Advancements in the Design and Development of CubeSat Attitude Determination and Control Testing at the Virginia Tech Space Systems Simulation Laboratory

Anthony T. Wolosik

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Aerospace Engineering

Jonathan T. Black, Chair
Gregory D. Earle
Cornel Sultan

August 3, 2018
Blacksburg, Virginia

Keywords: CubeSat, Attitude Control, Air Bearing,
Reaction Wheel Array, Helmholtz Cage

Advancements in the Design and Development of CubeSat Attitude Determination and Control Testing at the Virginia Tech Space Systems Simulation Laboratory

Anthony T. Wolosik

ABSTRACT

Among the various challenges involved in the development of CubeSats lies the attitude determination and control of the satellite. The importance of a properly functioning attitude determination and control system (ADCS) on any satellite is vital to the satisfaction of its mission objectives. Due to this importance, three-axis attitude control simulators are commonly used to test and validate spacecraft attitude control systems before flight. However, these systems are generally too large to successfully test the attitude control systems on-board CubeSat-class satellites. Due to their low cost and rapid development time, CubeSats have become an increasingly popular platform used in the study of space science and engineering research. As an increasing number of universities and industries take part in this new approach to small-satellite development, the demand to properly test, verify, and validate their attitude control systems will continue to increase. An approach to CubeSat attitude determination and control simulation is in development at the Virginia Tech Space Systems Simulation Laboratory. The final test setup will consist of an air bearing platform placed inside a square Helmholtz cage. The Helmholtz cage will provide an adjustable magnetic field to simulate that of a low earth orbit (LEO), and the spherical air bearing will simulate the frictionless environment of space. In conjunction, the two simulators will provide an inexpensive and adjustable system for testing any current, and future, CubeSat ADCS prior to flight. Using commercial off the shelf (COTS) components, the Virginia Tech CubeSat Attitude Control Simulator (CSACS), which is a low cost, lightweight air bearing testing platform, will be coupled with a 1.5-m-long square Helmholtz cage design in order to provide a simulated LEO environment for CubeSat ADCS validation.

Advancements in the Design and Development of CubeSat Attitude Determination and Control Testing at the Virginia Tech Space Systems Simulation Laboratory

Anthony T. Wolosik

GENERAL AUDIENCE ABSTRACT

The attitude determination and control subsystem is a vital component of a spacecraft. This subsystem provides the pointing accuracy and stabilization which allows a spacecraft to successfully perform its mission objectives. The cost and size of spacecraft are dependent on their specific applications; where some may fit in the palm of your hand, others may be the size of a school bus. However, no matter the size, all spacecraft contain some form of on-board attitude determination and control. This leads us to the introduction of a miniaturized class of spacecraft known as CubeSats. Their modular $10 \times 10 \times 10$ cm cube structural design allows for both low cost and rapid development time, making CubeSats widely used for space science and engineering research in university settings. While CubeSats provide a low cost alternative to perform local, real-time measurements in orbit, it is still very important to validate the attitude determination and control subsystem before flight to minimize any risk of failure in orbit. Thus, the contents of this thesis will focus on the development, design, and testing of two separate spacecraft attitude determination and control simulation systems used to create an on-orbit environment in a laboratory setting in order to properly validate university-built CubeSats prior to flight.

To my parents. For their continued support and patience throughout my life, especially my years at Virginia Tech, I can't thank them enough.

Acknowledgments

None of this work could have been completed without the dedicated efforts of various members in the Space Systems Simulations Laboratory. I would personally like to thank Dylan Thomas, Larry Hensley, Nick Tibbetts, Jordan Schafer, Keith Tiemann, and all of the other current and former Virginia Tech students who have contributed to this project.

I would like to express my deepest gratitude to my advisor, Dr. Jonathan Black, for his guidance, advice, encouragement, and support throughout my time as an undergraduate researcher and graduate student. Also, thanks to each member of my committee for their help, patience, and encouragement. Conducting research and taking courses with both Dr. Earle and Dr. Sultan are significant reasons as to why I chose my research focus in spacecraft dynamics and control.

Contents

List of Figures

List of Tables

List of Abbreviations

ACS	Attitude Control System
ADCS	Attitude Determination and Control System
AFIT	Air Force Institute of Technology
AFRL	Air Force Research Laboratory
ASCII	American Standard Code for Information Interchange
ASTREX	Advanced Space Structure Technology Research Experiment
AWG	American Wire Gauge
CAD	Computer-Aided Design
Cal Poly	California Polytechnic State University
CM	Center of Mass
CMG	Control Moment Gyro
COM	Communication System
COTS	Commercial Off The Shelf
CR	Center of Rotation
CSACS	CubeSat Attitude Control Simulator
CubeTAS	CubeSat Three Axis Simulator
DOA	Dead-On-Arrival
DOF	Degrees of Freedom
DSACSS	Distributed Spacecraft Attitude Control System Simulator
EELV	Evolved Expendable Launch Vehicle
EMF	Electromotive Force
EPS	Electrical Power System
ESPA	EELV Secondary Payload Adapter
GUI	Graphical User Interface
IGRF	International Geomagnetic Reference Field

IMU	Inertial Measurement Unit
LAICE	Lower Atmosphere Ionosphere Coupling Experiment
LCF	Lyapunov Candidate Function
LEO	Low Earth Orbit
MCS/LOS	Momentum Control System and Line of Sight
MEMS	Microelectromechanical System
MIT	Massachusetts Institute of Technology
MMU	Mass Moving Unit
MOI	Moment of Inertia
NASA	National Aeronautics and Space Administration
NPS	Naval Postgraduate School
OBC	On-Board Computer
OLS	Ordinary Least Squares Estimator
PD	Proportional-Derivative Controller
PL	Payload
P-POD	Poly-PicoSat Orbital Deployer
RWA	Reaction Wheel Array
SAA	South Atlantic Anomaly
SDL	Utah State University Space Dynamics Laboratory
STK	Systems Tool Kit
STR	Structure and Deployables
TACT	Triaxial Attitude Control System
THHN	Thermoplastic High Heat-resistant Nylon
UNAM	National Autonomous University of Mexico
USU	Utah State University

List of Symbols

A	3×3 matrix in the CSACS nonlinear equations of motion, or magnetic vector potential (dependent on usage)
A_w	3×4 distribution matrix to transform four reaction wheel vectors to the three body axes
B	3×1 column vector in the CSACS nonlinear equations of motion, or magnetic field at arbitrary location P (dependent on usage)
β	reaction wheel inclination angle
$\{\hat{\mathbf{b}}_1, \hat{\mathbf{b}}_2, \hat{\mathbf{b}}_3\}$	the three unit base vectors of a body-fixed reference frame
d	distance between coils for largest homogeneous magnetic field
$d\vec{l}$	infinitesimal length vector of the current element
e	back EMF
E	total energy of the CSACS
$\mathcal{F}_b$	body-fixed reference frame with base vectors $\{\hat{\mathbf{b}}_1, \hat{\mathbf{b}}_2, \hat{\mathbf{b}}_3\}$
$\mathcal{F}_i$	inertial reference frame with base vectors $\{\hat{\mathbf{i}}_1, \hat{\mathbf{i}}_2, \hat{\mathbf{i}}_3\}$
H	the CSACS angular momentum about CR
H_c	the CSACS angular momentum about CM
H_w	angular momentum of the reaction wheels
I	moment of inertia tensor of the CSACS, or current flowing through the element considered (dependent on usage)
$\{\hat{\mathbf{i}}_1, \hat{\mathbf{i}}_2, \hat{\mathbf{i}}_3\}$	the three unit base vectors of an inertial reference frame
I_m	moment of inertia of the DC motor shaft
I_w	moment of inertia of the reaction wheel
$\mathbf{k}$	Euler axis, or eigenaxis
K_d	quaternion feedback derivative gain

K_e	DC motor back EMF gain (V/(rad/s))
K_p	quaternion feedback proportional gain
K_τ	DC motor torque gain (N·m/A)
L	DC motor inductance, or square side length of a coil (dependent on usage)
λ	principal Euler angle
m	mass of individual MMU
m_{tot}	total mass of the CSACS system
M_p	maximum percent overshoot
μ_0	magnetic permeability of free space ($4\pi \times 10^{-7}$ Tesla·m/A)
$\mathbf{\Phi}$	condensed 3×3 matrix of the CSACS simplified equations of motion integrated over a short time period
$\mathbf{\Phi}_{OLS}$	OLS expansion of $\mathbf{\Phi}$ for many time steps
ϕ	roll angle
P_{in}	power input provided to the DC motor
ψ	yaw angle
$\overline{\mathbf{q}}$	4×1 unit quaternion matrix, where $\overline{\mathbf{q}} = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}^T$
q_0	scalar component of the quaternion
$\mathbf{q}$	Euler axis component of the quaternion, where $\mathbf{q} = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix}^T$
R	DC motor resistance
$\mathbf{R}^{bi}$	rotation matrix that transforms vectors from $\mathcal{F}_i$ to $\mathcal{F}_b$
r	position vector from the CSACS CR to CM
$\hat{r}$	OLS position vector from the CSACS CR to CM
δr	difference between estimated and original CM position vector
r_m	final MMU position vector from its original starting point
Δr_m	MMU move distance between algorithm iterations
T_{aero}	aerodynamic torque

T_c	reaction wheel torques produced along the three body axes of the CSACS (T_{c1}, T_{c2}, T_{c3})
T_d	disturbance torque
T_{ext}	combined applied external torques on the CSACS
T_g	gravitational torque
T	kinetic energy of the CSACS
t_s	settling time
τ_w	individual reaction wheel torques (τ_1, τ_2, τ_3, τ_4)
τ_L	workload torque of the DC motor
θ	pitch angle
$\mathbf{\Theta}$	column matrix whose 3 elements are the Euler angles ψ, θ, ϕ
U	potential energy of the CSACS
v	velocity of the CSACS CM, or voltage applied to DC motor (dependent on usage)
V	Lyapunov candidate function, based the CSACS total energy
v_b	column matrix whose 3 elements are the components of $\vec{\mathrm{v}}$ in $\mathcal{F}_b$
v_i	column matrix whose 3 elements are the components of $\vec{\mathrm{v}}$ in $\mathcal{F}_i$
$\vec{\mathrm{v}}$	arbitrary vector notation
ω_b^{bi}	angular velocity of $\mathcal{F}_b$ with respect to $\mathcal{F}_i$ expressed in $\mathcal{F}_b$
$\vec{\omega}^{bi}$	angular velocity of $\mathcal{F}_b$ with respect to $\mathcal{F}_i$
ω_n	natural frequency
$\vec{\omega}$	an angular velocity vector
$\Delta\Omega$	condensed 3×1 column vector of the CSACS simplified equations of motion integrated over a short time period
$\Delta\Omega_{OLS}$	OLS expansion of $\Delta\Omega$ for many time steps
ζ	damping ratio

Chapter 1

Introduction

1.1 Background

The CubeSat project began in 1999 as a collaboration between Professor Jordi Puig-Suari of California Polytechnic State University (Cal Poly) and Professor Bob Twiggs of Stanford University's Space Systems Development Lab. The purpose of the CubeSat project was to introduce an academic-friendly method of developing and launching satellites via standardization. CubeSats are small-satellites designed to be cheaper and have more rapid development times when compared to conventional satellites. The low cost and high risk acceptance of these systems allow scientists and researchers to conduct breakthrough scientific experimentation. This project has become an international collaborative effort with over 100 universities participating [32].

The CubeSat structural platform is based in (U), where 1U corresponds to a $10 \times 10 \times 10$ cm cube constrained to a mass of up to 1.33 kg. Cal Poly developed a launcher called the Poly-Picosat Orbital Deployer (P-POD), which is a standardized deployment system that ensures all CubeSat developers conform to the specified physical requirements [32]. Due to their uniform sizes and masses, CubeSats are inexpensively ejected from this type of system as secondary payloads on missions where the primary payload is a satellite or space capsule. NASA's launch services program has a set of specifications for CubeSat launches which also constrain the external size and shape configurations for consistent launches, along with preventing any interference with primary missions [33]. Currently, over 800 CubeSats have been launched, with projections of as many as 2,400 requiring launch through 2023 [9, 25].

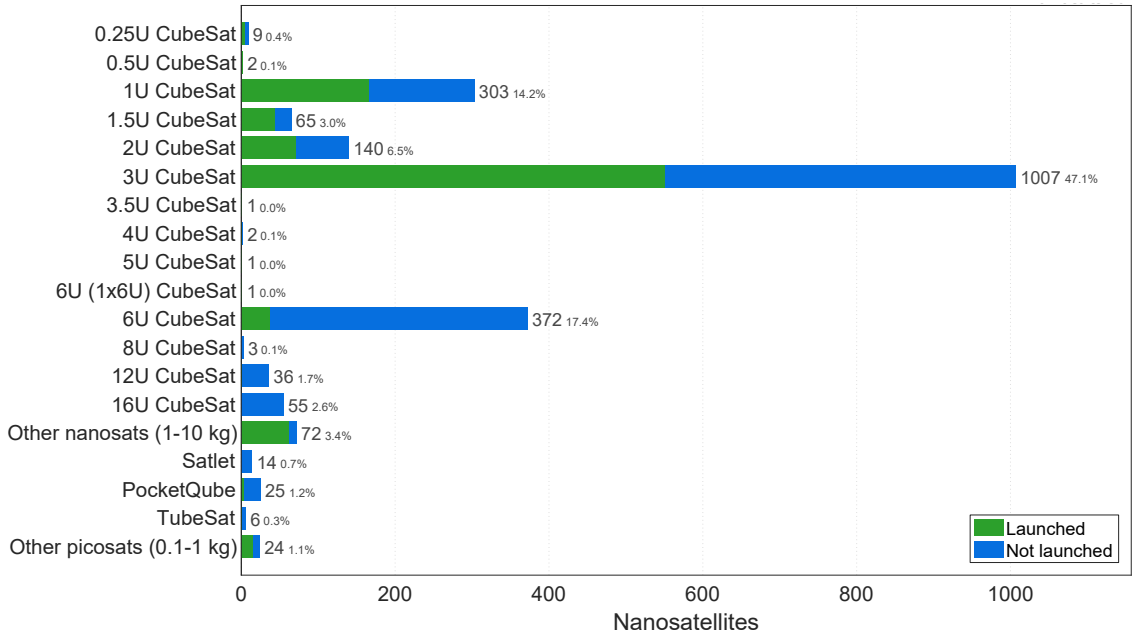


Figure 1.1: Nanosatellite Launch Statistics. Database of information includes: all CubeSats (0.25 – 27 U), nanosatellites from 1 – 10 kg (non-CubeSats are listed in kg), picosatellites from 0.1 – 1 kg, PocketQubes, TubeSats, and SunCubes. Reprinted from the Nanosatellite Database website with permission [25].

Different U configurations exist, such as the 3U, which is a modular combination of 3 single units stacked together, ultimately increasing accommodation for additional instrumentation and power. Since the most common CubeSats to date have been 1U and 3U, the P-POD accounts for the majority of CubeSat deployments. However, as Figure 1.1 shows, recent growth in this field of research has led to the development of 6U and larger configurations, along with more robust deployment systems, increasing the scientific and engineering capabilities that CubeSats can provide. Satellites are classified into different categories by mass; therefore, different configurations of CubeSats fall into different categories based on the number of units. For example, a 1U CubeSat may have a mass less than 1.33 kg and therefore lies on the border between the pico and nanosatellite classification, whereas a 12U configuration would be classified as a microsatellite according to Table 1.1.

Table 1.1: Satellite Size Classifications. The red box identifies the possible satellite classifications of all current CubeSat configurations ranging from 0.25 – 27 U, corresponding to a mass range of approximately 0.2 – 40 kg [17, 25].

Classification	Mass [kg]
Large Satellite	> 1000
Medium Satellite	500 – 1000
Small Satellite	100 – 500
Microsatellite	10 – 100
Nanosatellite	1 – 10
Picosatellite	0.1 – 1
Femtosatellite	< 0.1

The increasing number of micro-electronic components being developed has contributed to the ability to have more capable scientific payloads in orbit. Incorporating these components with the low cost development of CubeSats now allows scientists and engineers to develop and fly hardware without previous flight heritage, leading to rapid development and testing of complex exoatmospheric phenomena.

While CubeSats provide a low cost alternative to perform in situ, or local, real-time measurements in orbit, it is still very important to validate the system and all of its components on the ground to minimize any risk of failure in orbit. One of the more complicated, and vital, systems to test is the attitude determination and control, which stabilizes the CubeSat and orients it in desired directions during flight despite external disturbing torques [28]. Once the CubeSat is injected into its orbit, it will be in a micro-gravity environment, and without the application of force, it will be in a free tumble. Also, being in LEO, external disturbing forces on the spacecraft include: accelerations resulting from a central body, *i.e.*, Earth, which include spherical harmonics (dominated by the zonal J_2-term, which reflects

the oblateness of the Earth), atmospheric drag, third-body perturbations, solar radiation pressure, tides, magnetic field effects, as well as radiation from the South Atlantic Anomaly (SAA) [58]. In order for the spacecraft to accomplish its scientific mission, it must be pointed in its desired orientation. Achieving the correct pointing requirements relies on the specifications of the devices that both measure and control attitude. Attitude is measured by sensors that produce known quantities in the form of a three-dimensional vector. By combining information from at least two sensors and comparing their measurements with expected values at a nominally rotated position, attitude is determined. Examples of attitude determination sensors used in small satellites include: sun sensors, star trackers, Earth sensors, magnetometers, and inertial measurement units (IMUs). Controlling attitude requires actuators that impart torques about specific axes. Traditionally, spacecraft attitude control is accomplished using propulsion, reaction wheels, and control moment gyroscopes (CMGs). However, due to commercial availability and sizing constraints of CubeSats, magnetic torque rods (magnetorquers) accompanied with reaction wheels are typically used [12]. Reaction wheels increase or decrease rotation rates by using the conservation of angular momentum. Accompanying the reaction wheels are magnetorquers, which apply a current through coils that induce a magnetic dipole, thus causing a torque in relation to the Earth's magnetic field. Due to the drastic differences in gravity and atmospheric content between the space and laboratory setting, careful considerations must be taken into account to accurately test a CubeSat ADCS. One traditional method to verify attitude control systems in a laboratory setting is through the use of an air bearing simulator [47]. Thus, the design and function of such a system will be examined in order to provide a nearly frictionless testing environment for CubeSat attitude control systems.

Methods used to verify attitude determination vary considerably in accuracy and cost. As previously mentioned, attitude sensors are grouped together to increase the overall accuracy of the system. An example of this could be a sun sensor and magnetometer combination. Testing a sun sensor can be accomplished with a single light source, such as a flashlight. However, to test magnetometers, a magnetic field must be generated with comparable in-

tensity to that of the LEO environment the spacecraft will experience in-flight. There are different methods to generating a simulated magnetic field in a laboratory setting. These include induction coils, solenoids, and Helmholtz coils. Helmholtz coils are of particular interest because of their unique ability to generate uniform magnetic fields over large areas along a single axes. To create a more uniform magnetic field, sets of Helmholtz coils can be oriented to create a uniform magnetic field in all three axes. This configuration is called a Helmholtz cage and is critical to the testing of spacecraft magnetometers. The design and function of such a cage will also be examined in order to ultimately generate a LEO magnetic field environment in a laboratory setting.

1.2 Motivation

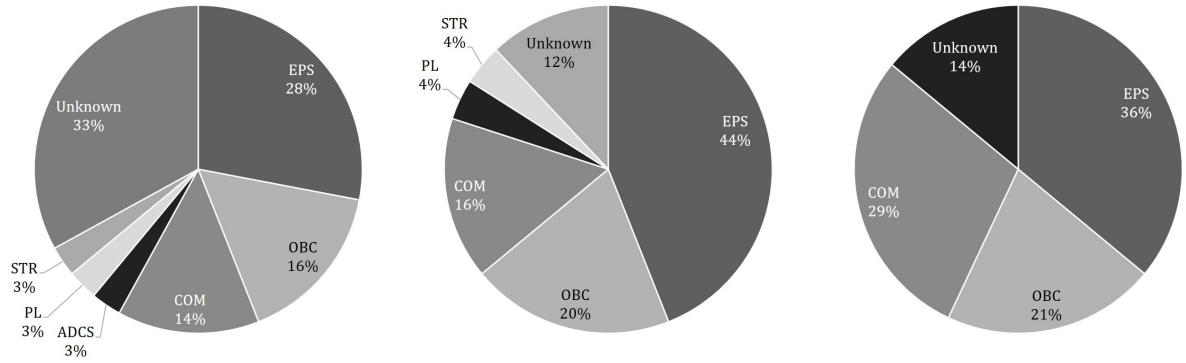
1.2.1 CubeSat Mission Success

Traditional spacecraft design is conducted by using highly reliable components, conservative designs, and extensive testing at subsystem and integrated system levels to achieve extended mission durations [27]. CubeSats, on the other hand, utilize COTS products, yielding an increased performance per mass at higher risk, but lower cost. For this reason, many univeristy-built CubeSat missions do not deploy in an operational state, or fail shortly after the spacecraft is ejected from its deployer. A probabilistic CubeSat reliability estimation tool developed by researchers at the Technical University of Munich and Delft University of Technology suggests that a large percentage of the early failure cases could have been detected and avoided by more careful and adequate system-level functional testing before flight [26, 27]. Still, many universities fail to incorporate system level functional testing before flight and end up launching a satellite that was never adequately functional. A summary of results developed from the researchers' estimation tool follows, which presents the reliability of various CubeSat subsystems [26].

The following six subsystems (with an additional "unknown" category where no specific subsystem was identified as the primary cause for failure) are defined below [26]:

- ADCS
- Communication System, including antennas (COM)
- Electrical Power System (EPS)
- On-Board Computer (OBC)
- Payload (PL)
- Structure and Deployables, not including antennas (STR)
- Unknown

Subsystem contributions to CubeSat failures are presented in Figure 1.2. The "unknown" category presents itself as the major source of failure in early stages of flight. COM was not established for the majority of dead-on-arrival (DOA) cases, and so, CubeSat developers acknowledge that about half of the DOA satellites fall in the "unknown" category. EPS is the next largest contributor to failure in the deployment stage, and the largest failure contribution in later stages, with greater than 40% of all failures after 30 days. COM accounts for roughly 30% of the failures after 90 days. STR, PL, and ADCS contribute together less than 10% of failures. In addition to the statistical data presented in Figure 1.2, an even more alarming statistic was determined from a survey of over 100 university CubeSat developers. The survey revealed that the likelihood of failure for a university-built CubeSat within the first 6 months of flight was estimated to be slightly below 50%, while at the same time, the likelihood of failure, if the survey participant was a team member of the to-be-launched CubeSat, was around 23%. Thus, the likelihood of failure of one's own mission is either optimistic, or the judgment of other missions is conservative.



(a) orbit insertion (incl. DOA) **(b)** 30 days **(c)** 90 days

Figure 1.2: Subsystem Contributions to CubeSat Failure. Reproduced with permission from M. Langer, 2016, 30th Annual AIAA/USU Conference on Small Satellites, p. 5. Copyright 2016 AIAA/USU [26].

On the surface, the ADCS does not contribute itself as a major source of failure; however, this is most likely based on the fact that university CubeSat missions are designed for simplicity and robustness. Mission operations are typically simple, with minimal- to no-active attitude control [57]. However, every year, the active control percentages have been increasing for CubeSat missions, mostly due to more demanding missions, larger payload sizes, advances in technology, and wider availability of advanced COTS attitude control subsystems. It appears that, for these reasons, active attitude control adoption will continue to increase [39]. This increase in active control usage may result in more advanced and challenging missions. Moreover, the better control authority over the CubeSat with active control may enable developers to design CubeSats for lower altitudes [63], to exploit the advantage of operating closer to Earth, which include shorter range, better resolution, short revisit, economical launch costs, and efficient debris mitigation processes. So while there are still many uncertainties in the reliability of each CubeSat subsystem before flight, one strategy to mitigate these uncertainties is common among all developers: improving mission success begins with full-system functional testing before flight [57].

1.2.2 Attitude Control Simulation Requirements

Due to the relatively low mass of CubeSats compared to traditional satellites, most operational attitude control simulators are incapable of supporting CubeSats since the total mass is either negligible or within error margins in relation to the overall mass of the simulator. An increased need to test the state-of-the-art CubeSat ADCS has started to surface in the small-satellite community. As this community continues to grow, more successful missions are being conducted which lead to investigators seeking larger payload capabilities in order to fly more advanced instrumentation [60]. To accommodate these increasing capabilities, new 6U CubeSats and 12U+ prototype designs are in development. Simulators must be capable of accommodating this rapidly evolving growth trend, as well as handling the increasingly advanced ADCS of larger frames. Thus, the requirements to design a CubeSat attitude control platform will be determined by actuator capabilities and ADCS sensitivity. These characteristics will define the maximum disturbance torque of the simulator which has no tangible influence on the test article. For realistic simulations, the maximum disturbance torque of the simulator is defined to be two orders of magnitude below the maximum control torque available in the CubeSat, or one order of magnitude below the total expected external torque on-orbit [13]. For magnetorquer-based CubeSat attitude control systems, Marquis estimated external torques on-orbit for the 6U CubeSat LAICE to be 2×10^{-6} N·m [30], Li *et al.* estimated the total disturbance torque for 1U CubeSats to be on the order of 5×10^{-7} N·m [29], and according to flight data from the 3U CubeSat Robusta-3a, external torques on-orbit do not exceed 10^{-4} N·m [13]. Additionally, COTS CubeSat reaction wheels were reviewed to ensure that the CSACS will be suitable for any other satellite of this class. Reaction wheel control torques range from $0.23 - 7$ mN·m, as shown in Figure 1.3. Thus, to be capable of testing all current and future CubeSat attitude control systems, the maximum uncompensated disturbance torque of the CSACS is defined on the order of 10^{-5} N·m.

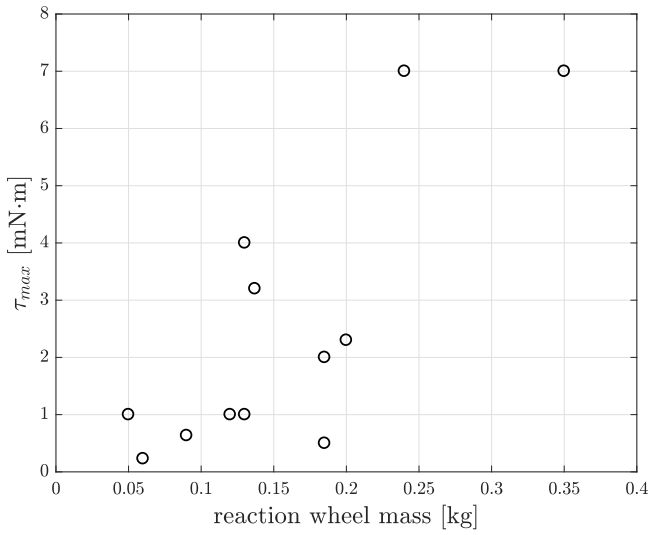


Figure 1.3: Maximum Reaction Wheel Torque vs. Mass. CubeSat reaction wheel data acquired from Blue Canyon Technologies, CubeSpace, Maryland Aerospace, Sinclair Interplanetary, and NanoAvionics. Note that τ_{max} represents the maximum torque of a single reaction wheel.

1.2.3 Magnetic Field Simulation Requirements

The goal of this system component is to simulate the magnetic field along a satellite's orbit, thereby enabling the calibration and testing of CubeSat magnetometers and magnetorquers in a laboratory setting, prior to flight. To achieve magnetic field simulation, we first require the simulator to nullify Earth's magnetic field locally; then it must create a comparable magnetic field to that of a low Earth orbit [38]. At Blacksburg's latitude, the Earth's magnetic field intensity is approximately 0.5 Gauss, and the maximum field strength required for simulation happens to also be approximately 0.5 Guass, as shown in Figure 1.4. Thus, accounting for a margin of safety, a magnetic field strength of approximately ± 2 Gauss is estimated as a maximum requirement for the Helmholtz cage, which is similar to the fixed design goals proposed in numerous reports in which the design of Virginia Tech's own Helmholtz Cage owes much credit [3, 21, 40].

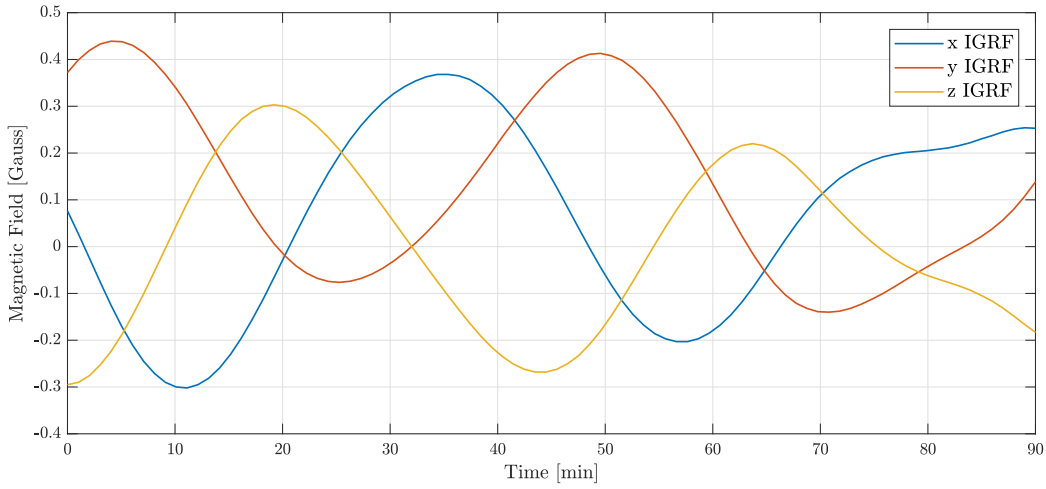


Figure 1.4: Earth Magnetic Field in Simulated Low Earth Orbit. International Geomagnetic Reference Field (IGRF) simulation produced using Systems Tool Kit (STK).

Since the simulator will eventually be used with more than one magnetometer, as well as even larger sets of hardware for end-to-end testing procedures, another requirement of the system is magnetic field homogeneity. For this current design iteration, generating a volume of constant magnetic field in the center of the cage is desired. Thus, to test multiple devices at the same time, along with accommodating the increasing trend of CubeSat payload size, the desired volume of magnetic field homogeneity will be fixed to a $30 \times 30 \times 30$ cm cube.

The choice of square coils has several key advantages. Working with a square design grants any operator with a more practical manufacturing and assembly procedure. Furthermore, a square coil provides a larger homogeneous magnetic field between sets of coils, ultimately generating a larger homogeneous magnetic field volume [38]. To drive the Helmholtz coils, three independent linear current supplies are utilized, which are controlled via digital interface, enabling PC interaction and data storage.

1.3 Objectives

Due to the unique nature of the problem, objectives for attitude control simulation, as well as magnetic field simulation, are established to drive all design considerations. Throughout the design of both simulators, affordability is kept in mind, such that any university can reproduce them and add their own testing contributions to the CubeSat community.

And so, summarizing the overall research objectives, we require:

- Two simulators, operating in conjunction, that provide a simulated LEO environment for CubeSat ADCS validation
- Affordable designs
- Open-source mentality

1.3.1 CSACS

The driving design constraint is the utilization of a black box approach. Due to the rapid growth of the CubeSat community and computational technologies, it is necessary to design a simulator that can successfully test CubeSat frames and attitude control systems that do not currently exist.

Modularity is also an important objective. All aspects of the simulator must be able to be simply modified or replaced. It is impossible to envision all future experiments that the CSACS may enable, but by designing the platform such that components can be swapped or modified, we can allow for new ideas and technologies to be featured. Future applications are numerous, and this feature enables risk-taking and critical experimentation.

The overarching goal of the CSACS is to dynamically balance a platform such that its center of mass (CM) is perfectly aligned with its center of rotation (CR). By completing such a task, it is possible to eliminate gravity from acting as a disturbance torque on the test article and will allow for more reliable results. Therefore, the platform must be able to identify its current angular position and subsequent CM and correct it, such that it is aligned with the air bearing CR.

Batteries are relatively heavy and require frequent charging, so the number on the platform must be minimized and carefully placed. The simulator must also be able to regulate and distribute voltages to the various hardware components which also have varying voltage and current draw requirements. Autonomous control through an on-board computing system is also required. The software must be supported by a microprocessor or microcontroller and operate with and without a user in the loop.

Summarizing the aforementioned objectives for the CSACS:

- Simulate the weightlessness and nearly-frictionless environment of space
- Utilize components that can be easily modified or replaced
- Operate autonomously for the duration of at least one full low Earth orbit

1.3.2 Helmholtz Cage

As for the Helmholtz cage, the overarching goal is to simulate the magnetic field comparable to that of a satellite in LEO. Keeping in mind the objective of modularity, the desired volume of magnetic field homogeneity inside the cage will be fixed to a $30 \times 30 \times 30$ cm cube. This will also allow for the cage design to be large enough, such that the CSACS can operate with full range of motion inside of it. Coupled with the CSACS, the Helmholtz cage must be able to operate for over an hour (90 min, preferred), which is the approximate time of one full orbit in LEO. Autonomous control, with and without a user in the loop, is also required to drive the Helmholtz coils.

Once again, summarizing the objectives for the Helmholtz cage:

- Simulate a magnetic field comparable to that of a satellite in LEO
- Operate autonomously for the duration of at least one full low Earth orbit

With the aforementioned objectives in mind and through design and modeling, trade-off studies, and analysis, advancements in the design and development of CubeSat attitude determination and control testing have been made. The remainder of this thesis will focus on the specifics of the platform and Helmholtz cage, along with simulated results.

Chapter 2

Literature Review

2.1 Spherical Air Bearing Systems

Air bearing spacecraft simulators have been used for attitude determination and control development and verification for nearly 60 years [47]. A preferred method of testing is performed by mounting a test article, *i.e.*, satellite, on a spherical air bearing. This type of system passes compressed air through small holes on the concave surface of a stator, thereby generating a thin cushion of air that supports a rotor, as shown in Figure 2.1.

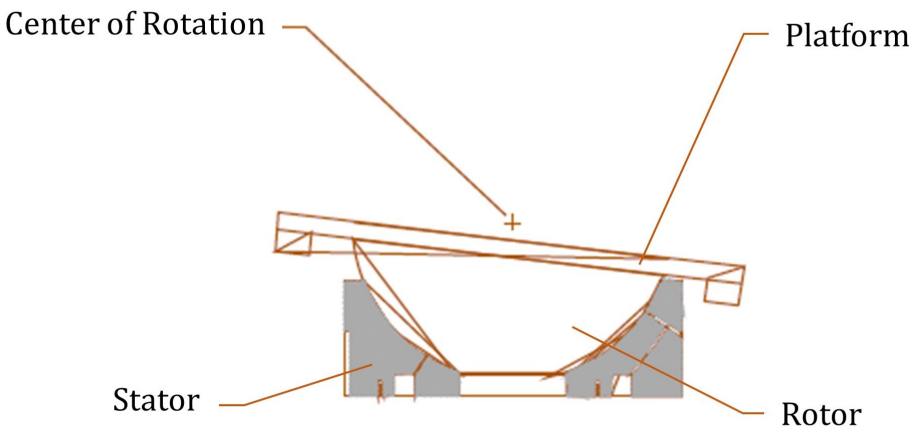


Figure 2.1: Hemispherical Air Bearing Schematic. The stator and air bearing (rotor) components are labeled accordingly, where the maximum tilt angle from the horizontal is limited by the platform mounted on top of the hemispherical air bearing interfering with the air bearing pedestal [2].

