-inverse approach cannot be practically implemented. Based on this analysis, the best performing CA method is the dynamic linear programming.

5.3 Cubesat Attachment

Although the docking and the mating maneuvers of the 3U Cubesat are discussed, it is necessary to describe the model of the 6U Cubesat to better understand the forthcoming section. Once again, it is necessary to distinguish between the case with fixed thrusters and the case with moving thrusters.

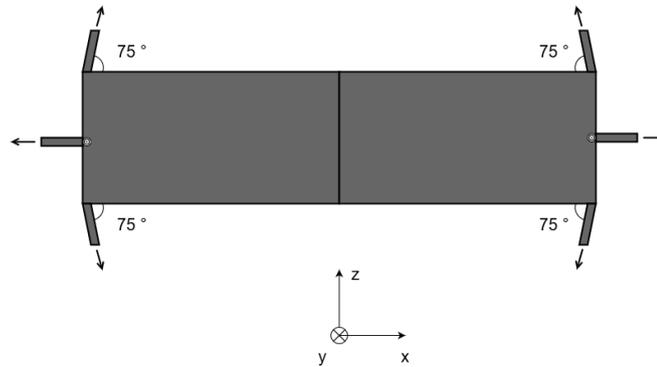


Figure 5.14: 6U Cubesat with fixed thrusters

Figure 5.14, 5.15 and 5.16 describe the 6U Cubesat model with fixed thrusters. The model consists of two 3U Cubesats connected along the x-axis. The overall 6U Cubesat possesses dimensions of length (l) = 10 cm, width (w) = 10 cm, and depth (d) = 60 cm. The mass of the 6U Cubesat is measured at 8 kg. The 6U Cubesat is equipped with 14 thrusters. In particular, thrusters 1, 2, 3, 4, 8,

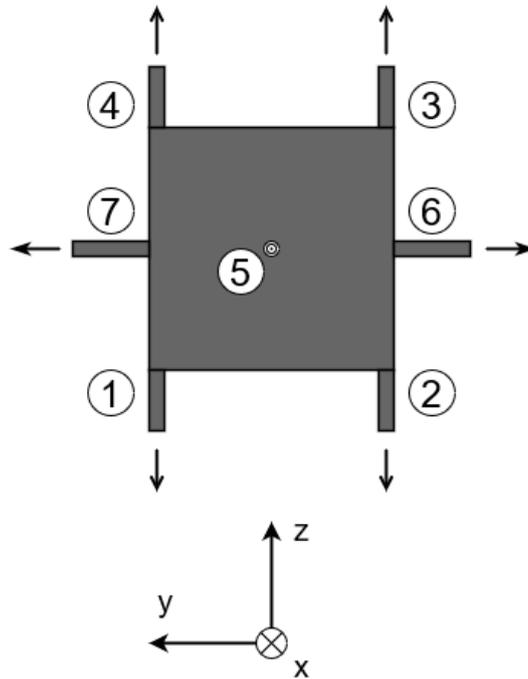


Figure 5.15: 6U Cubesat with fixed thrusters

9, 10, and 11 are inclined at an angle of 75 degrees relative to the spacecraft's surface, while thrusters 5, 6, 7, 12, 13, and 14 are positioned perpendicularly to the spacecraft's surface. Figure 5.17, 5.18 and 5.19 describe the 6U Cubesat model with moving thrusters. Among the 14 thrusters, thruster 1,2,3,4 and 9,10,11,12 can deflect of ± 5 degrees around the nominal deflection angle of 75 degrees. In order to accommodate the increased number of thrusters, the CA strategies developed in Chapter 4 are modified. Specifically, both the static and dynamic linear programming CA approaches are adjusted to utilize only seven out of the available 14 thrusters, specifically thrusters 1, 2, 3, 4, 5, 6, and 7. This selection is based on the following reasoning: each of the 3U Cubesats forming the 6U Cubesat can function as an independent unit capable of generating thrust in any direction. Consequently, utilizing just one unit is sufficient, and in the event of a failure, the second unit can be activated. This ensures that thrust can still be produced in any direction if a fault occurs. To incorporate this concept, additional constraints were introduced into the linear programming formulation. These constraints prevent the use of specific sets of actuators of one of the two 3U Cubesats forming the 6U Cubesat.

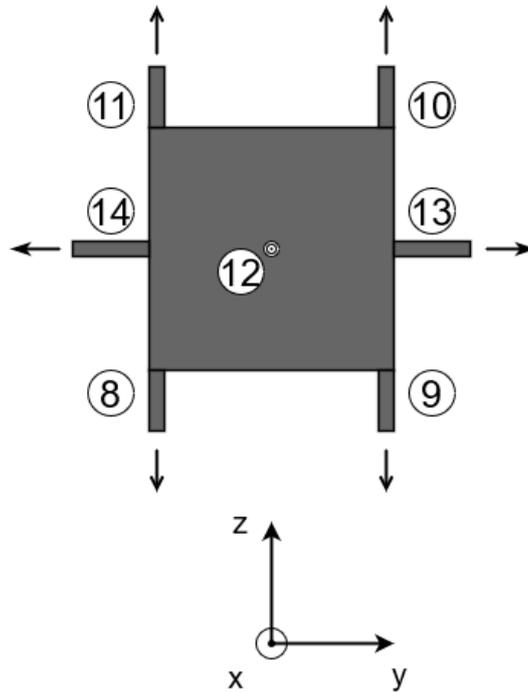


Figure 5.16: 6U Cubesat with fixed thrusters

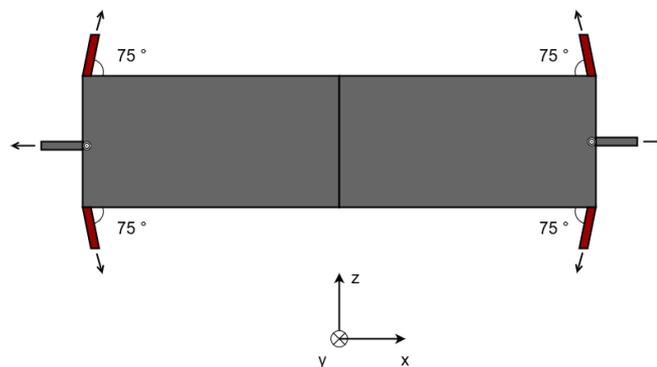


Figure 5.17: 6U Cubesat with moving thrusters

5.4 Straight Line Forced Motion Maneuver

The maneuver performed by the 6U Cubesat is the straight line V-bar approach maneuver. Straight line trajectories are common maneuvers employed for the final approach to the docking port. Figure 5.20 shows the maneuver trajectory.

The trajectory under consideration aims to attain a constant velocity V_x along

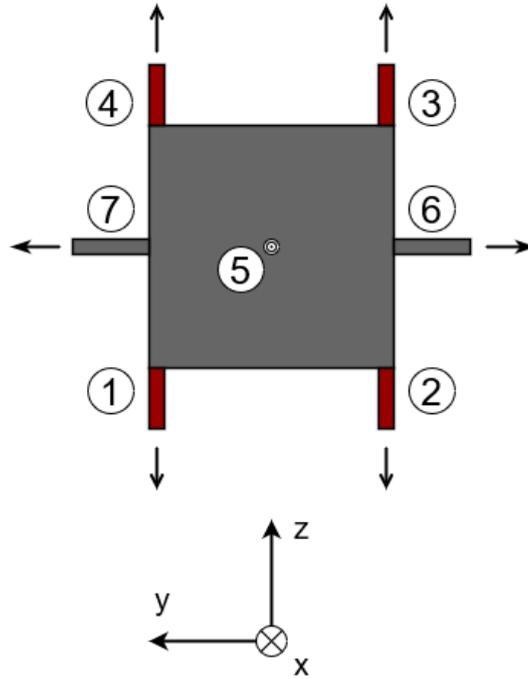


Figure 5.18: 6U Cubesat with moving thrusters

the V-bar axis between positions x_0 and x_1 , while maintaining zero velocities in the other directions. In the simplest scenario, the motion begins with an impulse ΔV_{x1} that imparts the desired velocity V_x in the x-direction. Subsequently, the motion is stopped by applying an impulse of equal magnitude but in the opposite direction ΔV_{x2} . Additionally, throughout the maneuver, a radial force F_z is applied to maintain a velocity of zero along the R-bar axis. The force per mass unit γ_z that must be applied is reported in Equation 5.4.

