# UNDERACTUATED ATTITUDE CONTROL OF A CUBESAT USING COLD GAS THRUSTERS AND NONLINEAR CONTROL METHODS

THESIS

Adam S. Cottrell, Captain, USSF

AFIT-ENY-MS-22-M-285

**DEPARTMENT OF THE AIR FORCE**
**AIR UNIVERSITY**

***AIR FORCE INSTITUTE OF TECHNOLOGY***

**Wright-Patterson Air Force Base, Ohio**

DISTRIBUTION STATEMENT A. APPROVED FOR PUBLIC RELEASE;

DISTRIBUTION IS UNLIMITED

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, the United States Department of Defense or the United States Government. This is an academic work and should not be used to imply or infer actual mission capability or limitations.

AFIT-ENY-MS-22-M-285

# UNDERACTUATED ATTITUDE CONTROL OF A CUBESAT USING COLD GAS THRUSTERS AND NONLINEAR CONTROL METHODS

THESIS

Presented to the Faculty

Department of Aeronautics and Astronautics

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Astronautical Engineering

Adam S. Cottrell, B.S.

Captain, USSF

24 March 2022

DISTRIBUTION STATEMENT A. APPROVED FOR PUBLIC RELEASE;

DISTRIBUTION IS UNLIMITED

AFIT-ENY-MS-22-M-285

# UNDERACTUATED ATTITUDE CONTROL OF A CUBESAT USING COLD GAS THRUSTERS AND NONLINEAR CONTROL METHODS

Adam S. Cottrell, B.S.
Captain, USSF

Approved:

| | |
|---|---|
| Maj Robert Bettinger, PhD (Chairman) | Date |
| Andrew Keys, PhD (Member) | Date |
| Maj Costantinos Zagaris, PhD (Member) | Date |

# Abstract

Cold-gas thrusters on small satellites, such as CubeSats, are typically used for attitude control. However, to become more agile, small CubeSats must also look to propulsion systems utilizing non-combustion thrust methods, such as cold-gas, for translational maneuvers. The combined thrust vector is often misaligned with the system's center of mass resulting in a disturbance torque. This must be counteracted by either an attitude determination and control system (ADCS), additional thrusters, or a control method to keep the satellite's attitude at or near equilibrium. Nonlinearities generated by the instantaneous thrust maneuvers are overcome via control techniques explored in this research to include on-off control, sliding mode control, and model reference adaptive control (MRAC). These methods are then compared to a baseline test without thruster modulation, where the reaction wheels must desaturate prior to continuing the maneuver. For a 1.5 m/s delta-v maneuver, the nonlinear control techniques completed the maneuver nearly 100 times faster than the baseline, while improving pointing accuracy throughout the burn by up to 5%.

# Acknowledgements

First and foremost, I would like to thank my advisor, Maj Bettinger, for your guidance, patience, and expertise throughout my time at AFIT. Challenges, both anticipated and not, would have been far more difficult to overcome without your leadership.

To my committee members, Dr. Keys and Maj Zagaris, thank you for both your guidance and feedback on this research. Additionally, thank you for the knowledge and enthusiasm provided in your classes! I would not have been able to write this thesis without it, and these lessons will prove to be invaluable references when needed in my career.

Without a doubt, I would not have been able to get through my coursework here without the help of all of my classmates. There are too many to name, but thank you to every one of you for every late night study session, group projects, and team meetings. I look forward to our careers crossing paths in the future, and seeing the great things you all will accomplish!

Last but certainly not least, to my girlfriend, Nora. Your endless support, patience, love, and kindness have helped me through this program more than you'll ever know. From the bottom of my heart, thank you.

Adam S. Cottrell

# Table of Contents

Page

Abstract . . . . . iv

Acknowledgements . . . . . v

List of Figures . . . . . viii

List of Tables . . . . . xii

List of Symbols . . . . . xiii

I. Introduction . . . . . 1

1.1 Background Motivation . . . . . 1
1.2 Research Objectives . . . . . 2
1.2.1 Thesis Prospectus . . . . . 2
1.2.2 Research Questions . . . . . 2
1.2.3 Research Objectives . . . . . 3
1.3 Methodology . . . . . 3
1.3.1 Spacecraft Rigid Body Motion . . . . . 3
1.3.2 Active Attitude Control . . . . . 4
1.3.3 Control Methods . . . . . 4
1.4 Thesis Overview . . . . . 5

II. Literature Review . . . . . 6

2.1 Chapter Overview . . . . . 6
2.2 Commercial-Off-the-Shelf CubeSat Components . . . . . 6
2.3 Thruster Selection Logic . . . . . 8
2.4 Underactuated Spacecraft Attitude Control . . . . . 9
2.5 Summary . . . . . 11

III. Methodology . . . . . 12

3.1 Chapter Overview . . . . . 12
3.2 Spacecraft Rigid Body Motion . . . . . 15
3.2.1 Rotational Kinematics . . . . . 15
3.2.2 Rotational Dynamics . . . . . 18
3.3 Active Attitude Control . . . . . 20
3.3.1 Momentum Exchange Devices . . . . . 20
3.3.2 Thruster Torques . . . . . 22
3.3.3 Stability . . . . . 25
3.4 Control Methods . . . . . 29
3.4.1 Baseline Control . . . . . 29

Page

3.4.2 On-Off Control . . . 31
3.4.3 Quaternion Feedback Control . . . 34
3.4.4 Sliding Mode Control . . . 38
3.4.5 Model Reference Adaptive Control . . . 43
3.5 Summary . . . 50

IV. Results and Analysis . . . 51

4.1 Overview . . . 51
4.2 Case Study: Deorbit Maneuver . . . 52
4.2.1 On-Off Control . . . 53
4.2.2 Quaternion Feedback Control . . . 57
4.2.3 Sliding Mode Control ($\vec{\omega}$-Based) . . . 59
4.2.4 Sliding Mode Control (Quaternion-Based) . . . 64
4.2.5 Model Reference Adaptive Control . . . 68
4.3 Case Study: Phase Change Maneuver . . . 72
4.3.1 Baseline Control . . . 72
4.3.2 On-Off Control . . . 74
4.3.3 Quaternion Feedback Control . . . 80
4.3.4 Sliding Mode Control ($\vec{\omega}$-based) . . . 83
4.3.5 Sliding Mode Control (Quaternion-Based) . . . 86
4.3.6 Model Reference Adaptive Control . . . 92
4.4 Summary . . . 95

V. Conclusions and Recommendations . . . 96

5.1 Conclusions of Research . . . 96
5.2 Significance of Research . . . 99
5.2.1 Scholarly Presentations . . . 100
5.3 Recommendations for Future Work . . . 101
Bibliography . . . 103

# List of Figures


| Figure | | Page |
|---|---|---|
| 1 | ECI and Orbital Frames | 12 |
| 2 | Simulink Top Level Control Loop | 13 |
| 3 | Guidance Navigation and Control Flowchart | 14 |
| 4 | 6U CubeSat Body Frame | 16 |
| 5 | Simulink Spacecraft Plant | 20 |
| 6 | Simulink ADCS | 21 |
| 7 | Thruster Layout | 24 |
| 8 | Simulink Thruster Inputs | 25 |
| 9 | Plant Phase Portrait | 27 |
| 10 | Plant Phase Portrait with saturated reaction wheels | 28 |
| 11 | Simulink Baseline Controller | 30 |
| 12 | Simulink Sliding Mode Controller SMC-$\vec{\omega}$ | 41 |
| 13 | Simulink SMC-Q | 42 |
| 14 | MRAC Block Diagram | 43 |
| 15 | Simulink Model Reference Adaptive Controller | 46 |
| 16 | On-Off Reaction Wheels: Extended burn | 53 |
| 17 | On-Off Thrusters: Extended burn | 54 |
| 18 | On-Off Reaction Wheels (staggered offset): Extended burn | 55 |
| 19 | On-Off Thrusters (staggered offset): Extended burn | 55 |
| 20 | On-Off Phase Plane: Extended burn | 56 |
| 21 | QFC Reaction Wheels: Extended burn | 57 |

Figure Page

22 QFC Thrusters: Extended burn . . . . . 58

23 QFC Phase Plane: Extended burn . . . . . 59

24 SMC ($\vec{\omega}$-based) Reaction Wheels: Extended burn . . . . . 60

25 SMC ($\vec{\omega}$-based) Thrusters: Extended burn . . . . . 61

26 SMC ($\vec{\omega}$-based) Phase Plane: Extended burn . . . . . 62

27 SMC ($\vec{\omega}$-based) Sliding Surface (zoomed): Extended burn . . . . . 63

28 SMC ($\vec{\omega}$-based) Sliding Surface Derivative (zoomed): Extended burn . . . . . 63

29 SMC (Quaternion-based) Reaction Wheels: Extended burn . . . . . 65

30 SMC (Quaternion-based) Thrusters: Extended burn . . . . . 65

31 SMC (Quaternion-based) Phase Plane: Extended burn . . . . . 66

32 SMC (Quaternion-based) Sliding Surface (zoomed): Extended burn . . . . . 67

33 SMC (Quaternion-based) Sliding Surface Derivative (zoomed): Extended burn . . . . . 67

34 MRAC Reaction Wheels: Extended burn . . . . . 69

35 MRAC Thrusters: Extended burn . . . . . 69

36 MRAC Phase Plane: 1.5 m/s burn . . . . . 70

37 MRAC Adaptation Parameter (body mechanics - zoomed): Extended burn . . . . . 71

38 MRAC Adaptation Parameter (ADCS Control - zoomed): Extended burn . . . . . 71

39 Baseline Thrusters: 1.5 m/s burn . . . . . 73

40 Baseline Reaction Wheels: 1.5 m/s burn . . . . . 73

Figure Page

41 On-Off Thrusters: 1.5 m/s burn . . . . . 75

42 On-Off Reaction Wheels: 1.5 m/s burn . . . . . 75

43 On-Off Control Monte Carlo Analysis: Efficiency versus Maneuver Time . . . . . 77

44 On-Off Control Monte Carlo Analysis: Time and Efficiency Distributions . . . . . 77

45 On-Off Wheels: 1.5 m/s burn (secondary mode) . . . . . 78

46 On-Off Control Monte Carlo Analysis: Time and Efficiency Distributions (lower distribution) . . . . . 79

47 On-Off Control Monte Carlo Analysis: Time and Efficiency Distributions (higher distribution) . . . . . 79

48 QFC Thrusters: 1.5 m/s burn . . . . . 81

49 QFC Reaction Wheels: 1.5 m/s burn . . . . . 81

50 QFC Monte Carlo Analysis: Efficiency versus Maneuver Time . . . . . 82

51 QFC Monte Carlo Analysis: Time and Efficiency Distributions . . . . . 82

52 SMC ($\vec{\omega}$-based) Thrusters: 1.5 m/s burn . . . . . 84

53 SMC ($\vec{\omega}$-based) Reaction Wheels: 1.5 m/s burn . . . . . 84

54 SMC-$\vec{\omega}$ Monte Carlo Analysis: Efficiency versus Maneuver Time . . . . . 85

55 SMC-$\vec{\omega}$ Monte Carlo Analysis: Time and Efficiency Distributions . . . . . 85

56 SMC (Quaternion-based) Thrusters: 1.5 m/s burn (mode 1) . . . . . 87

57 SMC (Quaternion-based) Reaction Wheels: 1.5 m/s burn (mode 1) . . . . . 87

Figure Page

58 SMC (Quaternion-based) Thrusters: 1.5 m/s burn (mode 2) . . . . . 88

59 SMC (Quaternion-based) Reaction Wheels: 1.5 m/s burn (mode 2) . . . . . 88

60 SMC-Q Monte Carlo Analysis: Efficiency versus Maneuver Time . . . . . 89

61 SMC-Q Monte Carlo Analysis: Time and Efficiency Distributions . . . . . 89

62 SMC-Q Monte Carlo Analysis: Time and Efficiency Distributions (lower distribution) . . . . . 90

63 SMC-Q Monte Carlo Analysis: Time and Efficiency Distributions (higher distribution)) . . . . . 91

64 MRAC Thrusters: 1.5 m/s burn . . . . . 92

65 MRAC Reaction Wheels: 1.5 m/s burn . . . . . 93

66 MRAC Monte Carlo Analysis: Efficiency versus Maneuver Time . . . . . 93

67 MRAC Monte Carlo Analysis: Time and Efficiency Distributions . . . . . 94

# List of Tables


| Table | | Page |
|---|---|---|
| 1 | CubeSat Physical Properties | 19 |
| 2 | BCT XACT-15 Properties and Settings [1] | 22 |
| 3 | Summary of Control Method Results | 97 |

# List of Symbols

**Roman**

| Symbol | Meaning |
|---|---|
| $\hat{a}$ | Estimated sliding mode plant function |
| $a_i$ | Model reference adaptive control dynamics function |
| $\hat{a}_i$ | Model reference adaptive control adaptation law dynamics |
| $\hat{b}$ | Estimated sliding mode attitude determination and control system function |
| $b(x)$ | Generic input function of state vector |
| $c$ | Model reference adaptive control input coefficient |
| $C$ | Quaternion feedback control gain |
| $COM_i$ | Offset of the $i$th thruster (m) |
| $f(x)$ | Generic plant function of state vector |
| $\hat{f}$ | Sliding mode control estimated system dynamics |
| $F$ | Sliding mode control estimated error of state dynamics |
| $f_i(x)$ | Sliding mode and Model reference adaptive control functions |
| $\vec{h}$ | Model reference adaptive control highest order state variable coefficient |
| $\hat{h}$ | Model reference adaptive control adaptation law highest order state variable coefficient |
| $\vec{h}_A$ | Attitude determination and control system angular momentum |
| $\vec{H}$ | Spacecraft Angular momentum vector |
| $J$ | Moment of inertia matrix (kg-m$^2$) |
| $K$ | Quaternion feedback control gain |
| $K_s$ | Quaternion feedback control gain (thrusters) |

| | |
|---|---|
| $K_{SM}$ | Sliding mode sliding surface gain |
| $K_M$ | Model reference adaptive control sliding surface gain |
| $m$ | Mass of the satellite |
| $P$ | Sliding mode control gain matrix |
| $q$ | Quaternion unit vector |
| $\bar{q}$ | Entries 1:3 of the quaternion vector |
| $s$ | Sliding surface |
| $\bar{q}$ | First three entries of the quaternion |
| $u$ | Generic control input |
| $\hat{u}$ | Estimated control input |
| $V$ | Lyapunov function |
| $x_O, y_O, z_O$ | Orbital frame vectors |
| $x_B, y_B, z_B$ | Body frame vectors |
| $X, Y, Z$ | Inertial frame vectors |
| $Z_A$ | Attitude determination and control system reaction wheel distribution matrix |
| $Z_A$ | Attitude determination and control system distribution matrix |
| $z_B$ | Body frame z vector |

**Greek**

| **Symbol** | **Meaning** |
|---|---|
| $\Delta v$ | Change in space vehicle velocity |
| $\vec{\eta}$ | Positive constant gain |
| $\gamma$ | Model reference adaptive control adapation law gain |
| $\lambda$ | Nonlinear controller gain |
| $\Lambda_s$ | Sliding mode controller switching logic |

| | |
|---|---|
| $\vec{\omega}$ | Rotational velocity of the CubeSat (rad/sec) |
| $\vec{\omega}_{RW}$ | Rotational velocity of the reaction wheels |
| $\phi$ | Roll Euler angle ($^\circ$ or rad) |
| $\psi$ | Yaw Euler angle ($^\circ$ or rad) |
| $\tilde{\omega}$ | Rotational velocity tracking error (rad/s) |
| $\vec{\tau}$ | Torque (Nm) |
| $\vec{\tau}_{RW}$ | Attitude determination and control system torque (Nm) |
| $\tau_{T,i}$ | Torque from the $i$th thruster (Nm) |
| $\theta$ | Pitch Euler angle ($^\circ$ or rad) |
| $\vec{\Theta}$ | Space vehicle attitude ($^\circ$ or rad) |

# List of Acronyms

| Acronym | Meaning |
|---|---|
| ADCS | Attitude determination and control system |
| C&DH | Command and data handling unit |
| COTS | Commercial-off-the-shelf |
| COM | Center of mass |
| DCM | Direction Cosine Matrix |
| DoD | Department of Defense |
| DOF | Degrees of Freedom |
| EOM | Equations of Motion |
| GNC | Guidance, Navigation, and Control |
| KKT | Karush-Kuhn-Tucker |
| MED | Momentum Exchange Device |
| MiPS | Vacco Standard Micropropulsion System |
| MOI | Moment of Inertia |
| MRAC | Model Reference Adaptive Control |
| PnP | Plug-and-play |
| PWM | Pulse-width modulation |
| SMC | Sliding Mode Control |
| QFC | Quaternion Feedback Control |

UNDERACTUATED ATTITUDE CONTROL OF A CUBESAT USING COLD GAS THRUSTERS AND NONLINEAR CONTROL METHODS

# I. Introduction

## 1.1 Background Motivation

CubeSats offer a plethora of convenience and unique characteristics that enable entry level space missions. However, these benefits do not come without cost. While maintaining strict geometric and mass constraints, there are bound to be emergent issues correlated with pieces of the mission that require fine tuning of geometric and mass properties. One example of this is the propulsion system's thrust vector with respect to the satellite's center of mass (COM). The CubeSat architecture's constraints lead to a general difficulty in aligning the thrust vectors perfectly with the COM. This generates a disturbance torque that must be addressed through some means of momentum management. Other issues encountered may include, but are not limited to, the use of proprietary Attitude Determination and Control System (ADCS) firmware, resulting in unknown dynamics within the system via the ADCS control law. As well as system latency limitations that disrupt the feedback of the system, resulting in further state error. With an increasing demand for agile small satellites to conduct missions like rendezvous and proximity operations [2], these issues are becoming increasingly prevalent. Underactuated control methods of cold-gas thrusters may be utilized may be used to address attitude control and unknown system parameters, in an effort to minimize the number of actuators required by agile CubeSat missions.

## 1.2 Research Objectives

### 1.2.1 Thesis Prospectus

This research involves the stability analysis of utilizing cold-gas thrusters on small satellites to achieve relatively large $\Delta v$ maneuvers. In the use case exemplified throughout this research, a four-thruster propulsion array is integrated in a 6U CubeSat. The total thrust vector is misaligned with the space vehicle's COM due to low precision during assembly, and contingencies after launch, resulting in an external torque. Nonlinear control methods are implemented to modulate the thrusters to maintain closed-loop stability for the duration of the maneuver. Optimization of this problem will save valuable valuable fuel required to impart $\Delta v$ on orbit of spacecraft utilizing cold-gas thrusters, while exploring the feasibility of agile small satellites.

### 1.2.2 Research Questions

- What control methods may be applied to an underactuated system using cold-gas thrusters for translational maneuvers, while preventing ADCS saturation?
- What are the effects of instantaneous thrusters utilizing nonlinear control techniques on the closed loop stability of the spacecraft's attitude?
- Where are the limits to the physical parameters of the spacecraft and thrusters to maintain stability for each control method?
- How can the tunable control gains be systematically optimized, on-orbit, for thruster efficiency? Are there thresholds of these parameter values that cause the system to go unstable? Are there set values that force closed-loop stability?
- What are the benefits and drawbacks of each control method? What will drive specific control method selection during spacecraft development?

### 1.2.3 Research Objectives

- Develop a Simulink model, simulating the rigid body motion of a spacecraft in three-dimensional space.
- Within the model, insert a controller block, simulating the ADCS, including Quaternion Feedback Control logic, and a 3-Orthogonal reaction wheel array with saturation limits. Validate the system is locally asymptotically stable.
- Utilize nonlinear control techniques for the system, where the ADCS is assumed to be part of the space vehicle plant, to develop control methods for future cross-comparison of thruster control.
- Add a second control block to simulate the propulsion unit, and implement the control techniques in a closed-loop system simulating a four-thruster array.
- Implement and compare each control method to the propulsion block. Tune each control method to maximize thruster efficiency while verifying stability.

## 1.3 Methodology

### 1.3.1 Spacecraft Rigid Body Motion

Rigid body motion is a foundational concept for spacecraft attitude kinematics and dynamics. The assumption of a rigid body is made in order to establish the required equations to describe rotational motion. Kinematics gives a time-dependent representation between an object's body frame and an external reference frame [3]. In this case, the spacecraft's attitude error will be the offset of the body frame, to the orbital frame. Euler angle rotations are used to develop a direction cosine matrix (DCM) to describe this relationship. From there, quaternion kinematics may describe the spacecraft's attitude in an effort to parameterize the system without

singularities. The rotational dynamic equations of motion (EOM) are used to quantify and propagate the effect of external torques on the spacecraft's attitude, as governed by Newton's Second Law for Rotation [3].

### 1.3.2 Active Attitude Control

Satellite attitude control is accomplished by selecting appropriate actuators with governing control laws selected by referencing the kinematic and dynamic EOM. This may be accomplished by either passive or active means. The primary focus of this research is exploring the interaction between two attitude control methods: momentum exchange and moment-generating thrust. Through these methods torques are imparted on the CubeSat in an effort to complete a maneuver, while maintaining the correct attitude. The momentum exchange device in this model is a 3 reaction wheel orthogonal array ADCS, which is controlled specifically to maintain the correct pointing angle. The thrusters' primary use is to change the velocity of the space vehicle. However, the torques generated from the thrusters are larger and more sustained than what the ADCS is designed for. Methods to control the thrust vector must then be incorporated to maintain control of the spacecraft's attitude. Stability analysis is used to ensure that the system output remains desirable from a given input. This is especially important in satellite attitude control, as an unstable attitude control system would result in the spacecraft entering a tumble. A variety of methods may be used to ensure stability to include, but not limited to, phase plane analysis and Lyapunov's indirect and direct methods [4].

### 1.3.3 Control Methods

The control methods explored in this research include a Baseline Control method that does not include thruster modulation, and four nonlinear control techniques.

The Baseline Control method is named as such to give a basis to which to compare the other methods to. On-Off control is a popular technique due to its simplicity, and is especially applicable when using impulsive actuators, such as cold-gas thrusters. Quaternion feedback control provides a stable control law using quaternion parametrization, which is to be used in both the ADCS and as a thruster modulation control input. Sliding mode control is a robust control method that provides utility in systems with high uncertainty [4]. Model reference adaptive control is developed without requiring knowledge of specific parameters of the system by combining the known dynamics with variables that adapt throughout the duration of the simulation. Each of these methods are implemented using a variation of the control algorithm introduced in On-Off control.

## 1.4 Thesis Overview

This chapter has given a general outline of the content explored in this research. Chapter II dives further into the adjacent concept literature, and draws the required connections to establish an understanding of the problems to be addressed including CubeSat components and modern attitude control. In Chapter III, the equations of rigid body motion are derived and discussed with applicable assumptions. These equations are then followed up by their application in the active attitude control development. Each control method is then established along with algorithms for implementation. Chapter IV produces two case studies of spacecraft maneuvers, providing data for stability analysis, and actionable performance metrics. To finish, Chapter V compares the results found in Chapter IV, and recommendations are given for particular use cases of each control method. Recommendations for future research regarding the methods and model used are also presented.

# II. Literature Review

## 2.1 Chapter Overview

The purpose of this chapter is to examine associated areas of research of in underactuated attitude control of CubeSats. Plug-and-play CubeSat implementation is a fundamental aspect of the growing market for standardized, small research satellites. Sec. 2.2 provides a brief history of CubeSats while discussing the benefits and drawbacks of its associated third party market. Thruster-selection logic is an integral part to any thruster system. A variety of selection methods are discussed in Sec. 2.3, ranging from canned commands to optimization techniques. Underactuated attitude control has been a novel area of research for decades, as outlined in Sec 2.4. Its importance to this body of work lies in its complex and nuanced nature.

## 2.2 Commercial-Off-the-Shelf CubeSat Components

At the turn of the 21st century, the satellite industry was quickly growing but remained prohibitively expensive for most academic institutions. California Polytechnic State University, San Luis Obispo, and Stanford University's Space Systems Development Laboratory aimed to changed that dynamic with a proposed design standard for small satellites. This standard would eventually result in the form factor we know as a CubeSat [5]. The idea is simply that these small satellites must be of some dimension of exactly 10 centimeter increments in any direction, resulting in a rectangular prism that is no more than 1.33 kilograms. A 10 by 10 centimeter section is called a unit, or simply U, and the satellite size would be classified by how many units were contained by the satellite. For example, a satellite with 3 longitudinal units is known as a 3U CubeSat.

While the CubeSat movement kick-started mass university involvement in orbital enterprises and research, it was not without challenges. Developing electronics to reliably operate in the space is a difficult task, as seen by an approximately 60% reliability rate after 3 years of operation [6]. One solution to this issue is to follow a traditional test regiment. However, this would be in direct conflict of the initial goal of the CubeSat model: to provide an accessible satellite platform. On the other hand, if each component was reliably tested and verified by a manufacturer to meet the requirements for the space environment, the reliability would be expected to trend upwards [6]. Each commercial-off-the-shelf (COTS) component may then become a viable solution, similar to building a modern desktop computer.

Universities requiring a reliable subsystem is not the only application for COTS components. There is also a growing need for rapid deploy-ability or a "responsive space" mindset, particularly in the Department of Defense (DoD) [7]. This is where the concept of plug-and-play (PnP) technologies finds its market. In short, the idea is that COTS components are ready to be installed on a spacecraft out of the box, to include software interfaces, or at least be quickly configurable upon arrival. This type of technology is widespread in under industries, such as personal computers with the USB standard. Researchers have even suggested using the USB standard on spacecraft as a means of a PnP solution as far back as 2005 [8]. A well-defined standard as widespread as USB has yet to be determined for the satellite industry, but it will be required to maximize reliability of CubeSats.

As alluded to by the need for PnP solutions for particular mission areas, there are obstacles to overcome when utilizing COTS components. While the interface provided by the manufacturer in theory should provide what is advertised, there may be additional information relevant to the CubeSat owner that is not provided by the system's output. Additionally, there may be little to no information about how the

component itself operates due to proprietary restrictions on its contents [9]. As will be discussed in Ch. 3, combining systems that interact directly with one another via their physical responses drives the need to know how exactly the system is operating. Without knowing the control logic of an actuator causing a torque, an additional actuator with a controller designed to counteract that torque must be robust enough, have well placed assumptions, and/or have a significant amount of adaptability. While a PnP solution with full observability of the subsystem may be on the horizon, control systems within CubeSats may need to consider each component as a "black box" to overcome emergent properties.

## 2.3 Thruster Selection Logic

Thruster selection, or jet selection, logic has been in use as far back as the Apollo era, where thrusters would fire to enact strictly sign-dependent torques on the spacecraft [10]. Thruster tables were originally used by the Apollo missions to apply adequate torque about a particular axis to maintain proper attitude control. This required extensive knowledge of the system to include high order polynomial approximations of every known mode of the system at discrete temperature intervals from bending modes to fuel sloshing at discrete fuel level approximations[11]. Thruster selection was then optimized via rigorous testing considering each change in mode, optimized for minimal fuel loss, and stored to be used in flight.

Candidate optimal groups construct the same type of state transition matrix, only in real time as a linear-programming problem [3]. Solving the system of nonlinear equations each time step to produce thrust values may prove to be prohibitively CPU intensive for the spacecraft. For fully-actuated thruster combinations, a feedback-linearizable method may be used to linearize the system and develop a system of linear equations to be solved with relative ease in each time step. However, if the thruster

combination is not feedback-linearizable, but underdetermined, a optimization process may ensue using methods such as Karush-Kuhn-Tucker (KKT) conditions [12]. For non-feedback-linearizable, overdetermined systems there is no possible solution and a complex nonlinear optimization function must be used. This is typically too intensive to occur in real-time at every time step.

Using a psuedoinverse method, an authority matrix is generated to convert a simple torque request into thrust responses, similar to that of the candidate optimal group method. While lacking robustness, it provides a relatively simple calculation for modern spacecraft [13]. The authority matrix may be developed in a variety of ways, to include on-off control based on dual opposing thrusters, to more advanced methods resulting in dynamic matrix generation in real-time. The benefit to this method lies in the flexibility to adjust the control algorithms based on desired performance. In Jewison's work [13], a reconfigurable logic controller was presented with actuation modes for a Nominal mode, Precision mode, Fuel-Efficient mode, and and Agile mode. While this method does provide a computationally simple solution to thruster selection, the system must be at least fully-actuated [13] [14].

## 2.4 Underactuated Spacecraft Attitude Control

Underactuated control is often defined as a system where there are less actuators than degrees of freedom [15]. A more complete definition is that any arbitrary combined output of a system's actuators cannot be achieved by a combination of the inputs [16]. This has been a well-studied concept in aerospace engineering in an effort to reduce the number of components required, thus bringing mass and complexity down. The challenge with underactuated control in spacecraft is the lack of controllability about a particular axes, as reflected in Brockett's theorem [17]. However, as far back as 1988, there have been solutions such as Aeyels and Szafranski's

single actuator stability proof [15]. Although, these solutions are self admittedly non-robust. In 2000, Tsiotras and Luo provided a proof for an underactuated spacecraft control system with symmetry to be asymptotically stable, so long as there was no motion about the third principal axes [18]. This is all to say that to achieve asymptotic stability requires significant constraints in an underactuated spacecraft.

Another caveat to these control proofs is that the control logic assumes a time invariant system. This prevents time variant solutions from being applied. Time varying methods have been developed, to include the work done by Morin and Samson in 1997 [19]. Model predictive control is a time-variant control technique that has been used to prove local exponential stability in an underactuated spacecraft [20]. This approach allows control parameters to adequately adjust to the system's nonlinearities as time progresses.

For satellites with symmetric properties, such as CubeSats, these methods are largely inapplicable. However, cascading subsystems may be used to overcome the stabilization effort. Although, global asymptotic stability still eludes the otherwise underactuated spacecraft, as controllability of a single subsystem in the cascade is not sufficient[15]. However, recent studies have indicated local asymptotic stability of an underactuated spacecraft is possible when the initial angular momentum vector falls within the angular momentum boundaries of two control moment gyroscopes along principle axes [21]. In practice, local asymptotic stability provides a working solution for satellite missions. If the region of stability is escaped due to an unexpected external torque, the spacecraft must be commanded to enter a momentum management mode to return to a nominal state.

The limitations discussed in this section showcase the complexity and constraints placed on underactuated attitude control systems. Today, the primary application for such systems has been for single-axis inertial pointing spacecraft starting with minimal

rotational velocity and close to the desired pointing angle [22]. These limitations drive the need for further research in underactuated control. More-so, systems with cascading attitude control subsystems may be considered to reach local marginal stability at a minimum. Exponentially or asymptotically stable control methods have thus far been too constrained for the majority of operational applications.