The thin cushion of air separating the stator and rotor provides an extremely low friction pivot with three degrees of freedom (DOF) that creates a nearly torque-free environment, such that the motion of the composite system in response to a force is only limited by the mass and moment of inertia of the test article [2]. By adjusting the structure so that the composite CM of the air bearing, fixture and satellite is coincident with CR of the air bearing, simulated weightlessness is achieved. Once in a weightless environment, the attitude control system (ACS) of a satellite can be fully tested before flight, and for these reasons, spherical air bearings are the preferred testing platform for ground-based attitude dynamics and control research. Ideally, a spherical air bearing testbed would offer the satellite unconstrained angular motion in the yaw, pitch, and roll axes. However, due to payload volume constraints and physical interference of the air bearing pedestal, unconstrained angular motion is not easily achievable. Careful design of the air bearing support system can increase the range of motion of the testbed; however, tilt angle from the horizontal plane is typically constrained to angles of less than $\pm 90^\circ$ as shown in Figure 2.2. Spherical air bearing testbeds are often custom built, and therefore come in many shapes and sizes. These varied configurations can be classified into three primary styles: tabletop, umbrella, and dumbbell, presented in Figures 2.3 and 2.4.

2.1.1 Tabletop

Tabletop-style platforms (Figure 2.3a) provide full 360° rotation in yaw, but constrain both pitch and roll rotation to less than $\pm 90^\circ$. The main platform of a tabletop-style system typically mounts to the top surface of a hemispherical air bearing, and then all components (including payload) are mounted directly to the main "tabletop" structure [47]. Because tabletop-style platforms can be fabricated inexpensively and typically require minimal floor space, they are ideal in university settings.

Examples of university developed tabletop-style air bearings follow. Georgia Tech developed a three reaction wheel based ACS platform for validation of various spacecraft control strate-

gies on the ground [19]. The Whorl-I, developed at Virginia Tech, contained a reaction wheel set plus cold gas thruster ACS to experimentally demonstrate formation flying coupled with the Whorl-II (Section 2.1.3) as a part of the Distributed Spacecraft Attitude Control System Simulator (DSACSS) [23, 46]. The National Autonomous University of Mexico (UNAM) built and designed a simulator that includes reaction wheels and magnetorquers to experimentally test spacecraft actuators, sensors, and their accompanying algorithms [41]. Similar to the UNAM air bearing platform, the Massachusetts Institute of Technology (MIT) developed a spacecraft simulator that mounts individual ADCS components on the testbed for small satellite (up to 180 kg ESPA-class) testing [6]. The small satellite attitude control simulator (SSACS), developed at Utah State University, implemented an adaptive balancing scheme for mass property estimation [36, 66]. Lastly, the Cal Poly Spacecraft Attitude Dynamics Simulator (CP/SADS) has been designed and continually iterated upon to simulate spacecraft attitude control using four reaction wheels as actuation [34].

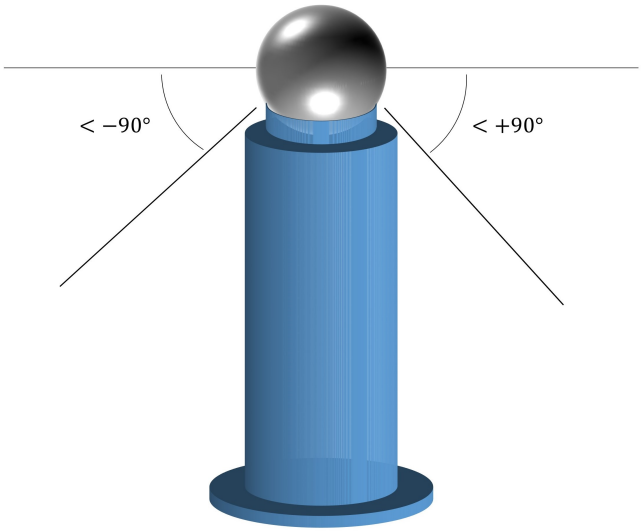
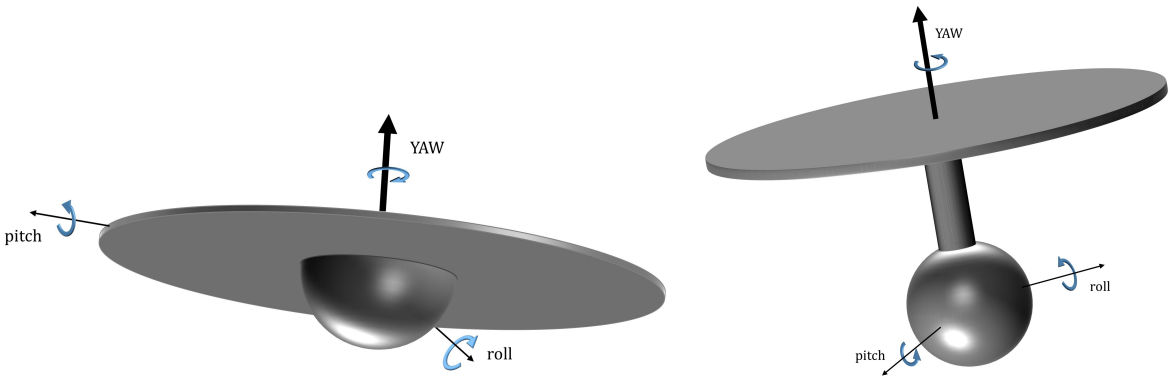


Figure 2.2: Spherical Air Bearing Pedestal Interference. Tabletop- and umbrella-style platforms (Figures 2.3a and 2.3b, respectively) provide unconstrained angular motion in yaw; however, pitch and roll motion are constrained to less than ±90°. Dumbbell-style configurations (Figure 2.4) provide unconstrained angular motion in both the yaw and roll axes, but pitch motion is still constrained to less than ±90° due to pedestal interference.



(a) Tabletop **(b)** Umbrella

Figure 2.3: Tabletop- and Umbrella-Style. Unconstrained yaw motion platforms. Note that the yaw axis is defined as nominally aligned with the gravity vector, with roll and pitch axes indistinguishable for the tabletop and umbrella systems. Both the hemispherical bearing for the tabletop and the fully spherical bearing for the umbrella rest on a pedestal, shown in Figure 2.2.

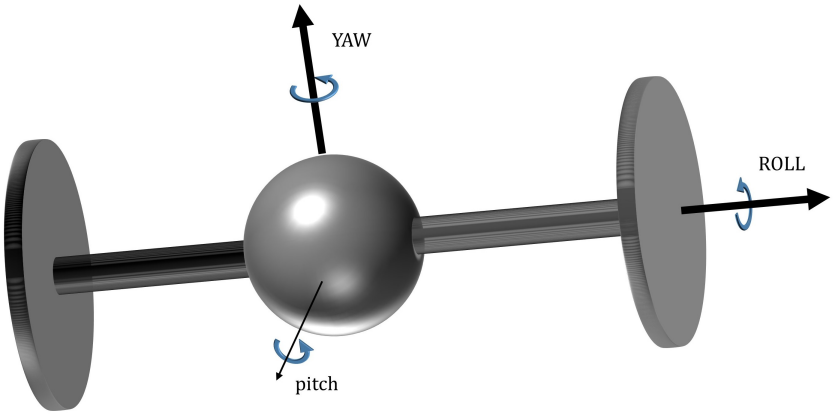


Figure 2.4: Dumbbell-Style. Unconstrained yaw and roll motion platform. For dumbbell systems, the yaw axis is defined as nominally aligned with the gravity vector, and the roll axis defined by the dumbbell arms. Again, the fully spherical bearing for the dumbbell rests on a pedestal, shown in Figure 2.2.

2.1.2 Umbrella

Similar to the tabletop, umbrella-style platforms (Figure 2.3b) provide full 360° rotation in yaw, but constrain both pitch and roll rotation to angles less than ±90°. Umbrella-style systems incorporate an extended rod attached to the top of a spherical air bearing. The main structure mounts to the top of the extended rod, and then all components (including payload) extend outward and downward, enclosing the air bearing and pedestal like an "umbrella" [47].

Examples of umbrella-style air bearings follow. The Honeywell, Inc. Momentum Control System and Line of Sight (MCS/LOS) air bearing platform incorporates structural control through active vibration isolation and discrete high-performance structural dampers, and is equipped with six CMGs to offer high-agility slew and scan capability [37]. The United States Air Force Research Laboratory (AFRL) used its Advanced Space Structure Technology Research Experiments (ASTREX) facility as an air bearing test bed for validation and integration of structural control [8]. The Naval Postgraduate School (NPS) has used an air bearing equipped with rate gyros, cold-gas thrusters, and reaction wheels to validate attitude stabilization control [42]. Lastly, AFIT developed their own hybrid tabletop/umbrella-style air bearing for attitude control research and development [31].

2.1.3 Dumbbell

Dumbbell-style systems (Figure 2.4) offset the payload and ballast mounting surfaces away from the CR by means of extending rods, mimicking the shape of a "dumbbell" [47]. This testbed allows for full 360° rotation in yaw and roll, with constrained pitch motion to less than ±90°.

Examples of university developed dumbbell-style air bearings follow. The Triaxial Attitude Control System (TACT), developed at the University of Michigan, utilized a reaction wheel/cold-gas thruster ACS to investigate unknown mass property determination [5]. The

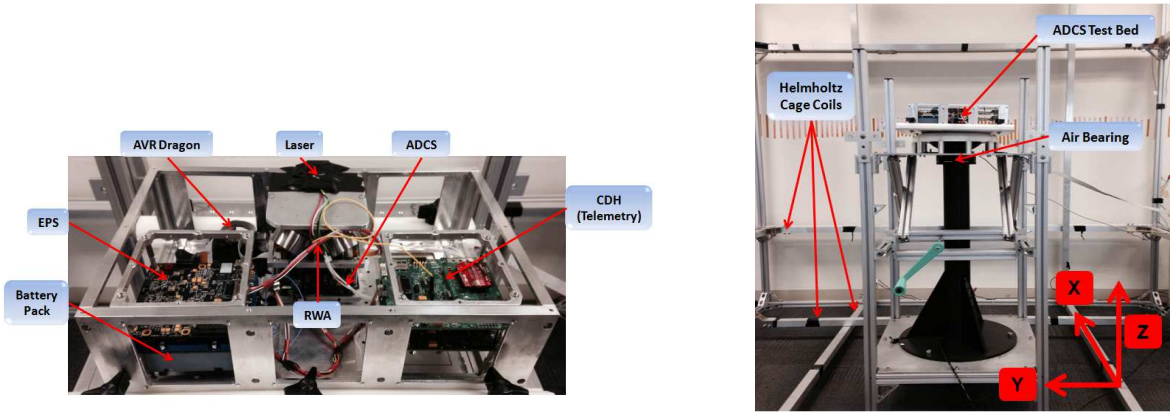
Whorl-II, developed at Virginia Tech, contained a reaction wheel set plus cold gas thruster ACS to experimentally demonstrate formation flying as a part of the DSACSS [46]. Lastly, the SimSat I, developed at AFIT, incorporated a reaction wheel ACS to support attitude control, precision pointing, and vibration suppression experimentation [14].

2.2 CubeSat ADCS Simulators

While all of the aforementioned spacecraft attitude control simulators have made significant contributions for laboratory attitude control testing, their total masses would dominate any applied torque that a CubeSat attitude control system would contribute. Thus, the growth of the CubeSat community has led to the recent development of appropriately scaled attitude control simulators, coupled with magnetic field simulators, enabling full CubeSat ADCS laboratory testing.

Air Force Institute of Technology

Through multiple design iterations of both dumbbell- and tabletop-style spacecraft simulators, researchers at AFIT have contributed to the development of a 6U CubeSat ADCS testbed [7, 56]. A 6U chassis is outfitted with three magnetorquers and a pyramidal reaction wheel array (described in detail in Section 3.2) for actuation, self contained battery pack and EPS, and command and data handling, for full three-axis simulation, shown in Figure 2.5a. The 1U-sized, 1.24 kg reaction wheel array (RWA) is capable of producing 15 mN·m torque. The torque rods fit within the walls of the 6U testbed and have a maximum magnetic moment of 0.60 $A{\cdot}m^2$. Currently, only yaw-axis control has been established. The air bearing is contained in a custom built Helmholtz cage for full magnetic field simulation, shown in Figure 2.5b. The $2.4 \times 2.4 \times 2.4$ m square coil design creates a 0.027 m^3 ($30 \times 30 \times 30$ cm) volume of magnetic field homogeneity, with a maximum field strength of ± 2 Gauss [3].



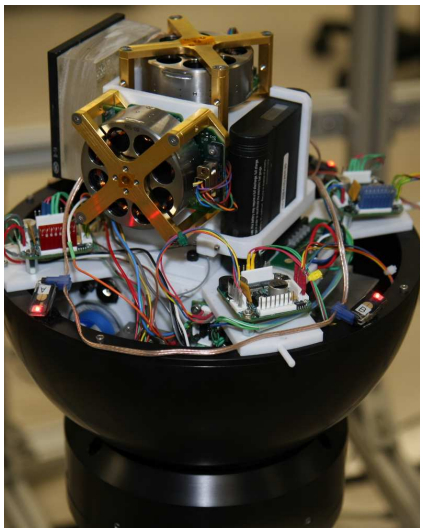
(a) AFIT 6U ADCS testbed [7, 56]

(b) AFIT 6U CubeSat testbed coupled with Helmholtz cage for magnetic field simulation [3]

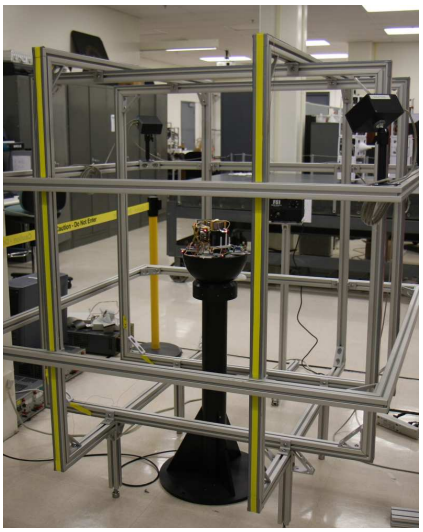
Figure 2.5: AFIT CubeSat ADCS Simulator

Naval Postgraduate School – UC Santa Cruz

The Naval Postgraduate School (NPS) with collaborations from UC Santa Cruz, developed the CubeSat three axis simulator (CubeTAS), as shown in Figure 2.6a. The CubeSat-scale air bearing incorporates an automated mass balancing system that aligns the composite system CM with the air bearing's CR, eliminating gravitational torque and enabling three-axis control testing [20]. Through the use of active mass-shifting control, experimental results demonstrated capabilities of the testbed to successfully simulate CubeSat rotational dynamics, as well as perform three-axis stabilization maneuvers [4]. The system also includes a Helmholtz cage for three-axis magnetic field simulation, as shown in Figure 2.6b. The $1.30 \times 1.38 \times 1.46$ m square coil design creates a 0.327 m^3 (approx. $69 \times 69 \times 69$ cm) volume of magnetic field homogeneity, with a maximum field strength of ± 2 Gauss [64]. Future work on the CubeTAS is said to include the validation of attitude control scenarios that incorporate magnetorquers, *e.g.*, orbit-insertion detumbling and reaction wheel desaturation via momentum dumping.



(a) CubeTAS air bearing [4, 20]

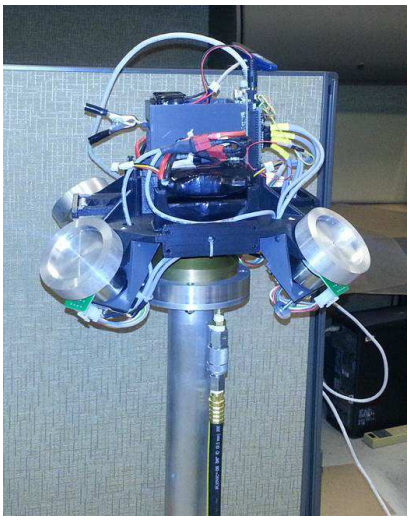


(b) CubeTAS assembly coupled with Helmholtz cage for magnetic field simulation [64]

Figure 2.6: NPS–UC Santa Cruz CubeSat ADCS Simulator

Utah State University – Space Dynamics Laboratory

Utah State University (USU) has developed a tabletop-style small satellite simulator, which can be coupled with a Helmholtz cage built at the USU Space Dynamics Laboratory (SDL), for full ADCS simulation testing, as shown in Figure 2.7. The attitude control simulator uses inertial sensors for attitude determination and a four-wheel momentum exchange system for control. A linear control model with trajectory generation and feedback linearization was implemented for yaw axis control [45]. The $1.85 \times 1.95 \times 2.05$ m square coil design creates a 0.227 m^3 (approx. $61 \times 61 \times 61$ cm) volume of magnetic field homogeneity, with a maximum field strength of ± 7.5 Gauss [44].



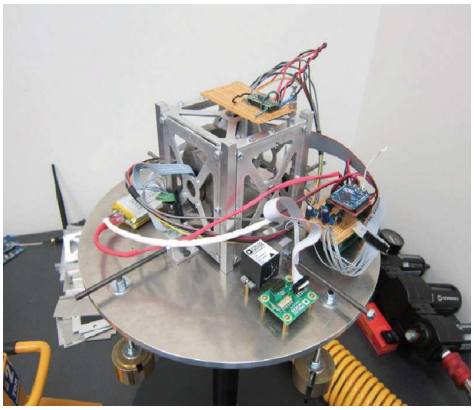
(a) USU small-satellite air bearing platform [45]

(b) SDL Helmholtz cage for magnetic field simulation [44]

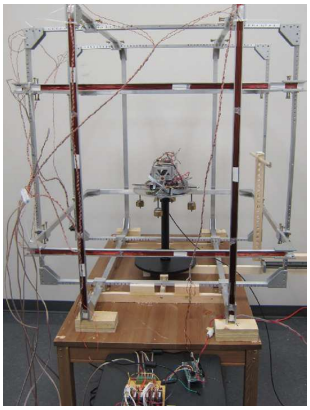
Figure 2.7: USU–SDL CubeSat ADCS Simulator

York University

York University in Toronto, Ontario developed a tabletop-style air bearing platform with custom built actuators, which all fit within a 1U CubeSat. The actuators include three reaction wheels and three magnetorquers [29]. The 0.214 kg reaction wheels are controlled by Faulhaber brushless flat micro-motors, capable of producing 0.6 mN·m torque. The torque rods fit within the walls of the 1U configuration and have a maximum magnetic moment of 0.37 A·m^2. The platform holds the 1U stack including actuators, a manual balancing system, OBC, transceiver, IMU, power distribution board, and batteries, as shown in Figure 2.8a. The manual balancing system was shown to cause some undesired gravitational torques evident in the test data; however, control algorithms were successfully tested using the in-house actuators. A Helmholtz cage is coupled with the air bearing platform to enable full CubeSat ADCS simulation, shown in Figure 2.8b. The $0.9652 \times 1.016 \times 1.0668$ m square coil design creates a 0.027 m^3 (approx. $30 \times 30 \times 30$ cm) volume of magnetic field homogeneity, with a maximum field strength of ± 1 Gauss [65].



(a) Tabletop-style air bearing for 1U CubeSat ACS testing [29]



(b) Air bearing assembly coupled with Helmholtz cage for magnetic field simulation [65]

Figure 2.8: York University CubeSat ADCS Simulator

Chapter 2 presents a literature review of both univeristy- and industry-designed spacecraft ADCS simulators. Section 2.1 introduces the three common configurations of spherical air bearing systems: tabletop, umbrella, and dumbbell. A review of both university- and industry-designed spherical air bearing spacecraft simulators follows, which shows the importance of laboratory attitude control testing. Lastly, Section 2.2 presents four separate university-developed CubeSat attitude control and magnetic field simulators. These university-developed CubeSat ADCS simulators emphasize the recent growth of the CubeSat community, and serve as a baseline for the design and development of Virginia Tech's own CubeSat attitude control simulator coupled with a square Helmholtz cage for magnetic field simulation.

Chapter 3

System Design Overview

3.1 CSACS

3.1.1 Air Bearing

A legacy Space Electronics, Inc. hemispherical gas air bearing, similar to the schematic shown in Figure 2.1, provides the nearly frictionless environment for the CSACS system to operate. The flat portion of the hemisphere provides a surface for the CSACS platform and allows for a limited 3–DOF during testing. The air bearing can support a 300 lb payload, and allows for a full 360° rotation in yaw, with ±15° tilt from the horizontal in pitch and roll. From an internal friction study by Space Electronics, Inc., there exists 0.01 mN·m of gravitational torque if the CM of the test article is aligned with the CR of the bearing within 10 millionths of an inch, which meets the maximum uncompensated disturbing torque requirement proposed in Section 1.2.2.

3.1.2 Tabletop-Style Platform

All hardware is mounted on a custom carbon fiber composite platform that is subsequently mounted to the flat surface of the hemispherical component of the air bearing, as shown in Figure 3.1. A 0.5 m diameter disk is used for a 6U platform.

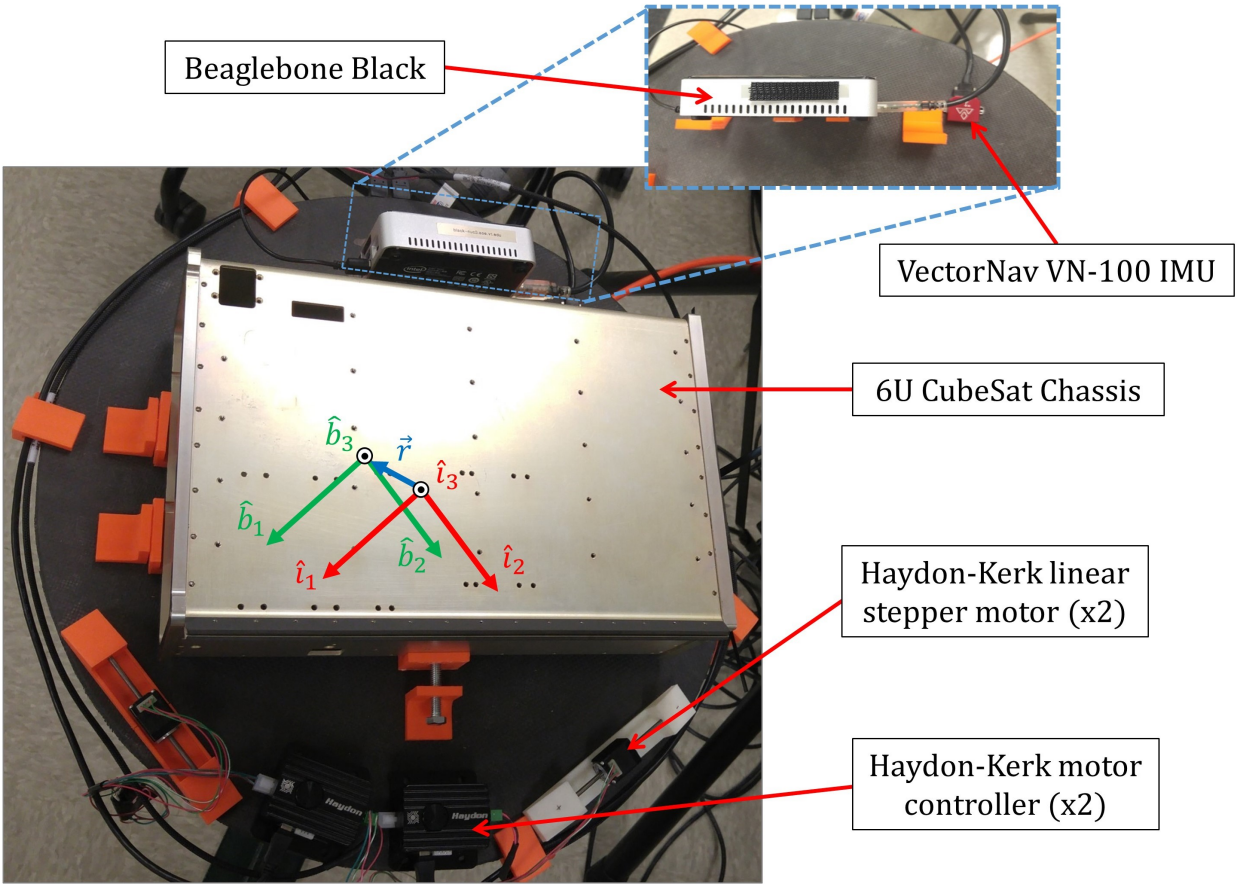


Figure 3.1: The Assembled CubeSat Attitude Control Simulator Platform. The carbon fiber platform holds all of the CSACS control hardware, and rests on top of a similar air bearing configuration, as shown in Figure 2.3a. This figure also contains a 6U CubeSat chassis, which would house all of the flight hardware of the system. Not shown in the figure are the battery packs mounted underneath the platform, along with the power distribution board. The origins of the inertial frame (red) and the body-fixed frame (green) are located at the air bearing CR and the CSACS CM, respectively. The r vector (blue) is the distance from the CR to the CM. Note that the linear stepper motors are aligned with the $\hat{b}_1$ and $\hat{b}_2$ axes.

3.1.3 Computing

A Beaglebone Black – Rev C is the command and control unit on-board the simulator. The Beaglebone features a Texas Instruments AM3358 1 GHz ARM® Cortex-A8 Processor, 512 MB of DDR3L DRAM, 4 GB onboard flash memory, a Linux operating system, and a USB host. The board runs on 5 VDC [1]. These specifications provide for future capability expansion and modularity, as well as meeting future software and control requirement for years to come.

3.1.4 Inertial Measurement Unit

The VectorNav VN-100 Rugged IMU (Figure 3.1) is used to measure the angular position of the platform. Using the latest solid-state microelectromechanical system (MEMS) technology, the VN-100 combines a set of 3-axis accelerometers, 3-axis gyroscopes, 3-axis magnetometers, a barometric pressure sensor, and a 32-bit processor for attitude determination [59]. The VN-100 is capable of providing the angular position of the CSACS in pitch, roll, and yaw to the hundred-millionth degree, which is required knowledge to eliminate parasitic gravitational torque. Currently, an updated Python library version is incorporated in the CSACS software for communication with the IMU and simple read/write software capabilities.

3.1.5 Stepper Motors and Controllers

The CSACS platform contains two sets of Haydon-Kerk stepper motors and controllers (Figure 3.1) which successfully align the composite CSACS CM within approximately 10 millionths of an inch (1×10^{-5} in.) of the air bearing's CR, to eliminate gravitational torques. The Haydon-Kerk PCM-4806E micro-controller has a built-in encoder and can control the motors to 1/64 step, providing a total resolution of 7.85×10^{-7} in. The Haydon-Kerk 21F4U2.5-ENG linear stepper motor has a step size of 6×10^{-5} in. over a 4 in. screw.

The motor assemblies have a custom designed 3D printed mount, as the screw must be fixed to allow for precise movement of the motor. The capabilities of these motors allow for the precision CM alignment with the CR of the air bearing.

3.1.6 Electrical Power System

A power distribution board successfully handles the current loads of the platform, while taking in a significantly higher voltage from two lithium-ion batteries which subsequently drops the voltages for each individual component in the CSACS system. Updates to the power distribution board include four outputs and allowance for expansion of additional components. The circuit takes in 14.8 V from two 7.4 V 2200 mA·h lithium-ion battery packs wired in series. The line voltage goes straight to a terminal and to the first LM338T linear voltage regulator. The circuit uses this regulator and resistor to drop the voltage to 12 V for each motor controller. The 12 V also goes to another linear voltage regulator that uses an identical circuit to drop the voltage down to 5 V for the Beaglebone.

3.1.7 Software

The platform is controlled using a real-time script developed in the Python programming language. The software provides the user with the ability to visualize the movement of the platform in real time, providing dynamic feedback between the IMU and stepper motors, and data recording for supplemental analysis. At its core, the program uses the `matplotlib` module in Python to plot the yaw, pitch, and roll values from the VectorNav every half second. Real-time, dynamic updates allow the user to visually determine the time necessary for complete balance and to monitor the platform in real time. The software displays the actual values of the VectorNav so that manual balancing can be achieved. Additionally, the software can communicate with the motor controllers through ASCII commands for automated table balancing.

3.2 Pyramidal Reaction Wheel Array

Once the CSACS CM is aligned with the air bearing's CR and no gravity torques are acting on the simulator, reaction wheels can be incorporated into the system for dynamic control of the platform. Reaction wheel actuation simply consists of a flywheel attached to a motor. The motor drives the flywheel in either direction at variable speeds to induce a torque on a body, while conserving the overall angular momentum of the system. Three reaction wheels, with each wheel's rotational axis aligned to each principal body axis of a spacecraft, define the simplest three-axis attitude control system. However, if any of the three reaction wheels do not perform nominally, then attitude cannot be controlled. For this reason, it is common to include a fourth wheel for redundancy and increased reliability. The additional wheel is installed with it's rotational axis misaligned with each of the spacecraft's principal body axes, which subsequently enables reduced control about any one of the principal axes. This way, if any one reaction wheel that is aligned with the spacecraft's principal axes fails, control is compensated by the torque provided from the off-axis wheel. For control of the CSACS, a four wheel pyramidal configuration, designed and developed at AFIT, will be implemented.

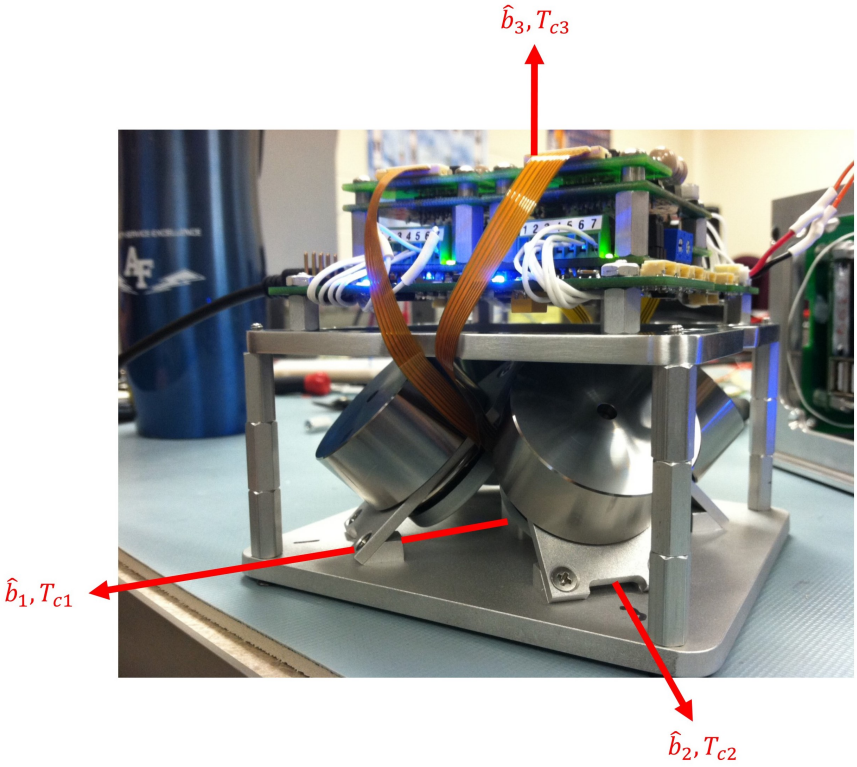


Figure 3.2: Four Wheel Pyramidal Reaction Wheel Array. The Air Force Institute of Technology designed, four wheel reaction wheel array (RWA) for overactuated attitude control of a 6U CubeSat [7, 56]. The torques produced along the three body axes are T_{c1}, T_{c2}, T_{c3}. The rotational axes of all four wheels are inclined to the $\hat{b}_1 - \hat{b}_2$ plane by an angle β, with wheels 1 and 3 aligned facing the $+\hat{b}_1$ and $-\hat{b}_1$ direction, respectively, and wheels 2 and 4 aligned facing the $+\hat{b}_2$ and $-\hat{b}_2$ direction, respectively.

The reaction wheel assembly (Figure 3.2) is an older generation model designed for AFIT's 6U CubeSat testbed. Each reaction wheel has a moment of inertia (MOI) of approximately 2.4×10^{-5} kg·m^2 and an inclination of 45° from the $\hat{b}_1 - \hat{b}_2$ plane. Four Maxon EC 32 flat 6W motors (Part Number 339259) drive the reaction wheels, and a pair of motors are controlled by Maxon DEC 24/1 digital amplifiers (Part Number 249630). An Atmel AVR® 32-bit microcontroller (Model AT32UC3C0512C) controls the RWA, which is also outfitted with an Analog Devices High Precision Tri-Axis IMU (Model ADIS15405). Dannemeyer predicted

the nominal and maximum torques in the $\hat{b}_1$- and $\hat{b}_2$-axes to be 15.02 and 31.0 mN·m, respectively; and the nominal and maximum torque in the $\hat{b}_3$-axis to be 21.24 and 43.84 mN·m, respectively [7]. The RWA operates at 6.3 W (steady-state), powered by an EPS stack configured with a 40 W·h battery pack. The combined mass is 1.24 kg, which supports the CSACS design requirement of lightweight components. After vibration qualification testing conducted by AFIT, the motors became disfunctional, meaning that this RWA is not flight ready. Despite being unqualified for flight, the RWA is still very useful in a laboratory setting for CubeSat attitude control development and testing. Additional information regarding the RWA can be found in Dannemeyer and Tibbs' Master's Theses [7, 56].

Sections 4.2 and 5.2 will present the development and simulation of an attitude control system for the CSACS, coupled with the pyramidal RWA. The controller is designed and capable of full, three-axis attitude control of a spacecraft; however, only yaw control will be considered due to the gravitational torques that develop when the CSACS is commanded to nonzero pitch or roll orientations. Beyond yaw-maneuvering capabilities, the RWA simply does not possess pitch and roll control authority due to the physical properties of the CSACS. Recommendations for three-axis control are presented in Chapter 6; however, we note that simply incorporating larger reaction wheels in the system reverts back to the overarching challenge of CubeSat attitude control simulation: if the simulator itself is too large, the mass of a CubeSat will be negligible compared to the overall mass of the simulator. Yaw control still presents itself very useful however, both for single-axis attitude control testing, as well as MOI determination.

