UC Berkeley
UC Berkeley Electronic Theses and Dissertations

Title
Trajectory Optimization and Control of Small Spacecraft Constellations

Permalink
https://escholarship.org/uc/item/2829t7dt

Author
Sin, Emmanuel

Publication Date
2021

Peer reviewed|Thesis/dissertation

Trajectory Optimization and Control of Small Spacecraft Constellations

by

Emmanuel Joseph Sin

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Mechanical Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Murat Arcak, Chair
Professor Francesco Borrelli
Professor Kameshwar Poolla
Professor Peter Seiler

Spring 2021

Trajectory Optimization and Control of Small Spacecraft Constellations

Abstract

Trajectory Optimization and Control of Small Spacecraft Constellations

by

Emmanuel Joseph Sin

Doctor of Philosophy in Engineering - Mechanical Engineering

University of California, Berkeley

Professor Murat Arcak, Chair

Large ($>$ 1000kg), expensive ($>$ \$1bn), monolithic, flagship-class satellites (e.g., Hubble, Chandra, James Webb telescopes) may soon be overshadowed by constellations of many, smaller, cheaper, more agile spacecraft working together as a coordinated system. In this work, we study several important attitude and orbit control problems relevant to the formation and operation of small spacecraft constellations. Our example problems include agile attitude maneuvering for multi-target acquisition in an Earth observation application, a centralized approach to constellation formation in low Earth orbit, a distributed approach to constellation formation in Mars areostationary orbit, and orbital rendezvous with objects in the inner Solar System. Our main tool is trajectory optimization, an approach that is appropriate for spacecraft applications where we often seek solutions that are optimal (e.g., minimum-time, minimum-fuel, minimum-control-effort) in the presence of mission or spacecraft constraints. We employ recent advances in sequential convex programming to efficiently handle the nonlinear dynamics and non-convex constraints that exist in our problem formulations.

To Jinyoung, Veronica, and Isabelle.

Thank you for believing in me.

Contents

List of Figures

List of Tables

Acknowledgments

I am blessed to have advisors, seniors, colleagues, friends, and family who have guided and supported me through the Ph.D. program. Thank you for all the wonderful memories.

I first thank Kyo-Woong (Paul) Choo for his mentorship in my professional career before entering graduate school. I have learned so much under your direction but above all, thank you for your active encouragement and support of my decision to pursue Ph.D. studies. I also thank Vice Chairman Woong-Chul Yang for starting me on this path and for my first lesson in dynamical systems and control.

I thank Professor J. Karl Hedrick for his guidance and for inspiring me to study dynamics and control. I will always remember his warm smile and the sincere attention and care he would give whenever I would meet with him. I am very grateful to Professor Francesco Borrelli who has supported me from the beginning, helping me navigate the Ph.D. process, serving on my Qualifying Examination Committee as well as my Dissertation Committee, introducing me to optimization-based control, and for the chats we would have in the hallways of Etcheverry - thank you for always checking up on me and making sure I was on track. Thank you Professor Kameshwar Poolla for not only serving on my Dissertation Committee but also providing me a home at BCCI, ensuring that we had all the resources we needed to conduct research and study. I also thank you for supporting me through those difficult times; I will always be grateful. Professor Peter Seiler - thank you for your guidance over the years and for serving on my Dissertation Committee. You have always asked the hard questions. And although I wasn't always able to respond immediately, I've cherished your feedback as it challenged me to go deeper and led me to new ideas.

It is an honor and privilege to have Professors Andy Packard and Murat Arcak as my research advisors. In research, Andy showed me when it was okay to approximate and when it was important to be rigorous. His constructive criticism was tough (but definitely needed) and his support was subtle yet showed how much he cared. Thank you Andy for challenging me to continuously push the boundaries and to never be content with a result for too long. Thank you for getting me to the finish line. I will always remember your excellence in teaching & research and your dedication to helping others, and I will strive to do the same in my life. Professor Arcak - you have guided and shaped my research direction at all stages of the PhD program, pointing me towards fundamental results and practical methods that have led to this dissertation. I very much appreciate your enthusiasm and support of my research interests - thank you for encouraging me to keep pursuing the topics. Furthermore, I thank you for expanding my academic and professional network, giving me many opportunities to collaborate with others. Over the years, your comments and edits to my papers and presentations have taught me how to be clear, succinct and effective. Thank you for honing these crucial communication skills. Finally, thank you for encouraging me whenever I would get discouraged. You showed me to play on my strengths.

I would also like to thank Professor Oliver O'Reilly for helping me review engineering mechanics in preparation for the Preliminary Examination, Professor David Auslander for advising me on my first satellite control project, Professor Mark Mueller for teaching the

practical aspects of working with actuators and sensors, Professor Dennis Lieu for teaching me about motors, and Professor Laurent El Ghaoui for introducing me to optimization models and applications. I very much appreciate Yawo Akpawu, Isabel Blanco, and ME Student Services for helping me with the GSI, GSR appointments and making sure I fulfill the requirements of the Ph.D. program. Your dedication to the students goes beyond duty.

I am blessed to have awesome friends and colleagues in Octavio Narvaez-Aroche, Arun Hegde, Jared Porter, Kate Schweidel, Akhil Shetty, and He (Galaxy) Yin. Thank you for being such a big part of my life. Although we are each going down separate paths after graduation, I know we will always be in touch.

Thank you BCCI family, including Pratyush Chakraborty, Dariush Fooladivanda, Donya Ghavidel, Emily Jensen, Sen Li, Jonathan Mather, Chris Meissen, Yanfang Mo, Junjie Qin, Hamidreza Tavafoghi, and Maria Vrakopoulou, for your friendship and the memories.

I am also very proud to be a member of the Arcak Lab. I have many thanks to Mikhail Burov, Alex Devonport, Pierre-Jean Meyer, Mindy Perkins, Stan Smith, and Adnane Saoud for the camaraderie and discussions around the coffee machine, which we naturally gravitated towards every morning and afternoon.

I am very grateful to the ONR family, including Douglas Philbrick, Jyot Buch, Shih-Chi Liao, and Neelay Junnarkar. Thank you for expanding my horizons and exposing me to the latest results in control theory. I also thank Henning Ward for our collaborative work on simulation environments.

Thank you Professor Kristofer Pister and the BLISS team, including Alexander Alvara, Nathan Lambert, Lydia Lee, Giuliana Hashemi-Asasi, Beau Kuhn for the opportunity to collaborate with you on solar sails! I've very much enjoyed our weekly discussions about asteroids and orbital mechanics.

I thank members of the VDL/MPC Lab, including Monimoy Bujarbaruah, Yong-Keun (Eric) Choi, Yeo-Jun Kim, Chan-Kyu Lee, and Dong-Han Lee, for their friendship and support. I am also grateful for the coffee breaks with Dong-Gun Lee and Ki-Woo Shin.

I very much appreciate Sreeja Nag, Vinay Ravindra, and Alan Li for the opportunity to study constellations and attitude control during my summer internship with the Bay Area Environmental Research Institute and NASA Ames Research Center.

Thank you Johanna Sedman for opening a place in your heart for us. Your love and warmth is boundless. I am still blown away by the special graduation event. Thank you for the extreme fortune and privilege to share that experience. I will cherish it forever.

Mom - thank you for your daily e-mail messages and for encouraging me when I needed it the most. Thanks Richard for taking care of everything so I could focus on my studies. Dad, I promise - no more school. I look forward to more fishing trips.

Jinyoung - thank you for being by my side every step of the way, through all the ups and downs. I am grateful for your infinite patience and wisdom. Thank you for believing in me. Thank you Veronica for imparting on me your remarkably mature advice, telling me to follow my dreams. Hi Isabelle! =) Thank you for all the joy.

Chapter 1

Introduction

The first artificial spacecraft to orbit the Earth was Sputnik, launched in October, 1957. With a mass of 25 kg, the 1-meter diameter spherical satellite had four long antennas that allowed it to intermittently transmit radio waves, signaling its presence to the world. A few months later, a 14 kg, 2-meter long pencil-shaped Explorer 1 launched into orbit as the first satellite to carry a science instrument - a cosmic ray sensor that led to the discovery of the Van Allen radiation belt. Since then, thousands of spacecraft have been launched and there are currently more than 3,000 operational satellites in Earth orbit today [1]. Employed in various commercial, defense, civil, and science applications with sophisticated sensors and instruments, satellite mass and size have increased over the decades. For example, the James Webb telescope, scheduled to launch in October 2021, has a mass of 6,500 kg and a footprint greater than the size of a tennis court. In civil and government applications, satellites have allowed us to further scientific knowledge (planetary science, astrophysics, heliophysics, Earth science) and have provided invaluable global positioning and navigation services. In the commercial space, satellites have connected the world by providing television and radio broadcasts, telecommunication services and space-based Internet. In military applications, they have allowed for reconnaissance and space-based defense. In general, these exquisite spacecraft require many years of development and hundreds of millions (even billions) of dollars in investment.

In the past decade we have witnessed several trends that are reshaping the spacecraft industry and are enabling exciting new opportunities across sectors and industries:

- proliferation of low-cost launch providers to low Earth orbit

- miniaturization of satellites

- advancement of continuous, low-thrust, electric propulsion technology

- megaconstellations in Low Earth Orbit (LEO)

Within the next decade, it is expected that more than 10,000 small satellites will be launched into orbit, 84 percent of which will be involved in megaconstellations [2]. In traditional mis-

sions involving a single satellite, satellite operators have planned and executed spacecraft attitude and orbital maneuvers "manually" from a mission control center, uplinking maneuver commands via ground stations distributed over the Earth's surface. In the future, small satellite constellation operators must manage hundreds and thousands of spacecraft, presenting interesting technical challenges in autonomy and coordination of multi-body systems.

In this dissertation we study attitude and orbital maneuvers relevant to the operation and maintenance of small spacecraft constellations. Our main tool is trajectory optimization. Using recent advances in sequential convex programming, we solve optimal control problem formulations involving high-fidelity dynamical models and non-convex constraints. In particular, the dissertation makes the following specific contributions:

- Problem formulations that address important maneuvers used in the formation, operation, and maintenance of spacecraft constellations

- The application of recent advances in sequential convex programming (SCP) to the problem formulations. Some new additions or extensions to SCP introduced in this dissertation include:

    1. the ability to use dynamical models where parameters are in the form of look-up tables or black-box functions

    2. multi-body problem formulations where trajectories spanning different time scales may be simultaneously optimized in a single program

    3. the use of forcing functions to represent known but time-dependent processes in the problem formulation

- A 3-dimensional approach to spacecraft constellation trajectory optimization

In **Chapter 2**, we present Background material that is used throughout the dissertation. We first present the dynamical models and equations that we assume govern or describe spacecraft motion. In particular, we derive the equations that describe the orbital dynamics of a spacecraft under the influence of central body gravity, atmospheric drag, oblate Earth gravity, solar radiation pressure, and continuous low-thrust via electric propulsion. We then introduce the models used for attitude dynamics, namely a version of Euler's equations used to describe the rotational motion of a gyrostat (i.e., rigid body with a system of reaction wheels) and the quaternion kinematics, used to represent attitude motion. The rest of the chapter describes a state-of-the-art implementation of sequential convex programming, the trajectory optimization method used to approach most of the problems studied in this work.

Chapter 3 assumes that a constellation is in its desired orbital configuration and is in the operational phase of its mission. As required by remote sensing missions, each satellite may be tasked to perform rapid acquisition, precise pointing or tracking of targets. Agile attitude maneuvering maximizes the utility of such remote sensing satellite constellations. We explicitly consider the physical properties of a reference spacecraft and its actuator characteristics to conduct agile remote sensing maneuvers beyond conventional slew-and-stabilize

maneuvers. We present two problem formulations. The Minimum-Time Slew Optimal Control Problem determines the minimum slew time, required energy, and optimal trajectory for a reference satellite to slew between any two orientations while satisfying state and input constraints. Given a desired attitude pointing schedule (i.e., sequence of attitude states), the Minimum-Effort, Multi-Target Pointing Optimal Control Problem produces a continuous attitude state/input trajectory that achieves the given schedule while minimizing control effort and satisfying constraints. In the presence of external disturbances or model mismatch, a low-level tracking controller can be used to mitigate deviation from the desired trajectory. We demonstrate our approach with an example of a reference satellite in Sun-synchronous orbit passing over globally-distributed, Earth-observation targets.

As a stepping stone to the multi-body problems of constellations, we first study continuous low-thrust orbital maneuvers for a single spacecraft in **Chapter 4**. The initial set of problem formulations focus on changing the size and shape of an orbit (e.g., orbit raising, orbit circularization). The next set of problems address the ability to change the orientation of orbits with respect to an inertial reference frame (e.g., inclination change, ascending node change, argument of periapsis change). For these problems we assume continuous low-thrust is achieved by an electric propulsion system. For the last problem of the chapter, we change our reference frame to investigate optimal rendezvous trajectories to objects in the Solar System that leverage solar radiation pressure in an approach called "solar sailing."

Constellation formation refers to a class of problems where a group of spacecraft must be maneuvered to achieve a desired spatial configuration. Constellations may be designed with certain relative position requirements; for example, satellites occupying the same orbital plane should maintain equal angular spacing. Hence, the spacecraft maneuvers must be coordinated. **Chapter 5** introduces a centralized approach to constellation formation, where we assume that a central entity (e.g., mission control at the constellation operator) has accurate state information from all satellites and can uplink optimized maneuver commands to all satellites of a constellation. This assumption is valid for constellations operating in low Earth orbit where satellites have low-latency communication with a network of ground stations. We introduce the optimal orbit phasing problem where the objective is to form an equally-spaced constellation in minimum-time.

Chapter 6 considers the case where a centralized approach to constellation formation is not possible. For example, due to the lack of ground stations or a communication infrastructure on Mars, it would be difficult to implement the centralized approach demonstrated in Chapter 3. Instead, we design a distributed control law that each satellite can implement, requiring only relative state information of neighboring satellites. The stability of the constellation about its equilibrium point under the proposed control law is proved based on the interconnection structure of the constellation and the passivity properties of the subsystems.

Chapter 2

Background

2.1 Notation, Nomenclature, and Conventions

A scalar is denoted by an unbolded, lowercase letter, e.g., $x \in \mathbb{R}$ is a real scalar-valued variable. Unless otherwise specified, a vector is denoted by a bold, lowercase letter; e.g., $\mathbf{x} \in \mathbb{R}^n$ is a real vector-valued variable of size n. An unbolded lowercase letter with a subscript i (e.g., x_i) refers to the ith entry of the vector $\mathbf{x}$. A bold lowercase letter with subscript i refers to the ith instance of a vector; e.g., $\mathbf{x}_i$ may refer to the state vector for the ith satellite of a constellation. Unless specifically noted, all vectors are column vectors $\mathbf{x} \in \mathbb{R}^n \iff \mathbf{x} \in \mathbb{R}^{n \times 1}$. A matrix is denoted by an unbolded, uppercase letter, e.g., $A \in \mathbb{R}^{m \times n}$ is a real matrix-valued variable with m rows and n columns.

Nomenclature and Conventions

a	semi-major axis, [AU]
p	semi-latus rectum, [AU]
$\mathbf{a_G}$	acceleration due to central body gravity, [m/s^2]
$\mathbf{a_D}$	acceleration due to atmospheric drag, [m/s^2]
$\mathbf{a_{J2}}$	acceleration due to J2 (oblate Earth) gravity, [m/s^2]
$\mathbf{a_T}$	acceleration due to thrust, [m/s^2]
$\mathbf{a_S}$	acceleration due to solar radiation pressure, [m/s^2]
AU	astronomical unit (mean distance from center of Earth to center of Sun), [m]
A_S	reference surface area for solar radiation pressure, [m^2]
A_r	actuator Jacobian, columns are rotor axes of rotation w.r.t. body-fixed frame
C_D	atmospheric drag coefficient, [ ]
α	solar sail cone angle, [rad]
β	solar sail clock angle, [rad]
ϵ	specific orbital energy [m^2/s^2]
$\epsilon_\mathbf{r}$	tolerance value associated with $\mathbf{r}$

$\mathbf{e}$	eccentricity vector, [ ]
e	eccentricity, [ ]
g	standard gravitational acceleration at surface of the Earth, [m/s^2]
μ	standard gravitational parameter of central body, [m^3/s^2]
i	inclination of orbit
$\mathbf{I}^n$	identity matrix of size n
I_{sp}	specific impulse of propulsion system, [s]
J_2	second zonal harmonic coefficient of Earth gravitational field, [ ]
J	mass moment of inertia matrix of rigid body in body-fixed frame, [kg· m^2]
J_r	mass moment of inertia of momentum rotor about axis of rotation, [kg· m^2]
J_{obj}	objective function in an optimization problem formulation
J_{tr}	thrust region penalty in objective function of an optimization problem formulation
J_{vc}	virtual control penalty in objective function of an optimization problem formulation
K	number of discretization nodes
l	true longitude, [rad]
p	semi-latus rectum, [AU]
P	magnitude of solar radiation pressure, [N/m^2]
S	reference surface area for atmospheric drag, [m^2]
m	mass, [kg]
N	number of spacecraft
n_x	number of states per spacecraft
n_u	number of inputs per spacecraft
r_E	mean radius of Earth, [m]
ρ	atmospheric density, [kg/m^3]
$\boldsymbol{\rho_r}$	rotor angular momentum vector, [Nms]
$\mathbf{r}$	position vector, [m]
$\mathbf{v}$	velocity vector, [m/s]
ν	true anomaly, [rad]
$\boldsymbol{\nu}$	virtual controls in an optimization problem formulation
$\mathbf{h}$	specific angular momentum vector of an orbit, [m^2/s]
$\mathbf{h}_r$	rigid body (or gyrostat) angular momentum vector, [Nms]
$\mathbf{q}$	unit quaternion, attitude of body-fixed frame w.r.t. inertial frame, [ ]
$\boldsymbol{\omega}$	angular velocity of rigid body about body-fixed frame axes, [rad/s]
ω	argument of periapsis, [rad]
Ω	right ascension of the ascending node, [rad]
w	weighting term in objective function of an optimization problem formulation
v_R	radial orbital speed, [m/s]
v_T	tangential orbital speed, [m/s]
v_N	normal orbital speed, [m/s]
v_C	circular orbital speed, [m/s]
t	time, [s]
t_0	initial time, [s]

t_f	final time, [s]
τ	normalized time, [ ]
$\boldsymbol{\tau_r}$	rotor torque vector, [Nm]
$\mathbf{T}$	thrust vector, [N]
$[\mathbf{h}]^{\mathrm{B}}$	vector $\mathbf{h}$ coordinated in body frame
$^{I}R^{B}$	rotation matrix from body frame to inertial frame, such that $[\mathbf{h}]^{\mathrm{I}} = {}^{I}R^{B}[\mathbf{h}]^{\mathrm{B}}$
$\mathbf{0}^{m \times n}$	zero matrix of size $m \times n$
$[\ ]^{\times}$	skew symmetric operator such that $\mathbf{a} \times \mathbf{b} = [\mathbf{a}]^{\times}\mathbf{b}$
$\omega^{\times}$	skew symmetric matrix corresponding to vector ω
$\Phi^{\dagger}$	pseudo-inverse of matrix Φ
$\mathbf{x}$	vector of state variables
$\mathbf{x}(t)$	vector of state variables evaluated at time t
$\mathbf{u}$	vector of input variables
u	argument of latitude, [rad]
$\dot{\mathbf{x}}$	derivative of state vector w.r.t. time, i.e., $\dot{\mathbf{x}} := \frac{d}{dt}\mathbf{x}$
$\mathbf{x}'$	derivative of state vector w.r.t. normalized time, i.e., $\dot{\mathbf{x}} := \frac{d}{d\tau}\mathbf{x}$
$\|\mathbf{x}\|$	Euclidean norm of $\mathbf{x}$, i.e., $\|\mathbf{x}\| := \|\mathbf{x}\|_2$
$\hat{\mathbf{x}}$	unit vector corresponding to $\mathbf{x}$, i.e., $\hat{\mathbf{x}} := \frac{\mathbf{x}}{\|\mathbf{x}\|}$
$\hat{\mathbf{r}}$	unit vector aligned with orbital radial direction
$\hat{\mathbf{t}}$	unit vector aligned with orbital tangential direction
$\hat{\mathbf{h}}$	unit vector aligned with orbital normal direction
$\hat{\mathbf{n}}$	unit vector normal to solar sail surface
$\hat{\mathbf{n}}_{AN}$	unit vector aligned with node vector pointing toward orbit ascending node
$\nabla_{\mathbf{x}}f$	gradient of scalar-valued function f w.r.t. $\mathbf{x}$
$\mathbf{f}(\cdot)$	vector-valued function representing dynamical equations of motion
$\mathbf{g}(\cdot)$	vector-valued function representing constraints
$D_{\mathbf{x}}\mathbf{f}(\bar{\mathbf{x}})$	partial derivative of vector-valued function $\mathbf{f}$ w.r.t. $\mathbf{x}$, evaluated at $\bar{\mathbf{x}}$,
	i.e., $D_{\mathbf{x}}\mathbf{f}(\bar{\mathbf{x}}) := \frac{\partial}{\partial \mathbf{x}}\mathbf{f}(\mathbf{x})\Big|_{\mathbf{x}=\bar{\mathbf{x}}}$

2.2 Dynamical Models

For simplicity, we study the translational and rotational motion of a spacecraft separately. The three degree-of-freedom (3-DOF) translational motion of the spacecraft is introduced in the following subsection on Orbital Dynamics. We then discuss the 3-DOF rotational motion in the subsequent subsection on Attitude Dynamics. When the models are considered together, we can represent the full 6-DOF motion of a spacecraft.

Orbital Dynamics

We first introduce the equations of Keplerian motion, that is, motion of a body influenced only by a central gravitational force. It can be shown that the motion of a spacecraft in a Keplerian orbit remains in a 2-dimensional plane, fixed in 3-dimensional space with an inertial frame of reference. The spacecraft's orbital trajectory takes on the shape of a conic section (i.e., circle, ellipse, parabola, hyperbola). We then introduce the most significant perturbations present in low Earth orbit, namely, gravitational acceleration due to the oblateness of the Earth (J2 gravity), acceleration due to atmospheric drag, and acceleration due to thrust from an onboard propulsion system. Finally, we introduce orbital parameters and properties that can be computed from the position and velocity states of the spacecraft.

Keplerian Motion

Under the assumptions for central force motion, namely:

- only force is due to gravity between a central body and an orbiting body,

- central body is spherical with uniformly distributed mass,

- central body's mass is significantly large than the orbiting body's mass

the equations of motion for the orbiting body are:

$$\dot{\mathbf{r}}(t) = \mathbf{v}(t) \tag{2.1}$$
$$\dot{\mathbf{v}}(t) = \mathbf{a_G}(\mu; \mathbf{r}(t)) \tag{2.2}$$

where the acceleration due to central body gravity is:

$$\mathbf{a_G}(\mu; \mathbf{r}(t)) = -\frac{\mu}{\|\mathbf{r}(t)\|^3}\mathbf{r}(t) \tag{2.3}$$

The position vector is $\mathbf{r}$ and the velocity vector is $\mathbf{v}$. The constant parameter μ is the standard gravitational parameter of the central body being considered (e.g., Earth, the Sun). The position and velocity vectors are coordinated in an inertial reference frame placed at the center of the central body with its "z-axis" aligned with rotation axis of the central body. When the central body is the Earth, the inertial reference frame is called the "Earth-Centered-Inertial" (ECI) frame, and when the central body is the Sun, it is called the "Sun-Centered-Inertial" (SCI) frame. For simplicity, we assume that the states describing orbital motion are coordinated in an inertial frame and drop the bracket notation, i.e., $\mathbf{r} = [\mathbf{r}]^{\text{ECI}}$.

Parameters of orbital motion

Given a spacecraft's position and velocity state vectors, expressed in a Cartesian coordinate system about an inertial reference frame, we may compute certain useful orbital parameters

and state representations. The specific angular momentum vector is defined as:

$$\mathbf{h} = \mathbf{h}(\mathbf{r}, \mathbf{v}) = \mathbf{r} \times \mathbf{v} \tag{2.4}$$

Unit vectors pointing in the radial and velocity directions are:

$$\hat{\mathbf{r}} = \hat{\mathbf{r}}(\mathbf{r}) = \frac{\mathbf{r}}{\|\mathbf{r}\|} \tag{2.5}$$

$$\hat{\mathbf{v}} = \hat{\mathbf{v}}(\mathbf{v}) = \frac{\mathbf{v}}{\|\mathbf{v}\|} \tag{2.6}$$

The unit vector pointing in the direction of the angular momentum vector (also called the orbit normal unit vector) is:

$$\hat{\mathbf{h}} = \hat{\mathbf{h}}(\mathbf{h}) = \frac{\mathbf{h}}{\|\mathbf{h}\|} = \frac{\mathbf{r} \times \mathbf{v}}{\|\mathbf{r} \times \mathbf{v}\|} = \hat{\mathbf{h}}(\mathbf{r}, \mathbf{v}) \tag{2.7}$$

The unit vector pointing in the orbital tangential direction is:

$$\hat{\mathbf{t}} = \hat{\mathbf{t}}(\hat{\mathbf{r}}, \hat{\mathbf{h}}) = \hat{\mathbf{h}} \times \hat{\mathbf{r}} = \frac{\mathbf{r} \times \mathbf{v}}{\|\mathbf{r} \times \mathbf{v}\|} \times \frac{\mathbf{r}}{\|\mathbf{r}\|} = \hat{\mathbf{t}}(\mathbf{r}, \mathbf{v}) \tag{2.8}$$

We may then compute the components of the velocity vector in the orbit radial, tangential, and normal directions:

$$v_R(\mathbf{r}, \mathbf{v}) = \mathbf{v}^\top \hat{\mathbf{r}}(\mathbf{r}) \tag{2.9}$$

$$v_T(\mathbf{r}, \mathbf{v}) = \mathbf{v}^\top \hat{\mathbf{t}}(\mathbf{r}, \mathbf{v}) \tag{2.10}$$

$$v_N(\mathbf{r}, \mathbf{v}) = \mathbf{v}^\top \hat{\mathbf{h}}(\mathbf{r}, \mathbf{v}) \tag{2.11}$$

In general, we may coordinate a vector between the inertial reference frame (located at the center of the central body) and the rotating "RTN" frame (located at the center of the orbiting spacecraft body) with the appropriate time-varying rotation matrices:

$$[\mathbf{x}(t)]^{\text{ECI}} = \left[\hat{\mathbf{r}}(\mathbf{r}(t)) \quad \hat{\mathbf{t}}(\mathbf{r}(t), \mathbf{v}(t)) \quad \hat{\mathbf{h}}(\mathbf{r}(t), \mathbf{v}(t)) \right] [\mathbf{x}(t)]^{\text{RTN}} \tag{2.12}$$

$$[\mathbf{x}(t)]^{\text{RTN}} = \begin{bmatrix} \hat{\mathbf{r}}(\mathbf{r}(t))^\top \\ \hat{\mathbf{t}}(\mathbf{r}(t), \mathbf{v}(t))^\top \\ \hat{\mathbf{h}}(\mathbf{r}(t), \mathbf{v}(t))^\top \end{bmatrix} [\mathbf{x}(t)]^{\text{ECI}} \tag{2.13}$$

At any given altitude, the tangential component of velocity (i.e., speed) required for a circular Keplerian orbit is:

$$v_C(\mu; \mathbf{r}) = \sqrt{\frac{\mu}{\|\mathbf{r}\|}} \tag{2.14}$$

It is also necessary for the radial and normal speeds to equal zero for a circular orbit. The angular speed for a circular orbit is:

$$\omega_C(\mu; \mathbf{r}) = \sqrt{\frac{\mu}{\|\mathbf{r}\|^3}} \tag{2.15}$$

The specific orbital energy (or vis-viva energy) is the sum of the specific potential energy and the total specific kinetic energy. Under the assumption of Keplerian motion (i.e., no perturbations), specific orbital energy is conserved. For elliptical orbits, it can be shown that the specific orbital energy is equal to the following expression involving the orbit's semi-major axis, a:

$$\epsilon = \frac{\|\mathbf{v}\|^2}{2} - \frac{\mu}{\|\mathbf{r}\|} = -\frac{\mu}{2a} \tag{2.16}$$

Using the vis-viva equation from above (also known as the orbital energy invariance law), we have the following expression for the orbit's semi-major axis:

$$a = a(\mathbf{r}, \mathbf{v}) = \left(\frac{2}{\|\mathbf{r}\|} - \frac{\|\mathbf{v}\|^2}{\mu} \right)^{-1} \tag{2.17}$$

The orbital period for a Keplerian (circular or elliptical) orbit is:

$$T_p(\mu, a) = 2\pi \sqrt{\frac{a^3}{\mu}} \tag{2.18}$$

The eccentricity vector points from the center of the central body towards the nearest point of approach of the orbiting spacecraft (i.e., the periapsis):

$$\mathbf{e} = [e_x \ e_y \ e_z]^\top = \mathbf{e}(\mathbf{r}, \mathbf{v}) = \frac{\mathbf{v} \times \mathbf{h}(\mathbf{r}, \mathbf{v})}{\mu} - \hat{\mathbf{r}}(\mathbf{r}) \tag{2.19}$$

The magnitude of the eccentricity vector is called the eccentricity of the orbit:

$$e = e(\mathbf{e}) = \|\mathbf{e}\| \tag{2.20}$$

A value of $e = 0$ corresponds to a circular orbit, $0 < e < 1$ for elliptical orbits, $e = 1$ for a parabolic trajectory, and $e > 1$ for hyperbolic trajectories. The unit vector pointing in the direction of the eccentricity vector is:

$$\hat{\mathbf{e}} = \hat{\mathbf{e}}(\mathbf{e}) = \frac{\mathbf{e}}{\|\mathbf{e}\|} \tag{2.21}$$

The eccentricity of the orbit may also be computed with the semi-major axis a and semi-latus rectum p:

$$e = e(\mathbf{r}, \mathbf{v}) = \sqrt{1 - \frac{p}{a}} \qquad \text{where} \quad p = p(\mathbf{r}, \mathbf{v}) = \frac{\|\mathbf{h}(\mathbf{r}, \mathbf{v})\|^2}{\mu} \tag{2.22}$$

We note that the specific orbital energy, specific angular momentum vector, eccentricity vector, semi-major axis, and semi-latus rectum remain constant for Kepler orbits but vary when perturbations are considered.