## 2.5 Summary

CubeSat design has a wealth of benefits, but it is not without drawbacks. The use of COTS components has driven the development time down drastically, and has enabled researchers to pursue missions in a standardized fashion. However, the limitations of COTS components may lead to difficult trade-offs. When specifically discussing the use of cold-gas thruster systems, the selection logic used must be capable enough to meet $\Delta v$ or attitude stability requirements, while the system must be sophisticated and efficient enough to implement such logic. For overactuated systems, this has been proven to be a rudimentary endeavor for modern CPUs [13]. However, underactuated systems exhibit an entirely different tradespace. The pursuit for globally asymptotically stable underactuated spacecraft control has not been achieved without significant caveats to include lack of spacecraft symmetry and complex nonlinear control optimization calculations to occur in real-time. Both of these traits are prohibitive to current CubeSat limitations. The following research explores the intersection of these concepts, where instead of requiring exponential stability, the expectation of marginal stability is set. This will then provide a real use-case for prospective missions, with actual COTS product examples.

# III. Methodology

## 3.1 Chapter Overview

This chapter discusses the foundational principals used to develop a model for spacecraft attitude control. The equations of motion (EOM), developed from the kinematics and dynamics, are applied to accurately reflect how a rigid body behaves when external torques are exerted in the analysis. Additionally, these equations are examined in the stability analysis, with full observability of the physical system. Closed-loop control methods may be drawn from the kinematics to actively apply torques via the ADCS to achieve local asymptotic stability in pointing of the spacecraft. The thruster control methods discussed in Ch. 4 utilize this known stability of the plant. This is done by applying disturbance torques, via individual thrusters, to be counteracted by the ADCS, while propagating a duty cycle for a translational maneuver of the spacecraft. Each equation developed in this chapter is then implemented in a Simulink model for analysis. Figure 2 shows the top level model.

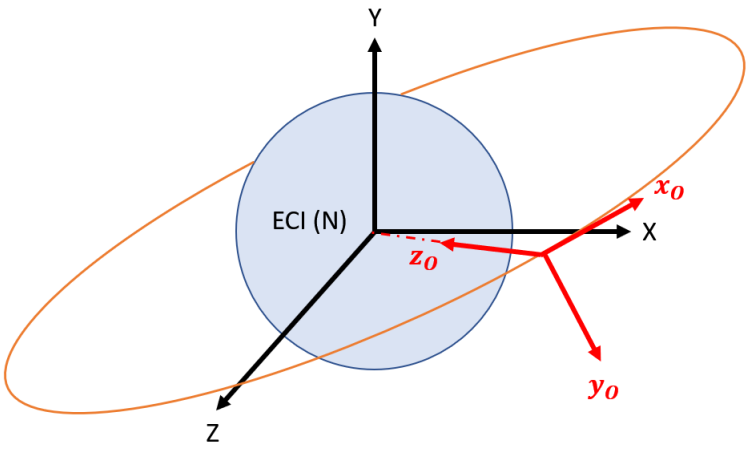


**Figure 1. ECI and Orbital Frames**

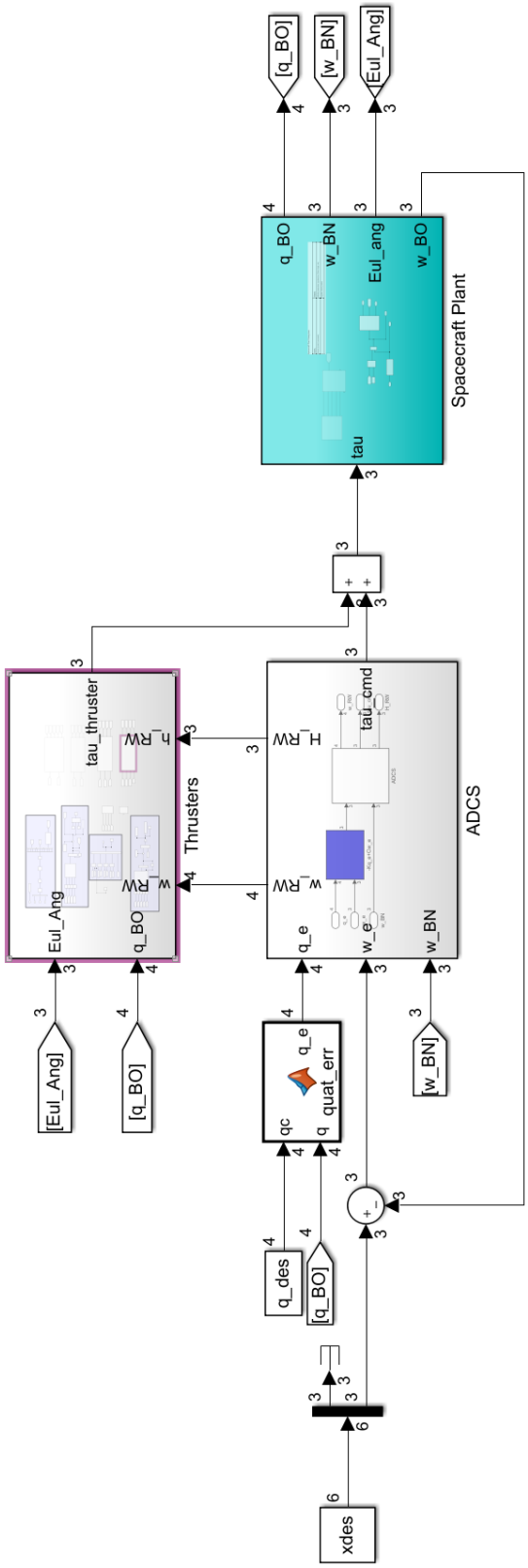


**Figure 2. Simulink Top Level Control Loop**

Figure 3 shows a simplified flowchart of GNC subsystem interactions, as considered in this model. The desired pointing angle and requested $\Delta v$ are treated as inputs from the ground, which are received by the C&DH. The C&DH then disseminates commands accordingly based on the feedback loop, in reference to these initial inputs. The ADCS receives pointing commands, but only returns state data regarding the reaction wheel speed, and attitude parameters. If the ADCS was not treated as a "black box" there could be additional commands send based on more detailed state feedback. In spite of this drawback, using only reaction wheel RPM, and attitude error, the C&DH runs a thruster selection logic in each time step to determine which of the four thrusters to fire. The propulsion system simply executes this command. Note that both the propulsion system and ADCS apply a torque to the dynamics. It is also important to clarify that the attitude error "detected" by the ADCS in this research does not undergo any estimation techniques to emulate the star tracker. The attitude error is treated as true data, and delayed within the Simulink model.

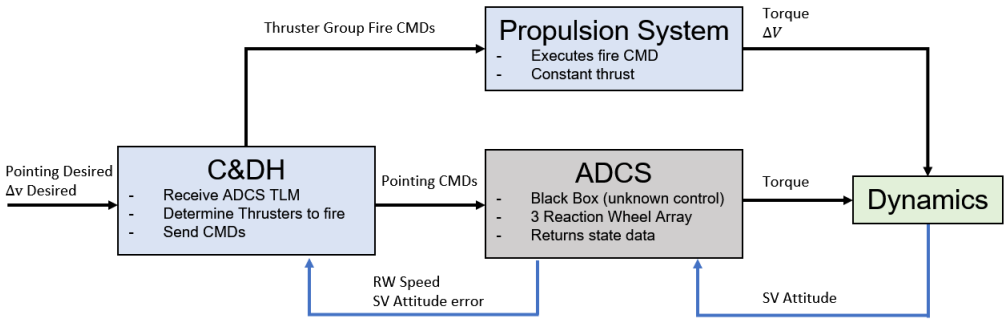


**Figure 3. Guidance Navigation and Control Flowchart**

## 3.2 Spacecraft Rigid Body Motion

### 3.2.1 Rotational Kinematics

Before determining how to control the dynamics of a spacecraft, establishing the kinematic equations is essential. Generally, rotational kinematics is referred to as the subject of rotational motion without including any forces. This allows for a distinct, time-dependent relationship between the spacecraft's fixed body frame, and the reference frame to which the spacecraft will be commanded to. There are multiple methods to extract these EOM, all with unique advantages and disadvantages. This work will generally work with both Euler Angles and quaternions. However, the quaternion kinematic equations will be utilized in the model.

Euler Angle rotations are arguably the easiest parameters to visualize when discussing satellite attitude. The concept essentially boils down to a Direction Cosine Matrix (DCM) developed from combining three rotation matrices in a particular order. The 3-2-1 Sequence refers to rotating about the yaw, $\psi$, pitch, $\theta$, and roll $\phi$, in sequence. This is shown below in Eqn. 3.1 where the Body frame is translated to the Orbital frame by the DCM [3]. Figure 4 shows a graphical view of the Body Frame. The Orbital frame shown in Fig. 1 is used as the desired reference frame for the body frame, as the case studies performed later in this research dictate an Earth-pointing attitude. For the majority of this work, attitude error will be synonymous with the Euler Angles, and may be used interchangeably. In Eqn. 3.1, $C_{BO}$ is defined as the DCM between the Body (B) and Orbital (O) frames. This notation will remain consistent throughout this work.

$$
\begin{aligned}
C_{BO} &= C_1(\phi)C_2(\theta)C_3(\psi) \\
&= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}
\end{aligned} \tag{3.1}
$$

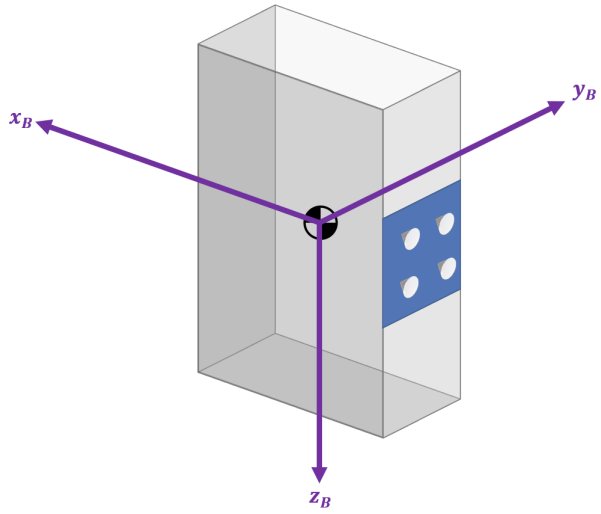


**Figure 4. 6U CubeSat Body Frame**

Quaternions are a more abstract parameter set than Euler Angles, but at a high level may be described as a rotational description of a 4-dimensional unit hypersphere. These parameters have become widely used in spacecraft attitude determination and control. The technical aspect that makes quaternions so attractive over Euler Angles, is their lack of singularities. Euler Angles are known for a particular singularity called "Gimbal Lock" where two axes align and cause an unpredictable rotation [23]. There are methods to overcome this, but utilizing quaternions avoids this issue altogether. Their relationship to the DCM is described in Eqn. 3.2, known as the scalar element of the unit quaternion, and Eqn. 3.3, often noted as the vector part [3].

$$q_4 = \pm\frac{1}{2}\sqrt{1 + C_{11} + C_{22} + C_{33}} \tag{3.2}$$

$$\bar{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} = \frac{1}{4q_4}\begin{bmatrix} C_{23} - C_{32} \\ C_{31} - C_{13} \\ C_{12} - C_{21} \end{bmatrix} \tag{3.3}$$

$$||q|| = q_1^2 + q_2^2 + q_3^2 + q_4^2 = \bar{q}^T\bar{q} + q_4^2 = 1 \tag{3.4}$$

$$q = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T$$

To describe the kinematics of a rigid body, a first order differential equation for each axis must be determined [3]. This is due to the spacecraft having 3 rotational degrees of freedom (DOF). The DCM Kinematic equations are a useful relation between the vehicle's angular velocity in the orbital frame, and the time rate of change of the DCM [3].

$$\begin{aligned} \omega_{BO_x} &= \dot{C}_{21}C_{31} + \dot{C}_{22}C_{32} + \dot{C}_{23}C_{33} \\ \omega_{BO_y} &= \dot{C}_{31}C_{11} + \dot{C}_{32}C_{12} + \dot{C}_{33}C_{13} \\ \omega_{BO_z} &= \dot{C}_{11}C_{21} + \dot{C}_{12}C_{22} + \dot{C}_{13}C_{23} \end{aligned} \tag{3.5}$$

With Eqns. 3.2, 3.3 and 3.5, the quaternion kinematic equations may be found [3]:

$$\dot{q} = \frac{1}{2}\begin{bmatrix} q_4 & -q_3 & q_2 & q_1 \\ q_3 & q_4 & -q_1 & q_2 \\ -q_2 & q_1 & q_4 & q_3 \\ -q_1 & -q_2 & -q_3 & q_4 \end{bmatrix}\begin{bmatrix} \omega_{BO_x} \\ \omega_{BO_y} \\ \omega_{BO_z} \\ 0 \end{bmatrix} \tag{3.6}$$

Thorough explanation of this derivation may be found in [24],[3],[25].

### 3.2.2 Rotational Dynamics

Kinematics is the study of motion without considering the causes. Dynamics, on the other hand, is entirely about the forces in the system. Rotational dynamics, specifically, are concerned with the torques acting upon an object [3]. As previously stated, rigid body motion is assumed for the purposes of this research. As will be discussed later, this is a valid assumption to make due to the nature of the control laws used in both the ADCS and propulsion unit.

The fundamental dynamics equation (Eqn. 3.7) briefly describes how rotational dynamics works at its core. If the angular momentum, $\vec{H}$, is not changing over time, there cannot be external torques, $\vec{\tau}$, acting upon the body [3].

$$\dot{\vec{H}} = \vec{\tau} \tag{3.7}$$

Angular momentum of the body may be described by Eqn. 3.8 [3]. Note that the angular momentum is determined by the inertial angular velocity. Angular momentum is conserved only in the inertial frame.

$$\vec{H}_{BN} = J\vec{\omega}_{BN} \tag{3.8}$$

The moment of inertia (MOI) matrix, $J$, is also an integral part to understanding the dynamics of the system. The $3 \times 3$ matrix represents the inertial properties of an object about each axis. In reality, this matrix is never a perfect diagonal on spacecraft, as it is very difficult to align the principal axes with the geometry perfectly. CubeSats have the benefit of being highly symmetrical as a standard, while also requiring their COM to be within 2 cm of its geometric center (stowed), in the $x_B$ and $y_B$ direction, while having up to 7 cm along the $z_B$ axis [5]. This drives the products of inertia to be relatively close to zero. While they are not negligible in precision applications,

it will be assumed in this paper that the 6U CubeSat, by default, has uniform mass distribution of a rectangular parallelepiped. The center of mass will therefore also be at its geometric center. As this research is mainly concerned with the stability of applying instantaneous torques, this is a valid assumption. Sensitivity analysis is performed in later chapters to expand upon this.

**Table 1. CubeSat Physical Properties**

| | |
|---|---|
| $m$ (kg) | 12 |
| $J_{xx}$ (kg-m$^2$) | 0.1 |
| $J_{yy}$ (kg-m$^2$) | 0.05 |
| $J_{zz}$ (kg-m$^2$) | 0.13 |
| $r_{COM}$ (m) | [0,0,0] |

Euler's Rotational EOM operates specifically in this design space, where the principal axes are aligned with the body frame. Similar to kinematics, there are three equations to describe the torque balance about each axis.

$$\begin{aligned} \tau_x &= J_{xx}\dot{\omega}_x + (J_{zz} - J_{yy})\omega_z\omega_y \\ \tau_y &= J_{yy}\dot{\omega}_y + (J_{xx} - J_{zz})\omega_x\omega_z \\ \tau_z &= J_{zz}\dot{\omega}_z + (J_{yy} - J_{xx})\omega_y\omega_x \end{aligned} \tag{3.9}$$

Between Eqns. 3.2, 3.3 and 3.9, the physics are adequately modeled. The torques to be considered in these calculations can range from aerodynamic, gravitational gradient, solar radiation pressure, magnetic, momentum exchange devices (MEDs), thrusters, and even debris impacts. This research only focuses on the behaviour of the ADCS, a reaction wheel momentum exchange device, and the cold-gas thrusters. Magnetorquers are also considered for the baseline control method, but are not the focus. Typically, magnetorquers are used to detumble the spacecraft, but they are

also useful to passively counteract the effects of the other disturbance torques, sans impacts, accruing momentum on the ADCS [26]. While accurate modeling of specific spacecraft magnetorquer capabilities and momentum management requirements is vital for space vehicle design, the COTS ADCS considered in this research have advertised this passive capability [1], and therefore environmental disturbance torques may be assumed null. The torques involved with establishing a stable balance between independent COTS ADCS and propulsion subsystems are much more significant. Figure 5 shows how the equations in this section are incorporated into the Simulink model.

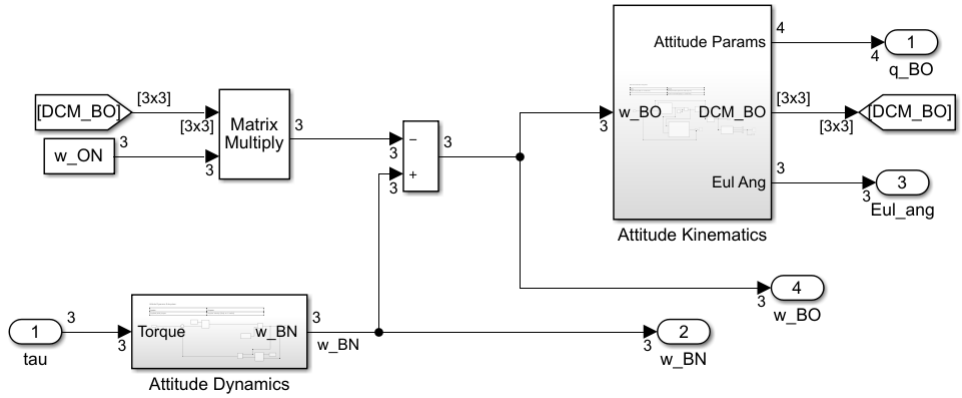


**Figure 5. Simulink Spacecraft Plant**

## 3.3 Active Attitude Control

### 3.3.1 Momentum Exchange Devices

Spacecraft attitude control may be conducted either passively or actively. Passive control traditionally consists of utilizing a three-axis stabilized gravity gradient or spin stabilization. Active methods yield a far more agile spacecraft with consistent pointing accuracy [26]. CubeSats ADCS's have rapidly increased in complexity, availability, variety, overall quality in recent history with reaction wheels being the most common devices [27]. While each COTS component may have proprietary

control logic on-board, the objective for this model was to use trusted and universally accepted methods to simulate a stable reaction wheel system.

The use of reaction wheels on CubeSats allows for some highly desirable advantages with minimal trade-offs. Unlike control moment gyroscopes, reaction wheel assemblies are capable of high precision pointing without risking singularities within their peak momentum bounds. The concept is simple: rotate a dense disk at variable speeds to maintain proper momentum and pointing. The primary drawback in this concept is the saturation limits of the system; the wheels may only spin so fast, reliably. To model this limitation, first the steering logic must be understood to extract the angular momentum of each reaction wheel. Using Eqn. 3.7, the logic may be described as [3]:

$$\dot{\vec{h}}_A = Z^{-1}(-\vec{\tau}_{req} - [\vec{\omega}^{\times}_{BN}]Z\vec{h}) \tag{3.10}$$

Where $\vec{\tau}_{req}$ is the requested torque from a control law, $Z$ is the distribution matrix of the reaction wheels, and $[\vec{\omega}^{\times}_{BN}]$ is the skew symmetric matrix of the spacecraft's angular velocity in the inertial frame. If $Z$ is noninvertible, a psuedo-inverse may be used. The ADCS modeled in this research attempted to emulate an industry-leading COTS ADCS from Blue Canyon Technologies, the XACT-15, where three reaction wheels are situated in an ortho-normal array and Z is a $3 \times 3$ identity matrix [1]. This is where saturation limits may be applied, as will be discussed later in this chapter.

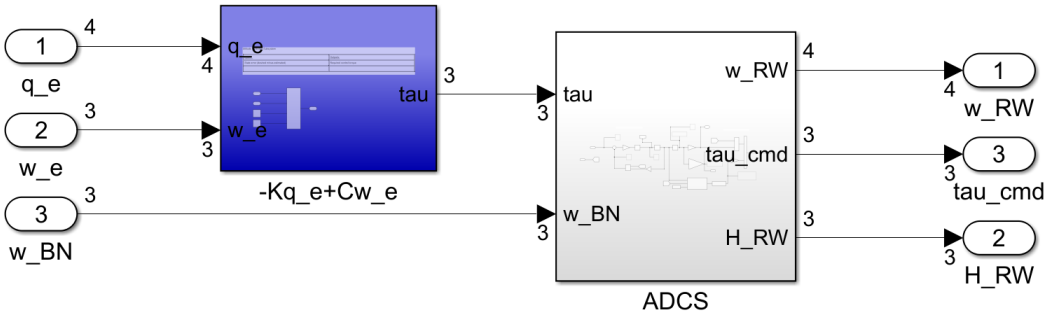


**Figure 6. Simulink ADCS**

**Table 2. BCT XACT-15 Properties and Settings [1]**

| | |
|---|---|
| $\vec{h}_{RW_{max}}$ (mN-m-s) per axis | 15 |
| $\dot{\vec{h}}_{RW_{max}}$ (mN-m) per axis | 4 |
| $Z_{RW}$ | $I_3$ |
| $\omega_{RW_{max}}$ rpm | 6,000 |
| $\omega_{RW_{des}}$ rpm | 4,000 |
| $\omega_{RW_{nom}}$ rpm | 500 |

### 3.3.2 Thruster Torques

Thrusters on satellites are used for momentum dumping, attitude control as well as translational maneuvers. Small satellites typically utilize the former, as it is a very quick way to desaturate the primary ADCS, without the need for much precision. Cold-gas propulsion used for attitude control has been an increasingly researched topic in the CubeSat space, due to actuators reaching the scale and precision required. While impulsive thrust is a common assumption made when commanding cold-gas thrusters, this research will utilize a near-continuous approach. Thrust time will be considered, but thrusters will turn on or off instantaneously.

Translational maneuvers are typically conducted via ion-thrusters and similar electronic propulsion solutions, increasing demand for agile small satellites has opened a market for cold-gas thrusters. Cold-gas thrusters are among the most mature technologies with respect to space vehicle subsystems [27]. The torques generated by these thrusters have typically been counteracted by secondary thrusters for momentum dumping, or a complex array of many thrusters. As explored in Ch. 2, fully-actuated control is typically desired when using a thruster-based ADCS, but is not always an option.

Thrust modulation must be used in these systems, as gimbal steering has not been adopted by COTS manufacturers yet. The same concept from momentum exchange devices applies, wherein the attitude must be controlled via torques described by a jet-selection logic [3]. The thrusters must then be modulated by the requested torque via a modulation technique such as Schmitt triggers, or Pulse-width modulation (PWM). The key to the linear programming problem discussed in [3], is that the system cannot be overdetermined. If it is underdetermined, there are infinitely many solutions in which to fire the thrusters and an optimal sequence may be found. However, this method does not consider the ADCS in its calculations. Typically, spacecraft will have full control of its associated subsystems, so a hybrid ADCS approach of combined control laws may be applied. Without such access to a proprietary COTS unit, the independent controllers may work against one another, causing saturation of the reaction wheels, or invalid thruster requests. For this reason, thrust modulation will be determined by ADCS telemetry at a minimum, and will incorporate more sophisticated techniques discussed in Section 3.4.

The propulsion system considered for this model is a variation of the VACCO Standard Micropropulsion System (MiPS) [28]. As seen in Fig. 7, the thrusters are at a $5^\circ$ angle towards the center of the unit. Recall in Fig. 4, the placement of the MiPS is on the $-x$ face. Each thruster is capable of 25 mN of thrust, which will be assumed constant and without operational losses. This assumption is made as the operational losses will be unavoidable, regardless of the thruster control scheme, and this analysis will focus predominantly on the pointing of the spacecraft.

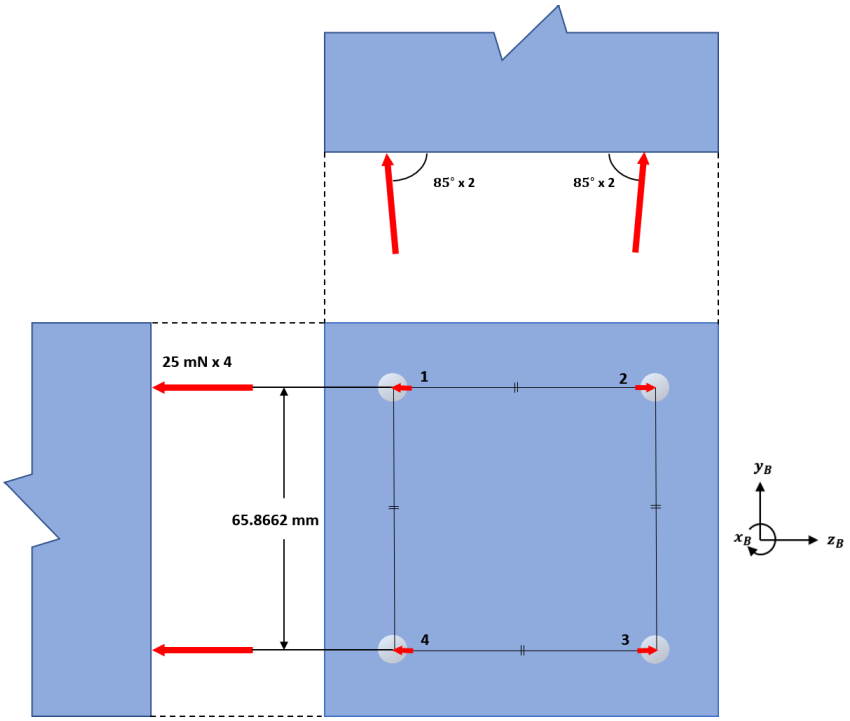


**Figure 7. Thruster Layout**

When implementing the thrusters into the Simulink model it is important to consider how the information will be transmitted. Consider Fig. 3, where the Command and Data Handling unit (C&DH) receives telemetry from the ADCS and must process it to be sent to the propulsion system. Each block and each transmission introduces delay. This is loosely defined as latency between the ADCS and propulsion system. Where there is latency, there is also typically temporal jitter, or fluctuations in latency. Figure 8 shows each latency item as well as a jitter of 100 ms. The ADCS and GNC also sample at specific rates. As CubeSats strive to be as computationally efficient as possible, sampling rates are typically matched between subsystems so that timestamps for state data align. The two sampling blocks are both set to 1 Hz, so they are redundant in this configuration. Although, it is not required to match sampling as such, so both blocks are included for best practice.

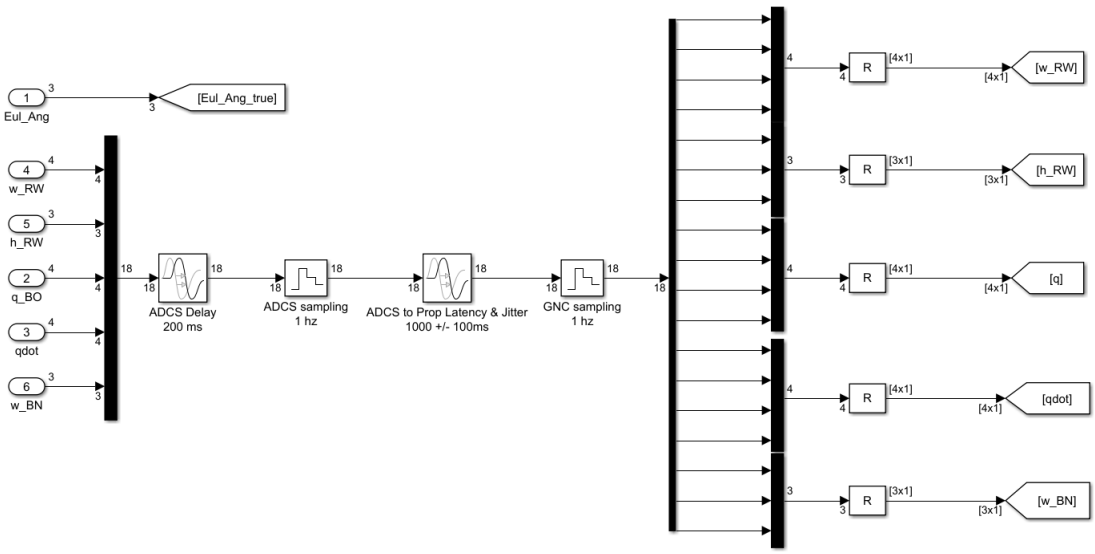


**Figure 8. Simulink Thruster Inputs**

### 3.3.3 Stability

Stability analysis is paramount in any control system. CubeSats must either be spin stabilized, utilize a gravity gradient, or have at least one of the attitude control methods discussed thus far in this section. Lyapunov Theory has been a mainstay in control system design over the past century [4]. This method is especially useful for nonlinear systems such as spacecraft motion. Using Lyapunov's direct method, a stable ADCS controller may be developed. Impulsive thrusters are fairly more difficult to model mathematically, utilizing relay describing functions [4]. This is not necessary with the use of Phase Plane analysis.

Before exploring stability analysis of the thrusters, the plant stability without thruster input must first be analyzed. Traditionally, the open-loop pertains to the plant's output for a given input with no feedback [29]. This idea is still implemented, only the ADCS will be included in the plant. For COTS components, of the appropriate specification, the space vehicle plant will then be locally asymptotically stable. Meaning, so long as the reaction wheels do not saturate, the spacecraft's attitude will always

reach the equilibrium point of no attitude error, nor rotational velocity error. The design space must be configured on a case-by-case basis as a function of initial reaction wheel speed, initial position, and initial rotational velocity.

To prove local asymptotic stability of the "Black Box" ADCS to be used in this model, first consider the quaternion feedback control (QFC) law [30]:

$$\vec{\tau} = -K\bar{q} - C\vec{\omega}_e \tag{3.11}$$

The $\bar{q}$ is the instantaneous error off the first three entries from the identity quaternion, $[0, 0, 0, 1]^T$. To determine if this control law is stable, consider the energy-like Lyapunov function and its derivative in Eqns. 3.12 and 3.13 [30]:

$$V(\vec{\omega}, q) = \frac{1}{2}\vec{\omega}^T K^{-1} J\vec{\omega} + \bar{q}^T\bar{q} + (q_4 - 1)^2 \tag{3.12}$$

Where $V(0, 0) = 0$ and $V$ is positive definite for all other values.

$$\dot{V} = -\vec{\omega}^T K^{-1} C\vec{\omega} \tag{3.13}$$

$$K^{-1}C > 0 \tag{3.14}$$

It can be said that this system is locally asymptotically stable so long that the relationship in Eqn. 3.14 holds, making $\dot{V}$ negative definite. The gains are selected to be:

$$K = \frac{\tau_{max}}{J_{min}} J \tag{3.15}$$

$$C = 1.414\sqrt{\frac{\tau_{max}}{J_{min}} J} \tag{3.16}$$

To assess stability graphically, a phase portrait may be used [4]. The phase plane shows the space vehicle's attitude, and rotational velocity on the x and y axes, respectively. The purpose of this plane is to highlight equilibrium points, limit cycles, or unstable behavior.