$$\gamma_z = 2\omega V_x \quad (5.4)$$

The total ΔV required for the transfer from x_0 to x_1 is given by 5.5.

$$\Delta V_{\text{tot}} = |\Delta V_{x1}| + |\gamma_z \Delta t| + |\Delta V_{x2}| \quad (5.5)$$

As previously discussed, straight line trajectories play a crucial role in the final approach to the target, necessitating a high level of precision. To quantify the required level of precision, it is common practice to establish a cone of approach within which the chaser spacecraft must remain. This cone serves as a defined

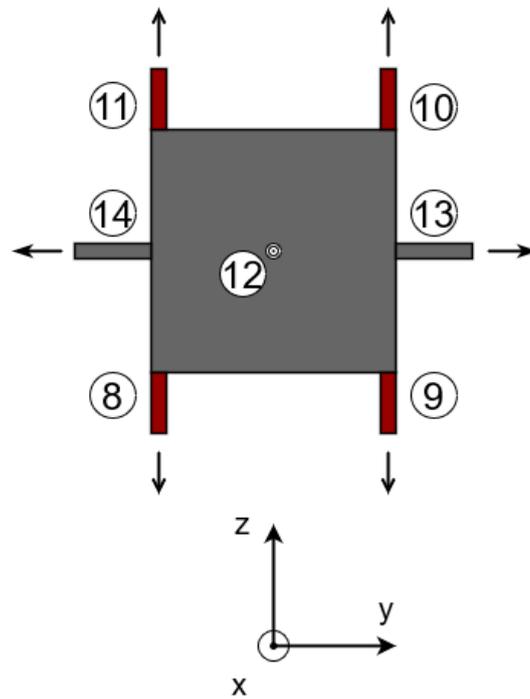


Figure 5.19: 6U Cubesat with moving thrusters

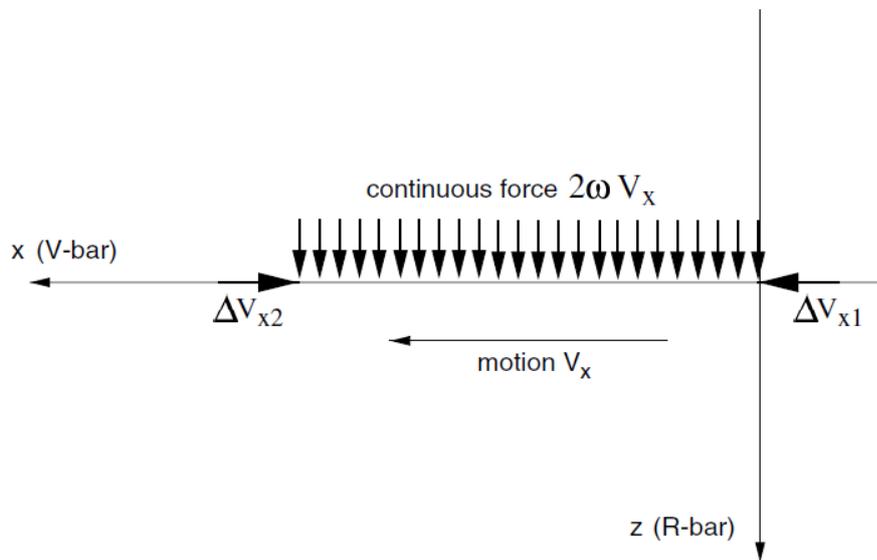


Figure 5.20: Straight line V-bar approach [5]

boundary that ensures the chaser maintains the desired trajectory during the maneuver.

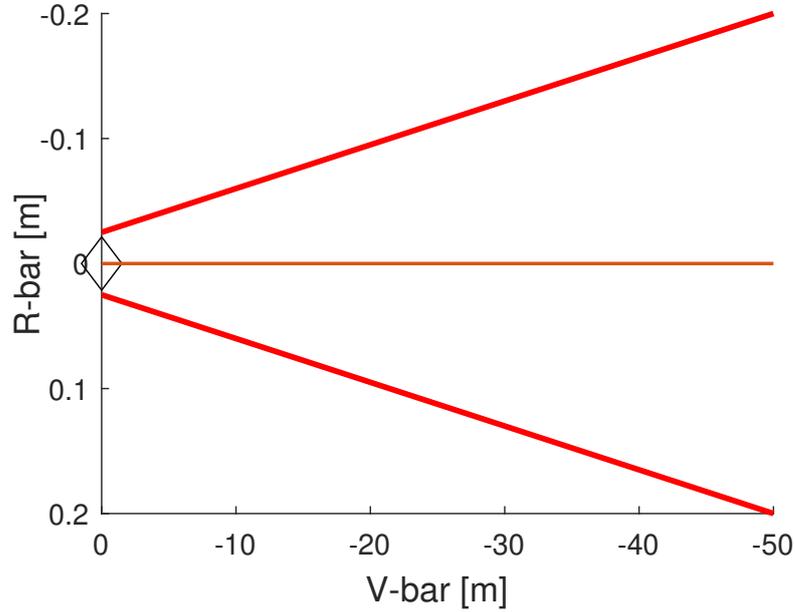


Figure 5.21: Straight line V-bar approach reference

Figure 5.21 shows the reference of the straight line V-bar approach maneuver with the cone of approach. The chaser must go from an initial condition $x_0 = [50, 0, 0]$ to a final condition $x_f = [0, 0, 0]$ which corresponds to the target position. In doing so, the satellite must lay within the cone of approach depicted in figure. Regarding the attitude regulation, similarly to the previous part of the mission scenario, the system is assumed to start from a random orientation with respect to the target and it must align to it. In other words, the attitude of the body frame of the chaser must be coincident with the attitude of the LVLH frame of the target. The chaser initial attitude is $q_0 = [0.688, 0.577, -0.413, -0.145]$ and $q_f = [1, 0, 0, 0]$. Once again, the system is simulated by adopting the three CA techniques analyzed in Chapter 4: the pseudo-inverse, the static linear programming assuming and the dynamic linear programming. The simulation result are reported below.