In addition to parameters that describe the size and shape of an orbit, we introduce parameters that describe the orientation of the orbital plane with respect to the central body-centered inertial frame. The inclination of an orbit with respect to the central body's equatorial (xy) plane is:

$$i = i(\mathbf{r}, \mathbf{v}) = \arccos\left(\frac{h_z}{\|\mathbf{h}(\mathbf{r}, \mathbf{v})\|}\right) \tag{2.23}$$

where h_z is the third element of the specific angular momentum vector.

For inclined orbits, the node vector pointing towards the ascending node of an orbit (i.e., the point on an orbit where a spacecraft crosses the central body's equatorial plane in a northerly direction) is defined as:

$$\mathbf{n}_A = [n_{A_x}\ n_{A_y}\ n_{A_z}]^\top = \mathbf{n}_A(\mathbf{r}, \mathbf{v}) = \hat{\mathbf{k}} \times \mathbf{h}(\mathbf{r}, \mathbf{v}) = [-h_y\ h_x\ 0\ ]^\top \tag{2.24}$$

where $\hat{\mathbf{k}} := [0\ 0\ 1]^\top$ is the positive "z-direction" of the inertial reference frame about which the central body rotates. The unit vector pointing in the direction of the node vector is:

$$\hat{\mathbf{n}}_A = \hat{\mathbf{n}}_A\left(\mathbf{n}_A(\mathbf{r}, \mathbf{v})\right) = \frac{\mathbf{n}_A(\mathbf{r}, \mathbf{v})}{\|\mathbf{n}_A(\mathbf{r}, \mathbf{v})\|} \tag{2.25}$$

The angle between some reference direction, that we call the *origin of longitude*, and the node vector is defined as the longitude of the ascending node. As a convention, we set the positive "x-direction" of the inertial reference frame to align with the origin of longitude. For geocentric orbits, where the Earth's equatorial plane serves as the reference plane and the First Point of Aries is designated as the origin of longitude, this angle is also called the right ascension of the ascending node (RAAN), measured in the counterclockwise direction (as seen from north) from the origin of longitude:

$$\Omega = \Omega\left(\mathbf{r}, \mathbf{v}\right) = \begin{cases} \arccos\left(\frac{n_{A_x}}{\|\mathbf{n}_A\|}\right), & n_{A_y} \geq 0 \\ 2\pi - \arccos\left(\frac{n_{A_x}}{\|\mathbf{n}_A\|}\right), & n_{A_y} < 0 \end{cases} \tag{2.26}$$

In the case of non-inclined (i.e. equatorial) orbits, the RAAN is undefined. As a convention, we set $\Omega = 0$ and the "ascending node" is aligned with the reference direction: $\mathbf{n}_A = [\,1\,0\,0\,]^\top$.

For inclined, elliptical orbits, the argument of periapsis is defined as the angle between the node vector and the eccentricity vector, measured in the counterclockwise direction from the node vector:

$$\omega = \omega\left(\mathbf{r}, \mathbf{v}\right) = \begin{cases} \arccos\left(\hat{\mathbf{n}}_A \cdot \hat{\mathbf{e}}\right), & e_z \geq 0 \\ 2\pi - \arccos\left(\hat{\mathbf{n}}_A \cdot \hat{\mathbf{e}}\right), & e_z < 0 \end{cases} \tag{2.27}$$

In the case of equatorial orbits (i.e., zero inclination and RAAN set to zero), the argument of periapsis is defined as:

$$\omega = \begin{cases} \arctan 2\left(e_y, e_x\right), & h_z \geq 0 \quad \text{(counterclockwise orbit)} \\ 2\pi - \arctan 2\left(e_y, e_x\right), & h_z < 0 \quad \text{(clockwise orbit)} \end{cases} \tag{2.28}$$

In the case of circular orbits, the argument of periapsis is undefined. As a convention, we set $\omega = 0$ so that the "periapsis" is aligned with the node vector: $\mathbf{e} = \mathbf{n}_A$.

For inclined, elliptical orbits, the true anomaly is defined as the angle between the eccentricity vector and the radial position vector, measured in the counterclockwise direction from the eccentricity vector:

$$\nu = \nu\left(\mathbf{r}, \mathbf{v}\right) = \begin{cases} \arccos\left(\hat{\mathbf{e}} \cdot \hat{\mathbf{r}}\right), & \left(\mathbf{r} \cdot \mathbf{v}\right) \geq 0 \\ 2\pi - \arccos\left(\hat{\mathbf{e}} \cdot \hat{\mathbf{r}}\right), & \left(\mathbf{r} \cdot \mathbf{v}\right) < 0 \end{cases} \tag{2.29}$$

For circular orbits, the true anomaly is undefined since the periapsis is not uniquely determined. Instead, the argument of latitude may be defined as the angle between the node vector and the position vector:

$$u = u\left(\mathbf{r}, \mathbf{v}\right) = \begin{cases} \arccos\left(\hat{\mathbf{n}}_A \cdot \hat{\mathbf{r}}\right), & r_z \geq 0 \\ 2\pi - \arccos\left(\hat{\mathbf{n}}_A \cdot \hat{\mathbf{r}}\right), & r_z < 0 \end{cases} \tag{2.30}$$

where r_z is the z-component of the position vector. For circular orbits with zero inclination, both the periapsis and the line of nodes are not uniquely determined. Instead, we may use the true longitude, defined as the angle between the origin of longitude and the position vector:

$$l = l\left(\mathbf{r}, \mathbf{v}\right) = \begin{cases} \arccos\left(\frac{r_x}{\|\mathbf{r}\|}\right), & v_x \geq 0 \\ 2\pi - \arccos\left(\frac{r_x}{\|\mathbf{r}\|}\right), & v_x < 0 \end{cases} \tag{2.31}$$

where r_x and v_x are the x-components of the position and velocity vectors, repectively.

Perturbations

In terms of magnitude, the gravitational force caused by an oblate Earth is the most significant perturbation to a spacecraft in low Earth orbit. The gravitational potential around an oblate Earth may be modeled with an infinite sum of spherical harmonics, functions defined on the surface of a sphere. The most significant term in the infinite sum is called the second zonal term with a corresponding second zonal harmonic coefficient ("J_2"). This J_2 term is over 1000 times larger than the rest of the sum, hence we approximate the oblate Earth gravitational perturbation with this single term. The acceleration due to J_2 gravity may be

modeled with the following expression :

$$\mathbf{a_{J2}}(\mu, J_2, r_E; \mathbf{r}(t)) = 1.5 J_2 \mu \frac{r_E^2}{\|\mathbf{r}(t)\|^5} \begin{bmatrix} 5\left(\frac{r_z(t)}{\|\mathbf{r}(t)\|}\right)^2 - 1 & 0 & 0 \\ 0 & 5\left(\frac{r_z(t)}{\|\mathbf{r}(t)\|}\right)^2 - 1 & 0 \\ 0 & 0 & 5\left(\frac{r_z(t)}{\|\mathbf{r}(t)\|}\right)^2 - 3 \end{bmatrix} \mathbf{r}(t)$$

(2.32)

where J_2 is the second zonal harmonic constant, r_E is the mean radius of the Earth and r_z is the third element of the position vector $\mathbf{r}$.

To illustrate the effect of J_2 gravity, we simulate the orbit of a highly inclined orbit with and without the J_2 perturbation. In Fig. 2.1, the first plot shows that the spacecraft's orbital plane remains fixed in inertial space when under the influence of central gravity alone. However, the second plot shows that the orbital plane does not remain fixed when J_2 gravity is included in the model. The normal to the orbital plane precesses about the rotation axis of the central body.
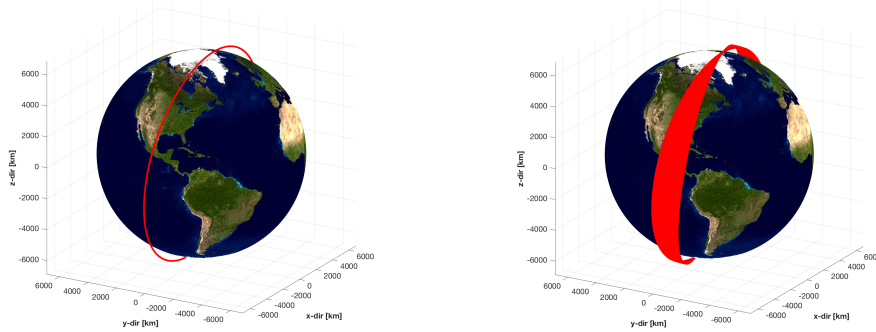
Figure 2.1: Effect of J2 perturbation on inclined low Earth orbit

Another significant perturbation in low Earth orbit is the acceleration due to atmospheric drag, which may be modeled with the following expression:

$$\mathbf{a_D}(C_D, \rho(\cdot); \mathbf{r}(t), \mathbf{v}(t), m(t), S(t)) = -\frac{1}{2}\frac{\rho(\mathbf{r}(t))}{m(t)} C_D S(t) \|\mathbf{v}(t)\| \mathbf{v}(t)$$

(2.33)

Atmospheric density, ρ, can be modeled as a function of the spacecraft altitude. Commonly used models include Harris-Priester [3] and the Jacchia-Roberts [4] models. When the spacecraft employs a propulsion system, we must consider a varying mass m. For a spacecraft orbiting at a certain altitude in the atmosphere, the drag coefficient C_D is a function of the Reynolds number, the Mach number, and the direction of the relative flow. It is common practice to assume a fitted or even constant drag coefficient (e.g. $C_D = 2.2$) for spacecraft with compact shapes [5]. The spacecraft's reference drag area S, that is exposed to the

incident atmosphere as it moves through the fluid, may also be assumed to be a constant value or a function of the spacecraft's attitude.

In addition to natural, environmental perturbations such as atmospheric drag and oblate Earth gravity, the spacecraft may generate its own perturbations through actuators, such as a propulsion system that produces thrust. The acceleration due to continuous thrust can be modeled as:

$$\mathbf{a_T}(m(t), \mathbf{T}(t)) = \frac{\mathbf{T}(t)}{m(t)} \tag{2.34}$$

where we assume that we may apply a thrust vector $\mathbf{T}$ in any direction but with a bounded magnitude. Since thrust is produced by accelerating and ejecting a reaction mass (i.e., propellant), we model the vehicle's total mass as a state variable governed by the following equation:

$$\dot{m}(t) = -\frac{\|\mathbf{T}(t)\|}{g_0 I_{sp}} \tag{2.35}$$

where g_0 is the standard acceleration due to gravity in a vacuum near the surface of the Earth and I_{sp} is the specific impulse, measured in seconds, of the propulsion system (engine and propellant). Specific impulse is a measure of how efficiently a propulsion system produces thrust (i.e., how many seconds an engine can accelerate a unit mass of propellant at a continuous value of standard gravity until the given mass is depleted). In general, chemical propulsion systems may produce large amounts of thrust (thousands of Newtons) with specific impulse values up to 500 seconds, whereas current electric propulsion systems produce thrust on the order of fractions of a Newton but with specific impulse values on the order of thousands of seconds.

Equations of Translational Motion in Low Earth Orbit

Considering the dominant perturbations described above, the equations of orbital motion in low Earth orbit are:

$$\dot{\mathbf{r}}(t) = \mathbf{v}(t) \tag{2.36}$$
$$\dot{\mathbf{v}}(t) = \mathbf{a_G}(\mathbf{r}(t)) + \mathbf{a_{J2}}(\mathbf{r}(t)) + \mathbf{a_D}(\mathbf{r}(t), \mathbf{v}(t), m(t)) + \mathbf{a_T}(m(t), \mathbf{T}(t)) \tag{2.37}$$
$$\dot{m}(t) = -\frac{\|\mathbf{T}(t)\|}{g_0 I_{sp}} \tag{2.38}$$

Other perturbations that may be included in the model are solar radiation pressure (i.e., force caused by solar radiation being absorbed and reflected on the spacecraft's surfaces) and gravitational accelerations caused by "third bodies" (e.g., Moon, Sun, other planets). These perturbations should be considered for spacecraft trajectories that go beyond low Earth orbit.

Attitude Dynamics

Attitude dynamics describes how applied torques affect the rotational motion and orientation of a body with respect to an inertial reference frame. The rotational motion is governed by rigid body dynamics and the corresponding change in attitude or orientation is described by attitude kinematics. In this section we derive an extension of the Euler's equations for gyrostats, a system of rigid bodies whose relative motion does not change the total system moment of inertia. A rigid spacecraft body consisting of a set of reaction wheels (i.e., spinning rotors) may be modeled as a gyrostat. We then present the unit quaternion as our choice for attitude parameterization and the equations for quaternion kinematics. As a convention common in aerospace applications, we assume our attitude parameterization describes the orientation of the spacecraft body frame with respect to some chosen inertial frame. That is, we use the quaternion to represent the rotation required to express a vector coordinated in the spacecraft body frame as a vector coordinated in the inertial frame.

Gyrostat Dynamics

Given the mass moment of inertia matrix J for a rigid body and its angular velocity with respect to a reference frame, the angular momentum vector $\mathbf{h}_r$ is constant when there are no external torques:

$$\boldsymbol{\tau} = 0 \implies \dot{\mathbf{h}}_r = 0 \implies \mathbf{h}_r = J\boldsymbol{\omega} = \text{constant} \tag{2.39}$$

Coordinated in an inertial frame, we have:

$$[\mathbf{h}_r]^{\mathrm{I}} = [J]^{\mathrm{I}}[\boldsymbol{\omega}]^{\mathrm{I}} \tag{2.40}$$

When there exist external torques, we note that the moment of inertia is no longer constant, i.e., $[\dot{J}]^{\mathrm{I}} \neq 0$:

$$[\dot{\mathbf{h}}_r]^{\mathrm{I}} = [J]^{\mathrm{I}}[\dot{\boldsymbol{\omega}}]^{\mathrm{I}} + [\dot{J}]^{\mathrm{I}}[\boldsymbol{\omega}]^{\mathrm{I}} = [\boldsymbol{\tau}]^{\mathrm{I}} \tag{2.41}$$

For this reason, we consider it better to work in a body-fixed frame where the inertia matrix $[J]^{\mathrm{B}}$ is constant for a rigid body or gyrostat:

$$[\dot{\mathbf{h}}_r]^{\mathrm{B}} = [J]^{\mathrm{B}}[\dot{\boldsymbol{\omega}}]^{\mathrm{B}} = [\boldsymbol{\tau}]^{\mathrm{B}} \tag{2.42}$$

A vector coordinated in the body frame or inertial frame are related through a rotation matrix: $[\mathbf{x}]^{\mathrm{I}} = {}^{\mathrm{I}}R^{\mathrm{B}}[\mathbf{x}]^{\mathrm{B}}$. We use the Transport Theorem to relate the time derivative of vector in an inertial frame to its time derivative in the rotating body frame:

$$[\dot{\mathbf{h}}_r]^{\mathrm{I}} = {}^{\mathrm{I}}R^{\mathrm{B}} \left([\dot{\mathbf{h}}_r]^{\mathrm{B}} + [\boldsymbol{\omega}]^{\mathrm{B}} \times [\mathbf{h}_r]^{\mathrm{B}} \right) \tag{2.43}$$

$$[\boldsymbol{\tau}]^{\mathrm{I}} = {}^{\mathrm{I}}R^{\mathrm{B}} \left([\dot{\mathbf{h}}_r]^{\mathrm{B}} + [\boldsymbol{\omega}]^{\mathrm{B}} \times [\mathbf{h}_r]^{\mathrm{B}} \right) \tag{2.44}$$

After coordinating in the body frame and plugging in expressions for the angular momentum vector and its derivative, we get:

$$^{\mathrm{B}}R^{\mathrm{I}}[\boldsymbol{\tau}]^{\mathrm{I}} = \left([\dot{\mathbf{h}}_r]^{\mathrm{B}} + [\boldsymbol{\omega}]^{\mathrm{B}} \times [\mathbf{h}_r]^{\mathrm{B}}\right) \tag{2.45}$$

$$[\boldsymbol{\tau}]^{\mathrm{B}} = [\dot{\mathbf{h}}_r]^{\mathrm{B}} + [\boldsymbol{\omega}]^{\mathrm{B}} \times [\mathbf{h}_r]^{\mathrm{B}} \tag{2.46}$$

$$= [J]^{\mathrm{B}}[\dot{\boldsymbol{\omega}}]^{\mathrm{B}} + [\boldsymbol{\omega}]^{\mathrm{B}} \times [J]^{\mathrm{B}}[\boldsymbol{\omega}]^{\mathrm{B}} \tag{2.47}$$

Dropping the bracket notation, we recognize the well-known Euler equations for rigid body dynamics, coordinated in a body-fixed frame:

$$\boldsymbol{\tau} = \dot{\mathbf{h}}_r + \boldsymbol{\omega} \times \mathbf{h}_r \tag{2.48}$$

$$= J\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times J\boldsymbol{\omega} \tag{2.49}$$

We now assume that the rigid body encases a set of rotors, each spinning about an axis fixed with respect to the rigid body. The total momentum for the coupled system is now:

$$\mathbf{h}_r = J\boldsymbol{\omega} + \boldsymbol{\rho} \tag{2.50}$$

where $\boldsymbol{\rho} \in \mathbb{R}^3$ is the effective angular momentum of the rotors, expressed in the body-fixed frame. If we plug in the new expressions for $\mathbf{h}_r$ and $\dot{\mathbf{h}}_r = J\dot{\boldsymbol{\omega}} + \dot{\boldsymbol{\rho}}$ into (2.48), we get:

$$\boldsymbol{\tau} = (J\dot{\boldsymbol{\omega}} + \dot{\boldsymbol{\rho}}) + \boldsymbol{\omega} \times (J\boldsymbol{\omega} + \boldsymbol{\rho}) \tag{2.51}$$

Rearranging terms, we have:

$$\dot{\boldsymbol{\omega}} = -J^{-1}\left(\boldsymbol{\omega} \times (J\boldsymbol{\omega} + \boldsymbol{\rho}) + \dot{\boldsymbol{\rho}} - \boldsymbol{\tau}\right) \tag{2.52}$$

The effective angular momentum of the rotors $\boldsymbol{\rho}$ can be expressed as:

$$\boldsymbol{\rho} := A_r \boldsymbol{\rho_r} \,, \tag{2.53}$$

where, if we assume that the number of rotors equals four ($n_r = 4$), then the rotor momenta vector is:

$$\boldsymbol{\rho_r} := \begin{bmatrix} \rho_{\mathbf{r}_1} & \rho_{\mathbf{r}_2} & \rho_{\mathbf{r}_3} & \rho_{\mathbf{r}_4} \end{bmatrix}^{\top} \tag{2.54}$$

and $A_r \in \mathbb{R}^{3 \times n_r}$ is the *actuator Jacobian*, whose columns represent the axes of rotation for the rotors.

We assume single-integrator dynamics for the rotors: $\dot{\boldsymbol{\rho}_{\mathbf{r}}} = \boldsymbol{\tau_r}$, where $\boldsymbol{\tau_r} \in \mathbb{R}^{nr}$ is the applied torque about each rotor:

$$\boldsymbol{\tau_r} := \begin{bmatrix} \tau_{\mathbf{r}_1} & \tau_{\mathbf{r}_2} & \tau_{\mathbf{r}_3} & \tau_{\mathbf{r}_4} \end{bmatrix}^{\top} \tag{2.55}$$

We also assume there are no external torques, that is, we set $\boldsymbol{\tau} = \mathbf{0}$ in (2.52). We then have the gyrostat equations of motion:

$$\dot{\boldsymbol{\omega}}(t) = -J^{-1}\big(\boldsymbol{\omega}(t) \times (J\boldsymbol{\omega}(t) + A_r\boldsymbol{\rho}_{\mathbf{r}}(t)) + A_r\boldsymbol{\tau}_{\mathbf{r}}(t)\big) \tag{2.56}$$

$$\dot{\boldsymbol{\rho}}_{\mathbf{r}}(t) = \boldsymbol{\tau}_{\mathbf{r}}(t) \tag{2.57}$$

where we explicitly show the time-dependence of the angular velocity state vector, rotor angular momentum state vector, and the rotor torque input vector.

Another way to approximately express the gyrostat equations are:

$$\dot{\boldsymbol{\omega}}(t) = -J^{-1}\big(\boldsymbol{\omega}(t) \times J\boldsymbol{\omega}(t) - \boldsymbol{\tau}_{ctrl}(t)\big)$$

$$\dot{\boldsymbol{\rho}}_{\mathbf{r}}(t) = \boldsymbol{\tau}_{\mathbf{r}}(t)$$

$$\boldsymbol{\tau}_{\mathbf{r}}(t) = A_r^{\dagger}\big(-\boldsymbol{\omega}(t) \times A_r\mathbf{r}(t) - \boldsymbol{\tau}_{ctrl}(t)\big)$$

A common approach in attitude control is to directly design a control law for the effective torque on the rigid body: $\boldsymbol{\tau}_{ctrl} \in \mathbb{R}^3$. A control law can be designed with constraints on $\boldsymbol{\tau}_{ctrl}$ and the body angular velocity $\boldsymbol{\omega}$, also known as the *slew rate*. However, it is then assumed that the onboard attitude control and determination system (ADCS) can achieve the effective torque command $\boldsymbol{\tau}_{ctrl}$ by applying the appropriate actuator command $\boldsymbol{\tau}_{\mathbf{r}} \in \mathbb{R}^4$. This may be problematic when the actual rotor dynamics are more complicated than the single-integrator assumption and when there are actuator constraints. Momentum actuators, such as reaction wheels, often have bounds on the magnitude of the rotor momentum and the magnitude of the rotor torque. Furthermore, the use of the pseudo-inverse on the actuator Jacobian assumes a minimum Euclidean norm solution for the choice of $\boldsymbol{\tau}_{\mathbf{r}}$ to achieve $\boldsymbol{\tau}_{ctrl}$. For these reasons, we use the gyrostat equations as represented by (2.56)-(2.57) and seek to design optimal trajectories and control laws for the rotor momenta and torques. While the equations of orbital motion, from the previous section, are coordinated in the inertial frame, the gyrostat equations are coordinated in the body frame.

Quaternion Kinematics

Let us consider an arbitrary unit vector $\mathbf{d} \in \mathbb{R}^3$, where $\|\mathbf{d}\| = 1$, as an axis of rotation. We define $\theta \in (-\pi, \pi]$ as an angle of rotation about this axis. As a convention, we assume rotation in the counterclockwise direction is positive.

Following [6] and [7], we define the unit quaternion (or Euler Symmetric Parameters) as:

$$\mathbf{q} := \left[\begin{bmatrix} \mathbf{q}_v \\ q_s \end{bmatrix}\right] := \left[\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix}\right] := \left[\begin{bmatrix} \mathbf{d}\sin(\frac{\theta}{2}) \\ \cos(\frac{\theta}{2}) \end{bmatrix}\right] \tag{2.58}$$

We use the convention of stacking the vector part on top of the scalar part. We may confirm that, by definition, $\|\mathbf{q}\|_2 = 1$. Furthermore, we note that the unit quaternion representation

of a rotation is not unique, that is, $\mathbf{q} = -\mathbf{q}$ as demonstrated below:

$$\left[\left[\begin{array}{c} \mathbf{d}\sin(\frac{\theta+2\pi}{2}) \\ \\ \cos(\frac{\theta+2\pi}{2}) \end{array}\right]\right] = \left[\left[\begin{array}{c} -\mathbf{d}\sin(\frac{\theta}{2}) \\ \\ -\cos(\frac{\theta}{2}) \end{array}\right]\right] = -\mathbf{q} \tag{2.59}$$

We define quaternion multiplication between two quaternions $\mathbf{q}^a$ and $\mathbf{q}^b$ as:

$$\mathbf{q}^a\mathbf{q}^b = \left[\begin{array}{c} q_s^a\mathbf{q}_v^b + q_s^b\mathbf{q}_v^a + \mathbf{q}_v^a \times \mathbf{q}_v^b \\ q_s^a q_s^b - (\mathbf{q}_v^a)^\top\mathbf{q}_v^b \end{array}\right] \tag{2.60}$$

$$= \left[\begin{array}{cc} q_s^a I + (\mathbf{q}_v^a)^\times & \mathbf{q}_v^a \\ -(\mathbf{q}_v^a)^\top & q_s^a \end{array}\right]\left[\begin{array}{c} \mathbf{q}_v^b \\ q_s^b \end{array}\right] \tag{2.61}$$

$$= \left[\begin{array}{cc} q_s^b I - (\mathbf{q}_v^b)^\times & \mathbf{q}_v^b \\ -(\mathbf{q}_v^b)^\top & q_s^b \end{array}\right]\left[\begin{array}{c} \mathbf{q}_v^a \\ q_s^a \end{array}\right] \tag{2.62}$$

The quaternion multiplication operation is associative and distributive, but, in general, it is not commutative, i.e., $\mathbf{q}^a\mathbf{q}^b \neq \mathbf{q}^b\mathbf{q}^a$.

We introduce certain properties of the (unit) quaternion. First, we note that there exists an identity quaternion:

$$\mathbf{q}^{\mathrm{I}} := \left[\left[\begin{array}{c} \mathbf{d}\sin(\frac{0}{2}) \\ \\ \cos(\frac{0}{2}) \end{array}\right]\right] = \left[\begin{array}{c} 0 \\ 0 \\ 0 \\ 1 \end{array}\right] \tag{2.63}$$

such that $\mathbf{q}\mathbf{q}^{\mathrm{I}} = \mathbf{q}^{\mathrm{I}}\mathbf{q} = \mathbf{q}$. Furthermore, we have the quaternion inverse, as known as the quaternion conjugate:

$$\mathbf{q}^{-1} := \left[\left[\begin{array}{c} \mathbf{d}\sin(\frac{-\theta}{2}) \\ \\ \cos(\frac{-\theta}{2}) \end{array}\right]\right] = \left[\left[\begin{array}{c} -\mathbf{d}\sin(\frac{\theta}{2}) \\ \\ \cos(\frac{\theta}{2}) \end{array}\right]\right] = \left[\left[\begin{array}{c} -\mathbf{q}_v \\ \\ q_s \end{array}\right]\right] =: \mathbf{q}^+ \tag{2.64}$$

such that $\mathbf{q}^+\mathbf{q} = \mathbf{q}\mathbf{q}^+ = \mathbf{q}^{\mathrm{I}}$. Using the quaternion conjugate, the following relationship holds: $\mathbf{q}^3 = \mathbf{q}^2\mathbf{q}^1 \iff \mathbf{q}^1 = (\mathbf{q}^2)^+\mathbf{q}^3$ for different quaternions $\mathbf{q}^1$ and $\mathbf{q}^2$.

As defined in [7] and used in [8] and [9], we may measure the attitude error between a given quaternion $\mathbf{q}$ and a desired quaternion $\bar{\mathbf{q}}$ by computing the error quaternion:

$$\mathbf{q}^e = \left[\begin{array}{c} q_1^e \\ q_2^e \\ q_3^e \\ q_4^e \end{array}\right] := \bar{\mathbf{q}}^+\mathbf{q} = \left[\begin{array}{cc} \bar{q}_s^+ I + (\bar{\mathbf{q}}_v^+)^\times & \bar{\mathbf{q}}_v^+ \\ -(\bar{\mathbf{q}}_v^+)^\top & \bar{q}_s^+ \end{array}\right]\left[\begin{array}{c} q_1 \\ q_2 \\ q_3 \\ q_4 \end{array}\right] = \left[\begin{array}{cccc} \bar{q}_4 & \bar{q}_3 & -\bar{q}_2 & -\bar{q}_1 \\ -\bar{q}_3 & \bar{q}_4 & \bar{q}_1 & -\bar{q}_2 \\ \bar{q}_2 & -\bar{q}_1 & \bar{q}_4 & -\bar{q}_3 \\ \bar{q}_1 & \bar{q}_2 & \bar{q}_3 & \bar{q}_4 \end{array}\right]\left[\begin{array}{c} q_1 \\ q_2 \\ q_3 \\ q_4 \end{array}\right] \tag{2.65}$$

Note that if $\mathbf{q} = \bar{\mathbf{q}}$, then their corresponding error quaternion equals the identity quaternion, i.e., $\mathbf{q}^e = \bar{\mathbf{q}}^+\mathbf{q} = \bar{\mathbf{q}}^+\bar{\mathbf{q}} = \mathbf{q}^{\mathrm{I}}$.