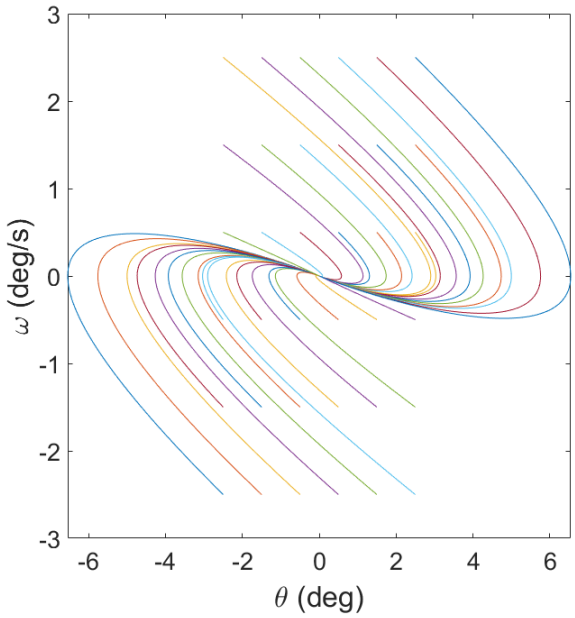


**Figure 9. Plant Phase Portrait**

Figure 9 shows the system's behavior from a spread of initial positions and angular velocities about the x axis. Notice that all converge to 0, indicating asymptotic stability. However, these are relatively small displacements. Figure 10 shows what happens when the ADCS reaction wheels are saturated. The simulations on the right portion of the figure show the reaction wheel tends to infinity due to settling at a 0.1 rad/s angular velocity. This same behavior is seen in the other axes. Between these two figures, the system may be considered to be locally asymptotically stable with an equilibrium point at $\vec{\Theta}_{BO} = [0, 0, 0]^T$ and $\vec{\omega}_{BO} = [0, 0, 0]^T$.

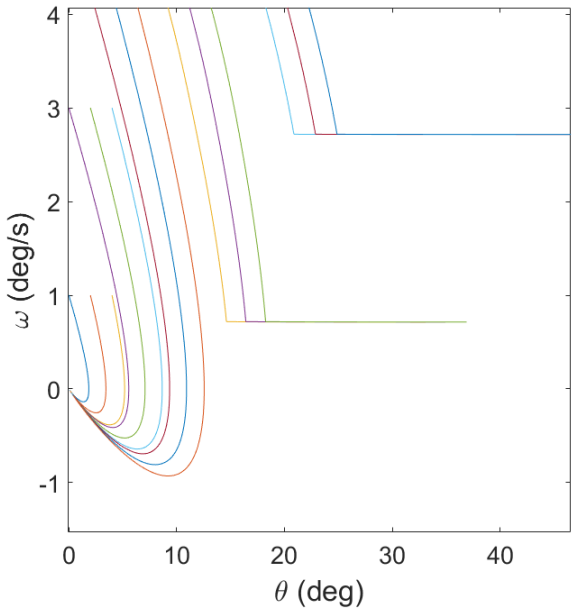


**Figure 10. Plant Phase Portrait with saturated reaction wheels**

While phase portrait analysis proved what was already known for the plant from the Lyapunov Direct Method, this will be a satisfactory analysis method for the closed-loop system with thruster modulation. Due to delays in the system, as well as the impulsive nature of thrusters, the phase portraits to be analyzed in Ch. 4 do not converge to the equilibrium point, but rather to a limit cycle encircling the equilibrium. Limit cycles can describe what is known as marginally stable behavior [4]. This is also sometimes called Lyapunov stable. Due to the thrusters never quite reaching equilibrium, the space vehicle oscillates about each axis resulting in $\Delta v$ to be expended at a sub-optimal angle. The smaller these oscillations, the smaller the limit cycle, and hence, generally, better performance to be seen by the $\Delta v$ efficiency metrics. The phase plane is a useful tool for nonlinear systems such as this.

## 3.4 Control Methods

### 3.4.1 Baseline Control

The first control method analyzed is simply no modulation of individual thrusters. This is representative of "nominal" thrusting without a modulated thrust solution. The metrics from this method will give a basis to which each subsequent control method may be compared. The essence of the Baseline Control is derived from the problem statement, and accurately shows the importance of solving this torque issue. When firing all thrusters simultaneously, the reaction wheels quickly saturate, due to the combined thrust vector offset on the order of millimeters from the center of mass (COM). For a propulsion system such as this, the only option to adjust this offset is to adjust the COM of the spacecraft entirely. Such precision is not always feasible for CubeSat missions, especially when considering this property would also need to be unaffected by launch.

The basic logic that goes into this technique is to simply fire all thrusters at the same time. Depending on the on-orbit offset of the thrust vector from the COM, the reaction wheels will exceed an operational RPM in a relatively short amount of time. Either the space vehicle's state machine, or the ADCS override, will switch to "detumble" where magentorquers are used to desaturate the reaction wheels. This is to prevent the spacecraft's attitude from becoming unstable. The magnetic torque is assumed to be $5 \times 10^{-6}$ Nm , which is reflective of the XACT-15's dipole moment[1], interacting with the approximated maximum average of the Earth's magnetic field (polar orbit). It is worth reiterating that this is the only control method to use magnetorquers, as the goal is to not saturate the reaction wheels and achieve a continually stable response.

The detumble-mode only ceases once the reaction wheels have returned to their nominal state. This simulates a "reset" of the spacecraft's initial desired state before

thrusting. The cycle then repeats until the desired $\Delta v$ has been achieved. As will be discussed in further detail in Ch. 4, the time to complete a small 1.5 m/s translational maneuver may take substantial time and resources, even assuming optimal magnetic torque. Additionally, the pointing accuracy during each thrust sequence is diminished as the ADCS torque may not be large enough to overcome the torque from the 4 thrusters. The Baseline Control method would not be an acceptable solution for most missions. Through thruster modulation,

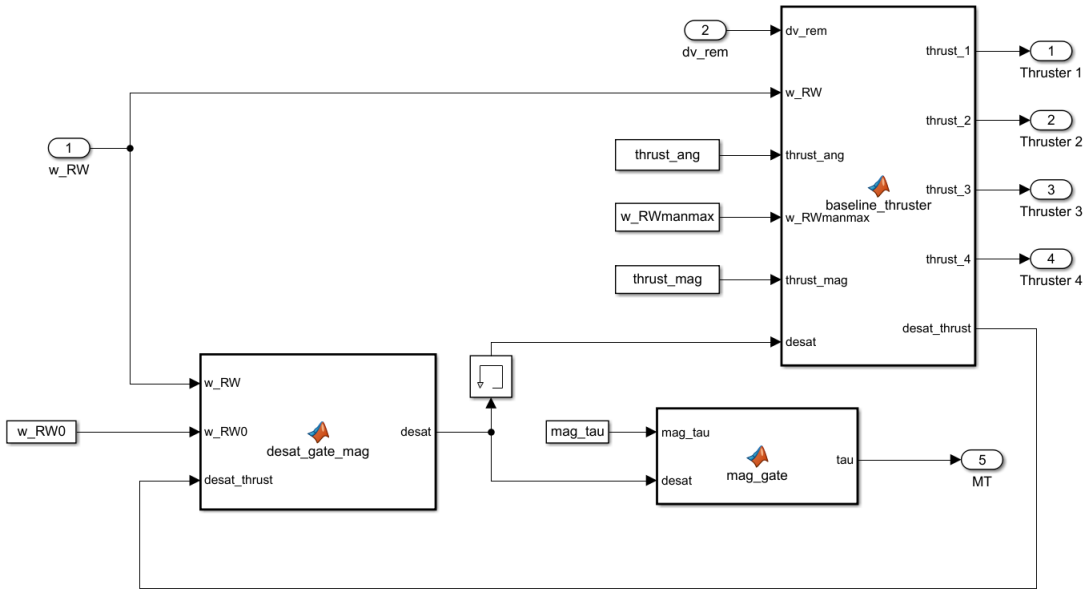


**Figure 11. Simulink Baseline Controller**

### 3.4.2 On-Off Control

On-off control attempts to approach a solution with minimal computational power. Impulsive and instantaneous thruster systems have used On-Off sequences for a variety of applications, from momentum dumping to attitude control [31]. Typically attitude control is approached much the same way as described earlier in this chapter: Derive the kinematic and dynamic relations, then translate the effects of the actuators to the system dynamics and develop a stable control law.

The control scheme used in this research will take a different approach. The simple Boolean logic of the controller will receive telemetry from the ADCS, specifically reaction wheel speed or momentum, and determine if each thruster is permitted to fire. If the reaction wheel has reached its maximum allowable momentum in the positive direction, the thrusters enacting positive torques will turn off. This then results in a negative torque from the remaining thrusters, thus reducing wheel speed. Once the reaction wheel speed has decreased to below the set threshold, the thrusters will begin to fire again. In the full model, this is completed for each axis, where a single thruster must pass conditions for all three reaction wheels. Recalling Fig. 7, it can be seen that the $y_B$ and $z_B$ axes experiencing the most torque. The $x_B$ axis only sees slight torques due to the nozzle angle of each thruster, as well as gyroscopic torques from the other reaction wheels. Regardless, it must be included in the algorithm to ensure the $x_B$ reaction wheel does not exceed operational limits, but not necessarily in the same fashion. Consider the pseudocode in Alg. 1.

**Algorithm 1:** On-Off Control Logic

**Data:** $\Delta v$ remaining, ADCS RW speed

**Result:** Thruster group fire command

All thrusters set to off;

**if** *$\Delta v$ remaining > 0 AND $|x - RW|$ < Maneuver max RPM* **then**

  **if** *y-RW speed $\geqslant$ -Desired RPM* **then**

    **if** *z-RW speed $\leqslant$ Desired RPM* **then**

      Fire thruster 1;

    **end**

  **end**

  **if** *y-RW speed $\leqslant$ Desired RPM* **then**

    **if** *z-RW speed $\leqslant$ Desired RPM* **then**

      Fire thruster 2;

    **end**

  **end**

  **if** *y-RW speed $\geqslant$ -Desired RPM* **then**

    **if** *z-RW speed $\geqslant$ -Desired RPM* **then**

      Fire thruster 3;

    **end**

  **end**

  **if** *y-RW speed $\leqslant$ Desired RPM* **then**

    **if** *z-RW wheel speed $\geqslant$ -Desired RPM* **then**

      Fire thruster 4;

    **end**

  **end**

**end**

Any thruster is only allowed to fire under the conditions that there is $\Delta v$ remaining within the burn, and also the $x_B$ reaction wheel has not exceeded a "Manuever Maximum RPM." This differs from the "Desired RPM" used for the other reaction wheels, as the $x_B$ wheel will not experience quick or sudden changes from low torque. It is simply a check to ensure it stays within bounds. If this limit is met, the reaction wheel either will almost immediately drop below the uppter bound of 6000, or continue to "bounce" off of the -6000 RPM limit naturally due to gyroscopic torques as the orbit propagates in a prograde orbit. This behavior will become prevalent in the other control laws.

When considering applying this method to a particular spacecraft or use-case the following questions must be addressed. Which thrusters are applying which torques to which reaction wheels? In this model, the ADCS and the propulsion system are operating in the body frame. When using COTS components, the corresponding reaction wheel may not align as such. What is the deadband due to latency between the ADCS and propulsion unit? This drives the amount of overshoot experienced by the reaction wheels, which are being driven to the desired RPM. For example, if the ADCS internally flags stop conditions when a reaction wheel reaches 6000 RPM, the desired RPM must be set low enough such that overshoot due to latency of the reaction wheel does not reach 6000 RPM with margin. Setting the desired RPM too low will cause the reaction wheels to oscillate about 0 rad/s, which may or may not be acceptable for the ADCS hardware. This model will use a desired RPM of 4000, and maneuver maximum RPM of 6000 to incorporate these constraints.

### 3.4.3 Quaternion Feedback Control

QFC was described in the previous section as a use case for the ADCS. The methods used in this research will follow a similar approach, only where the thrusters are not directly applying the input. Similar to the On-Off algorithm, the QFC method relies on thruster modulation given the reaction wheel RPM. With the nonlinear control algorithms, the sign of the requested thrust about each axis is also taken into consideration.

By using the control logic described in Eqn. 3.11 as the Control torque input, $\vec{u}$, Algs. 2 through 6 will modulate the thrusters accordingly. There is an additional check placed in this algorithm due to a frequent issue in testing. In Ch. 4, it will be discussed that the $y$ and $z$ reaction wheels oscillate about the desired RPM, finding a local equilibrium state. However, due to the additional input requirement in the nonlinear control algorithms, a conflict between the commanded torque and $y$ reaction wheel speed occurs for all thrusters. When this occurs, the command torque may maintain the same sign, thus resulting in the thrusters to never fire. A check condition is then placed at the bottom of the algorithm to jump-start the system. It is important to tune this such that it does not interfere with the nominal firing sequence of the controller by being too short, but also not so long such that the conflict continues for minutes on end, unnecessarily increasing the maneuver time.

**Algorithm 2:** QFC/SMC Logic

**Data:** $\Delta v$ remaining, ADCS RW speed, Control torque, $n$

**Result:** Thruster group fire command

All thrusters set to off;

**if** *$\Delta v$ remaining > 0 AND $|x - RW\ speed| <$ Maneuver max RPM* **then**

  **if** *$|y\text{-}RW\ speed| <$ Desired RPM AND $|z\text{-}RW\ speed| <$ Desired RPM* **then**

    Fire all thrusters;

  **end**

  **else**

    **if** *Alg. 3 == true* **then**

      Fire Thruster 1;

    **end**

    **if** *Alg. 4 == true* **then**

      Fire Thruster 2;

    **end**

    **if** *Alg. 5 == true* **then**

      Fire Thruster 3;

    **end**

    **if** *Alg. 6 == true* **then**

      Fire Thruster 4;

    **end**

  **end**

  **if** *$\Delta v$ has not changed in $n$ time steps* **then**

    Fire all thrusters for 1 time step;

  **end**

**end**

**Algorithm 3:** QFC/SMC Thruster 1 Logic

**Data:** ADCS RW speed, Control torque

**Result:** Thruster group fire command

**if** *Control torque-x* $\leqslant$ *0* **then**

  **if** *y-RW speed* $\geqslant$ *-Desired RPM AND Control torque-y* $\leqslant$ *0* **then**

    **if** *z-RW speed* $\leqslant$ *Desired RPM AND Control torque-z* $\geqslant$ *0* **then**

      Output = True;

    **end**

  **end**

**end**

**Algorithm 4:** QFC/SMC Thruster 2 Logic

**Data:** ADCS RW speed, Control torque

**Result:** Thruster group fire command

**if** *Control torque-x* $\geqslant$ *0* **then**

  **if** *y-RW speed* $\leqslant$ *Desired RPM AND Control torque-y* $\geqslant$ *0* **then**

    **if** *z-RW speed* $\leqslant$ *Desired RPM AND Control torque-z* $\geqslant$ *0* **then**

      Output = True;

    **end**

  **end**

**end**

**Algorithm 5:** QFC/SMC Thruster 3 Logic

**Data:** ADCS RW speed, Control torque

**Result:** Thruster group fire command

**if** *Control torque-x* $\geqslant$ *0* **then**

  **if** *y-RW speed* $\geqslant$ *-Desired RPM AND Control torque-y* $\leqslant$ *0* **then**

    **if** *z-RW speed* $\geqslant$ *-Desired RPM AND Control torque-z* $\leqslant$ *0* **then**

      Output = True;

    **end**

  **end**

**end**

**Algorithm 6:** QFC/SMC Thruster 4 Logic

**Data:** ADCS RW speed, Control torque

**Result:** Thruster group fire command

**if** *Control torque-x* $\leqslant$ *0* **then**

  **if** *y-RW speed* $\leqslant$ *Desired RPM AND Control torque-y* $\geqslant$ *0* **then**

    **if** *z-RW wheel speed* $\geqslant$ *-Desired RPM AND Control torque-z* $\leqslant$ *0*

    **then**

      Output = True;

    **end**

  **end**

**end**

### 3.4.4 Sliding Mode Control

Sliding Mode Control (SMC) is a technique typical utilized for systems with low precision or high uncertainty, such as magnetorquer attitude control [32]. These model uncertainties may be labeled as either structured, inaccuracies within the model, or unstructured, inaccuracies on the system order. The former umbrellas the use case for this particular attitude control problem, as the time delay presents error in the "truth" data for the states being fed to the thruster control logic. There are two methods to approach SMC with respect to attitude control. The first method will utilize the equations of motion in their current state, and derive the control law in terms of $\vec{\omega}_{BN}$. This sliding mode technique will be referred to in this work as SMC ($\vec{\omega}$-based). Another method is to consider the relationship of the system dynamics to quaternion representation, similar to QFC. First, consider the SMC-$\vec{\omega}$ varient. Recall the dynamics EOM in Eqn. 3.9, and reconsider them with the ADCS and thruster torques.

$$J\dot{\vec{\omega}}_{BN} + \vec{\omega}_{BN} \times J\vec{\omega}_{BN} = \vec{\tau}_T + \vec{\tau}_{RW} \tag{3.17}$$

The generalized control affine form may be written as Eqn. 3.18 [4]. $x^{(n)}$ represents the state variable of the $n$th order, which is then constructed by functions of lower orders of that state variable, and the input.

$$x^{(n)} = f(x) + b(x)u \tag{3.18}$$

Where $f(x)$ and $b(x)$ may be some unknown functions, but the imprecision is known to be upper bounded by some function. Equation 3.19 describes how the spacecraft attitude dynamics fit into this construct.

$$
\begin{aligned}
f_1(x) &= f_1(\vec{\omega}_{BN}) = -J^{-1}\vec{\omega}_{BN} \times J\vec{\omega}_{BN} \\
f_2(x) &= f_2(\vec{\omega}_{BN}) = -J^{-1}\vec{\tau}_{RW} \\
b(x) &= 1
\end{aligned}
\tag{3.19}
$$

The estimation error may be written as:

$$
F \geqslant |f_1 - \hat{f}_1| + |f_2 - \hat{f}_2| \tag{3.20}
$$

Where the estimation error is the worst case scenario, where $f_1$ is the maximum angular acceleration the spacecraft may experience, $f_2$ is the maximum angular acceleration of the spacecraft from the ADCS, and $\hat{f}_{1,2} = -f_{1,2}$ for the most error possible. With a 4 mm combined thruster offset, the worst torque that may be seen about a particular axis is moment arm created by the sum of 2 thrusters at a distance of 36.9331 mm due to the geometry of the thruster layout. These values will be identical for $y$ and $z$, but torque about $x$ due to the thrusters will be near negligible.

$$
F \geqslant 2J^{-1}(\vec{\tau}_{max,T} + \vec{\tau}_{max,RW}) \tag{3.21}
$$

The sliding surface is the core of SMC. This theoretical surface is what the controller drives the system towards by controlling $\dot{s}$ to 0. While this ensures stability, it may also lead to chattering. As chattering will already exist due to the instantaneous thrusters and sampling rates of the hardware, this drawback is essentially negligible, giving a strong case for SMC. The operator in Eqn. 3.22 is applied $n-1$ times, where $n$ is the order of the system. While the equations show a first order system, it is technically second order in terms of attitude angle, therefore $n = 2$. $\tilde{\omega}$ is then defined as the tracking error, or $\tilde{\omega} = \vec{\omega}_{BN} - \vec{\omega}_{ON}$.

$$\vec{s} = \left( \frac{d}{dt} + \lambda_S \right)^{n-1} \tilde{\omega} = \dot{\tilde{\omega}} + \lambda_S \tilde{\omega} \tag{3.22}$$

$$\dot{\vec{s}} = \ddot{\tilde{\omega}} + \lambda_S \dot{\tilde{\omega}} \tag{3.23}$$

Where $\lambda$ is a positive gain, and $\dot{\omega}_{BN}$ may be defined as:

$$\begin{aligned} \dot{\tilde{\omega}} &= f_1 + f_2 + u - \dot{\tilde{\omega}}_d \\ &= -J^{-1}\vec{\omega}_{BN} \times J\vec{\omega}_{BN} - J^{-1}\vec{\tau}_{RW} + u - \dot{\tilde{\omega}}_d \end{aligned} \tag{3.24}$$

Recall that the system is being driven to the orbital frame, or an angular acceleration of 0, so $\dot{\tilde{\omega}}_d = [0, 0, 0]^T$. The control law below may be used [4]:

$$\vec{u} = \hat{u} - K \odot sgn(\vec{s}) \tag{3.25}$$

Where $\hat{u}$ and $K$ are selected to ensure the sliding condition in Eqn. 3.26 is true. $\vec{\eta}$ must be a $3 \times 1$ vector to maintain dimensional continuity. The operator $\odot$ represents a element by element multiplication.

$$-\vec{\eta}|sgn(\vec{s})| \leqslant \vec{s}\dot{\vec{s}} \tag{3.26}$$

$$\hat{u} = -\hat{f}_1 - \hat{f}_2 + \dot{\tilde{\omega}}_d - \frac{1}{\lambda}\ddot{\tilde{\omega}} = F - \frac{1}{\lambda}\ddot{\tilde{\omega}} \tag{3.27}$$

$$K_{SM} = F \odot sgn(\vec{s}) + J\frac{\vec{\eta}}{\lambda_S} \tag{3.28}$$

In general, the time for the control will reach the sliding surface in less than $|s|/\eta$ for each time step [4]. $\vec{\eta}$ is then selected to be $[100, 100, 100]^T$ to be large enough to minimize this theoretical time.

Combining Eqns. 3.25, 3.27, 3.28, and 3.20 yields the final input equation:

$$\vec{u} = \frac{1}{\lambda_S}(\ddot{\vec{\omega}}_{BN} + J\vec{\eta} \odot sgn(\vec{s})) \tag{3.29}$$

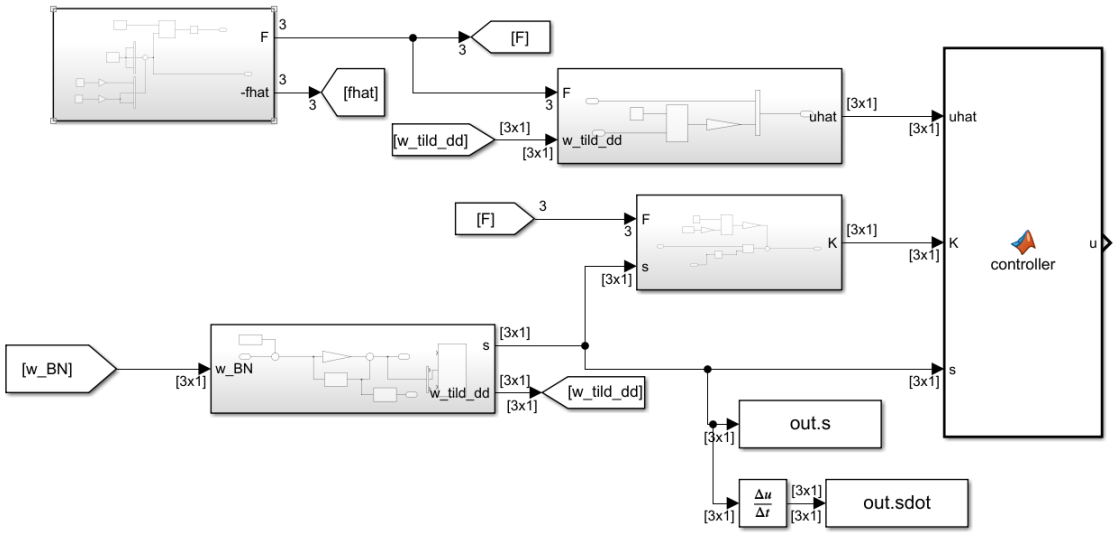


**Figure 12. Simulink Sliding Mode Controller SMC-$\vec{\omega}$**

There is another sliding mode variation may be implemented utilizing quaternions [33]. Consider the sliding surface:

$$\vec{s} = P\bar{q} + \vec{\omega}_{BO} \tag{3.30}$$

The sliding surface is only achieved when both $\bar{q}$ and $\vec{\omega}_{BO}$ are null, thus constraining the system to equilibrium. The control law may then be defined as [33]:

$$\vec{u} = -K_s\vec{s} + \dot{J}\vec{\omega}_{BN} + \frac{1}{2}\dot{J}s + \vec{\omega}_{BN} \times J_0\vec{\omega}_{BN} - \frac{1}{2}J_0P\dot{\bar{q}} + \Lambda_s \tag{3.31}$$

The control law in Fig. 3.31 is even capable tracking and adjusting to slight changes in the moment of inertia. As this work considers a rigid body with stationary COM, $\dot{J} = 0$, $J_0 = J$, and $\vec{u}$ may be simplified to:

$$\vec{u} = -K_s\vec{s} + \vec{\omega}_{BN} \times J\vec{\omega}_{BN} - \frac{1}{2}JP\dot{\vec{q}} + \Lambda_s \tag{3.32}$$

Where $P$ and $K_s$ are positive diagonal gain matrices. $\Lambda_s$ is the control action with:

$$\Lambda_s = -F \odot sgn(\vec{s}) \tag{3.33}$$

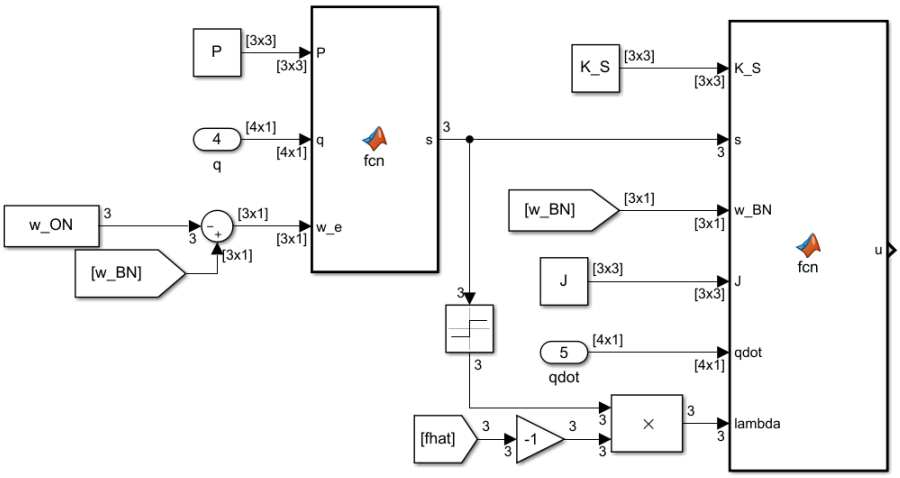


**Figure 13. Simulink SMC-Q**

This variation of sliding mode will then be referred to as SMC (quaternion-based) to differentiate from the former SMC ($\vec{\omega}$-based). Alg. 1 is then modified to create Alg. 2 through Alg. 6. The core of this logic remains the same, but an extra constraint is added to the conditions for each thruster. The controller must output the appropriate sign for each axis, given the thruster geometry. This ensures that no reaction wheel will become saturated. The maneuver maximum safeguard still remains as a check for the $x$ reaction wheel.

The initial IF statement in Alg. 2 turning all thrusters on is to begin each at full thrust. If the COM has not shifted significantly enough away from the thrust vector to create moment arms to saturate the reaction wheels, there is no need for thrust modulation. Additionally, the sliding mode control behavior will favor only two of the thrusters too heavily without this initial "kick-off." Better performance was seen from including the initial fire sequence.

### 3.4.5 Model Reference Adaptive Control

Model Reference Adaptive Control (MRAC) is a powerful adaptive control technique that is a relative increase in sophistication over robust control methods, like sliding mode. Adaptive controllers will consistently improve performance as adaptation continues, and often require no priori knowledge about the plant [4]. MRAC inacts this by utilizing a reference model and adaptation law to eventually feed into the control logic. For a nonlinear control MRAC scheme, the system order is known, but some system parameters are not. A generic MRAC setup is shown in Fig. 14 [4].