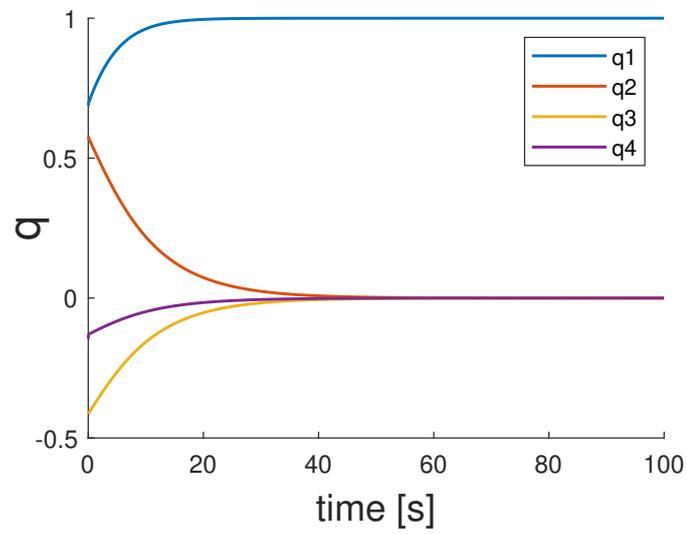


Figure 5.22: Attitude regulation

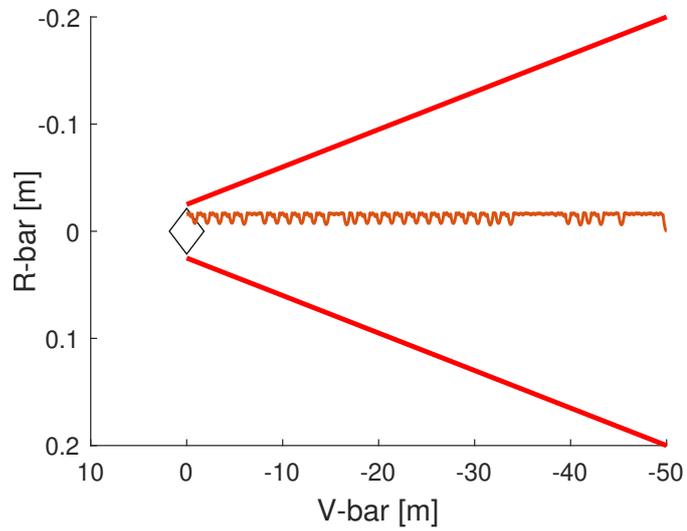


Figure 5.23: Straight line V-bar approach with pseudo-inverse CA

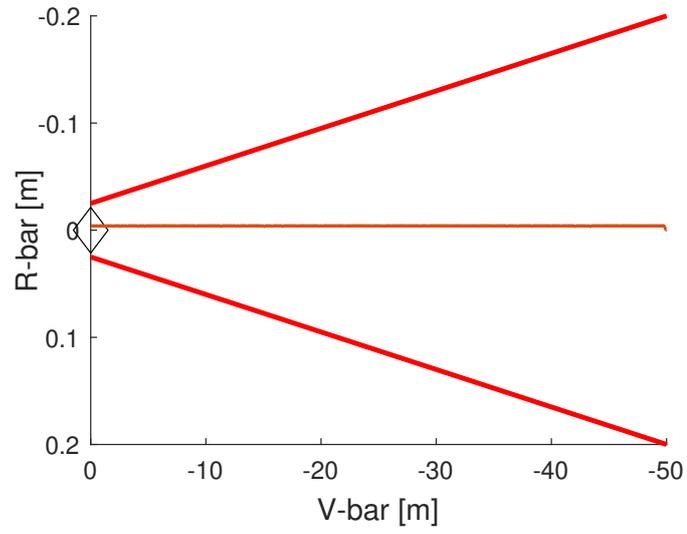


Figure 5.24: Straight line V-bar approach with linear programming CA