3.3 Helmholtz Cage

On the basis of the results achieved from mathematical modeling (Section 4.3), the construction of a Helmholtz cage follows. To meet the design requirements, it is determined that the cage will have to have minimum dimensions of 1.5 × 1.5 × 1.5 m. The support structure for the cage is constructed of 1 inch T-slotted aluminum framing. This framing was chosen for its ease of assembly. Furthermore, the slotting allows for the coils to be moved after they are mounted in order for testing hardware to be easily removed or placed inside. The coils are constructed of U-channel aluminum that is 1 inch deep and 1.5 inches wide. The wire used is 12 AWG copper Thermoplastic High Heat-resistant Nylon (THHN). Each coil contains approximately 35 turns of wire. The corners of each coil are constructed of U-channel aluminum that has been notched and bent to a 5 inch radius. The largest coil is 1.5 m square, the medium coil is 1.4492 m, and the smallest is 1.3984 m. Three power supplies are used to drive the coils. These power supplies are Keithley model 2260B-30-36 Multi-range DC power supplies. Each power supply is capable of operating at up 30 amperes and 100 volts for a maximum output of 300 watts.

Table 3.1: Mechanical and Electrical Coils Characteristics. Helmholtz cage dimensions, coil windings, operating amperage, and voltage that achieve the 30 × 30 cm – 2 Gauss design requirement.

	Large Coil (z axis)	Medium Coil (x axis)	Small Coil (y axis)
Dimensions (m)	1.5	1.4492	1.3984
Coil Winds	35	34	33
Operating Amperage (A)	5.32	5.32	5.32
Operating Voltage (V)	30	30	30

The location of the cage greatly effects the initial magnetic field within it when powered, as its orientation to north as well as the objects around it can shift the net magnetic field magnitude and direction. A power supply for each coil is connected to a set of relays and an Arduino to allow the polarity of the field to switch between positive and negative. A software solution was created to obtain IGRF data from a simulated flight path in STK and control the power supply as needed. Then, a graphical user interface (GUI) easily handles variables that need to be changed on a per-usage basis, while also displaying data that is useful to know for the experiment itself.

Chapter 3 presents the system design overview of the CSACS, pyramidal reaction wheel array, and Helmholtz cage, which allows for full CubeSat ADCS simulation in a laboratory setting. Section 3.1 introduces the CSACS platform, which is a lightweight, tabletop-style spherical air bearing platform, designed using a black-box approach. A VectorNav VN-100 IMU determines the roll, pitch, and yaw of the interchangeable carbon-fiber test platforms, and linear controlled actuators balance the table within approximately 10 millionths of an inch of the CR of a hemispherical gas air bearing. A power regulation board, designed for system modularity, generates power across the platform's hardware. Embedded software controls the capabilities that provide the user with real-time platform balancing. Section 3.2 introduces the AFIT designed, four wheel RWA for overactuated attitude control of a 6U CubeSat. The incorporation of this ADCS will provide active, single-axis yaw control of the CSACS platform after the system has been balanced. Lastly, Section 3.3 presents a $1.5 \times 1.5 \times 1.5$ m square Helmholtz cage design. When fully functional, the CSACS platform will operate inside of the Helmholtz cage, where a $30 \times 30 \times 30$ cm volume of magnetic field homogeneity, with a maximum field strength of ± 2 Gauss, is capable of being generated for full LEO simulations.

Chapter 4

Mathematical Modeling

4.1 CSACS

4.1.1 Rotational Kinematics

Rotation matrices are direction cosine matrices that allow any vector to be expressed in any reference frame. To properly transform the CSACS system dynamics between inertial and body-fixed frames, a 3-2-1 rotation sequence is used. Consider the rotation from $\mathcal{F}_i$ to $\mathcal{F}_b$, using three *Euler angles* denoted ψ, θ, and ϕ, which correspond to yaw, pitch, and roll, respectively, as developed in [16] and [52].

The first rotation is about the $\hat{\mathbf{i}}_3$ axis, by angle ψ. The resulting frame is $\{\hat{\mathbf{i}}'\}$ or $\mathcal{F}_{i'}$, as shown in Figure 4.1a. The rotation matrix from $\mathcal{F}_i$ to $\mathcal{F}_{i'}$ is

$$\mathbf{R}^{i'i} = \mathbf{R}_3(\psi) = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \Rightarrow \mathrm{v}_{i'} = \mathbf{R}_3(\psi)\mathrm{v}_i \tag{4.1}$$

The second rotation is about the $\hat{\mathbf{i}}'_2$ axis, by angle θ. The resulting frame is $\{\hat{\mathbf{i}}''\}$ or $\mathcal{F}_{i''}$, as shown in Figure 4.1b. The rotation matrix from $\mathcal{F}_{i'}$ to $\mathcal{F}_{i''}$ is

$$\mathbf{R}^{i''i'} = \mathbf{R}_2(\theta) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \Rightarrow \mathrm{v}_{i''} = \mathbf{R}_2(\theta)\mathrm{v}_{i'} = \mathbf{R}_2(\theta)\mathbf{R}_3(\psi)\mathrm{v}_i \tag{4.2}$$

with $\mathbf{R}^{i''i} = \mathbf{R}_2(\theta)\mathbf{R}_3(\psi)$ being the rotation matrix transforming vectors from $\mathcal{F}_i$ to $\mathcal{F}_{i''}$.

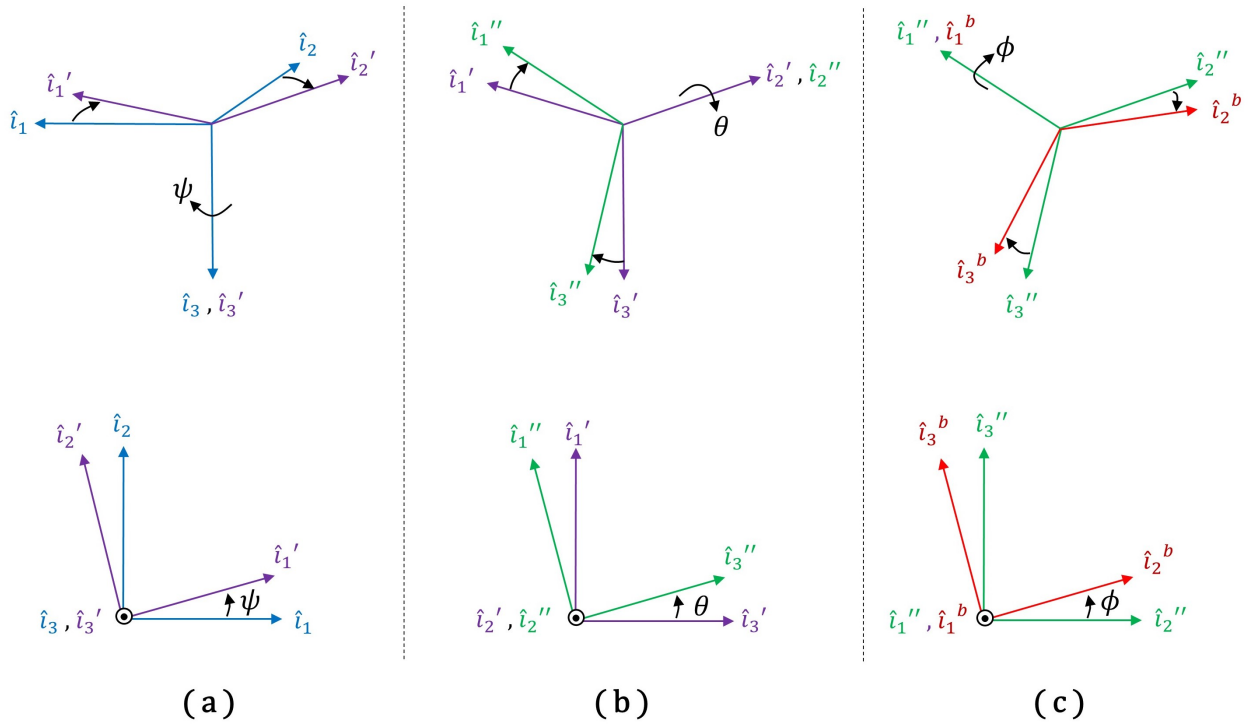


Figure 4.1: 3-2-1 Euler Angle Rotation. **(a)** First rotation of angle ψ around the $\hat{\mathbf{i}}_3$ axis. **(b)** Second rotation of angle θ around the $\hat{\mathbf{i}}_2'$ axis. **(c)** Third rotation of angle ϕ around the $\hat{\mathbf{i}}_1''$ axis [50].

The third rotation is about the $\hat{\mathbf{i}}_1''$ axis, by angle ϕ. The resulting frame is $\{\hat{\mathbf{b}}\}$ or $\mathcal{F}_b$, as shown in Figure 4.1c. The rotation matrix from $\mathcal{F}_{i''}$ to $\mathcal{F}_b$ is

$$\mathbf{R}^{bi''} = \mathbf{R}_1(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \Rightarrow \mathrm{v}_b = \mathbf{R}_1(\phi)\mathrm{v}_{i''} = \mathbf{R}_1(\phi)\mathbf{R}_2(\theta)\mathbf{R}_3(\psi)\mathrm{v}_i \tag{4.3}$$

Here, we can see that $\mathbf{R}^{bi} = \mathbf{R}_1(\phi)\mathbf{R}_2(\theta)\mathbf{R}_3(\psi)$ is the rotation matrix transforming vectors from $\mathcal{F}_i$ to $\mathcal{F}_b$.

We have now developed the 3-2-1 rotation from an inertial coordinate system, $\mathcal{F}_i$, to a body-fixed system, $\mathcal{F}_b$. The expanded 3-2-1 rotation matrix is presented in Equation 4.4

$$\mathbf{R}^{bi} = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix} \tag{4.4}$$

Now, we will derive the angular velocities from the Euler angles. The Euler rate equations become part of the equations of motion for CSACS. The rate equations represent the time rate of change of the Euler angles in terms of the Euler angles and the body-fixed angular rates. The derivation for the 3-2-1 rotation is similar to the development of the rotation matrices above. We will develop the angular velocity one *frame-to-frame* at a time, *i.e.*, from $\mathcal{F}_i$ to $\mathcal{F}_{i'}$ to $\mathcal{F}_{i''}$ to $\mathcal{F}_b$ [53].

Again, the first rotation is about the $\hat{\mathbf{i}}_3 \equiv \hat{\mathbf{i}}'_3$ axis, through angle ψ. The resulting frame is $\{\hat{\mathbf{i}}'\}$ or $\mathcal{F}_{i'}$. The angular velocity of $\mathcal{F}_{i'}$ with respect to $\mathcal{F}_i$ is

$$\vec{\omega}^{i'i} = \dot{\psi}\hat{\mathbf{i}}_3 = \dot{\psi}\hat{\mathbf{i}}'_3 \tag{4.5}$$

We can express Equation 4.5 in any frame, but $\mathcal{F}_i$ and $\mathcal{F}_{i'}$ are trivial in this case

$$\begin{aligned} \omega_i^{i'i} &= \begin{bmatrix} 0 & 0 & \dot{\psi} \end{bmatrix}^T \\ \omega_{i'}^{i'i} &= \begin{bmatrix} 0 & 0 & \dot{\psi} \end{bmatrix}^T \end{aligned} \tag{4.6}$$

keeping in mind the notation that $\omega_i^{i'i}$ is the angular velocity of $\mathcal{F}_{i'}$ with respect to $\mathcal{F}_i$ expressed in $\mathcal{F}_i$.

The second rotation is about the $\hat{\mathbf{i}}'_2 \equiv \hat{\mathbf{i}}''_2$ axis, through angle θ. The resulting frame is $\{\hat{\mathbf{i}}''\}$ or $\mathcal{F}_{i''}$. The angular velocity of $\mathcal{F}_{i''}$ with respect to $\mathcal{F}_{i'}$ is

$$\vec{\omega}^{i''i'} = \dot{\theta}\hat{\mathbf{i}}'_2 = \dot{\theta}\hat{\mathbf{i}}''_2 \tag{4.7}$$

We can express Equation 4.7 in any frame, but $\mathcal{F}_{i'}$ and $\mathcal{F}_{i''}$ are trivial in this case

$$\begin{aligned} \omega_{i'}^{i''i'} &= \begin{bmatrix} 0 & \dot{\theta} & 0 \end{bmatrix}^T \\ \omega_{i''}^{i''i'} &= \begin{bmatrix} 0 & \dot{\theta} & 0 \end{bmatrix}^T \end{aligned} \tag{4.8}$$

keeping in mind the notation that $\omega_{i'}^{i''i'}$ is the angular velocity of $\mathcal{F}_{i''}$ with respect to $\mathcal{F}_{i'}$ expressed in $\mathcal{F}_{i'}$.

The third rotation is about the $\hat{\mathbf{i}}_1'' \equiv \hat{\mathbf{b}}_1$ axis, through angle ϕ. The resulting frame is $\{\hat{\mathbf{b}}\}$ or $\mathcal{F}_b$. The angular velocity of $\mathcal{F}_b$ with respect to $\mathcal{F}_{i''}$ is

$$\vec{\omega}^{bi''} = \dot{\phi}\hat{\mathbf{i}}_1'' = \dot{\phi}\hat{\mathbf{b}}_1 \tag{4.9}$$

We can express Equation 4.9 in any frame, but $\mathcal{F}_{i''}$ and $\mathcal{F}_b$ are trivial in this case

$$\begin{aligned} \omega_{i''}^{bi''} &= \begin{bmatrix} \dot{\phi} & 0 & 0 \end{bmatrix}^T \\ \omega_b^{bi''} &= \begin{bmatrix} \dot{\phi} & 0 & 0 \end{bmatrix}^T \end{aligned} \tag{4.10}$$

keeping in mind the notation that $\omega_b^{bi''}$ is the angular velocity of $\mathcal{F}_b$ with respect to $\mathcal{F}_{i''}$ expressed in $\mathcal{F}_b$.

Now, we can solve for ω_b^{bi} by simply adding the angular velocities. However, because we have expressed our angular velocities in different reference frames; to add them together, we need to express all of them in the same frame. We accomplish this by using our previously developed rotation matrices

$$\begin{aligned} \omega_b^{bi} &= \omega_b^{bi''} + \omega_b^{i''i'} + \omega_b^{i'i} \\ &= \omega_b^{bi''} + \mathbf{R}^{bi''}\omega_{i''}^{i''i'} + \mathbf{R}^{bi'}\omega_{i'}^{i'i} \\ &= \omega_b^{bi''} + \mathbf{R}^{bi''}\omega_{i''}^{i''i'} + \mathbf{R}^{bi''}\mathbf{R}^{i''i'}\omega_{i'}^{i'i} \end{aligned} \tag{4.11}$$

arriving at the angular velocity of $\mathcal{F}_b$ with respect to $\mathcal{F}_i$ expressed in $\mathcal{F}_b$

$$\omega_b^{bi} = \omega_b^{bi''} + \mathbf{R}_1(\phi)\omega_{i''}^{i''i'} + \mathbf{R}_1(\phi)\mathbf{R}_2(\theta)\omega_{i'}^{i'i} \tag{4.12}$$

Carrying out the multiplications and additions, Equation 4.12 can be expanded to

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} -\sin\theta & 0 & 1 \\ \cos\theta\sin\phi & \cos\phi & 0 \\ \cos\theta\cos\phi & -\sin\phi & 0 \end{bmatrix} \begin{bmatrix} \dot{\psi} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} \tag{4.13}$$

Writing Equation 4.13 in the form $\omega_b^{bi} = S(\mathbf{\Theta})\dot{\mathbf{\Theta}}$, we can then form a solution for the Euler angle rates by inverting $S(\mathbf{\Theta})$, *i.e.*, $\dot{\mathbf{\Theta}} = S^{-1}(\mathbf{\Theta})\omega_b^{bi}$. In expanded form,

$$\begin{bmatrix} \dot{\psi} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \\ 0 & \cos\phi & -\sin\phi \\ 1 & \frac{\sin\phi\sin\theta}{\cos\theta} & \frac{\cos\phi\sin\theta}{\cos\theta} \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{4.14}$$

For this specific Euler angle set (3-2-1), there exists a kinematic singularity when $\theta = 90°$. However, since the CSACS is mechanically constrained to $\pm 15°$ tilt from the horizontal in pitch and roll, the kinematic singularity is not a concern.

Note that in the remaining sections of this report, we will denote $\omega_b^{bi} = \begin{bmatrix} \omega_x & \omega_y & \omega_z \end{bmatrix}^T$ in shorthand notation ω_b, in order to reduce notation clutter.

4.1.2 Derivation of Equations of Motion

Following the work established in [55], the CSACS can be modeled as a rigid body with the CM placed at a distance r from the CR of the air bearing itself, as shown in Figure 4.2. The model uses an inertial coordinate frame $\{\hat{\mathbf{i}}\}$ that is fixed to the Earth, and a body-centered coordinate frame $\{\hat{\mathbf{b}}\}$ which is fixed to the CSACS. The angular momentum about the CR of the air bearing is

$$H = m_{tot}(r \times v) \tag{4.15}$$

where m_{tot} is the composite system mass, r is the vector from the CR of the air bearing to the CM of the CSACS, and v is the velocity of the CM [15]. First, we can express the time rate of change of angular momentum as

$$\frac{dH}{dt} = T_{ext} \tag{4.16}$$

where T_{ext} is the applied external torques. Then, when the CM is not located at the CR, the time rate of change of angular momentum can be expanded as

$$\frac{dH}{dt} = (r \times m_{tot}\ddot{r}) + \dot{H}_c + [\omega_b \times (r \times m_{tot}\dot{r})] + (\omega_b \times H_c) \tag{4.17}$$

where H_c is the angular momentum about the CM, and ω_b is the angular velocity of the body-fixed frame [15]. The assumptions required to use Equation 4.17 include: the translational acceleration of the CR is zero and the CSACS is modeled as rigid body [15].

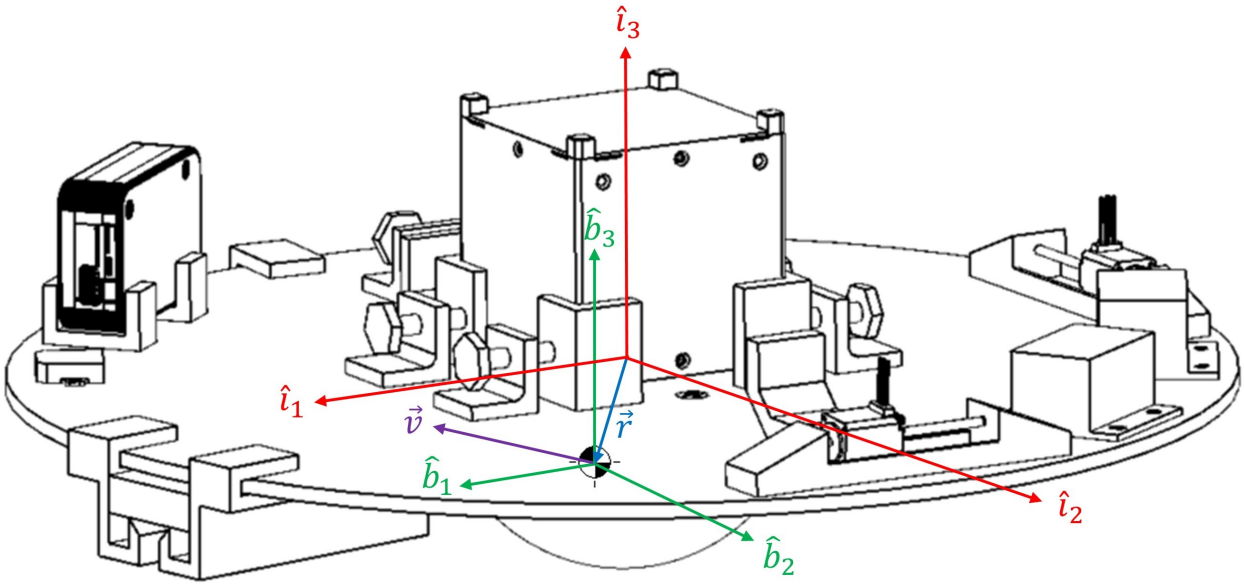


Figure 4.2: The CSACS System Model Reference Frames. The assembled CSACS with a 1U CubeSat as the demonstration payload is shown. The origins of the inertial frame (red) and the body-fixed frame (green) are located at the air bearing CR and the CSACS CM, respectively. The r vector (blue) is the distance from the CR to the CM, and the v vector (purple) is the velocity of the CM. Note that the linear stepper motors are aligned with $\hat{b}_1$ and $\hat{b}_2$, which correspond to the body-fixed frame x- and y-axes, respectively.

The right-hand side of Equation 4.16 can be split into separate terms

$$T_{ext} = T_g + T_{aero} \tag{4.18}$$

where T_g is the gravitational torque due to the CM offset vector r, and T_{aero} is the torque caused by aerodynamic disturbances.

These terms are defined as

$$T_g = m_{tot} \cdot g \begin{bmatrix} 0 & -\cos\phi\cos\theta & \sin\phi\cos\theta \\ \cos\phi\cos\theta & 0 & \sin\theta \\ -\sin\phi\cos\theta & -\sin\theta & 0 \end{bmatrix} r \tag{4.19}$$

$$T_{aero} = -\begin{bmatrix} b_x\omega_x^2 & b_y\omega_y^2 & b_z\omega_z^2 \end{bmatrix}^T \tag{4.20}$$

where b_x, b_y, and b_z are damping coefficients. Now that we have defined each contributing external torque, it is beneficial to combine the torques into the single term, T_{ext}

$$T_{ext} = \begin{bmatrix} -m_{tot}gr_y\cos\phi\cos\theta + m_{tot}gr_z\sin\phi\cos\theta \\ m_{tot}gr_x\cos\phi\cos\theta + m_{tot}gr_z\sin\theta \\ -m_{tot}gr_x\sin\phi\cos\theta - m_{tot}gr_y\sin\theta \end{bmatrix} - \begin{bmatrix} b_x\omega_x^2 \\ b_y\omega_y^2 \\ b_z\omega_z^2 \end{bmatrix} \tag{4.21}$$

Next, define the vector H_c used in Equation 4.17 as

$$H_c = I\omega_b = \begin{bmatrix} I_{xx}\omega_x + I_{xy}\omega_y + I_{xz}\omega_z \\ I_{yx}\omega_x + I_{yy}\omega_y + I_{yz}\omega_z \\ I_{zx}\omega_x + I_{zy}\omega_y + I_{zz}\omega_z \end{bmatrix} \tag{4.22}$$

Substituting Equation 4.18 and Equation 4.17 into Equation 4.16 yields

$$A\dot{\omega}_b + B = T_{ext} \tag{4.23}$$

where A and B are defined in Equation 4.24 and Equation 4.25

$$A = \begin{bmatrix} m_{tot}r_y^2 + m_{tot}r_z^2 + I_{xx} & -m_{tot}r_xr_y + I_{xy} & -m_{tot}r_xr_z + I_{xz} \\ -m_{tot}r_xr_y + I_{xy} & m_{tot}r_x^2 + m_{tot}r_z^2 + I_{yy} & -m_{tot}r_yr_z + I_{yz} \\ -m_{tot}r_xr_z + I_{xz} & -m_{tot}r_yr_z + I_{yz} & m_{tot}r_y^2 + m_{tot}r_x^2 + I_{zz} \end{bmatrix} \tag{4.24}$$

$$B = \begin{bmatrix} B_x & B_y & B_z \end{bmatrix}^T \tag{4.25}$$

The components of B are long, so they are explicitly defined in Appendix A.1. We can now rearrange Equation 4.23 to the final equations of motion

$$\dot{\omega}_b = A^{-1}(T_{ext} - B) \tag{4.26}$$

Equation 4.26 is then integrated and solved simultaneously with the Euler rate equations from Equation 4.14. The equations of motion can be greatly simplified by assuming that ω_b, r, products of inertia, and T_{aero} are small compared to other terms [15]. These assumptions give way to a simplified form of the equations of motion

$$\dot{\omega}_b = m_{tot} \cdot g \begin{bmatrix} \frac{-r_y \cos\phi\cos\theta + r_z \sin\phi\cos\theta}{I_{xx}} \\ \frac{r_x \cos\phi\cos\theta + r_z \sin\theta}{I_{yy}} \\ \frac{-r_x \sin\phi\cos\theta - r_y \sin\theta}{I_{zz}} \end{bmatrix} \tag{4.27}$$

4.1.3 Center of Mass Location for Dynamic Balancing

To simulate the torque-free motion of a satellite in the laboratory setting, it is crucial to minimize the disturbance torques acting on the CSACS. When the CM of the CSACS is not exactly at the CR of the air bearing, gravity creates a moment about the CR. This undesirable gravitational torque can be minimized by moving the CM very close to the CR. This process is called "balancing." A common technique that is used to balance spherical air bearing simulators is known as manual balancing. Although manual balancing can be an iterative and time-consuming process, it is simple in its conception and execution. An operator simply adjusts weights in various locations on the simulator in order to properly align the CM with the CR. As an improvement over the iterative process of manual balancing, an automatic balancing procedure is derived below. This technique automatically calculates the CM location and adjusts it to a location close to the CR.

The first step in the automatic balancing process is locating the CM of the CSACS. The simplified equations of motion (see Equation 4.27) are integrated over a short time period to produce a set of three equations at each time step. Assuming that pitch and roll motion remain constant over a short time step, the integration procedure simplifies even further. After integration, we are left with three equations containing three unknowns, which are the CM offset vector components of r.

These equations are written as

$$\omega_{x_{t_2}} - \omega_{x_{t_1}} = \frac{-m_{tot} g \Delta t}{2 I_{xx}} \{[(\cos\phi\cos\theta)_{t_2} + (\cos\phi\cos\theta)_{t_1}] r_y - [(\sin\phi\cos\theta)_{t_2} + (\sin\phi\cos\theta)_{t_1}] r_z\} \quad (4.28)$$

$$\omega_{y_{t_2}} - \omega_{y_{t_1}} = \frac{m_{tot} g \Delta t}{2 I_{yy}} \{[(\cos\phi\cos\theta)_{t_2} + (\cos\phi\cos\theta)_{t_1}] r_x - [(\sin\theta)_{t_2} + (\sin\theta)_{t_1}] r_z\} \quad (4.29)$$

$$\omega_{z_{t_2}} - \omega_{z_{t_1}} = \frac{-m_{tot} g \Delta t}{2 I_{zz}} \{[(\sin\phi\cos\theta)_{t_2} + (\sin\phi\cos\theta)_{t_1}] r_x - [(\sin\theta)_{t_2} + (\sin\theta)_{t_1}] r_y\} \quad (4.30)$$

Placing these equations in matrix form gives

$$\begin{bmatrix} \Delta\omega_x \\ \Delta\omega_y \\ \Delta\omega_z \end{bmatrix} = \begin{bmatrix} 0 & \Phi_{12} & \Phi_{13} \\ \Phi_{21} & 0 & \Phi_{23} \\ \Phi_{31} & \Phi_{32} & 0 \end{bmatrix} \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} \quad (4.31)$$

where the six values of Φ are assumed constant over a given time step. These six values are determined through careful manipulation of Equations 4.28, 4.29, and 4.30, and are provided in detail in Appendix B.1.

Condensing Equation 4.31 even further, we can write

$$\Delta\Omega = \mathbf{\Phi} r \quad (4.32)$$

Then, we can solve these equations for r by using the method of least squares. A full derivation for finding an estimate of a solution is provided in Appendix B.2. Again, we collect these equations into matrix form, and then expand the matrices over multiple time steps, as shown in Equation 4.33

$$\begin{bmatrix} (\Delta\omega_x)_{t_0} \\ (\Delta\omega_y)_{t_0} \\ (\Delta\omega_z)_{t_0} \\ (\Delta\omega_x)_{t_1} \\ (\Delta\omega_y)_{t_1} \\ (\Delta\omega_z)_{t_1} \\ \vdots \end{bmatrix} = \begin{bmatrix} 0 & (\Phi_{12})_{t_0} & (\Phi_{13})_{t_0} \\ (\Phi_{21})_{t_0} & 0 & (\Phi_{23})_{t_0} \\ (\Phi_{31})_{t_0} & (\Phi_{32})_{t_0} & 0 \\ 0 & (\Phi_{12})_{t_1} & (\Phi_{13})_{t_1} \\ (\Phi_{21})_{t_1} & 0 & (\Phi_{23})_{t_1} \\ (\Phi_{31})_{t_1} & (\Phi_{32})_{t_1} & 0 \\ & \vdots & \end{bmatrix} \begin{bmatrix} \hat{r}_x \\ \hat{r}_y \\ \hat{r}_z \end{bmatrix} \quad (4.33)$$

which can again be written in condensed form as

$$\Delta\Omega_{OLS} = \mathbf{\Phi}_{OLS}\hat{r} \tag{4.34}$$

and the least squares approximation for r is given by

$$\hat{r} = \mathbf{\Phi}^{\dagger}_{OLS}\ \Delta\Omega_{OLS} \tag{4.35}$$

where $\mathbf{\Phi}^{\dagger}_{OLS} = \left[\mathbf{\Phi}^{T}_{OLS}\ \ \mathbf{\Phi}_{OLS}\right]^{-1}\mathbf{\Phi}^{T}_{OLS}$ is the pseudoinverse of $\mathbf{\Phi}_{OLS}$, and $\hat{r}$ is the ordinary least squares estimator (OLS) position vector of the CM of the CSACS relative to the CR of the air bearing.

Once the estimated CM offset vector $\hat{r}$ is known, the CM can be automatically adjusted via mass moving units (MMUs). These MMUs are the linear stepper motors described in Section 3.1.5. Translating a stepper motor causes a change in the CM of the composite system. There are two MMUs mounted on the CSACS, one aligned with the x-axis, and the other aligned with the y-axis. Correlating the CM changes caused by the movement of both MMUs, the CM offset vector can be minimized after verifying that the initial offset is within margin of the CR.

Determining the MMU move distance to compensate for the CM offset is determined from the general form equation that defines the CM of a body, or

$$r = \frac{1}{m_{tot}}\sum_{i=1}^{n} m_i r_i \tag{4.36}$$

where r is the CM offset vector, m_{tot} is the combined system mass, m_i is the mass of each body, and r_i is the distance to the CM of each body [15].

The entire CSACS platform is assumed to consist of three components. The first component is the entire CSACS system minus both MMUs, and the other two components are the MMUs themselves. In order to simplify the derivation, we will assume that the CM offset vector is a result solely based on location of the MMUs. In other words, the CSACS system itself has no CM offset, and the MMUs are only displaced along their respective axes of orientation.

This assumption allows each CM offset vector to be written as

$$r_1 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T \tag{4.37}$$

$$r_2 = \begin{bmatrix} r_{m_x} & 0 & 0 \end{bmatrix}^T \tag{4.38}$$

$$r_3 = \begin{bmatrix} 0 & r_{m_y} & 0 \end{bmatrix}^T \tag{4.39}$$

where r_m is the distance vector to the CM of the MMUs. Our assumptions allow us to now write Equation 4.36 as

$$r = \frac{1}{m_{tot}} \begin{bmatrix} m_2 r_{m_x} \\ m_3 r_{m_y} \\ 0 \end{bmatrix} \tag{4.40}$$

The quantity of interest is the change in each MMU position that brings the composite system's CM to zero, or

$$\Delta r_m = -m_{tot} \begin{bmatrix} r_x/m_2 \\ r_y/m_3 \\ 0 \end{bmatrix} \tag{4.41}$$

Given that both MMU masses are equal, Equation 4.40 becomes

$$r = \frac{m}{m_{tot}} r_m \tag{4.42}$$

and Equation 4.41 becomes

$$\Delta r_m = -\frac{m_{tot}}{m} r \tag{4.43}$$

The automatic balancing procedure is only limited by the maximum distance each MMU can move. The maximum linear distance of each MMU is approximately 0.055 m. With an estimated system mass $m_{tot} = 7.75$ kg and MMU mass $m = 0.043$ kg, taking the magnitude of Equation 4.42 corresponds to a maximum correctable CM offset of 0.30 mm. This result simply means that the magnitude of the x and y CM location of the composite CSACS system must be manually balanced to within 0.30 mm before the automatic balancing procedure can be effective. Due to difficulties in determining the physical parameters of the CSACS, this is an iterative process that effectively relocates the CM closer to the CR.

4.2 Pyramidal Reaction Wheel Array

4.2.1 Quaternion Representation

The Euler angle sequence approach to relating the orientation of two refernce frames is reasonably straightforward to develop and visualize; however, we will see that it is not an ideal approach when describing spacecraft attitude dynamics [16, 51]. In Section 4.1.1, it is shown that because of the CSACS having constrained rotation in pitch and roll, the kinematic singularity when $\theta = 90°$ is not a concern, but for unconstrained 3-DOF rotation, this is not the case.

According to Euler's rotation theorem, only one rotation about a fixed axis is necessary to relate one independent orthonormal frame to another [24]. So instead of using the Euler angle sequence approach, which incorporates three rotations and three orientation angles to describe rotational motion, only one rotation about a single "fixed" axis is required. This axis, denoted by $\hat{\mathbf{k}}$, is called the *Euler axis/eigenaxis*, and the angle, denoted by λ, is called the *principal Euler angle*.

For a rotation from $\mathcal{F}_i$ to $\mathcal{F}_b$ about axis $\hat{\mathbf{k}}$ through angle λ, we express the rotation matrix $\mathbf{R}^{bi}$ in terms of $\hat{\mathbf{k}}$ and λ, just as we did for Euler angles. The rotation is about $\hat{\mathbf{k}}$, thus the eigenaxis vector has the same components in $\mathcal{F}_i$ and $\mathcal{F}_b$, meaning that we can write

$$\mathbf{R}^{bi}\mathbf{k} = \mathbf{k} \tag{4.44}$$

Thus, we can represent the general rotation matrix $\mathbf{R}$ in terms of the Euler axis and principal Euler angle

$$\mathbf{R}^{bi} = \cos\lambda \cdot \mathbb{1} + (1 - \cos\lambda)\mathbf{k}\mathbf{k}^T - \sin\lambda \cdot \mathbf{k}^{\times} \tag{4.45}$$

where $\mathbf{k}$ is the column vector of the components of $\hat{\mathbf{k}}$ in either $\mathcal{F}_i$ or $\mathcal{F}_b$, $\mathbb{1}$ is the 3×3 identity matrix, and the superscript $^{\times}$ denotes the skew-symmetric matrix representation of $\mathbf{k}$, *i.e.*

$$\mathbf{k}^{\times} = \begin{bmatrix} 0 & -k_3 & k_2 \\ k_3 & 0 & -k_1 \\ -k_2 & k_1 & 0 \end{bmatrix} \tag{4.46}$$

Then, to compute the skew matrix $\mathbf{k}^{\times}$ and principal Euler angle λ for a given rotation $\mathbf{R}$, it can be shown that

$$\lambda = \cos^{-1}\left[\frac{1}{2}(\text{trace } \mathbf{R} - 1)\right] \tag{4.47}$$

$$\mathbf{k}^{\times} = \frac{1}{2\sin\lambda}(\mathbf{R}^T - \mathbf{R}) \tag{4.48}$$

So now we can describe a rotation in terms of an Euler angle sequence or an Euler axis/angle set. There are additional parameterizations of attitude that help describe rotational motion, and so, we introduce one of the most widely used parameterizations called *Euler parameters*, otherwise known as *quaternions*.