We recall our convention of using attitude parameterizations to represent the orientation of the body frame with respect to an inertial frame, not vice versa. That is, we use attitude parameterizations (e.g., rotation matrix, unit quaternion, Euler axis and angle) to represent the rotation required to express a 3-dimensional vector coordinated in the body frame as a vector coordinated in an inertial frame. The following relationship using the rotation matrix:

$$[\mathbf{x}]^{\mathrm{I}} = {}^{\mathrm{I}}R^{\mathrm{B}}[\mathbf{x}]^{\mathrm{B}} \tag{2.66}$$

may also be represented using the unit quaternion:

$$\begin{bmatrix} [\mathbf{x}]^{\mathrm{I}} \\ 0 \end{bmatrix} = \mathbf{q} \begin{bmatrix} [\mathbf{x}]^{\mathrm{B}} \\ 0 \end{bmatrix} \mathbf{q}^+ \tag{2.67}$$

where we represent a 3-dimensional vector $\mathbf{x}$ as a "pure" quaternion $\begin{bmatrix} \mathbf{x}^\top & 0 \end{bmatrix}^\top$. The rotation matrix corresponding to a quaternion is found with the following equation, derived in (B.1):

$$ {}^{\mathrm{I}}R^{\mathrm{B}} = \mathrm{I} + 2\mathbf{q}_v^\times \left(q_s \mathrm{I} + \mathbf{q}_v^\times \right) \tag{2.68}$$

The time derivative of the quaternion is:

$$\dot{\mathbf{q}}(t) = \frac{1}{2}\mathbf{q}(t) \begin{bmatrix} \boldsymbol{\omega}(t) \\ 0 \end{bmatrix} \tag{2.69}$$

The expression shows that the time derivative of the quaternion is equal to one-half of the product of the quaternion, representing the attitude of the body frame w.r.t. inertial frame, and the pure quaternion representation of the body angular velocity w.r.t inertial frame. The derivation of this expression is shown in (B.2).

Equations of Rotational Motion

The equations of rotational motion for a gyrostat, consisting of a rigid body and a set of actuation rotors with rotation axes fixed w.r.t. the rigid body, are

$$\dot{\mathbf{q}}(t) = \frac{1}{2}\mathbf{q}(t) \begin{bmatrix} \boldsymbol{\omega}(t) \\ 0 \end{bmatrix} \tag{2.70}$$

$$\dot{\boldsymbol{\omega}}(t) = -J^{-1}\left(\boldsymbol{\omega}(t) \times (J\boldsymbol{\omega}(t) + A_r\boldsymbol{\rho_r}(t)) + A_r\boldsymbol{\tau_r}(t) - \boldsymbol{\tau}_{\mathbf{ext}}(t) \right) \tag{2.71}$$

$$\dot{\boldsymbol{\rho_r}}(t) = \boldsymbol{\tau_r}(t) \tag{2.72}$$

where have combined equations (2.56), (2.57) and (2.69). External moments (e.g., due to gravity gradient, atmospheric drag, solar radiation pressure, Earth's magnetic field) may be include in the $\boldsymbol{\tau}_{\mathbf{ext}}$ term.

2.3 Trajectory Optimization

As surveyed by Betts [10], there is an expanse of literature on trajectory optimization methods that may be characterized by, for example, the solution approach (i.e., indirectly satisfying necessary conditions for optimality or directly solving a transcribed version of the problem), or the transcription process (shooting versus collocation). In [11], various direct collocation methods are introduced while [12] focuses on an indirect pseudospectral method that has been used in practice, for example, to control the International Space Station with a zero-propellant maneuver. Regardless of the approach, many trajectory optimization methods treat a problem in its original nonlinear, non-convex form, requiring the use of a nonlinear programming solver [13], which may be computationally inefficient with long solution times.

Recent advances in sequential convex programming (SCP) have enabled efficient computation of locally optimal trajectories for nonlinear systems with non-convex constraints and objectives. SCP is an iterative method that repeatedly formulates and solves a convex, finite-dimensional parameter optimization problem that approximates the original non-convex optimal control problem. A convex formulation is typically achieved by linearizing the nonlinear system around a nominal trajectory (i.e., the solution from the previous iteration) and approximating any non-convex constraints and objective with Taylor series expansions. Fast and reliable Interior Point Method algorithms [14] may be used to solve these convex subproblems. In the early works of [15] and [16], successive convexification, a specific implementation of the SCP method, is introduced to find minimum-fuel and minimum-time trajectories in a 6-DOF rocket landing problem. In [17]- [18], certain details of the implementation, including choice of discretization method, constraint satisfaction between temporal nodes, convexification of non-convex constraints, and algorithm convergence properties are explored. Finally, the works of [19]- [20] introduce state-triggered constraints and address real-time, onboard implementations that may produce solutions in a fraction of a second.

In this section, we first describe how a general optimal control problem (OCP) is transcribed into a convex, finite-dimensional parameter optimization problem (OPT). We then review the sequential convex programming algorithm used in this paper. Consider the following continuous-time dynamical system:

$$\dot{\mathbf{x}}(t) := \frac{d}{dt}\mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \qquad \forall\ t \in [t_0, t_f] \tag{2.73}$$

defined over the given time span, where $\mathbf{x}(t) \in \mathbb{R}^{n_x}$ is the state of the system and $\mathbf{u}(t) \in \mathbb{R}^{n_u}$ is the input to the system.

Scaling of System Variables and Parameters

As suggested by [13], [12], [21], scaling (or nondimensionalization) is widely used in numerical methods for trajectory optimization. Using the initial conditions and parameters intrinsic to the dynamical system or problem, we define the following designer units:

Orbital Problems

$$r_0 := \|\mathbf{r}(t_0)\| \qquad \text{displacement (initial distance)} \qquad (2.74)$$

$$s_0 := 2\pi\sqrt{\frac{r_0^3}{\mu}} \qquad \text{time (initial orbital period)} \qquad (2.75)$$

$$v_0 := \frac{r_0}{s_0} \qquad \text{speed} \qquad (2.76)$$

$$g_0 := \frac{r_0}{s_0^2} \qquad \text{acceleration} \qquad (2.77)$$

$$m_0 := m(t_0) \qquad \text{mass (initial mass)} \qquad (2.78)$$

$$T_0 := \frac{m_0 r_0}{s_0^2} \qquad \text{force} \qquad (2.79)$$

$$P_0 := \frac{m_0}{r_0 s_0^2} \qquad \text{pressure} \qquad (2.80)$$

$$S_0 := r_0^2 \qquad \text{area} \qquad (2.81)$$

$$\mu_0 := \frac{r_0^3}{s_0^2} \qquad \text{standard gravitational parameter} \qquad (2.82)$$

$$\rho_0 := \frac{m_0}{r_0^3} \qquad \text{atmospheric density} \qquad (2.83)$$

Attitude Problems

$$r_0 := 1[\text{m}] \qquad \text{displacement} \qquad (2.84)$$

$$s_0 := 1[\text{s}] \qquad \text{time} \qquad (2.85)$$

$$m_0 := m \qquad \text{mass} \qquad (2.86)$$

$$J_0 := m_0 r_0^2 \qquad \text{mass moment of inertia} \qquad (2.87)$$

$$\omega_0 := \frac{1}{s_0} \qquad \text{angular velocity} \qquad (2.88)$$

$$\rho_{r0} := \frac{m_0 r_0^2}{s_0} \qquad \text{angular momentum} \qquad (2.89)$$

$$\tau_{r0} := \frac{m_0 r_0^2}{s_0^2} \qquad \text{moment} \qquad (2.90)$$

$$(2.91)$$

Using the designer units listed above, we scale system variables and parameters such that the equations of motion are nondimensionalized:

Orbital System Variables

$$\bar{m} = \frac{m}{m_0} \qquad \bar{\mathbf{r}} = \frac{\mathbf{r}}{r_0} \qquad \bar{\mathbf{v}} = \frac{\mathbf{v}}{v_0} \qquad \bar{\mathbf{T}} = \frac{\mathbf{T}}{T_0} \qquad (2.92)$$

Attitude System Variables

$$\bar{\mathbf{q}} = \mathbf{q} \qquad \bar{\boldsymbol{\omega}} = \frac{\boldsymbol{\omega}}{\omega_0} \qquad \bar{\boldsymbol{\rho}}_{\mathbf{r}} = \frac{\boldsymbol{\rho}_{\mathbf{r}}}{\rho_{r0}} \qquad \bar{\boldsymbol{\tau}}_{\mathbf{r}} = \frac{\boldsymbol{\tau}_{\mathbf{r}}}{\tau_{r0}} \qquad (2.93)$$

System Parameters

$$
\begin{aligned}
\bar{t} &= \frac{t}{s_0} & \bar{I}_{sp} &= \frac{I_{sp}}{s_0} & \bar{g} &= \frac{g}{g_0} & \bar{\mu} &= \frac{\mu}{\mu_0} \\
\bar{r}_E &= \frac{r_E}{r_0} & \bar{S} &= \frac{S}{S_0} & \bar{\rho} &= \frac{\rho}{\rho_0} & \bar{P} &= \frac{P}{P_0} \\
\bar{C}_D &= C_D & \bar{J}_2 &= J_2 & \bar{J} &= \frac{J}{J_0} & \bar{J}_r &= \frac{J_r}{J_0} & (2.94)
\end{aligned}
$$

In all problem formulations to follow, we assume that system variables and parameters have already been scaled, and we drop the overbar notation signifying scaled quantities.

Time Normalization

In addition to scaling system variables and parameters, we normalize the independent variable of time t in the dynamical system (2.73) so that our problems, originally defined on $t_0 \leq t \leq t_f$, are transformed to be defined on $\frac{t_0}{t_f} \leq \tau \leq 1$. Normalized time τ is related to the original time t via a a normalization factor t_f:

$$\tau := \frac{t}{t_f} \qquad (2.95)$$

In free-final-time (e.g., minimum-time) problems, t_f is treated as a decision variable to be solved for [16]. In fixed-final-time problems, we can treat t_f either as a constant parameter or as a decision variable, that we constrain to take on a fixed value. For the fixed-final-time problems in this dissertation, we take the latter approach to maintain general notation that is also used for free-final-time problems.

Since $t = t_f \tau \Rightarrow dt = t_f d\tau \Rightarrow \frac{dt}{d\tau} = t_f$, the derivative of the scaled state with respect to normalized time is:

$$\mathbf{x}'(t) := \frac{d}{d\tau}\mathbf{x}(t) = \frac{dt}{d\tau}\frac{d}{dt}\mathbf{x}(t) = t_f \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \qquad \forall\, t \in [t_0, t_f] \qquad (2.96)$$

Note that the original time t can be expressed as a function of τ, i.e., $t(\tau) = t_f \tau$. If we assume $t_0 = 0$, we can represent (2.96) in terms of normalized time:

$$\mathbf{x}'(t(\tau)) = \mathbf{x}'(\tau) = t_f \mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau)) =: \tilde{\mathbf{f}}(\mathbf{x}(\tau), \mathbf{u}(\tau), t_f) \qquad \forall\, t \in [0\,, 1] \qquad (2.97)$$

In this final expression, it is clear that the dynamical system is a function of state $\mathbf{x}$, input $\mathbf{u}$, and final time t_f.

Linearization of Dynamics

Assuming (2.97) is a nonlinear but differentiable function, we may approximate it with a first-order Taylor expansion about a given trajectory $\{\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{t}_f\}$:

$$\mathbf{x}'(\tau) = \tilde{\mathbf{f}}(\mathbf{x}(\tau), \mathbf{u}(\tau), t_f) \approx A(\tau)\mathbf{x}(\tau) + B(\tau)\mathbf{u}(\tau) + \boldsymbol{\Sigma}(\tau)t_f + \boldsymbol{\xi}(\tau) \quad \forall\, \tau \in [0\,,1] \quad (2.98)$$

where we denote the first-order partial derivative matrices of $\tilde{\mathbf{f}}(\mathbf{x}(\tau), \mathbf{u}(\tau), t_f)$, evaluated about $\{\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau), \bar{t}_f\}$, as

$$A(\tau) := D_{\mathbf{x}}\tilde{\mathbf{f}}(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau), \bar{t}_f) = \bar{t}_f D_{\mathbf{x}}\mathbf{f}(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau)) \qquad (2.99)$$

$$B(\tau) := D_{\mathbf{u}}\tilde{\mathbf{f}}(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau), \bar{t}_f) = \bar{t}_f D_{\mathbf{u}}\mathbf{f}(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau)) \qquad (2.100)$$

$$\boldsymbol{\Sigma}(\tau) := D_{t_f}\tilde{\mathbf{f}}(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau), \bar{t}_f) = \mathbf{f}(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau)) \qquad (2.101)$$

with the following dynamical approximation offset term:

$$\boldsymbol{\xi}(\tau) := -(A(\tau)\bar{\mathbf{x}}(\tau) + B(\tau)\bar{\mathbf{u}}(\tau)) \qquad (2.102)$$

Discretization of Linearized Dynamics

As described in [22], the process of converting an optimal control problem into a parameter optimization problem begins by dividing the time duration of the optimal control problem into intervals using K temporal nodes. The nodes may be chosen to be equally spaced, creating $K-1$ equally-sized time intervals:

$$0 =: \tau_1 < \tau_2 < \cdots < \tau_k < \ldots < \tau_{K-1} < \tau_K := 1 \qquad (2.103)$$

We refer to state, input, and offset terms at each node with the following shorthand notation:

$$\mathbf{x}_k := \mathbf{x}(\tau_k), \ \mathbf{u}_k := \mathbf{u}(\tau_k), \ \boldsymbol{\xi}_k := \boldsymbol{\xi}(\tau_k) \qquad \forall\, k = 1, \ldots, K \qquad (2.104)$$

We use the First-Order Hold (FOH)-interpolation based discretization method in our implementation. As demonstrated in [23], the FOH discretization provides fast computational time and achieves similar accuracy when compared to more advanced pseudospectral methods. Furthermore, it was shown that if convex input constraints are satisfied at the nodes, then inter-nodal convex input constraint satisfaction is also guaranteed [17]. For convenience, we review the process used in [15]- [24] below.

The FOH interpolation represents the input within each of the $K-1$ intervals as:

$$\mathbf{u}(\tau) = \lambda_k^- \mathbf{u}_k + \lambda_k^+ \mathbf{u}_{k+1} \qquad \forall\, \tau \in [\tau_k, \tau_{k+1}]\ ,\ k = 1, \ldots, K-1 \qquad (2.105)$$

where

$$\lambda_k^- := \frac{\tau_{k+1} - \tau}{\tau_{k+1} - \tau_k}\ , \qquad \lambda_k^+ := \frac{\tau - \tau_k}{\tau_{k+1} - \tau_k} \qquad (2.106)$$

The exact discretization of (2.98) is then:

$$\mathbf{x}_{k+1} = A_k\mathbf{x}_k + B_k^-\mathbf{u}_k + B_k^+\mathbf{u}_{k+1} + \boldsymbol{\Sigma}_k t_f + \boldsymbol{\xi}_k \qquad \forall\, k = 1,\ldots,K-1 \tag{2.107}$$

$$A_k := \Phi(\tau_{k+1}, \tau_k) \tag{2.108}$$

$$B_k^- := A_k \int_{\tau_k}^{\tau_{k+1}} \Phi^{-1}(\tau, \tau_k)\lambda_k^-(\tau)B(\tau)d\tau \tag{2.109}$$

$$B_k^+ := A_k \int_{\tau_k}^{\tau_{k+1}} \Phi^{-1}(\tau, \tau_k)\lambda_k^+(\tau)B(\tau)d\tau \tag{2.110}$$

$$\boldsymbol{\Sigma}_k := A_k \int_{\tau_k}^{\tau_{k+1}} \Phi^{-1}(\tau, \tau_k)\boldsymbol{\Sigma}(\tau)d\tau \tag{2.111}$$

$$\boldsymbol{\xi}_k := A_k \int_{\tau_k}^{\tau_{k+1}} \Phi^{-1}(\tau, \tau_k)\boldsymbol{\xi}(\tau)d\tau \tag{2.112}$$

The state transition matrix satisfies the following matrix differential equation and initial condition within each interval:

$$\frac{d}{d\tau}\Phi(\tau, \tau_k) = A(\tau)\Phi(\tau, \tau_k), \quad \Phi(\tau_k, \tau_k) = \mathrm{I}^{n_x} \tag{2.113}$$

In practice, the integrands of (2.109)-(2.112) along with (2.113) and (2.97) are numerically integrated from the start to the end of each interval using a nominal trajectory $\{\bar{\mathbf{x}}_k, \bar{\mathbf{u}}_k, \bar{t}_f, \bar{\boldsymbol{\xi}}_k\}$ for $k = 1,\ldots,K-1$. Note that we initialize the numerical integration for each interval with points from a nominal trajectory, rather than with the terminal points found by integration in the previous temporal interval. This approach resembles a multiple-shooting discretization method, which has shown to improve convergence of the SCP algorithm by keeping solutions closer to the nominal trajectory. In contrast, a single shooting method allows approximation errors to grow in later temporal intervals [23].

Sequential Convex Programming Method

For the trajectory optimization applications in this dissertation, we use the Penalized Trust Region (PTR) variant of SCP described in [20]. A key difference with Successive Convexification (SCvx) studied in [18] is that PTR treats *trust regions* as soft constraints placed in the objective whereas SCvx enforces hard trust region constraints that are updated based on a rule. An advantage of SCvx is that convergence of this method is guaranteed. However, the method employs slack variables that may cause the approximately solved problem to be far from the original problem if they take on non-zero values in the solution. Hence, a converged solution via SCvx may be infeasible for the original problem. In the following subsections we describe the PTR implementation of SCP, which employs *virtual controls* and trust regions, as well as an approach to constraint convexification.

Virtual Controls

While iteratively solving a trajectory optimization problem using sequential convex programming, the approximated convex problem may become infeasible at an intermediate iteration before convergence. This *artificial infeasibility* [16] is frequently encountered in the early iterations of the algorithm if the dynamics are linearized about a poor initial guess. To alleviate this issue, slack variables called *virtual controls* $\boldsymbol{\nu}_k \in \mathbb{R}^{n_x}$ are added to the discrete-time equations of motion (2.107) at the temporal nodes:

$$\mathbf{x}_{k+1} = A_k\mathbf{x}_k + B_k^-\mathbf{u}_k + B_k^+\mathbf{u}_{k+1} + \boldsymbol{\Sigma}_k t_f + \boldsymbol{\xi}_k + \boldsymbol{\nu}_k \qquad \forall \, k = 1, \dots, K-1 \qquad (2.114)$$

These virtual controls act as dynamic relaxation terms that take on nonzero values, when necessary, to prevent dynamic infeasibility. In turn, use of these slack variables is heavily penalized with a term added to the objective function:

$$J_{vc}\left(\{\boldsymbol{\nu}_k\}_{k=1}^K\right) = w_{vc} \sum_{k=1}^K \|\boldsymbol{\nu}_k\|_1 = w_{vc}\|\boldsymbol{\nu}\|_1 \qquad (2.115)$$

where w_{vc} is a large positive weight and $\boldsymbol{\nu} = [\boldsymbol{\nu}_1^\top, \dots, \boldsymbol{\nu}_K^\top]^\top$. Minimization of the 1-norm term encourages sparsity in the virtual control vector $\boldsymbol{\nu}$.

Trust Regions

We also implement a trust region penalty term to ensure that the solver does not stray too far from a nominal trajectory, where the linearized model becomes less accurate. In PTR, the deviation of decision variables from the solution of the previous iteration is penalized with the 2-norm of weighted deviations:

$$J_{tr}\left(\{\mathbf{x}_k, \mathbf{u}_k\}_{k=1}^K, t_f\right) = w_{tr} \sum_{k=1}^K \left\| \begin{bmatrix} \mathbf{x}_k \\ \mathbf{u}_k \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{x}}_k \\ \bar{\mathbf{u}}_k \end{bmatrix} \right\|_2 + w_{tr}\|t_f - \bar{t}_f\|_2 \qquad (2.116)$$

Note that we use a single, scalar weight $w_{tr} > 0$ in the penalty term above to equally penalize deviations in the state, input and final-time decision variables, assuming the variables have been scaled. However, we may modify the expression to have relative or adaptive weighting.

Constraint and Objective Approximation

Let us consider a set of general (non-convex) constraints at each time τ on the continuous-time variables $\mathbf{x}$ and $\mathbf{u}$:

$$\mathbf{g}_\tau\left(\mathbf{x}(\tau), \mathbf{u}(\tau)\right) \leq \mathbf{0} \qquad \forall \, \tau \in [0\,,1] \qquad (2.117)$$

These constraints can be approximated by linearizing about a nominal trajectory:

$$\tilde{\mathbf{g}}_\tau(\mathbf{x}(\tau), \mathbf{u}(\tau)) := \mathbf{g}_\tau(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau)) + D\mathbf{g}_\tau(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau)) \left(\begin{bmatrix} \mathbf{x}(\tau) \\ \mathbf{u}(\tau) \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{x}}(\tau) \\ \bar{\mathbf{u}}(\tau) \end{bmatrix} \right) \leq \mathbf{0} \quad \forall \, \tau \in [0\,,1]$$

$$(2.118)$$

We further approximate the constraints by explicitly enforcing them only at the temporal nodes of our discretization:

$$\tilde{\mathbf{g}}_k(\mathbf{x}_k, \mathbf{u}_k) \leq \mathbf{0} \qquad \forall\, k = 1, \ldots, K \tag{2.119}$$

In general, inter-nodal constraint satisfaction is not guaranteed. However, the constraints at the nodes can be carefully designed so that constraints are enforced for all time [17]. In a similar fashion, any non-convex cost terms J_{obj} may also be approximated with a Taylor series expansion about a nominal trajectory to form $\tilde{J}_{obj}$.

We may also approximate integral constraints involving non-convex functions:

$$\int_0^{t_f} h(\mathbf{x}(t), \mathbf{u}(t))dt \leq 0 \tag{2.120}$$

$$\implies \int_0^1 t_f h(\mathbf{x}(\tau), \mathbf{u}(\tau))d\tau \leq 0 \tag{2.121}$$

$$\implies \int_0^1 t_f \tilde{h}(\mathbf{x}(\tau), \mathbf{u}(\tau))d\tau \leq 0 \tag{2.122}$$

$$\implies \frac{t_f}{K-1} \left(\sum_{k=2}^{K-1} \tilde{h}(\mathbf{x}_k, \mathbf{u}_k) + \frac{\tilde{h}(\mathbf{x}_1, \mathbf{u}_1) + \tilde{h}(\mathbf{x}_K, \mathbf{u}_K)}{2} \right) \leq 0 \tag{2.123}$$

where non-convex constraint terms $h(\mathbf{x}, \mathbf{u})$ may be approximated with their first-order Taylor series expansions $\tilde{h}(\mathbf{x}, \mathbf{u})$, evaluated about a nominal trajectory. Note that we use the trapezoidal rule to approximate the definite integral.

Problem Transcription

In this section, we summarize the process to transcribe an infinite-dimensional optimal control problem (OCP) into a convex, finite-dimensional optimization problem (OPT).

Consider a general OCP formulation, where the objective function, dynamics and constraints may be non-convex in the decision variables:

$$\underset{\mathbf{x}, \mathbf{u}}{\text{minimize}} \quad J_{obj}\left(\mathbf{x}, \mathbf{u}, t_f\right) \tag{2.124}$$

subject to:

$$\dot{\mathbf{x}}(t) = \mathbf{f}\left(\mathbf{x}(t), \mathbf{u}(t)\right) \qquad\qquad \forall\, t \in [t_0, t_f] \quad (2.125)$$

$$\mathbf{g}_t(\mathbf{x}(t), \mathbf{u}(t)) \leq \mathbf{0} \qquad\qquad \forall\, t \in [t_0, t_f] \quad (2.126)$$

$$\mathbf{x}(t_0) = \mathbf{x}_{initial} \tag{2.127}$$

After the time normalization step, the final time t_f becomes a decision variable, and the

problem is defined on the fixed interval $\tau \in [0, 1]$:

$$\underset{\mathbf{x},\mathbf{u},t_f}{\text{minimize}} \quad J_{obj}\left(\mathbf{x}, \mathbf{u}, t_f\right) \tag{2.128}$$

subject to:

$$\mathbf{x}'(\tau) = \tilde{\mathbf{f}}\left(\mathbf{x}(\tau), \mathbf{u}(\tau), t_f\right) \qquad\qquad \forall\, \tau \in [0, 1] \quad (2.129)$$
$$\mathbf{g}_\tau(\mathbf{x}(\tau), \mathbf{u}(\tau)) \leq \mathbf{0} \qquad\qquad \forall\, \tau \in [0, 1] \quad (2.130)$$
$$\mathbf{x}(0) = \mathbf{x}_{initial} \tag{2.131}$$

Non-convex objective and constraint functions are approximated as convex functions by using the procedure described in the previous section. The dynamics are also approximated by linearizing about a nominal trajectory $\{\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau), \bar{t}_f\}$ to get:

$$\underset{\mathbf{x},\mathbf{u},t_f}{\text{minimize}} \quad J_{obj}\left(\mathbf{x}, \mathbf{u}, t_f\right) \tag{2.132}$$

subject to:

$$\mathbf{x}'(\tau) = A(\tau)\mathbf{x}(\tau) + B(\tau)\mathbf{u}(\tau) + \boldsymbol{\Sigma}(\tau)t_f + \boldsymbol{\xi}(\tau) \qquad \forall\, \tau \in [0, 1] \quad (2.133)$$
$$\tilde{\mathbf{g}}_\tau(\mathbf{x}(\tau), \mathbf{u}(\tau)) \leq \mathbf{0} \qquad\qquad\qquad \forall\, \tau \in [0, 1] \quad (2.134)$$
$$\mathbf{x}(0) = \mathbf{x}_{initial} \tag{2.135}$$

Finally, we discretize the linearized, continuous-time model, and employ both virtual control and trust region terms to get the following OPT formulation:

$$\underset{\{\mathbf{x}_k,\mathbf{u}_k,\boldsymbol{\nu}_k\}_{k=1}^K,t_f}{\text{minimize}} \quad J_{obj}\left(\{\mathbf{x}_k, \mathbf{u}_k\}_{k=1}^K, t_f\right) + J_{tr}\left(\{\mathbf{x}_k, \mathbf{u}_k\}_{k=1}^K, t_f\right) + J_{vc}\left(\{\boldsymbol{\nu}_k\}_{k=1}^K\right) \tag{2.136}$$

subject to:

$$\mathbf{x}_{k+1} = A_k\mathbf{x}_k + B_k^-\mathbf{u}_k + B_k^+\mathbf{u}_{k+1} + \boldsymbol{\Sigma}_k t_f + \boldsymbol{\xi}_k + \boldsymbol{\nu}_k \qquad \forall\, k = 1, \ldots, K-1 \quad (2.137)$$
$$\tilde{\mathbf{g}}_k(\mathbf{x}_k, \mathbf{u}_k) \leq \mathbf{0} \qquad\qquad\qquad\qquad \forall\, k = 1, \ldots, K-1 \quad (2.138)$$
$$\mathbf{x}_1 = \mathbf{x}_{initial} \tag{2.139}$$

Algorithm

The sequential convex programming algorithm used in this paper is listed in Algorithm 1 where $\mathbf{S}^0$ is an initial guess at the solution. As a notation convention used in the algorithm, superscript i refers to the solution at the ith iteration of the algorithm:

$$\mathbf{S}^i := \begin{cases} \{\bar{\mathbf{x}}_1^i, \ldots, \bar{\mathbf{x}}_k^i, \ldots, \bar{\mathbf{x}}_K^i\}, \\ \{\bar{\mathbf{u}}_1^i, \ldots, \bar{\mathbf{u}}_k^i, \ldots, \bar{\mathbf{u}}_K^i\}, \\ \{\bar{\boldsymbol{\nu}}_1^i, \ldots, \bar{\boldsymbol{\nu}}_k^i, \ldots, \bar{\boldsymbol{\nu}}_K^i\}, \\ \qquad\qquad \bar{t}_f^{\,i} \end{cases} \tag{2.140}$$

The algorithm stops when either (1) the user-defined maximum number of SCP iterations N_{max} have been executed, or (2) the algorithm has converged on a solution, where we define convergence at the ith iteration as the satisfaction of the following two conditions:

$$\left(J_{vc}\left(\mathbf{S}^i\right) \leq \epsilon_{vc}\right) \ \wedge \ \left(J_{tr}\left(\mathbf{S}^i\right) \leq \epsilon_{tr}\right) \tag{2.141}$$

where ϵ_{vc} and ϵ_{tr} are user-defined convergence tolerances. The first condition ensures that a negligible amount of virtual controls is used, indicating that the converged solution is dynamically feasible. The second condition ensures that the solution remains sufficiently close to the nominal trajectory upon which the linearized dynamics assumed in the problem formulation is accurate. If the SCP algorithm converges on a solution, we record the iteration at which it converged.

Algorithm 1: Sequential Convex Programming

   Input   : $\mathbf{S}^0$
   Output : $\mathbf{S}^i, \text{flag}$
   for $i = 1 : N_{max}$ **do**
      | transcribe $\mathbf{OPT}^i\left(\mathbf{S}^{i-1}\right)$
      | $\mathbf{S}^i \leftarrow \mathbf{OPT}^i\left(\mathbf{S}^{i-1}\right)$
      | **if** $\mathbf{S}^i$ *converged* **then**
      | | $N_{converge} = i$
      | **return**
      | **if** $i = N_{max}$ **then**
      | | $N_{converge} = \emptyset$
      | **return**

Chapter 3

Agile Attitude Maneuvers

3.1 Motivation

Due to the proliferation of launch providers to low Earth orbit (LEO) and the trend towards smaller and more cost-efficient spacecraft, satellite constellations are enabling scientific missions and commercial applications that are otherwise impossible with a single, larger satellite. For example, a constellation of satellites in LEO may be coordinated to point towards coastal regions around the world to measure ocean color, atmospheric properties, phytoplankton concentrations, and ultimately assess the health of global coral reef ecosystems [25]. LEO constellations may also be employed to measure episodic precipitation and subsequent water flow in flood-prone cities [26]- [27]. Furthermore, constellations may be tasked to measure soil moisture in targeted regions to assess the risk of wildfires [28]. In addition to climate and environment monitoring, Earth-observing constellations are providing commercial and economic value by measuring, for instance, agricultural crop growth, infrastructure development, or logistical activity at airports and shipping container routes [29]. Beyond these Earth observation applications, LEO satellite constellations are also enabling space-based Internet and telecommunication services [30].