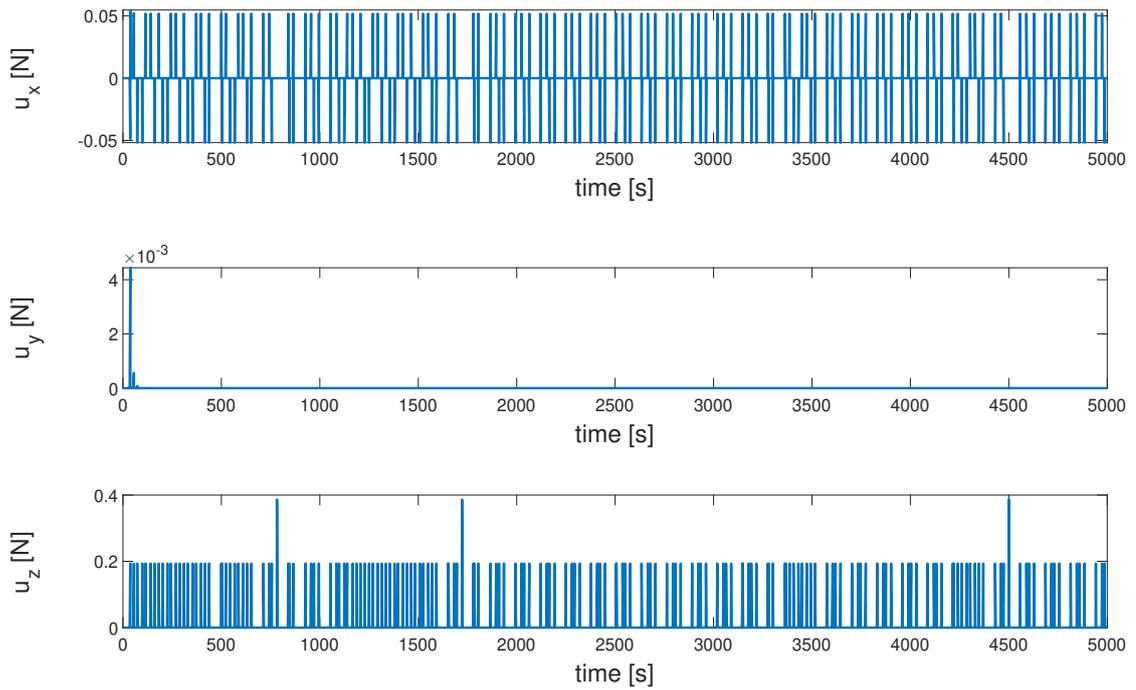


Figure 5.25: V-bar approach control input using pseudo-inverse CA

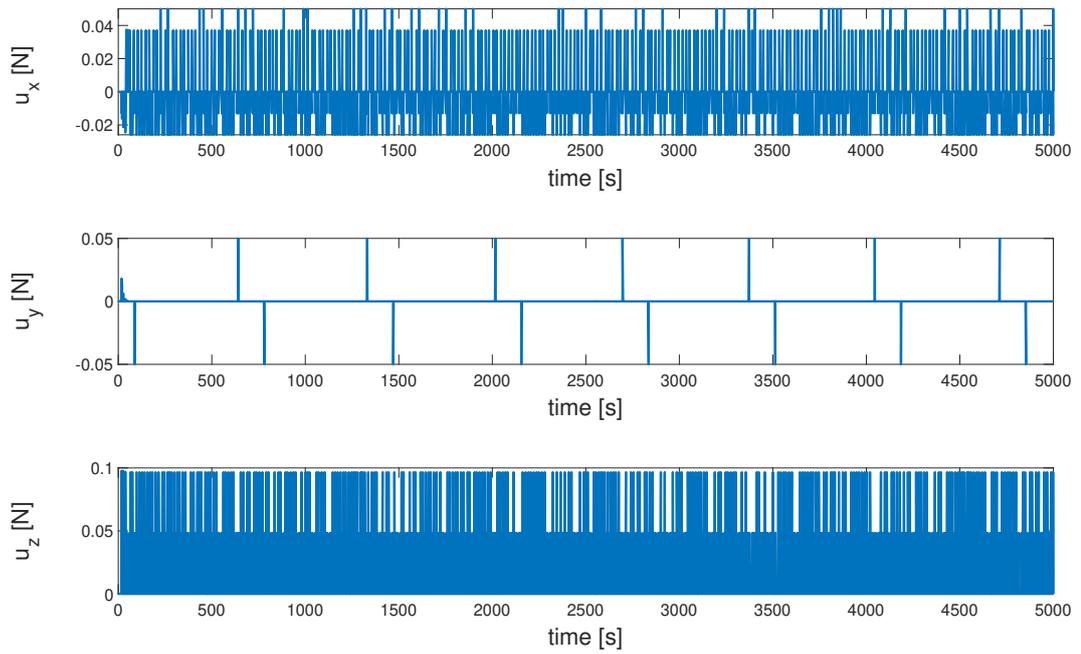


Figure 5.26: V-bar approach control input using static linear programming CA

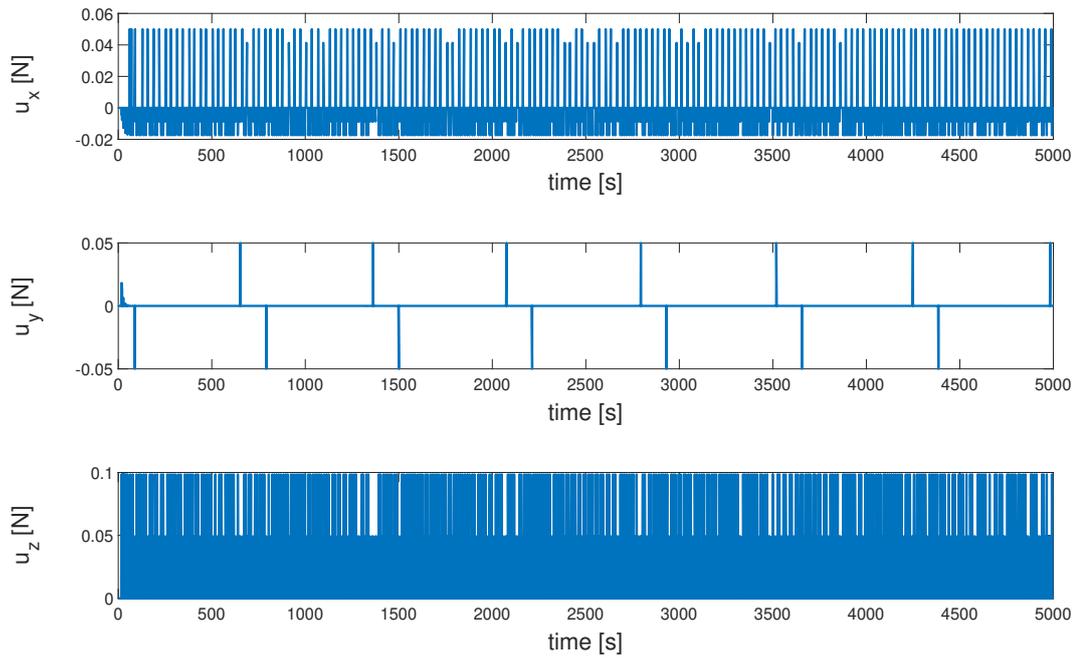