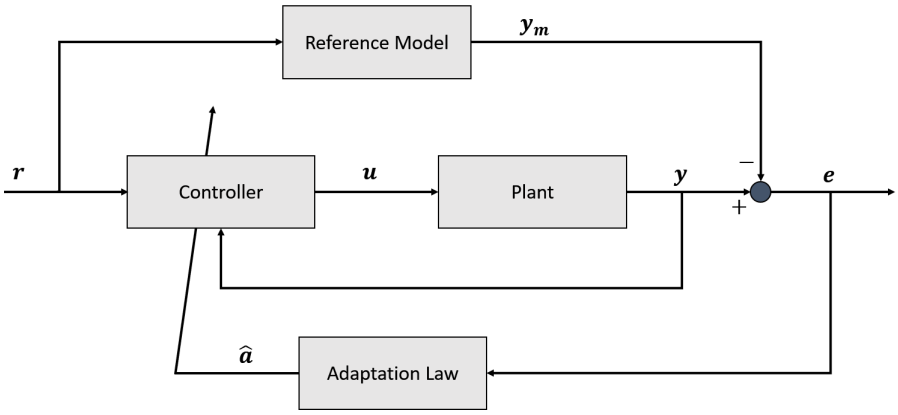


**Figure 14. MRAC Block Diagram**

Consider the control affine form from Eqn. 3.18, but written as [4]:

$$Hx^{(n)} + \sum_{i=1}^{n} a_i f_i(x,t) = u \tag{3.34}$$

Recall in sliding mode the two functions considered were the spacecraft's dynamics and the torque from the ADCS. This method uses the same equations, much for the same reason. The main benefit to MRAC over sliding mode is the coefficient terms, which adapt after each time step. These coefficients will effectively respond to the uncertainties within the model. Specifically for $a_1$, how the impulses and sampling affect the control logic, but also for future iterations of this research where complexities are added. For example, if fuel sloshing dynamics were added, $a_1$ would attempt to adapt to the modes perpetuated by that system. For this case, there are not many uncertainties other than the hardware disturbances. $a_2$ will adapt according to the ADCS's control law. This is especially powerful, given this particular use case, as the control law is unknown to the propulsion unit. The dynamics of the momentum exchange device, is known, so it may be included as $f_2$.

$$f_1 = \vec{\omega}_{BN} \times J\vec{\omega}_{BN} \tag{3.35}$$

$$f_2 = \dot{\vec{h}}_A + \vec{\omega}_{BN} \times Z\vec{h}_A \tag{3.36}$$

The MRAC control law may be written as [4]:

$$u = Hx_r^{(n)} + \sum_{i=1}^{n} a_i f_i(x,t) - k_M s \tag{3.37}$$

Where $H \rightarrow \vec{H}$ is the highest order state variable coefficient, and in this case a $3 \times 1$ vector, and $k_M$ is a constant gain.

$$\vec{s} = \vec{\omega}_{BN} - \vec{\omega}_r \tag{3.38}$$

$$\vec{\omega}_r = \vec{\omega}_d - \lambda e \tag{3.39}$$

The state error is found by $e = \vec{\omega}_{BN} - \vec{\omega}_{ON}$, and $\lambda$ is a positive constant gain. The Reference Model block is defined as an "ideal" response of the system to an input. The $y_m$ output of the reference model should match desired performance characteristics, such as rise time or settling time. In the dual-opposed thrusters system, the ideal response should result in the minimum amount of pointing error, while indirectly reaching and maintaining the desired maximum speed of the reaction wheel. Like sliding mode, some chattering is expected due to the nonlinearities found in instantaneous thrust.

$$J\dot{\vec{s}} + k_M \vec{s} = 0 \tag{3.40}$$

The Adaptation Law block will then take that ideal state, and adjust controller parameters accordingly. The objective of the adaptation law is to drive the tracking error to zero, while maintaining stability. The control law is rewritten to be:

$$\vec{u} = J^{-1}(\hat{H} \odot \vec{\omega}_r - k_M \vec{s} + \hat{a}_1 \odot f_1 - \hat{a}_2 \odot f_2) \tag{3.41}$$

$$\vec{H} \odot \dot{s} + k_M J^{-1} \vec{s} = \tilde{H} \odot \dot{\vec{\omega}}_r + \tilde{a}_1 \odot f_1 - \tilde{a}_2 \odot f_2 \tag{3.42}$$

Where $\tilde{a}_i = \hat{a}_i - a_i$. The adaptation law may then be described via the following equations:

$$\dot{\hat{H}} = -\gamma \vec{s} \odot \vec{\omega}_r \tag{3.43}$$

$$\dot{\hat{a}}_1 = -\gamma \vec{s} \odot f_1 \tag{3.44}$$

$$\dot{\hat{a}}_2 = \gamma \vec{s} \odot f_2 \tag{3.45}$$

Where $\gamma$ is a positive constant. Equations 3.43-3.45 are then numerically integrated and fed back into Eqn. 3.41 to determine the control input.

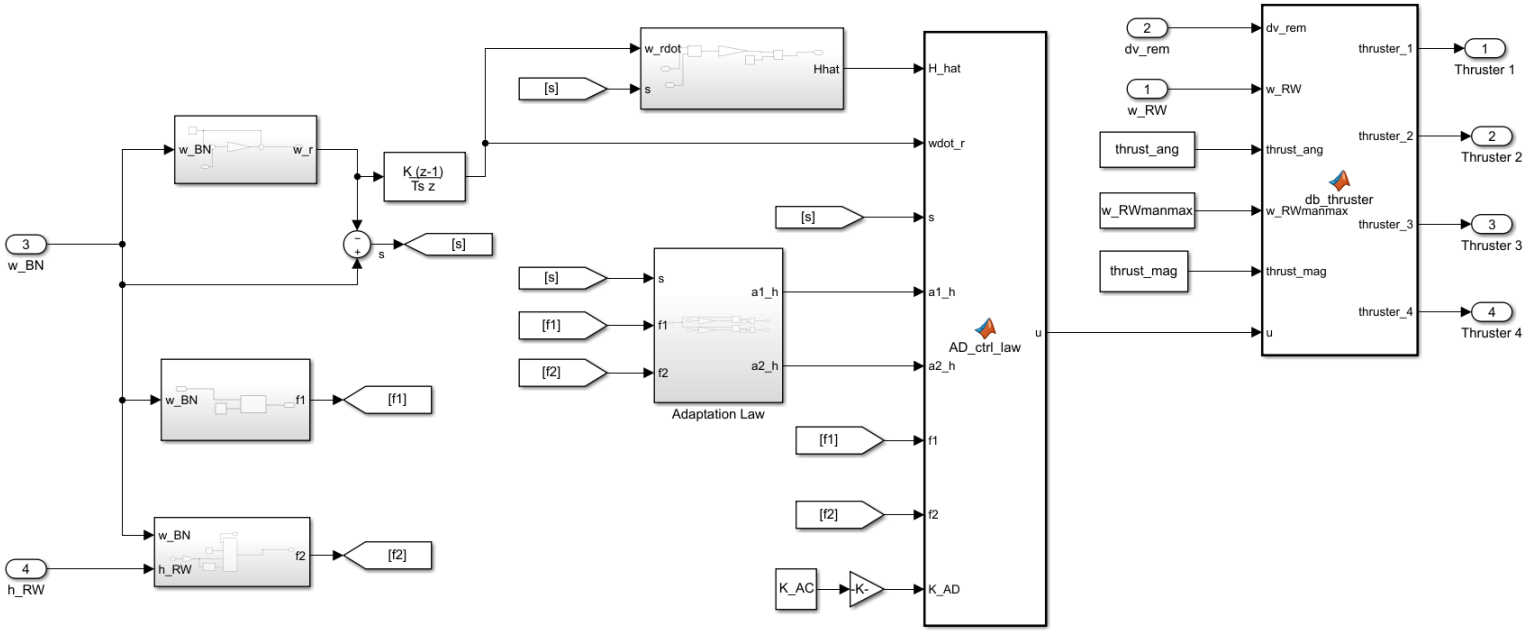


**Figure 15. Simulink Model Reference Adaptive Controller**

**Algorithm 7:** MRAC Logic

**Data:** $\Delta v$ remaining, ADCS RW speed, Control torque

**Result:** Thruster group fire command

All thrusters set to off;

**if** *$\Delta v$ remaining > 0 AND $|x - RW\,speed| <$ Maneuver max RPM* **then**

  **if** *$|$y-RW speed$|$ < Desired RPM AND $|$z-RW speed$|$ < Desired RPM* **then**

    Fire all thrusters;

  **end**

  **else**

    **if** *Alg. 8 == true* **then**

      Fire Thruster 1;

    **end**

    **if** *Alg. 9 == true* **then**

      Fire Thruster 2;

    **end**

    **if** *Alg. 10 == true* **then**

      Fire Thruster 3;

    **end**

    **if** *Alg. 11 == true* **then**

      Fire Thruster 4;

    **end**

  **end**

**end**

**Algorithm 8:** MRAC Thruster 1 Logic

**Data:** ADCS RW speed, $\vec{u}$

**Result:** Thruster group fire command

**if** *y-RW speed* $\geqslant$ *-Desired RPM AND Control torque-y* $\leqslant$ *0* **then**

  **if** *z-RW speed* $\leqslant$ *Desired RPM AND Control torque-z* $\geqslant$ *0* **then**

    Output = True;

  **end**

**end**

**Algorithm 9:** MRAC Thruster 2 Logic

**Data:** ADCS RW speed, Control torque

**Result:** Thruster group fire command

**if** *y-RW speed* $\leqslant$ *Desired RPM AND Control torque-y* $\geqslant$ *0* **then**

  **if** *z-RW speed* $\leqslant$ *Desired RPM AND Control torque-z* $\geqslant$ *0* **then**

    Output = True;

  **end**

**end**

**Algorithm 10:** MRAC Thruster 3 Logic

**Data:** ADCS RW speed, Control torque

**Result:** Thruster group fire command

**if** *y-RW speed* $\geqslant$ *-Desired RPM AND Control torque-y* $\leqslant$ *0* **then**

  **if** *z-RW speed* $\geqslant$ *-Desired RPM AND Control torque-z* $\leqslant$ *0* **then**

    Output = True;

  **end**

**end**

**Algorithm 11:** MRAC Thruster 4 Logic

**Data:** ADCS RW speed, Control torque

**Result:** Thruster group fire command

**if** *y-RW speed* $\leqslant$ *Desired RPM AND Control torque-y* $\geqslant$ *0* **then**

  **if** *z-RW wheel speed* $\geqslant$ *-Desired RPM AND Control torque-z* $\leqslant$ *0* **then**

    Output = True;

  **end**

**end**

The only difference between the algorithms is that MRAC does not require a check for the $x$ reaction wheel in each thruster. As will be seen later, the adaptation parameters account for the flag condition set for $x - RW$ speed at the beginning of Alg. 7. This evidence gives confidence for later improvements on the model, and can be seen even more clearly in Ch. 4 when conducting the Monte Carlo analysis.

## 3.5 Summary

This chapter has been a lengthy discussion setting up all of the mathematics needed to run this simulation. Initially, the kinematics and dynamics were required to determine how the spacecraft would behave throughout the simulation - and to derive the respective control laws. The ADCS is the primary momentum management device on the CubeSat. However, the thrusters impart sustained torques large enough to saturate the ADCS reaction wheels when all are firing. For this reason a modulation scheme must be developed. Nonlinear control methods are typically used to include complex EOM, uncertainty in parameters, and adaptation to unknown external inputs. On-off control is among the simplest control schemes as it does not require torque input, but rather state data to operate the Boolean algorithm. QFC was initially established to control the ADCS of the spacecraft plant, but is also considered for thruster modulation as the same dynamics apply. SMC is a popular method for systems with low precision and high uncertainty, giving a robust control response. MRAC's self adaptation uses methods similar to sliding mode, but also includes measures to account for changes within the system as they occur in real-time. This research explores these control techniques enacted through a similar algorithm introduced in On-Off control to overcome the overdetermined system. From the Simulink model incorporating the equations explained in this chapter, stability analysis via the phase plane may be conducted.

# IV. Results and Analysis

## 4.1 Overview

To analyze the methods described in Ch. 3, the establishment of particular use cases is necessary. The two considerations for this research include an extended "deorbit" burn lasting 5000 sec, and a relatively small $\Delta v$ of 1.5 m/s. The reaction wheel speeds are analyzed as a functional stability check of the controller. Each method must keep the reaction wheels within operational limits, such that the ADCS does not enact a recovery, or desaturation, mode. The thruster sequences are also tracked as to understand the actual output of the combined controller and algorithm. Sliding surfaces and adaptation parameters, for SMC and MRAC respectively, are displayed to help showcase the controller's efficacy. The final conclusions are then determined by $\Delta v$ and simulation time, shown for each control method through Monte Carlo analysis of 10,000 scenarios of varying physical properties.

The total $\Delta v_{spent}$ accrues as the simulation progresses. This is described in Eqn. 4.1, where a discrete integration may be assumed due to there being no ramp from each thruster being off to on. Each thruster contributes an equal amount of $\Delta v$ per second, so each may be summed as well. This is the primary metric driving the stopping condition for the simulation, when $\Delta v_{spent} = \Delta v_{req}$. Alternatively, a pure $\Delta v$ (in the desired direction) may be accrued instead. The former method was chosen as a way to directly compare $\Delta v$ efficiency over the same $\Delta v$ expelled from the spacecraft.

$$\Delta v_{spent} = \sum_{t=0}^{t_f} \sum_{i=1}^{4} \frac{F_i(t)\Delta t}{m} \tag{4.1}$$

The $\Delta v$ spent not in line with the spacecraft's velocity vector, is essentially wasted. This is found by converting the thrust vectors, originally in the body frame,

to the inertial frame, and finding the projection of this vector onto the $y_O$ and $z_O$ axes to determine loss.

$$\Delta v_{waste} = \sum_{t=0}^{t_f} \sum_{i=1}^{4} \frac{\Delta t}{m} (C_{BO} |\vec{F}_i(t)|) \cdot \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \tag{4.2}$$

Efficiency is then simply calculated below, normalized by the total $\Delta v$ spent.

$$Eff_{\Delta v} = \frac{\Delta v_{spent} - \Delta v_{waste}}{\Delta v_{spent}} \times 100\% \tag{4.3}$$

## 4.2 Case Study: Deorbit Maneuver

The deobrit maneuver is an important first step when conducting this research for several reasons. First and foremost, the stability of the controller over a theoretical maximum burn time is paramount. In Section 4.3, the 1.5 m/s $\Delta v$, based on a typical operational maneuver, will see a more detailed analysis. However, the shorter burn time does not necessarily allow for the system to converge to an equilibrium state. The Baseline control method will only be discussed in the small maneuver, as the reader will see its thrust cycle is consistent. Therefore, a deorbit analysis is not required for the Baseline. The deorbit time was selected to be 5000 seconds, based on the longest time for a control method to adequately converge into a thrust sequence. It was also during the deorbit maneuver that respective gains for applicable controllers were tuned, again to reach a consistent thrust sequence, and therefore stable response. Phase plane analysis may then be conducted to describe the consistency of each method.

### 4.2.1 On-Off Control

As discussed in Ch. 3, On-Off control is fairly straight-forward. Its simple algorithm allows for the convergence of the reaction wheels immediately following the initiation burn, as shown in Fig. 16. The $y$ and $z$ reaction wheels see a higher frequency of oscillations due to the higher torques, which is to be expected. What is more impressive is the stable response seen by the $x$ reaction wheel, which never meets the maneuver maximum of 6000 RPM. This is due to the balance between the other two reaction wheels and their opposing $z_B$ components in the thrust vectors.

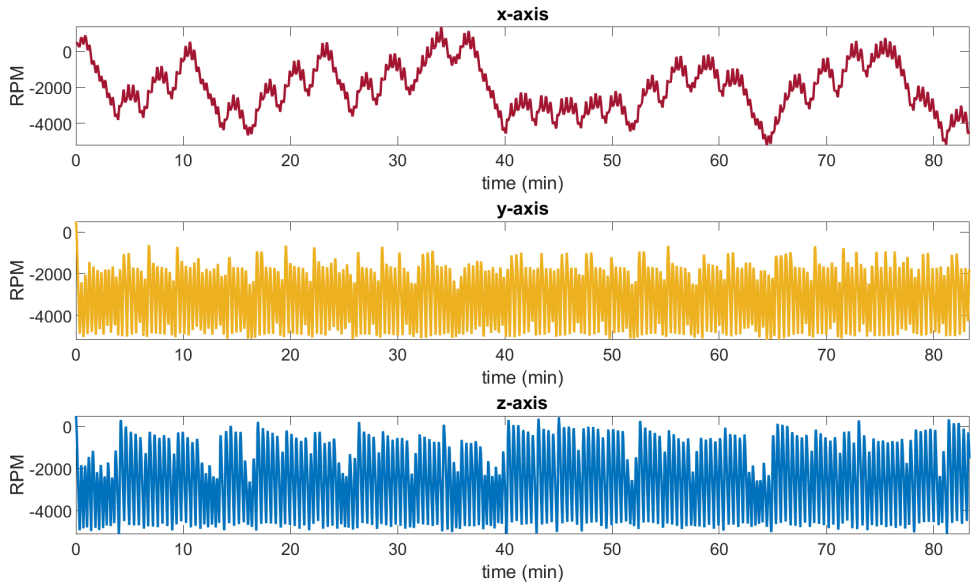


**Figure 16. On-Off Reaction Wheels: Extended burn**

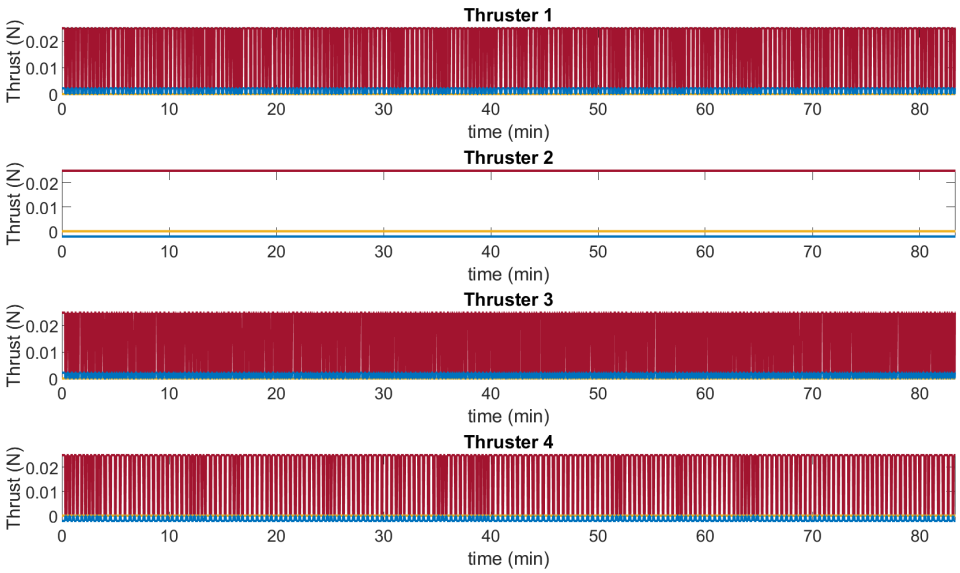


**Figure 17. On-Off Thrusters: Extended burn**

The thruster sequence shows a consistent burn of Thruster 2, as its offset from the center of mass is lesser in both $y$ and $z$. If the offsets were such that the maximum $y_B$ and $z_B$ lever arms were opposite of one another, there would be a switching behavior between either Thruster 2 and Thruster 3, or Thruster 1 and Thruster 4. Compare Figs. 16 and 18 as well as Figs. 17 and 19. At approximately 28 minutes, the $x$ reaction wheel hits the lower bound of the maneuver maximum RPM. This causes the algorithm to halt, waiting for the orbital motion to naturally bring the $x$ above -6000 RPM. This is not a desirable outcome. While the center of mass should be as closely aligned to the thrust vector as possible, it is vital for the algorithms in this research that one thruster specifically has the most mechanical advantage in both $y_B$ and $z_B$.

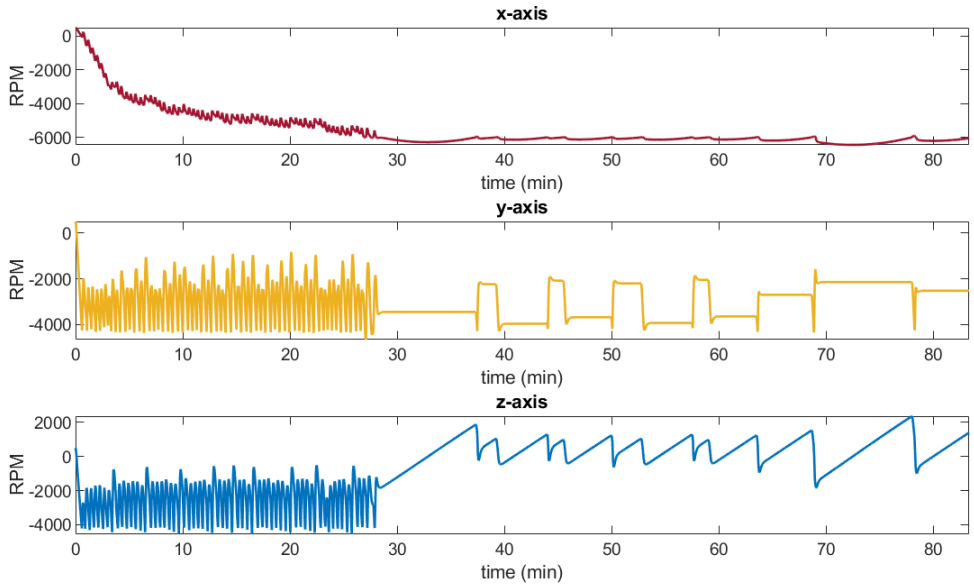


**Figure 18. On-Off Reaction Wheels (staggered offset): Extended burn**

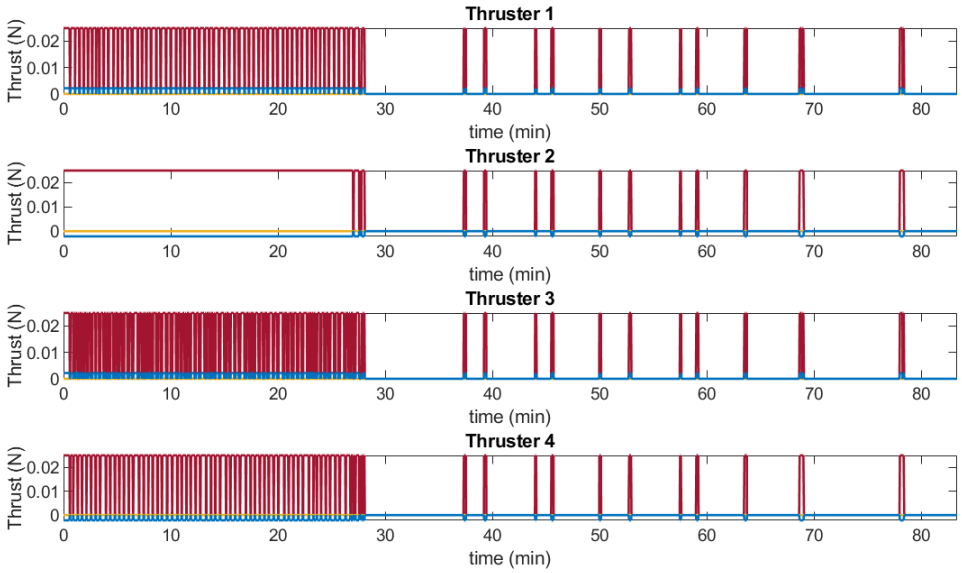


**Figure 19. On-Off Thrusters (staggered offset): Extended burn**

The phase portrait of the On-Off controller in Fig. 20 shows limit cycle-like behavior. Due to the jitter of the input signal, it is virtually impossible to maintain a true limit cycle. Nevertheless, the overall shape of the limit cycle is apparent. The subplots showing the final 1 minute of the simulation give a clear view of the encirclement of the origin, indicating Lyapunov stability [4]. Due to the magnitude of pointing error seen in the the phase portrait, a qualitative measure of the $\Delta v$ efficiency may be inferred, but a quantitative measure will prove to be more succinct. For the single simulation aggregated over 5000 seconds, the $\Delta v$ efficiency was found to be roughly 88.09%. Margin for error of this metric will be discussed in the Monte Carlo analysis of the phase change maneuver, but it gives a close approximation to the average efficiency expected.

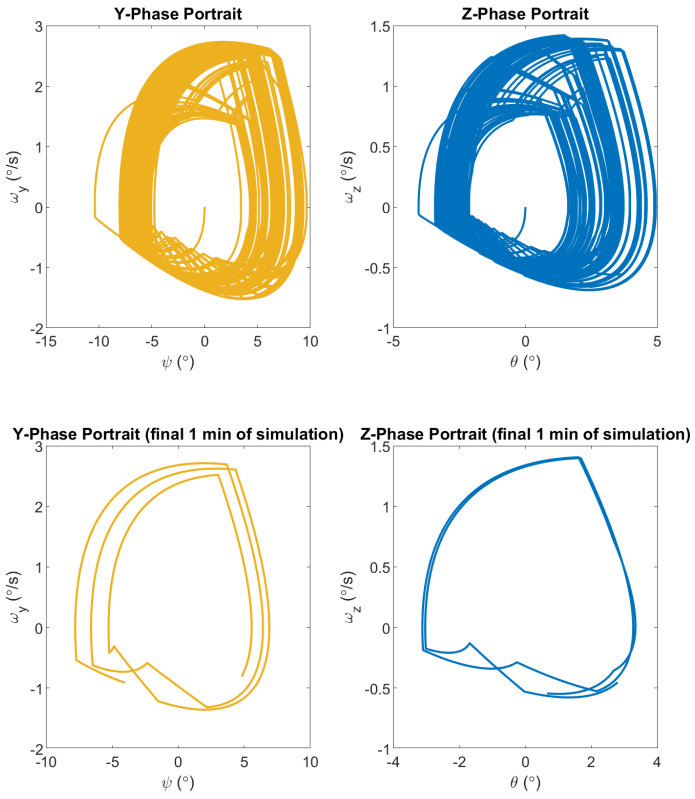


**Figure 20. On-Off Phase Plane: Extended burn**

### 4.2.2 Quaternion Feedback Control

QFC was used as an example of another relatively simple nonlinear control concept. As it was originally established for the ADCS, the controller does not take into account any switching conditions, as seen in SMC and MRAC. This allows for a smooth transition for continuous actuators, such as reaction wheels. Although, as seen in the following figures, the lack of this switching condition leads to a less robust response. What is apparent is that the system finds equilibrium at the upper bound of the $x$ reaction wheel in the algorithm. While this is not necessarily a desired outcome, this at least shows how the algorithm plays a vital role in maintaining stability. This type of response is seen in the other control laws when jitter causes the input signal to diverge. The gains $K$ and $C$ were left to be those found in Eqns. 3.15 and 3.16, respectively.

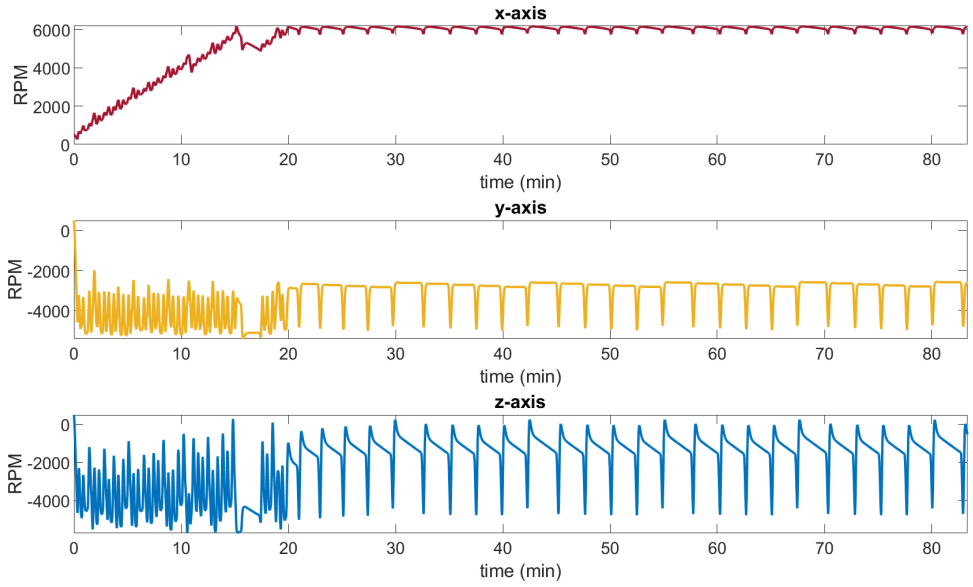


**Figure 21. QFC Reaction Wheels: Extended burn**

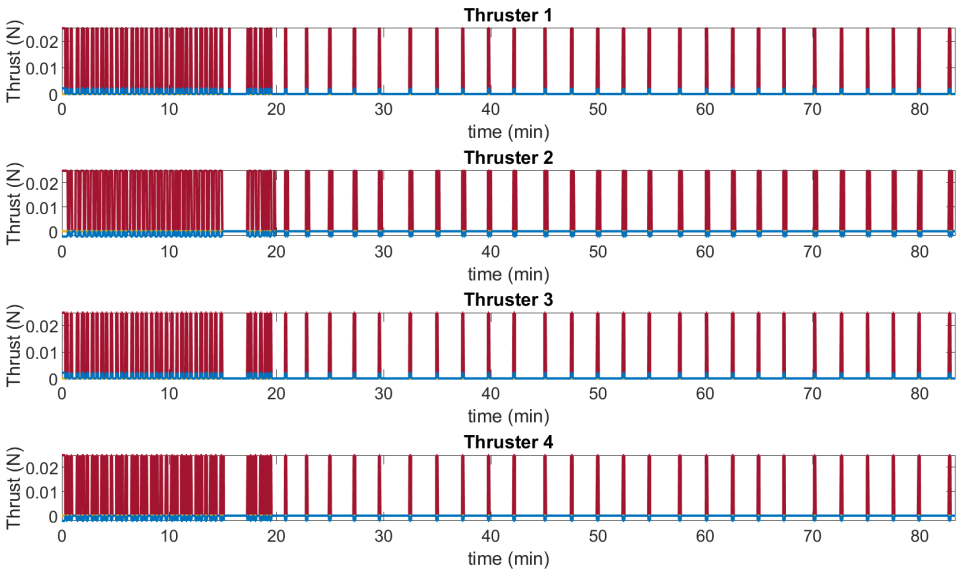


**Figure 22. QFC Thrusters: Extended burn**

The resulting thruster response from the $x$ reaction wheel consistently reaching 6000 RPM yields a pulsing response, where Thruster 2 sees additional thrust. It may be inferred that this is for the same reason Thruster 2 was consistently firing in the On-Off controller; minimal mechanical advantage. Nonetheless, the phase portrait in Fig. 23 shows close to a limit cycle, where the system nearly reaches the equilibrium point at the origin. It can also be seen that the magnitude of error is close to half that of what is seen in Fig. 20. However, when calculating the $\Delta v$ efficiency to be 88.22%, only a marginal increase was observed over On-Off. In the upper subplots showing the full simulation's phase portrait, it does appear that there are two modes the system may converge to, and the last 5 minutes of the simulation only caught this convergence to the origin. Regardless, the system is seen to be stable.

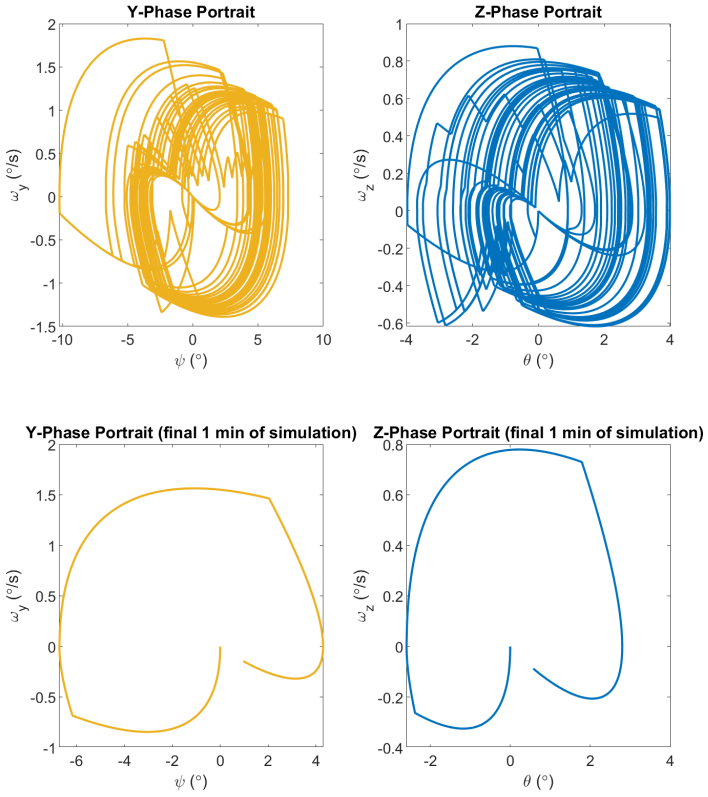


**Figure 23. QFC Phase Plane: Extended burn**

### 4.2.3 Sliding Mode Control ($\vec{\omega}$-Based)

The sliding mode techniques begin to show the advantages of robust control methods. Sliding mode relies on its switching condition to consistently drive the system response to the sliding surface. The positive constant gain, $\vec{\eta}$, is inversely related to the time to reach the sliding surface, while the other gain, $\lambda_S$, is simply a multiplier. When tuning these gains, the reaction wheel response was the primary feedback mechanism. If the reaction wheels were finding an equilibrium state without saturating in $x$, the efficiency was typically higher, and thruster firing was more consistent. If the gains were two low, (below $10^{-2}$), the system would respond similar to the QFC. This intuitively makes sense, as tuning these gains to 0 would essentially

drive the controller swithcing conditions to 0, thus nullifying the intended benefits of SMC. Stable behavior was discovered for SMC-$\vec{\omega}$ at $\lambda_S = 60$, and $\vec{\eta} = [10, 10, 10]^T$.