Defining four new variables in terms of $\mathbf{k}$ and λ

$$q_0 = \cos\frac{\lambda}{2} \tag{4.49}$$

$$\mathbf{q} = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix}^T = \mathbf{k}\sin\frac{\lambda}{2} \tag{4.50}$$

where the 3×1 vector $\mathbf{q}$ forms the Euler axis component, and q_0 forms the scalar component. Collectively, these four variables define a quaternion. We use the notation $\overline{\mathbf{q}}$ to denote the 4×1 vector containing each variable

$$\overline{\mathbf{q}} = \begin{bmatrix} q_0 & \mathbf{q}^T \end{bmatrix}^T \tag{4.51}$$

Corresponding the relative orientation of two reference frames using $\overline{\mathbf{q}}$:

$$\mathbf{R} = (q_0^2 - \mathbf{q}^T\mathbf{q})\mathbb{1} + 2\mathbf{q}\mathbf{q}^T - 2q_0\mathbf{q}^{\times} \tag{4.52}$$

Expressing $\overline{\mathbf{q}}$ in terms of the elements of $\mathbf{R}$:

$$q_0 = \pm\frac{1}{2}\sqrt{1 + \text{trace } \mathbf{R}} \tag{4.53}$$

$$\mathbf{q} = \frac{1}{4q_0}\begin{bmatrix}\mathbf{R}_{23} - \mathbf{R}_{32}\\ \mathbf{R}_{31} - \mathbf{R}_{13}\\ \mathbf{R}_{12} - \mathbf{R}_{21}\end{bmatrix} \tag{4.54}$$

To summarize, we now have three unique methods that parameterize a rotation sequence: Euler angles, the Euler axis/angle set, and quaternions

$$\mathbf{R} = \mathbf{R}_i(\theta_3)\mathbf{R}_j(\theta_2)\mathbf{R}_k(\theta_1)$$
$$\mathbf{R} = \cos\lambda \cdot \mathbb{1} + (1 - \cos\lambda)\mathbf{k}\mathbf{k}^T - \sin\lambda \cdot \mathbf{k}^\times$$
$$\mathbf{R} = (q_0^2 - \mathbf{q}^T\mathbf{q})\mathbb{1} + 2\mathbf{q}\mathbf{q}^T - 2q_0\mathbf{q}^\times$$

where θ_1, θ_2, θ_3 are generalized Euler angles, and the generalized subscripts i, j, k in the Euler angle formulation indicate that there are twelve unique Euler angle sequences, *i.e.*, $k \in \{1,2,3\}$, $j \in \{1,2,3\}\backslash k$, and $i \in \{1,2,3\}\backslash j$ [16].

Now, just as we did in Section 4.1.1, we can derive the quaternion kinematic equations from the rotation matrix. By exploiting the facts that $\mathbf{R}^{-1} = \mathbf{R}^T \Rightarrow \mathbf{R}^T\mathbf{R} = \mathbb{1}$ and $\|\overline{\mathbf{q}}\| = 1$, we can compute the time derivative of $\mathbf{R}$ to characterize the kinematic equations describing the orientation, where

$$\dot{\mathbf{R}} = \omega_b^\times \mathbf{R} \tag{4.55}$$

Relating Equations 4.52 and 4.55, we arrive at the quaternion kinematic equations, where detailed derivations can be found in [10, 24]

$$2\dot{q}_0 = -\mathbf{q}^T\omega_b \tag{4.56}$$

$$2\dot{\mathbf{q}} = q_0\omega_b - \omega_b \times \mathbf{q} \tag{4.57}$$

Combining Equations 4.56 and 4.57, we arrive at

$$\dot{\overline{\mathbf{q}}} = \begin{bmatrix}\dot{q}_0\\ \dot{\mathbf{q}}\end{bmatrix} = \frac{1}{2}\begin{bmatrix}-\mathbf{q}^T\\ q_0\mathbb{1} + \mathbf{q}^\times\end{bmatrix}\omega_b \tag{4.58}$$

Although there are now four equations to deal with rather than three, it is clear that there are no trigonometric nonlinearities and no kinematic singulatrities, making these equations ideal for unconstrained rotational motion.

4.2.2 Reformulating the Equations of Motion

Reformulating the CSACS equations of motion to include the RWA described in Section 3.2, we will begin with the total angular momentum of the CSACS

$$H = H_c + H_w \tag{4.59}$$

where we can recall $H_c = I\omega_b$ is the CSACS angular momentum about the CM, and then define H_w as the angular momentum of the reaction wheels.

Starting with the matrix formulation of Euler's rotational equations of motion [61], we can write

$$\dot{H} = -\omega_b^{\times} H + T_{ext} \tag{4.60}$$

where we recall that the superscript $^{\times}$ denotes the 3×3 skew-symmetric matrix representation of $\omega_b = \begin{bmatrix}\omega_x & \omega_y & \omega_z\end{bmatrix}^T$, that is

$$\omega_b^{\times} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{4.61}$$

Next, taking the time derivative of Equation 4.59, we can write

$$\dot{H} = I\dot{\omega}_b + \dot{H}_w \tag{4.62}$$

Then, equating Equations 4.60 and 4.62

$$I\dot{\omega}_b = -\omega_b^{\times} H + T_{ext} - \dot{H}_w \tag{4.63}$$

Substituting Equation 4.59 into 4.63, we arrive at

$$\dot{\omega}_b = I^{-1}\left[-\omega_b{}^{\times}\left(I\omega_b + H_w\right) + T_{ext} - \dot{H}_w\right] \tag{4.64}$$

Recall that the reaction wheel array works on the principal of conservation of angular momentum, thus the angular momentum generated from each wheel is transferred to the CSACS

with opposite sign, *i.e.*, $\dot{H}_w = -T_c$, where T_c is the commanded torque determined by the controller. Recall that the total external torque acting on the CSACS is represented as, $T_{ext} = T_{aero} + T_g$. This brings us to the nonlinear attitude dynamics of the CSACS, combined with the RWA, that we will use for our control model

$$\dot{\omega}_b = I^{-1}\left[-{\omega_b}^{\times}\left(I\omega_b + H_w\right) + T_c + T_{ext}\right] \tag{4.65}$$

4.2.3 External Torques

In Section 4.1.2, we developed the external torques that could be acting on the CSACS: gravitational torque from CM offset T_g, and aerodynamic torque T_{aero}. Due to the difficulties of developing an analytical model of aerodynamic torque, coupled with the gravitational torque's dominating presence as a disturbing torque, we will only consider T_g as an external torque for the controller model.

Note that we can represent Equation 4.19 from Section 4.1.2 in the form

$$T_g = m_{tot} \cdot g \begin{bmatrix} 0 & -\cos\phi\cos\theta & \sin\phi\cos\theta \\ \cos\phi\cos\theta & 0 & \sin\theta \\ -\sin\phi\cos\theta & -\sin\theta & 0 \end{bmatrix} r = -m_{tot} g r^{\times} \mathbf{R} \hat{e}_3 \tag{4.66}$$

Thus, expressing the external torque for the controller model in quaternion notation, we can write

$$T_{ext} = T_g = -m_{tot} g r^{\times} \mathbf{R} \hat{e}_3 \tag{4.67}$$

where $\mathbf{R}$ is the quaternion representation of the 3-2-1 rotation sequence developed in Section 4.1.1, that is

$$\mathbf{R} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{4.68}$$

4.2.4 Pyramidal Distribution Matrix

The distribution matrix for a pyramidal reaction wheel configuration transforms the four individual torques produced by each wheel to three body-axis torques, such that their contributions can be implemented in the dynamics model developed in Sections 4.2.2.

The pyramid configuration of reaction wheels is shown in Figure 4.3. Following the work of Sidi [48], the relationship between each reaction wheel torque (τ_1, τ_2, τ_3, τ_4) and the torque produced along each body axis of the CSACS (T_{c1}, T_{c2}, T_{c3}) can be shown as

$$T_{c1} = \tau_1 \cos\beta - \tau_3 \cos\beta \tag{4.69}$$

$$T_{c2} = \tau_2 \cos\beta - \tau_4 \cos\beta \tag{4.70}$$

$$T_{c3} = \tau_1 \sin\beta + \tau_2 \sin\beta + \tau_3 \sin\beta + \tau_4 \sin\beta \tag{4.71}$$

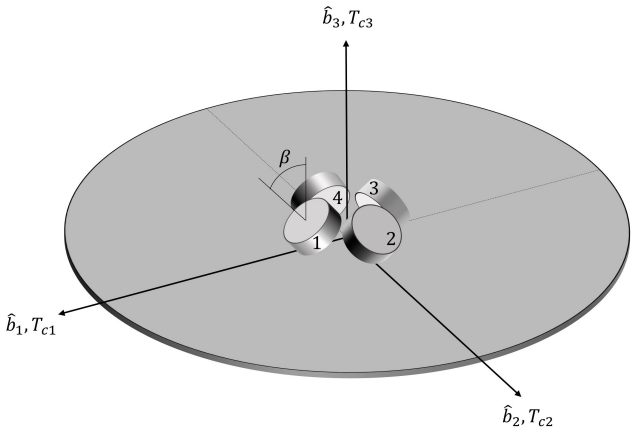


Figure 4.3: Pyramid Configuration of Four Reaction Wheels. Top view schematic of the AFIT RWA (Figure 3.2) mounted in the center of the CSACS for dynamic control. Again, the torques produced along the three body axes are T_{c1}, T_{c2}, T_{c3}. The rotational axes of all four wheels are inclined to the $\hat{b}_1 - \hat{b}_2$ plane by angle β, with wheels 1 and 3 aligned facing the $+\hat{b}_1$ and $-\hat{b}_1$ direction, respectively, and wheels 2 and 4 aligned facing the $+\hat{b}_2$ and $-\hat{b}_2$ direction, respectively. τ_1, τ_2, τ_3, and τ_4 are the torques about the reaction wheel rotational axes.

In matrix form, Equations 4.69, 4.70, and 4.71 are represented as

$$\begin{bmatrix} T_{c1} \\ T_{c2} \\ T_{c3} \end{bmatrix} = \begin{bmatrix} \cos\beta & 0 & -\cos\beta & 0 \\ 0 & \cos\beta & 0 & -\cos\beta \\ \sin\beta & \sin\beta & \sin\beta & \sin\beta \end{bmatrix} \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} \tag{4.72}$$

From Section 3.2, knowing that the reaction wheel inclination angle $\beta = 45°$, the distribution matrix becomes

$$A_w = \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} & 0 \\ 0 & \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix} \tag{4.73}$$

Note that we will also use the distribution matrix to transform individual wheel speeds to the three body axes for use in Equation 4.65. Thus, the angular momentum produced by the reaction wheels becomes

$$H_w = A_w I_w \omega_w \tag{4.74}$$

where I_w is the reaction wheel moment of inertia (scalar value since wheel inertias are identical), and ω_w is the 4×1 vector consisting of each reaction wheel angular rate.

4.2.5 Control Allocation

The controller, which is developed in Section 4.2.7, commands specific torques to each reaction wheel in order to align the CSACS body axes to their desired attitude. The attitude control provided by the reaction wheels is an overactuated control system. That is, there are more actuators ($r = 4$) than there are degrees of freedom ($n = 3$) in the system. Thus, we want to find an optimal torque distribution for the pyramidal RWA.

Following the work developed in [22, 49], the generalized control vector $\tau_c \in \mathbb{R}^n$ must be distributed to an input control vector $u \in \mathbb{R}^r$. Relating a generalized control vector and input control can be shown by

$$\tau_c = T(\alpha)Ku \tag{4.75}$$

where $\alpha \in \mathbb{R}^r$ and $u \in \mathbb{R}^r$ are defined as

$$\alpha = [\alpha_1 \ \alpha_2 \ \cdots \ \alpha_r]^T \qquad u = [u_1 \ u_2 \ \cdots \ u_r]^T \tag{4.76}$$

and $f = Ku \in \mathbb{R}^r$ is defined as the produced torque vector, $K \in \mathbb{R}^{r\times r}$ is a diagonal matrix $K = \text{diag}\{K_1 \ K_2 \ \cdots \ K_r\}$, and the distribution matrix, A_w, derived in Section 4.2.4 is denoted $T(\alpha)$ and defined as $T(\alpha) = [t_1 \ t_2 \ \cdots \ t_r]$, where $T(\alpha) \in \mathbb{R}^{n\times r}, \ t_i \in \mathbb{R}^n$.

Distributing the generalized control vector τ_c to input control u becomes a minimization problem of the produced torque vector $f = Ku$, where a familiar least squares derivation (Appendix B.2) leads us to the solution of the optimal commanded control action provided by the reaction wheels

$$u = K^{-1}T_W^{\dagger}\tau_c \tag{4.77}$$

where $T_W^{\dagger} = W^{-1}T^T\left(TW^{-1}T^T\right)^{-1}$ is the right pseudoinverse transformation, and $W \in \mathbb{R}^{r\times r}$ is a diagonal, positive definite "weight" matrix. When $W = \mathbb{1}$, all actuators get equal weight and the pseudoinverse becomes

$$T_W^{\dagger} = T^T\left(TT^T\right)^{-1} \tag{4.78}$$

Sidi shows the same results using the calculus of variation [48].

4.2.6 Reaction Wheel DC Motor Model

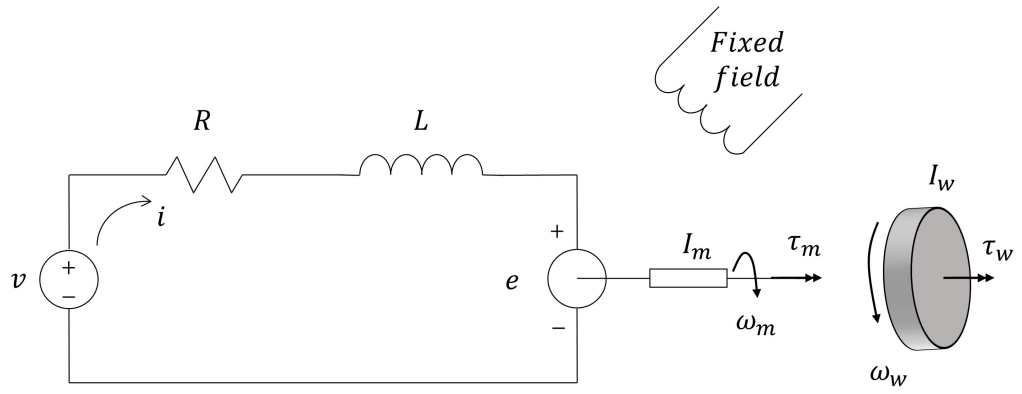


Figure 4.4: Simple DC Motor Circuit Diagram. A DC motor is a type of rotary electrical transformer, where the rotational speed of the motor is controlled by varying the current strength in the field winding, *i.e.*, a voltage source v is applied to the motor's armature via an induced magnetic field.

The reaction wheels are driven by the DC motors introduced in Section 3.2, which have parameters of inductance $L = 0.378$ mH, resistance $R = 2.2\ \Omega$, current i, back electromotive force (EMF) e, and voltage v. From Kirchoff's voltage law, we can derive

$$v = e + iR + \frac{di}{dt}L \tag{4.79}$$

First, we define the DC motor torque as proportional to the current i

$$\tau_m = K_\tau i \tag{4.80}$$

where $K_\tau = 5.67$ N·m/A is the torque gain provided by the motor's datasheet [35].

According to Faraday's law of induction, the back EMF is

$$e = K_e \omega_m \tag{4.81}$$

where $K_e = 1/56\pi$ V/(rad/s) is the back EMF gain provided by the motor's datasheet [35], and ω_m is the angular velocity of the motor shaft.

Next we form the workload torque of the DC motor τ_L, which consists of the reaction wheel mounted to the motor shaft as shown in Figure 4.4

$$\tau_L = I_w \dot{\omega}_w \tag{4.82}$$

The motor shaft has the same angular velocity and acceleration as a reaction wheel, thus, computing the moment balance of the rotor and wheel yields

$$I_m \dot{\omega}_m = \tau_m - \tau_L \tag{4.83}$$

Substituting Equation 4.82 into 4.83, knowing that $\dot{\omega}_m = \dot{\omega}_w$, we arrive at our relationship for the torque generated by a reaction wheel in the DC motor model

$$\tau_w = (I_m + I_w)\, \dot{\omega}_w \tag{4.84}$$

For simplicity during simulation, the settling time of the DC motor is assumed negligible compared to the maneuvers of the CSACS. In the Simulink® model, the torque command is sent to the motor, and the angular velocity is calculated from Equation 4.84. Then, the voltage applied to the motor is determined from Equation 4.81. Lastly, power input to the motors P_{in} is calculated from

$$P_{in} = iv \tag{4.85}$$

4.2.7 Controller Design

Reference Trajectory

Tracking regulators rely on continuous and differential trajectories, thus the reference model takes a unit-step input and filters it to provide a smooth signal for the controller to follow. A second-order filter for the reference trajectory is used, given by

$$\frac{\overline{\mathbf{q}}_d}{\overline{\mathbf{q}}_r} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{4.86}$$

where $\overline{\mathbf{q}}_r$ is the reference quaternion, $\overline{\mathbf{q}}_d$ the desired quaternion, the damping ratio is chosen as $\zeta \approx 1$, such that the output from the reference model is approximately critically damped, and the undamped natural frequency ω_n is chosen such that the reference trajectory settling time $t_s = 60$ sec with a maximum percent overshoot criteria $M_p = 2\%$.

Quaternion Feedback Controller

In quaternion feedback control, attitude is controlled by determining the attitude error quaternion vector, which represents the difference between the attitude at arbitrary time t and the reference attitude [10, 48, 61]. Suppose a desired orientation is given by the rotation matrix $\mathbf{R}_d$, and actual orientation given by rotation matrix $\mathbf{R}$. Simply subtracting $\mathbf{R}_d$ from $\mathbf{R}$ would not be a valid rotation matrix for the attitude error. Instead, deviation between the desired and actual orientation can be described by the rotation matrix $\mathbf{R}_e \in SO(3)$, defined by

$$\mathbf{R}_e \triangleq \mathbf{R}_d^T \mathbf{R} \tag{4.87}$$

Using the special properties of rotation matrices and unit quaternions, we know that

$$\mathbf{R}^{-1} = \mathbf{R}^T \tag{4.88}$$

$$\mathbf{R}(\overline{\mathbf{q}})^{-1} = \mathbf{R}(\overline{\mathbf{q}}^*) \tag{4.89}$$

where $\overline{\mathbf{q}}^* = q_0 - \mathbf{q}$ is the complex conjugate of $\overline{\mathbf{q}}$.

Then, the definition provided in Equation 4.87 implies that

$$\mathbf{R}(\overline{\mathbf{q}}_e) \triangleq \mathbf{R}^T(\overline{\mathbf{q}}_d)\mathbf{R}(\overline{\mathbf{q}}) = \mathbf{R}(\overline{\mathbf{q}}_d^*)\mathbf{R}(\overline{\mathbf{q}}) \tag{4.90}$$

where $\overline{\mathbf{q}}_d$ is the desired quaternion, $\overline{\mathbf{q}}_d^*$ is the complex conjugate of $\overline{\mathbf{q}}_d$, $\overline{\mathbf{q}}$ is the actual quaternion, and $\overline{\mathbf{q}}_e$ is the error quaternion.

Then, we can express successive rotations in quaternion representation as

$$\mathbf{R}(\overline{\mathbf{q}}_e) = \mathbf{R}(\overline{\mathbf{q}}_d^*)\mathbf{R}(\overline{\mathbf{q}}) = \mathbf{R}(\overline{\mathbf{q}}_d^* \otimes \overline{\mathbf{q}}) \tag{4.91}$$

where $\otimes$ denotes the *quaternion product* operator. Following the general procedure to compute a quaternion product, as presented in Appendix C.1, the error in unit quaternions can be expanded as

$$\overline{\mathbf{q}}_e = \overline{\mathbf{q}}_d^* \otimes \overline{\mathbf{q}} = \begin{bmatrix} q_{0e} \\ q_{1e} \\ q_{2e} \\ q_{3e} \end{bmatrix} = \begin{bmatrix} q_{0d} & q_{1d} & q_{2d} & q_{3d} \\ -q_{1d} & q_{0d} & q_{3d} & -q_{2d} \\ -q_{2d} & -q_{3d} & q_{0d} & q_{1d} \\ -q_{3d} & q_{2d} & -q_{1d} & q_{0d} \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \tag{4.92}$$

which can then be condensed to the form

$$\overline{\mathbf{q}}_e = \overline{\mathbf{q}}_d^* \otimes \overline{\mathbf{q}} = \begin{bmatrix} q_{0e} \\ \mathbf{q}_e \end{bmatrix} = \begin{bmatrix} q_{0d} & \mathbf{q}_d^T \\ -\mathbf{q}_d & \left(q_{0d}\mathbb{1} - \mathbf{q}_d^\times\right) \end{bmatrix} \overline{\mathbf{q}} = \begin{bmatrix} q_{0d}q_0 + \mathbf{q}_d^T\mathbf{q} \\ q_{0d}\mathbf{q} - q_0\mathbf{q}_d - \mathbf{q}_d \times \mathbf{q} \end{bmatrix} \tag{4.93}$$

Then, expressing the attitude error quaternion vector as the last three elements of $\overline{\mathbf{q}}_e$

$$\mathbf{q}_e = q_{0d}\mathbf{q} - q_0\mathbf{q}_d - \mathbf{q}_d \times \mathbf{q} \tag{4.94}$$

and then computing the time derivative as

$$\dot{\mathbf{q}}_e = \dot{q}_{0d}\mathbf{q} + q_{0d}\dot{\mathbf{q}} - \dot{q}_0\mathbf{q}_d - q_0\dot{\mathbf{q}}_d - \dot{\mathbf{q}}_d \times \mathbf{q} - \mathbf{q}_d \times \dot{\mathbf{q}} \tag{4.95}$$

leads us to the creation of the nonlinear quaternion feedback proportional-derivative (PD) controller

$$u = -K_p\mathbf{q}_e - K_d\dot{\mathbf{q}}_e \tag{4.96}$$

where u is the control torque vector, K_p is the positive definite proportional gain, and K_d the positive definite derivative gain. Although the controller is PD in terms of the quaternion and its derivative, notice that it is still a nonlinear controller in terms of the Euler angles, which are the actual control variables. Thus, a stability analysis follows.

Stability Analysis

We have shown that our nonlinear system is a *trajectory tracking* problem, *i.e.*, we have developed a feedback control law which drives the system along a particular path or trajectory [18]. The nonlinear system stability is accomplised by studying the behavior of an energy-like function, referred to as the Lyapunov candidate function (LCF) [18]. Summarizing the work of Khalil, we present the following theorems provided in [54]

Theorem 1 (Lyapunov's Stability Theorem) *Let* $\mathbf{x} = \mathbf{0}$ *be an equilibrium point for the system*

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) \tag{4.97}$$

and suppose that there exists a continuously differentiable function $V(\mathbf{x})$*, defined on domian* $D \subset \mathbb{R}^n$ *which contains the equilibrium. If*

$$V(\mathbf{x}) > 0 \qquad \textit{and} \qquad \dot{V}(\mathbf{x}) \leq 0$$

in D*, then* $\mathbf{x} = \mathbf{0}$ *is stable. If*

$$V(\mathbf{x}) > 0 \qquad \textit{and} \qquad \dot{V}(\mathbf{x}) < 0$$

in D*, then* $\mathbf{x} = \mathbf{0}$ *is asymptotically stable.*

Definition: The equilibrium $\mathbf{x}_e = \mathbf{0}$ of Equation 4.97 is called *globally asymptotically stable* if it is asymptotically stable and every trajectory converges to an equilibrium.

Theorem 2 (Global Asymptotic Stability) *Let* $\mathbf{x} = \mathbf{0}$ *be an equilibrium point for the system*

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$$

and suppose that there exists a continuously differentiable function $V(\mathbf{x})$*, defined on all of* $\mathbb{R}^n$*. If*

$$\begin{aligned} V(\mathbf{x}) > 0 \quad & \forall\ \mathbf{x} \neq \mathbf{0}, \\ \dot{V}(\mathbf{x}) < 0 \quad & \forall\ \mathbf{x} \neq \mathbf{0},\ \textit{and} \\ V(\mathbf{x}) \to \infty \quad & \textit{as}\ \ \|\mathbf{x}\| \to \infty \end{aligned}$$

then $\mathbf{x} = \mathbf{0}$ *is globally asymptotically stable.*

Thus, the stability analysis of the CSACS coupled with a quaternion feedback controller is based on the choice of a Lyapunov candidate function. The total energy of the CSACS system is chosen as the LCF, and so, the energy considerations and Lyapunov analysis are presented below.

The quadratic representation of the kinetic energy of the CSACS is presented in the body frame

$$T = \frac{1}{2}\omega_b^T I \omega_b \tag{4.98}$$

and the gravitational potential energy contribution due to the CM offset is given by

$$U = m_{tot} g \, \|r\| \, \mathbf{c}_3 \cdot \hat{e}_3 \tag{4.99}$$

where $\mathbf{c}_3$ is the third column vector (direction cosine) element of the quaternion representation of the 3-2-1 rotation sequence, as shown in Equation 4.68, *i.e.*,

$$
\begin{aligned}
\mathbf{R} &= \begin{bmatrix} \mathbf{c}_1 & \mathbf{c}_2 & \mathbf{c}_3 \end{bmatrix} \\
&= \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \\
&= \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 + q_0 q_3) & 2(q_1 q_3 - q_0 q_2) \\ 2(q_1 q_2 - q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 + q_0 q_1) \\ 2(q_1 q_3 + q_0 q_2) & 2(q_2 q_3 - q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}
\end{aligned}
$$

Thus, the total energy of the CSACS is computed as the sum of kinetic and potential energies.

$$E = T + U \tag{4.100}$$

Substituting Equations 4.98 and 4.99, we have

$$E = \frac{1}{2}\omega_b^T I \omega_b + m_{tot} g \left\| r \right\| \mathbf{c}_3 \cdot \hat{e}_3 \tag{4.101}$$

$$E = \frac{1}{2}\omega_b^T I \omega_b + m_{tot} g \left\| r \right\| c_{33} \tag{4.102}$$

Thus, the LCF is then defined as

$$V(\mathbf{x}) = \frac{1}{2}\omega_b^T I \omega_b + m_{tot} g \left\| r \right\| c_{33} \tag{4.103}$$

Now, we want to study the stability of the origin in particular, which can be defined as

$$\mathbf{x}^* = \begin{bmatrix} \omega_b & \overline{\mathbf{q}}^T \end{bmatrix}^T \tag{4.104}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^T \tag{4.105}$$

and so, it is clear that for our Lyapunov candidate function, we can define $\mathbf{x}$ as

$$\mathbf{x} = \begin{bmatrix} \omega_b & c_{33} \end{bmatrix}^T \tag{4.106}$$

and the following properties are satisfied at the origin

$$V(\mathbf{x}) > 0 \qquad \forall\ \mathbf{x} \in \mathbb{R}^7 - \{\mathbf{0}\} \tag{4.107}$$

$$V(\mathbf{0}) = 0 \tag{4.108}$$

From the properties of unit quaternions, we know that

$$\|\overline{\mathbf{q}}\| = q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \tag{4.109}$$

$$\Rightarrow q_0^2 + q_3^2 - 1 = -q_1^2 - q_2^2 \tag{4.110}$$

Thus, we can rearrange and expand terms in the Lyapunov candidate function

$$V = \frac{1}{2}\omega_b^T I \omega_b + m_{tot} g \|r\| c_{33} \tag{4.111}$$

$$= \frac{1}{2}\omega_b^T I \omega_b + m_{tot} g \|r\| \left(q_0^2 - q_1^2 - q_2^2 + q_3^2\right) \tag{4.112}$$

$$= \frac{1}{2}\omega_b^T I \omega_b + m_{tot} g \|r\| \left(q_0^2 + q_0^2 + q_3^2 - 1 + q_3^2\right) \tag{4.113}$$

$$= \frac{1}{2}\omega_b^T I \omega_b + m_{tot} g \|r\| \left(2(q_0^2 + q_3^2) - 1\right) \tag{4.114}$$

$$= \frac{1}{2}\omega_b^T I \omega_b + 2m_{tot} g \|r\| \left(q_0^2 + q_3^2\right) - m_{tot} g \|r\| \tag{4.115}$$

$$= \frac{1}{2}\omega_b^T I \omega_b + 2m_{tot} g \|r\| q_0^2 + 2m_{tot} g \|r\| q_3^2 - m_{tot} g \|r\| \tag{4.116}$$

Now, taking the derivative of the Lyapunov function

$$\dot{V} = \frac{1}{2}\dot{\omega}_b^T I \omega_b + \frac{1}{2}\omega_b^T I \dot{\omega}_b + 4m_{tot} g \|r\| q_0 \dot{q}_0 + 4m_{tot} g \|r\| q_3 \dot{q}_3 \tag{4.117}$$

$$\dot{V} = \omega_b^T I \dot{\omega}_b + 4m_{tot} g \|r\| \left(q_0 \dot{q}_0 + q_3 \dot{q}_3\right) \tag{4.118}$$

Recalling the CSACS attitude dynamics equations from Equation 4.65

$$I\dot{\omega}_b = -{\omega_b}^{\times} \left(I\omega_b + H_w\right) + T_c + T_{ext} \tag{4.119}$$

where Equations 4.96 and 4.67 are substituted for T_c and T_{ext}, respectively

$$I\dot{\omega}_b = -{\omega_b}^{\times} \left(I\omega_b + H_w\right) - K_p \mathbf{q}_e - K_d \dot{\mathbf{q}}_e - m_{tot} g r^{\times} \mathbf{R} \hat{e}_3 \tag{4.120}$$

Now, inserting Equation 4.120 into 4.118

$$\dot{V} = \omega_b^T \left[-\omega_b{}^{\times} \left(I\omega_b + H_w \right) - K_p \mathbf{q}_e - K_d \dot{\mathbf{q}}_e - m_{tot} g r^{\times} \mathbf{R} \hat{e}_3 \right] + 4 m_{tot} g \left\| r \right\| \left(q_0 \dot{q}_0 + q_3 \dot{q}_3 \right) \quad (4.121)$$

Note that $\omega_b^T \omega_b^{\times} = 0$:

$$\Rightarrow \dot{V} = \omega_b^T \left(-K_p \mathbf{q}_e - K_d \dot{\mathbf{q}}_e \right) - m_{tot} g \omega_b^T r^{\times} \mathbf{R} \hat{e}_3 + 4 m_{tot} g \left\| r \right\| \left(q_0 \dot{q}_0 + q_3 \dot{q}_3 \right) \quad (4.122)$$

Next, we can expand the term related to gravitational torque in a similar fashion to the expansion in Equations 4.111 – 4.116, which gives us

$$\Rightarrow \dot{V} = \omega_b^T \left(-K_p \mathbf{q}_e - K_d \dot{\mathbf{q}}_e \right) + m_{tot} g \left[\left(r_x \omega_y - r_y \omega_x \right) c_{33} + \left(r_z \omega_x - r_x \omega_z \right) c_{23} + \left(r_y \omega_z - r_z \omega_y \right) c_{13} \right] + 4 m_{tot} g \left\| r \right\| \left(q_0 \dot{q}_0 + q_3 \dot{q}_3 \right) \quad (4.123)$$

$$= \omega_b^T \left(-K_p \mathbf{q}_e - K_d \dot{\mathbf{q}}_e \right) + m_{tot} g \left\{ \left[r_x \omega_y - r_y \omega_x \right] \left(q_0^2 - q_1^2 - q_2^2 + q_3^2 \right) + \left[r_z \omega_x - r_x \omega_z \right] \left(2(q_0 q_1 + q_2 q_3) \right) + \left[r_y \omega_z - r_z \omega_y \right] \left(2(q_1 q_3 - q_0 q_2) \right) \right\} + 4 m_{tot} g \left\| r \right\| \left(q_0 \dot{q}_0 + q_3 \dot{q}_3 \right) \quad (4.124)$$

Now for the sake of our problem, the balancing algorithm developed in Section 4.1.3 essentially eliminates the CM offset in the x- and y-directions, *i.e.*, $r_x, r_y \approx 0$. This assumption leaves us with

$$\dot{V} = \omega_b^T \left(-K_p \mathbf{q}_e - K_d \dot{\mathbf{q}}_e \right) + m_{tot} g \left(r_z \omega_x c_{23} - r_z \omega_y c_{13} \right) + 4 m_{tot} g r_z \left(q_0 \dot{q}_0 + q_3 \dot{q}_3 \right) \quad (4.125)$$

$$= \omega_b^T \left(-K_p \mathbf{q}_e - K_d \dot{\mathbf{q}}_e \right) + m_{tot} g \left\{ r_z \omega_x (2(q_0 q_1 + q_2 q_3)) - r_z \omega_y (2(q_1 q_3 - q_0 q_2)) \right\} + 4 m_{tot} g r_z \left(q_0 \dot{q}_0 + q_3 \dot{q}_3 \right) \quad (4.126)$$

and since we are only concerned with yaw control, *i.e.*, z-axis control, the r_z term vanishes as well. Thus, the Lyapunov derivative has been simplified quite elegantly to the form

$$\dot{V} = \omega_b^T \left(-K_p \mathbf{q}_e - K_d \dot{\mathbf{q}}_e \right) \quad (4.127)$$

Now, we will insert the attitude error quaternion and its derivative into Equation 4.127

$$\dot{V} = \omega_b^T[-K_p(q_{0d}\mathbf{q} - q_0\mathbf{q}_d - \mathbf{q}_d \times \mathbf{q}) \\ - K_d(\dot{q}_{0d}\mathbf{q} + q_{0d}\dot{\mathbf{q}} - \dot{q}_0\mathbf{q}_d - q_0\dot{\mathbf{q}}_d - \dot{\mathbf{q}}_d \times \mathbf{q} - \mathbf{q}_d \times \dot{\mathbf{q}})] \quad (4.128)$$

From Equation 4.128, it is not easy to predict that $\dot{V} < 0$, so let's take some additional steps to further simplify the Lyapunov function.