Figure 5.27: V-bar approach control input using dynamic linear programming CA

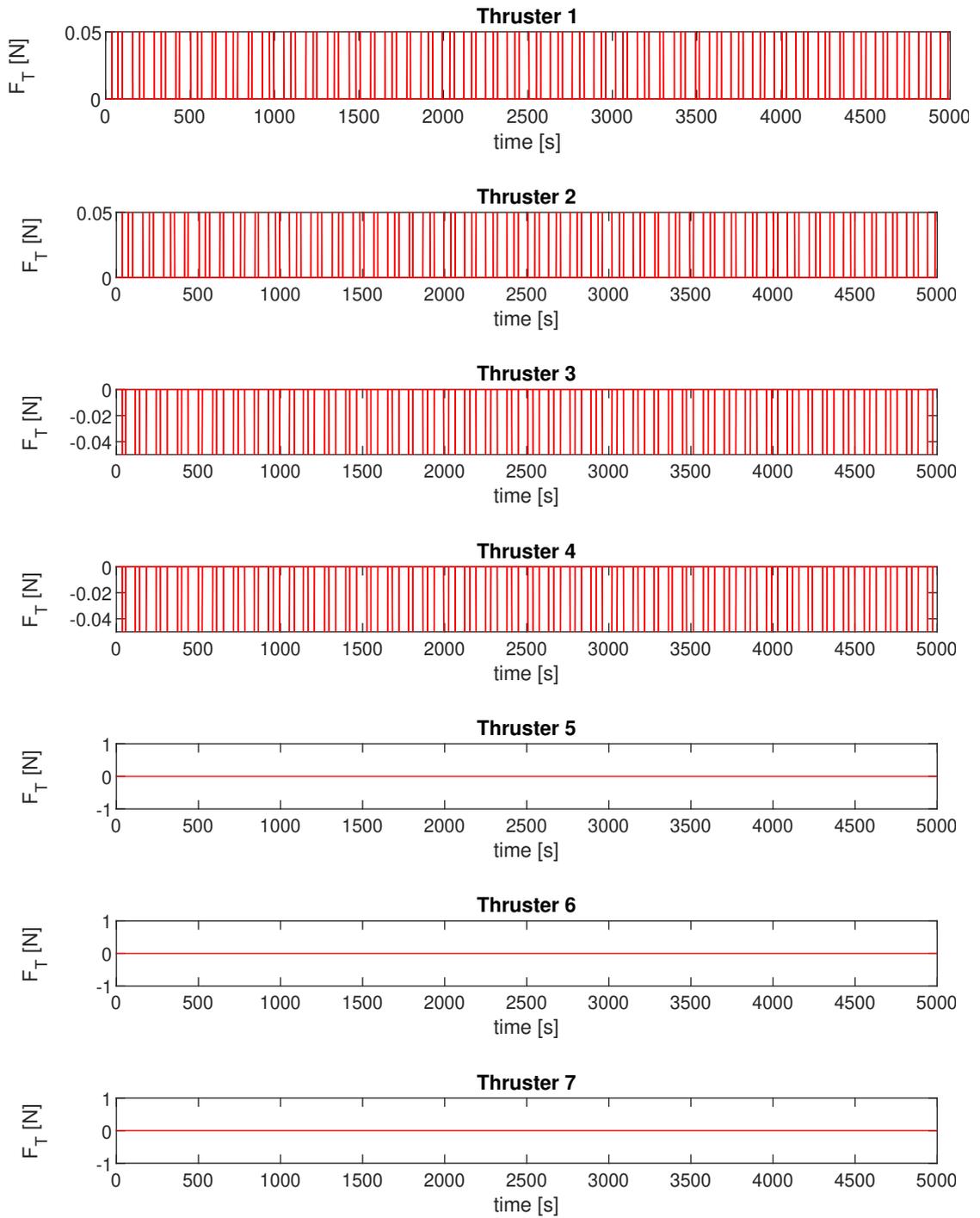


Figure 5.28: Pseudo-inverse thrust force

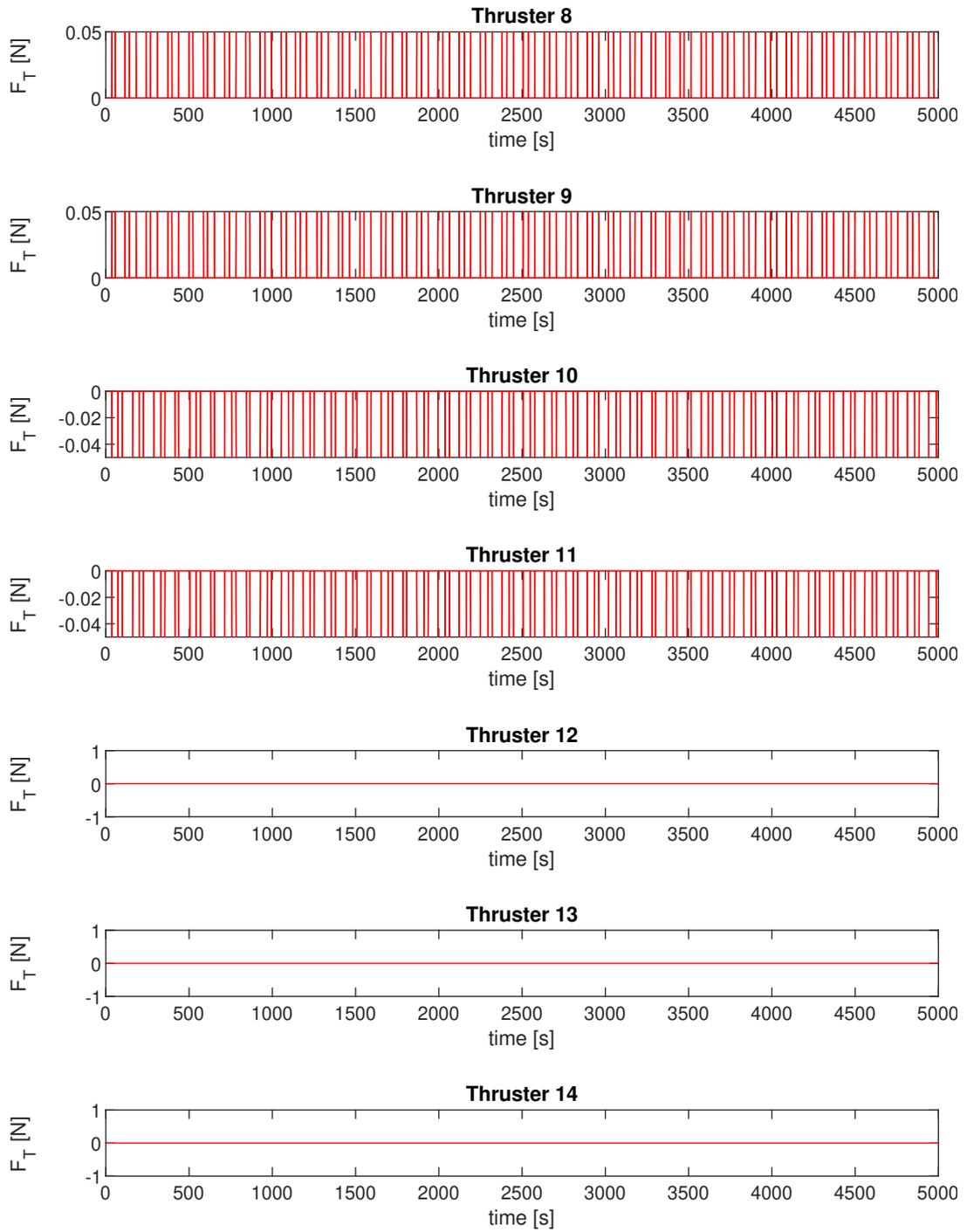


Figure 5.29: Pseudo-inverse thrust force

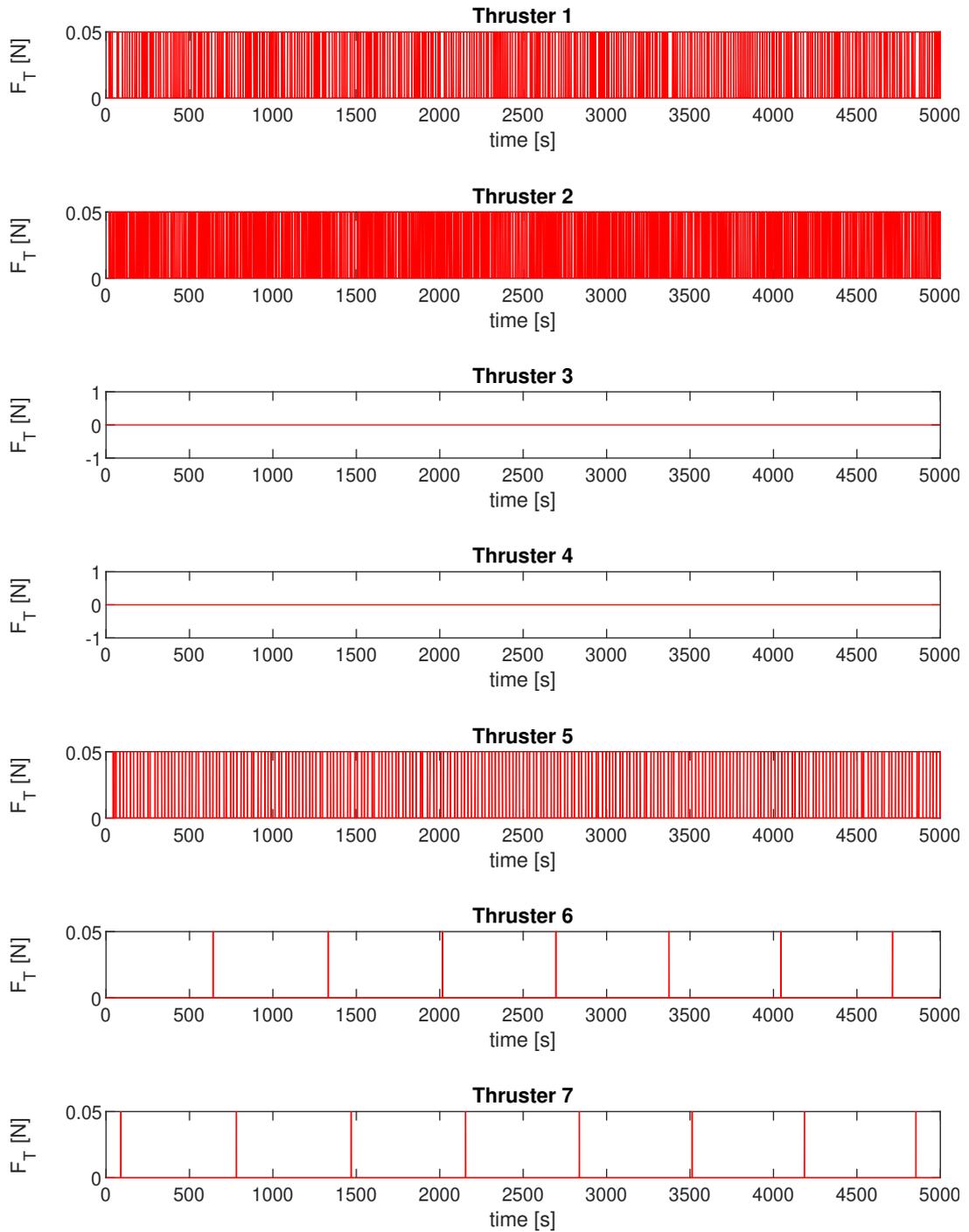