Using the control law derived in Eqn. 3.29, SMC-$\vec{\omega}$ shows interesting behavior. Figure 24 displays the reaction wheel speeds, where the $y$ and $z$ wheels are fairly stable with oscillations of approximately 1/4 of the magnitude seen in On-Off. The $x$ reaction wheel exhibits a non-uniform response over the course of the maneuver. This is due to the added condition placed on each thruster requiring a +/- command torque for each axis, including $x$. The $x$ reaction wheel slowly climbs from the thruster input but is interrupted with intermediate breaks where a linear drop in RPM is observed. This is due to the conflict condition at the end of Alg. 2. The algorithm will fire all thrusters after the $\Delta v$ has not changed for 100 time steps, so that the system does not dwell in this "stuck" state for a significant amount of the maneuver, but also that it allows the controller to continue as intended.

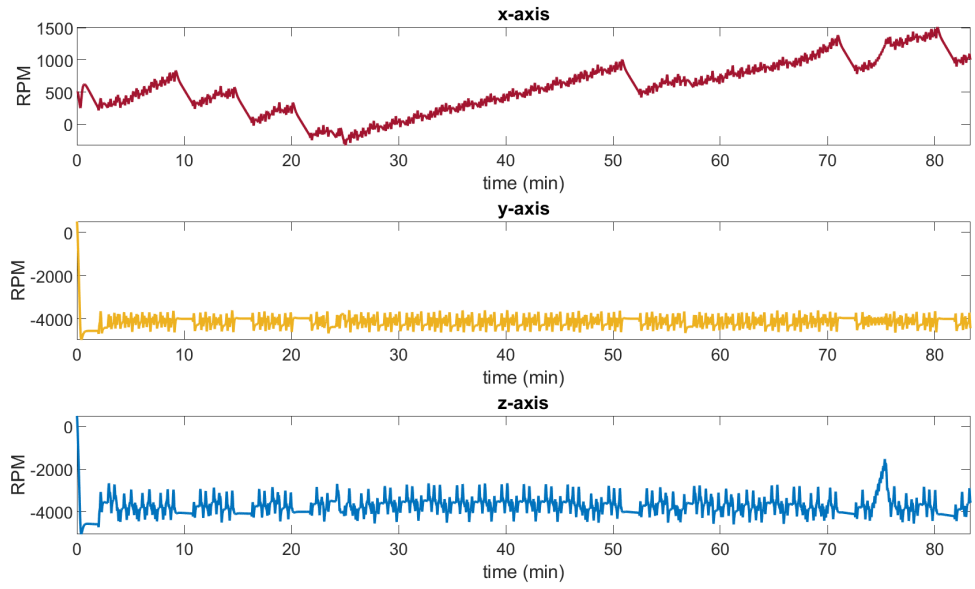


**Figure 24. SMC ($\vec{\omega}$-based) Reaction Wheels: Extended burn**

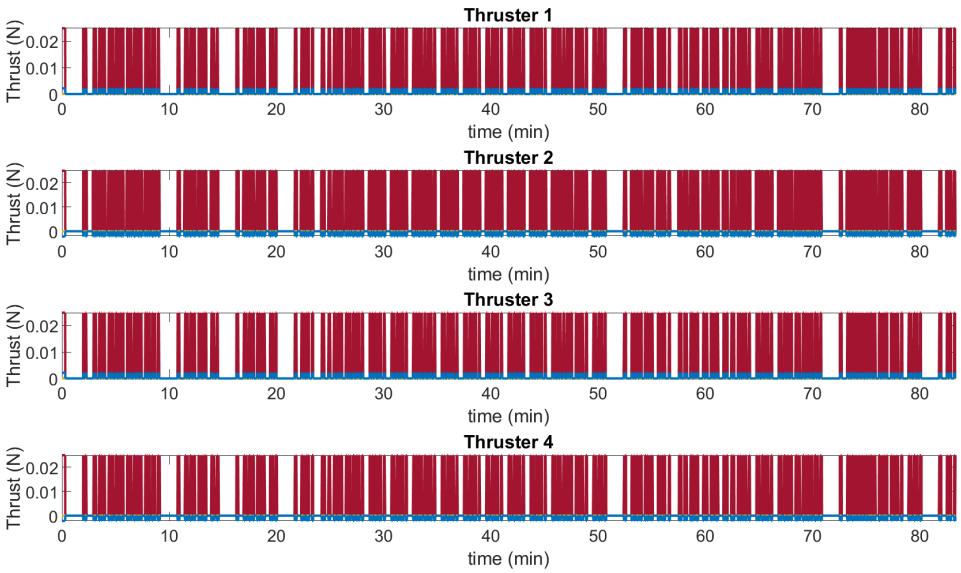


**Figure 25. SMC ($\vec{\omega}$-based) Thrusters: Extended burn**

The phase portraits in Fig. 26 show a few notable things. In the top two subplots, its is evident that the system maintains a local proximity to the origin. However, the final minute of the simulation displays that this is anything but a uniform limit cycle. Typically, a closed-form encirclement of the origin would be seen for a marginally stable system. It appears the dwell times of the thrusters in Fig. 25 lead to two different unique limit cycle-like trajectories that conflict with one another. It may also be that the switching condition is abrupt and irregular enough to increase the variance of the phase plane trajectory in such a haphazard manner. The jitter of the input signal primarily causes this phenomena, but maintaining a path close to the origin over the 5000 sec. maneuver does give confidence that the system will behave appropriately for this application. Even with the irregularities, the deorbit maneuver saw a $\Delta v$ efficiency of 90.076%.

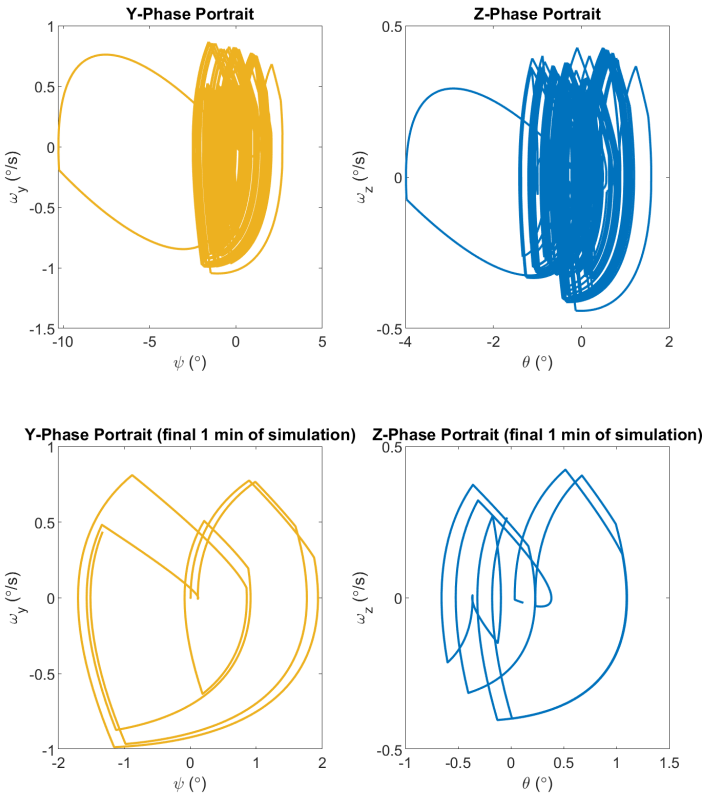


**Figure 26. SMC ($\vec{\omega}$-based) Phase Plane: Extended burn**

Another important aspect of SMC, is verifying the status of the sliding surface. Recall that in Eqn. 3.22, $\vec{s}$ is a function of $\tilde{\omega}$ and $\dot{\tilde{\omega}}$. Both of which must remain close to 0. Therefore it is expected that both $\vec{s}$ and $\dot{\vec{s}}$ will oscillate about 0. This is seen in Figs. 27 and 28. The plots are a slight crop of the original data, as the beginning the simulation with delays resulted in a large spike in the sliding variable. This would quickly resolve within 5 seconds, and the trend of the sliding surface is evident for the rest of the simulation.

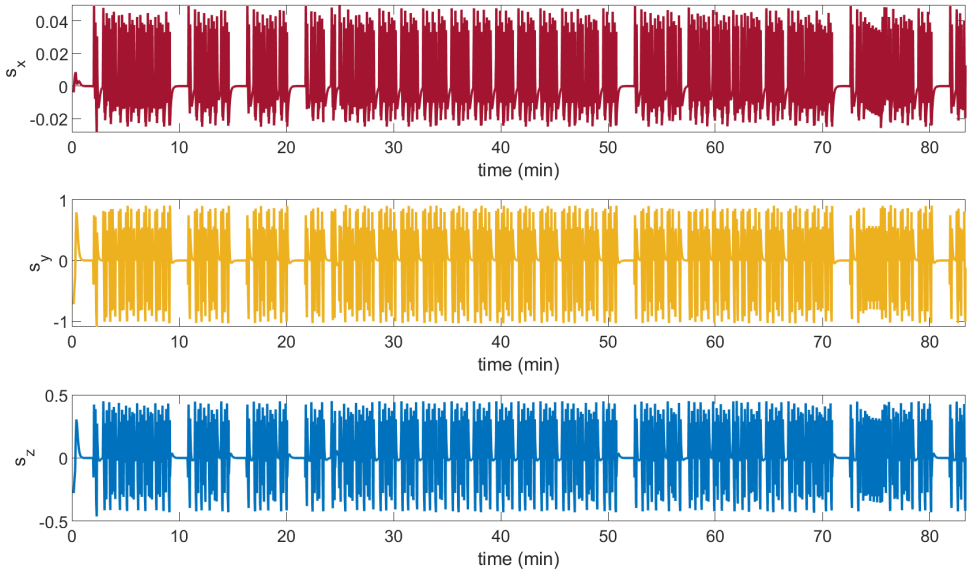


**Figure 27. SMC ($\vec{\omega}$-based) Sliding Surface (zoomed): Extended burn**

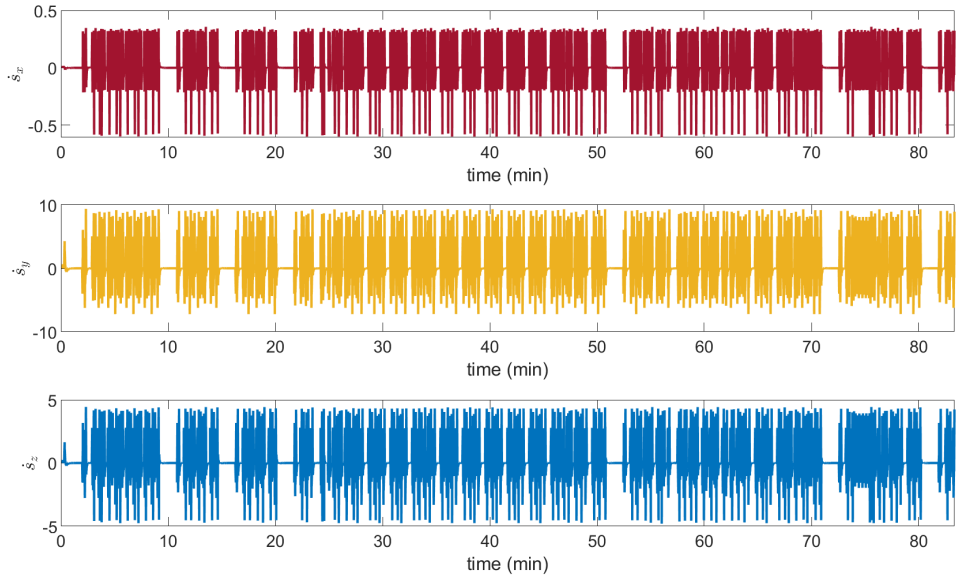


**Figure 28. SMC ($\vec{\omega}$-based) Sliding Surface Derivative (zoomed): Extended burn**

### 4.2.4 Sliding Mode Control (Quaternion-Based)

The quaternion-based SMC takes a slightly different approach to handling the dynamics of the system, and combining the methods discussed in the other SMC design. The same approach to tuning gains was taken for SMC-Q as done for its $\vec{\omega}$ counterpart. The $P$ and $K$ were adjusted until a stable response was seen by the thrusters and resulting reaction wheel behavior. $P$ was selected to be $diag([80, 80, 80])$ while $K$ was tuned to $diag([0.03, 0.03, 0.03])$. Similar to $\lambda_s$ in the previous section, $K$ is inversely related to the rate of convergence [33]. What is interesting, is that $K$ was required to be much lower for the quaternion method, as any higher values to drive the system to behave similar to QFC. This is not a desireable outcome, as it is more reflective of the algorithm than the controller. When $P$ was dialed too low, the thrusters would simply not fire. It appears the balance of these gains are critical to the performance of the system. More shall be discussed in the next section on this topic when discussing the Monte Carlo.

Interestingly, the SMC models do not behave similarly to one another. As seen in Fig. 29, the reaction wheels behave erratically in response to the thrusters. It is notable that the $y$ wheel maintains its RPM, and what appears to be oscillations in the $x$ and $z$ wheels without reaching the maneuver maximum. Figure 30 displays the thrusters firing without a noticeable pattern, but much more infrequent than SMC-$\vec{\omega}$'s response. This is most likely due to the difference in gain magnitude, relative to the time to reach the sliding surface. It may be inferred that the SMC-Q's much smaller gain relates to a slower rise time, which explains this behavior.

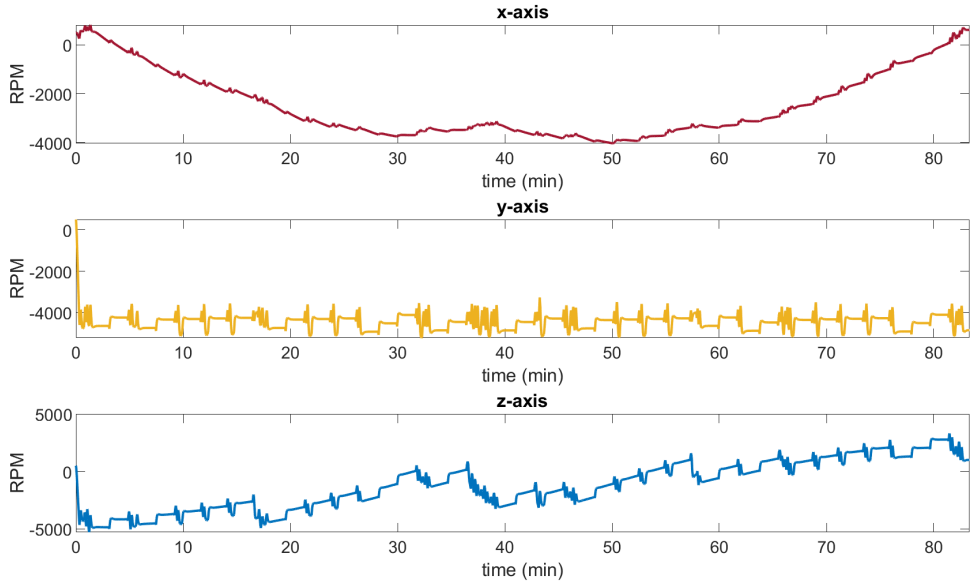


**Figure 29. SMC (Quaternion-based) Reaction Wheels: Extended burn**

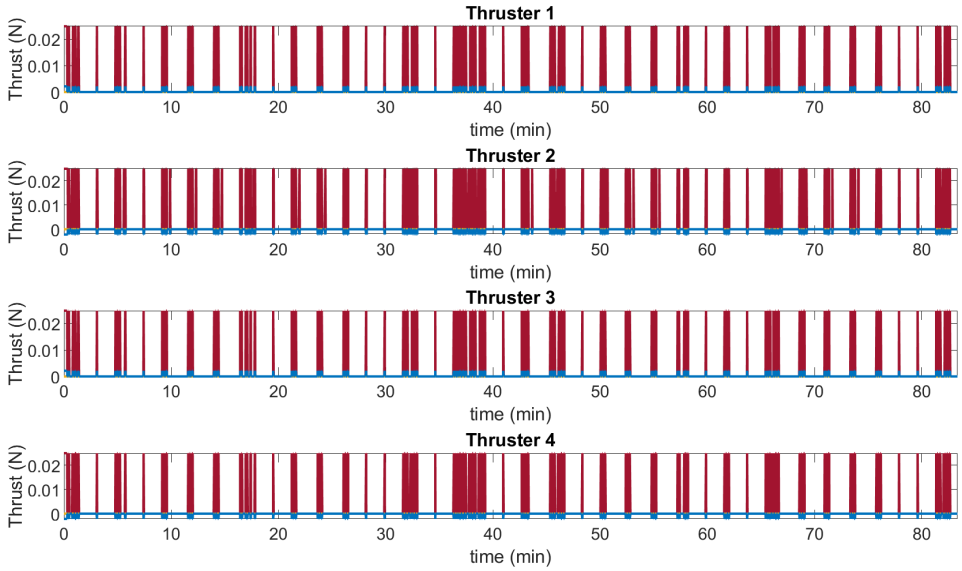


**Figure 30. SMC (Quaternion-based) Thrusters: Extended burn**

The phase portraits in Fig. 31 do show somewhat similar behavior to SMC-$\vec{\omega}$'s. In the final minute of the simulation, both trajectories show a path that does not close within one encirclement. The full simulation subplots do show a clear pattern exists about the origin, indicating a marginally stable system. As with SMC-$\vec{\omega}$, Lyapunov stability may not be claimed with confidence, but there is no evidence that this system is unstable. With this more dispersed thrust method, the efficiency seen in the deorbit maneuver was roughly 89.242%. With the sliding surface being driven to 0, it can be seen that the system does respond appropriately in Figs. 32 and 33.

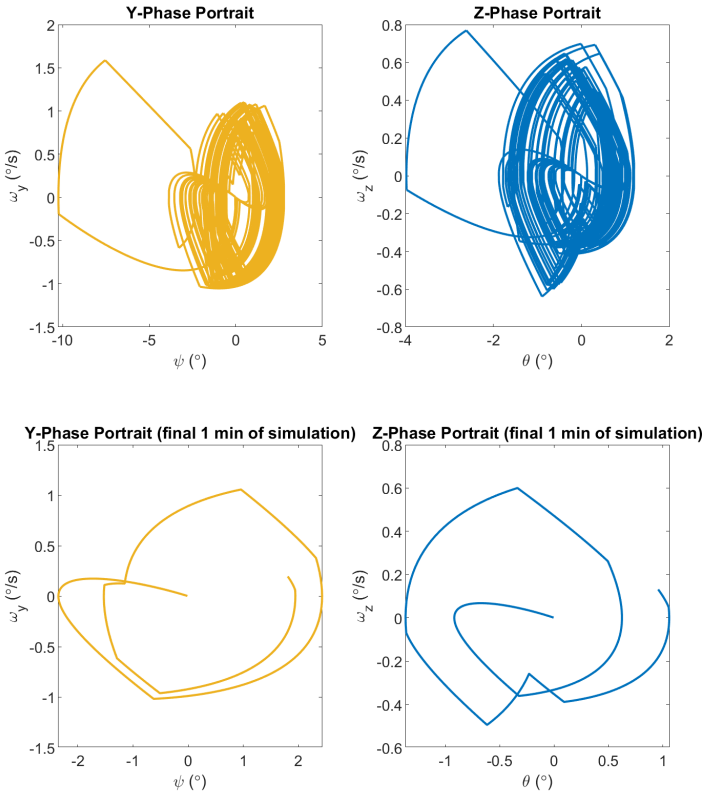


**Figure 31. SMC (Quaternion-based) Phase Plane: Extended burn**

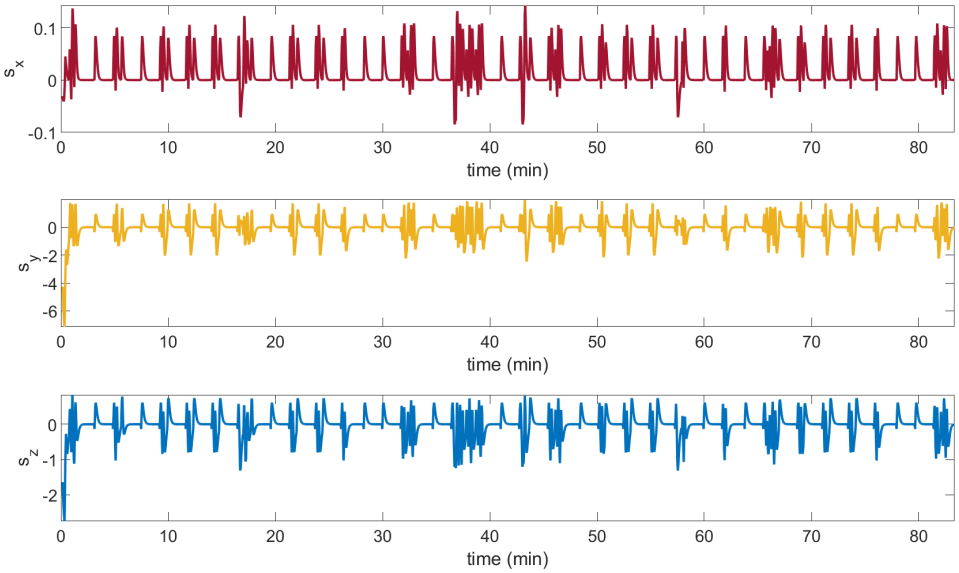


**Figure 32. SMC (Quaternion-based) Sliding Surface (zoomed): Extended burn**

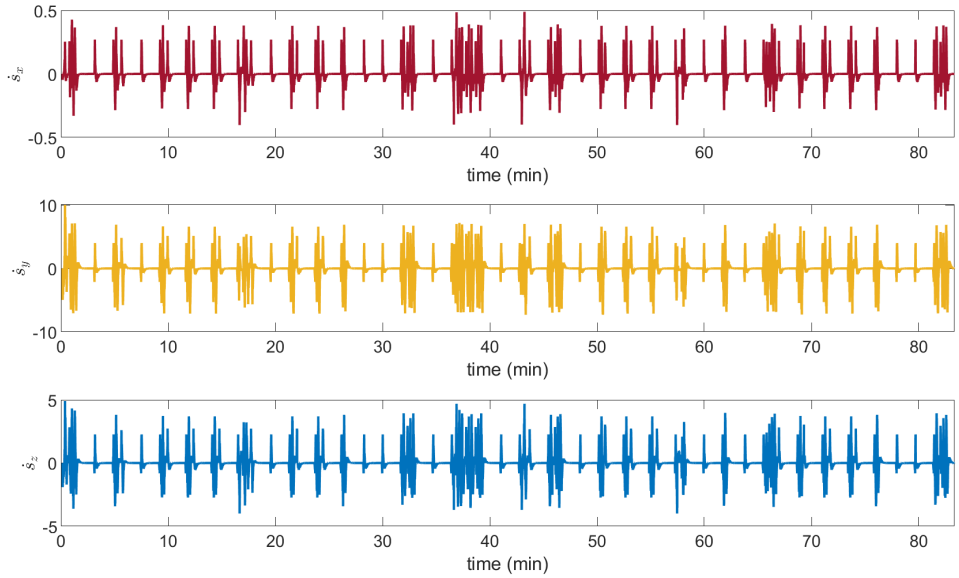


**Figure 33. SMC (Quaternion-based) Sliding Surface Derivative (zoomed): Extended burn**

### 4.2.5 Model Reference Adaptive Control

The MRAC method uses two adaptation parameters, one for the uncertainty in the dynamics/latency of the system, and another accounting for the ADCS control law, which is unknown to the propulsion controller. There are three gains to that require tuning for this MRAC configuration. $\lambda_M$ represents performance specifications of the assumed first order system [4]. This was adjusted first to ensure the system response did not yield large gaps between thrusts, as well as a somewhat consistent response given the uncertainty of the signal. $\gamma$ is the gain that determines the "weight" of the adaptation parameters, and was adjusted until the $y$ and $z$ reaction wheels exhibited stable characteristics. Finally, $k$ was adjusted to fine tune the switching ability, similar to the SMC gain, $\lambda_s$. For this system, these values were found to be: $\lambda_M = 0.49$, $\gamma = 200$, $k = 100$.

It is notable that the MRAC reaction wheels are being driven to the algorithmic saturation points. While this true for a maneuver requiring as much time as this simulation. The same characteristic response of the algorithm limits can be seen at approximately the 70 minute mark, where MRAC begins to look closer to QFC than its original thrust sequence. However, the stable ascent of the $x$ reaction wheel takes enough time, that this may be considered acceptable, yielding a unique response. Of all the methods thus far, the thruster sequence in Fig. 35 cycles the thrusters the most. What is also unique is in the $z$ reaction wheel experiencing a decrease in absolute RPM before returning to the desired speed. It is unclear why this phenomena is occurring, but it may be inferred that it is due to the adaptation parameters adjusting as the simulation progresses.

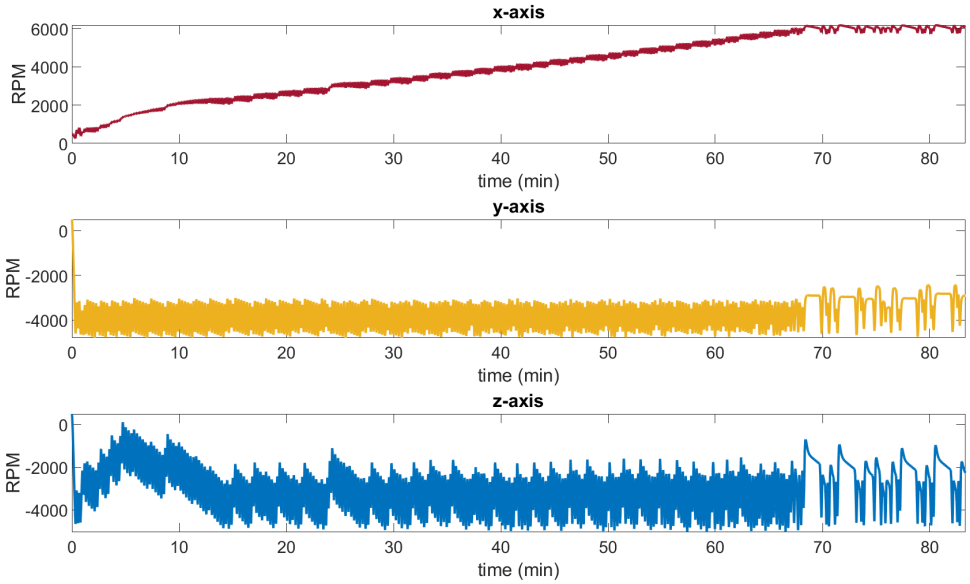


**Figure 34. MRAC Reaction Wheels: Extended burn**

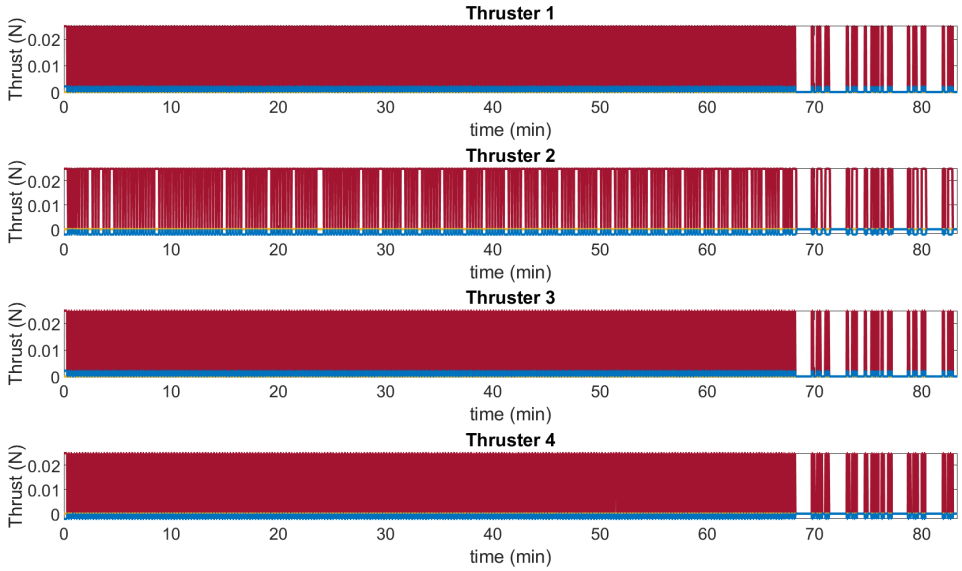


**Figure 35. MRAC Thrusters: Extended burn**

The phase portraits in Fig. 36 show a more desirable form than what was seen in the SMC methods. While the loop still does not follow an explicit limit cycle, the general shape is prominent throughout the simulation, and is highlighted in the lower subplots. The adaptations parameters, shown in Figs. 37 and 38 show a slight adjustment overtime, that may be considered negligible. Like the sliding surfaces in SMC, there was an initial spike in both $\hat{a}_1$ and $\hat{a}_2$ at the beginning of the simulation due to initial discontinuities from the delays. The system then achieves this seemingly equilibrium state immediately after this spike.