A satellite constellation can be viewed as an interconnection of cyber-physical systems to be precisely coordinated throughout each stage of a mission. Once a group of satellites is deployed into a desired orbital plane by a launch vehicle, the satellites enter the *orbit acquisition stage* where they must be phased relative to each other to achieve desired angular spacing. In [31], it is assumed that the orientation of the satellites may be controlled, with respect to their orbital velocities, so that either a minimum or maximum surface area is exposed to the LEO atmosphere. By inducing either a low or high atmospheric drag on each satellite, using a bang-bang control approach, the resulting differential drag between satellites is used to create angular separation. In [32], simulated annealing is used to design time-optimal differential drag commands for a group of up to 100 satellites. The method is demonstrated on actual constellations deployed in LEO. Assuming accurate attitude pointing, [33] formulates a linear program that produces differential drag commands taking on

continuous values between the minimum and maximum values, allowing the constellation to form not only under a minimum-time objective but also with a maximum altitude (or equivalently, maximum constellation lifetime) objective. A distributed controller approach is presented in [34], where it is assumed that the attitude of the satellites can be controlled to apply continuous low-thrust in the appropriate direction. Regardless of where the control authority is derived from (e.g., differential drag commands or thrust commands), these constellation acquisition methods require accurate attitude pointing.

In the subsequent *operational stage*, the satellites must perform various scheduled tasks including targeted remote sensing (e.g., imaging, radiometry), downlinking/uplinking data to/from ground stations, orbital station-keeping and other maintenance activities. In [25]-[28], an automated scheduler is developed to run autonomously on either ground stations (with schedules uplinked to satellites) or onboard in a distributed approach. Based on dynamic programming or mixed integer programming, the scheduler produces imaging schedules for each satellite that maximizes the number of observations and/or observation times for the constellation as a whole. In addition to maximizing spatial imaging coverage, [29] also addresses the problem of maximizing data downlinked by the constellation. We note that the schedulers in these works make inherent assumptions on the agility and pointing performance of the attitude control subsystem on board each satellite. For instance, in order to produce a feasible schedule that provides sufficient time to slew between desired orientations, the scheduler must know the dynamically-feasible, minimum slew time between any two desired orientations. This minimum slew time depends on the physical properties of the spacecraft (i.e., mass moment of inertia, actuator configuration) and actuator constraints (e.g., maximum rotor torque and momentum). Once an optimized schedule is produced for the constellation, each satellite must then generate and track an attitude trajectory that realizes its given schedule. Such an attitude trajectory may be optimized to minimize desired-actual attitude error at specific times of the trajectory and minimize control effort over the course of the trajectory.

Predicting future spacecraft will require agile attitude control systems that provide rapid multi-target pointing and tracking capabilities, [8] proposes a feedback regulator to conduct large-angle, rest-to-rest slew maneuvers using the Euler's eigenaxis rotation between any two orientations. Resembling feedback linearization, the proposed law introduces a nonlinear term to cancel out the coupling between body angular velocities and replaces it with linear error-quaternion and body rate feedback terms. In [35], a large class of attitude tracking control laws that have the general form of proportional-derivative (PD) feedback and feedforward compensation is obtained with proofs of global asymptotic stability in the closed-loop. Minimum-time, rest-to-rest slew maneuvers for an inertially symmetric rigid spacecraft with independent three-axis controls are studied in [36], which shows that the optimal maneuver is not, in general, an eigenaxis rotation but one that includes significant nutation of the instantaneous axis of rotation. Furthermore, the structure of the optimal control is different for small and large reorientation angles. While [8], [35], and [36] focus on rigid body dynamics under ideal, body-fixed control torques, [9] considers actuator dynamics and presents a feedback control logic that produces near minimum-time eigenaxis

slew maneuvers under actuator saturation, slew rate limit, and control bandwidth limit. The control laws are used in [37] to demonstrate rapid multi-ground-target acquisition by stepping through reference set-points that define successive scan trajectories. We note that the attitude control strategy in [25] also uses a minimum-time, eigenaxis slew maneuver and switches to a PD control law for small angles. Based on closed-loop simulations with this control strategy, a polynomial fit of minimum maneuver time as a function of the eigenaxis slew angle is used in a scheduler [26].

Although eigenaxis-based minimum-time control laws may be applied to general minimum-time problems to produce near-optimal maneuvers, they do not explicitly address nonlinearities that arise from, for instance, actuator dynamics nor do they explicitly consider general state and input constraints (e.g., bounds on momentum, power and energy). Moreover, when directly applying a feedback control strategy to track a desired sequence of discrete orientations, the feedback gains between each pair of orientations must be carefully tuned to achieve settling times that satisfy a desired pointing schedule, without missing any targets. In these minimum-time or minimum-attitude-error applications, we may use trajectory optimization methods to not only explicitly deal with nonlinearities and constraints, but to also automate the generation of continuous attitude trajectories for autonomous execution.

Main Contributions of Chapter

The main contributions of this chapter are the formulation and application of two optimal control problems (OCPs):

1. An off-line method to produce accurate estimates of the optimal slew time and required energy for a reference satellite to conduct a rest-to-rest maneuver between any two arbitrary orientations. The time and energy estimates can be used by a scheduler to produce dynamically-feasible pointing schedules for each satellite of a constellation. The corresponding problem is called the *Minimum-Time Slew OCP*.

2. An on-line method to produce continuous attitude trajectories that satisfy desired multi-target pointing schedules with minimum control effort. This method may be run on-board to generate trajectories for a low-level tracking controller to then follow. The corresponding problem is called the *Minimum-Effort Multi-Target Pointing OCP*.

These two contributions can be applied together to support the scheduler [38]. The first contribution allows a constellation scheduler to make informed multi-target pointing decisions (i.e., pointing schedules) based on an accurate assessment of a spacecraft's maneuvering capabilities. The second contribution allows a spacecraft to plan a continuous attitude trajectory that feasibly satisfies a desired pointing schedule generated by the constellation scheduler.

Preliminaries

A satellite with a reaction wheel based attitude determination and control system (ADCS) can be modeled as a gyrostat [39], consisting of a rigid body (i.e., spacecraft bus) encasing rotors that are in a fixed orientation. The state vector for this system of rigid bodies may be represented as:

$$\mathbf{x} = [\mathbf{q}^\top \;\; \boldsymbol{\omega}^\top \;\; \boldsymbol{\rho_r}^\top]^\top = [q_1 \;\; q_2 \;\; q_3 \;\; q_4 \;\; \omega_x \;\; \omega_y \;\; \omega_z \;\; \rho_{r_1} \;\; \rho_{r_2} \;\; \rho_{r_3} \;\; \rho_{r_4}]^\top \qquad (3.1)$$

where $\mathbf{q} \in \mathbb{R}^4$ is the unit quaternion that describes the orientation of the spacecraft's body-fixed frame with respect to an inertial frame, $\boldsymbol{\omega} \in \mathbb{R}^3$ is the angular velocity vector of the spacecraft with components expressed about the body-fixed frame, and $\boldsymbol{\rho_r} \in \mathbb{R}^4$ represents the angular momentum of each spinning rotor about its axis of rotation. Note that we assume the spacecraft has four actuation rotors. The input vector consists of the torques produced by the rotors about their axes:

$$\mathbf{u} = \boldsymbol{\tau_r} = [\tau_{r_1} \;\; \tau_{r_2} \;\; \tau_{r_3} \;\; \tau_{r_4}]^\top \qquad (3.2)$$

The differential equations describing the motion of this 11-state, 4-input system consist of the quaternion kinematics, gyrostat equation, and a single-integrator model for the rotors:

$$
\begin{aligned}
\dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \\
&= \begin{bmatrix} \mathbf{f_q}\left(\mathbf{q}(t), \boldsymbol{\omega}(t)\right) \\ \mathbf{f_\omega}\left(\boldsymbol{\omega}(t), \boldsymbol{\rho_r}(t), \boldsymbol{\tau_r}(t)\right) \\ \mathbf{f_{\rho_r}}\left(\boldsymbol{\tau_r}(t)\right) \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\mathbf{q}(t) \begin{bmatrix} \boldsymbol{\omega}(t) \\ 0 \end{bmatrix} \\ -J^{-1}\left(\boldsymbol{\omega}(t) \times (J\boldsymbol{\omega}(t) + A_r\boldsymbol{\rho_r}(t)) + A_r\boldsymbol{\tau_r}(t)\right) \\ \boldsymbol{\tau_r}(t) \end{bmatrix}
\end{aligned} \qquad (3.3)
$$

The positive definite matrix J represents the mass moment of inertia of the spacecraft in the body-fixed frame and $A_r \in \mathbb{R}^{3\times4}$ is the actuator Jacobian, where each column represents a rotor's axis of rotation with respect to the body-fixed frame. We model our reference

Parameter	Value	Units	Description
m	110	[kg]	Satellite mass
$l \times w \times h$	$60 \times 60 \times 95$	[cm]	Satellite dimensions (cuboid)
J	$\begin{bmatrix} 8.5 & 0.0 & 0.0 \\ 0.0 & 8.5 & 0.0 \\ 0.0 & 0.0 & 6.0 \end{bmatrix}$	[kg m^2]	Satellite inertia matrix
J_r	0.01	[kg m^2]	Rotor inertia
$\rho_{\mathbf{r}}^{max}$	0.80	[Nms]	Maximum rotor momentum
$\tau_{\mathbf{r}}^{max}$	0.06	[Nm]	Maximum rotor torque
A_r	$\begin{bmatrix} -0.68 & 0.68 & 0.68 & -0.68 \\ -0.68 & -0.68 & 0.68 & 0.68 \\ 0.26 & 0.26 & 0.26 & 0.26 \end{bmatrix}$	[ ]	Actuator Jacobian (4 rotors)

Table 3.1: Reference satellite parameters

satellite based on the physical parameters listed in Table 3.1, representative of Planet's Skysat, a satellite capable of agile maneuvering and imaging with sub-meter resolution [40]. Note that we assume a pyramid actuator Jacobian that has the torque envelope illustrated in Fig. B.2, with normalized units.

3.2 Minimum-Time Slewing

We formulate the *Minimum-Time Slew OCP* for a gyrostat to perform a time-optimal, large-angle, rest-to-rest slew maneuver:

$$\underset{\boldsymbol{\tau}_{\mathbf{r}}, t_f}{\text{minimize}} \quad \int_{t_0}^{t_f} 1 \ dt \tag{3.4}$$

subject to:

$$\dot{\mathbf{q}}(t) = \frac{1}{2}\mathbf{q}(t) \begin{bmatrix} \boldsymbol{\omega}(t) \\ 0 \end{bmatrix} \qquad\qquad \forall \ t \in [t_0, t_f] \tag{3.5}$$

$$\dot{\boldsymbol{\omega}}(t) = -J^{-1}\big(\boldsymbol{\omega}(t) \times (J\boldsymbol{\omega}(t) + A_r\boldsymbol{\rho}_{\mathbf{r}}(t)) + A_r\boldsymbol{\tau}_{\mathbf{r}}(t)\big) \qquad \forall \ t \in [t_0, t_f] \tag{3.6}$$

$$\dot{\boldsymbol{\rho}}_{\mathbf{r}}(t) = \boldsymbol{\tau}_{\mathbf{r}}(t) \qquad\qquad \forall \ t \in [t_0, t_f] \tag{3.7}$$

$$\|\boldsymbol{\omega}(t)\|_2 < \omega^{max} \qquad\qquad \forall \ t \in [t_0, t_f] \tag{3.8}$$

$$\|\boldsymbol{\rho}_{\mathbf{r}}(t)\|_\infty < \rho_{\mathbf{r}}^{max} \qquad\qquad \forall \ t \in [t_0, t_f] \tag{3.9}$$

$$\|\boldsymbol{\tau}_{\mathbf{r}}(t)\|_\infty < \tau_{\mathbf{r}}^{max} \qquad\qquad \forall \ t \in [t_0, t_f] \tag{3.10}$$

$$\mathbf{q}(t_0) = \mathbf{q}_{initial} \ , \quad \boldsymbol{\omega}(t_0) = \boldsymbol{\omega}_{initial} \ , \quad \boldsymbol{\rho}_{\mathbf{r}}(t_0) = \boldsymbol{\rho}_{\mathbf{r}_{initial}} \tag{3.11}$$

$$\mathbf{q}(t_f) = \mathbf{q}_{final} \ , \quad \boldsymbol{\omega}(t_f) = \boldsymbol{\omega}_{final} \tag{3.12}$$

The integral term in (3.4) captures the minimum-time objective. By following the time normalization procedure in Sec. 2.3, we note that the objective function can be written as:

$$J_{obj}\left(\mathbf{x}, \mathbf{u}, t_f\right) = \int_{t_0}^{t_f} 1 \ dt = \int_0^1 t_f \ d\tau = t_f \tag{3.13}$$

The quaternion kinematics are represented by (3.5) and the gyrostat dynamics are included in (3.6). The gyrostat dynamics may be modified to include other actuators, such as magnetorquers and propulsive thrusters. We may also include modeled environmental disturbances, such as moments due to gravity gradient, atmospheric drag, solar radiation pressure, and the magnetic field of Earth. Furthermore, the actuator dynamics in (3.7) may use higher-fidelity models that consider, for example, brushless DC motor dynamics and rotational friction. The quaternion kinematics and the gyrostat dynamics are non-convex in the state and input variables, however, they may be linearized using the approach described in Sec. (2.3). The angular velocity slew rate is bounded using (3.8). This constraint may be included when, for example, sensors such as star trackers limit how fast the spacecraft body can slew while maintaining accurate sensor measurements. Bounds on the maximum rotor momentum and

rotor torque are enforced by (3.9) and (3.10), respectively. Finally, the initial and final conditions are given in (3.11) - (3.12).

The Minimum-Time Slew OCP may be transcribed and solved using the sequential convex programming (SCP) method of Sec. 2.3. We note that the bounds on the state and input variables are convex. However, the constraints are approximately enforced since they are applied only at the temporal nodes of the discretization.

Example Application

A target-sensing or data-downlinking schedule prescribes a sequence of desired instrument and/or antenna pointing directions at specified times. To plan a schedule for each satellite in the constellation, the scheduler must consider the time it takes for the satellite to slew from one orientation to another. For example, if the angular displacement between two consecutively scheduled targets on the Earth's surface is large, it may not be feasible for the satellite to slew from one pointing orientation to another in a given amount of time. The slew time depends not only on a satellite's orbital motion and the relative distance between targets, but also on the spacecraft's mass moment of inertia and its actuator specifications. To our knowledge, a closed-form expression for computing minimum slew times between any arbitrary orientations does not exist (even for a symmetric rigid body with independent three-axis control).

Our approach is to apply the Minimum-Time Slew OCP over a gridded space of 3D rotations. Since an attitude maneuver between any two arbitrary orientations can be summarized by a single rotation about some *eigenaxis*, we parameterize the rotations using the Euler axis-angle attitude parameterization. As illustrated in Fig. 3.1, we consider 100 equidistributed axes of rotation and 88 rotations of increasing magnitude: $\theta \in$ (-180, +180] degrees to cover a relatively fine grid of the entire space of 3D rotations, resulting in 8,800 unique Euler axis-angle tuples. We set up 8,800 instances of the Minimum-Time Slew OCP, where we set the initial value of the quaternion in (3.11) to be the identity quaternion (i.e., $\mathbf{q}_{initial} = \mathbf{q}^I$) and the final desired quaternion in (3.12) to be the quaternion representation of an Euler axis-angle rotation. The initial and final body angular velocities are set to zero for a rest-to-rest maneuver. The initial rotor momenta are set to zero and the final rotor momenta are free variables. Using the sequential convex programming method, we efficiently solve each problem instance off-line with a computation time on the order of seconds. More efficient SCP implementations [20] may solve each problem in milliseconds and the overall procedure can be parallelized for even faster off-line computation. By recording the minimum maneuver time and ADCS energy consumption of each solved problem instance, we can produce lookup tables or data-fitted functions to be used by a scheduler.

Results

To illustrate the Minimum-Time Slew OCP, we study the solution for a large-angle (60-degree) slew maneuver about an arbitrary non-principal axis. Using the trajectory opti-
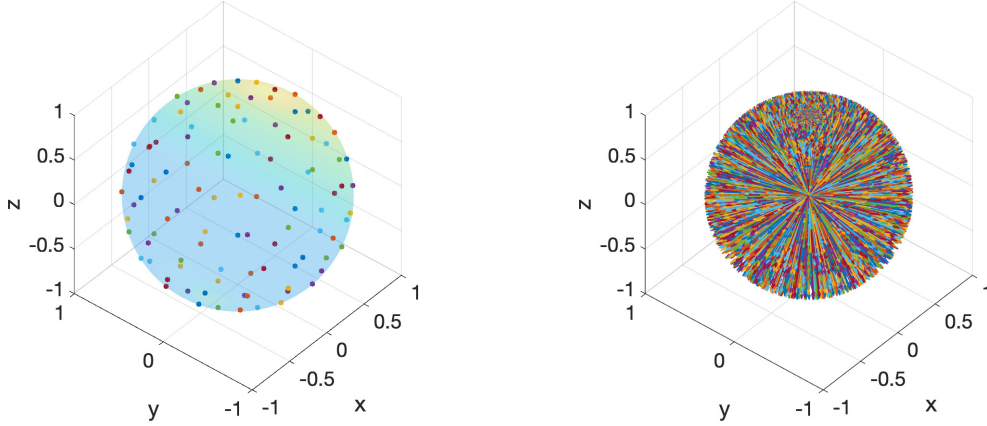
Figure 3.1: Rotation axes represented by equidistributed points on unit sphere and depiction of 8,800 unique rotations of a pointing vector.

mization method described in Sec. 2.3, we start the SCP Algorithm (1) with a crude initial guess $\mathbf{S}^0$, as represented in (2.140), that consists of a spherical linear interpolated [41] trajectory between desired quaternions, zero body angular velocity, zero rotor momenta, and zero rotor torque. We recall that our reference satellite is modeled after Planet's Skysat with physical parameters listed in Table 3.1. In Table 3.2, we note that the Minimum-Time Slew OCP is transcribed into an OPT using $K = 30$ discretization nodes (2.103). We choose a relatively small number of nodes for the Minimum-Time Slew OPT since we expect that it is sufficient for the short time scales in this class of problems. That is, a minimum-time slew between any two orientations should be on the order of seconds.

	Problem Formulation		
OPT	K	γ	ρ
Min-Time ($\theta = 60°$)	30	-	-
Min-Error, Mult-Pt	601	1e5	1e0
(Eqn No.)	(2.103)	(3.14)	(3.14)

Table 3.2: Objective function parameters for Attitude OCPs

In Table 3.3, we list the parameters used to decide SCP algorithm convergence (2.141). We note that the algorithm converges on a solution for the 60-degree minimum-time slew OCP in ten iterations with a run-time of 20.3 seconds. For each of the 8,800 instances of the Minimum-Time Slew OCP, we find a solution before reaching the maximum number of iterations, $N_{max} = 20$. The problems are written in MATLAB and solved using CVX, a package for specifying and solving convex programs [42], on a 2015 MacBook Pro with 2.5 GHz Intel Core i7 processor.

	SCP Algorithm					Convergence Results	
OPT	N_{max}	w_{vc}	w_{tr}	ϵ_{vc}	ϵ_{tr}	$N_{converge}$	$t_{converge}$
Min-Time ($\theta = 60°$)	20	1e5	1e-1	1e-5	1e-5	10	20.3 sec
Min-Error, Mult-Pt	20	1e5	1e-1	1e-3	1e-4	9	144.6 sec
(Eqn No.)	Algo (1)	(2.115)	(2.116)	(2.141)	(2.141)	Algo (1)	-

Table 3.3: SCP algorithm parameters and convergence results for Attitude OCPs

In Fig. 3.2, we illustrate the spacecraft's motion where the blue vector represents the body z-axis, aligned with an instrument's pointing vector. The trajectories traced out by the body axes represent the minimum-time slew maneuver to achieve the desired reorientation. We qualitatively note that this maneuver about an arbitrary non-principal axis does not resemble an eigenaxis slew. However, for slews about any of the principal axes, we confirm that the minimum-time maneuver is an eigenaxis slew aligned with the principal axis.
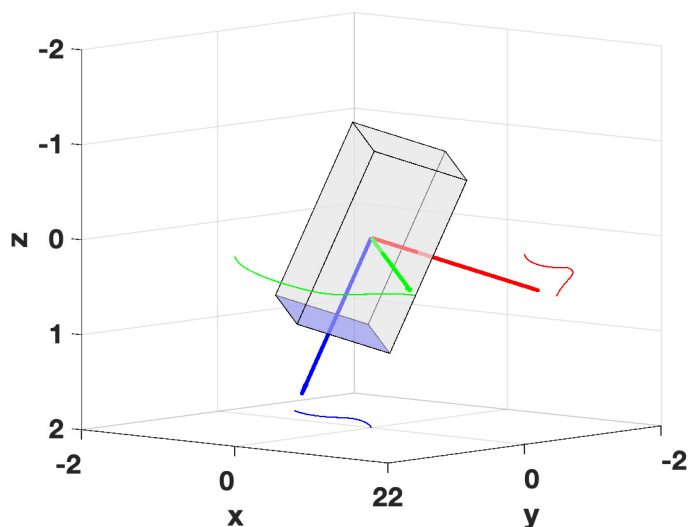

Figure 3.2: Minimum-time slew maneuver (60-degree rotation about an arbitrary axis)

We characterize the solution for this particular problem instance using five quantities: maneuver time, rotor momenta, rotor torque, ADCS instantaneous power drawn and ADCS cumulative energy consumption. In Fig. 3.3, we note that the quaternion trajectory starts from a nominal orientation (represented by the identity quaternion) and reaches the desired terminal nodes, corresponding to the desired Euler axis-and-angle rotation. Figure 3.4 shows that the body starts and ends with zero angular velocity, representing a rest-to-rest slew maneuver. The angular speed is greatest about the body z-axis, confirming the predominantly yaw-axis rotation depicted in Fig. 3.2.
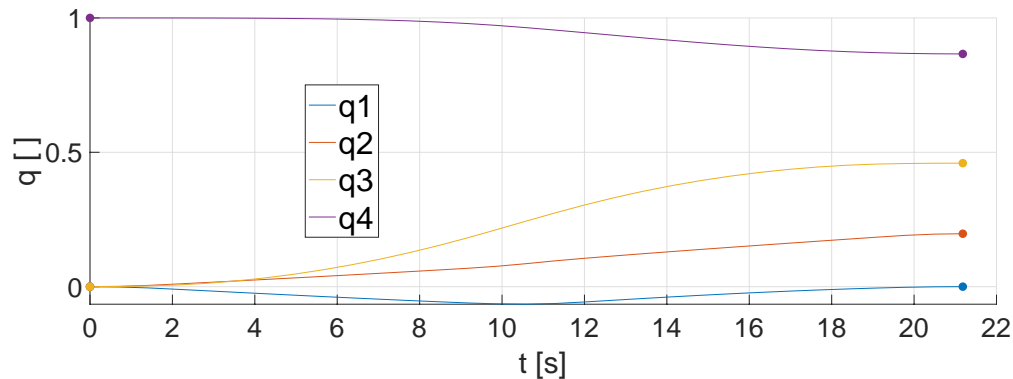
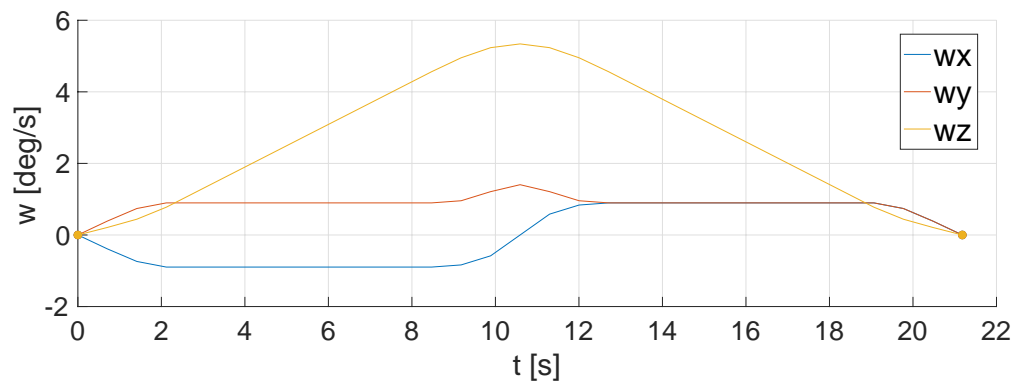Figure 3.3: Quaternion trajectory for minimum-time slew

Figure 3.4: Body angular velocity for minimum-time slew

Figure 3.5 shows that the rotor momenta do not saturate for this 60-degree maneuver. For larger rotation magnitudes, we expect that the momentum trajectories may approach the magnitude bound of 0.80 Nms, impacting maneuver time. We note the non-zero rotor momenta at the end of the maneuver caused by the input torques. In practice, momentum will also build up in the rotors as they counteract torques due to environmental disturbances. This momentum buildup may be dumped with magnetorquers or other compensating mechanisms. In Fig. 3.6, we observe that the torque from each rotor saturates at 0.06 Nm and follows a "bang-bang" structure, consistent with classical minimum-time maneuvers [43]-[44]. This observation confirms our intuition that performing a rotation in minimum time requires the actuators to perform at their torque limits.

In Fig. 3.7, we note that the peak instantaneous power drawn by the attitude determination and control system (ADCS) stays below 15 W and in Fig. 3.8 we observe that the total energy consumption is less than 150 J for the maneuver. Given mission constraints on the instantaneous power or energy consumption, we may enforce constraints (3.27) and (3.28) in our problem formulation.

The optimal value associated with this instance of the Minimum-Time Slew OCP is 21.2
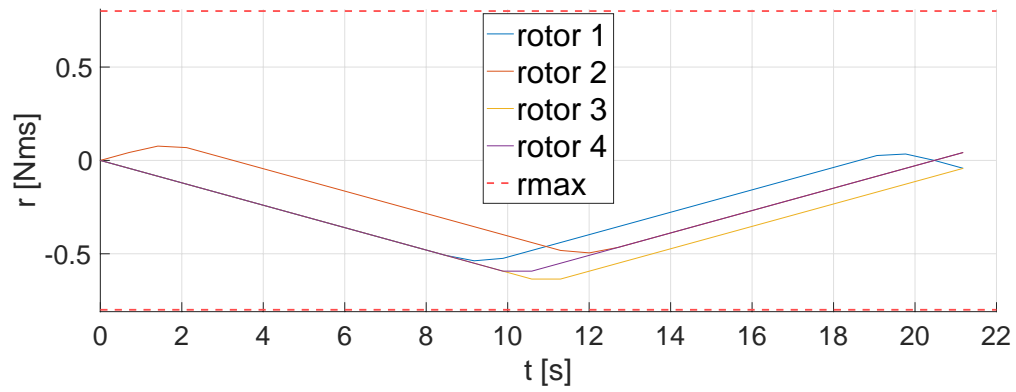
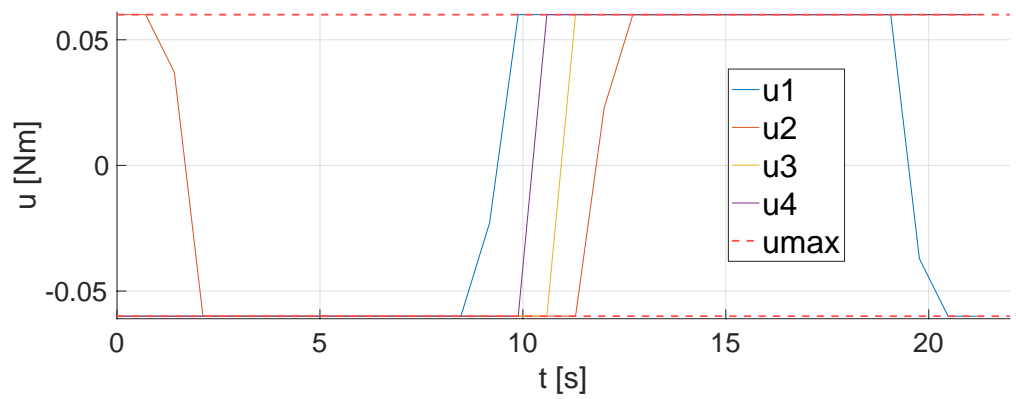Figure 3.5: Rotor momenta for minimum-time slew
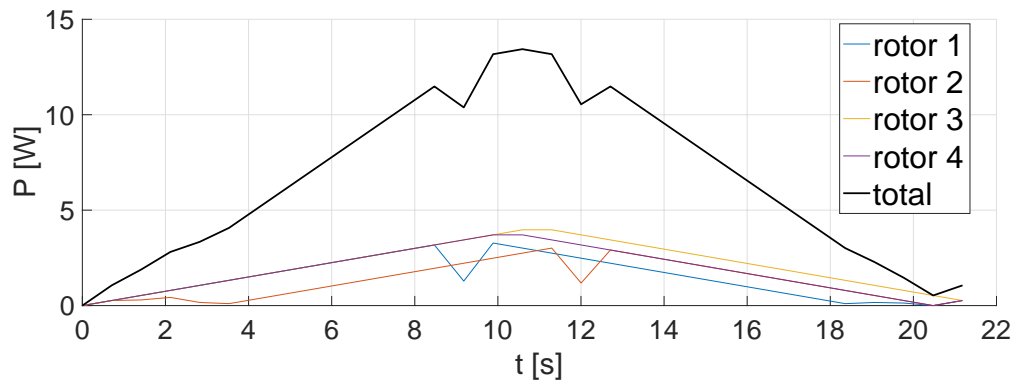

Figure 3.6: Rotor torque for minimum-time slew

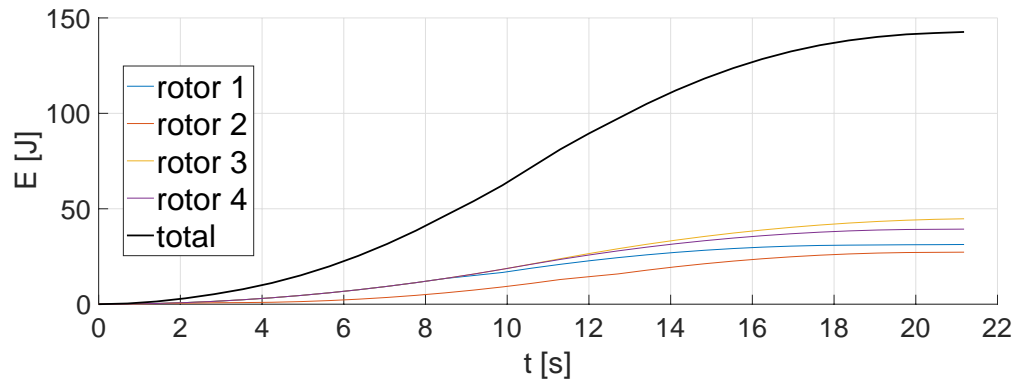Figure 3.7: ADCS instantaneous power drawn for minimum-time slew

Figure 3.8: ADCS energy consumption for minimum-time slew

seconds. That is, we expect that it requires at least 21.2 seconds to conduct this 60-degree, rest-to-rest slew maneuver. We expect that a smaller spacecraft with less mass moment of inertia, but the same actuator specifications, requires less maneuver time to conduct the same reorientation. Hence, the trend towards small satellites not only results in cost efficiency but also more nimble operation.