We are going to reassess the original LCF and attempt to introduce a new term that will help reduce it's derivative complexity, while still having favorable properties in a Lyapunov sense. We will select $K_p\left(\mathbf{q}^T\mathbf{q} + (1-q_0)^2\right)$ as the new term to be added to the LCF in Equation 4.116. Note that we neglect the potential energy term due to our assumptions made in Equations 4.124 – 4.127, that is

$$V = \frac{1}{2}\omega_b^T I\omega_b + K_p\big(\mathbf{q}^T\mathbf{q} + (1-q_0)^2\big) \quad (4.129)$$

Also, note that

$$\mathbf{q}^T\mathbf{q} + (1-q_0)^2 = \mathbf{q}^T\mathbf{q} + q_0^2 - 2q_0 + 1 = 2(1-q_0) \quad (4.130)$$

Thus, Equation 4.129 becomes

$$V = \frac{1}{2}\omega_b^T I\omega_b + K_p(2(1-q_0)) \quad (4.131)$$

Taking the derivative of the new Lyapunov function

$$\dot{V} = \omega_b^T I\dot{\omega}_b - 2K_p\dot{q}_0 \quad (4.132)$$

Recall Equation 4.56 from Section 4.2.1 $\left(2\dot{q}_0 = -\mathbf{q}^T\omega_b\right)$, substituting this relation into Equation 4.132 gives us

$$\dot{V} = \omega_b^T I\dot{\omega}_b + K_p\mathbf{q}^T\omega_b \quad (4.133)$$

Again, inserting Equation 4.65, our Lyapunov derivative function

$$\dot{V} = \omega_b^T\left[-{\omega_b}^{\times}(I\omega_b + H_w) + T_c\right] + K_p\mathbf{q}^T\omega_b \quad (4.134)$$

$$= \omega_b^T\left(-K_p\mathbf{q}_e - K_d\dot{\mathbf{q}}_e\right) + K_p\mathbf{q}^T\omega_b \quad (4.135)$$

For sake of simplicity, if we take our reference as zero, 4.135 becomes

$$\dot{V} = \omega_b^T \left(-K_p \mathbf{q} - K_d \dot{\mathbf{q}}\right) + K_p \mathbf{q}^T \omega_b \tag{4.136}$$

Which leaves us with

$$\dot{V} = -K_d \omega_b^T \dot{\mathbf{q}} \tag{4.137}$$

Our Lyapunov derivative function will now be easier to verify. Recall Equation 4.57 from Section 4.2.1 ($2\dot{\mathbf{q}} = q_0 \omega_b - \omega_b \times \mathbf{q}$), substituting into 4.137 yields

$$\Rightarrow \dot{V} = -\frac{1}{2} K_d \omega_b^T \left[q_0 \mathbb{1} + \mathbf{q}^\times\right] \omega_b \tag{4.138}$$

Thus, for $\dot{V} < 0$, we require

$$\left[q_0 \mathbb{1} + \mathbf{q}^\times\right] > 0 \tag{4.139}$$

By Sylvester's criterion [43], for $[q_0 \mathbb{1} + \mathbf{q}^\times] > 0$, we require that all of the leading principal minors are positive, which can be shown by

$$q_0 > 0 \tag{4.140}$$

$$q_0^2 + q_3^2 > 0 \tag{4.141}$$

$$q_0 \left(q_0^2 + q_1^2 + q_2^2 + q_3^2\right) > 0 \tag{4.142}$$

Thus, $\dot{V} < 0$ provided that $q_0 > 0$. Recalling that $q_0 = \cos \frac{\lambda}{2}$, we therefore must constrain the principal Euler angle $\lambda \in (-\pi,\ \pi)$. For our particular problem at hand, where tilt in pitch and roll are constrained to angles less than $\pm 15°$, our stability constraint on q_0 is indeed satisfied. Furthermore, a physical interpretation of this stability constraint can shown by satisfying the following two conditions: 1) the combined CSACS CM must be below the air bearing's CR, and 2) the CSACS CM is located farther down the negative z-axis than it is out along the x- or y-axes. Clearly, these physical conditions have already been satisfied by the auto-balancing procedure developed in Section 4.1.3. Thus, this method works properly and the origin is asymptotically stable.

4.3 Helmholtz Coils

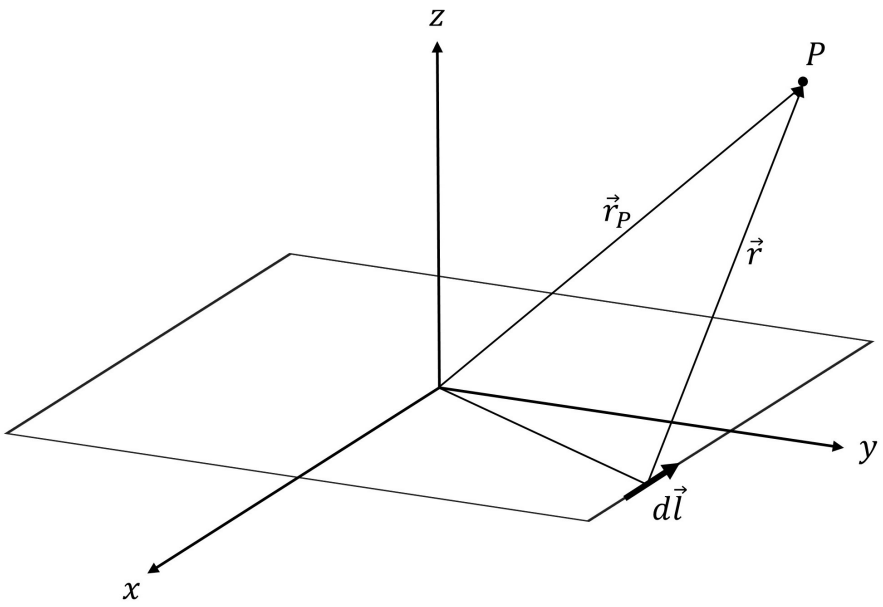


Figure 4.5: Reference Geometry for Square Coil Magnetic Field Evaluation. Only one reference coil is shown here for clarity; however, the following derivation applies to a pair of coaxial coils.

Based on the work provided in [38], an analytical model of a square Helmholtz coil is first developed to identify the optimal spacing of two coils in a Helmholtz cage configuration. The mathematical model is derived from the differential elements of the square coil loop using the Biot-Savart law, expressed by Equation 4.143. Computing the resulting magnetic field at arbitrary point P induced by an applied current distribution involves taking a vector product.

$$dB = \mu_0 \frac{I}{4\pi} \cdot \frac{d\vec{l} \times \vec{r}}{r^3} \tag{4.143}$$

The square coil configuration is shown in Figure 4.5, where a side length L lies in the x-y plane, and the z-axis is perpendicular to the coil. The magnetic field at arbitrary location P is obtained by integrating Equation 4.143 along the coil, where μ_0 is the magnetic permeability

of free space, $4\pi \times 10^{-7}$ Tesla·meters/ampere, r is the position from a differential current element to arbitrary field point P, $d\vec{l}$ is the differential length vector of a current element, and I is the current flowing through the element considered. It is then possible to determine the optimal distance d between two coils which generates the largest homogeneous magnetic field between them. For a pair of coils, the B-field generated at an arbitrary location is determined by summing the B-field vectors from both coils.

Computing the line integrals around each coil and then taking the sum of their solutions, the magnetic field generated from both coils in the z-axis direction is provided by

$$B_z(z) = 2\frac{\mu_0}{\pi} IL \left[\frac{1}{(4z^2 + 4dz + d^2 + L^2)\sqrt{z^2 + dz + \frac{d^2}{4} + \frac{L^2}{2}}} + \frac{1}{(4z^2 - 4dz + d^2 + L^2)\sqrt{z^2 - dz + \frac{d^2}{4} + \frac{L^2}{2}}} \right] \tag{4.144}$$

Differentiating B_z, we arrive at

$$\frac{dB_z}{dz}(z) = -\frac{\mu_0}{\pi} IL^2 \left[\frac{(2z + d)(12z^2 + 12dz + 3d^2 + 5L^2)}{(4z^2 + 4dz + d^2 + L^2)^2 \left(z^2 + dz + \frac{d^2}{4} + \frac{L^2}{2}\right)^{\frac{3}{2}}} + \frac{(2z - d)(12z^2 - 12dz + 3d^2 + 5L^2)}{(4z^2 - 4dz + d^2 + L^2)^2 \left(z^2 - dz + \frac{d^2}{4} + \frac{L^2}{2}\right)^{\frac{3}{2}}} \right] \tag{4.145}$$

which equals zero when $z = 0$.

We desire a uniform magnetic field centered between both coils, thus the second derivative of $B_z(z)$ must equal zero as well. We arrive at this solution by differentiating $dB_z/dz(z)$ and letting $z = 0$

$$d = 0.5445 \cdot L \tag{4.146}$$

Equation 4.146 presents the optimal distance between two coils that creates a maximally uniform magnetic field [38].

Substituting the result in Equation 4.146 into Equation 4.144, we form the governing relation for the magnetic field at the center of a square coil pair, as shown in [3]

$$B(0) = \frac{8\mu_0 NI}{\pi L}\left[\frac{1}{(\gamma^2+1)\sqrt{\gamma^2+2}}\right] \tag{4.147}$$

where N is the number of coil windings and $\gamma = 0.5445$. Since Helmholtz coils typically consist of N coil windings around a loop, the current of a single-turn coil is equivalent to N times the current I for an N-turn coil. Therefore, substituting NI for I in the derivation provided by Equations 4.143 – 4.145, we arrive at the B-field for an N-turn coil. Equation 4.144 is then used to model all three axes of the Helmholtz cage, which is analogous to calculating the magnetic vector potential A due to both coils, and then taking the curl of A ($\nabla \times A$) in Cartesian coordinates around that point [11].

Chapter 4 presents the mathematical models of the CSACS, pyramidal RWA, and Helmholtz cage. In section 4.1 the rotational kinematics are first derived from a 3-2-1 Euler angle sequence, and then the equations of motion of the CSACS are developed. Using the rotational kinematics and equations of motion, an automatic balancing procedure is developed which numerically determines the CM of the CSACS and then adjusts the position of the linear stepper motors (MMUs), thus eliminating gravitational torques acting on the CSACS. After the CSACS platform is balanced, Section 4.2 incorporates the RWA such that the CSACS platform can be controlled along a trajectory about its yaw-axis. The rotational kinematics and equations of motion are reformulated using quaternion notation, and then a trajectory tracking controller is developed for the CSACS. Lastly, Section 4.3 presents an analytical model which determines the optimal spacing between two square coils for the Helmholtz configuration.

Chapter 5

Simulations and Results

5.1 CSACS Balancing

In order for the CSACS to simulate weightlessness, the composite CM of the rotating system must be exactly aligned with the CR of the air bearing [2]. An important issue to address in the balancing algorithm is system stability, meaning the tendency of the CSACS to stay within its allowable range of motion, free from contact with the pedestal support. An unstable configuration will tip to one side of the air bearing pedestal, and when this occurs, no dynamic data can be collected.

Balancing instability is caused by one of two configurations. Either the CSACS CM is above the CR of the air bearing, or the CM is out farther along the x- or y-axis than it is down the negative z-axis. As the results show, MMUs in the x- and y-axes provide more than enough balancing stability control for the entire system. The only caveat is that operator must ensure the CM z-component remains in the stable region, below the air bearing's CR. Once the system is adequately balanced, automatic balancing control can proceed.

The effectiveness of the auto-balance procedure is measured by the reduction in periodic oscillation of the CSACS. Due to uncertainties in the CSACS inertia tensor and final MMU positioning, the balancing algorithm is most effective upon iteration. Results show that four iterations of the balancing algorithm applied to the CSACS yield the best results.

With Equations 4.26 and 4.14, the balancing algorithm was developed in Matlab to simulate the nonlinear dyanmics of the CSACS. Currently, the CSACS mass and inertia values are approximated from a computer-aided design (CAD) model. The system mass and inertia tensor, along with MMU mass are as given

$$m_{tot} = 7.75 \text{ kg} \qquad m = 0.043 \text{ kg} \qquad I = \begin{bmatrix} 0.07917 & 0.009440 & 0.002150 \\ 0.009440 & 0.07767 & -0.001614 \\ 0.002150 & -0.001614 & 0.1364 \end{bmatrix} \text{ kg} \cdot \text{m}^2$$

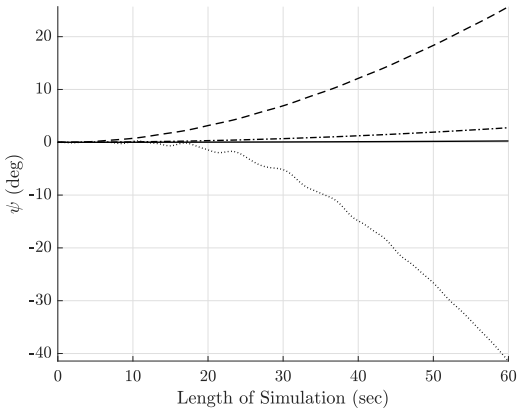
Also, the CSACS was given a zero initial state

$$\begin{aligned} \boldsymbol{\Theta}_0 &= \begin{bmatrix} \psi & \theta & \phi \end{bmatrix}^T \\ &= \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T \text{ rad} \end{aligned}$$

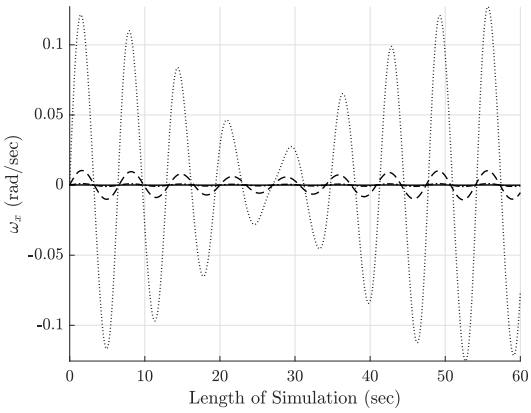
$$\begin{aligned} \dot{\omega}_{b0} &= \begin{bmatrix} \dot{\omega}_x & \dot{\omega}_y & \dot{\omega}_z \end{bmatrix}^T \\ &= \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T \text{ rad/sec} \end{aligned}$$

$$\begin{aligned} r_0 &= \begin{bmatrix} r_x & r_y & r_z \end{bmatrix}^T \\ &= \begin{bmatrix} 0.02454 & -0.01388 & -0.99764 \end{bmatrix}^T \text{ mm} \end{aligned}$$

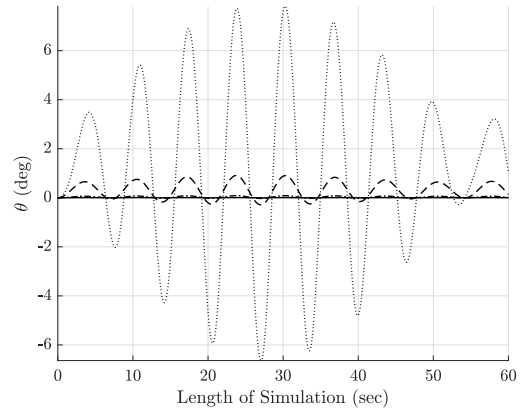
where r_0 is the "true" measured position of the CM relative to the CR at the beginning of the simulation, determined from the CAD model of the CSACS.



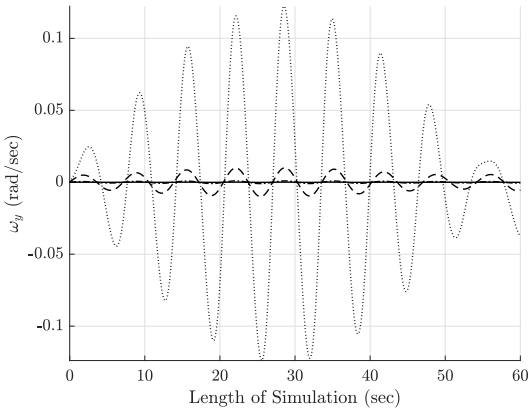
(a) yaw state time-history



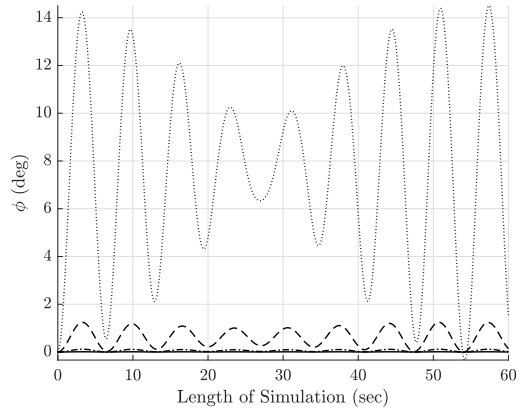
(b) x-angular velocity state time-history



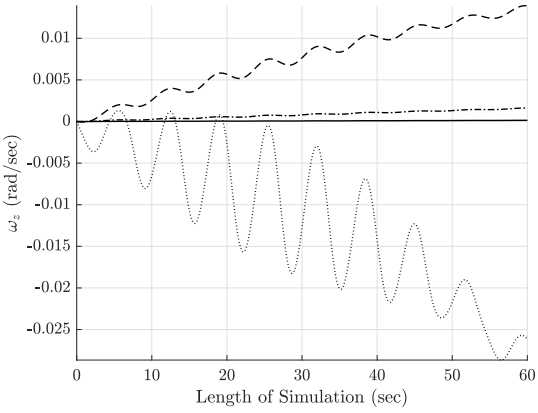
(c) pitch state time-history



(d) y-angular velocity state time-history



(e) roll state time-history



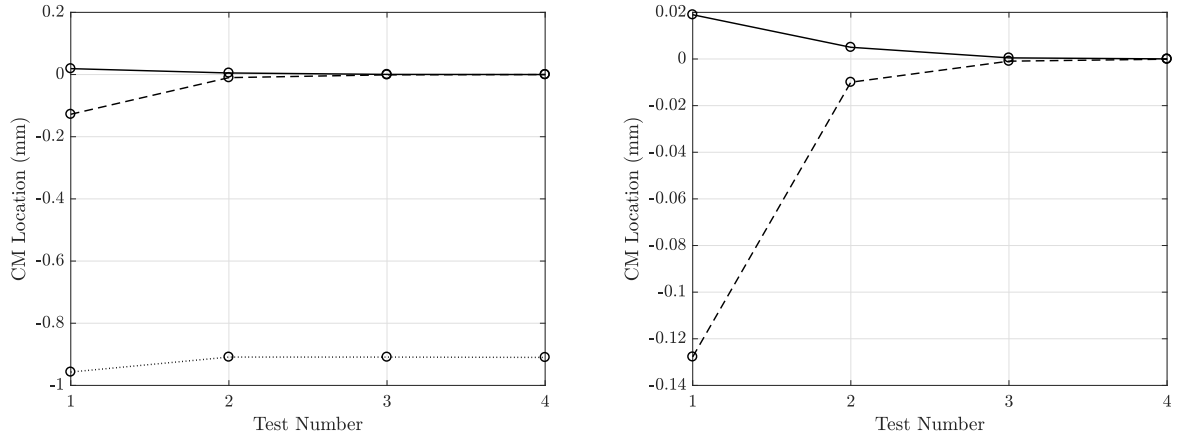
(f) z-angular velocity state time-history

Figure 5.1: Nonlinear Dynamics Simulation of the CSACS. 60 second simulation using the full, nonlinear equations of motion. Test 1 (dotted), Test 2 (dashed), Test 3 (dash-dotted), and Test 4 (solid).

The results presented in Figures 5.1 and 5.2 show how the balancing algorithm works. Test 1, shown by the dotted lines in Figure 5.1, shows the dynamic motion of the CSACS before the MMUs have been adjusted. Initial simulations show quadratic yaw about the table's z-axis, while the roll and pitch oscillate with a period of approximately 7 seconds. Peak amplitudes of each test run are presented in Table 5.1. Test 1 shows a significant amount of unwanted motion, mainly due to the offset CM; however, as the balancing algorithm iterates, periodic motion decreases. This demonstrates why it is crucial to have a robust method for balancing the CSACS. Test 2, shown by the dashed lines in Figure 5.1, shows the dynamic motion of the CSACS after the MMUs have been adjusted after Test 1. Likewise, Test 3, shown by the dash-dotted lines in Figure 5.1, shows the dynamic motion of the CSACS after the MMUs have been adjusted after Test 2. Lastly, Test 4, shown by the solid lines in Figure 5.1, shows the final dynamic motion of the CSACS after the MMUs have been adjusted after Test 3.

After the simulation finishes producing the state time-histories, the algorithm runs a linear least-squares estimator on the position of the CSACS CM with respect to its CR, for each test number. Then, the algorithm adjusts the positions of the MMUs in order to minimize the CM offset. The estimated CM position vectors, error estimations, MMU move distances between iterations, and final MMU position vectors are presented in Table 5.2. We can see that through 4 test iterations, periodic motion of the CSACS is reduced by 99.9% by shifting the MMUs 4.4241 mm in the negative x direction and 25.017 mm in the positive y direction.

Lastly, it is important to note that to reduce the time period of motion T, the r_z component should be relocated closer to the CSACS CM. Relocating the r_z component can be done by manual balancing, or by adding an additional MMU component for the z-axis, if the operator deems this necessary. Currently though, the resolution of the MMUs provide enough balancing control such that the time period of the CSACS is negligible compared to the reduced amplitude of motion each iteration.



(a) **(b)**

Figure 5.2: CSACS CM Location Measurements. x-axis (solid), y-axis (dashed), z-axis (dotted). Figures 5.2a and 5.2b both present the same data. Figure 5.2b focuses on the x- and y-components of the CM since there is no MMU control, *i.e.*, no CM relocation, in the z-axis.

Table 5.1: Tilt Angle Reduction Through Auto-Balancing Procedure. Maximum amplitudes of pitch and roll angles are shown for each test number. As the CM moves closer to the CR each test iteration, maximum amplitude over the state time-history diminishes. Percent improvement is calculated from Test 1 to Test 4.

	Max. Pitch Amplitude θ	Max. Roll Amplitude ϕ
Test 1:	7.81°	14.5°
Test 2:	0.909°	1.25°
Test 3:	0.0868°	0.117°
Test 4:	0.00776°	0.0105°
Percent Improvement:	99.9 %	99.9 %

Table 5.2: CSACS Balancing Algorithm CM Results. Estimated CM vector ($\hat{r}$), difference between estimated and original CM vector (δr), MMU move distance from prior iteration (Δr_m), and final MMU position vector from its original starting point (r_m) are presented for each test number. Recall that the auto-balance procedure is an iterative process and the table values are determined after each test number's state-time histories have been numerically determined.

	$\begin{bmatrix}\hat{r}_x & \hat{r}_y & \hat{r}_z\end{bmatrix}^T$ mm	$\begin{bmatrix}\delta r_x & \delta r_y & \delta r_z\end{bmatrix}^T$ mm
Test 1:	$\begin{bmatrix}0.018986 & -0.12776 & -0.95695\end{bmatrix}^T$	$\begin{bmatrix}-0.0055536 & 0.010989 & 0.040685\end{bmatrix}^T$
Test 2:	$\begin{bmatrix}0.0050237 & -0.0099632 & -0.90883\end{bmatrix}^T$	$\begin{bmatrix}-0.0019516 & 0.012879 & 0.0088813\end{bmatrix}^T$
Test 3:	$\begin{bmatrix}0.00048309 & -0.00093347 & -0.90890\end{bmatrix}^T$	$\begin{bmatrix}-0.024057 & 0.013782 & 0.0088740\end{bmatrix}^T$
Test 4:	$\begin{bmatrix}0.000042752 & -0.000083977 & -0.90992\end{bmatrix}^T$	$\begin{bmatrix}-0.024497 & 0.13867 & 0.087723\end{bmatrix}^T$

	$\begin{bmatrix}\Delta r_{m_x} & \Delta r_{m_y}\end{bmatrix}^T$ mm	$\begin{bmatrix}r_{m_x} & r_{m_y}\end{bmatrix}^T$ mm
Test 1:	$\begin{bmatrix}-3.4235 & 23.037\end{bmatrix}^T$	$\begin{bmatrix}-3.4235 & 23.037\end{bmatrix}^T$
Test 2:	$\begin{bmatrix}-0.90584 & 1.7965\end{bmatrix}^T$	$\begin{bmatrix}-4.3293 & 24.833\end{bmatrix}^T$
Test 3:	$\begin{bmatrix}-0.087108 & 0.16832\end{bmatrix}^T$	$\begin{bmatrix}-4.4164 & 25.002\end{bmatrix}^T$
Test 4:	$\begin{bmatrix}-0.0077087 & 0.015142\end{bmatrix}^T$	$\begin{bmatrix}-4.4241 & 25.017\end{bmatrix}^T$

5.2 CSACS Control

The reaction wheel control model is implemented after the balancing algorithm has effectively translated the MMUs to center the CSACS CM below the air bearing's CR. Using the results from Table 5.2 (Test 4), our initial CM offset vector for control simulation is

$$\hat{r} = \begin{bmatrix} 0.000042752 & -0.000083977 & -0.90992 \end{bmatrix}^T \text{ mm} \tag{5.1}$$

Additionally, applying the final MMU position vector $r_m = \begin{bmatrix} -4.4241 & 25.017 \end{bmatrix}^T$ mm to the CAD model of the CSACS, *i.e.*, manually translating the x- and y-axis MMUs in the model, our new inertia tensor becomes

$$I = \begin{bmatrix} 0.07920 & 0.009731 & 0.002153 \\ 0.009731 & 0.07767 & -0.001633 \\ 0.002153 & -0.001633 & 0.1365 \end{bmatrix} \text{ kg} \cdot \text{m}^2 \tag{5.2}$$

The mass of the CSACS remains $m_{tot} = 7.75$ kg, meaning that the balancing algorithm must be performed with the RWA already positioned on the CSACS.

Now, with the physical parameters set, we can move on to the quaternion feedback controller simulation parameters, presented in Table 5.3. Recalling from Section 4.2.7, the reference trajectory's natural frequency ω_n is based on a nearly critically damped second-order system with maximum percent overshoot criteria of $M_p = 2\%$. Thus, choosing a settling time $t_s = 60$ sec results in a natural frequency of $\omega_n = 4/\zeta t_s = 0.067$ rad/s. With our natural frequency determined, the quaternion feedback gain's are based on the relations shown by Wie *et. al* [61, 62]

$$K_p = 2\omega_n^2 \qquad K_d = 2\zeta\omega_n \tag{5.3}$$

Thus, $K_p = 0.00889$ and $K_d = 0.133$.

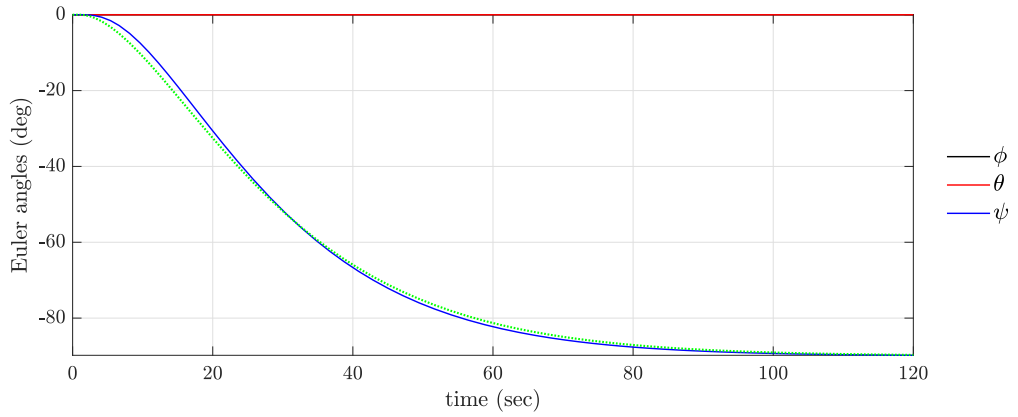
Table 5.3: Quaternion Feedback Controller Simulation Parameters. User defined initial angular velocity and orientation, desired orientation, settling time, and damping ratio. Recall that only yaw-axis control is simulated due to actuation limits. Damping ratio is chosen such that the trajectory response is critically damped, *i.e.*, poles of the trajectory response transfer function are negative, real, and equal, *and* the system does not oscillate.

Initial Angular Velocity:	$\omega_{b0} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ deg/sec
Initial Euler Angles:	$\boldsymbol{\Theta}_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ deg
Desired Euler Angles:	$\boldsymbol{\Theta}_d = \begin{bmatrix} -90 & 0 & 0 \end{bmatrix}^T$ deg
Settling Time:	$t_s = 60$ sec
Damping Ratio:	$\zeta = 1$

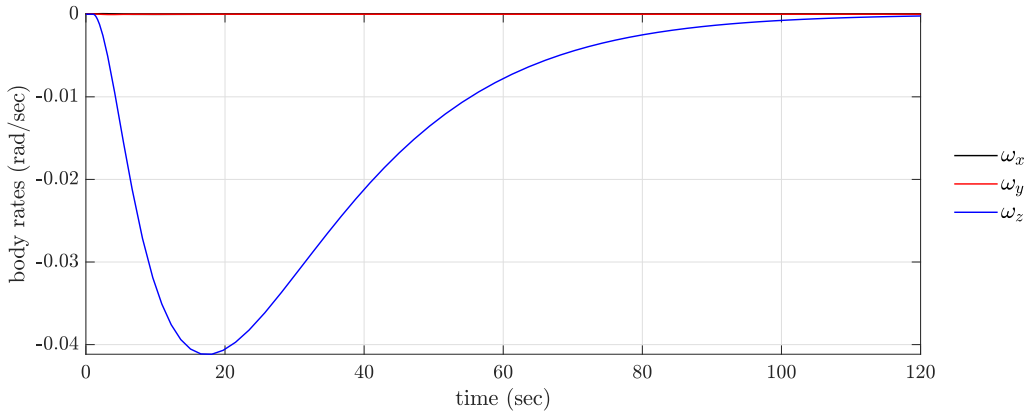
Based on this information and the derivations provided in Section 4.2, the quaternion feedback controller is designed and simulated using Matlab and Simulink. Simulation results from five separate cases follow: four functioning wheels, disabling wheel 1, disabling wheel 2, disabling wheel 3, and disabling wheel 4, respectively.

Case 1 – Four Functioning Wheels

⇒ Power consumption for specified maneuver is 0.40 W.

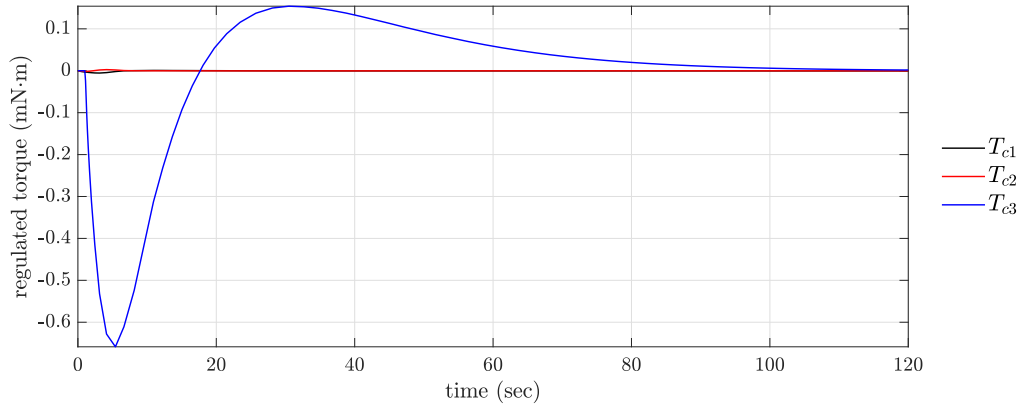


(a) CSACS attitude orientation (dotted green curve is the desired trajectory)

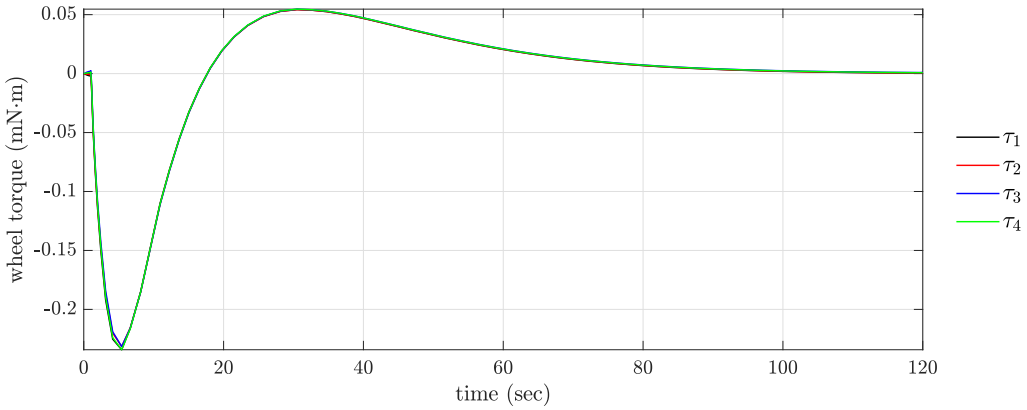


(b) CSACS body rates

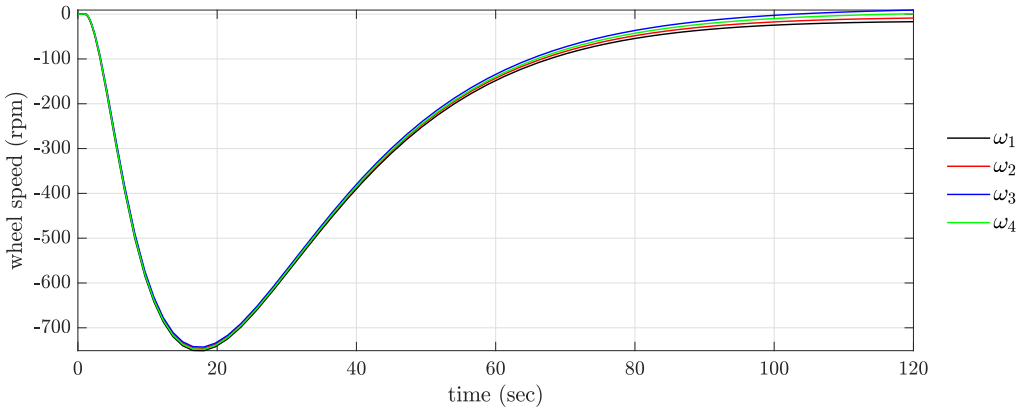
Figure 5.3: Case 1 – CSACS Orientation and Body Rates.