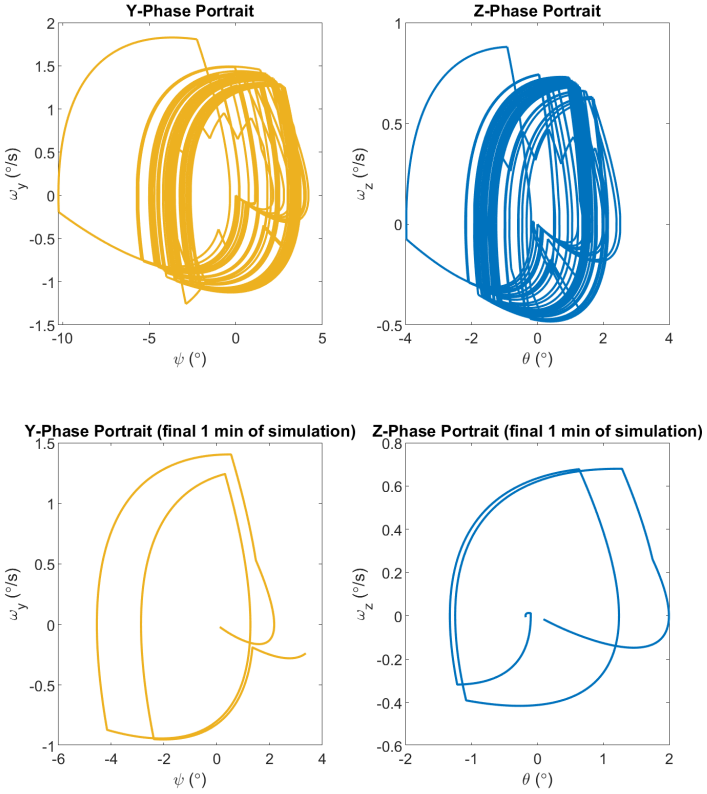


**Figure 36. MRAC Phase Plane: 1.5 m/s burn**

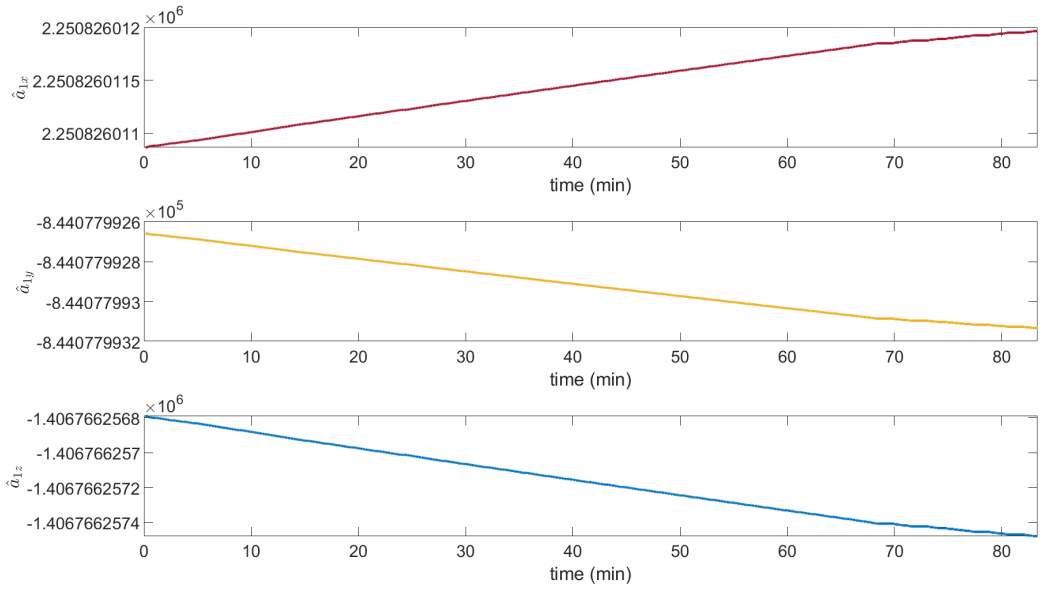


**Figure 37. MRAC Adaptation Parameter (body mechanics - zoomed): Extended burn**

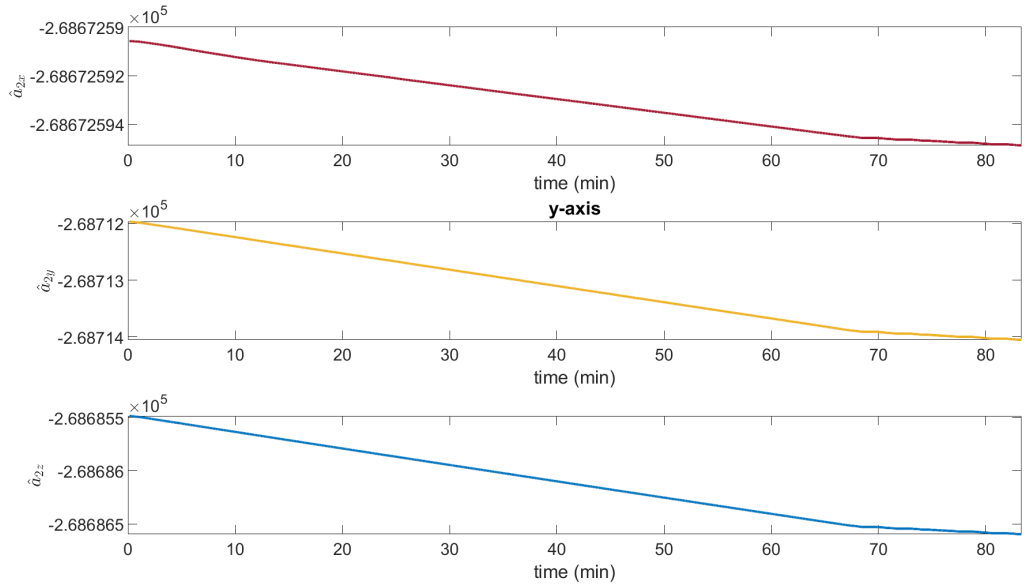


**Figure 38. MRAC Adaptation Parameter (ADCS Control - zoomed): Extended burn**

## 4.3 Case Study: Phase Change Maneuver

A phase change is a standard orbital maneuver. The objective of the maneuver is to execute a relatively small $\Delta v$ in either the velocity ($+x_O$), or negative velocity ($-x_O$) direction to enter what is essentially a "drifting" orbit. This is done with the intent to rotate the spacecraft 180 degrees and perform the same $\Delta v$ in the opposite direction after a specific amount of dwell time to achieve a change in true anomaly [34]. The amount of $\Delta v$ typically required for these maneuvers provides a use case for thruster modulation. As discussed in this section, a 1.5 m/s $\Delta v$ is conducted for each control method for comparative analysis.

### 4.3.1 Baseline Control

The baseline control method plays a significant role in this paper. It exemplifies how extreme this scenario can be for a simple translational maneuver. As the controller fires all thrusters, the reaction wheels quickly saturate due to their offset. Where typical propulsion systems for this use case by either adding more thrusters [35], or gimballing the primary thruster [36]. Both cases require additional hardware, mass, and uncertainty simply to achieve stable sustained $\Delta v$. If no solutions were implemented, the maneuver would be possible, but the time required may not fit within mission requirements.

Consider the firing sequence in Fig. 39 for a 1.5 m/s $\Delta v$ maneuver. Each spike in the plots represent each pulse of the thrusters before a reaction wheel hits an allowable maneuver maximum. Once that limit is reached, as seen in Fig. 40, the spacecraft enters a "Detumble Mode" where the magnetorquers use the Earth's magnetosphere to dump momentum. While the wheels desaturate, the thrusters are unable to fire, resulting in long dwell times of approximately 45 minutes. This is assuming maximum average magnetic torque from the magnetorquers being ideally applied.

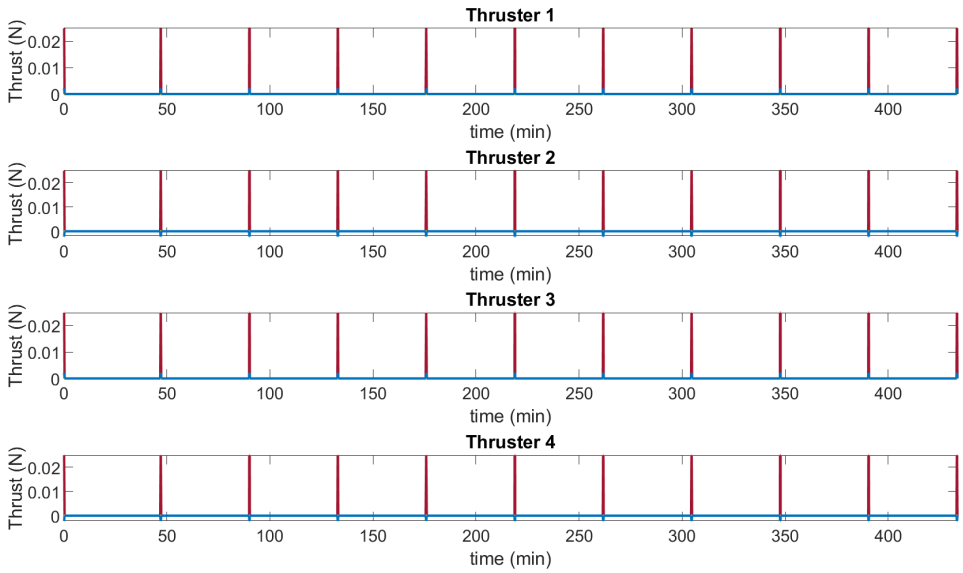


**Figure 39. Baseline Thrusters: 1.5 m/s burn**

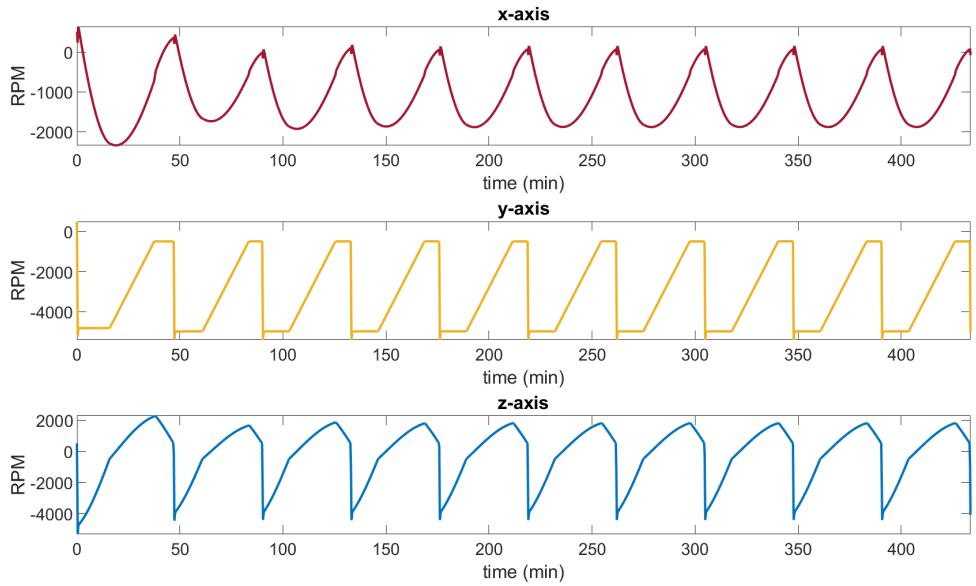


**Figure 40. Baseline Reaction Wheels: 1.5 m/s burn**

This behavior is also reflected in the reaction wheel response. It is seen in Fig. 40 that the $y$ and $z$ axes appear nearly vertical when the thrusters are firing. The $y$ axis sees a linear return to nominal speed of 500 RPM, as $Y$ and $y_O$ are always aligned for this orbital path. $x$ and $z$ see oscillations due to the orbital frame slowly rotating about $y$, resulting in the angular momentum vector changing with respect to $x_O$ and $z_O$.

The Baseline Control method yields a maneuver time of 8 hours, which is roughly 5 orbital periods for a LEO satellite. While there is nothing explicitly preventing this long of a maneuver, a primary benefit of cold gas thrusters over its competitors is relatively high thrust for low $\Delta v$ times. At 8 hours, the Baseline loses much of this draw. Then when considering the growing demand for agile small satellites for rendezvous and proximity operations, the need for thruster modulation is clear. This notion is supported further by a relatively low $\Delta v$ efficiency of 84.97%. This is lower than each method saw during the deorbit maneuver, so therefore it is expected to be the lowest during this maneuver as well.

### 4.3.2 On-Off Control

On-Off control showed among the most stable responses analyzed in the deorbit maneuver. The response seen in Figs. 41 and 42 essentially gives a zoomed-in view of the general thruster sequence that occurs strictly due to the On-Off algorithm. As expected, Thruster 2 is consistently firing, while the others are modulated. Thrusters 1, 3 and 4 do not fire in any particular pattern as observed in this simulation, due to the random jitter. The reaction wheels see oscillations as the system attempts to reach equilibrium, while their maximum absolute value does not reach the maneuver maximum of 6000 RPM.

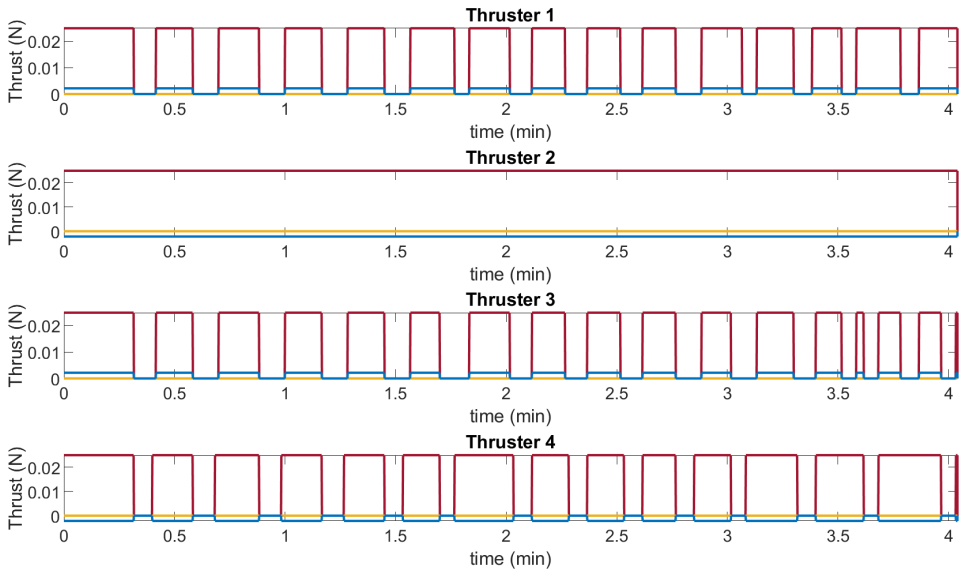


**Figure 41. On-Off Thrusters: 1.5 m/s burn**

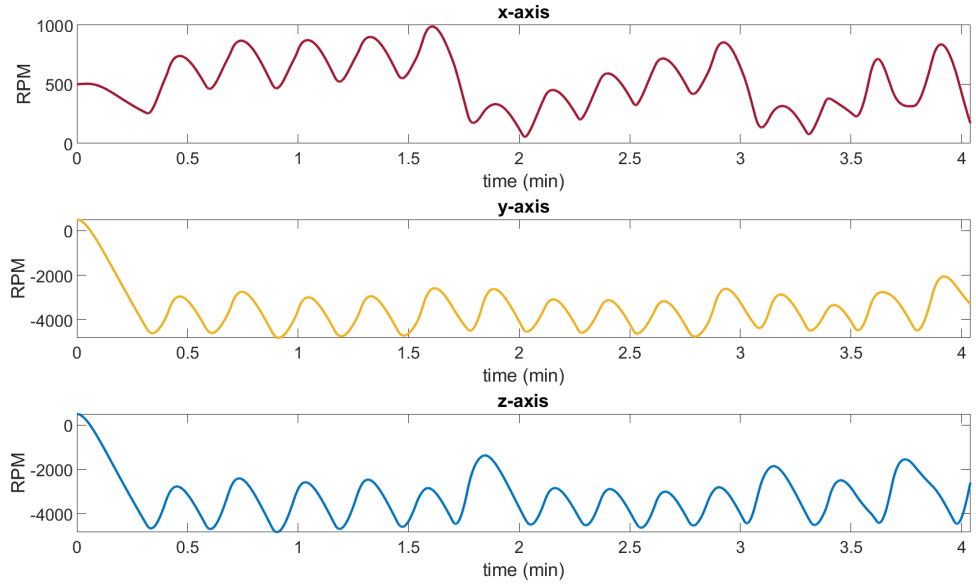


**Figure 42. On-Off Reaction Wheels: 1.5 m/s burn**

The Monte Carlo simulation shown was conducted for 10,000 simulations where mass ranged from 10-12 kg, and the moments of inertia ranged by ± 20%. While 6U CubeSats have a maximum mass requirement of 12 kg [5], the VACCO propulsion system may use upwards of 1.3 kg of propellant [28]. As a particular 6U CubeSat design may not utilize the full mass allowance, the mass variance was rounded up to 2 kg. The moments of inertia may also vary based on the CubeSat design, so a 20% range was chosen to satisfy a significant spread of possible configurations.

The Monte Carlo simulation results are displayed in Figs. 43 through 44. Upon visual inspection, obvious bimodality is seen in thruster efficiency. This is further supported through statistical bimodality analysis, by determining a Bimodality Coefficient, BC, from the skewness and kurtosis observed [37]. The BC for this distribution was found to be 0.63, exceeding the 0.555 criteria needed to distinguish bimodality. Upon further inspection of the model, this bimodal relationship is not due to variance in the physical properties, but again due to the impact of inconsistent feedback frequency. Figure 45 displays the reaction wheels of a simulation with the same initial parameters as in Fig. 42. Notice that the $x$ RPM steadily decreases in this second simulation versus the oscillatory pattern seen in the first. This behavior is triggered if the controller favors the $-x$ thrusters, 1 and 4, and becomes a seemingly unstable tendency. However, due to the flag for the $x$ reaction wheel in the first "if" statement of Alg. 1, the system would return to a stable response. Therefore, the algorithm remains stable, but the optimal thrust pattern may not be enforced without further manipulation of the algorithm.

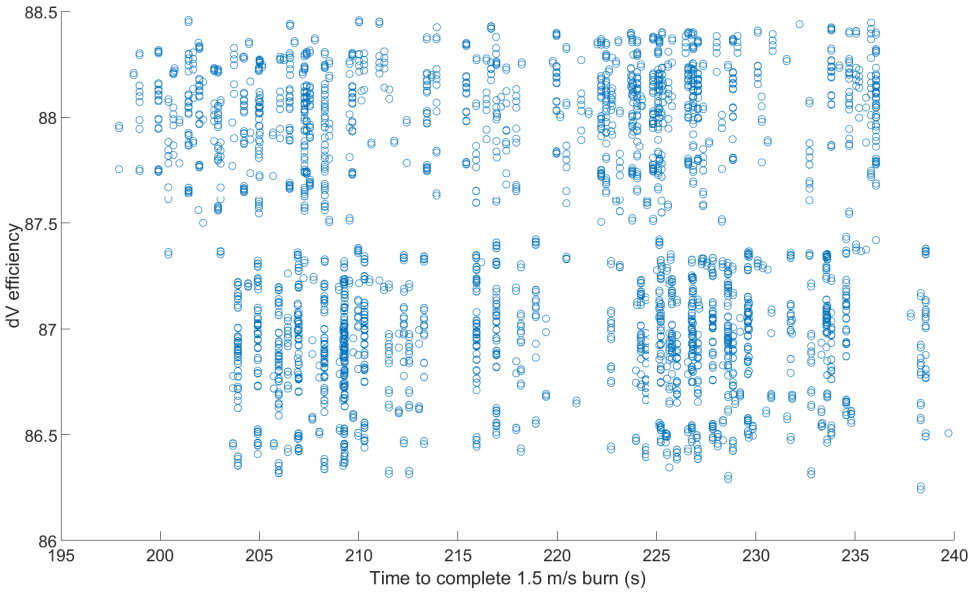


**Figure 43. On-Off Control Monte Carlo Analysis: Efficiency versus Maneuver Time**

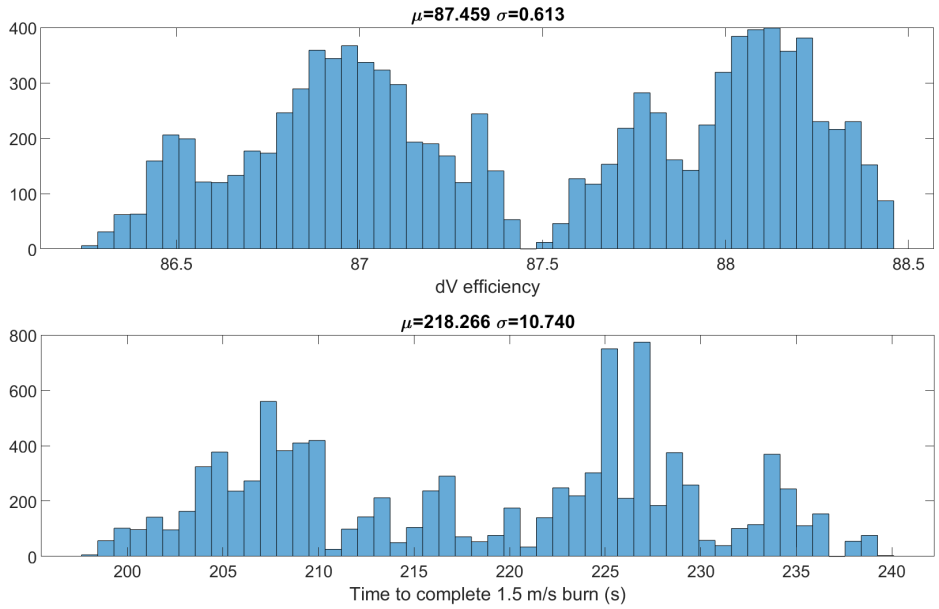


**Figure 44. On-Off Control Monte Carlo Analysis: Time and Efficiency Distributions**

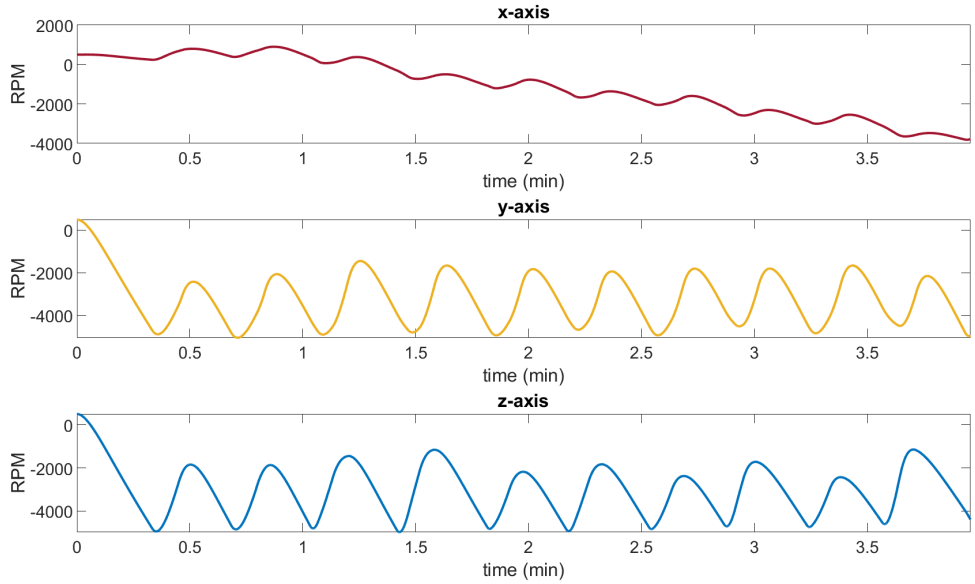


**Figure 45. On-Off Wheels: 1.5 m/s burn (secondary mode)**

Consider the two modes as separate analyses. The lower distribution may be seen below, where the upper bound of the efficiency data was selected via the average of the the complete data set, 87.459%. The same indexed data points for the maneuver time were selected to recreate the histogram seen in Fig. 46. While only a slight efficiency increase is seen in this mode over the Baseline control method, the resulting average time to complete the maneuver was only 219.348 seconds.

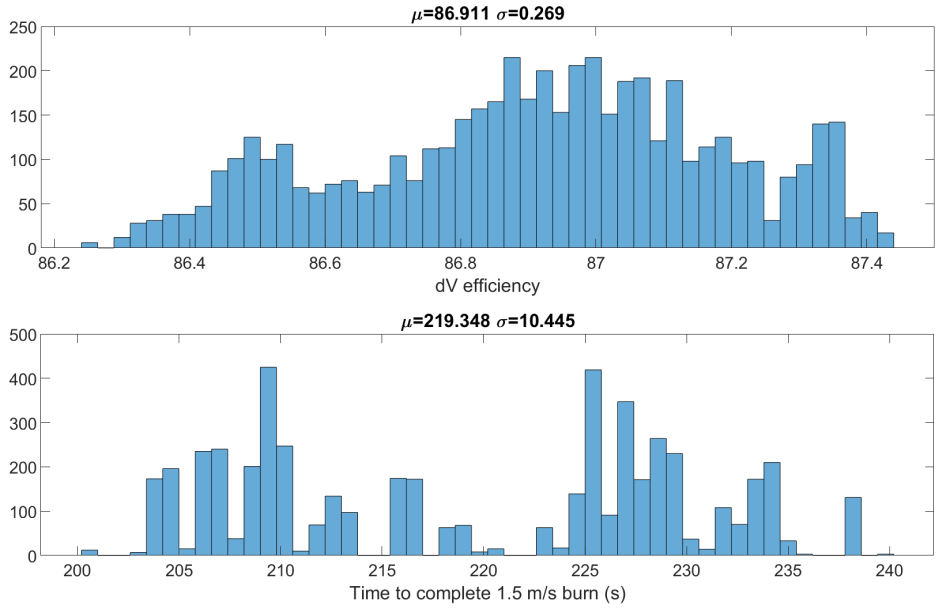


**Figure 46. On-Off Control Monte Carlo Analysis: Time and Efficiency Distributions (lower distribution)**

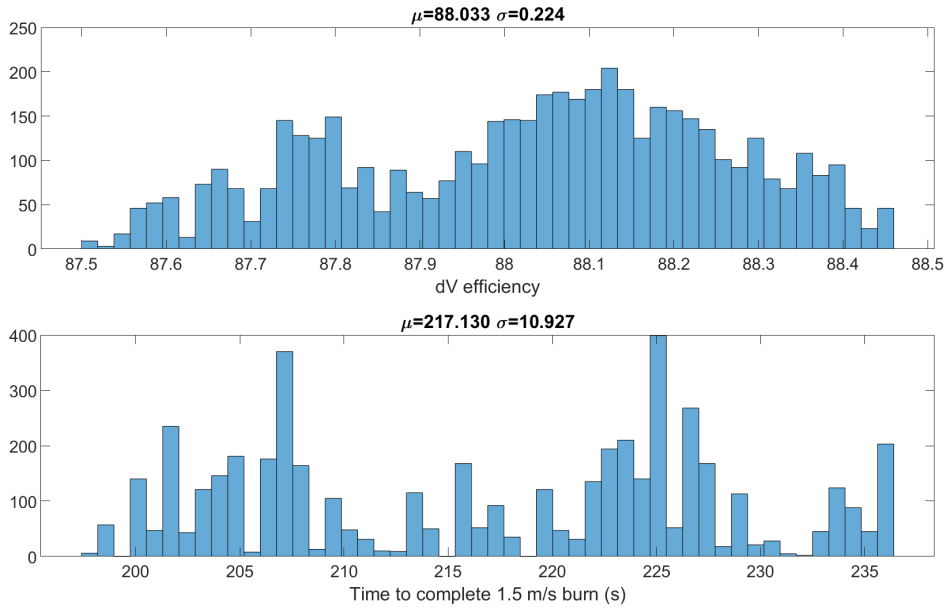


**Figure 47. On-Off Control Monte Carlo Analysis: Time and Efficiency Distributions (higher distribution)**

The higher distribution of the entire data set is shown in Fig. 47. This mode yields an improved efficiency of 88.028%, and lower maneuver time of 217.13 seconds. It can be seen when comparing Figs. 46 and 47 that there is not a notable correlation between time to complete the maneuver, and thruster efficiency

Even considering the uncertainty of results due to the bimodal tendencies of the On-Off control algorithm, significant improvements over the Baseline control method are evident. An improvement of either 2% or 3% can be expected in $\Delta v$ efficiency, while the time to complete the maneuver is consistently between 200 and 240 seconds. On-Off control proves that even a simple algorithm in this configuration provides a significant benefit, and establishes a solidified foundation to allow the nonlinear controllers to operate.

### 4.3.3 Quaternion Feedback Control

The QFC method in the deorbit maneuver did not perform well due to the $x$ reaction wheel quickly hitting the maximum allowable RPM. In the phase-change maneuver, the same behavior trend can be seen, but not enough time passes to reach the same state. What is evident in this simulation is the thruster pattern in Fig. 48, where Thruster 2 provides the most thrust, but unlike On-Off, is now modulated. It also appears that Thruster 4 is fired more frequently than Thrusters 1 or 3. As discussed in the On-Off bimodality, this is likely the cause of the steady increase in $x$ reaction wheel.