We perform the same trajectory optimization for each of the 8,800 problem instances described in the Example Application. After solving each problem instance, we plot the corresponding slew maneuver time and ADCS energy consumption as functions of the rotation magnitude in Fig. 3.9 and Fig. 3.10, respectively. For each of the 100 equidistributed axes of rotation, we observe that the optimal maneuver time is a power function of the rotation magnitude. We also observe that the energy consumed to conduct a minimum-time maneuver is a linear function of the rotation magnitude. These relationships allow us to approximate the fastest maneuver time and the required energy to conduct any arbitrary rotation by the reference satellite described in Table 3.1. The relationships may be represented in lookup tables or function approximators that can be quickly evaluated by an on-line constellation scheduler.

To empirically verify the optimality of the solutions found by our approach, we solve the Minimum-Time Slew OPT for rotations about the principal axes (i.e., about body x-, y-, and z-axes). Using least-squares fitting, we determine the coefficients for a minimum-time power model, $T = a\theta^b$, and an energy linear model, $E = c\theta + d$, valid for eigenaxis rotations $|\theta| > 1°$ about each axis. As shown in Table 3.4, the power coefficient in the minimum-time model approximately takes on the value of $b = 0.5$ for principal axis rotations. Hence, we observe that the optimal maneuver time for a principal axis rotation is proportional to the square-root of the rotation magnitude (in radians). We note that the well-known minimum-time, rest-to-rest, double-integrator problem also reveals this relationship, as mentioned in [36] and proved in [43] - [44]. Intuitively, we may infer that a minimum-time maneuver about a principal axis is equivalent to the minimum-time eigenaxis slew, which in turn can be modeled as the double-integrator problem. Since the maneuver is purely about a principal
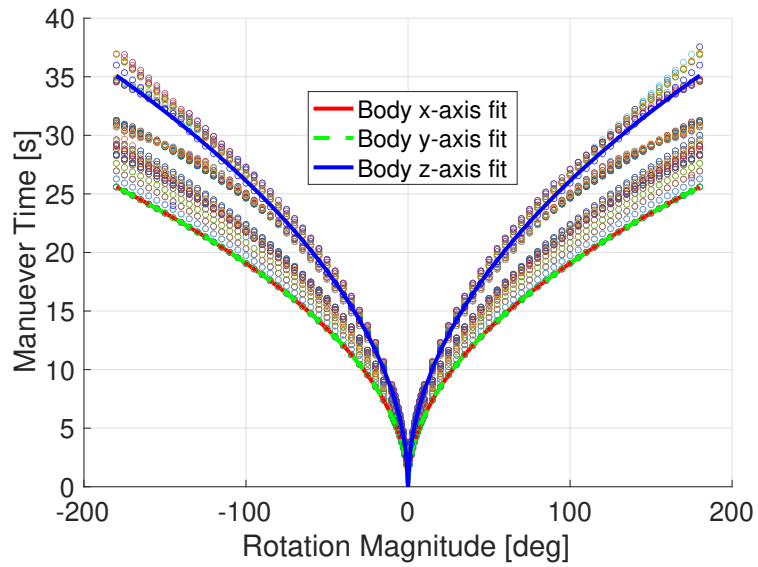
Figure 3.9: Slew time as a function of eigenaxis rotation magnitude θ
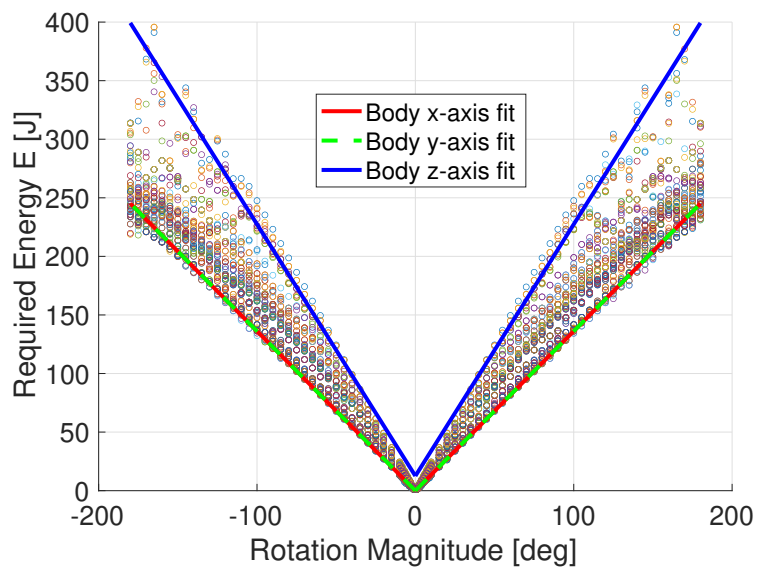
Figure 3.10: Energy as a function of eigenaxis rotation magnitude θ

axis, the body angular velocities are decoupled, simplifying the rotational dynamics into a double-integrator system from torque input to rotation angle. This result provides assurance that our method is indeed finding approximately optimal values and solutions.

Body Axis	a	b	c	d
x	14.4371	0.5000	78.0329	-0.0225
y	14.4371	0.5000	78.0329	-0.0225
z	19.7292	0.5033	123.1154	12.5189

Table 3.4: Minimum-time and energy model coefficients (valid for $\theta > 1°$)

Beyond double-integrator dynamics about principal axes, our approach can be applied to high-fidelity spacecraft models and arbitrary rotations. Since the Minimum-Time Slew OCP can be applied to non-rest-to-rest and non-principal axis rotations, the optimal slew times found by our approach may be more accurate than those based on the classical rest-to-rest eigenaxis slew solution. Furthermore, we may explicitly consider constraints that are not included in the classical formulation (e.g., bounds on body slew rate or rotor momenta).

3.3 Minimum-Effort Multi-Target Pointing

We formulate the *Minimum-Effort Multi-Target Pointing OCP* for a gyrostat to achieve a multi-target pointing schedule with minimum control effort:

$$\underset{\boldsymbol{\tau_r}}{\text{minimize}} \quad \sum_{k\in\mathbb{K}}\left\{\left\|\bar{\mathbf{q}}^+(t_k)\mathbf{q}(t_k) - \mathbf{q}^{\mathrm{I}}\right\|_2 + \gamma\left\|\boldsymbol{\omega}(t_k) - \bar{\boldsymbol{\omega}}(t_k)\right\|_2\right\} + \rho\int_{t_0}^{t_f}\left\|\boldsymbol{\tau_r}(t)\right\|_2^2 dt \quad (3.14)$$

subject to:

$$\dot{\mathbf{q}}(t) = \frac{1}{2}\mathbf{q}(t)\begin{bmatrix}\boldsymbol{\omega}(t)\\0\end{bmatrix} \qquad\qquad \forall\, t\in[t_0,t_f] \quad (3.15)$$

$$\dot{\boldsymbol{\omega}}(t) = -J^{-1}\big(\boldsymbol{\omega}(t)\times(J\boldsymbol{\omega}(t) + A_r\boldsymbol{\rho_r}(t)) + A_r\boldsymbol{\tau_r}(t)\big) \qquad \forall\, t\in[t_0,t_f] \quad (3.16)$$

$$\dot{\boldsymbol{\rho_r}}(t) = \boldsymbol{\tau_r}(t) \qquad\qquad \forall\, t\in[t_0,t_f] \quad (3.17)$$

$$\|\boldsymbol{\omega}(t)\|_2 < \omega^{max} \qquad\qquad \forall\, t\in[t_0,t_f] \quad (3.18)$$

$$\|\boldsymbol{\rho_r}(t)\|_\infty < \rho_{\mathbf{r}}^{max} \qquad\qquad \forall\, t\in[t_0,t_f] \quad (3.19)$$

$$\|\boldsymbol{\tau_r}(t)\|_\infty < \tau_{\mathbf{r}}^{max} \qquad\qquad \forall\, t\in[t_0,t_f] \quad (3.20)$$

$$\mathbf{q}(t_0) = \mathbf{q}_{initial}\,,\quad \boldsymbol{\omega}(t_0) = \boldsymbol{\omega}_{initial}\,,\quad \boldsymbol{\rho_r}(t_0) = \boldsymbol{\rho_{\mathbf{r}}}_{initial} \qquad (3.21)$$

$$\mathbf{q}(t_f) = \mathbf{q}_{final}\,,\quad \boldsymbol{\omega}(t_f) = \boldsymbol{\omega}_{final} \qquad (3.22)$$

In this problem formulation we maintain the same constraints as those in Sec. 3.2. The final time t_f is no longer a decision variable but a fixed parameter. Given a schedule (i.e., sequence) of desired quaternions and angular velocities at specified times: $\{t_k, \bar{\mathbf{q}}(t_k), \bar{\boldsymbol{\omega}}(t_k)\}\ \forall\ k\in$

$\mathbb{K}$, where $\mathbb{K}$ is a finite set of indices, the objective in (3.14) minimizes the error at those discrete time points while also minimizing the continuous control effort. The control effort is represented by the the energy of the input signal, defined as $\int_{t_0}^{t_f} \|\mathbf{u}(t)\|_2^2 \, dt$. The definite integral term representing the control effort is approximated with the trapezoidal rule:

$$\int_{t_0=0}^{t_f} \|\boldsymbol{\tau}_{\mathbf{r}}(t)\|_2^2 \, dt = \int_0^1 t_f \, \|\boldsymbol{\tau}_{\mathbf{r}}(\tau)\|_2^2 \, d\tau \tag{3.23}$$

$$\approx \frac{t_f}{K-1} \left(\sum_{k=2}^{K-1} \|\boldsymbol{\tau}_{\mathbf{r}}[k]\|_2^2 + \frac{\|\boldsymbol{\tau}_{\mathbf{r}}[1]\|_2^2 + \|\boldsymbol{\tau}_{\mathbf{r}}[K]\|_2^2}{2} \right) \tag{3.24}$$

The user-defined parameter $\gamma > 0$ weighs the angular velocity error relative to the quaternion error while $\rho > 0$ weighs the control penalty term relative to the total attitude error.

We note that an alternative "constraint formulation" of the *Minimum-Effort Multi-Target Pointing OCP* would remove the attitude penalty terms in the objective and implement them as constraints:

$$\left\| \bar{\mathbf{q}}^+(t_k)\mathbf{q}(t_k) - \mathbf{q}^{\mathrm{I}} \right\|_2 \leq \epsilon_{\mathbf{q}} \qquad \forall \, k \in \mathbb{K} \tag{3.25}$$

$$\left\| \boldsymbol{\omega}(t_k) - \bar{\boldsymbol{\omega}}(t_k) \right\|_2 \leq \epsilon_{\boldsymbol{\omega}} \qquad \forall \, k \in \mathbb{K} \tag{3.26}$$

where $\epsilon_{\mathbf{q}} \geq 0$ and $\epsilon_{\boldsymbol{\omega}} \geq 0$ are user-defined error tolerances. However, if the error tolerance values are set too tight for a given attitude schedule, then the problem may be infeasible. Hence, when specific error tolerance values are not required, we may choose to solve the original "penalty formulation" of the problem.

Additional constraints that we may include in the problem formulations are bounds on the maximum instantaneous power drawn P_{max} and maximum energy consumed E_{max} by the attitude control system (i.e., all four rotors combined):

$$\sum_{i=1}^{4} \left| \tau_{\mathbf{r}_i}(t) \cdot \frac{1}{J_r} \rho_{\mathbf{r}_i}(t) \right| < P_{max} \qquad \forall \, t \in [t_0, t_f] \tag{3.27}$$

$$\int_{t_0}^{t_f} \left\{ \sum_{i=1}^{4} \left| \tau_{\mathbf{r}_i}(t) \cdot \frac{1}{J_r} \rho_{\mathbf{r}_i}(t) \right| \right\} dt < E_{max} \tag{3.28}$$

where J_r is the rotor inertia. Apart from the equations of motion, we note that the instantaneous power and energy constraints are non-convex due to being bilinear in the decision variables of $\boldsymbol{\tau}_{\mathbf{r}}$ and $\boldsymbol{\rho}_{\mathbf{r}}$.

Example Application

Given a desired attitude pointing schedule $\{t_k, \bar{\mathbf{q}}(t_k), \bar{\boldsymbol{\omega}}(t_k)\} \, \forall \, k \in \mathbb{K}$, we use the Minimum-Effort Multi-Target Pointing OCP to plan a continuous trajectory that passes through each attitude point in the sequence. As an example, we consider the remote-sensing application

described in [26], where a 24-satellite, 3-plane Walker-Delta constellation is simulated in 710 km altitude, 98.5 deg inclined, circular sun-synchronous orbits (SSO) over a 6-hour duration with the orbital mechanics module of the D-SHIELD software suite [28]. Simulation results include the orbital states of the satellites at each time step as well as *access times* when user-defined target *grid points* (described by latitude and longitude coordinates) are observable by a satellite. In our example we consider 42 urban regions that experience frequent episodic precipitation and are prone to flooding [26]. Each of these globally-distributed *watersheds* covers an 80 km^2 area spanned by 121 grid points.
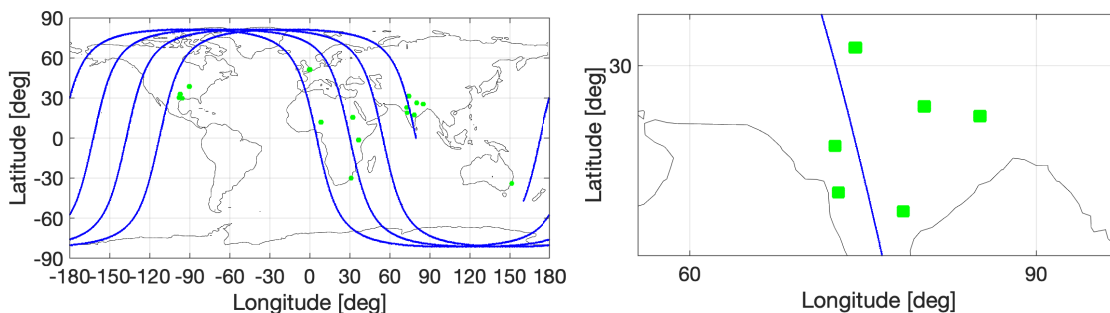

Figure 3.11: Groundtrack of a satellite in a 710km SSO orbit over 16 observable watersheds and a zoomed-in view of a cluster of 6 watersheds

We focus on a single satellite of the constellation and observe in Fig. 3.11 that the satellite can access 16 of the 42 watersheds during the 6-hour time span. Only a subset of the 42 watersheds are accessible due to either access restrictions, bounds on maximum off-pointing angles, or occlusion by the Earth's surface. Fig. 3.11 also provides a zoomed-in view of the groundtrack, showing a dense cluster of 6 regions with a total of 726 potentially accessible target grid points. As the satellite passes over this dense cluster, the satellite may be commanded by the scheduler to perform a rapid sequence of agile slewing maneuvers to acquire, point, and track desired target grid points. The commanded schedule during this part of the orbital trajectory may be the most difficult to execute by an attitude control system, requiring high-frequency intra-region slewing as well as large-angle, inter-region slewing. We design a pointing schedule that maximizes the number of observed grid points in this 6-region cluster, treating it as a stress test for the Minimum-Effort Multi-Target Pointing OCP.

From left to right, the plots in Fig. 3.12 show the satellite's orbit expressed in both the Earth-Centered Inertial (ECI) and Earth-Centered Earth-Fixed (ECEF) frames. The origin of the ECI frame is placed at the Earth's center and the frame remains fixed in inertial space. The satellite's orbital plane also appears fixed to a stationary observer while the Earth rotates beneath. The ECEF frame is also placed at the Earth's center but rotates with the Earth's rotation about its axis. To an observer rotating with the frame, points on the Earth's surface remain fixed but the satellite orbit appears to change. Plots using the ECI frame are useful for describing the motion of spacecraft in orbit while plots using the ECEF frame are useful for describing the position of points on the Earth's surface. In either plot, the red vectors

point from the satellite to green grid points on the Earth's surface. We note that the stress test focuses on a subset of the 16 accessible regions shown in Fig. 3.12 . At any given time, a desired *pointing frame* may be uniquely defined with its z-axis aligned with the red pointing vector, x-axis in the orbital plane towards the direction of motion, and y-axis completing the right-hand triad. The orientation of this frame with respect to the ECI frame is the desired pointing attitude to acquire. When a desired attitude is defined at a particular time instance, we may also define the desired body angular velocity to be zero for an interval about the specified time instance (e.g., if $\exists \ \bar{\mathbf{q}}(t_k)$, then $\{\bar{\boldsymbol{\omega}}(t_{k-1}), \bar{\boldsymbol{\omega}}(t_k), \bar{\boldsymbol{\omega}}(t_{k+1})\} = \mathbf{0}$). Specifying zero velocity at the time of target observation may mitigate the effects of motion blur while imaging.
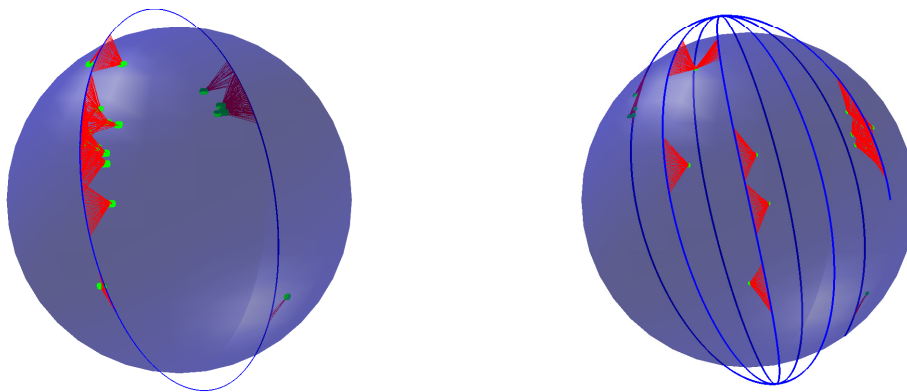

Figure 3.12: Satellite orbit expressed in ECl and ECEF coordinate systems with green urban regions on Earth surface and red pointing vectors

Using the following guidelines, we manually design an attitude pointing schedule (i.e., sequence of desired pointing orientations) to maximize grid point (GP) observations across the dense 6-region cluster:

- Include as many of the 726 GPs as feasibly possible (6 regions, 121 GPs per region).

- Visit each of the six regions (or as many as possible).

- Finish observing a region before moving to the next region (i.e., cannot return to a previously observed regions).

- Observe GPs using a "sweeping" pattern, similar to the whiskbroom pattern used by early Landsat satellites [45].

- The first accessible points in a region are observed first, deciding the direction of the intra-region, sweeping pattern.

- The first accessible regions are observed first, deciding the direction of the inter-region slewing.

As the satellite passes over the 6-region cluster, it has a ∼10-minute window from the time when one of the 726 GPs first becomes accessible to the time that none are accessible. From an altitude of 710km, we estimate that the satellite requires at least 2 seconds to conduct a minimum-time, rest-to-rest slew from a grid point in the nadir direction to an adjacent grid point 8 km away. We also estimate that it can require at least 15 seconds to slew the several hundreds of kilometers between certain pairs of regions in the cluster. Our slew time estimates, between neighboring GPs (intra-region slewing) and neighboring regions (inter-region slewing), are based on results gained from the Minimum-Time Slew OCP application of the previous section.
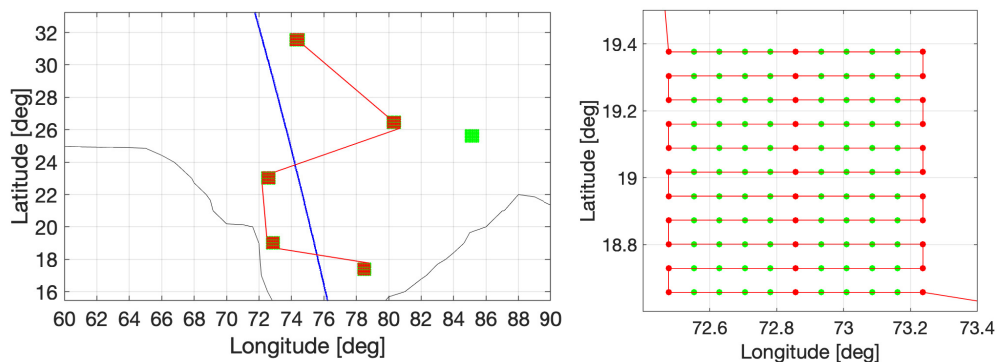

Figure 3.13: Inter-region slewing pattern visits 5 regions and intra-region slewing points at 33 GPs per region

The first plot of Fig. 3.13 shows a manually-designed schedule that visits five of the six regions. The second plot of Fig. 3.13 illustrates the "sweeping" pattern that is used to observe 33 GPs per region. A red marker signifies a GP chosen for observation and a green marker is a GP not chosen for observation. The red path illustrates the sequence in which the GPs are observed as well as the approximate trajectory of the satellite's pointing vector on the Earth's surface. With 33 GPs in five regions, the manually designed schedule covers a total of 165 GPs (or 22.7 percent of the total number of accessible GPs). We omit the region located near 25° lattitude and 85° longitude since its inclusion reduces the total number of GP observations, assuming the same number of GP observations per region. Given the 10-minute window and the slew time necessary between GPs and between regions, we find that our 165-GP pointing schedule is ambitious and close to the maximum number of grid points that a satellite may feasibly observe across the 6-region cluster. The other grid points must be observed on a subsequent revisit or by another satellite. Compared to this heuristics-based schedule, we expect that an optimization-based schedule could further maximize the number of GP observations. An optimization-based scheduler may explicitly use minimum-time estimates (found in the previous section) to optimize an attitude trajectory that maximizes the number of observations or the quality of measurements, which is a function of the look angle and slew rate at time of sensor measurement [28].

Results

We apply the Minimum-Effort Multi-Target Pointing OCP to the 165-GP attitude pointing schedule for the 6-region cluster example. In Table 3.2, we list the penalty weights used in the objective function (3.14) as well as the number of discretization nodes used to transcribe the OCP into an OPT. We use $K = 601$ nodes so that each node represents a second of the 10-minute observation schedule.

The SCP convergence parameters for the Minimum-Effort Multi-Target Pointing OPT are listed in 3.3. With 601 discretization nodes, the algorithm requires nine iterations and 144.6 seconds of run-time to converge to a solution. Figure 3.14 illustrates how the virtual control and trust region penalty terms for the Minimum-Effort Multi-Target Pointing OPT decrease below user-defined tolerance values by the 9th SCP iteration. The small value for J_{vc} signifies that the dynamics, state constraints and input constraints are satisfied using the input trajectory $\{\boldsymbol{\tau}_\mathbf{r}[k]\}_{k=1}^{K}$ and only a negligible contribution from virtual controls. Furthermore, the small value for J_{tr} shows that the solution at the 9th iteration has not changed significantly from that of the previous iteration. This signifies that the current solution is based on an accurate approximation of the nonlinear dynamics, linearized about the previous solution, and that negligible improvement may be found in further iterations.
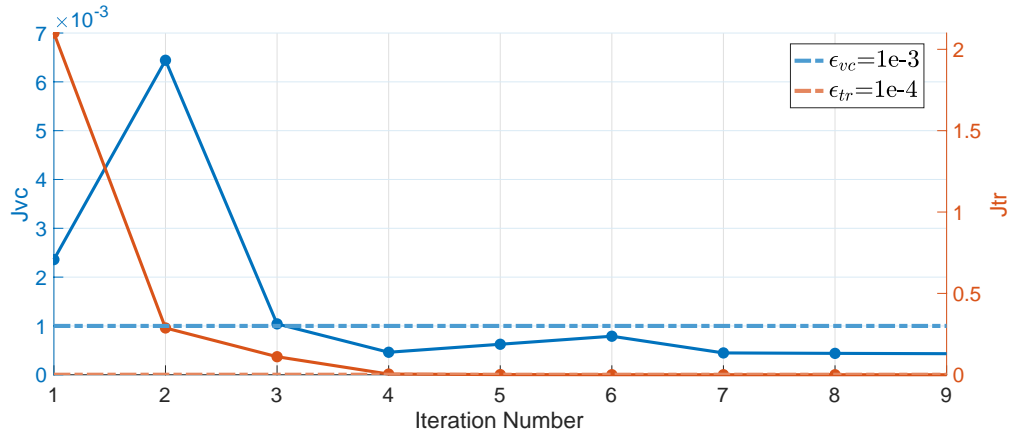

Figure 3.14: SCP algorithm convergence as Virtual Control J_{vc} and Trust Region J_{tr} penalty terms fall under tolerance values (ϵ_{vc}, ϵ_{tr})

We recall that the schedule is a discrete sequence of desired pointing orientations:

$$\{t_k, \bar{\mathbf{q}}(t_k), \bar{\boldsymbol{\omega}}(t_k)\} \ \forall \ k \in \mathbb{K}, \ \text{where} \ |\mathbb{K}| = 165$$

In Fig. 3.15, we represent the 165 desired quaternion points with un-filled, circular markers. The colored lines with filled markers represent a feasible, continuous quaternion trajectory that passes through all desired points at the specified times.

To quantify the error in the quaternion trajectory with respect to the desired schedule, we introduce performance metrics:
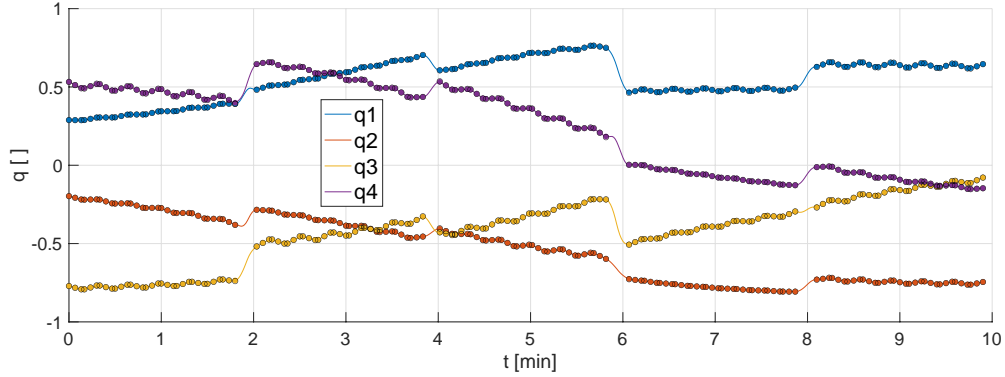
Figure 3.15: Quaternion trajectory for minimum-effort multi-target pointing maneuver
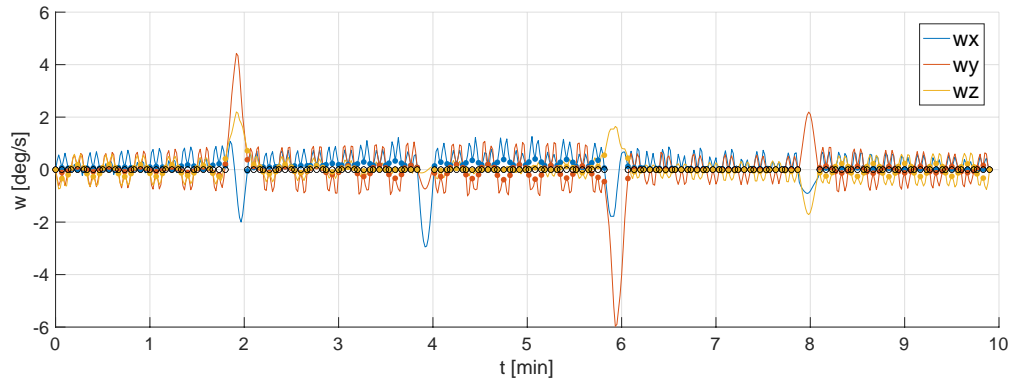

Figure 3.16: Body angular velocity for minimum-effort multi-target pointing maneuver

$$q_e^{max} = \max_{k \in \mathbb{K}} \; \left\| \bar{\mathbf{q}}_k^+ \mathbf{q}_k - \mathbf{q}^{\mathrm{I}} \right\|_2 \tag{3.29}$$

$$q_e^{avg} = \frac{1}{|\mathbb{K}|} \sum_{k \in \mathbb{K}} \; \left\| \bar{\mathbf{q}}_k^+ \mathbf{q}_k - \mathbf{q}^{\mathrm{I}} \right\|_2 \tag{3.30}$$

Equation (3.29) measures the largest point-wise error among the 165 observation points specified by the attitude schedule, whereas equation (3.30) measures the average error across all 165 points. As recorded in Table 3.5, we compute the maximum point-wise quaternion error to be $q_e^{max} = 0.0079$ and the average quaternion error to be $q_e^{avg} = 0.0014$.