(a) RWA regulated body-axis torques

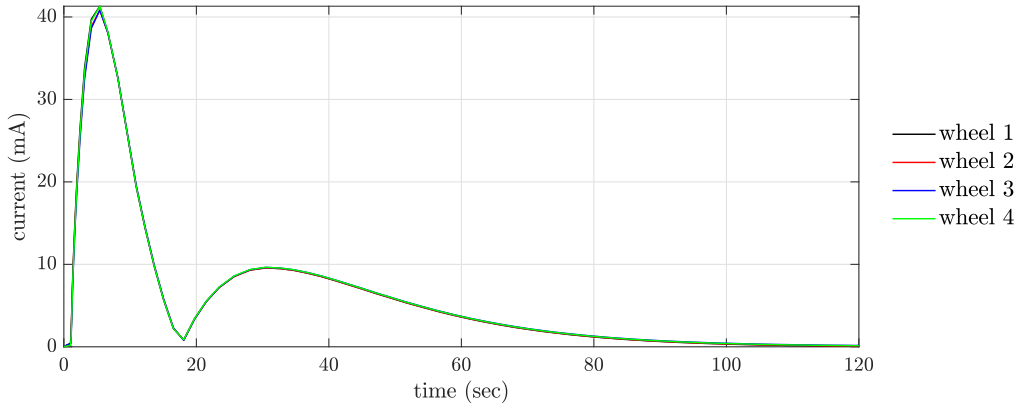


(b) RWA torques

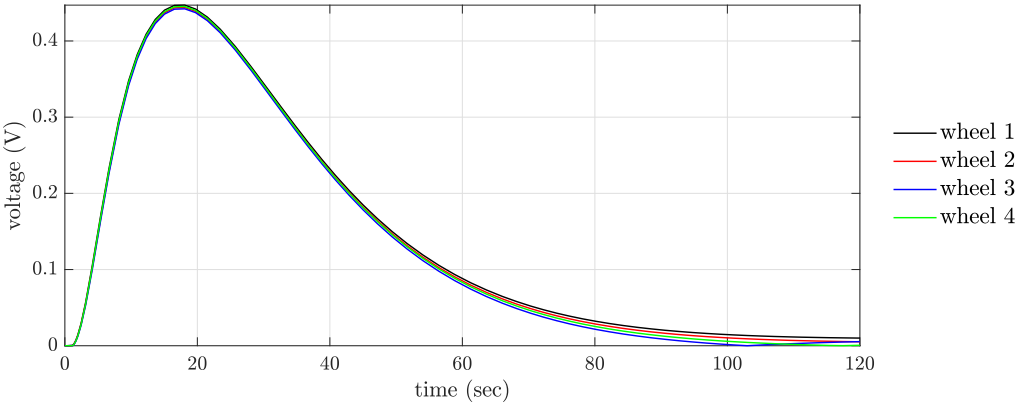


(c) RWA speeds

Figure 5.4: Case 1 – RWA Torques and Wheel Speeds



(a) RWA command current

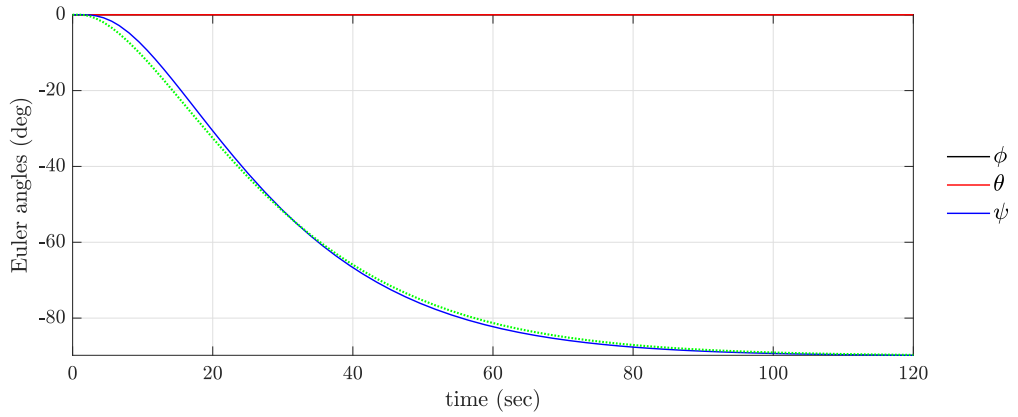


(b) RWA command voltage

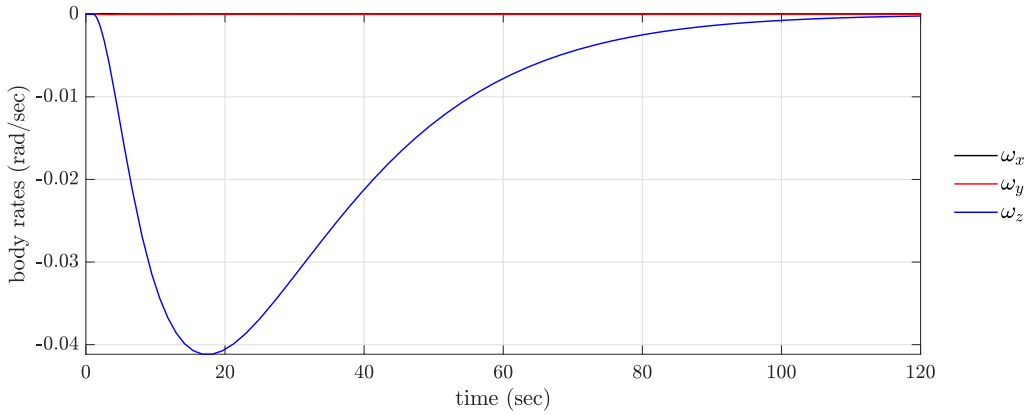
Figure 5.5: Case 1 – RWA Commanded Current and Voltage

Case 2 – Wheel 1 Disabled

⇒ Power consumption for specified maneuver is 0.86 W.

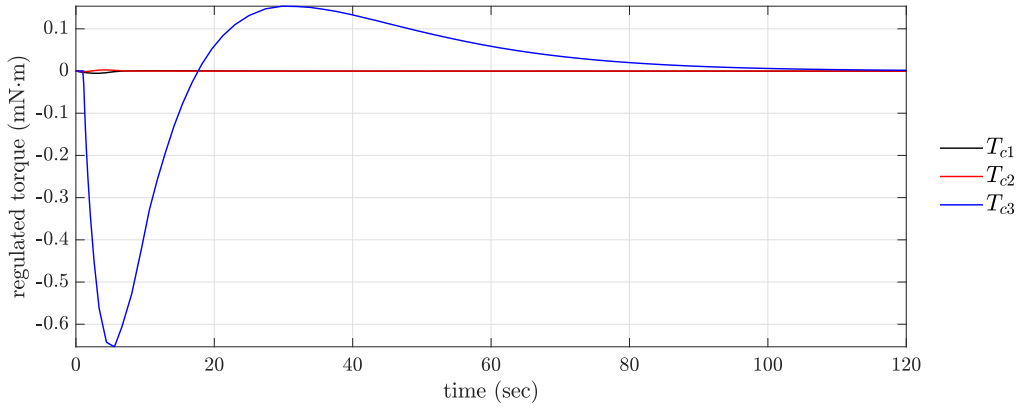


(a) CSACS attitude orientation (dotted green curve is the desired trajectory)

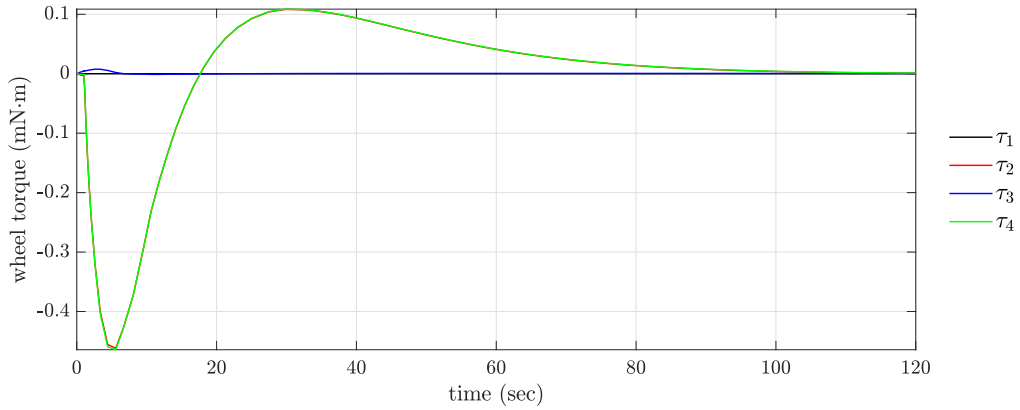


(b) CSACS body rates

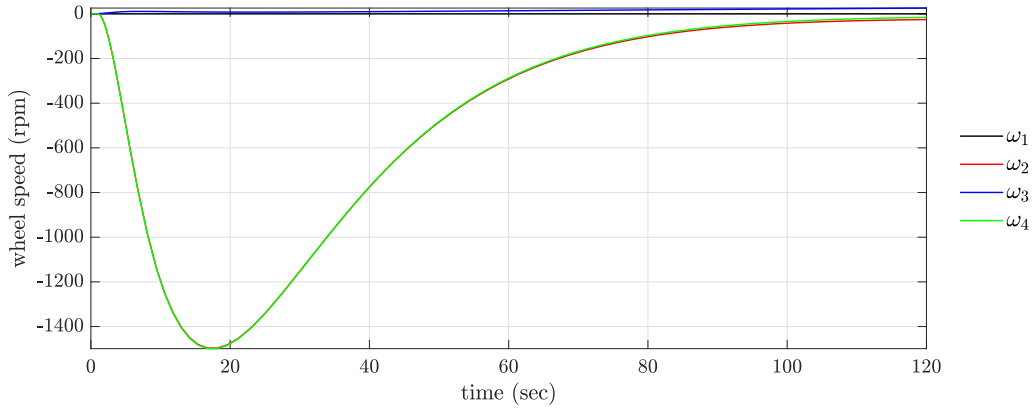
Figure 5.6: Case 2 – CSACS Orientation and Body Rates.



(a) RWA regulated body-axis torques

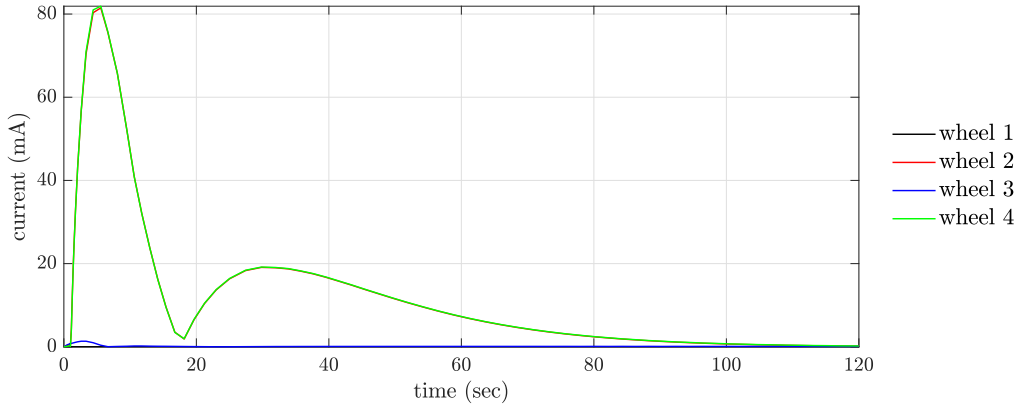


(b) RWA torques

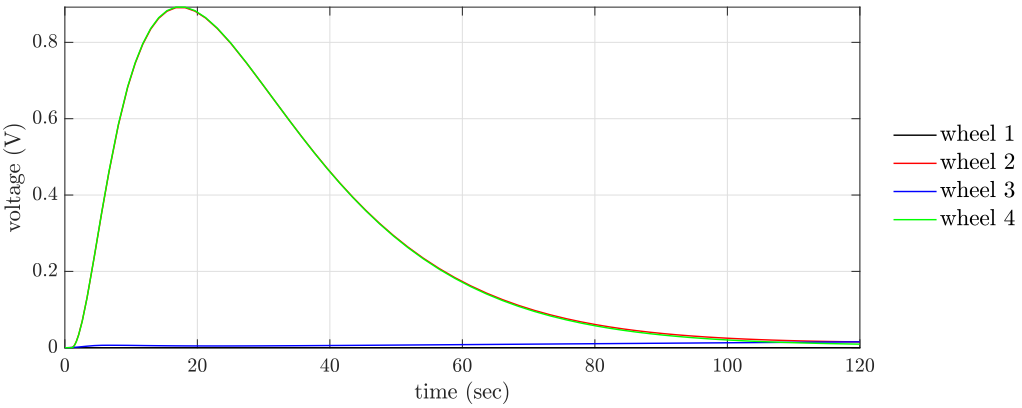


(c) RWA speeds

Figure 5.7: Case 2 – RWA Torques and Wheel Speeds



(a) RWA command current

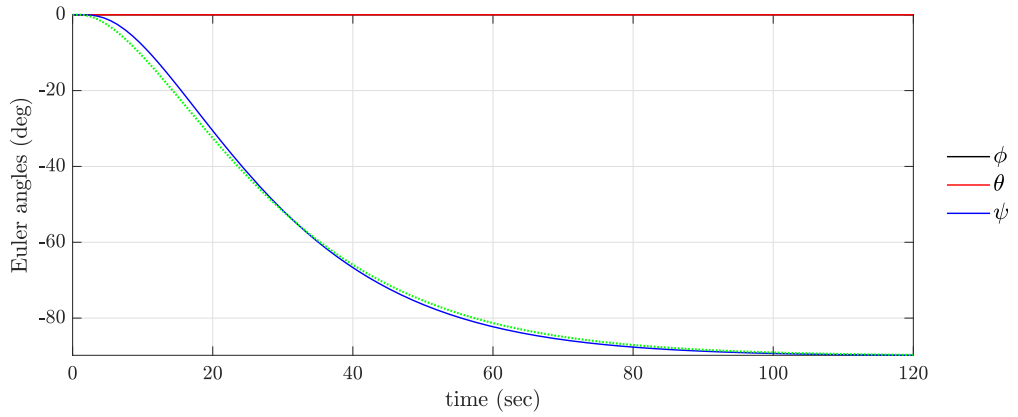


(b) RWA command voltage

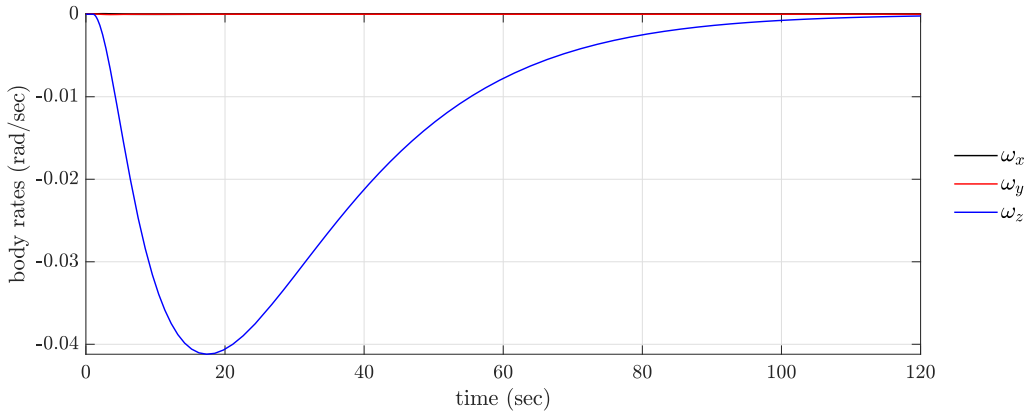
Figure 5.8: Case 2 – RWA Commanded Current and Voltage

Case 3 – Wheel 2 Disabled

⇒ Power consumption for specified maneuver is 1.06 W.

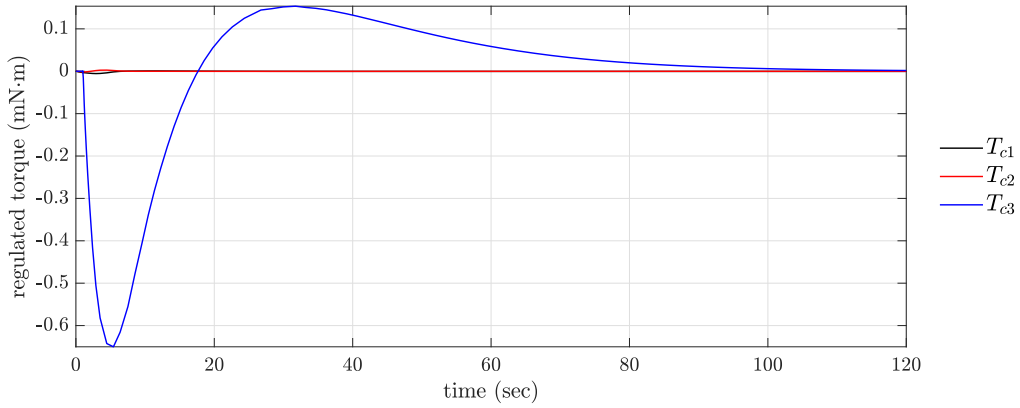


(a) CSACS attitude orientation (dotted green curve is the desired trajectory)

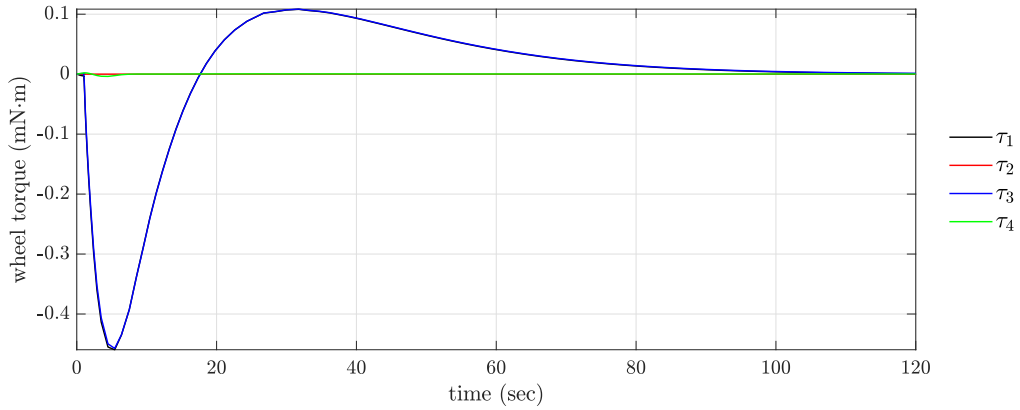


(b) CSACS body rates

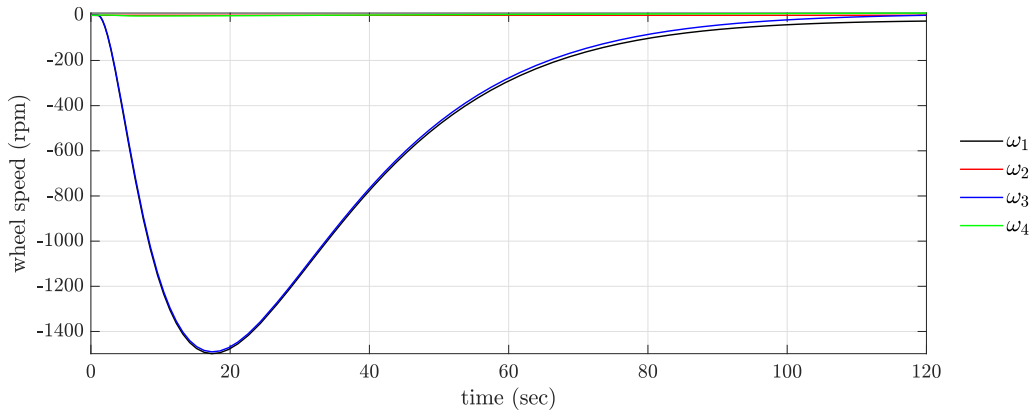
Figure 5.9: Case 3 – CSACS Orientation and Body Rates.



(a) RWA regulated body-axis torques

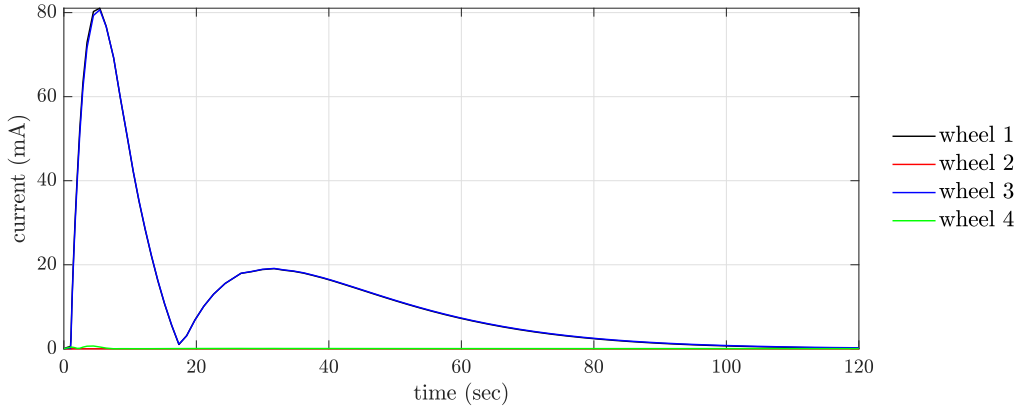


(b) RWA torques

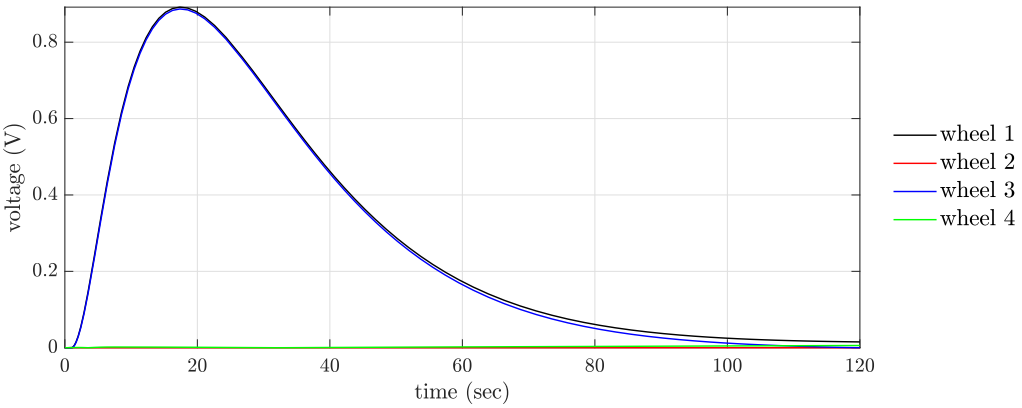


(c) RWA speeds

Figure 5.10: Case 3 – RWA Torques and Wheel Speeds



(a) RWA command current

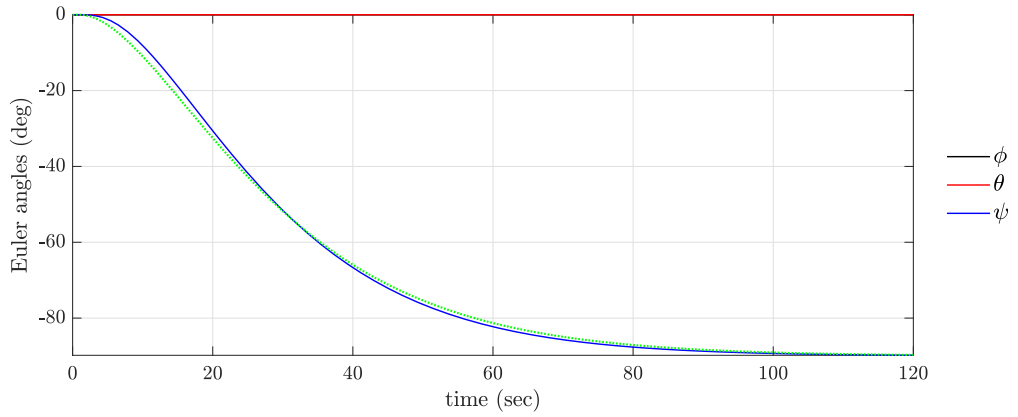


(b) RWA command voltage

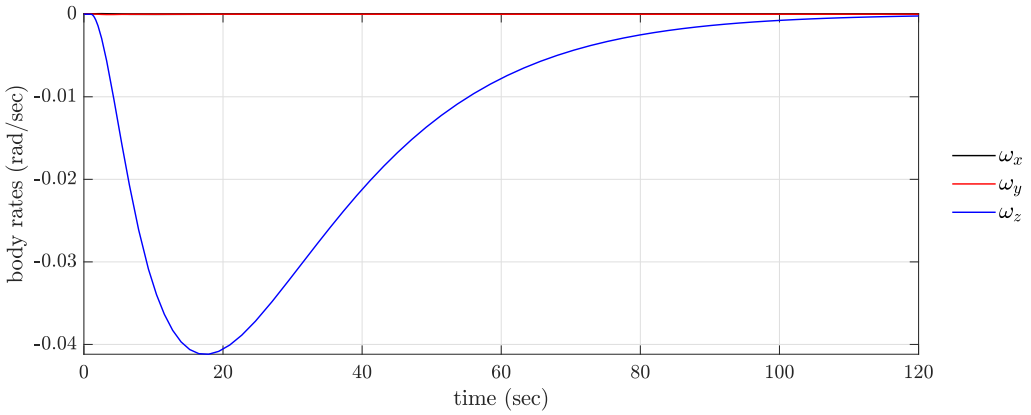
Figure 5.11: Case 3 – RWA Commanded Current and Voltage

Case 4 – Wheel 3 Disabled

$\Rightarrow$ Power consumption for specified maneuver is 0.86 W.

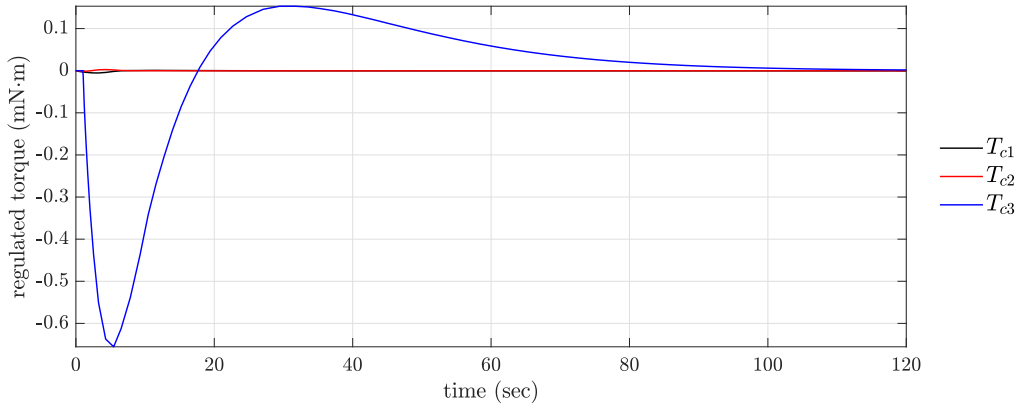


(a) CSACS attitude orientation (dotted green curve is the desired trajectory)

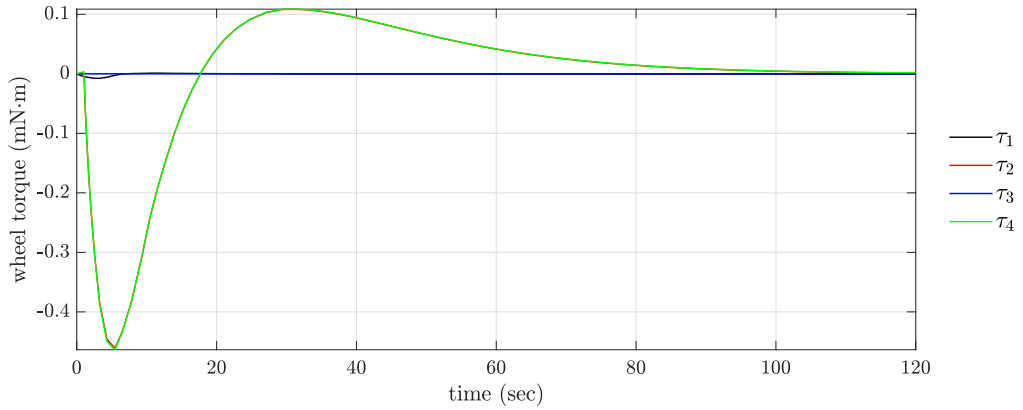


(b) CSACS body rates

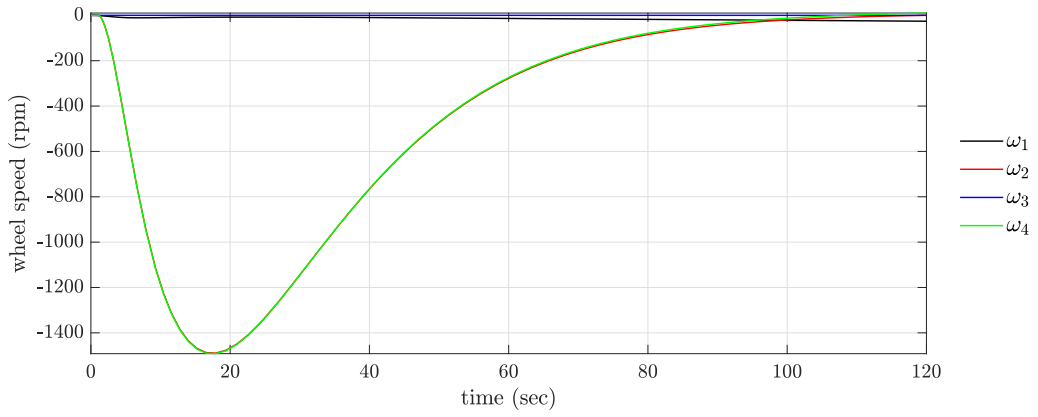
Figure 5.12: Case 4 – CSACS Orientation and Body Rates.



(a) RWA regulated body-axis torques

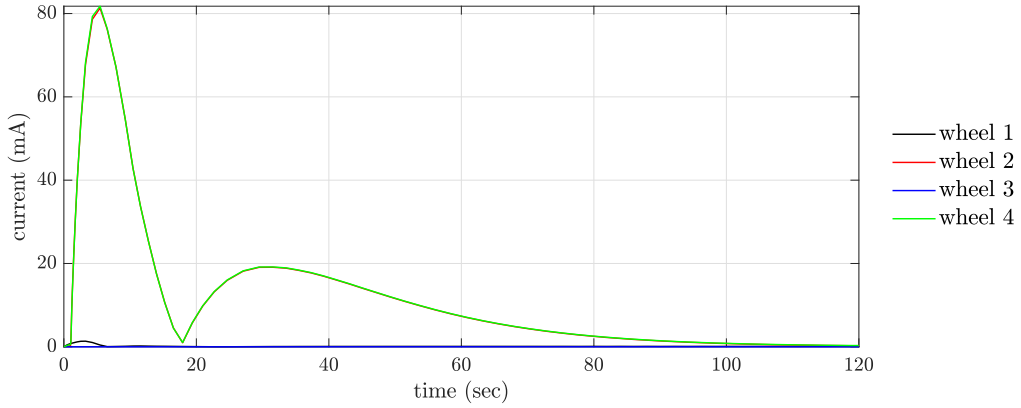


(b) RWA torques

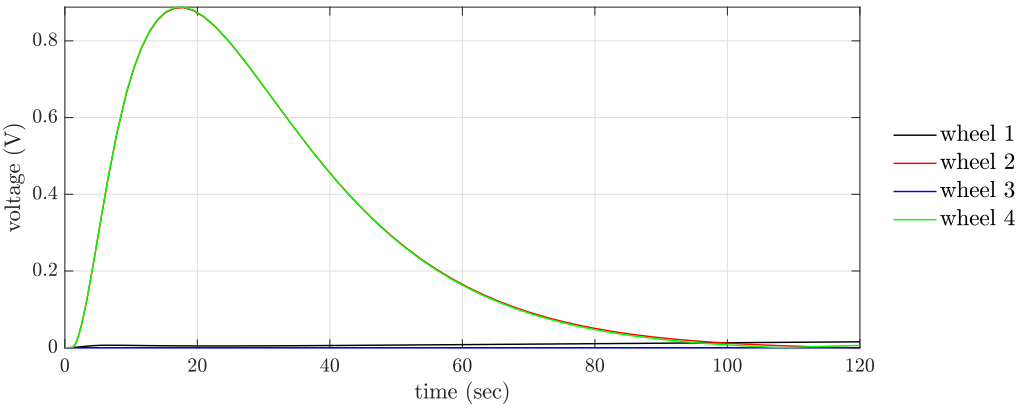


(c) RWA speeds

Figure 5.13: Case 4 – RWA Torques and Wheel Speeds



(a) RWA command current

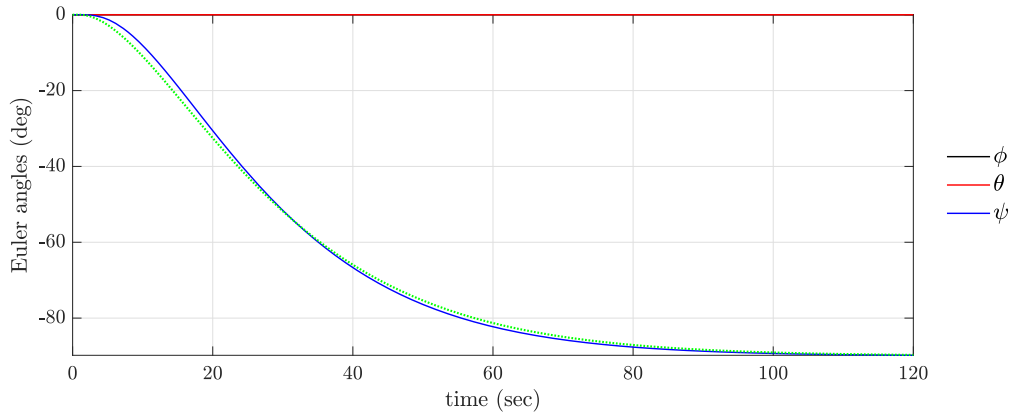


(b) RWA command voltage

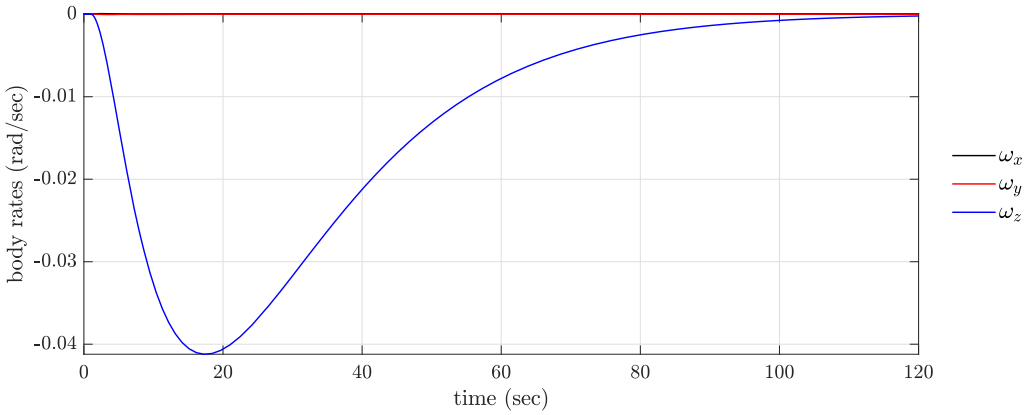
Figure 5.14: Case 4 – RWA Commanded Current and Voltage

Case 5 – Wheel 4 Disabled

⇒ Power consumption for specified maneuver is 1.12 W.