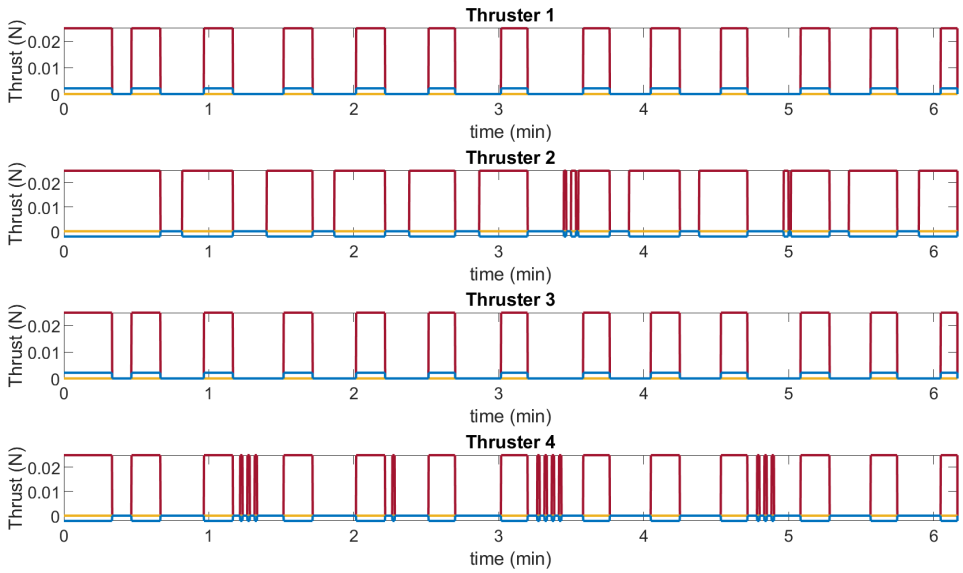


**Figure 48. QFC Thrusters: 1.5 m/s burn**

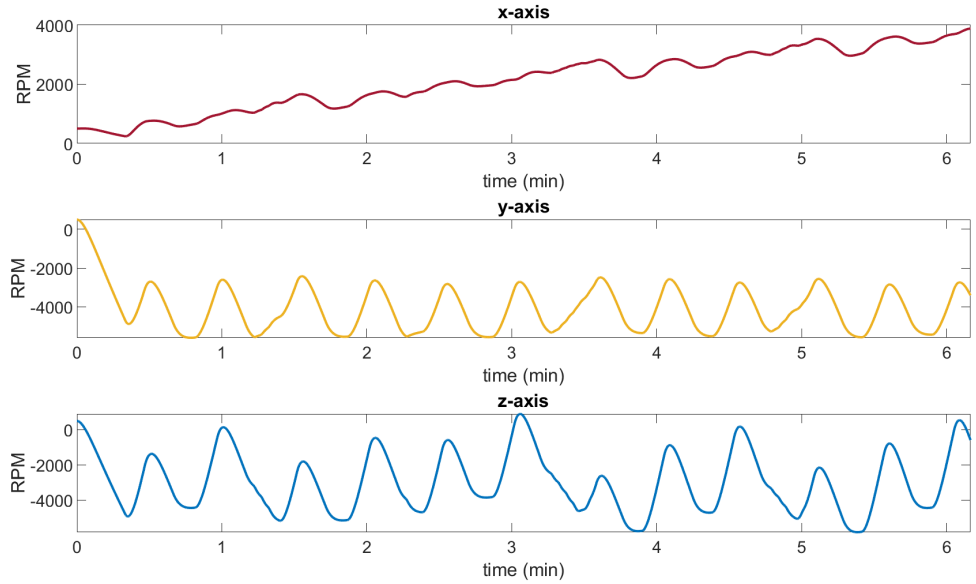


**Figure 49. QFC Reaction Wheels: 1.5 m/s burn**

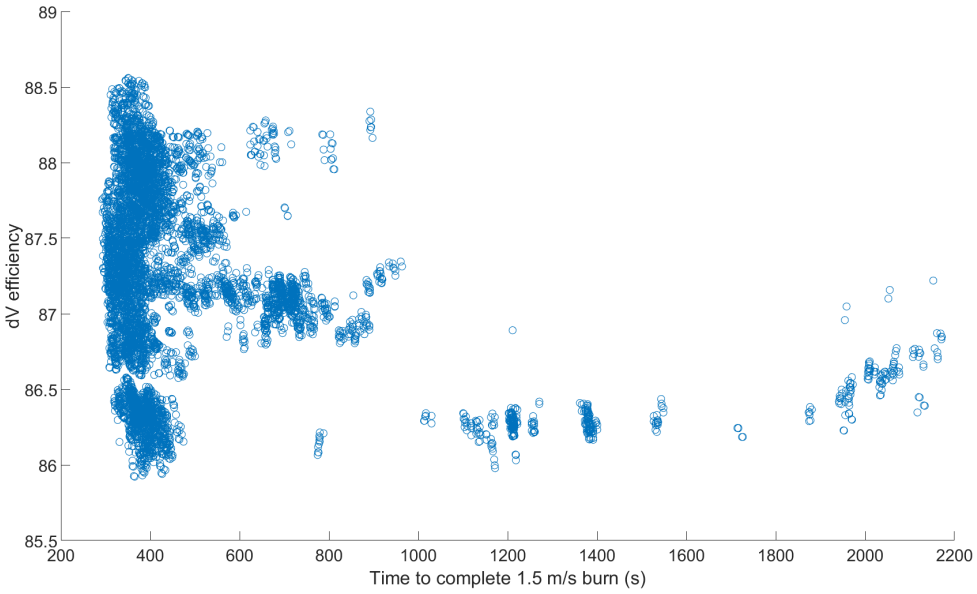


**Figure 50. QFC Monte Carlo Analysis: Efficiency versus Maneuver Time**

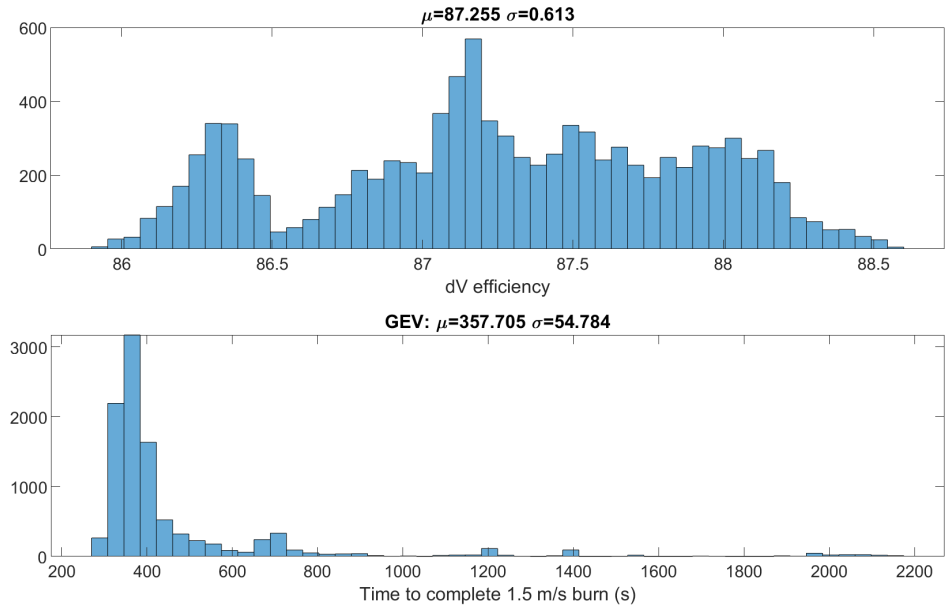


**Figure 51. QFC Monte Carlo Analysis: Time and Efficiency Distributions**

QFC shows an approximately normal distribution of $\Delta v$ efficiencies for the 10,000 simulations in Fig. 51, while also showing a skewed distribution in maneuver time. As discussed in Ch. 3, Alg. 2 tends to hang up when the $y$ reaction wheel is below the desired RPM, and has a commanded torque of $-x$, $\pm y$, and $\mp z$. This condition is met often enough with some parameters, that the $\Delta v$ check loop is entered multiple times, resulting in a longer maneuver.

This method resulted with an average efficiency of 87.225% and Generalized Extreme Value average of 357.705 seconds. This is a marginal improvement over On-Off, thus making it an excellent alternative compared to the Baseline Control Method. Although, the outlier simulations resulting in 800+ seconds to achieve the $\Delta v$ are cause for concern. To further mitigate this behavior, adjustment of the time-step counter within the algorithm should be tuned to minimize the maneuver time for a specific vehicle.

### 4.3.4 Sliding Mode Control ($\vec{\omega}$-based)

SMC-$\vec{\omega}$ adds another layer of complexity to the system with the sliding surface. From the deorbit maneuver, it was also observed that this method had the tendency to stop thrusting due to the algorithm getting stuck. This is seen again immediately after the initialization burn, as well as near the end of the maneuver. This is an undesirable property of the control law, and shows the $\Delta v$ check solution is stable.

The Monte Carlo analysis shows a normal distribution in efficiency, and skewed distribution in maneuver time to the left. With a small number of outlying simulations that took upwards of 2200 seconds, these outliers are a direct result of the system consistently reaching the algorithm's failure point. Each time the system stops firing, 100 time steps must pass, slowly but surely adding a significant amount of time to complete the $\Delta v$.

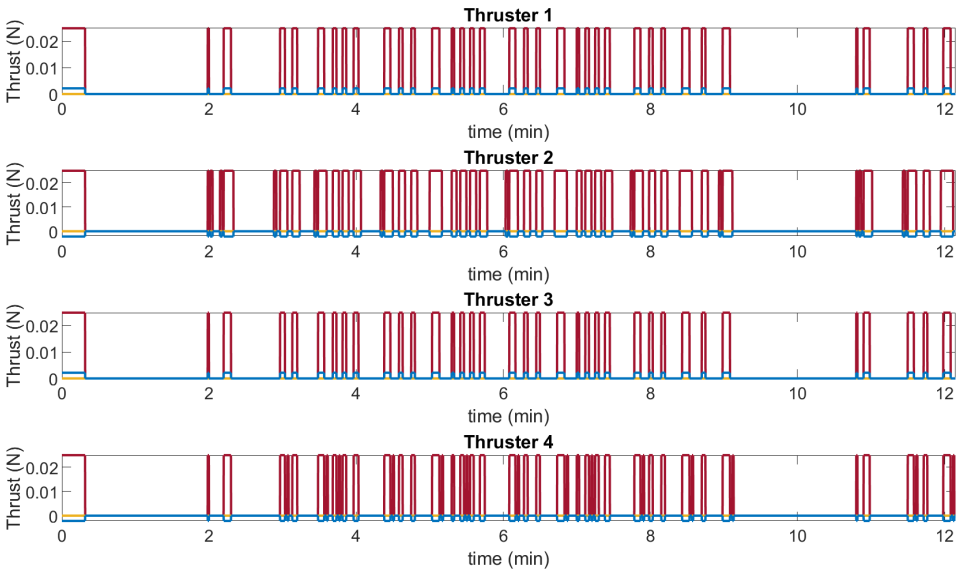


**Figure 52. SMC ($\vec{\omega}$-based) Thrusters: 1.5 m/s burn**

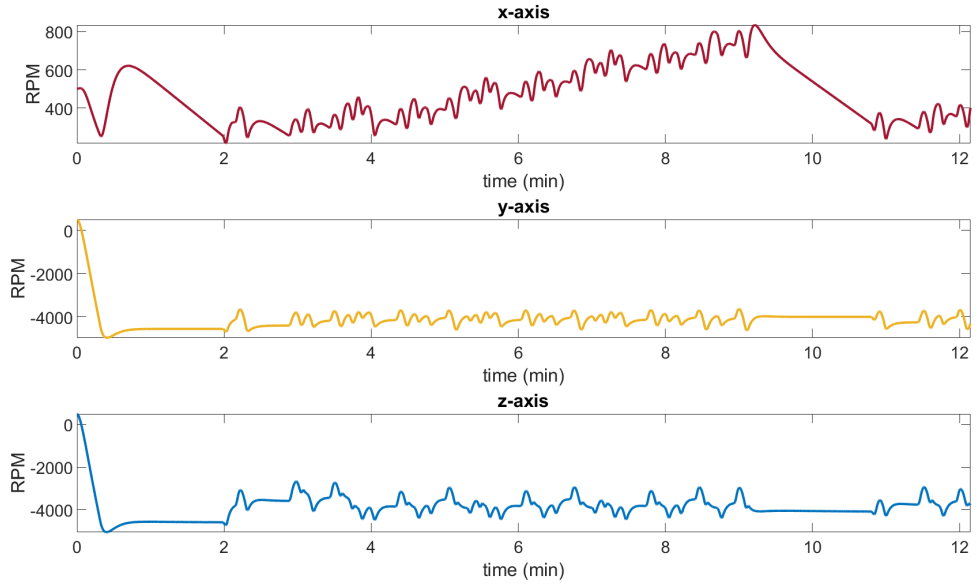


**Figure 53. SMC ($\vec{\omega}$-based) Reaction Wheels: 1.5 m/s burn**

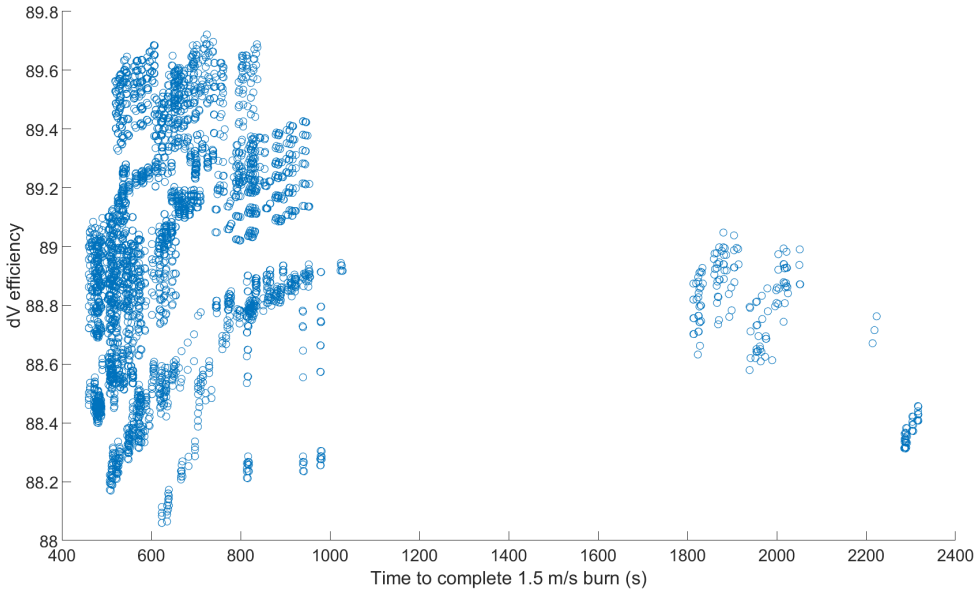


**Figure 54. SMC-$\vec{\omega}$ Monte Carlo Analysis: Efficiency versus Maneuver Time**

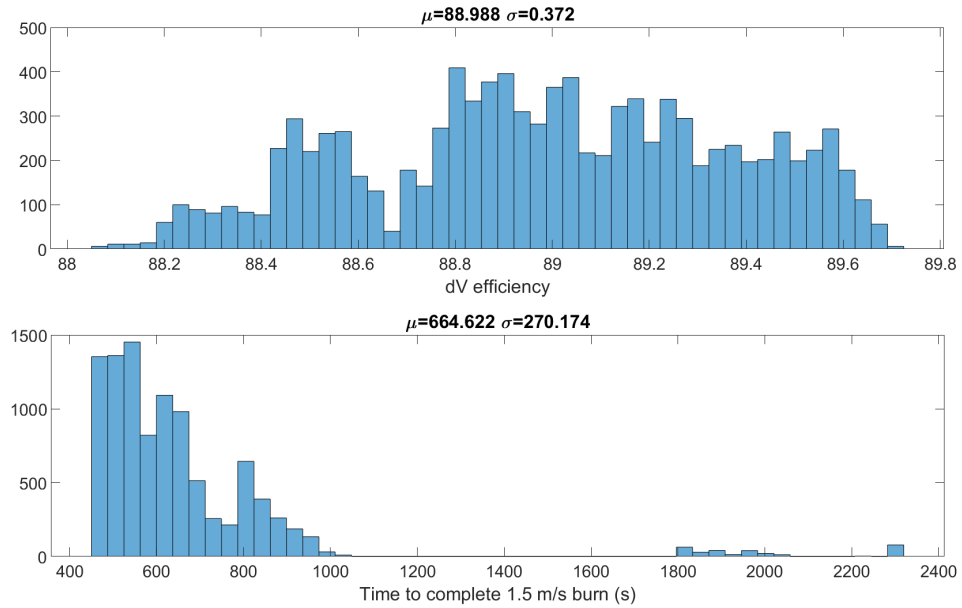


**Figure 55. SMC-$\vec{\omega}$ Monte Carlo Analysis: Time and Efficiency Distributions**

Efficiency for SMC-$\vec{\omega}$ was found to have a mean of 88.98%, besting the On-Off control by nearly a whole percent. The Generalized Extreme Value distribution was found to be the best fit for the maneuver time, resulting in an adjusted average of 548.443 seconds. While this is nearly double the time of the On-Off controller, SMC-$\vec{\omega}$ provides a high efficiency solution. Similar to QFC, the algorithm hang-ups are to be considered with caution. The drawback for SMC-$\vec{\omega}$, is that these hangups occur regardless of the satellite's physical properties. Minimizing the jitter in the system is required to mitigate the adverse effect.

### 4.3.5 Sliding Mode Control (Quaternion-Based)

From the deorbit maneuver simulation, it was discovered that SMC-Q would respond in two different fashions due to the signal irregularity. Shown in Fig. 29 was the mode that showed a unique result, while maintaining a stable response with high efficiency. This will be referred to as "mode 1" in this section. Mode 2, showed a response very similar to SMC-$\vec{\omega}$, where the thrusters would fire for a matter of minutes before hitting the algorithm's fault zone, as seen in Fig. 57. Upon further investigation, the behavior seen in Figs. 29 and 59 was due to repeatedly hitting this condition.

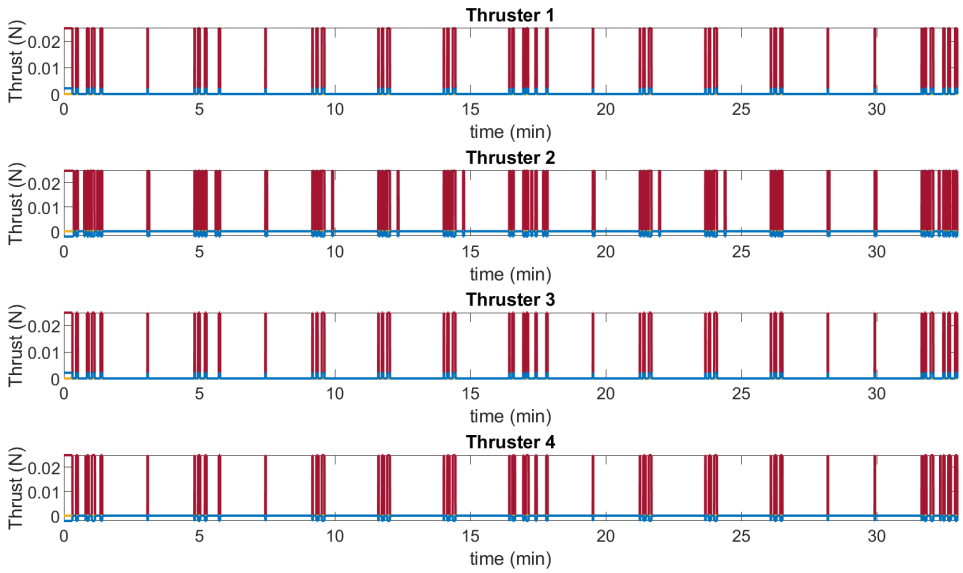


**Figure 56. SMC (Quaternion-based) Thrusters: 1.5 m/s burn (mode 1)**

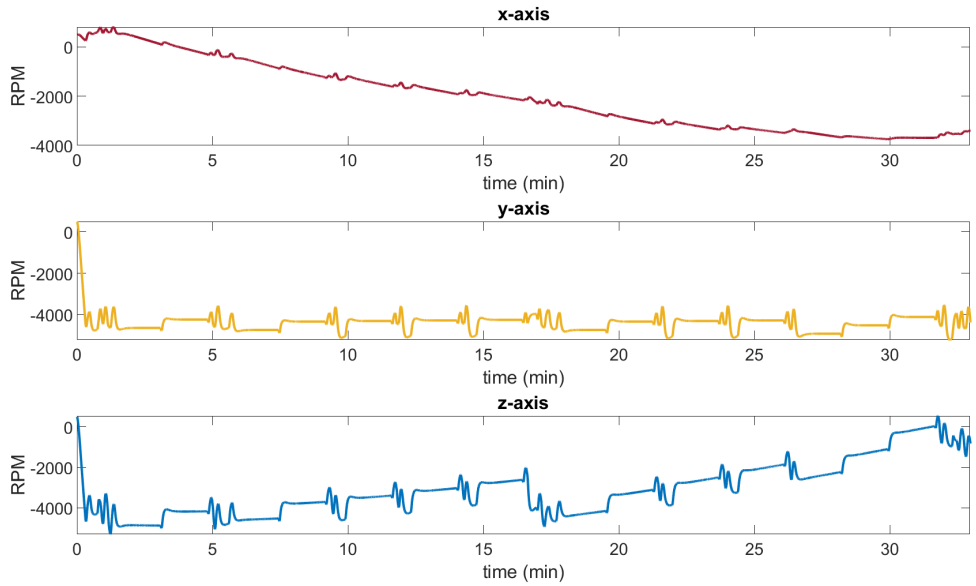


**Figure 57. SMC (Quaternion-based) Reaction Wheels: 1.5 m/s burn (mode 1)**

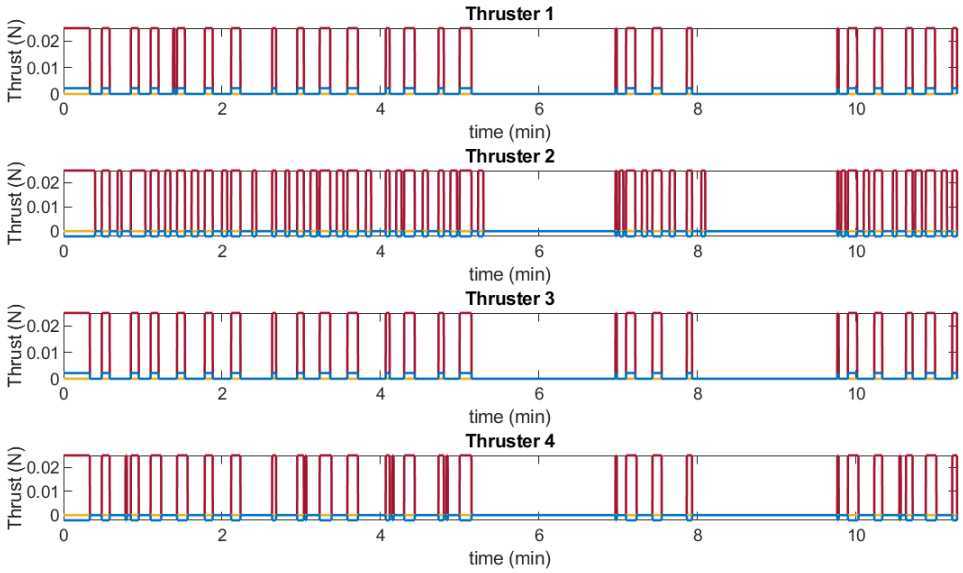


**Figure 58. SMC (Quaternion-based) Thrusters: 1.5 m/s burn (mode 2)**

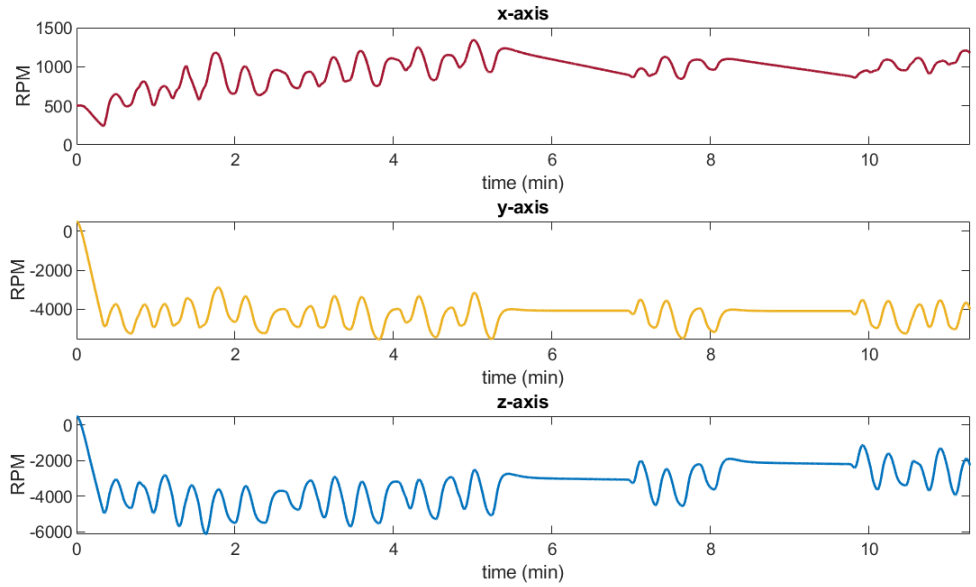


**Figure 59. SMC (Quaternion-based) Reaction Wheels: 1.5 m/s burn (mode 2)**

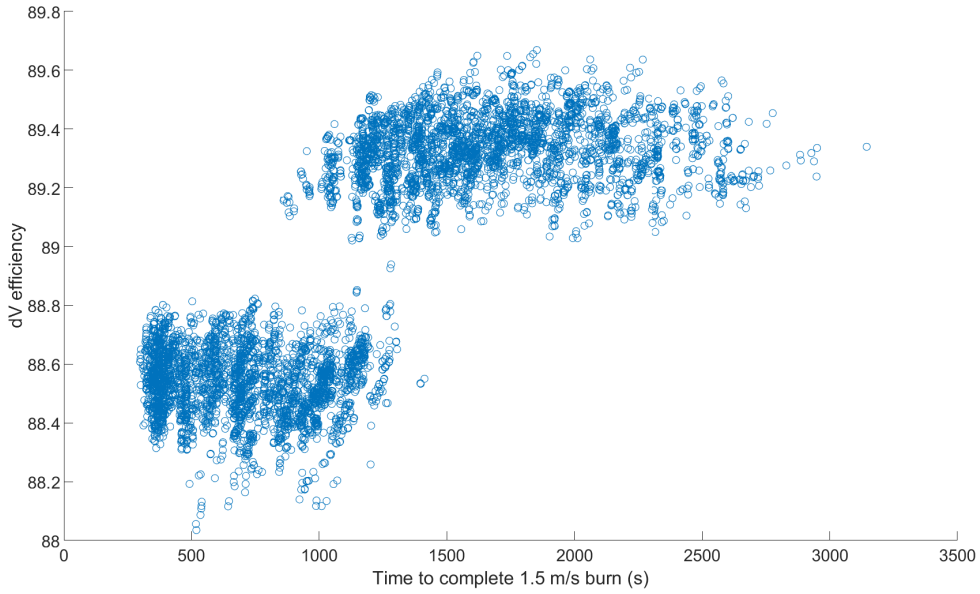


**Figure 60. SMC-Q Monte Carlo Analysis: Efficiency versus Maneuver Time**

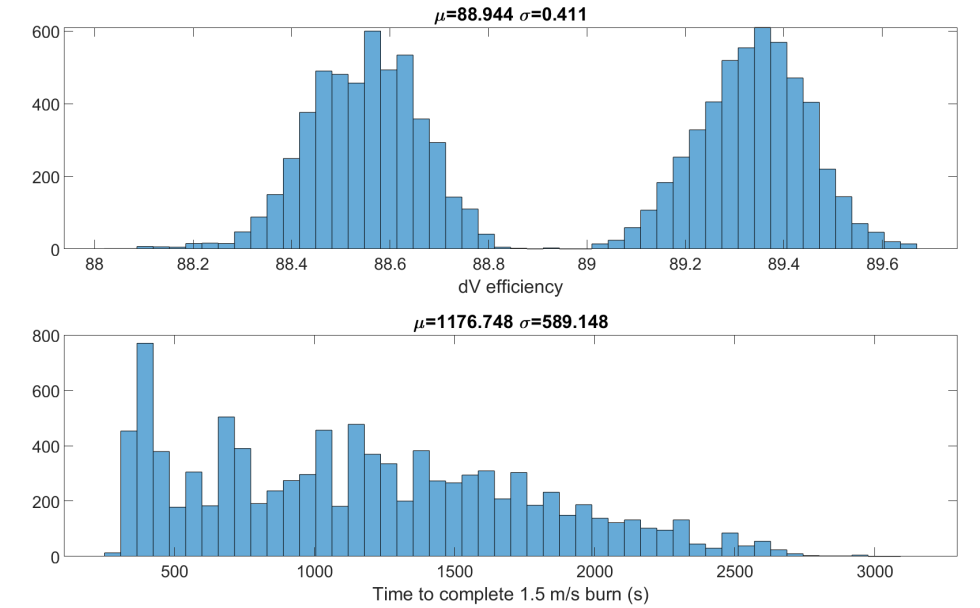


**Figure 61. SMC-Q Monte Carlo Analysis: Time and Efficiency Distributions**

The lower efficiency distribution seen in Fig. 62 yields a normal distribution, while coupled with a uniform distribution in the maneuver time. When compared to Fig. 63, Fig. 62 also has a significantly lower average time to maneuver time. From this it may be inferred that the lower distribution corresponds to mode 2. Conversely, the higher efficiency distribution must be mode 1. When considering the thrust profiles of each, this is expected. Mode 1 fires thrusters nearly 50% less than mode 2 over the course of the 1.5 m/s burn.