Figure 5.30: Static linear programming thrust force

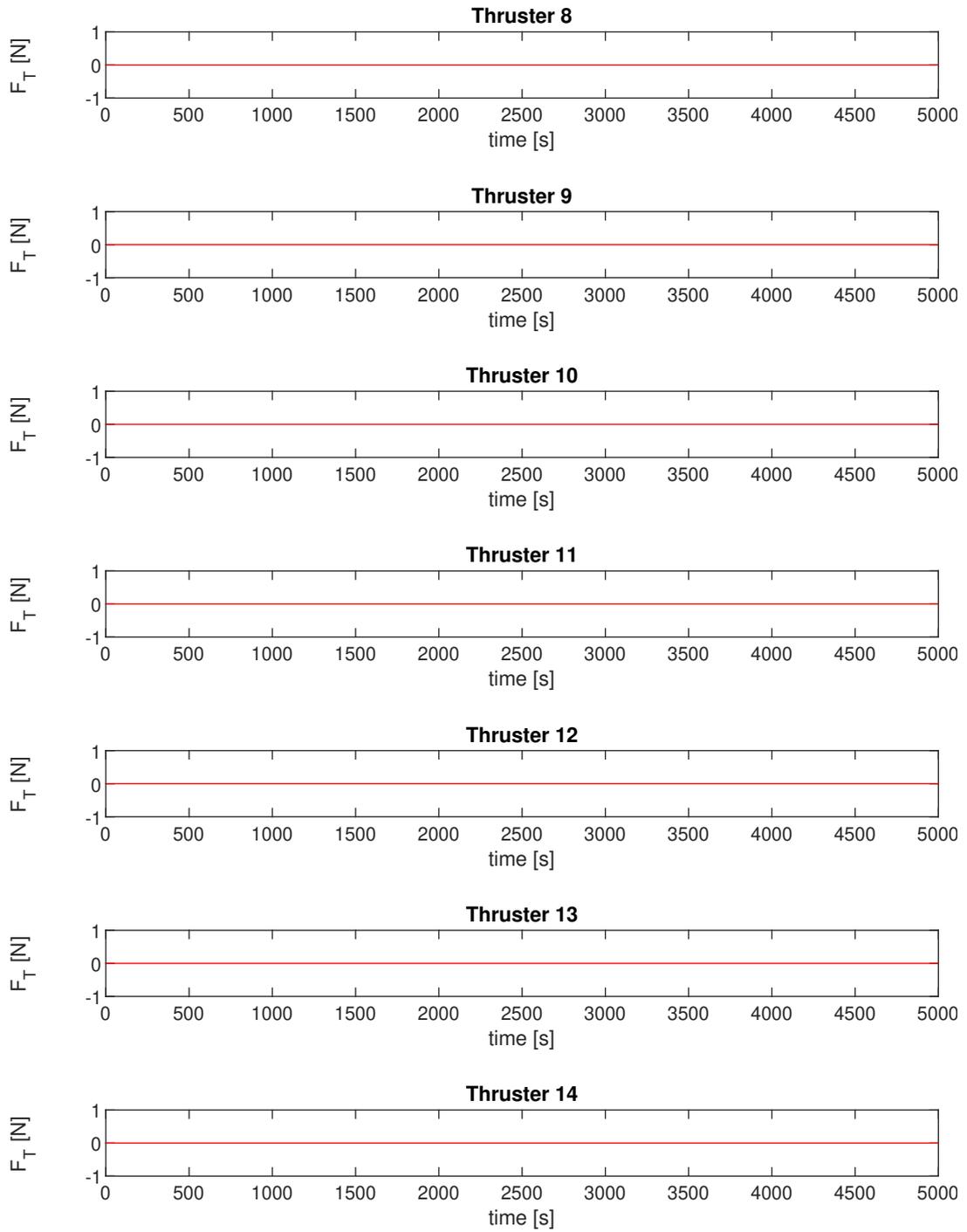


Figure 5.31: Static linear programming thrust force

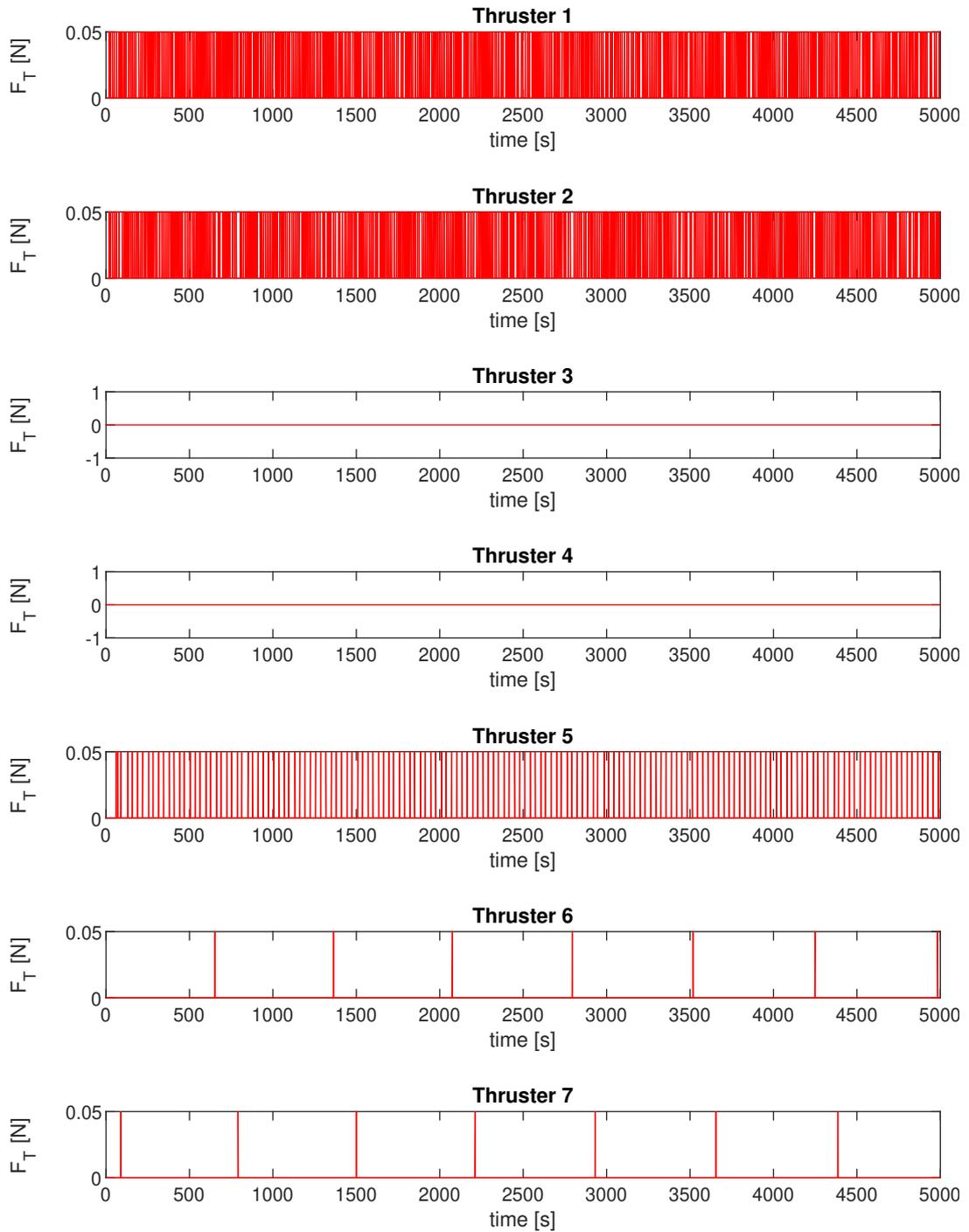


Figure 5.32: Dynamic linear programming thrust force

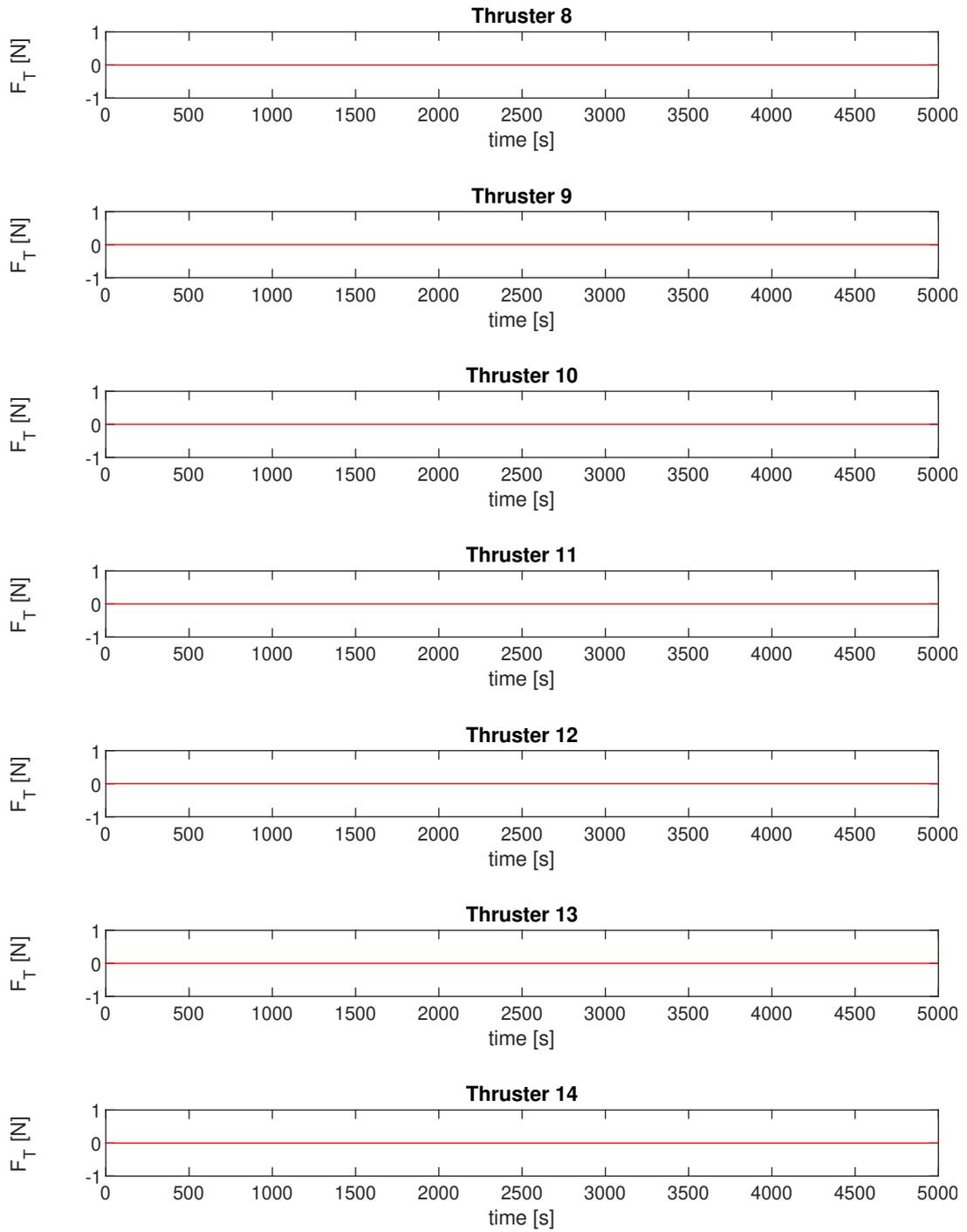