Figure 3.16 shows the corresponding angular velocity trajectory and observe that the spacecraft reaches near-zero angular velocity at all 165 desired points. Similar to the performance metrics for the quaternion trajectory, we also introduce equations (3.31) and (3.32) to quantify the maximum point-wise error and the average error in the angular velocity trajectory with respect to the desired schedule. We record a maximum point-wise error of

Quaternion Error		Angular Velocity Error	
q_e^{max}	q_e^{avg}	ω_e^{max}	ω_e^{avg}
0.0079	0.0014	0.8135	0.1980

Table 3.5: Reference trajectory error with respect to desired GP schedule

$\omega_e^{max} = 0.8135$ and an average error of $\omega_e^{avg} = 0.1980$.

$$\omega_e^{max} = \max_{k \in \mathbb{K}} \ \|\boldsymbol{\omega}_k - \bar{\boldsymbol{\omega}}_k\|_2 \tag{3.31}$$

$$\omega_e^{avg} = \frac{1}{|\mathbb{K}|} \sum_{k \in \mathbb{K}} \ \|\boldsymbol{\omega}_k - \bar{\boldsymbol{\omega}}_k\|_2 \tag{3.32}$$

To further reduce the angular velocity error, we may increase the relative state penalty γ in the Minimum-Effort Multi-Target Pointing OCP objective (3.14). We may also choose to include an explicit constraint on the error using (3.26) and attempt to find a solution.

In Fig. 3.16 we also note that the angular velocity components peak when conducting large-angle slewing between the five regions. The large peaks are due to the limited time spans allotted for the inter-region slews specified in the schedule. Imposing body slew rate constraints would reduce the amplitude of the peaks but may also increase quaternion error, especially for the initial points in each region when high slew rates are required to move quickly from the previous region. A "relaxed" schedule that allows for more time to conduct inter-region slews could satisfy slew rate constraints with minimal quaternion error but at the expense of observing fewer GPs.

As shown in Fig. 3.17, the rotor momenta stay within the 0.8 Nms bounds and peak when conducting large angle maneuvers, such as slewing between regions. Despite the large distances between certain pairs of regions, the rotor momenta do not saturate. This observation suggests that the reference spacecraft is capable of very large-angle slew maneuvers before rotor momenta saturation.

Despite the control effort penalty in the Minimum-Effort Multi-Target Pointing objective (3.14), which encourages minimal control effort, we observe that the rotor torques saturate at the 0.06 Nm bounds in Fig. 3.18 in a "bang-bang," minimum-time fashion. This suggests that the spacecraft is working at its performance limits to achieve an aggressive pointing schedule. We expect actuator saturation to occur for schedules that maximize the number of observations in a finite amount of time.

The rotor momentum and torque trajectories inform us of the reference spacecraft's agile performance limits. Schedules requiring large-angle slew maneuvers may saturate rotor momentum bounds, limiting how far the spacecraft's pointing vector can span in a given amount of time. Schedules requiring rapid slewing between attitude points may saturate rotor torque bounds. Roughly speaking, for a given spacecraft design (i.e., mass moment-of-inertia, actuator specifications), rotor momentum and torque bounds dictate spatial range-of-motion and speed, respectively.
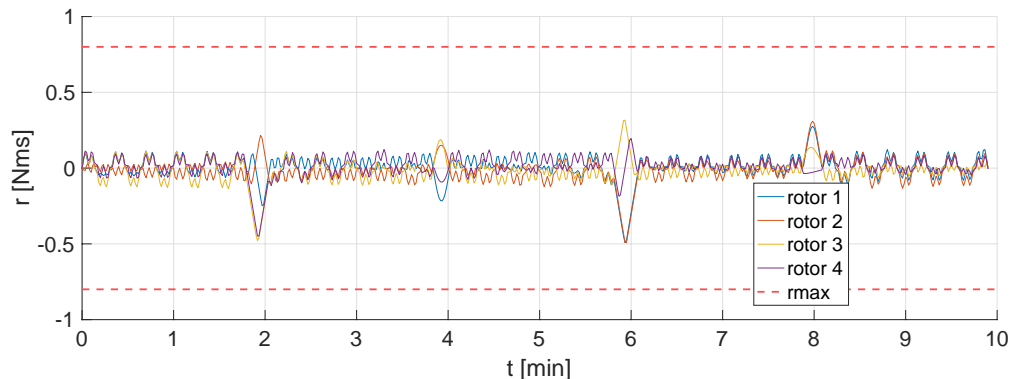
Figure 3.17: Rotor momenta for minimum-effort multi-target pointing maneuver
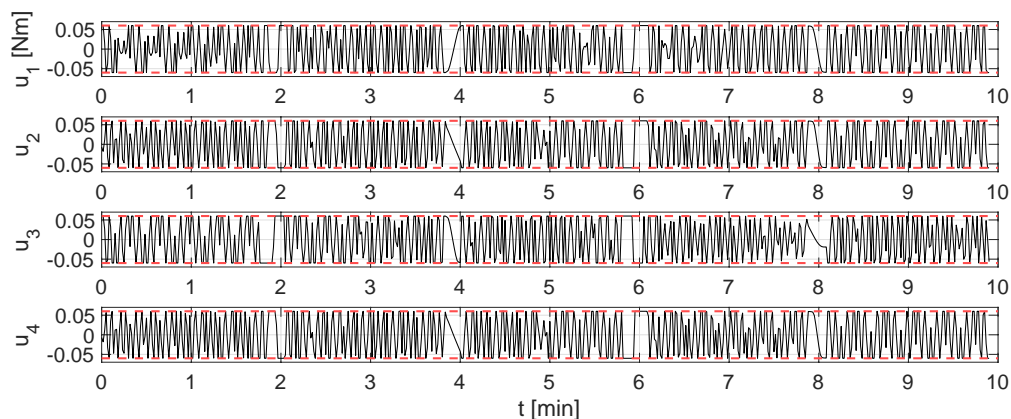

Figure 3.18: Rotor torque for minimum-effort multi-target pointing maneuver

Similar to the body angular velocity and rotor momentum trajectories, we observe in Fig. 3.19 that the total instantaneous power drawn by the rotors peaks when slewing between regions. We recall that if maximum power bounds must be enforced on the rotors, we may add constraint (3.27) in our problem formulation.

In Fig. 3.20, we observe that the ADCS requires approximately 550 J to perform the desired ten-minute, minimum-effort multi-target pointing trajectory. If this trajectory is too energetically expensive, we may increase the relative control effort weight ρ in the problem objective to further penalize energy consumption. However, the choice may come at the expense of increased attitude error. If energy consumption must be constrained, we may explicitly enforce it by adding (3.28) to the problem formulation. If we cannot find a feasible trajectory in the presence of the energy consumption constraint, the attitude pointing schedule may need to be relaxed to observe fewer regions.
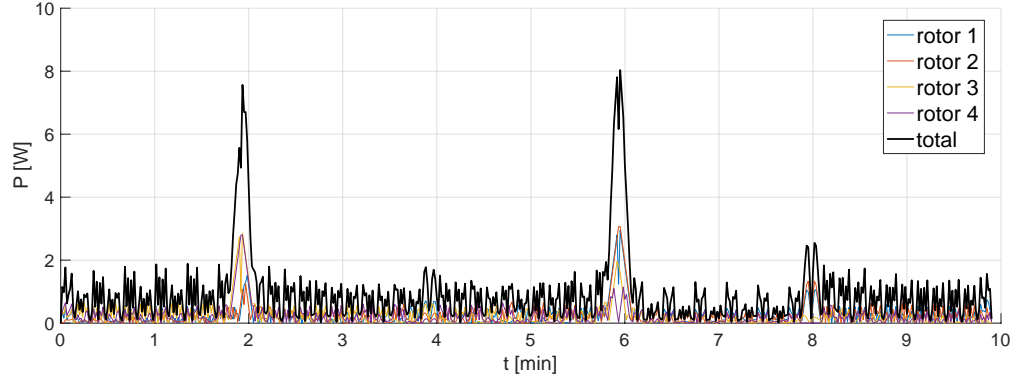
Figure 3.19: ADCS instantaneous power for minimum-effort multi-target pointing maneuver
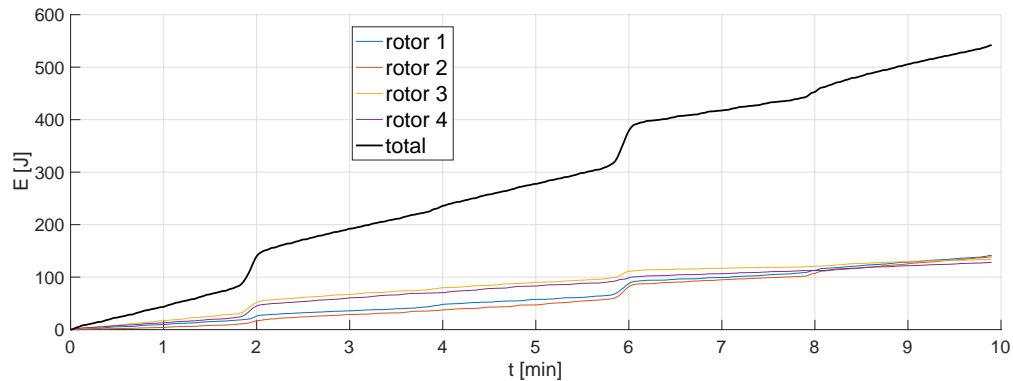

Figure 3.20: ADCS energy consumption for minimum-effort multi-target pointing maneuver

Open-loop versus Closed-loop Performance

With a solution to the Minimum-Effort Multi-Target Pointing OCP, we can apply the reference torque trajectory as a linearly-interpolated, open-loop control command $\boldsymbol{\tau}_{\mathbf{r}}^{OL} := \bar{\boldsymbol{\tau}}_{\mathbf{r}}$ in simulation with the dynamics described in (3.3) and parameters listed in Table 3.1. We confirm that the state trajectory of the open-loop simulation matches the solution $\{\bar{\mathbf{q}}, \ \bar{\boldsymbol{\omega}}, \ \bar{\boldsymbol{\rho}}_{\mathbf{r}}\}$ with negligible error, signifying that the OPT approximation, solved using sequential convex programming, has produced a feasible solution for the original OCP formulation.

In the presence of external disturbances, parameter uncertainty or unmodeled dynamics, the open-loop simulated trajectories do not match the solution well. A feedback control law must be applied to mitigate the model mismatch and effectively track the desired attitude trajectory. As described in Sec. B.4, we design an optimal tracking controller based on the solution to the LQR problem. The control law is used as a feedback term to regulate deviations from the desired trajectory. The rotor torque command that we apply to (3.3) consists of both feedforward and feedback terms: $\boldsymbol{\tau}_{\mathbf{r}}^{CL} := \bar{\boldsymbol{\tau}}_{\mathbf{r}} + \delta\boldsymbol{\tau}_{\mathbf{r}}$, where $\delta\boldsymbol{\tau}_{\mathbf{r}}$ is the time-varying state feedback tracking law.

To compare open-loop versus closed-loop performance in presence of model mismatch, we perturb the inertia matrix used in simulation. While we assume a nominal inertia matrix J in the OCP formulation and the LQR tracking control law design, we use a significantly perturbed $\tilde{J}$ in simulation:

$$J = \begin{bmatrix} 8.5 & 0.0 & 0.0 \\ 0.0 & 8.5 & 0.0 \\ 0.0 & 0.0 & 6.0 \end{bmatrix} \text{ kg·m}^2 \qquad \tilde{J} = \begin{bmatrix} 15.0 & \text{-1.0} & 2.0 \\ \text{-1.0} & 7.0 & \text{-3.0} \\ 2.0 & \text{-3.0} & 9.0 \end{bmatrix} \text{ kg·m}^2 \qquad (3.33)$$

As illustrated in Fig. 3.21, applying the optimal torque reference command in open-loop produces the dashed quaternion trajectory, which fails to pass through the desired quaternion points specified by the multi-target pointing schedule. However, when using the LQR tracking controller, the closed-loop trajectory passes through most of the desired quaternion points. We also confirm that the amplitude of the angular velocity peaks during inter-region slewing are reduced in Fig. 3.22.
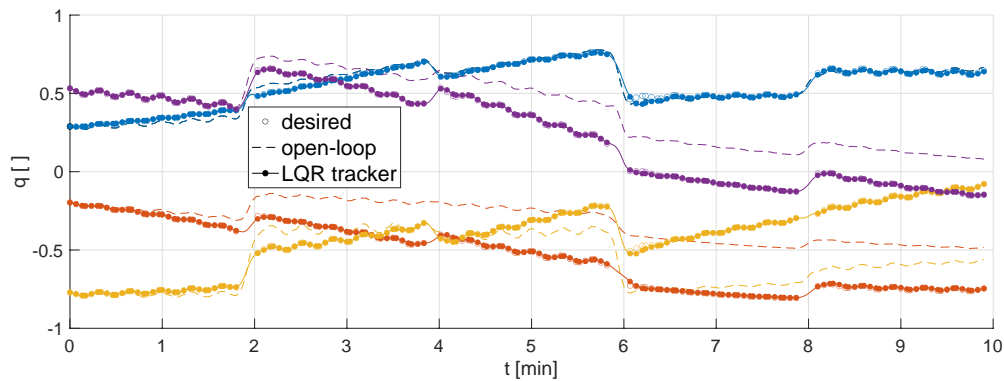

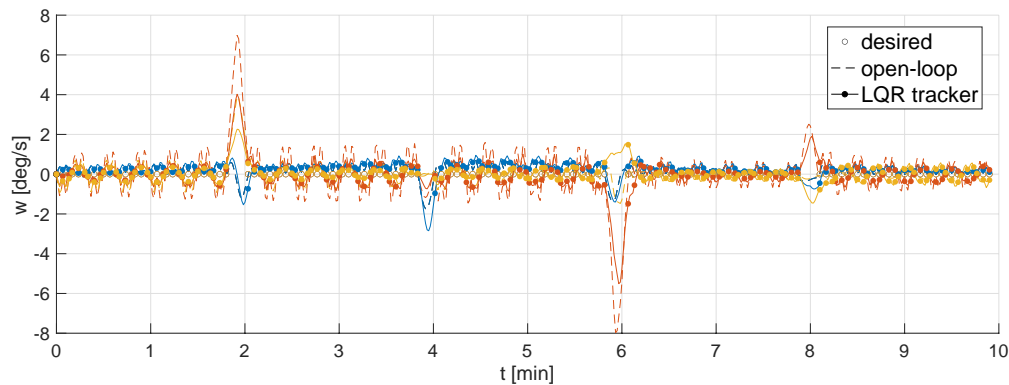Figure 3.21: LQR closed-loop vs. open-loop quaternion trajectories

Figure 3.22: LQR closed-loop vs. open-loop angular velocity trajectories

Using the performance metrics defined in Sec. B.5, we measure the error of both the open-loop and closed-loop (LQR tracker) simulated trajectories with respect to the optimized reference trajectory.

	Quaternion Error		Angular Velocity Error	
Control Strategy	q_e^{max}	q_e^{avg}	ω_e^{max}	ω_e^{avg}
Open-loop	0.5947	0.3417	3.5724	0.4116
LQR tracker	0.0653	0.0085	2.0227	0.3515

Table 3.6: Simulation trajectory error with respect to reference trajectory

As shown in Table 3.6, the closed-loop (LQR tracker) trajectory results in smaller error across all four performance metrics, when compared to the errors of the open-loop trajectory. In particular, the maximum point-wise quaternion (i.e., attitude) error is reduced by an order of magnitude when using the tracking controller. In Fig. 3.23, we directly plot the quaternion and angular velocity errors and observe that the maximum point-wise error in the open-loop quaternion trajectory coincides with the final point in the trajectory. Since the error accumulates over time, we can infer that schedules prescribed for long durations of time will result in very large attitude errors if optimal torque commands are applied in open-loop. In contrast, the LQR tracker significantly reduces error in the closed-loop quaternion trajectory. Furthermore, both the maximum and average angular velocity error is reduced with the LQR tracker. Further reduction in the angular velocity error may be achieved (at the expense of either increased quaternion error or more control effort) by tuning the LQR weight matrices (B.30).
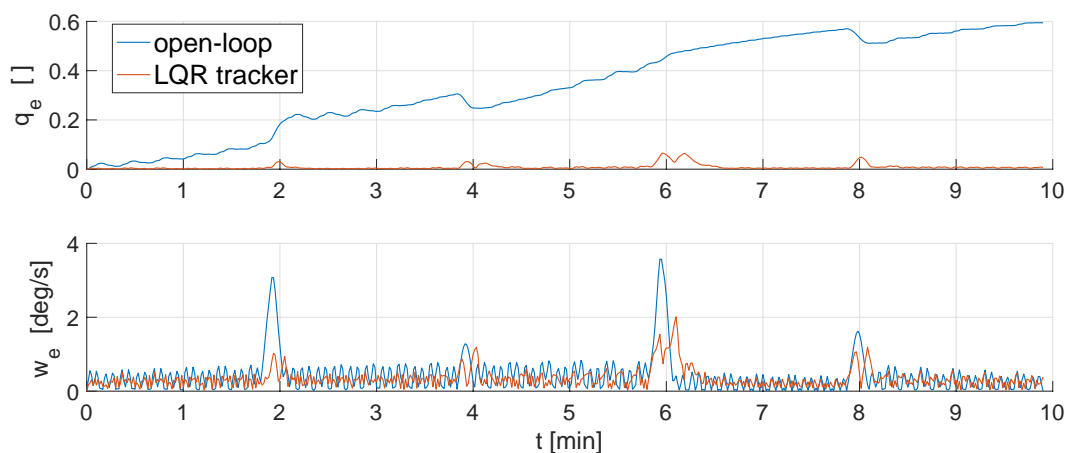

Figure 3.23: LQR closed-loop vs. open-loop attitude error

3.4 Summary

We have formulated two optimal control problems, the Minimum-Time Slew OCP and the Minimum-Effort Multi-Target Pointing OCP, that can be applied to high-fidelity spacecraft models with practical mission objectives and constraints. The 3-DOF, rotational motion of a spacecraft may be modeled with nonlinear gyrostat and actuator dynamics, quaternion kinematics, arbitrary physical parameters, and arbitrary actuator configurations. Constraints such as bounds on rotor torque, rotor momentum, body slew rate, ADCS power and energy are included. Using state-of-the-art techniques in sequential convex programming, we transcribe the non-convex, continuous-time optimal control problems into convex, finite-dimensional optimization problems (OCP→OPT) to be solved efficiently.

The Minimum-Time Slew OPT accurately estimates the required time and energy to conduct time-optimal, rest-to-rest maneuvers between any two arbitrary orientations. Compared with simulation-based approaches that implement near-optimal, eigenaxis-slew control laws about body-fixed control axes, our trajectory optimization approach is not restricted to eigenaxis slews and can be applied to more general spacecraft models and constraints. The time and energy estimates can be computed on a grid of the space of 3D rotations to produce models for use by a constellation scheduler.

Given a desired schedule of discrete pointing orientations, the Minimum-Effort Multi-Target Pointing OPT produces a dynamically-feasible, continuous reference trajectory that can achieve the pointing schedule with minimal effort. To mitigate the effects of model mismatch or external disturbances, a trajectory-tracking controller is used to regulate deviations from the trajectory. Compared to existing approaches that step from one discrete reference set-point to another, our method allows us to design the entire multi-target pointing trajectory as a single unabridged maneuver. We have applied both problems formulations to an example multi-target observation mission and reference spacecraft, demonstrating their potential use by a remote sensing constellation scheduler.

The two problem formulations are complementary and may significantly elevate the performance of agile, remote sensing satellite constellations. With more accurate estimates of the minimum time and energy required to conduct particular reorientations, a constellation scheduler is better informed to design aggressive multi-target pointing schedules that maximize constellation utility (e.g., quantity or science value of observations). The satellites of the constellations may then execute the schedules by planning feasible attitude trajectories that explicitly consider state and input constraints. Furthermore, given recent advances in the computing power of small satellite on-board computers, real-time solver implementations allow a constellation to respond quickly to changes in the environment or mission objectives.

Chapter 4

Continuous Low-Thrust Orbit Maneuvers

4.1 Motivation

Orbital maneuvering using impulsive thrust from chemical rockets is a well-established technology. Continuous low-thrust maneuvering, however, is still an active area of research. We consider two methods of continuous low-thrust. In the first application of this chapter, we use continuous low-thrust (e.g., from an electric propulsion engine) to conduct a simultaneous orbit raise and circularization maneuver. In the second, we use solar sails to harness solar radiation pressure for rendezvous trajectories to near Earth objects (NEO). The approach described in this chapter is applied to single-spacecraft problems before scaling up to multi-spacecraft constellation problems in the subsequent chapter.
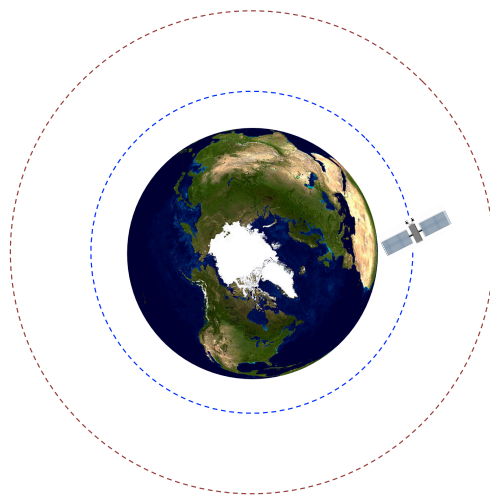

Figure 4.1: Initial parking orbit and desired operational orbit

Preliminaries

We consider three fundamental objective functions that are often considered in orbital trajectory optimization problems. A minimum-fuel solution may be found by maximizing the total mass of the spacecraft over the duration of the problem, which is equivalent to minimizing the following objective:

$$J_{obj}\left(\mathbf{r}, \mathbf{v}, m, \mathbf{T}, t_f\right) = -\int_{t_0}^{t_f} \dot{m}(t)\ dt = -m(t_f) + m(t_0) \tag{4.1}$$

A minimum-time solution can be found by minimizing the following objective:

$$J_{obj}\left(\mathbf{r}, \mathbf{v}, m, \mathbf{T}, t_f\right) = \int_{t_0}^{t_f} 1\ dt = t_f - t_0 \tag{4.2}$$

A minimum-control-effort solution is achieved by minimizing the energy of the input signal:

$$J_{obj}\left(\mathbf{r}, \mathbf{v}, m, \mathbf{T}, t_f\right) = \int_{t_0}^{t_f} \|\mathbf{T}(t)\|_2^2\ dt \tag{4.3}$$

Given a particular objective function, the most basic version of an orbital trajectory optimization problem may be defined as follows:

$$\underset{\mathbf{T},\ t_f}{\text{minimize}} \quad J_{obj}\left(\mathbf{r}, \mathbf{v}, m, \mathbf{T}, t_f\right) \tag{4.4}$$

subject to:

$$\dot{\mathbf{r}}(t) = \mathbf{v}(t) \qquad\qquad \forall t \in [t_0, t_f] \tag{4.5}$$

$$\dot{\mathbf{v}}(t) = \mathbf{a_G}(\mathbf{r}(t)) + \mathbf{a_T}(m(t), \mathbf{T}(t)) \qquad\qquad \forall t \in [t_0, t_f] \tag{4.6}$$

$$\dot{m}(t) = -\frac{\|\mathbf{T}(t)\|}{g_0 I_{sp}} \qquad\qquad \forall t \in [t_0, t_f] \tag{4.7}$$

$$r_{min} \le \|\mathbf{r}(t)\| \le r_{max} \qquad\qquad \forall t \in [t_0, t_f] \tag{4.8}$$

$$T_{min} \le \|\mathbf{T}(t)\| \le T_{max} \qquad\qquad \forall t \in [t_0, t_f] \tag{4.9}$$

$$m_{min} \le m(t_f) \qquad\qquad \forall t \in [t_0, t_f] \tag{4.10}$$

$$\mathbf{r}(t_0) = \bar{\mathbf{r}}\ , \quad \mathbf{v}(t_0) = \bar{\mathbf{v}}\ , \quad m(t_0) = \bar{m} \tag{4.11}$$

In this basic version, we only consider the accelerations due to central body gravity and continuous low thrust in (4.6). As propellant is expelled to produce thrust, the mass of the vehicle decreases according to (4.7). Using (4.8), we may impose bounds on the altitude of the spacecraft to ensure that it is maneuvering within operational guidelines. Similarly, we use (4.7) to impose bounds on the thrust magnitude, ensuring that the trajectory is feasible given the maximum and minimum thrust provided by the engine. Constraint (4.9) ensures that the trajectory is feasible given the finite amount of fuel on board. Finally, (4.10) states the initial conditions of the spacecraft.

4.2 Orbit Raise and Circularization

When a spacecraft is launched, it is generally first placed into a parking orbit by the launch vehicle, as shown in Fig. 4.1. From this orbit, the spacecraft may perform an "orbit raise" maneuver to reach a higher altitude used for the mission. A final circularization maneuver is performed to exit the transfer orbit and enter a final desired circular orbit. In [46], it is shown that applying continuous low-thrust in the tangential direction allows a spacecraft to slowly spiral outwards from a circular orbit. In [47] and [48], continuous low-thrust steering control laws are developed based on the Pontryagin Maximum Principle. In [48], numerical methods are proposed to determine the initial conditions of Lagrange multipliers used in the steering laws, however, the authors admit that the process may be time-consuming. In this section, we consider orbit raising problem formulations with input and state constraints that may be solved efficiently.

Problem Formulation without Perturbations

The following problem formulation is used to simultaneously raise the orbit of a spacecraft and circularize under the assumption of no environmental perturbations:

$$\underset{\mathbf{T},\, t_f}{\text{minimize}} \quad J_{obj}\left(\mathbf{r}, \mathbf{v}, m, \mathbf{T}, t_f\right) \tag{4.12}$$

subject to:

$$\dot{\mathbf{r}}(t) = \mathbf{v}(t) \qquad\qquad \forall t \in [t_0, t_f] \tag{4.13}$$

$$\dot{\mathbf{v}}(t) = \mathbf{a_G}(\mathbf{r}(t)) + \mathbf{a_T}(m(t), \mathbf{T}(t)) \qquad\qquad \forall t \in [t_0, t_f] \tag{4.14}$$

$$\dot{m}(t) = -\frac{\|\mathbf{T}(t)\|}{g_0 I_{sp}} \qquad\qquad \forall t \in [t_0, t_f] \tag{4.15}$$

$$r_{min} \leq \|\mathbf{r}(t)\| \leq r_{max} \qquad\qquad \forall t \in [t_0, t_f] \tag{4.16}$$

$$T_{min} \leq \|\mathbf{T}(t)\| \leq T_{max} \qquad\qquad \forall t \in [t_0, t_f] \tag{4.17}$$

$$m_{min} \leq m(t_f) \qquad\qquad \forall t \in [t_0, t_f] \tag{4.18}$$

$$\|\mathbf{r}(t_f)\| = r^{des} \tag{4.19}$$

$$v_T\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) = v_C\left(\mathbf{r}(t_f)\right) \tag{4.20}$$

$$v_R\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) = 0 \tag{4.21}$$

$$v_N\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) = 0 \tag{4.22}$$

$$\mathbf{r}(t_0) = \bar{\mathbf{r}}, \quad \mathbf{v}(t_0) = \bar{\mathbf{v}}, \quad m(t_0) = \bar{m} \tag{4.23}$$

Note that we have implemented the basic orbital trajectory optimization formulation with the addition of constraint (4.19), used to specify the final desired altitude, and (4.20) - (4.22), used to achieve the velocity required for a circular orbit. The approximation of non-convex constraints is described in (C.2).