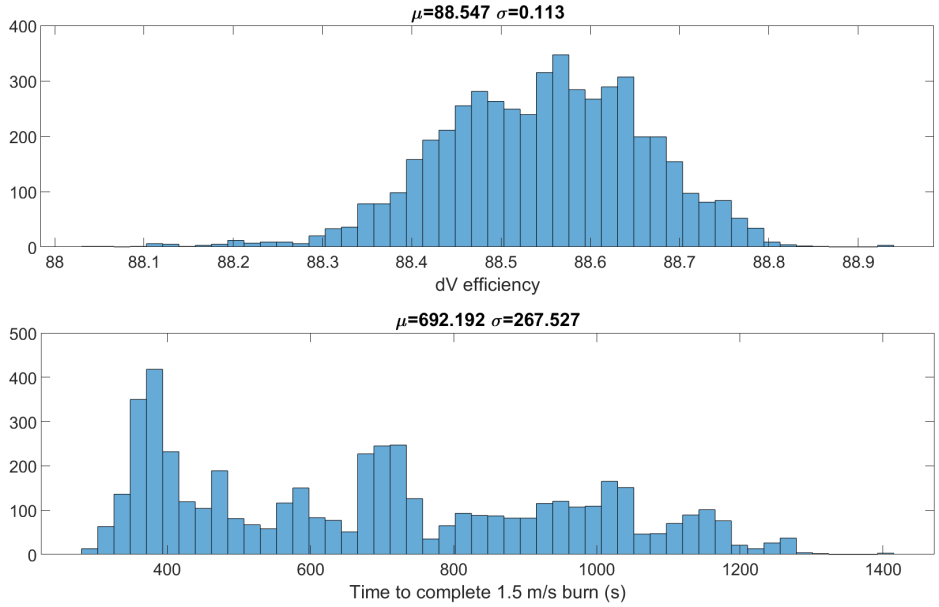


**Figure 62. SMC-Q Monte Carlo Analysis: Time and Efficiency Distributions (lower distribution)**

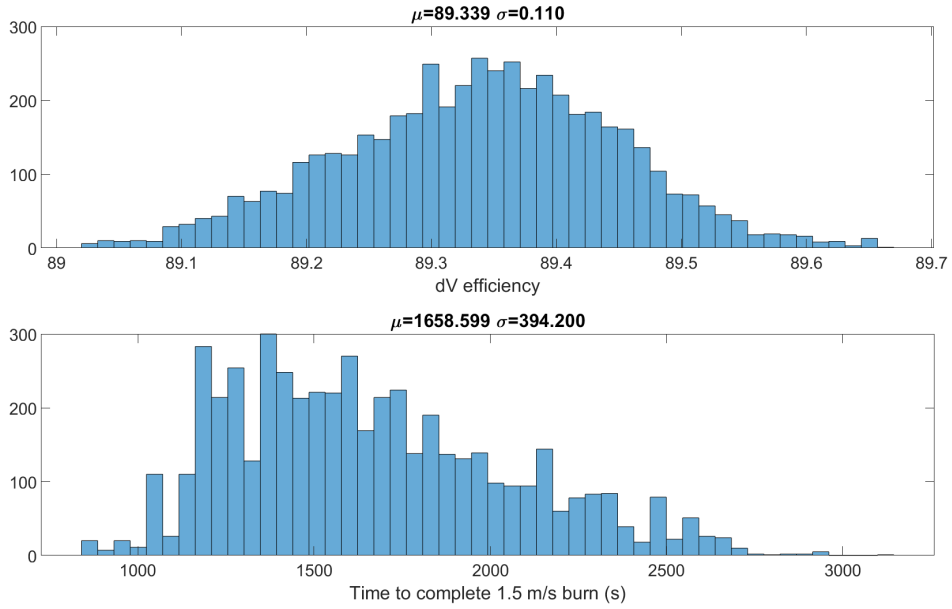


**Figure 63. SMC-Q Monte Carlo Analysis: Time and Efficiency Distributions (higher distribution))**

The bimodal SMC-Q yields insightful results, while also providing strong performance. The trade-off in this control method is fairly straight-forward between the two modes: efficiency for time. The slower system performs at 89.33% efficiency, but takes nearly 30 minutes to complete the small maneuver on average. The other mode of the system conducts the $\Delta v$ in 692.19 seconds on average, but still results in a respectable 88.547% efficiency. Similar the other SMC method, the gain tuning was unable to avoid this issue, and thus jitter must be minimized to force the second mode.

### 4.3.6 Model Reference Adaptive Control

MRAC utilizes methods from On-Off, sliding mode, and adds adaptation parameters to adjust to the uncertainties of the system. The benefits of these added parameters was evident in the extended burn, as the algorithm never reached the dead-zone that plagued the previous controllers. While the $x$ reaction wheel does tend upwards throughout each simulation, the controller keeps the system operating smoothly. Interestingly in Fig. 64, Thruster 2 is firing almost constantly. Surprisingly this behavior did not exist in previous control methods. The thruster modulation looks nearly identical to On-Off's in Fig. 41. This may be explained by the fact that unlike QFC and SMC, MRAC does not require the $x$ torque input, nor the $\Delta v$ check. If the controller works as intended, the system will maintain an equilibrium position, regardless of the jitter or changes in physical properties.

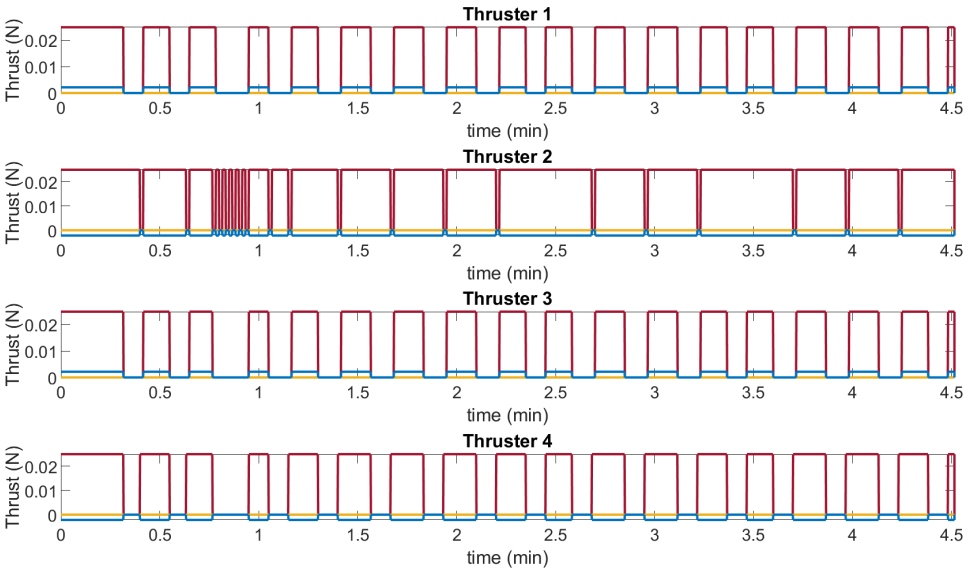


**Figure 64. MRAC Thrusters: 1.5 m/s burn**

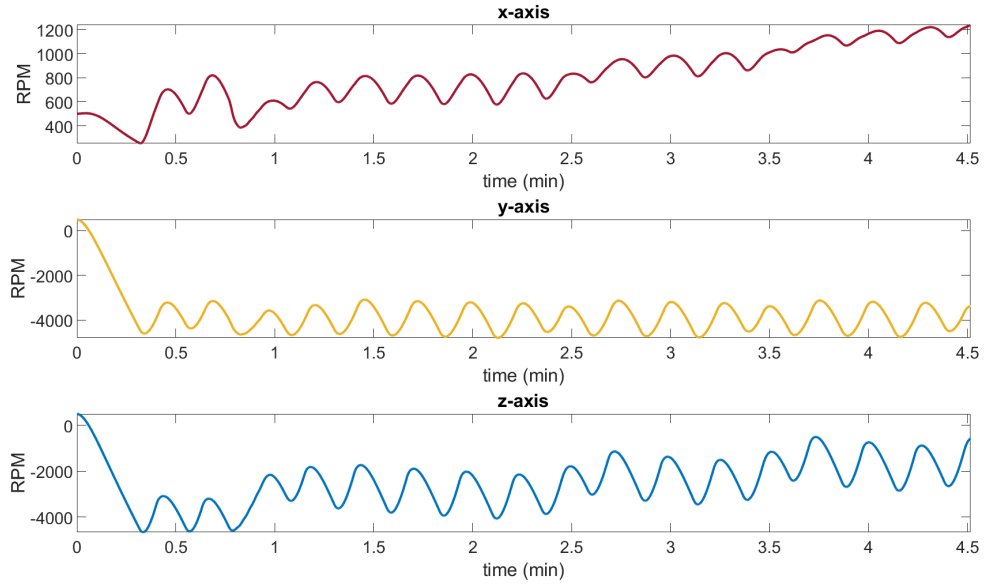


**Figure 65. MRAC Reaction Wheels: 1.5 m/s burn**

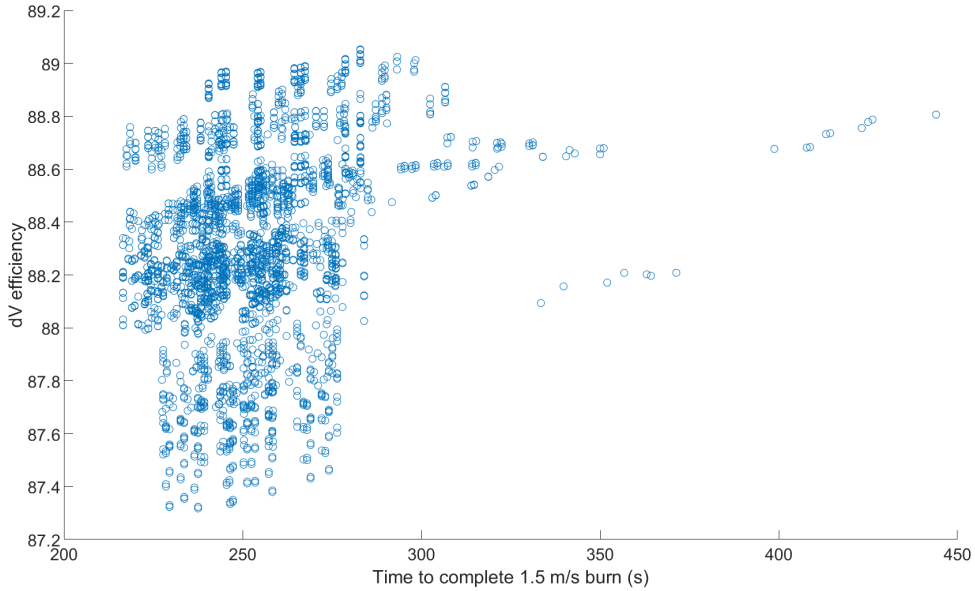


**Figure 66. MRAC Monte Carlo Analysis: Efficiency versus Maneuver Time**

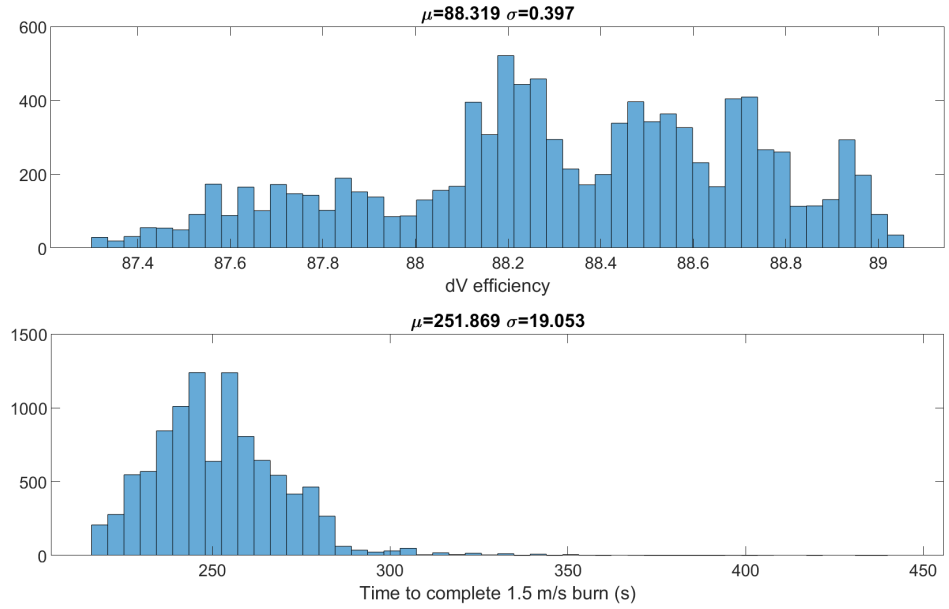


**Figure 67. MRAC Monte Carlo Analysis: Time and Efficiency Distributions**

While On-Off showed a bimodal system, MRAC has a relatively normal distribution. It appears in Fig. 67, that the efficiency is nearly uniform, but has a very consistent maneuver time. This is a strong indication that the adaption parameters are adjusting accordingly to keep the system approaching a similar result. While there are outliers in the time metric, they are not nearly as outlandish as those seen in QFC and SMC-$\vec{\omega}$, and may largely disregarded without concern.

MRAC gives the most consistent results outside of the Baseline Control, with impressive performance. The average efficiency lands at 88.319%, with an average maneuver time of only 251.869 seconds. When compared to other methods, MRAC proves to be the most balanced and consistent approach. Although, this comes at the cost of added complexity, which may be a hindrance on the C&DH. By combining robust and adaptive control, the performance metrics show the second-quickest maneuver average, while also providing the second highest average efficiency.

## 4.4 Summary

The control methods in this research consisted of a simple On-Off method, quaternion feedback control, two sliding mode controls based off different approaches to the governing EOM, and an adaptive control technique, also based on the dynamic EOM. These controllers were tuned via a long-duration burn, similar to what a deorbit maneuver would yield. The controllers were tuned accordingly during this maneuver. The resulting performance was then observed in the phase plane to analyze the stability properties. All methods showed marginally stable tendencies, with On-Off, QFC, and MRAC converging close enough to a closed limit cycle to be determined Lyapunov stable. The SMC techniques were close, but saw too much variance to make such a distinction with confidence.

The controllers were then compared using a small $\Delta v$ maneuver, simulating a phase change for the 6U CubeSat model. This provided a closer view of the thruster sequencing and reaction wheel oscillations. A Monte Carlo Analysis was also run for each nonlinear method, with varying mass and inertial properties of the plant. This resulted in unique results for each controller, where On-Off and SMC-Q were identified as bimodal in their efficiency metrics. On-Off conducted the manuever in the fastest average time of only 218 seconds, while SMC-$\vec{\omega}$ yielded the highest average $\Delta v$ efficiency at 89%. MRAC proved to be the most complex method, while also providing the most balanced metrics of 88.3% $\Delta v$ average efficiency, and 252 seconds average maneuver time.

# V. Conclusions and Recommendations

## 5.1 Conclusions of Research

This research simulated the attitude control of a 6U spacecraft undergoing a translational maneuver using cold-gas thrusters misaligned with the vehicle's COM. The thruster selected for the model was the VACCO Standard MiPS, where there are 4 cold gas thruster nozzles along the same face with an internal tilt of 5 degrees. The challenge imposed by this array is that the thruster control is overdetermined, resulting in the invalidation of traditional thruster selection logic based on attitude error. Additionally, the ADCS chosen, BCT XACT-15, must be treated as a black-box, where the control logic is unknown. Combining the unknown torque, the impulsive nature of the cold gas thrusters, along with uncertain latency within the system, yields potentially unstable attitude control while thrusting. To overcome this, a control algorithm was developed to select thrusters based on ADCS reaction wheel speeds. Using nonlinear control methods, a control torque condition was added to this algorithm in an effort to enhance system performance.

Overall, six control methods were analyzed. The Baseline Control method shows the system's response to no thruster modulation, where desaturation of the reaction wheels must occur intermittently via magnetorquers. On-Off control consisted only of the thruster selection algorithm, driving reaction wheel speeds to a set RPM. QFC was originally developed to model a stable control method for the ADCS. It was also used for thruster control to determine if immediate benefit could be found by adding a condition to the algorithm based on the control torque. From there, two SMC methods were tested via the same control algorithm as QFC. Finally an adaptive technique, MRAC, was implemented with a lightly-modified control algorithm. Two maneuvers were conducted for each method. The first being dubbed a "deorbit"

maneuver was used to determine long term stability of the system, as well as tune respective controller gains. The shorter maneuver was selected as a simple case study of a typical phase change maneuver a satellite in LEO may conduct. A Monte Carlo analysis was conducted for all but the Baseline Control method to determine average performance metrics. 10,000 simulations were run using each of the 5 methods in question. The randomized parameters used included the mass of the satellite, and each primary moment of inertia. These results may be seen in Table 3.

**Table 3. Summary of Control Method Results**

| Control Method | Mean $\Delta v$ Efficiency (%) | Std. | Mean Maneuver Time (s) | Std. | BC |
|---|---|---|---|---|---|
| Baseline Control | 84.97 | N/A | $2.601 \times 10^4$ | N/A | N/A |
| On-Off | 87.459 | 0.614 | 218.3 | 10.74 | 0.6305 |
| QFC | 87.255 | 0.613 | 357.7 | 54.784 | 0.4853 |
| SMC-$\vec{\omega}$ | 88.988 | 0.372 | 548.4 | 85.095 | 0.4698 |
| SMC-Q | 88.944 | 0.411 | 1176.7 | 589.148 | 0.7735 |
| MRAC | 88.319 | 0.397 | 251.9 | 19.053 | 0.4728 |

The Baseline Control method saw the worst performance in both $\Delta v$ efficiency and maneuver time. While the efficiency is only marginally worse, the 7 hours and 13 minutes required to complete a simple 1.5 m/s maneuver would prove to be unacceptable for most missions. Of course, the burn time is entirely dependent on the thrust vector from the spacecraft's center of mass, so this is an extreme case at 4 mm off $y_B$ and $z_B$. These results show that thruster modulation will be required unless extreme precision can be made when aligning these properties. With the uncertainty of center of mass involved with fuel sloshing, harness displacement, and other mass

altering events, the Baseline Control method would not be recommended without a tertiary momentum management subsystem.

On-Off provided surprisingly excellent results in both $\Delta v$ efficiency and maneuver time. The efficiency was increased over the baseline by 3.5% while reducing the maneuver time by over 2 orders of magnitude. All of this was done without using complex mathematics, robust control, or adaptive control. On-Off control also yielded the lowest maneuver time of all the control methods. However, an interesting emergent behavior was observed, where 2 modes of the reaction wheel response were presented. While this is due to unavoidable jitter, the respective averages of each mode are relatively similar to the overall mean in Table 3. On-Off control should be selected for missions that require a near-minimal maneuver time, as well as a low CPU demand.

The QFC method was conducted in an effort to explore the applicability of adding a torque control condition, generated by the QFC control logic, to the On-Off algorithm. This control law was proven to be locally stable in Ch. 3 from the ADCS development, but it proved to be lacking the robustness required to enhance the existing On-Off control. That being said, the results observed from QFC were more consistent than On-Off, as there was no second mode. QFC provides another relatively computationally simple solution with predictable results. Regardless, it is difficult to recommend as the lower efficiency mode of On-Off Control is only marginally less efficient (0.3%) while requiring 50% more time on average to complete the maneuver.

SMC-$\vec{\omega}$ showcased the highest average efficiency with lowest standard deviation while providing the median maneuver time. However, it is worth noting that the distribution was highly skewed left in time to complete the simulation. As discussed in Section 4.3.4, this is due to periodic hangups in the control algorithm. The standard deviation of the maneuver time is roughly 15% of the Generalized Extreme Value average because of this. SMC-$\vec{\omega}$ may be a good control option for missions requiring

the highest efficiency possible, while accepting a relatively high uncertainty in time to complete the maneuver.

The SMC-Q control method showed high efficiency similar to SMC-$\vec{\omega}$, but the simulations required over double the amount of time on average. SMC-Q also displayed a bimodal relationship in $\Delta v$ efficiency similar to On-Off. The higher distribution, if considered on its own, had an average efficiency of 89.339 %, but this same distribution yielded an average time of 1658 seconds. Like On-Off, the jitter between the ADCS and propulsion unit cause the bimodality, and may prove to be very difficult to alleviate for only marginal improvements. While SMC-Q has the potential for a stable control response, SMC-$\vec{\omega}$ outperforms it on average in both efficiency and maneuver time, without seeing the bimodal inconsistencies present in this method.

The final controller analyzed was the MRAC method. MRAC boasts a relatively high efficiency at 88.319 % with a low standard deviation, as well as the second lowest average maneuver time, also with low standard deviation. These combined metrics provide evidence of the robust and adaptive techniques utilized in MRAC. Although, this comes at the cost of higher computational loads, as it is the most complex. This method would be an excellent choice for missions with high uncertainty in physical properties, or missions which are capable of conducting the computational load and require a balanced, consistent thrust response.

## 5.2 Significance of Research

This research pursues a solution to expanding the capabilities of CubeSat propulsion control. While thruster selection is not a new topic, traditional methods rely on additional thrusters strategically placed such that the system is adequately actuated. CubeSats do not always have the volume or mass allowed to include extra actuators, so creative solution must be found. Conversely, from the methods presented in this

work, agile CubeSat missions may utilize underactuated thruster control, freeing more volume and mass for other mission-supporting hardware. Additionally, thruster selection logic often relies upon concrete knowledge of the system's physical construction, mass, and MOI. While spacecraft and their components are typically manufactured and assembled with high precision, mishaps on orbit may alter the physical properties of the vehicle enough to require re-calibration. The algorithmic approach used in this paper do not need such calibration.

COTS components facilitate the market-space for rapid deployment of satellites by supplying researchers and CubeSat developers with standardized products. However, proprietary knowledge about these products may limit the capabilities of adjacent subsystems. This was explored in this research by assuming the ADCS as a black-box and manipulating the MiPS based only on ADCS state data. Without full knowledge of the internal workings of each component, assumptions must be made, eroding the reliability of the model. By applying robust and adaptive techniques on sourced state data, these uncertain characteristics need not be quantified. This concept may be applied beyond the scope of spacecraft stability by considering which components are driving the stability of the system, and how can the controllable adjacent subsystems be used to support this stability.

### 5.2.1 Scholarly Presentations

1. Cottrell, A., Bettinger, R., "Underactuated Attitude Control of a CubeSat Using Cold Gas Thrusters and Nonlinear Control Methods," Presented at the *47th AIAA Dayton-Cincinnati Aerospace Sciences Symposium (DCASS)*, Dayton, OH, March 2022.

## 5.3 Recommendations for Future Work

The analysis conducted in this research may be expanded upon in a variety of avenues. To increase the robustness of this work, the reccomendations for future research include:

1. Implement control logic with a hardware-in-the-loop simulation assembly including emulators of the VACCO Standard MiPS and BCT XACT-15, or similar COTS units. Using hardware to verify the behavior of the model used in this work may provide further insight to the emergent properties of the latency and jitter of the system. Additionally, using similar COTS components will validate the general use case of this work.

2. Replace the thruster output with a digital twin, modeling the internal dynamics of the propulsion unit. This research assumes constant thrust. Added complexity to include valve response, spray from the nozzle, as well as command response will improve accuracy of the model to real systems.

3. Develop and implement a novel thruster selection logic based on attitude error, utilizing PWM or Schmitt Trigger techniques to accomplish the commanded thrust. The primary benefit of this work is that the system is guaranteed to be stable by treating the ADCS as part of the plant. This should be contested by an opposing method under the same system assumptions. From there, an optimal control approach may be pursued.

4. Implement variable mass properties to include, but not limited to, changing mass due to $\Delta v$, dynamic COM and MOI due to fuel sloshing, and solar panel vibrations/flex. The added complexity of these characteristics will test the core concept of this research's approach; namely that the algorithms should maintain stability regardless of these effects.

5. Add orbital dynamics to the simulation. The model analyzed in this work only considered attitude control. The effects of the different control methods may be simulated with a propagating orbital model coupled with the attitude control to give a complete orbital maneuver analysis. This would provide insight to position error of the spacecraft as a primary metric alongside $\Delta v$ efficiency and time to complete the maneuver.

6. Add environmental disturbances using accurate Earth atmospheric and magnetic field models. While the torques exerted by these disturbances are small, they are not inconsequential over the lifetime of the spacecraft. Coupled with the orbital dynamics model, a full dynamic representation may be conducted within the Simulink model for specific maneuvers.

# Bibliography


[1] "Attitude control systems data sheet." `https://bluecanyontech.com/`, 2021. Last Accessed: 09.09.2021.

[2] R. Chen, "CubeSats dance: One water-powered NASA spacecraft commands another in orbit." https://www.nasa.gov/image-feature/ames/cubesats-dance-one-water-powered-nasa-spacecraft-commands-another-in-orbit, 2019.

[3] B. Wie, *Space Vehicle Dynamics and Control*. AIAA Education Series, 1801 Alexander Bell Drive, Reston, VA 20191: AIAA, 2 ed., 2008.

[4] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, New Jersey 07632: Prentice Hall, 1991.

[5] CDS, "CubeSat design specification," 2015. Rev. 13.

[6] M. Langer and J. Bouwmeester, "Reliability of CubeSats - statistical data, developers' beliefs and the way forward," 2016.

[7] J. Hansen and P. Graven, "A hierarchy of guidance, navigation, and control elements for responsive space missions," 2010.

[8] J. Lyke, S. Cannon, D. Fronterhouse, D. Lanza, and T. Byers, "A plug-and-play system for spacecraft components based on the USB standard," 2005.

[9] S. T. Bednarski, "CubeSat attitude determination and control system (ADACS) characterization and testing for rendezvous and proximity operations (RPO)," 2021.

[10] F. H. Martin and R. H. Batting, "Computer-controlled steering of the Apollo spacecraft," 1967.

[11] B. Marcantel, "Project Apollo: Stability analysis of Apollo attitude control during service propulsion thrusting," 1969.

[12] J. S. Arora, *Introduction to Optimum Design*. Elsevier, 4th ed., 2017.

[13] C. M. Jewison, "Reconfigurable thruster selection algorithms for aggregative spacecraft systems," 2014.

[14] C. M. Pong, "Autonomous thruster failure recovery for underactuated spacecraft," 2008.

[15] K. M. Fauske, "Attitude stabilization of an underactuated rigid spacecraft," 2003.

[16] R. Tedrake, "Underactuated robotics: Learning, planning, and control for efficient and agile machines course notes for MIT 6.832," 2009.

[17] G. A. Leonov, "Brockett's problem in the theory of stability of linear differential equations," 2001.

[18] P. Tsiotras and J. Luo, "Control of underactuated spacecraft with bounded inputs," *Automatica*, 2000.

[19] P. Morin and C. Samson, "Time-varying exponential stabilization of a rigid spacecraft with two control torques," *IEEE Transactions on Automatic Control*, vol. 42, pp. 528–534, 1997.

[20] C. D. Petersen, F. Leve, and I. Kolmanovsky, "Model predictive control of an underactuated spacecraft with two reaction wheels," *Journal of Guidance, Control, and Dynamics*, vol. 40, pp. 320–332, 2017.

[21] Y. Bunryo, S. Satoh, Y. Shoji, and K. Yamada, "Feedback attitude control of spacecraft using two single gimbal control moment gyros," *Advances in Space Research*, vol. 68, pp. 2713–2726, 10 2021.

[22] A. Zavoli, G. D. Matteis, F. Giulietti, and G. Avanzini, "Single-axis pointing of an underactuated spacecraft equipped with two reaction wheels," *Journal of Guidance, Control, and Dynamics*, vol. 40, pp. 1465–1471, 2017.

[23] E. Hemingway and O. O'Reilly, "Perspectives on Euler angle singularities, gimbal lock, and the orthogonality of applied forces and applied moments," *Multibody System Dynamics*, vol. 44, 09 2018.

[24] J. B. Kuipers, *Quaternions and Rotation Sequences*. Princeton University Press, 5th ed., 2002.

[25] P. C. Hughes, *Spacecraft Attitude Dynamics*. Dover Publipcations, 2004.

[26] J. R. Wertz, D. F. Everett, and J. J. Puschell, *Space Mission Engineering: The New SMAD*. Microcosm Astronautics Books, 2018.

[27] NASA, "Small spacecraft technology state of the art report," 2021.

[28] VACCO, "Standard propulsion system data sheet," 2022.

[29] K. Ogata, *Modern Control Engineering*. Prentice Hall, 5th ed., 2010.

[30] B. Wie, H. Weiss, and A. Arapostathis, "Quarternion feedback regulator for spacecraft eigenaxis rotations," *Journal of Guidance, Control, and Dynamics*, vol. 12, pp. 375–380, 1989.

[31] Y. Yoshimura, S. Hokamoto, and T. Matsuno, "Satellite position and attitude control by on-off thrusters considering mass change," 2012.

[32] R. Wisniewski, "Linear time varying approach to satellite attitude control using only electromagnetic actuation," 1997.

[33] B. V. Sarli, A. L. da Silva, and P. Paglione, "Sliding mode attitude control using thrusters and pulse modulation for the aster mission," *Computational and Applied Mathematics*, vol. 34, pp. 535–556, 7 2015.

[34] H. D. Curtis, *Orbital Mechanics for Engineering Students*. Elsevier Ltd., 4th ed., 2021.

[35] J. Bowen, M. Villa, and A. Williams, "CubeSat based rendezvous, proximity operations, and docking in the CPOD mission," 2015.

[36] M. Tsay, J. Frongillo, K. Hohman, and B. K. Malphrus, "Lunarcube: A deep space 6U CubeSat with mission enabling ion propulsion technology," 2015.

[37] R. A. Bettinger, "Bimodality in reentry latitude predictions for spacecraft in prograde orbits," *Journal of Applied Statistics*, 2021. DOI: 10.1080/02664763.2021.2008328.

# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 074-0188*

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to an penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.
**PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

| 1. REPORT DATE *(DD-MM-YYYY)* | 2. REPORT TYPE | 3. DATES COVERED *(From – To)* |
|---|---|---|
| 24 Mar 2022 | Master's Thesis | Oct 2020 – Mar 2022 |

**4. TITLE AND SUBTITLE**

Underactuated Attitude Control of a CubeSat Using Cold Gas Thrusters and Nonlinear Control Methods

**5a. CONTRACT NUMBER** N/A
**5b. GRANT NUMBER** N/A
**5c. PROGRAM ELEMENT NUMBER** N/A

**6. AUTHOR(S)**

Cottrell, Adam S., Capt, USSF

**5d. PROJECT NUMBER** N/A
**5e. TASK NUMBER** N/A
**5f. WORK UNIT NUMBER** N/A

**7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(ES)**

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/ENY)
2950 Hobson Way, Building 640
WPAFB OH 45433-8865

**8. PERFORMING ORGANIZATION REPORT NUMBER**

AFIT-ENY-MS-22-M-285

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**

Intentionally Left Blank

**10. SPONSOR/MONITOR'S ACRONYM(S)** N/A
**11. SPONSOR/MONITOR'S REPORT NUMBER(S)** N/A

**12. DISTRIBUTION/AVAILABILITY STATEMENT**

**Distribution A:** Approved for Public Release; Distribution Unlimited

**13. SUPPLEMENTARY NOTES**

The material is declared a work of the U.S. Government and is not subject to copyright protection in the United States

**14. ABSTRACT**

Impulsive thrusters on small satellites, such as CubeSats, are typically used for attitude control. However, to become more agile, small CubeSats must also look to propulsion systems utilizing impulsive thrusters, such as cold-gas, for translational maneuvers. The combined thrust vector is often misaligned with the system's center of mass resulting in a disturbance torque. This must be counteracted by either an attitude determination and control system (ADCS), additional thrusters, or a control method to keep the satellite's attitude at or near equilibrium. Nonlinearities generated by the impulsive maneuvers are overcome via control techniques explored in this research to include on-off control, sliding mode control, and model reference adaptive control (MRAC). These methods were then compared to a baseline test without thruster modulation, where the reaction wheels must de-saturate prior to continuing the maneuver. For a 1.5 m/s delta-v maneuver, the nonlinear control techniques completed the maneuver nearly 100 times faster than the baseline, while improving pointing accuracy throughout the burn by up to 5%.

**15. SUBJECT TERMS**

Underactuated control; attitude control; nonlinear control; cascading control; CubeSat, COTS; impulsive propulsion

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT | b. ABSTRACT | c. THIS PAGE | | | Robert A. Bettinger, Ph.D, Maj (Advisor) |
| U | U | U | UU | 122 | 19b. TELEPHONE NUMBER (937) 255-6565, ext. 4578 (Robert.Bettinger@afit.edu) |