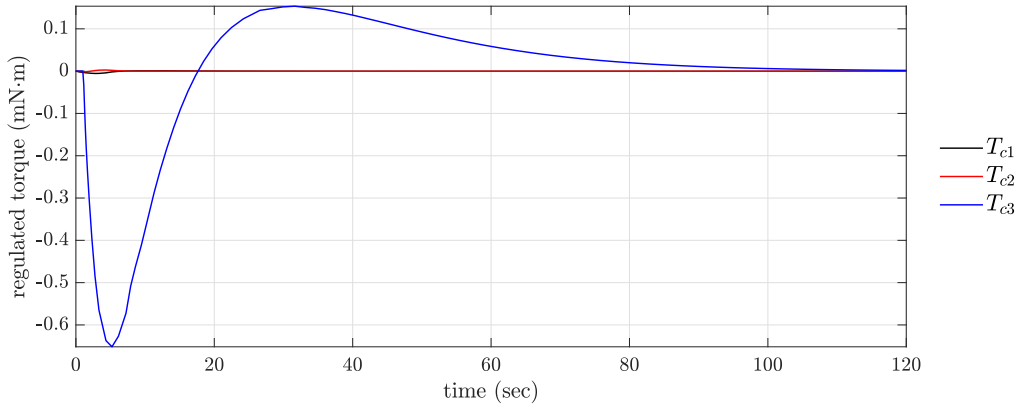


(a) CSACS attitude orientation (dotted green curve is the desired trajectory)

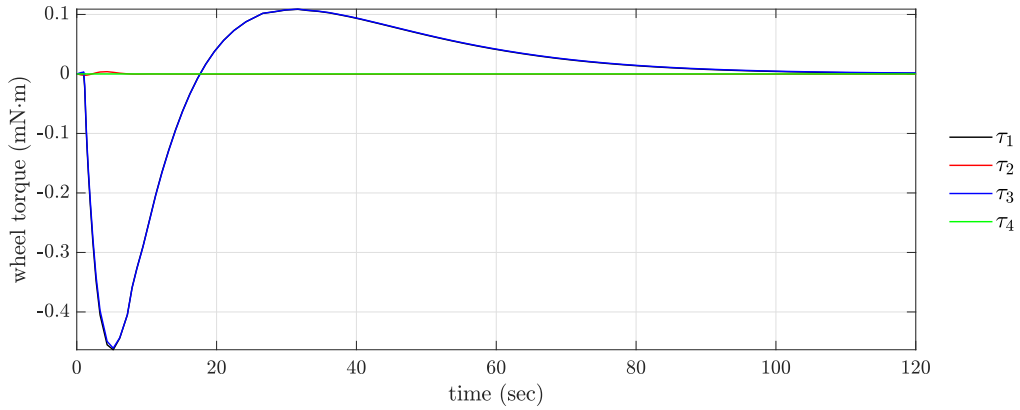


(b) CSACS body rates

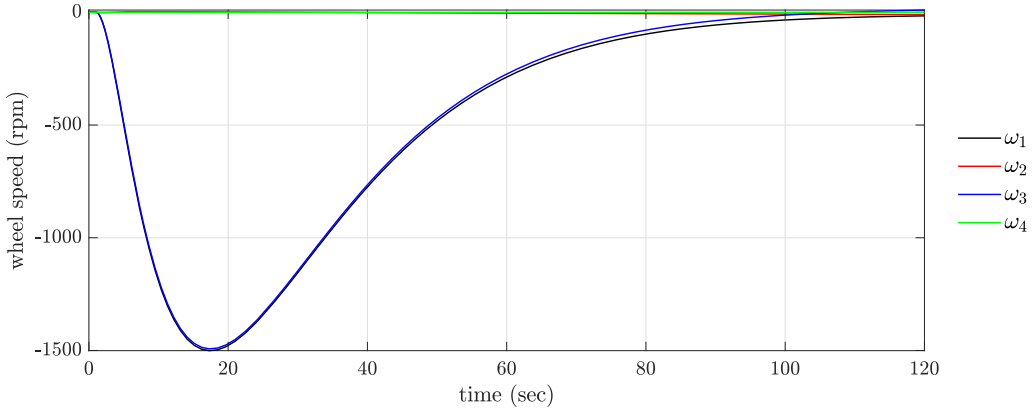
Figure 5.15: Case 5 – CSACS Orientation and Body Rates.



(a) RWA regulated body-axis torques

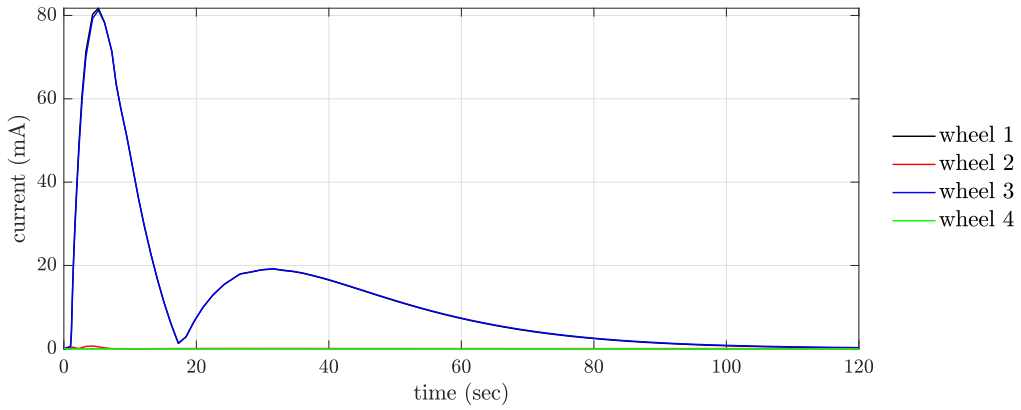


(b) RWA torques

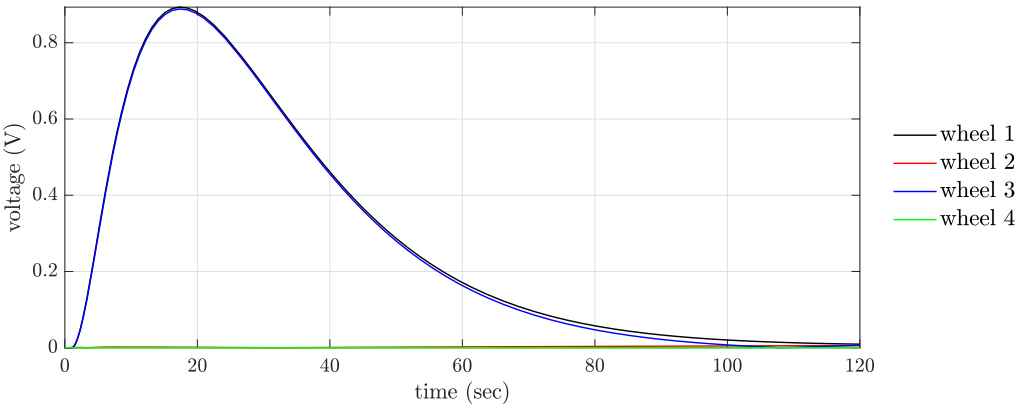


(c) RWA speeds

Figure 5.16: Case 5 – RWA Torques and Wheel Speeds



(a) RWA command current



(b) RWA command voltage

Figure 5.17: Case 5 – RWA Commanded Current and Voltage

The results presented in Figures 5.3 – 5.17 show the quaternion feedback controller simulations for each of the aforementioned cases: four functioning wheels, disabling wheel 1, disabling wheel 2, disabling wheel 3, and disabling wheel 4, respectively. The implementation of a pyramidal RWA shows the effectiveness of redundant actuators for spacecraft attitude control. While the power consumption essentially doubles with the loss of one reaction wheel, the commanded maneuver trajectory is not affected. As settling time increases, power consumption reduces drastically. For example, with four functioning wheels, a 2 minute -90° yaw-maneuver reduces power consumption to approximately 0.07 W, and a 3 minute -90° maneuver to 0.03 W, as presented in Appendices D.1 and D.2, respectively. A 60 second settling time and -90° yaw-rotation trajectory were chosen as the main simulation parameters in Figures 5.3 – 5.17 to mimic the simulation parameters set by Dannemeyer and Tibbs; however, both Dannemeyer and Tibbs implemented different controllers on systems with very different mass properties [7, 56]. Additionally, Dannemeyer and Tibbs did not monitor current and power draw in-the-loop during their simulations. Nonetheless, the simulation results presented in Figures 5.3 – 5.17 fall well within the operating limits of the RWA determined in [7]. If the user wishes to reduce the maneuver time and/or alter the rotation trajectory, he or she should closely monitor the power consumption and motor actuation limits. Based on the motor data provided in [35], the maximum continuous torque, current, and wheel rotation rates are incorporated in the Simulink model such that the user will be notified if the actuators become saturated. Relating the feedback gains to the familiar second-order system terms proved to be effective for keeping the CSACS on it's desired trajectory. This result comes from the fact that the linearized dynamics of the CSACS can assume the form of a second-order system very closely [34, 61, 62]. However, the controller still is PD, so the user may further tune the gains. It is noticed that if K_p is decreased, the frequency of oscillation decreases, and if K_d is increased, the system damps faster.

5.3 Helmholtz Cage

On the basis of the mathematical model described in Section 4.3, the following simulations assess the Helmholtz cage configuration's physical design. Taking into account the real system configuration presented in Table 3.1 of Section 3.3, each set of coils must be inserted one into the other, and so, their sizes cannot be the same in each axis of orientation. Simulation results show that coils of 1.5 m side length, 35 windings, and 5.32 A applied current achieve the ±2 Gauss – $30 \times 30 \times 30$ cm design requirement in the z-axis. Likewise, coils of 1.4492 m side length, 34 windings, and 5.32 A applied current achieve design requirements in the x-axis, and coils of 1.3984 m side length, 33 windings, and 5.32 A applied current achieve design requirements in the y-axis. It is important to note, however, that the operating current to achieve typical LEO magnetic field values (Figure 1.4) should be no larger than 1.2 A.

The results of the z-axis simulations are presented below. The graphs from Figures 5.18, 5.19, and 5.20 show the B_z-component on three planes (x-y, x-z, and y-z). The other components of the field, B_x and B_y, equal zero in each plane. The plot in Figure 5.18 shows the B_z-component on the x-y plane, while the plot in Figure 5.19 shows the same field on the x-z and y-z planes. The plot in Figure 5.20 shows the intensity lines of the B_z-component in the x-y plane. As the lines extend outward, a 1% relative error with respect to B_z at the center of the coils exists. The plots relative to the B_x- and B_y-components are omitted because they are null at every location. Exploiting numerical simulation, we can also analyze the B-field behavior at an arbitrary point different from the system center. Thus, Figures 5.21 and 5.22 show simulations in the $z = 0.15$ m plane. In the same plane, the B_x- and B_y-components are not zero. Figure 5.22 shows intensity lines extending outward at 1% the relative value with respect to B_x. The B_x-component is indistinguishable within 0.2 m of the center of the coils.

In order to correspond experimental data with theoretical results, the Helmholtz cage will undergo an extensive testing campaign. A graduated rail to move multiple magnetometers along each axis will be implemented to set and read the B-field values for each coil pair. The measurements can then be compared to the numerical simulation results presented in this section in order to correctly bias the settings for each axis. To this aim, two acquisitions will be taken at each position along the graduated rail: the first to evaluate Earth's local B-field, which is then subtracted as an offset error, and the second to read the total B-field with the simulator in an operational state. The results reported in Figures 5.24, 5.25, and 5.26 show theoretical results, and how eventual calibration testing will overlay these results with physical recordings.

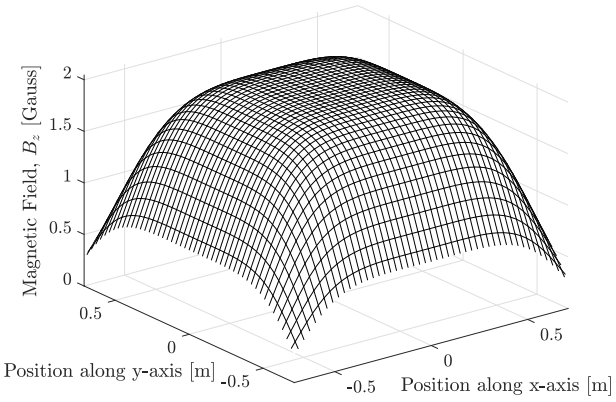


Figure 5.18: B_z on the x-y plane for 1.5 m-long square coils, wound 35 times, and applied 5.32 A current.

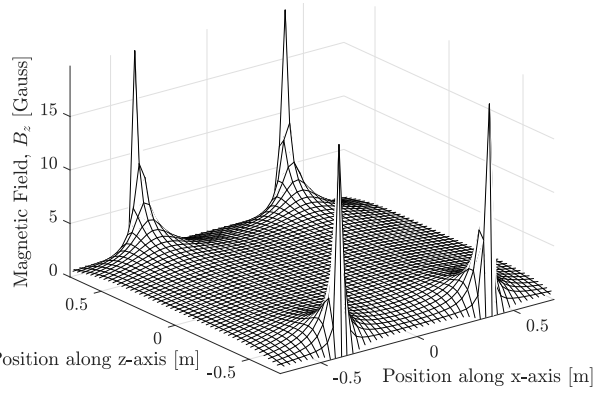


Figure 5.19: B_z on the x-z plane for 1.5 m-long square coils, wound 35 times, and applied 5.32 A current. B-field spikes correspond to the location of the coils. The identical result applies to B_z on the y-z plane.

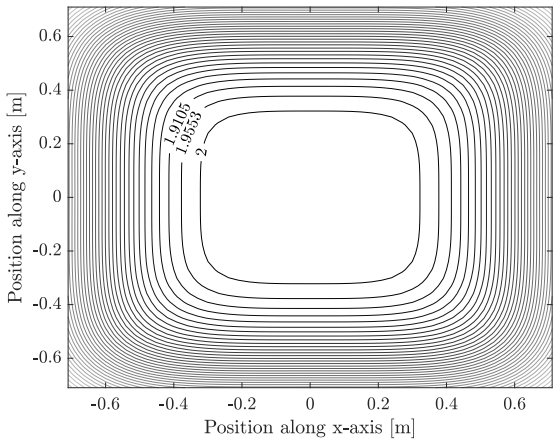


Figure 5.20: Contour plot of B_z on the x-y plane for 1.5 m-long square coils, wound 35 times, and applied 5.32 A current.

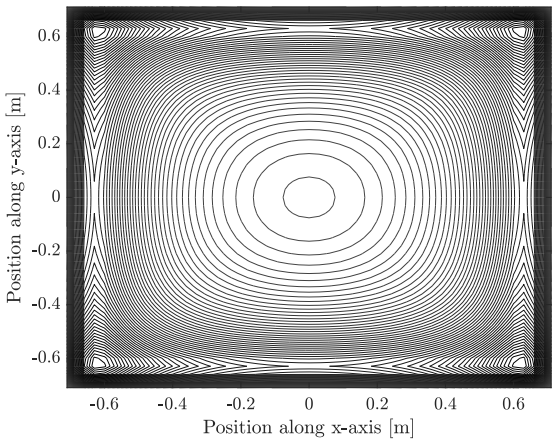


Figure 5.21: Contour plot of B_z on the z = 0.15 m plane for 1.5 m-long square coils, wound 35 times, and applied 5.32 A current. The center contour represents the optimal B-field value centered between both coils. Each line outward corresponds to a 1% change.

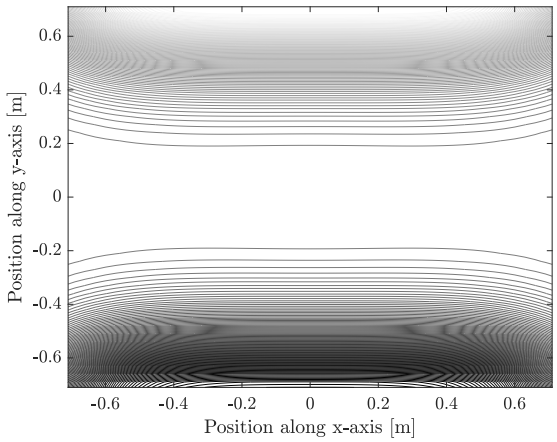


Figure 5.22: Contour plot of B_x on the $z = 0.15$ plane for 1.5 m-long square coils, wound 35 times, and applied 5.32 A current. Each line outward corresponds to a 1% change of the B_x-component. B_x value in center is null.

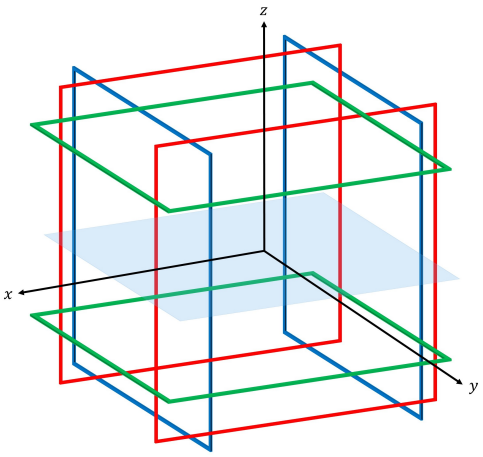


Figure 5.23: Green coils generate B_z, blue coils generate B_x, and red coils generate B_y.

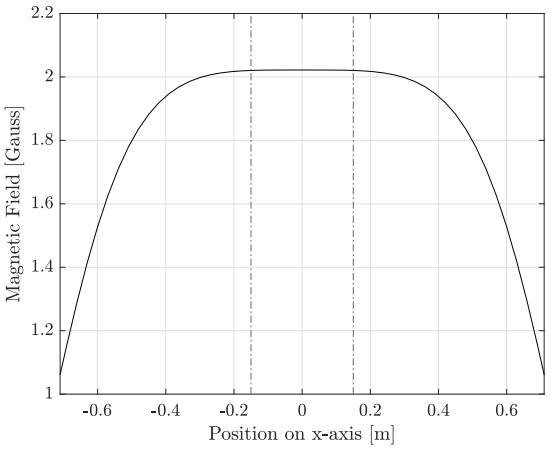


Figure 5.24: B_z numerical simulation data for 1.5 m coil on x-axis. 30 cm region of magnetic field homogeneity is shown.

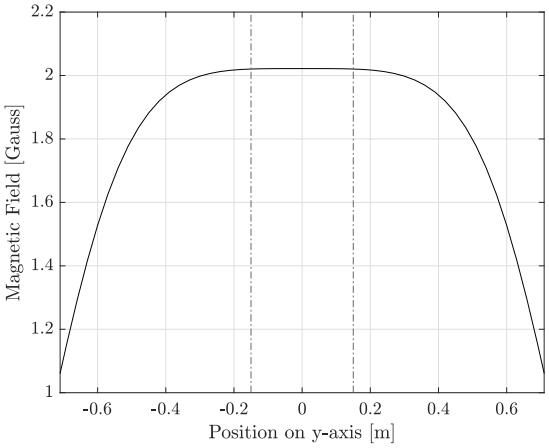


Figure 5.25: B_z numerical simulation data for 1.5 m coil on y-axis. 30 cm region of magnetic field homogeneity is shown.

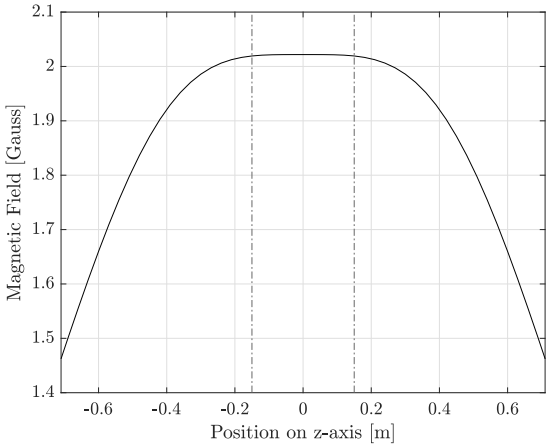


Figure 5.26: B_z numerical simulation data for 1.5 m coil on z-axis. 30 cm region of magnetic field homogeneity is shown.

Chapter 5 presents the simulations and results of the CSACS auto-balancing algorithm, the pyramidal RWA quaternion feedback controller, and the Helmholtz cage. Section 5.1 presents the results CSACS auto-balancing algorithm, which effectively reduces the periodic motion of the CSACS platform by 99.9% after four iterations of adjusting the MMUs. To completely reduce periodic motion and perfectly align the CSACS CM with the air bearing's CR, the r_z component must be relocated closer to the CSACS CM. Relocating the r_z component can be done by manual balancing, or by adding an additional MMU component for the z-axis, if the operator deems this necessary. For the sake of yaw-axis controllability, the resolution of the x- and y-component MMUs provide enough balancing control such that the time period of the CSACS is negligible compared to the reduced amplitude of motion each iteration. With the CSACS balanced, Section 5.2 presents the trajectory tracking simulations for the pyramidal RWA integrated with the CSACS. Five RWA configurations are simulated: four functioning wheels, disabling wheel 1, disabling wheel 2, disabling wheel 3, and disabling wheel 4. By relating second-order system parameters in the generated trajectory to the linearized second-order system dynamics parameters, a nearly critically damped system with 2% criteria produces favorable trajectories for each RWA configuration. Lastly, Section 5.3 simulates the z-axis component of the Helmholtz cage and verifies that the physical parameters of the cage are capable of achieving the ± 2 Gauss – $30 \times 30 \times 30$ cm design requirement. Using the governing equations developed in Section 4.3, simulations for the x- and z-components of the magnetic field produce results that meet the Helmholtz cage design requirements as well.

Chapter 6

Conclusions and Future Work

The CSACS system, developed in the Virginia Tech Space Systems Simulation Laboratory, provides an inexpensive and easy to use CubeSat ADCS simulation environment for the intended use of students, educators, and professionals in both academic and industry settings. The platform is designed using a black-box approach. A VectorNav VN-100 IMU determines the roll, pitch, and yaw of the interchangeable carbon-fiber test platforms, and linear controlled actuators balance the table within approximately 10 millionths of an inch of the CR of a hemispherical gas air bearing. A power regulation board, designed for system modularity, generates power across the platform's hardware. Embedded software controls the capabilities that provide the user with real-time platform balancing.

Utilizing the nonlinear dynamics of the CSACS, a platform balancing procedure was developed that effectively reduces the gravitational torques that influence the motion of the CSACS. A custom-built algorithm determines the composite CM by analyzing the integrated equations of motion of the CSACS. The algorithm then uses the method of least squares to estimate the position vector from the CR to the CM. The CM of the CSACS is then moved near the air bearing's CR by means of linear stepper motor adjustment on the CSACS. Simulation results show that it is possible to autonomously reach a stable equilibrium of the CSACS system. This development shows that an automatic balancing system allows critical savings in space, mass, and complexity compared to traditional manual balancing approaches [41]. After balancing is complete, incorporation of a four wheel RWA shows full yaw-axis control capability of the CSACS.

Additionally, the design and numerical analysis of a Helmholtz cage to test attitude determination and control algorithms for CubeSats up to 3U in size has been conducted. Results show that a ± 2 Gauss magnetic field in a $30 \times 30 \times 30$ cm cube at the center of the cage is achievable. Ultimately, the long term goal for this project is to automate the power supplies to run real-time STK simulations with the CSACS system placed inside the cage. Placing the CSACS platform inside the Helmholtz Cage will expose a CubeSat ADCS to a LEO-simulated environment, both magnetically and gravitationally.

Future work entails proper determination of the mass properties of the CSACS systems. Once adequate mass properties are acquired, these values can be fully integrated with the platform balancing procedure previously developed, and validated through experimental testing. Additionally, once the mass properties of the CSACS are verified, MOI determination of various test articles can be determined on the CSACS by correlating the combined system's periodic motion. Incorporation of the self-balancing algorithm embedded in the control computer should also be examined given the CSACS motor actuation limits. Integration of a reaction wheel stack consisting of four wheels will be implemented on the CSACS platform in order to provide dynamic simulation capability. Full three-axis control can be considered, but is in no way limited by the following techniques: implementing active feedback control of x-, y-, and z-axis MMUs which consistently align the CSACS CM with the air bearing's CR, increasing the size of the reaction wheels, *i.e.*, torque generation, or separating the RWA and individually relocating each reaction wheel away from the CR of the air bearing, thus creating a larger moment arm. Furthermore, calibration of the cage is needed to accurately run these simulations and can be achieved with a magnetometer. Once the Helmholtz cage and the CSACS become integrated, magnetorquers can be incorporated into the CSACS dynamics model to provide additional control, as well as serve as a momentum dumping device for when the reaction wheels become saturated.

Thus, the continued research and development of the CSACS system and Helmholtz cage will eventually lead to a stand-alone solution that enables the rapid protoyping of CubeSat attitude determination control systems for both Virginia Tech, and the public's use.

Bibliography

[1] Beagleboard:BeagleBoneBlack. `http://elinux.org/Beagleboard:BeagleBoneBlack`.

[2] R. Boynton. "Using a Spherical Air Bearing to Simulate Weightlessness". Technical Report 2297, Space Electronics, Inc., 1996. A Presentation at the 55th Annual Conference of the Society of Allied Weight Engineers.

[3] M. R. Brewer. *CubeSat Attitude Determination and Helmholtz Cage Design*. MS Thesis, Air Force Institute of Technology, March 2012.

[4] S. Chesi. *Attitude Control of Nanosatellites Using Shifting Masses*. PhD Dissertation, University of California Santa Cruz, March 2015.

[5] S. Cho, J. Shen, N. H. McClamroch, and D. S. Bernstein. "Equations of Motion for the Triaxial Attitude Control Testbed". In *40th IEEE Conference on Decision and Control*, doi:10.1109/CDC.2001.980388, December 2001. IEEE.

[6] C. W. Crowell. *Development and Analysis of a Small Satellite Attitude Determination and Control System Testbed*. MS Thesis, Massachusetts Institute of Technology, May 2011.

[7] E. R. Dannemeyer. *Design and Analysis of an Attitude Determination and Control Subsystem (ADCS) for AFITs 6U Standard Bus*. MS Thesis, Air Force Institute of Technology, March 2014.

[8] A. Das, J. L. Berg, G. A. Norris, D. F. Cossey, T. J. Strange III, and W. T. Schlaegel. "ASTREX – A Unique Test Bed for CSI Research". In *29th Conference on Decision and Control*, doi:10.1109/CDC.1990.203978, December 1990. IEEE.

[9] B. Doncaster, C. Williams, and J. Shulman. "Nano/Microsatellite Market Forecast". Technical report, SpaceWorks Enterprises, Inc., 2017.

[10] O. Egeland and J. T. Gravdahl. *Modeling and Simulation for Automatic Control*. Marine Cybernetics AS, Norwegian University of Science and Technology, Trondheim, Norway, 2002.

[11] P. Euripides. Square Helmholtz Coils. `http://demonstrations.wolfram.com/SquareHelmholtzCoils`.

[12] A. Farhat, J. Ivase, and A. Snapp. "Attitude Determination and Control System for CubeSat". Technical report, Worcester Polytechnic Institute, 2013.

[13] I. Gavrilovich, S. Krut, M. Gouttefarde, F. Pierrot, and L. Dusseau. "Test Bench for NanoSatellite Attitude Determination and Control System Ground Tests". In *Small Satellites Systems and Services Symposium*, Dynamic Small Satellites in a Dynamic Economy, Porto Petro, Spain, May 2014.

[14] J. W. Geitgey. *The Determination of Remaining Satellite Propellant Using Measured Monents of Inertia*. MS Thesis, Air Force Institute of Technology, June 2006.

[15] D. Greenwood. *Principles of Dynamics*. Prentice-Hall, Inc., Englewood Cliffs, NJ, second edition, 1988.

[16] C. Hall. "Chapter 3 – Kinematics". Lecture Notes, 2008. AOE 4140 Spacecraft Attitude Dynamics and Control.

[17] C. Kakoyiannis and P. Constantinou. "Electrically Small Microstrip Antennas Targeting Miniaturized Satellites: the CubeSat Paradigm". In *Microstrip Antennas*, chapter 12, pages 273–316. IntechOpen Limited, 2018. Mobile Radio Communications Laboratory, School of Electrical and Computer Engineering, National Technical University of Athens, Greece.

[18] H. K. Khalil. *Nonlinear Systems*. Prentice Hall, Upper Saddle River, NJ, third edition, 2002.

[19] B. Kim, E. Velenis, P. Kriengsiri, and P. Tsiotras. "Designing a Low-Cost Spacecraft Simulator". In *IEEE Control Systems Magazine*, doi:10.1109/MCS.2003.1213601, September 2003. IEEE.

[20] J. J. Kim and B. N. Agrawal. "Automatic Mass Balancing of Air-Bearing-Based Three-Axis Rotational Spacecraft Simulator". *Journal of Guidance, Control, and Dynamics*, 32(3):1005–1017, 2009.

[21] A. Klesh, S. Seagraves, M. Bennett, D. Boone, J. Cutler, and H. Bahcivan. "Dynamically Driven Helmholtz Cage for Experimental Attitude Determination". *Advances in the Astronautical Sciences*, 135:147–160, January 2010.

[22] I. Kök. *Comparison and Analysis of Attitude Control Systems of a Satellite Using Reaction Wheel Actuators*. MS Thesis, Luleå University of Technology, October 2012.

[23] S. A. Kowalchuk. *Investigation of Nonlinear Control Strategies Using GPS Simulator and Spacecraft Attitude Control Simulator*. PhD Dissertation, Virginia Polytechnic Institute and State University, September 2007.

[24] J. B. Kuipers. *Quaternions and Rotation Sequences: A Primer With Applications to Orbits, Aerospace, and Virtual Reality*. Princeton University Press, Princeton, NJ, 1999.

[25] E. Kulu. Nanosatellite Database. `http://www.nanosats.eu`, May 2018.

[26] M. Langer and J. Bouwmeester. "Reliability of CubeSats - Statistical Data, Developers Beliefs and the Way Forward". In *30th Annual AIAA/USU Conference on Small Satellites*. AIAA/USU, August 2016. Advanced Technologies II, SSC16-X-2.

[27] M. Langer, M. Weisgerber, J. Bouwmeester, and A. Hoehn. "A Reliability Estimation Tool for Reducing Infant Mortality in Cubesat Missions". In *2017 IEEE Aerospace Conference*, doi:10.1109/AERO.2017.7943598, March 2017. IEEE.

[28] W. J. Larson and J. R. Wertz. *Space Mission Analysis and Design*. Microcosm, Hawthorne, CA, third edition, 1999.

[29] J. Li, M. Post, T. Wright, and R. Lee. "Design of Attitude Control Systems for CubeSat-Class Nanosatellite". *Journal of Control Science and Engineering*, 2013(4):15, 2013.

[30] P. J. Marquis. *Atmospheric Torques and Passive Attitude Stabilization of the LAICE CubeSat*. MS Thesis, Virginia Polytechnic Institute and State University, May 2015.

[31] C. McChesney. *Design of Attitude Control Actuators for a Simulated Spacecraft*. MS Thesis, Air Force Institute of Technology, March 2012.

[32] A. Mehrparbar. "CubeSat Design Specification". Technical report, California Polytechnic State University, February 2014. Revision 13.

[33] A. Mitskevich. "Launch Services Program: Program Level Dispenser and CubeSat Requirements Document". Technical report, NASA, John F. Kennedy Space Center, Florida, January 2014. Revision B.

[34] C. O. Mittelsteadt and E. A. Mehiel. "The Cal Poly Spacecraft Attitude Dynamics Simulator – CP/SADS". In *2007 AIAA Guidance, Navigation and Control Conference and Exhibit*, doi:10.2514/6.2007-6443, August 2007. AIAA.

[35] Maxon Motor. EC 32 flat ø32 mm, brushless, 6 Watt. `https://www.maxonmotor.com/medias/sys_master/root/8825434734622/17-EN-261.pdf`.

[36] T. Olsen. *Design of an Adaptive Balancing System for the Small Satellite Attitude Control Simulator (SSACS)*. MS Thesis, Utah State University, 1995.

[37] M. A. Peck and A. R. Cavender. "An Air Bearing Testbed for Momentum-Control Systems and Spacecraft Line of Sight". In *13th AAS/AIAA Space Flight Mechanics Winter Meeting*, no. AAS 03-127, 2003. AAS/AIAA.

[38] F. Piergentili, G. P. Candini, and M. Zannoni. "Design, Manufacturing, and Test of a Real-Time, Three-Axis Magnetic Field Simulator". *IEEE Transactions on Aerospace and Electronic Systems*, 47(2):1369–1379, April 2011.

[39] H. C. Polat, J. Virgili-Llop, and M. Romano. "Survey, Statistical Analysis and Classification of Launched CubeSat Missions with Emphasis on the Attitude Control Method". *Journal of Small Satellites*, 4(3):513–530, October 2016.

[40] F. M. Poppenk, R. Amini, and G. F. Brouwer. "Design and Application of a Helmholtz Cage for Testing Nano-Satellites". In *6th International Symposiun on Environmental Testing for Space Programmes*, volume 135. ESA Communication Production Office, January 2007.

[41] J. Prado, G. Bisiacchi, L. Reyes, E. Vicente, F. Contreras, M. Mesinas, and A. Juarez. "Three-Axis Air-Bearing Based Platform for Small Satellite Attitude Determination and Control Simulation". *Journal of Applied Research and Technology*, 3(3):222–237, 2005.

[42] M. Romano and B. N. Agrawal. "Acquisition, Tracking and Pointing Control of the Bifocal Relay Mirror Spacecraft". *Acta Asctronautica*, 53:509–519, October 2003.

[43] W. J. Rugh. *Linear System Theory*. Prentice Hall, Upper Saddle River, NJ, second edition, 1996.

[44] K. P. Ryan. *Experimental Testing of the Accuracy of Attitude Determination Solutions for a Spin-Stabilized Spacecraft*. MS Thesis, Utah State University, August 2011.

[45] M. A. Samuels. *The Design and Testing of a Three-Degree-of-Freedom Small Satellite Simulator Using a Linear Controller with Feedback Linearization and Trajectory Generation*. MS Thesis, Utah State University, 2014.

[46] J. L. Schwartz. *The Distributed Spacecraft Attitude Control System Simulator: From Design Concept to Decentralized Control*. PhD Dissertation, Virginia Polytechnic Institute and State University, July 2004.

[47] J. L. Schwartz, M. A. Peck, and C. D. Hall. "Historical Review of Air-Bearing Spacecraft Simulators". *Journal of Guidance, Control, and Dynamics*, 26(4):513–522, 2003.

[48] M. J. Sidi. *Spacecraft Dynamics and Control: A Practical Engineering Approach*. Cambridge University Press, Cambridge, United Kingdom, 1997.

[49] A. J. Sørensen. "Marine Control Systems: Propulsion and Motion Control of Ships and Ocean Structures". Lecture Notes, 2013. Department of Marine Technology, NTNU.

[50] C. Sultan. "Attitude Kinematics I: Rotation Matrix and Euler Angles". Lecture Slides, 2014. AOE 4140 Spacecraft Attitude Dynamics and Control.

[51] C. Sultan. "Attitude Kinematics II: Euler Angle and Axis, Euler Parameters and Kinematic Differential Equations". Lecture Slides, 2014. AOE 4140 Spacecraft Attitude Dynamics and Control.

[52] C. Sultan. "Rotational Kinematics L6". Lecture Slides, 2016. AOE 5204 Vehicle Dynamics and Control.

[53] C. Sultan. "Rotational Kinematics L7". Lecture Slides, 2016. AOE 5204 Vehicle Dynamics and Control.

[54] C. Sultan. "Lecture 3: Stability of Equilibria – Time-Invariant Systems". Lecture Notes, 2017. AOE 5774 Nonlinear Systems Theory.