Problem Formulation with Perturbations

In the following problem formulation, we consider the dominant perturbations in low Earth orbit. Namely, we include the accelerations due to oblate Earth "J_2" gravity and atmospheric drag (4.26). Since a circular orbit will immediately become elliptical under the influence of J_2 gravity, we do not enforce constraints (4.19)-(4.22) in this formulation. We must be content with nearly circular orbits in the presence of perturbations such as J_2 gravity. We instead replace the desired radius and circularization constraints with a terminal constraint on the final semi-major axis (4.31) and the final eccentricity (4.32):

$$\underset{\mathbf{T},\, t_f}{\text{minimize}} \quad J_{obj}\left(\mathbf{r}, \mathbf{v}, m, \mathbf{T}, t_f\right) \tag{4.24}$$

subject to:

$$\dot{\mathbf{r}}(t) = \mathbf{v}(t) \qquad\qquad\qquad\qquad\qquad\qquad\qquad \forall t \in [t_0, t_f] \tag{4.25}$$

$$\dot{\mathbf{v}}(t) = \mathbf{a_G}(\mathbf{r}(t)) + \mathbf{a_{J2}}(\mathbf{r}(t)) + \mathbf{a_D}(\mathbf{r}(t), \mathbf{v}(t), m(t)) + \mathbf{a_T}(m(t), \mathbf{T}(t)) \quad \forall t \in [t_0, t_f] \tag{4.26}$$

$$\dot{m}(t) = -\frac{\|\mathbf{T}(t)\|}{g_0 I_{sp}} \qquad\qquad\qquad\qquad\qquad \forall t \in [t_0, t_f] \tag{4.27}$$

$$r_{min} \le \|\mathbf{r}(t)\| \le r_{max} \qquad\qquad\qquad\qquad \forall t \in [t_0, t_f] \tag{4.28}$$

$$T_{min} \le \|\mathbf{T}(t)\| \le T_{max} \qquad\qquad\qquad\qquad \forall t \in [t_0, t_f] \tag{4.29}$$

$$m_{min} \le m(t_f) \qquad\qquad\qquad\qquad\qquad \forall t \in [t_0, t_f] \tag{4.30}$$

$$a\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) = a^{des} \tag{4.31}$$

$$e\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) = e^{des} \tag{4.32}$$

$$\mathbf{r}(t_0) = \bar{\mathbf{r}}, \quad \mathbf{v}(t_0) = \bar{\mathbf{v}}, \quad m(t_0) = \bar{m} \tag{4.33}$$

Example Application

We consider a spacecraft in an initial 250 km altitude circular orbit at an inclination of 28.5 degrees, the approximate latitude of the launch facilities in Cape Canaveral, Florida, USA. The objective is to reach a 550 km circular orbit in minimum-time. The spacecraft has an initial wet mass of 100 kg, of which 20 kg is propellant mass. The electric propulsion system has a maximum thrust magnitude of 0.1 N and a specific impulse of 3000 seconds. We constrain the spacecraft to maneuver above 200 km altitude, to prevent excessive drag, and below 700 km, to avoid other spacecraft. Given these example parameters, we solve two formulations of the simultaneous orbit raise and circularization problem. The first formulation only considers the accelerations due to central body gravity and continuous low-thrust. The second formulation includes the perturbations of atmospheric drag and J_2 gravity. For both problems, we design an initial guess at the solution by propagating the spacecraft from its initial state with maximum thrust in the tangential direction. In the case of the problem formulation with perturbations, we propagate the spacecraft states with J_2 gravity and atmospheric drag included in the simulation environment.

In Fig. 4.2, we show the optimal thrust trajectory, expressed in the spacecraft's RTN frame (orbit radial, tangential, and normal directions), for both problem formulations, i.e., with and without perturbations. We first note that the minimum-time to conduct a 250km-to-500km orbit raise is approximately 48 hours when not considering perturbations. However, when considering J_2 gravity and atmospheric drag, the maneuver requires approximately 51 hours. We attribute the difference to the time required to overcome the effects of perturbations during the orbit raise and circularization maneuver. For example, atmospheric drag acts directly against the velocity of the spacecraft, retarding any maneuvers made through the atmosphere. The J_2 gravity perturbation adds non-radial accelerations depending on position of the spacecraft. The solutions to both problems apply maximum thrust in the tangential direction for a significant duration of the maneuver. We also note that thrust is applied in the normal direction is much greater when J_2 gravity is considered.
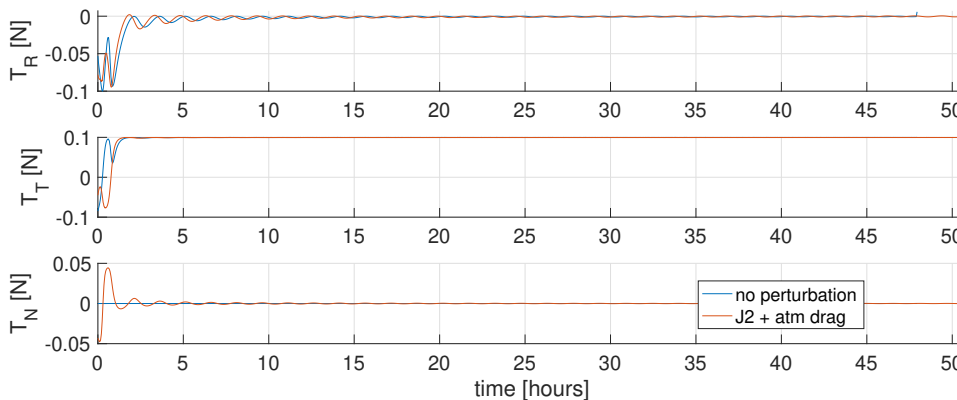

Figure 4.2: Optimal thrust trajectory for problem with and without perturbations

In the following figures of this section, we study simulation results of applying the optimal thrust trajectories in open-loop. In Fig. 4.3, we compare the acceleration from central body gravity to the acceleration from thrust for the unperturbed problem. We first note that the acceleration due to thrust is much smaller than that of gravity, by a factor of almost 10^{-4}. Despite this, the spacecraft is able to conduct the orbit raise maneuver. As expected, the acceleration due to gravity is only present in the (negative) radial direction and decreases as the spacecraft raises altitude. Although a component of thrust acceleration exists in the radial direction, it is significantly smaller than that of central body gravity. Most of the thrust acceleration is applied in the tangential direction and we observe that the minimum-time strategy is to initially decelerate before applying maximum acceleration for the rest of the maneuver. We observe a small amount of thrust maneuvering in the normal direction that we attribute to the initial inclination of the orbit. In general, we find that the solution to the orbit raise problem does not stray far from the initial guess.

In Fig. 4.4, we compare all four accelerations (central body gravity, J_2 gravity, atmospheric drag, thrust) in the perturbed case. Central body gravity remains the dominant
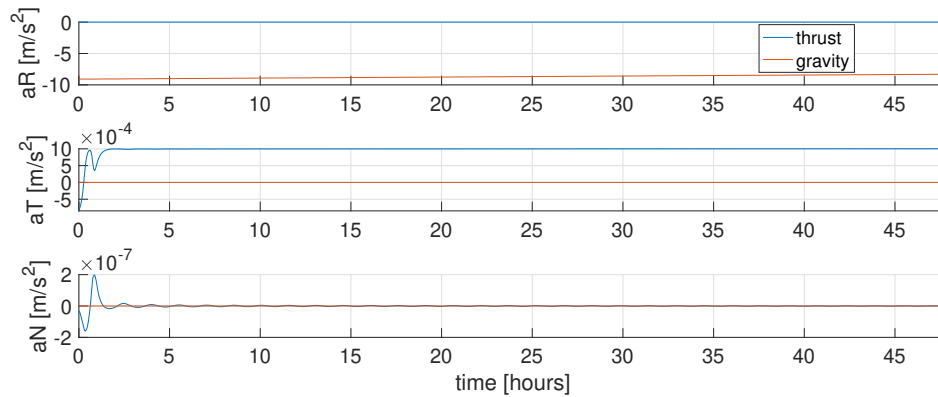
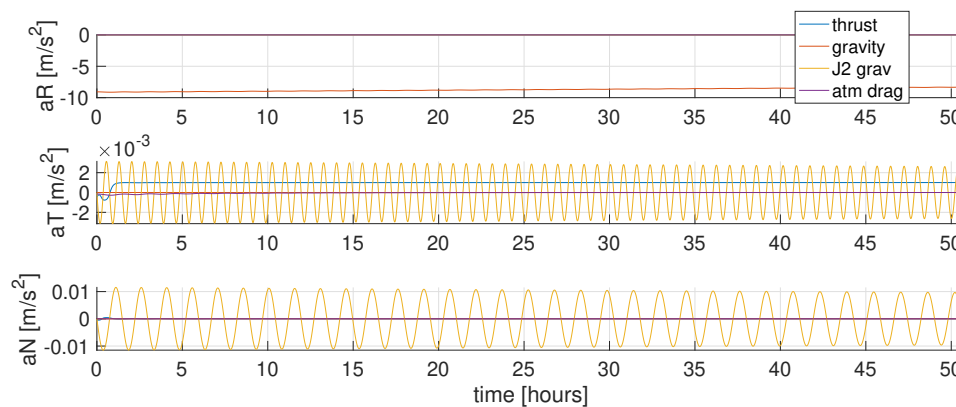Figure 4.3: Optimal accelerations for problem without perturbations

Figure 4.4: Optimal accelerations for problem with perturbations

acceleration in the radial direction. In both tangential and normal direction, J_2 gravity is the most significant acceleration on the spacecraft, with a maximum value that is more than two times larger than the maximum thrust magnitude. However, unlike the acceleration due to drag, the J_2 acceleration is periodic, changing with the orbital position of the spacecraft. The optimal maneuver is similar to the unperturbed case where maximum thrust is applied in the positive tangential direction for most of the maneuver.

Figure 4.5 shows the altitude climb resulting from the solutions to the unperturbed and perturbed problems. In both instances, the spacecraft reaches the desired altitude of 550 km. For the unperturbed case, this is achieved by satisfying constraint 4.19. However, in the perturbed case we note the bounded oscillation of the spacecraft altitude under J_2 gravity. Hence, we use constraint 4.31, which is essentially a constraint on the average altitude of the spacecraft, ensuring that the spacecraft oscillates about the desired 550 km altitude orbit. From this plot it is also clear that perturbations affect the time required to complete an

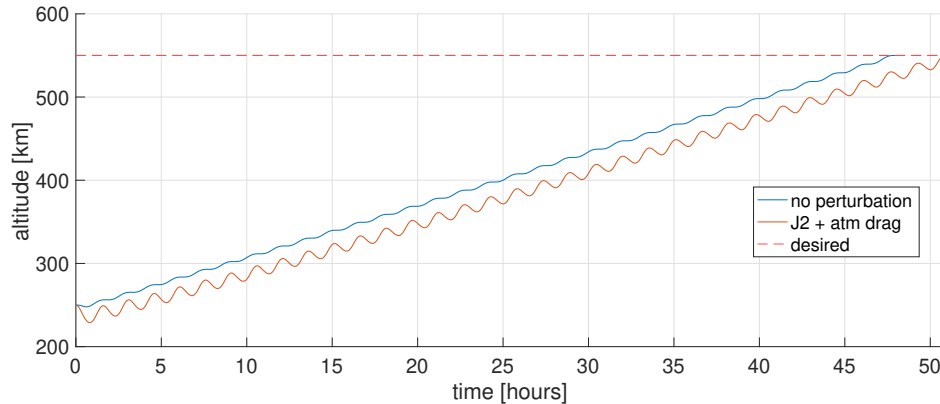orbit raise maneuver, resulting in a longer-duration trajectory.

Figure 4.5: Minimum-time altitude climb for problem with and without perturbations

We recall the circularization condition implied by constraints on the velocity components (4.20)-(4.22) or by a constraint on the orbit eccentricity (4.32). In Fig. 4.6, we observe that both approaches lead to the spacecraft obtaining the necessary speed for a circular orbit at the desired 550 km altitude. We find that the zero eccentricity constraint is a more relaxed condition compared to the constraints on the velocity components, well-suited for the perturbed problem formulation. Furthermore, an advantage of using a constraint on the eccentricity is that we can design not only circular orbits but also elliptical orbits that may be useful in certain applications (e.g., Molniya orbits).
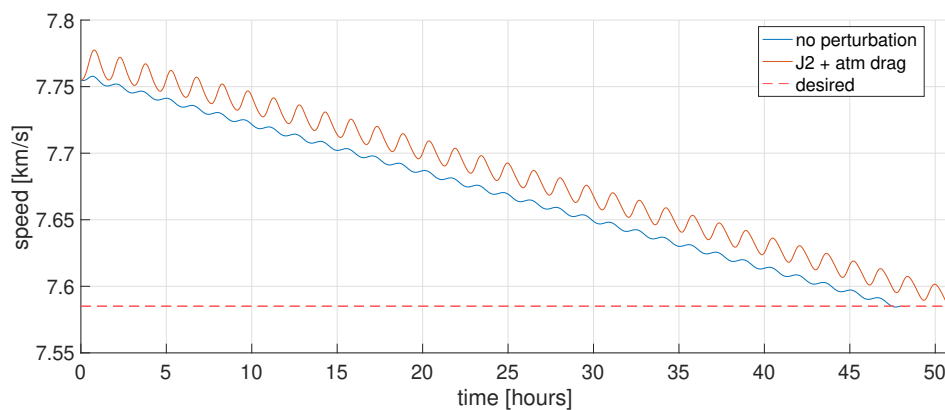

Figure 4.6: Optimal speed for problem with and without perturbations

In Fig. 4.7, we note that the radial component of the velocity reaches zero at the end of the maneuver in both the unperturbed and perturbed cases. Furthermore, the tangential
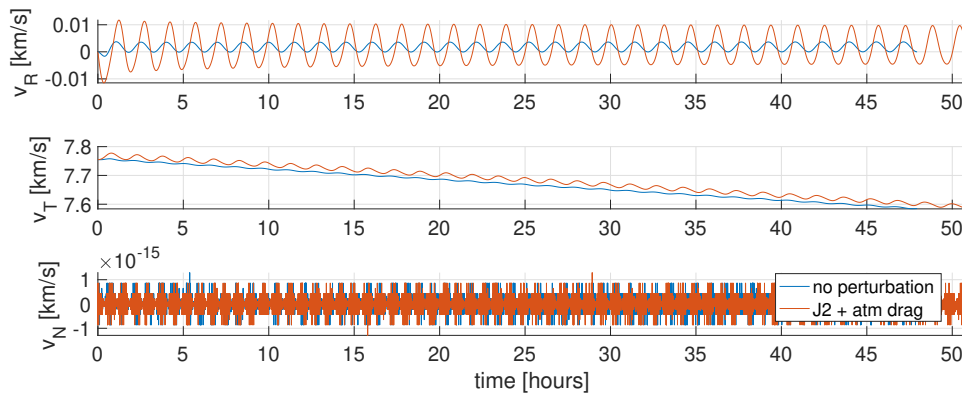
Figure 4.7: Optimal velocity components for problem with and without perturbations

component of the velocity matches the speed required for a 550 km circular orbit. The velocity in the normal direction is negligible.

In Fig. 4.8, we show how the orbital elements change during the maneuver in both the unperturbed and perturbed cases. The first plot shows the semi-major axis of the orbit and we note that the perturbed trajectory requires more time to reach the desired semi-major axis (i.e., altitude) compared to the unperturbed trajectory. We attribute the delay to the atmospheric drag that retards the spacecraft motion throughout the maneuver. We also note the oscillations in the perturbed trajectory due to J_2 perturbation. The second plot shows
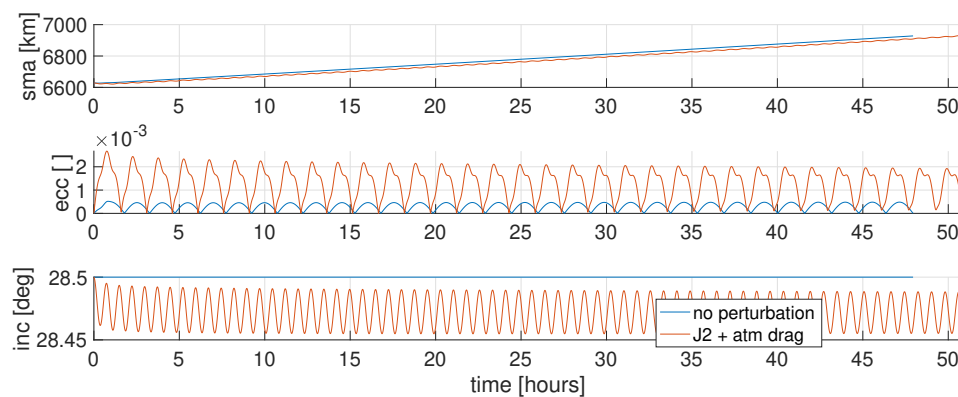
Figure 4.8: Optimal orbital elements for problem with and without perturbations

the instantaneous eccentricity of the trajectories. The application of thrust acceleration causes both unperturbed and perturbed trajectories to take on elliptical orbits, reaching minimum eccentricity values at the periapsis of every orbital revolution. We also observe how the J_2 perturbation causes more elliptical orbits. Finally, the third plot shows how J_2 gravity causes small oscillations in the orbital inclination.

4.3 Orbital Rendezvous using Solar Radiation Pressure

In the previous section, we described how the basic orbital trajectory optimization problem can be used to change the size, shape, or orientation of an orbit. We now consider the problem of performing a rendezvous maneuver with a moving target, such as a planet or asteroid. Rendezvous requires the spacecraft's position and velocity vectors to match those of the target. In this section we consider continuous low-thrust derived not from onboard propulsion but from solar sails that take advantage of the radiation pressure from the Sun [49].
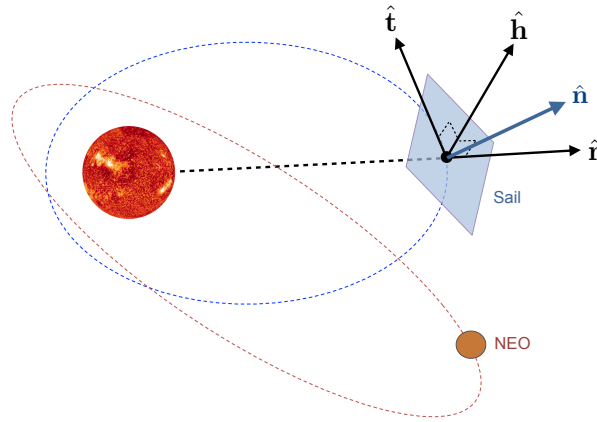
Figure 4.9: Solar sail orbital rendezvous with Near Earth Object

Problem Formulation

We consider an orbital rendezvous problem where our input acceleration is derived from solar radiation pressure (C.5). Our problem formulation is described below:

$$\underset{\alpha,\ \beta,\ t_f}{\text{minimize}} \quad \int_{t_0}^{t_f} 1 \ dt \tag{4.34}$$

subject to:

$$\dot{\mathbf{r}}(t) = \mathbf{v}(t) \qquad\qquad \forall t \in [t_0, t_f] \tag{4.35}$$

$$\dot{\mathbf{v}}(t) = \mathbf{a_G}(\mathbf{r}(t)) + \mathbf{a_S}(\mathbf{r}(t), \mathbf{v}(t), \alpha(t), \beta(t)) \qquad \forall t \in [t_0, t_f] \tag{4.36}$$

$$-\frac{\pi}{2} \leq \alpha(t) \leq \frac{\pi}{2} \qquad\qquad \forall t \in [t_0, t_f] \tag{4.37}$$

$$0 \leq \beta(t) \leq 2\pi \qquad\qquad \forall t \in [t_0, t_f] \tag{4.38}$$

$$\|\mathbf{r}(t_f) - \mathbf{r}^{des}(t_f)\| \leq \epsilon_r \tag{4.39}$$

$$\|\mathbf{v}(t_f) - \mathbf{v}^{des}(t_f)\| \leq \epsilon_v \tag{4.40}$$

$$\mathbf{r}(t_0) = \bar{\mathbf{r}} \ , \quad \mathbf{v}(t_0) = \bar{\mathbf{v}} \tag{4.41}$$

Unlike propulsive methods of acceleration that require a propellant to be spent, solar sails do not require fuel and only rely on the Sun's radiation. Hence, we use a minimum-time objective in the formulation. In the equations of motion, we include the accelerations due to the Sun's gravity and solar radiation pressure (4.36). The cone angle and clock angle are restricted to the intervals in (4.37) and (4.38) to ensure that the solar sail's orientation may be uniquely defined by a given pair. Most importantly, the rendezvous condition is described by constraints (4.39) and (4.40) which ensure that the position and velocity of the spacecraft are within a specified Euclidean distance of those of the target body (e.g., planet, asteroid, near Earth object). We assume that the position and velocity of the target body are known, and we treat them as forcing functions in our problem formulation. Furthermore, we note that these two constraints are not only functions of the state variables but also of the final time, a decision variable in our problem formulation.

Example Application

We consider a spacecraft in a circular, ecliptic orbit at a distance 1 [AU] from the center of the Sun. The objective is to rendezvous with a target body that is in a 2[AU] circular orbit, with an inclination of 5 degrees from the ecliptic plane. As a reference, we note that the orbit of Mars has a semi-major axis of approximately 1.524 [AU], an eccentricity of 0.0934, and an inclination of 1.85 degrees.

In Fig. 4.10, we show the time-optimal rendezvous trajectory of the spacecraft in the Sun-centered inertial frame, where the Sail's trajectory is shown in blue and the near Earth object (NEO)'s trajectory is shown in red.. The black vectors on the Sail's trajectory show the direction of the SRP acceleration vector; we note that it has both a radial and tangential component.
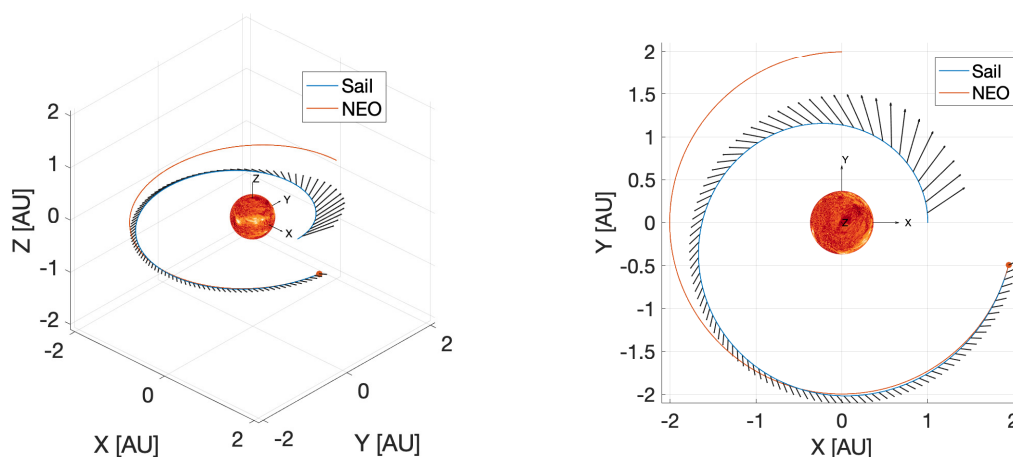

Figure 4.10: Sail-NEO rendezvous trajectory with SRP acceleration vector

In Fig. 4.11, we show that the Sail's orbital "altitude" reaches that of the NEO in

approximately 2 years. The speed of the Sail also matches that of the NEO at the final
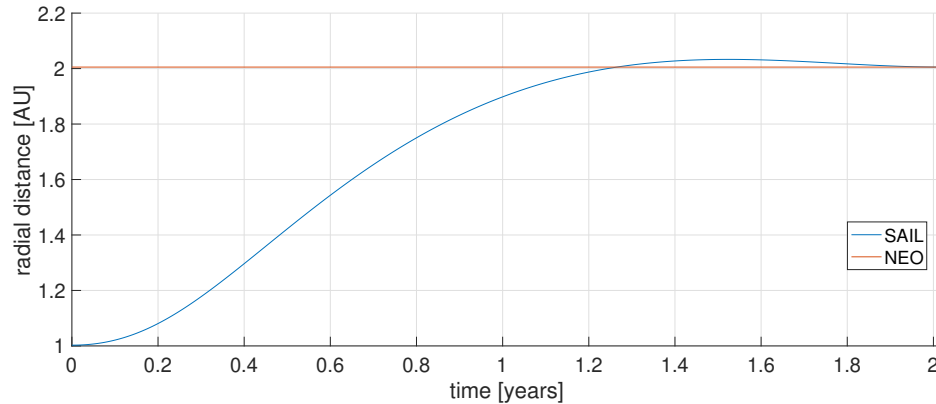
Figure 4.11: Radial distance of Sail and NEO from Sun

time in Fig. 4.12. By observing the components of the position vector in the inertial frame
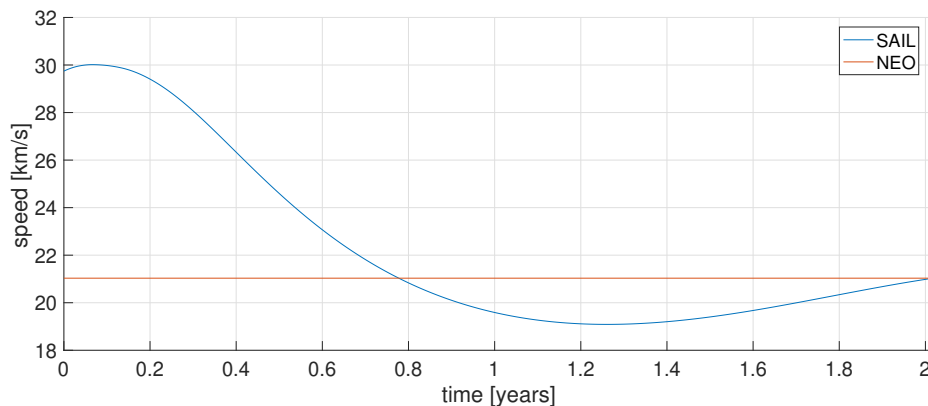
Figure 4.12: Speed of Sail and NEO

(Fig. 4.13), we confirm that the Sail's position is co-located with the position of the NEO at the final time. Similarly, we verify that the components of the Sail's velocity vector align with those of the NEO (Fig. 4.14), confirming the rendezvous condition. Once the maneuver is complete and we set the cone angle to 90 degrees, the spacecraft's motion will coincide with that of the NEO. Using the position and velocity vectors, we may represent the orbital elements for the Sail and the NEO in Fig. 4.15. In particular, we focus on the instantaneous semi-major axis, eccentricity, and inclination of the two bodies and confirm that they are in the same orbit. In Fig. 4.16, we note that most the Sail's speed is in its tangential direction of motion. However, a significant speed of 4 km/s is achieved in the initial part of the trajectory. As expected, the tangential speed decreases as we spiral out
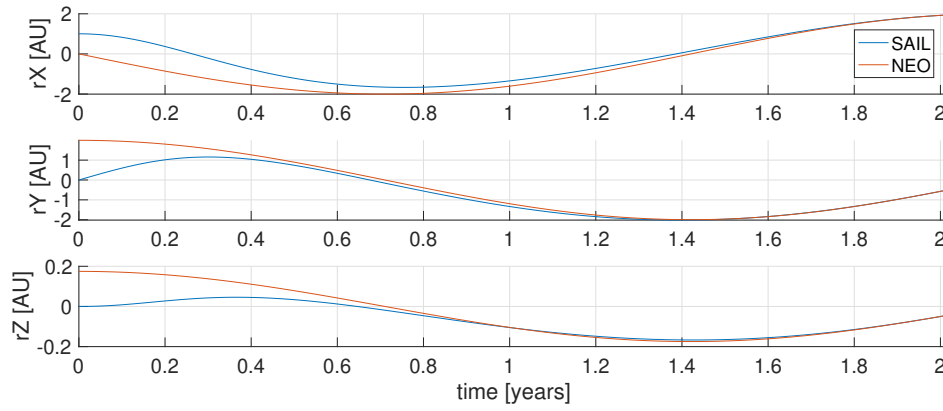
Figure 4.13: Position of Sail and NEO expressed in SCI frame
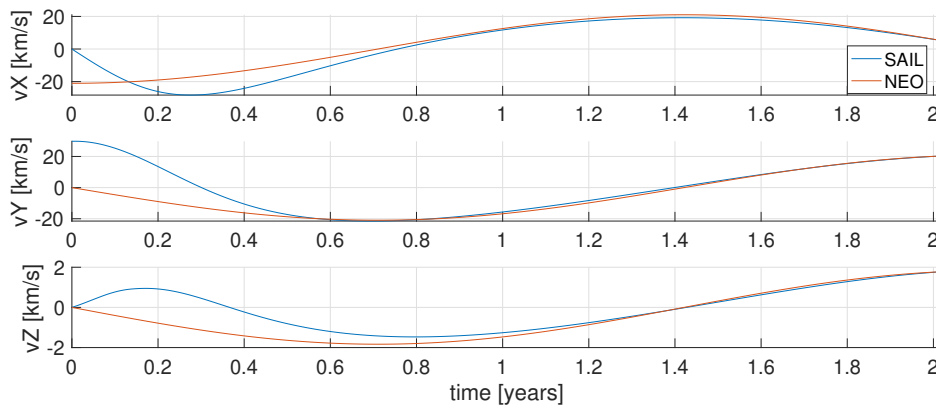
Figure 4.14: Velocity of Sail and NEO expressed in SCI frame
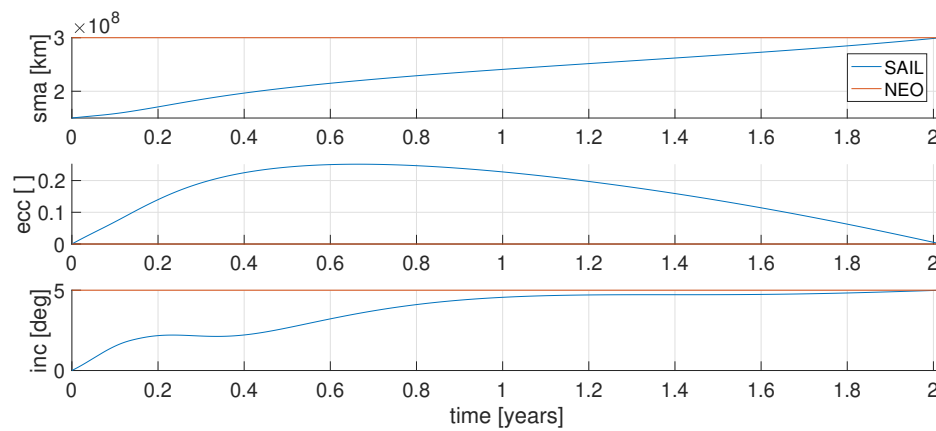
Figure 4.15: Semi-major axis, eccentricity, inclination of Sail and NEO

from the center of the Sun. We attribute the small component of velocity in the orbital normal direction to numerical error. The optimal cone and clock angle are shown in Fig.
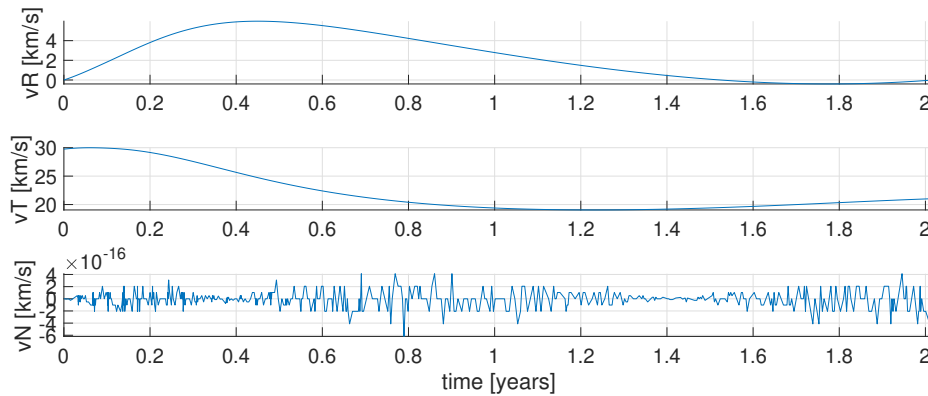
Figure 4.16: Velocity of Sail expressed in RTN frame

4.17. We note that the cone angle remains in the range of 35 degrees, which is the optimal angle for maximizing orbital semi-major axis. Furthermore, we observe that the clock angle eventually reaches a value near 90 degrees after the 5 degree orbital inclination error has been zeroed out. Finally, in Fig. 4.18 we observe that the inward "pull" of the Sun's gravity
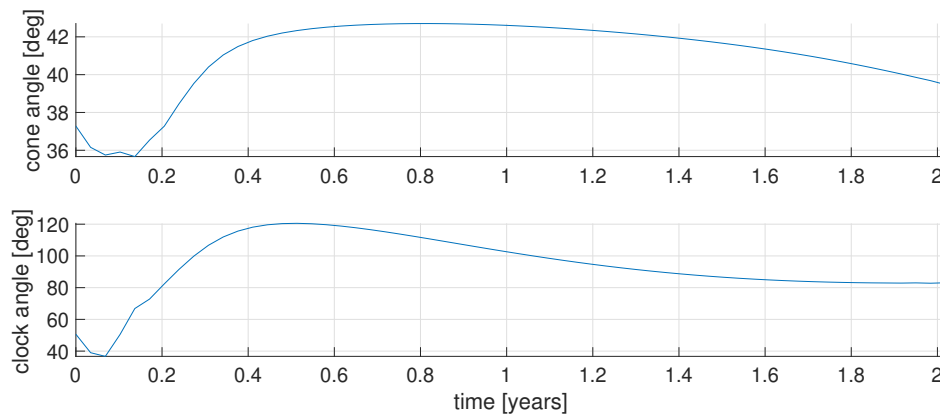
Figure 4.17: Cone and clock angles of Sail

is the dominant acceleration component. Nevertheless, the spacecraft is able to maneuver outwards by applying acceleration in the tangential direction of orbital motion.