Figure 5.33: Dynamic linear programming thrust force

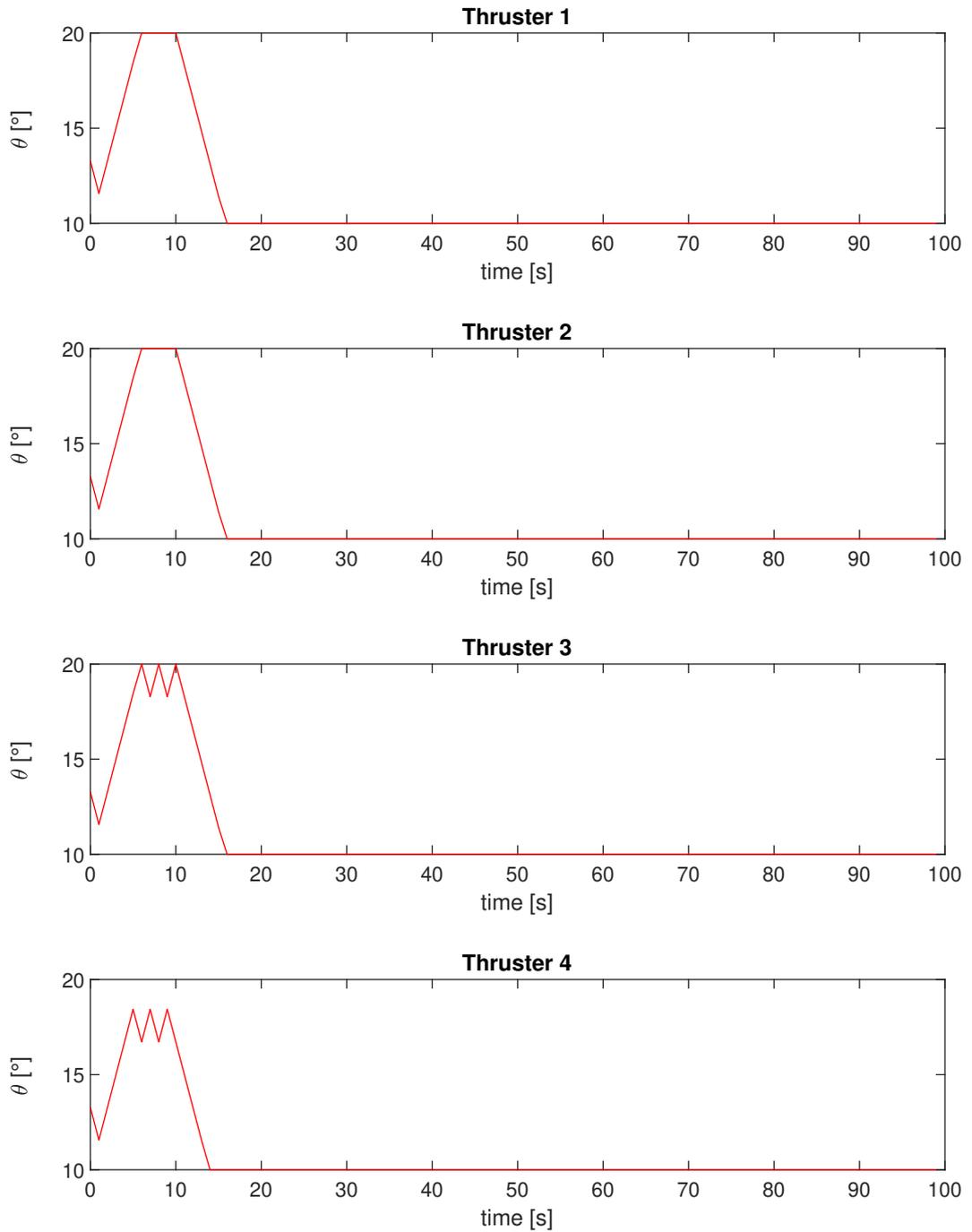


Figure 5.34: Thruster deflection angles

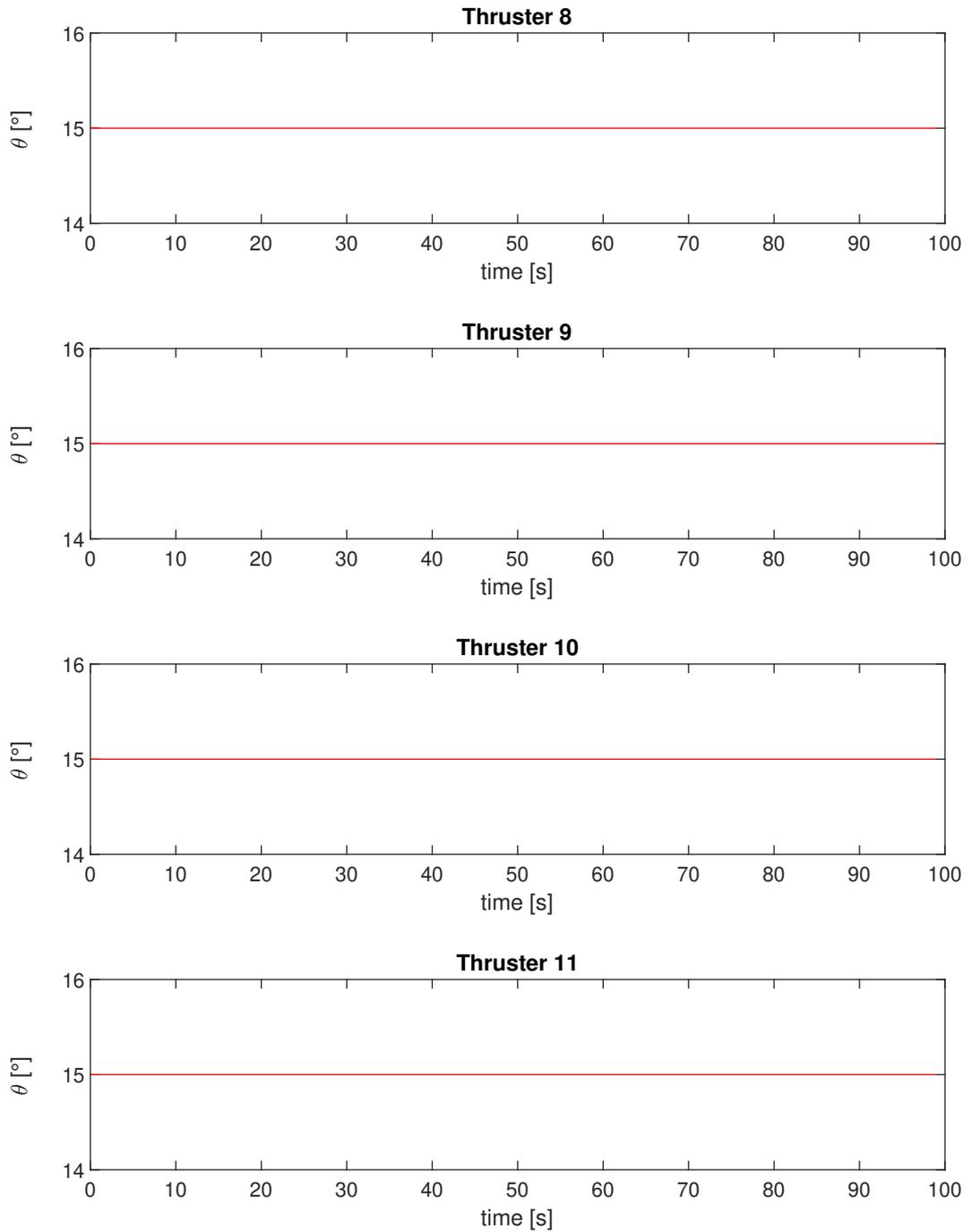


Figure 5.35: Thruster deflection angles

Maneuver	Minimum ΔV
Straight Line V-bar Approach	0.133 [m/s]