[55] D. Thomas, A. Wolosik, and J. Black. "CubeSat Attitude Control Simulator Design". In *2018 AIAA Modeling and Simulation Technologies Conference*, doi:10.2514/6.2018-1391.c1, January 2018. AIAA.

[56] M. L. Tibbs. *Design and Test of an Attitude Determination and Control System for a 6U CubeSat Using AFIT's CubeSat Testbed*. MS Thesis, Air Force Institute of Technology, March 2015.

[57] M. Tolmasoff, R. D. Santos, and C. Venturini. "Improving Mission Success of CubeSats". Technical report, The Boeing Company, June 2017.

[58] D. A. Vallado and W. D. McClain. *Fundamentals of Astrodynamics and Applications*. Microcosm Press, Hawthorne, CA, third edition, 2007.

[59] VectorNav Technologies, Inertial Navigation Modules. *VN-100 User Manual*. Document Number UM001 v2.23.

[60] J. Westerhoff, G. Earle, R. Bishop, G. R. Swenson, S. Vadas, J. Clemmons, R. Davidson, L. Fanelli, C. Fish, V. Garg, A. Ghosh, B. B. Jagannatha, E. Kroeker, P. Marquis, D. Martin, S. Noel, C. Orr, and R. Robertson. "LAICE CubeSat Mission for Gravity Wave Studies". *Advances in Space Research*, 56(7):1413–1427, October 2015.

[61] B. Wie. *Space Vehicle Dynamics and Control*. American Institue of Aeronautics and Astronautics, Inc., Reston, VA, second edition, 2008.

[62] B. Wie, H. Weiss, and A. Arapostathis. "Quaternion Feedback Regulator for Spacecraft Eigenaxis Rotations". *Journal of Guidance, Control, and Dynamics*, 12(3):375–380, 1989.

[63] A. Wolosik and J. Black. "Development of a Low-Flying CubeSat Mission for F-Region Characterization". In *55th AIAA Aerospace Sciences Meeting*, doi:10.2514/6.2017-0161, January 2017. AIAA.

[64] H. Woo, O. R. Perez, S. Chesi, and M. Romano. "CubeSat Three Axis Simulator (CubeTAS)". In *AIAA Modeling and Simulation Technologies Conference*, doi:10.2514/6.2011-6271, August 2011. AIAA.

[65] T. Wright. *Development of Magnetometer-Based Orbit and Attitude Determination for Nanosatellites*. MS Thesis, York University, August 2014.

[66] J. S. Young. *Development of an Automatic Balancing System for a Small Satellite Attitude Control Simulator*. MS Thesis, Utah State University, 1998.

Appendix A

CSACS Nonlinear Equations of Motion

A.1 Expanded B Vector Terms

Terms for the 3×1 column vector B in Equation 4.25

$$B_x = (-2m_{tot}r_y r_z + I_{zy})\omega_y^2 + (2m_{tot}r_y r_z - I_{yz})\omega_z^2 + (-m_{tot}r_x r_z + I_{xz})\omega_x\omega_y + (m_{tot}r_x r_y - I_{xy})\omega_x\omega_z + (m_{tot}r_y^2 - m_{tot}r_z^2 - I_{yy} + I_{zz})\omega_y\omega_z$$

$$B_y = (2m_{tot}r_x r_z - I_{zx})\omega_x^2 + (-2m_{tot}r_x r_z + I_{xz})\omega_z^2 + (m_{tot}r_y r_z - I_{zy})\omega_x\omega_y + (-m_{tot}r_x r_y + I_{xy})\omega_y\omega_z + (-m_{tot}r_x^2 - m_{tot}r_z^2 + I_{xx} - I_{zz})\omega_x\omega_z$$

$$B_z = (-2m_{tot}r_x r_y + I_{xy})\omega_x^2 + (2m_{tot}r_x r_y - I_{xy})\omega_y^2 + (-m_{tot}r_y r_z + I_{yz})\omega_x\omega_z + (-m_{tot}r_x r_z + I_{xz})\omega_y\omega_z + (m_{tot}r_x^2 - m_{tot}r_y^2 - I_{xx} + I_{yy})\omega_x\omega_y$$

Appendix B

Locating the CM of the CSACS

B.1 Expanded Φ Matrix Terms

Terms for the 3×3 matrix $\mathbf{\Phi}$ in Equation 4.31

$$\Phi_{12} = \frac{-m_{tot} g \Delta t}{2 I_{xx}} ((\cos\phi \cos\theta)_{t_2} + (\cos\phi \cos\theta)_{t_1})$$

$$\Phi_{13} = \frac{m_{tot} g \Delta t}{2 I_{xx}} ((\sin\phi \cos\theta)_{t_2} + (\sin\phi \cos\theta)_{t_1})$$

$$\Phi_{21} = \frac{m_{tot} g \Delta t}{2 I_{yy}} ((\cos\phi \cos\theta)_{t_2} + (\cos\phi \cos\theta)_{t_1})$$

$$\Phi_{23} = \frac{m_{tot} g \Delta t}{2 I_{yy}} ((\sin\theta)_{t_2} + (\sin\theta)_{t_1})$$

$$\Phi_{31} = \frac{-m_{tot} g \Delta t}{2 I_{zz}} ((\sin\phi \cos\theta)_{t_2} + (\sin\phi \cos\theta)_{t_1})$$

$$\Phi_{32} = \frac{-m_{tot} g \Delta t}{2 I_{zz}} ((\sin\theta)_{t_2} + (\sin\theta)_{t_1})$$

B.2 Formulation of the Method of Linear Least Squares

To derive a least squares estimator for x, consider the system

$$y = Ax + \epsilon \tag{B.1}$$

where $a_{n0}, a_{n1}, a_{n2}, \cdots, a_{nm}$ are $m+1$ basis functions contained in row n of the matrix A. It is important to note that the terminology "linear" refers only to the system's dependence on its parameters, that is, the x terms. The functions themselves can be highly nonlinear, *e.g.*, the a terms can be sinusoids, as in $y = x_0 + \sin(\omega t)x_1 + \cos(\omega t)x_2$. Denoting the estimated values for x as $\hat{x}$, Equation B.1 can be expanded as

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} a_{10} & a_{11} & a_{12} & \cdots & a_{1m} \\ a_{20} & a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n0} & a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix} \begin{bmatrix} \hat{x}_0 \\ \hat{x}_1 \\ \vdots \\ \hat{x}_m \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix} \tag{B.2}$$

where $A \in \mathbb{R}^{n\times(m+1)}$ is the matrix of the calculated values of the basis functions at the measured values of the independent variables, with m being the number of variables in the system, and n the number of data points. Since $n > m+1$, most of the time, A is not a square matrix and has full column rank, *i.e.*, full rank. The column vector $y \in \mathbb{R}^{n\times1}$ contains the observed values of the dependent variable, the column vector $\hat{x} \in \mathbb{R}^{(m+1)\times1}$ contains the unknown estimated values for x, and the column vector $\epsilon \in \mathbb{R}^{n\times1}$ contains the residuals.

The idea behind the OLS consists of choosing $\hat{x}_i$ such that the sum of the squared residuals, *i.e.*, $\sum_{i=1}^{n} \epsilon_i^2$, in the sample is as small as possible. Mathematically, this means that in order to estimate $\hat{x}$, we need to minimize $\sum_{i=1}^{n} \epsilon_i^2$.

Exploiting the fact that

$$\sum_{i=1}^{n} \epsilon_i^2 = \begin{bmatrix} \epsilon_1 & \epsilon_2 & \cdots & \epsilon_n \end{bmatrix} \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix} = \|\epsilon\|^2 = \epsilon^T \epsilon \tag{B.3}$$

and knowing that $\epsilon = y - A\hat{x}$ (from Equation B.2), we can substitute this relationship into Equation B.3 to arrive at

$$\epsilon^T \epsilon = (y - A\hat{x})^T (y - A\hat{x}) \tag{B.4}$$

This leaves us with the following minimization problem:

$$\min_{\hat{x}} \quad \epsilon^T \epsilon = (y - A\hat{x})^T (y - A\hat{x}) \tag{B.5}$$

Equation B.5 can be expanded as follows

$$\min_{\hat{x}} \quad \epsilon^T \epsilon = \left(y^T - \hat{x}^T A^T\right)(y - A\hat{x}) \tag{B.6a}$$

$$\min_{\hat{x}} \quad \epsilon^T \epsilon = y^T y - y^T A\hat{x} - \hat{x}^T A^T y + \hat{x}^T A^T A\hat{x} \tag{B.6b}$$

$$\min_{\hat{x}} \quad \epsilon^T \epsilon = y^T y - 2\hat{x}^T A^T y + \hat{x}^T A^T A\hat{x} \tag{B.6c}$$

The simplification from Equation B.6b to B.6c is performed by understanding that $\hat{x}^T A^T y = \left(y^T A\hat{x}\right)^T = y^T A\hat{x}$. Being that both terms are scalars, the transpose is identical.

Now, in order to minimize the expression in Equation B.6c, we must differentiate with respect to $\hat{x}$ and set the derivative equal to zero. In order to do so, we will utilize the following statements:

$$\frac{\partial\left(\hat{x}^T A^T y\right)}{\partial \hat{x}} = \nabla_{\hat{x}}\left(\hat{x}^T \left[A^T y\right]\right) = A^T y$$

$$\frac{\partial\left(\hat{x}^T A^T A\hat{x}\right)}{\partial \hat{x}} = \nabla_{\hat{x}}\left(\hat{x}^T \left[A^T A\right]\hat{x}\right) = 2A^T A\hat{x}$$

Using the previously defined statements, we can minimize $\epsilon^T\epsilon$ with respect to $\hat{x}$ by computing the gradient $\nabla_{\hat{x}}\left(\epsilon^T\epsilon\right)$ and solving $\nabla_{\hat{x}}\left(\epsilon^T\epsilon\right) = 0$

$$\nabla_{\hat{x}}\left(\epsilon^T\epsilon\right) = -2A^Ty + 2A^TA\hat{x} = 0 \tag{B.7a}$$

$$= A^TA\hat{x} = A^Ty \tag{B.7b}$$

Pre-multiplying Equation B.7b with $\left(A^TA\right)^{-1}$ gives us the least squares estimator for x

$$\hat{x} = A^{\dagger}y \tag{B.8}$$

where $A^{\dagger} = \left(A^TA\right)^{-1}A^T$ is the pseudoinverse of A.

Appendix C

Quaternion Feedback Control

C.1 Quaternion Product

Following the derivation provided in [24], we define the arbitrary quaternions $\overline{\mathbf{p}}$ and $\overline{\mathbf{q}}$ as

$$\overline{\mathbf{p}} = p_0 + \mathbf{p} = p_0 + \hat{\mathbf{i}}p_1 + \hat{\mathbf{j}}p_2 + \hat{\mathbf{k}}p_3 \tag{C.1}$$

$$\overline{\mathbf{q}} = q_0 + \mathbf{q} = q_0 + \hat{\mathbf{i}}q_1 + \hat{\mathbf{j}}q_2 + \hat{\mathbf{k}}q_3 \tag{C.2}$$

Then, the *quaternion product* of $\overline{\mathbf{p}}$ and $\overline{\mathbf{q}}$ is defined as

$$\overline{\mathbf{p}} \otimes \overline{\mathbf{q}} = p_0 q_0 - \overline{\mathbf{p}} \cdot \overline{\mathbf{q}} + p_0\overline{\mathbf{q}} + q_0\overline{\mathbf{p}} + \overline{\mathbf{p}} \times \overline{\mathbf{q}} \tag{C.3}$$

If we then designate the product as the quaternion vector $\overline{\mathbf{r}}$, we have

$$\overline{\mathbf{p}} \otimes \overline{\mathbf{q}} = \overline{\mathbf{r}} = r_0 + \mathbf{r} = r_0 + \hat{\mathbf{i}}r_1 + \hat{\mathbf{j}}r_2 + \hat{\mathbf{k}}r_3 \tag{C.4}$$

then we have

$$r_0 = p_0 q_0 - p_1 q_1 - p_2 q_2 - p_3 q_3 \tag{C.5}$$

$$r_1 = p_0 q_1 + p_1 q_0 + p_2 q_3 - p_3 q_2 \tag{C.6}$$

$$r_2 = p_0 q_2 - p_1 q_3 + p_2 q_0 + p_3 q_1 \tag{C.7}$$

$$r_3 = p_0 q_3 + p_1 q_2 - p_2 q_1 + p_3 q_0 \tag{C.8}$$

or, if written in matrix form

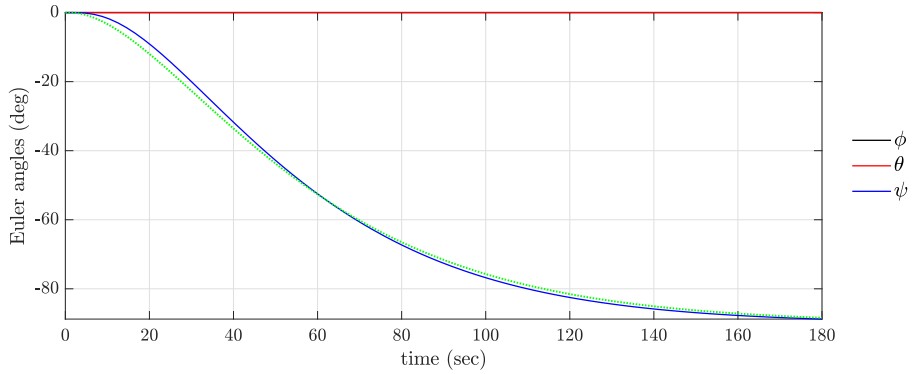
$$\begin{bmatrix} r_0 \\ r_1 \\ r_2 \\ r_3 \end{bmatrix} = \begin{bmatrix} p_0 & -p_1 & -p_2 & -p_3 \\ p_1 & p_0 & -p_3 & p_2 \\ p_2 & p_3 & p_0 & -p_1 \\ p_3 & -p_2 & p_1 & p_0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \tag{C.9}$$

Appendix D

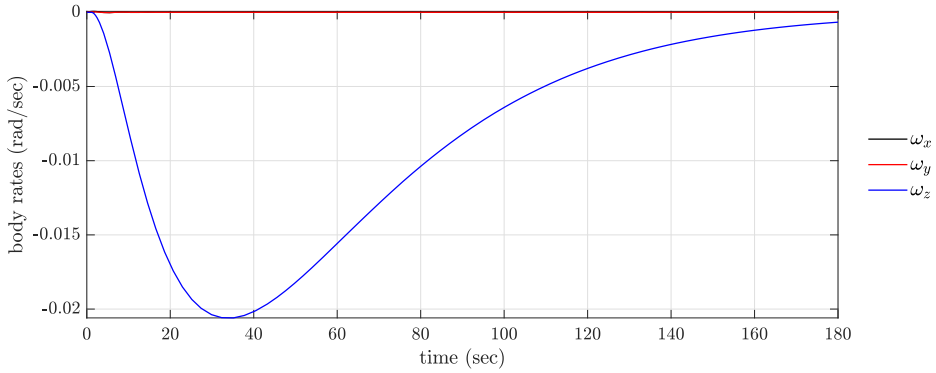
Additional CSACS Control Results

D.1 2 Minute Settling Time Maneuver

Four functioning wheels, -90° yaw-rotation: Power consumption is 0.07 W.

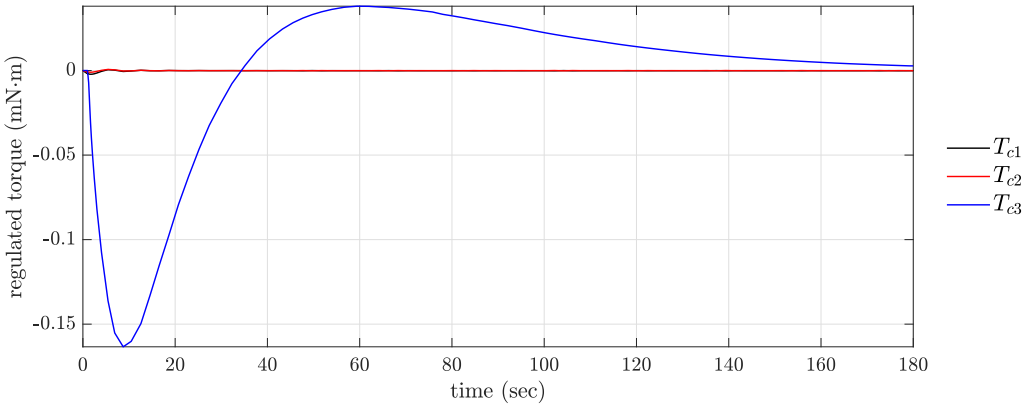


(a) CSACS attitude orientation (dotted green curve is the desired trajectory)

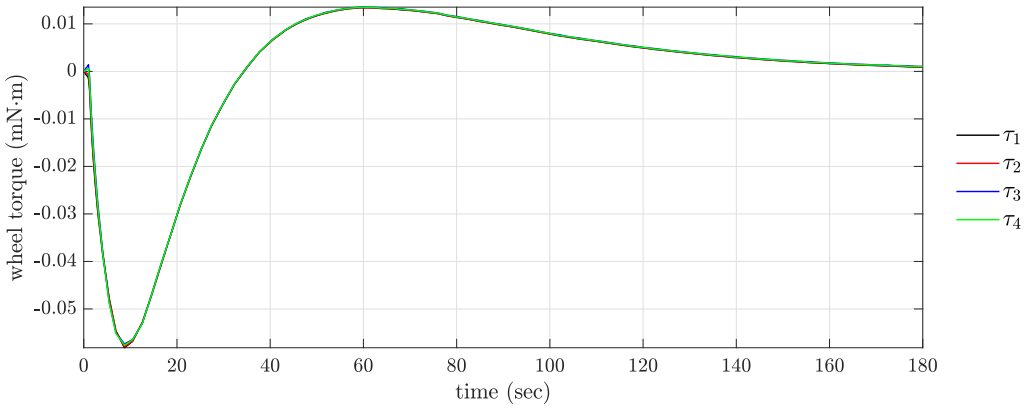


(b) CSACS body rates

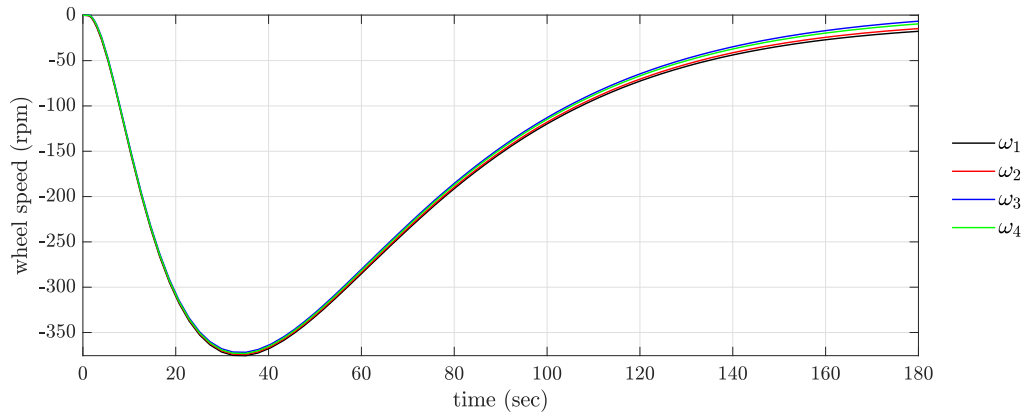
Figure D.1: 2 Minute Settling Time – CSACS Orientation and Body Rates.



(a) RWA regulated body-axis torques

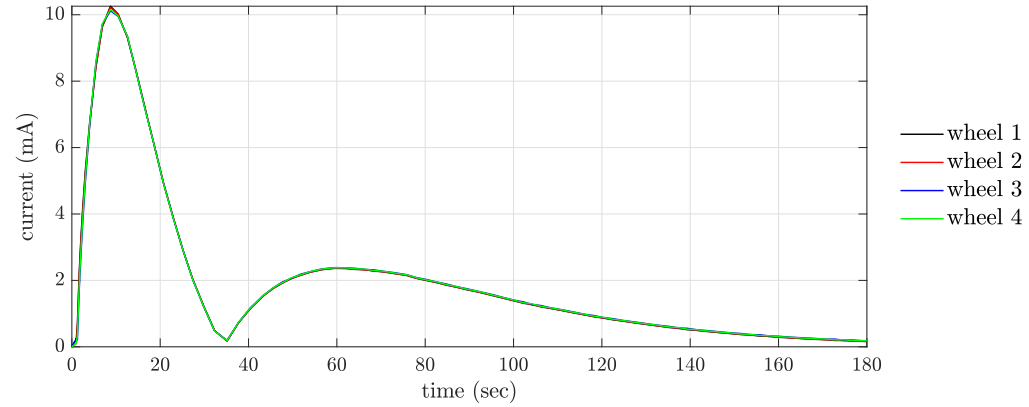


(b) RWA torques

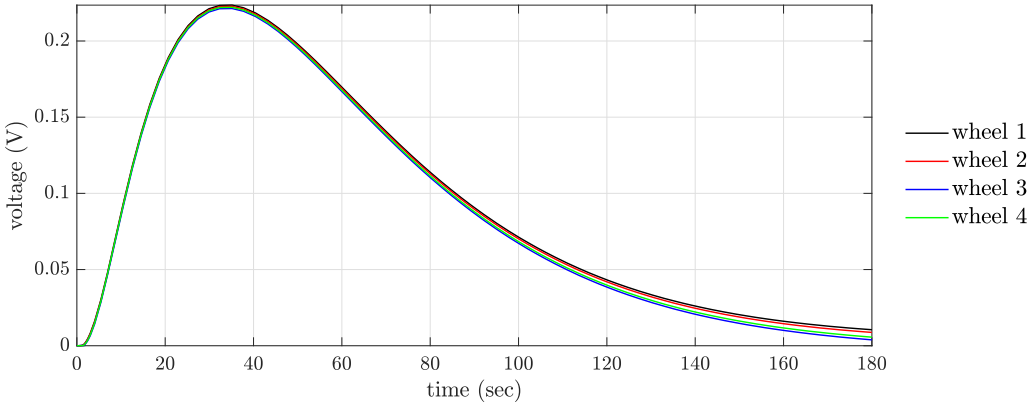


(c) RWA speeds

Figure D.2: 2 Minute Settling Time – RWA Torques and Wheel Speeds



(a) RWA command current

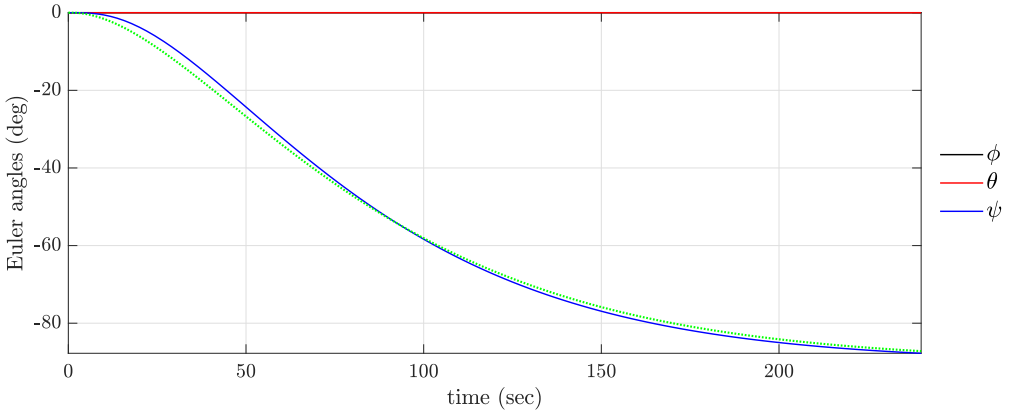


(b) RWA command voltage

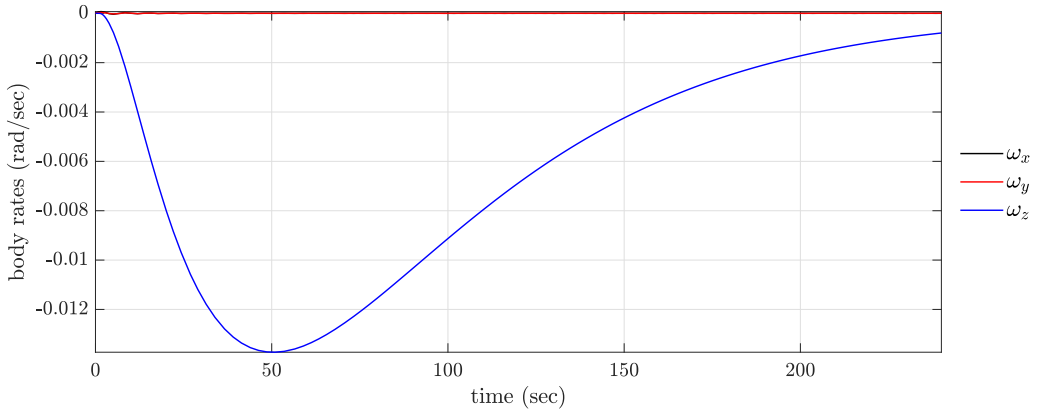
Figure D.3: 2 Minute Settling Time – RWA Commanded Current and Voltage

D.2 3 Minute Settling Time Maneuver

Four functioning wheels, -90° yaw-rotation: Power consumption is 0.03 W.

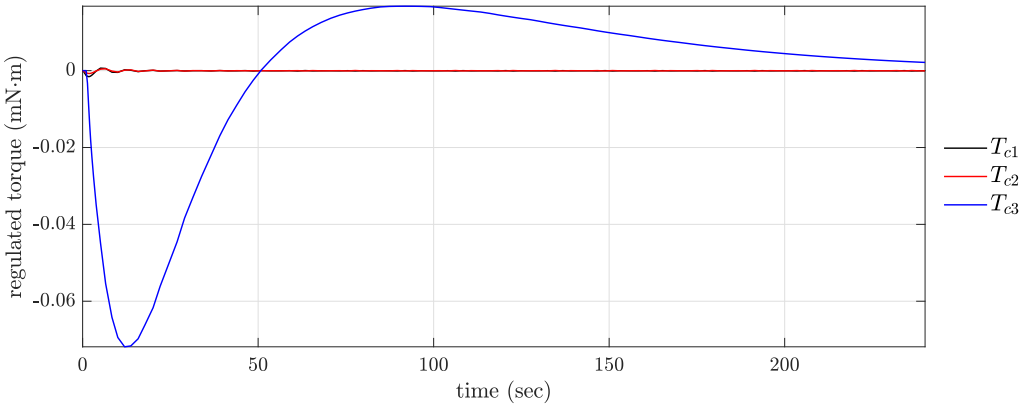


(a) CSACS attitude orientation (dotted green curve is the desired trajectory)

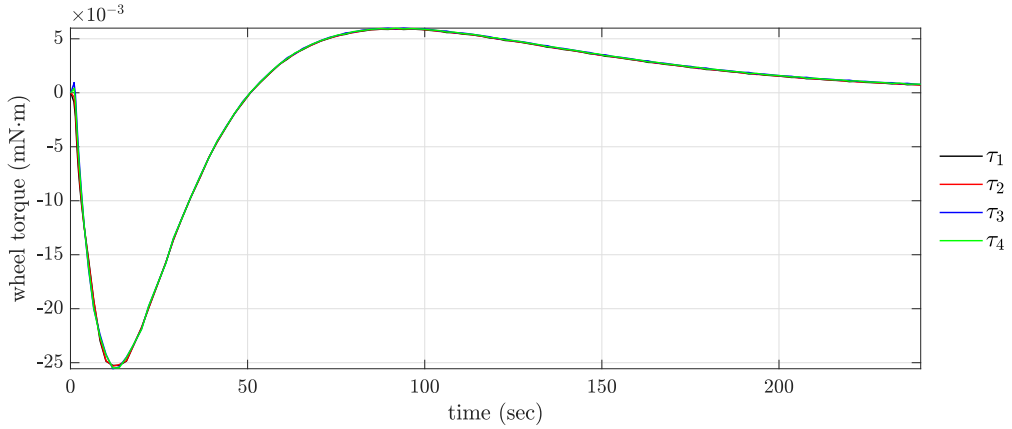


(b) CSACS body rates

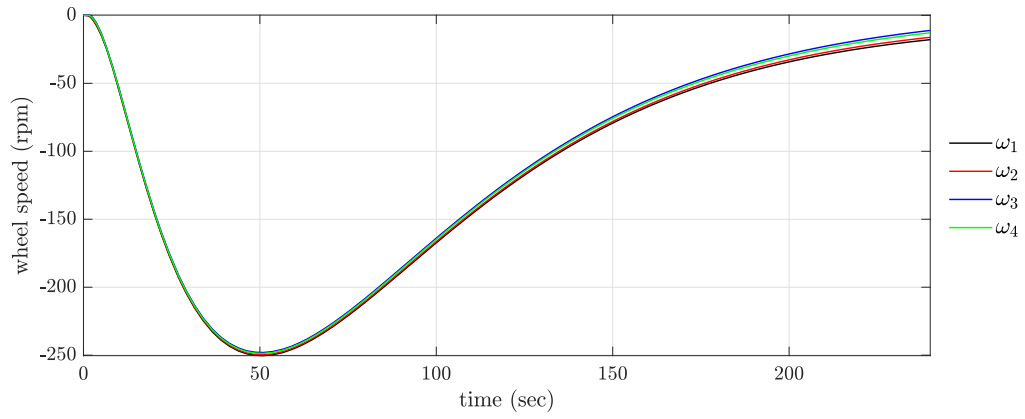
Figure D.4: 3 Minute Settling Time – CSACS Orientation and Body Rates.



(a) RWA regulated body-axis torques

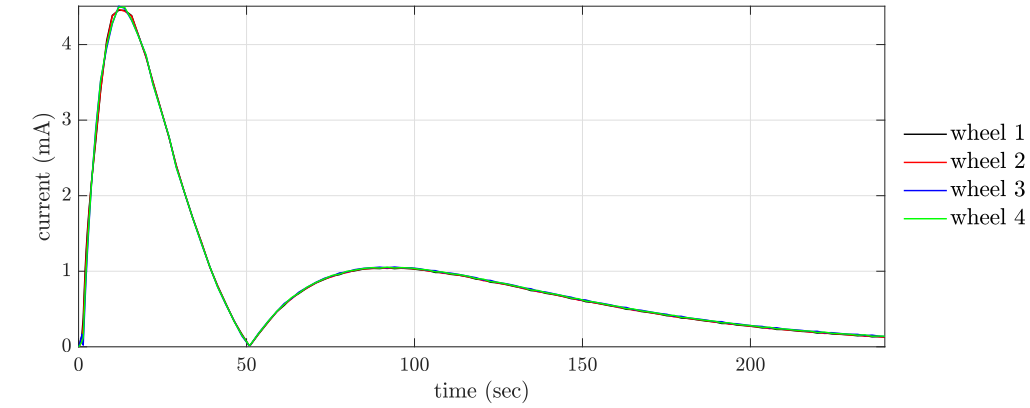


(b) RWA torques

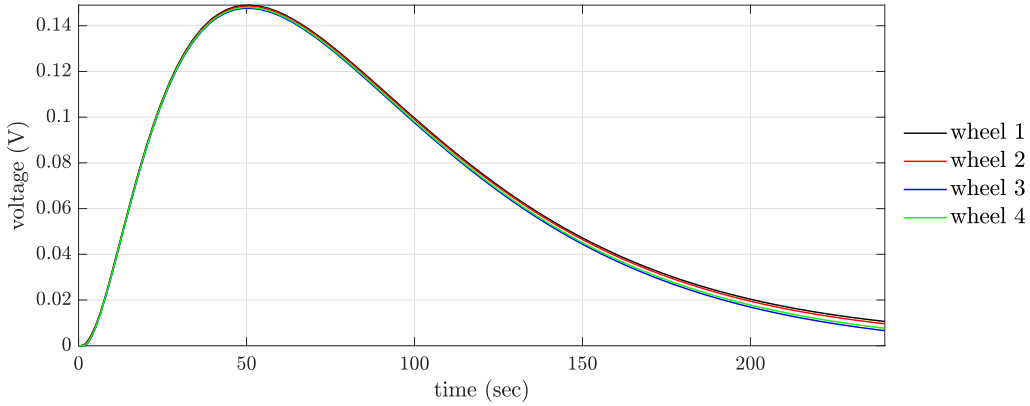


(c) RWA speeds

Figure D.5: 3 Minute Settling Time – RWA Torques and Wheel Speeds



(a) RWA command current



(b) RWA command voltage

Figure D.6: 3 Minute Settling Time – RWA Commanded Current and Voltage

Appendix E

Software Contents

E.1 Balancing Algorithm

`CSACS_Sim.m` – Main program to simulate the nonlinear dynamics of the CSACS, given initial values for the state vector.

`csascseom.m` – Function to calculate the time derivatives of the state vector, using the full equations of motion.

`r_est.m` – Function to estimate the position vector to the CM of the CSACS from the CR of the air bearing, using the dynamic data.

`rm_est.m` – Function to estimate the distance to move the MMUs to remove the CM offset.

`user_inp.m` – Function to allow user input of initial conditions.

E.2 Quaternion Feedback Controller

`RWA_Controller.slx` – Simulink model of the full quaternion feedback (PD) controller for the pyramidal RWA, where the following functions are all embedded in the model.

`Init_Sim.m` – A file to setup the Simulink reaction wheel simulation for nonlinear quaternion feedback control.

`euler2q.m` – Transforms Euler angles to quaternion parameters via a 3-2-1 sequence, where ψ is the first rotation, θ is the second rotation, and ϕ is the third rotation.

`q2euler.m` – Computes the Euler angles from the unit quaternion $\overline{\mathbf{q}} = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}^T$ via a 3-2-1 sequence, where ψ is the first rotation, θ is the second rotation, and ϕ is the third rotation.

`skew.m` – Function that generates the skew-symmetric representation of a 3×1 column vector.

`ucalloc.m` – Unconstrained control allocation function, adapted from the Marine Guidance, Navigation, and Control Toolbox, Copyright© 2008 Thor I. Fossen and Tristan Perez.

`csacs_nonlinear_dyn.m` – A file to calculate the nonlinear dynamic equations of the CSACS, coupled with the RWA.

`Rquat.m` – Sub-function called by `csacs_nonlinear_dyn.m` that generates the 3-2-1 rotation matrix sequence from the unit quaternion elements, where ψ is the first rotation, θ is the second rotation, and ϕ is the third rotation.

`figures.m` – A file to plot the time-histories of Euler angles, body rates, regulated torques, reaction wheel torques, reaction wheel speeds, currents, and voltages (from Section 5.2) – execute after simulation. Additionally, the overall power consumption for the specified maneuver is printed.

E.3 3D Animation

`animate.m` – 3D animation display of the CSACS dynamics simulations. (Note: user must first generate trajectory and motor speed data by either running `CSACS_Sim.m` or `RWA_Controller.slx`)

`CSACS_Geometry.m` – Sub-script called by `animate.m` that generates the CSACS geometry for animation.

`R1.m` – Simple "1" axis rotation

`R2.m` – Simple "2" axis rotation

`R3.m` – Simple "3" axis rotation