Table 5.3: Minimum ΔV required

Straight line V-bar approach maneuver	
6U Cubesat	
Control Allocation	ΔV [m/s]
pseudo-inverse	0.143
static linear programming	0.163
dynamic linear programming	0.153

Table 5.4: Straight line V-bar approach maneuver

Figure 5.22 presents the attitude regulation carried out by the 6U Cubesat utilizing all three previously mentioned CA techniques. Once again, only one figure is reported due to the similar performance of the CA methods in achieving the desired attitude. As shown in Figure 5.22, the Cubesat regulate its attitude within 50 seconds. However, regarding the positional tracking of the straight line V-bar approach trajectory, different performance arises. Figure 5.23 and 5.24 shows respectively the positional tracking of the Cubesat adopting pseudo-inverse CA and linear programming CA. Although the Cubesat manages to remain within the cone of approach adopting all three CA strategies, the pseudo-inverse CA exhibits lower performance compared to the linear programming CA methods. Figure 5.28, 5.29, 5.30, 5.31, 5.32 and 5.33 shows respectively the thrust force v generated by the two set of thrusters (thrusters 1, 2, 3, 4, 5, 6 and 7 and thrusters 8, 9, 10, 11, 12, 13 and 14) adopting the three CA techniques. The pseudo-inverse CA exploits thrusters 1, 2, 3 and 4 and thrusters 8, 9, 10 and 11. Conversely, the static and dynamic CA mainly utilize thrusters 1,2 and 5. Figure 5.25, 5.26 and 5.27 illustrates the control input for each CA techniques. Furthermore, Figure 5.34 and 5.35 illustrate the variation in thruster deflection angles when adopting the dynamic linear programming control strategy. It should be noted that only the graphs for thrusters 1 and 2 hold significance, as the other thrusters are not utilized. Within a 20 second, thrusters 1 and 2 progressively achieve their most vertical orientation in relation to the main body of the Cubesat. This particular configuration allows for enhanced thrust force generation in the radial direction. When examining Table 5.3 and 5.4, it can be observed that the pseudo-inverse CA method yields the lowest ΔV value, followed by the dynamic linear programming CA approach. However, the pseudo-inverse CA technique assigns negative thrust force values to thrusters 3, 4, 10, and 11, making it unfeasible for this specific

maneuver. Considering both its poor tracking performance and the inability to account for physical constraints of the thrusters, the dynamic linear programming CA method emerges as the most effective CA approach.

Chapter 6

Conclusion

This concluding chapter serves to summarize the research conducted in this master thesis project, outlining the workflow followed throughout. The objective of this thesis was to design and evaluate control allocation techniques for optimizing fuel consumption in miniaturized satellites, specifically focusing on Cubesats. In Chapter 1, an introduction was provided on Cubesats, including relevant historical background, and the concept of control allocation, which forms the core of this thesis project, was introduced. Chapter 2 and Chapter 3 presented the contextual framework in which control allocation is implemented. Chapter 2 delved into the fundamental concepts related to modeling the Cubesat system, covering translational and rotational dynamics, kinematics, and a comprehensive analysis of the Cubesat's actuators, namely the thrusters. Chapter 3 introduced the control strategies employed in the Cubesat, specifically the Linear Quadratic Regulator for position control and a proportional-derivative attitude controller for orientation control. Chapter 4, the central chapter of this thesis, focused on the development of control allocation strategies. Three techniques were presented, namely the pseudo inverse control allocation with fixed thrusters and the mixed optimization problem with fixed and moving thrusters. The mathematical formulation of these control allocation techniques was discussed in detail. Finally, Chapter 5 presented the mission scenario where the Cubesat underwent testing by performing maneuvers utilizing the various control allocation techniques developed in Chapter 4. Additionally, a comprehensive comparison of the control allocation strategies was conducted, considering the performance of the maneuvers, fuel consumption, and the feasibility of implementing the control allocation techniques in real-world scenarios.

6.1 Final Considerations

In the examination of various control allocation techniques, the mixed optimization approach with moving thrusters stands out as the most promising option. While the pseudo-inverse allocation technique appears favorable due to its lightweight nature and ease of implementation, its inability to account for constraints renders it unsuitable for most aerospace applications. Conversely, the mixed optimization approach, whether with fixed or moving thrusters, offer feasible solutions. The linear programming structure of the mixed optimization problem is exploited to achieve a complex yet efficient control allocation strategy. An additional benefit of this approach is its adaptability to multiple objectives chosen according to the mission requirements. Apart from the optimization of the fuel consumption fuel consumption, it is possible to incorporate additional functionalities, such as utilizing a designated subset of thrusters. Among the control allocation techniques, the mixed optimization approach with moving thrusters offers the best overall performances in terms of feasibility, maneuver tracking and fuel consumption

6.2 Future Work

Despite the superior performance of the mixed optimization approach with moving thrusters, it is crucial to deeply asses the feasibility of implementing such a solution. The utilization of gimballed thrusters, which rely on a thruster deflection mechanism, introduces several considerations. Specifically, it is necessary to investigate the potential impact on the mechanical design caused by the incorporation of these gimballed thrusters. Additionally, since the gimbals are actuated by piezo-electric motors, it is crucial to quantify the energy consumption of these piezo-electric motors and figure out whether the fuel savings justify their use. In conclusion, a thorough analysis of these factors is essential to evaluate the practicality and viability of implementing this peculiar control allocation strategy.

Bibliography

- [1] *Cubesat*. URL: <https://en.wikipedia.org/wiki/CubeSat> (visited on 07/05/2023) (cit. on p. 2).
- [2] *Technology Cubesats*. URL: https://www.esa.int/Enabling_Support/Space_Engineering_Technology/Technology_CubeSats (visited on 07/05/2022) (cit. on p. 6).
- [3] *In-Space Propulsion*. URL: https://www.nasa.gov/smallsat-institute/sst-soa/in-space-propulsion#_Toc114987897 (visited on 07/05/2022) (cit. on p. 7).
- [4] *Gimbal for Steering Propelled Cubesats*. URL: <https://technology.nasa.gov/patent/MFS-TOPS-74> (visited on 07/05/2022) (cit. on pp. 10, 11).
- [5] Wigbert Fehse. *Automated rendezvous and docking of spacecraft*. Vol. 16. Cambridge university press, 2003 (cit. on pp. 11, 12, 36, 50).
- [6] Carlo Novara. «Non-linear control and aerospace applications». In: *Academic material related to the Master's Degree in Mechatronic Engineering course* () (cit. on pp. 14, 15, 21).
- [7] Massimo Canale. «Digital control technologies and architectures». In: *Academic material related to the Master's Degree in Mechatronic Engineering course* () (cit. on p. 19).
- [8] *Reaction wheel for satellite and Cubesat*. URL: <https://eu.aspina-group.com/en/learning-zone/columns/2023022201/> (visited on 07/05/2022) (cit. on p. 20).
- [9] Tor A Johansen and Thor I Fossen. «Control allocation—A survey». In: *Automatica* 49.5 (2013), pp. 1087–1103 (cit. on p. 23).
- [10] Michael W. Oppenheimer, David B. Doman, and Michael A. Bolender. «Control Allocation for Over-actuated Systems». In: *2006 14th Mediterranean Conference on Control and Automation*. 2006, pp. 1–6. DOI: 10.1109/MED.2006.328750 (cit. on p. 24).

BIBLIOGRAPHY

- [11] Giuseppe Calafiore. «Convex optimization and engineering applications». In: *Academic material related to the Master's Degree in Mechatronic Engineering course* () (cit. on p. 25).
- [12] T Krovel. «Optimal tuning of PWPF modulator for attitude control». PhD thesis. 2005 (cit. on p. 33).