We note that this problem may be repeated for the same two bodies (i.e., Sail and NEO) when they are at different positions in the orbit. By choosing the appropriate "departure" date and time, we may find not only the optimal rendezvous trajectory but also the optimal departure time.
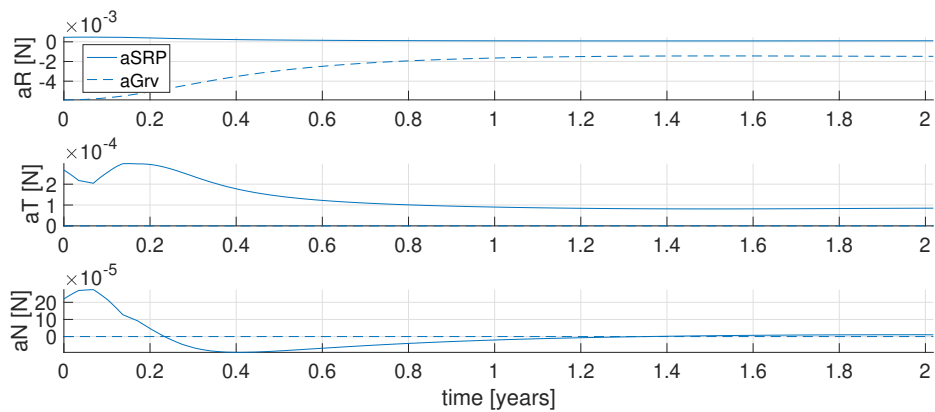
Figure 4.18: Accelerations due to SRP and gravity expressed in RTN frame

4.4 Summary

We have developed a basic orbital trajectory optimization framework upon which application-specific dynamics, constraints and objectives can be added. We demonstrated this general approach on two different examples. We first optimized a minimum-time trajectory to perform a simultaneous orbit raise and circularization maneuver in low Earth orbit. In addition to central body gravity and continuous low-thrust, we considered the dominant perturbations in low Earth orbit: J_2 gravity and atmospheric drag. We then optimized a rendezvous trajectory to an asteroid in the heliocentric inertial frame, considering the Sun's gravity and the acceleration due to solar radiation pressure. In the subsequent chapter, we continue to build upon this framework to also approach multi-spacecraft constellation problems.

Chapter 5

Centralized Approach to Constellation Formation

5.1 Motivation

With the trend towards smaller satellites, launch vehicles intended for carrying large payloads may now deploy tens and even hundreds of satellites into orbit. When a group of small satellites is deployed into its designated orbit from the same launch vehicle, the satellites are subsequently spaced out into a desired constellation formation. In general, constellations are designed so that satellites occupying the same plane are positioned in an equally-spaced formation. The procedure for spacing the satellites is called *constellation phasing*. To aid the phasing procedure, a launch vehicle will enter a spin so that each satellite is given a small initial ejection speed upon deployment. However, to fully phase a constellation, the spacecraft must use onboard propulsion or differential drag to change their relative positions. The use of differential drag has been demonstrated on orbit by Planet Labs [50] and usually requires months of time to fully phase a constellation. In comparison, continuous low-thrust propulsion may require only days or weeks, as demonstrated by SpaceX [51]. In both [32] and [52], the authors focus on time-optimal trajectories to constellation phasing by deciding the amount of time each satellite spends in a high- or low- drag configuration (i.e., by inducing differential drag) or the amount of time applying thrust. Both works assume planar motion and [52] assumes thrust is only applied in the tangential direction (i.e., no thrust in the radial direction). With such problem formulations, it may be difficult to assess the impact of perturbations that act normal to the orbital plane, such as J_2 gravity. Furthermore, the orientation of the constellation in 3-dimensional space may not be controlled with a planar problem formulation.

In this chapter, we propose a 3-dimensional, centralized approach to constellation phasing using the trajectory optimization framework described in Chapter 4. This approach allows us to consider out-of-plane perturbations and maneuvers.

5.2 Constellation Phasing in Circular Orbits

In Fig. 5.1, we illustrate an equally-spaced constellation consisting of four satellites. The radial unit vector $\hat{\mathbf{r}}_i$ points to the ith satellite and the angle θ_i is the angular spacing between the ith and $(i+1)^{st}$ satellites.
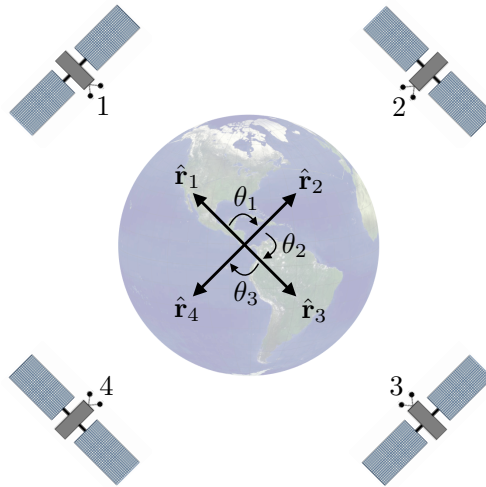
Figure 5.1: Constellation with equal spacing

Problem Formulation

The following problem formulation is used to form a single-plane, circular constellation from a set of initial satellite states. We assume the only accelerations acting on the spacecraft are central body gravity and continuous low-thrust. The objective, dynamics and most of the constraints are shared with the single spacecraft problem formulation of the previous chapter. However, we now have N as many constraints for each satellite. We note the addition of two new sets of constraints. First, the set of constraints in (5.12) enforce all spacecraft to occupy the same orbital plane at the end of the phasing maneuver. The constraints represented by (5.13) ensure that the relative angular spacing between neighboring satellites take on desired values. For an equally spaced constellation we note that the desired angular spacings should be $\theta_i^{des} = \frac{2\pi}{N}$ for all $i = 2, \ldots, N$. In this problem formulation, the spacecraft are only coupled by constraint set (5.13) and the objective function.

$$\underset{\{\mathbf{T}_i\}_{i=1}^N,\, t_f}{\text{minimize}} \quad J_{obj}\left(\{\mathbf{r}_i, \mathbf{v}_i, m_i, \mathbf{T}_i\}_{i=1}^N, t_f\right) \tag{5.1}$$

subject to:

$$\dot{\mathbf{r}}_i(t) = \mathbf{v}_i(t) \qquad\qquad \forall t \in [t_0, t_f], \forall i \in \{1, N\} \tag{5.2}$$

$$\dot{\mathbf{v}}_i(t) = \mathbf{a_G}(\mathbf{r}_i(t)) + \mathbf{a_T}(m_i(t), \mathbf{T}_i(t)) \qquad \forall t \in [t_0, t_f], \forall i \in \{1, N\} \tag{5.3}$$

$$\dot{m}_i(t) = -\frac{\|\mathbf{T}_i(t)\|}{g_0 I_{sp}} \qquad\qquad \forall t \in [t_0, t_f], \forall i \in \{1, N\} \tag{5.4}$$

$$r_{min} \le \|\mathbf{r}_i(t)\| \le r_{max} \qquad\qquad \forall t \in [t_0, t_f], \forall i \in \{1, N\} \tag{5.5}$$

$$T_{min} \le \|\mathbf{T}_i(t)\| \le T_{max} \qquad\qquad \forall t \in [t_0, t_f], \forall i \in \{1, N\} \tag{5.6}$$

$$m_{min} \le m_i(t_f) \qquad\qquad \forall t \in [t_0, t_f], \forall i \in \{1, N\} \tag{5.7}$$

$$\|\mathbf{r}_i(t_f)\| = r^{des} \qquad\qquad \forall i \in \{1, N\} \tag{5.8}$$

$$v_T\left(\mathbf{r}_i(t_f), \mathbf{v}_i(t_f)\right) = v_C\left(\mathbf{r}_i(t_f)\right) \qquad \forall i \in \{1, N\} \tag{5.9}$$

$$v_R\left(\mathbf{r}_i(t_f), \mathbf{v}_i(t_f)\right) = 0 \qquad\qquad \forall i \in \{1, N\} \tag{5.10}$$

$$v_N\left(\mathbf{r}_i(t_f), \mathbf{v}_i(t_f)\right) = 0 \qquad\qquad \forall i \in \{1, N\} \tag{5.11}$$

$$\hat{\mathbf{h}}^{des} \cdot \hat{\mathbf{h}}\left(\mathbf{r}_i(t_f), \mathbf{v}_i(t_f)\right) = 1 \qquad\qquad \forall i \in \{1, N\} \tag{5.12}$$

$$\hat{\mathbf{r}}_{i-1}(t_f)^\top \hat{\mathbf{r}}_i(t_f) = \cos(\theta_{i-1}^{des}) \qquad\qquad \forall i \in \{2, N\} \tag{5.13}$$

$$\mathbf{r}_i(t_0) = \bar{\mathbf{r}}_i\ , \quad \mathbf{v}_i(t_0) = \bar{\mathbf{v}}_i\ , \quad m_i(t_0) = \bar{m}_i \qquad \forall i \in \{1, N\} \tag{5.14}$$

Example Application

In our numerical example, we solve a minimum-time constellation phasing problem with the following objective:

$$J_{obj}\left(\{\mathbf{r}_i, \mathbf{v}_i, m_i, \mathbf{T}_i\}_{i=1}^N, t_f\right) = \int_{t_0}^{t_f} 1\ dt \tag{5.15}$$

We assume that all spacecraft have the same initial conditions, reflecting an equatorial, circular orbit at 600 km altitude. Each spacecraft has an initial wet mass of 100 kg, of which 20 kg is propellant mass. The electric propulsion system on each spacecraft has a maximum thrust magnitude of 0.1 N and a specific impulse of 3000 seconds. We constrain the spacecraft to maneuver in a radial band between 300 km and 900 km altitudes. The constraints on the final states require that all satellites return to the same 600 km altitude, circular orbit at the end of the phasing maneuver.

In order to start the trajectory optimization process, we provide an initial guess inspired by the solution to a planar version of the problem that assumes the use of differential drag [33]. For each satellite we apply zero thrust in the radial and normal directions. However, in the tangential direction, we apply the thrust profile shown in Fig. 5.2. The small values
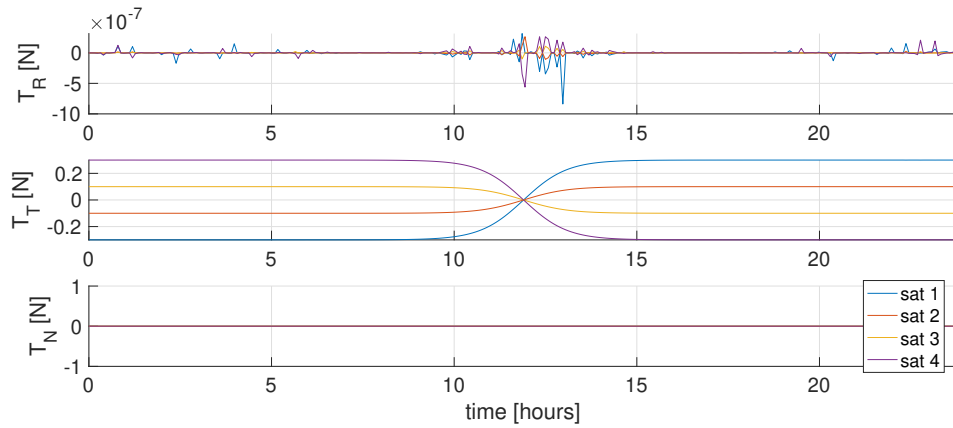
Figure 5.2: Initial Guess for thrust commands

of radial thrust in the figure are due to numerical error when mapping RTN frame thrust values to ECI frame values, and vice versa.

The trajectory optimization problem converges on a solution with the thrust profile shown in Fig. 5.3. We first note the non-negligible radial thrust trajectories. Furthermore, the tangential thrust trajectories are qualitatively very different from those of the initial guess. No thrust is applied in the normal direction.
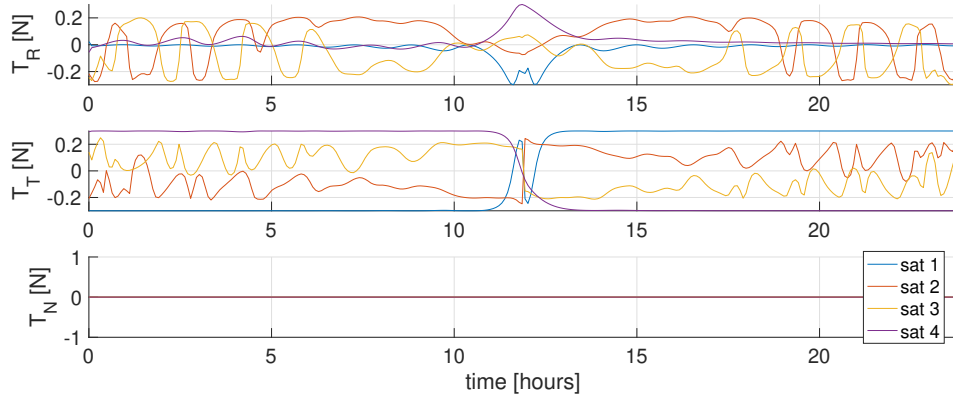
Figure 5.3: Optimal thrust commands

Figure 5.4 shows an equally-spaced constellation achieved by applying the solution to the trajectory optimization problem in open-loop.

In Fig. 5.5, we observe that the angular spacing of the satellites reach the desired 90 degree spacing at the end of the maneuver.

Although we have shown that the angular spacing constraint has been satisfied at the end of the maneuver, we must confirm that the constellation will remain equally spaced after
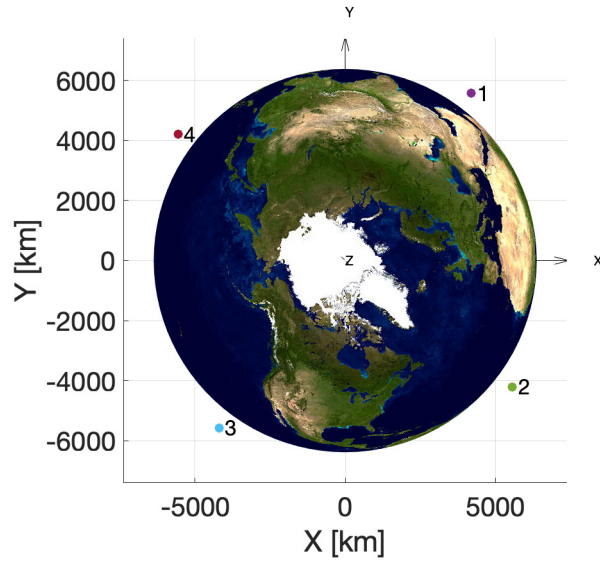
Figure 5.4: Equally-spaced constellation of four satellites in circular low Earth orbit
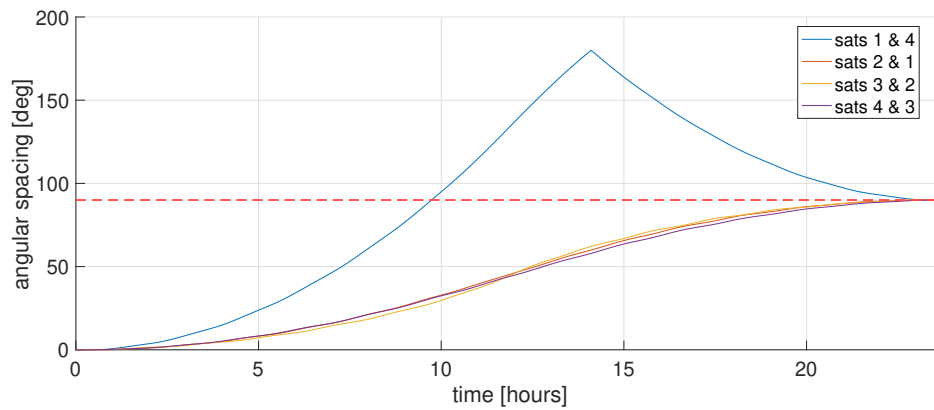
Figure 5.5: Angular spacing of neighboring satellites during phasing maneuver

the maneuver. In Fig 5.6, we show the change in altitude of each of the spacecraft during the phasing maneuver. We note that at the end of the maneuver, all satellites have returned to the desired orbital altitude of 600 km.
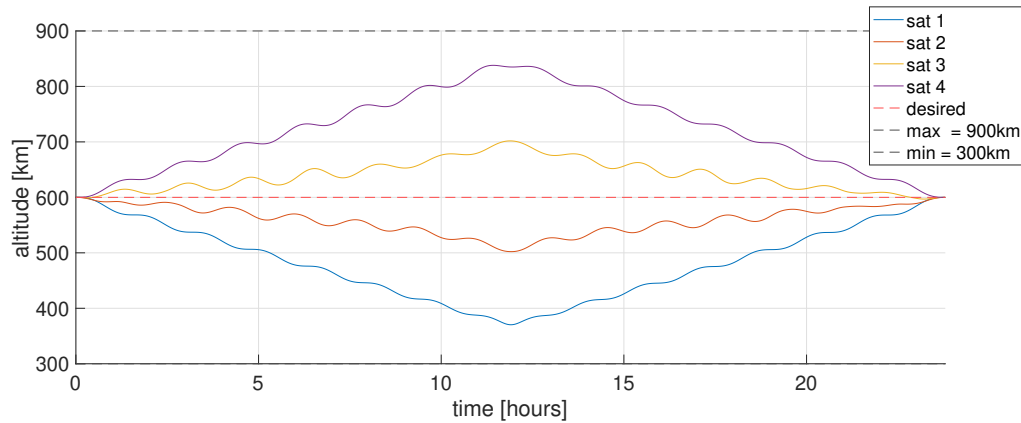
Figure 5.6: Altitude of constellation during phasing maneuver

In Fig. 5.7, we observe the change in orbital speed for each satellite and note that as a spacecraft drops in altitude, its orbital speed increases. We also note that all satellites achieve the same final orbital speed at the end of the maneuver.
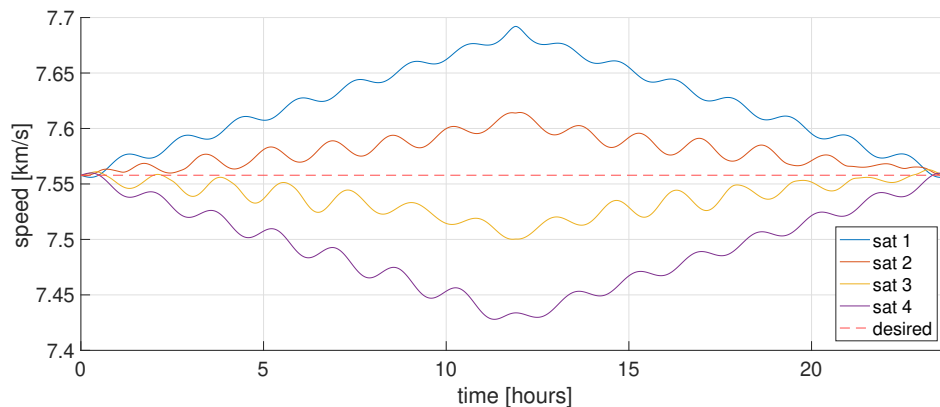
Figure 5.7: Spacecraft speed during phasing maneuver

Achieving the same altitude and speed are necessary to maintain an equally-spaced constellation but we must also check the velocity components of each satellite. In Fig. 5.8, we confirm that the radial and normal velocity components of all satellites reach zero at the end of the maneuver. Furthermore, the tangential speed of all spacecraft reach the same value. We have confirmed that the constellation will remain in formation under the dynamics assumed in this problem.
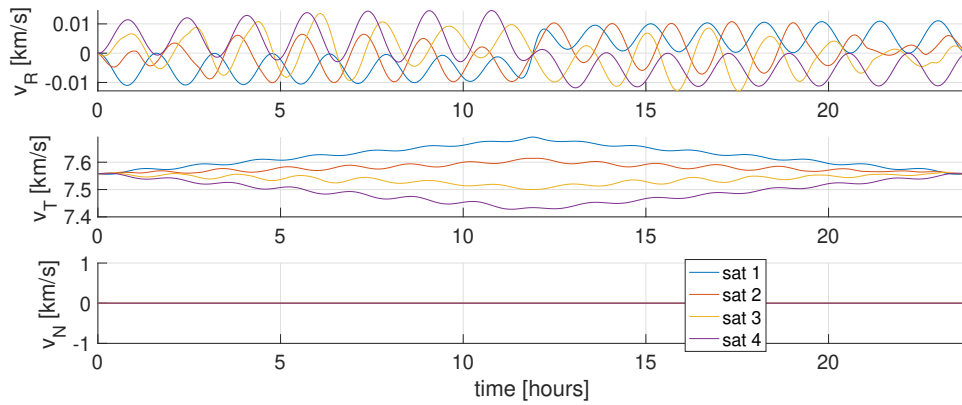
Figure 5.8: RTN velocity components during phasing maneuver

Another representation of the spacecraft states are the orbital elements. In Fig. 5.9, we confirm that the satellites have reached the same semi-major axis and zero eccentricity at the end of the maneuver. Furthermore, the inclination of the satellites' orbits remain zero.
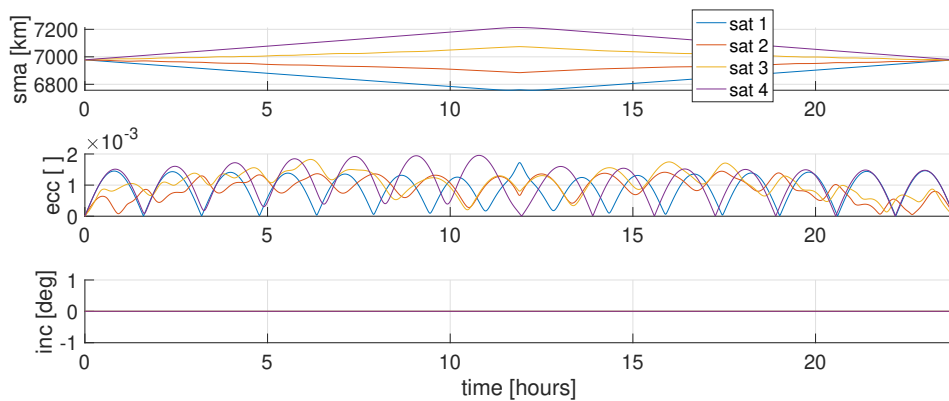
Figure 5.9: Orbital elements during phasing maneuver

Finally, in Fig. 5.10 we show that the decrease in total spacecraft mass due to propellant use. This minimum-time formulation results in uneven fuel usage among the satellites. We may infer that the first and last satellites use the most fuel while the satellites towards the middle of the group consume the least. In future work, we may consider different formulations to minimize the difference in fuel usage.
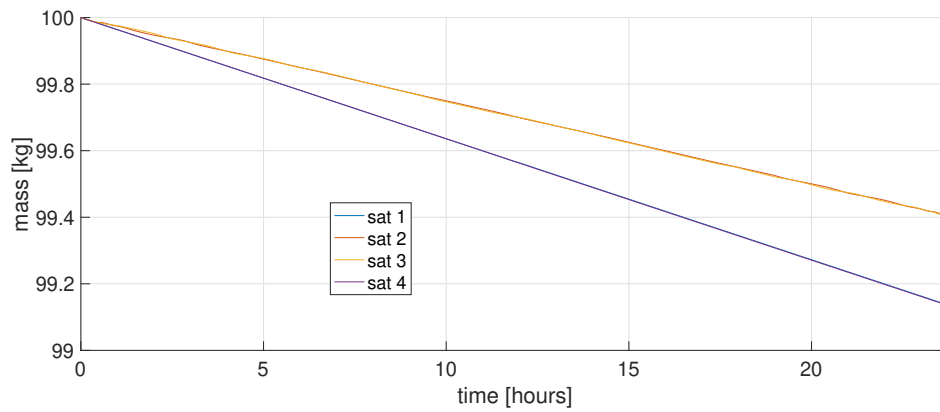
Figure 5.10: Spacecraft mass during phasing maneuver

5.3   Summary

We have formulated a constellation phasing problem that not only produces an equally-spaced formation but also enforces constraints on the final states of each spacecraft. By placing the spacecraft into circular orbits with the same orbit normal direction, we ensure that the constellation maintains the desired formation after the phasing maneuver. Future work will include perturbations including atmospheric drag and J_2 gravity, requiring a general problem formulation that allows the specification of elliptical orbits.

Chapter 6

Distributed Approach to Constellation Formation

6.1 Motivation

A satellite constellation is a group of satellites that are coordinated to achieve objectives that may not be possible with a single satellite. Constellations have been applied to serve as telecommunications or broadcasting networks, provide global imagery and weather services, and enable global positioning and navigation capabilities. The control of such constellations can be divided into two different problems: *formation* and *station-keeping.* formation refers to the process of forming the constellation once the satellites have been deployed by the delivery vehicle. For example, we may spread out a cluster of satellites in a desired orbital plane to form an equally-spaced constellation. Once the desired constellation is acquired, station-keeping refers to the process of maintaining relative positions and velocities in the presence of disturbances. The formation of a small spacecraft constellation in low Earth orbit, using a centralized approach, is studied in [33]. A centralized approach may be used if, for example, a large number of ground stations are available to measure and control the satellites.

In this chapter, we shift our focus to a distributed approach for acquiring and station-keeping a constellation. A distributed control strategy is appealing for satellite constellations in situations where centralized control is difficult or impossible. For example, as thousands of satellites are employed in constellations, the resulting uplink/downlink demands on a network of Earth-based ground stations may become unmanageable. A distributed strategy is also critical for a constellation orbiting a planet without ground stations.

Passivity-based methods are well suited for distributed control of large-scale, interconnected systems [53–55]. We model our constellation as an interconnected system where we assume each satellite has a communication link with neighboring satellites, sharing relative angular position information. An internal feedback control law is designed for the satellites and we certify that each satellite and communication link is equilibrium independent pas-
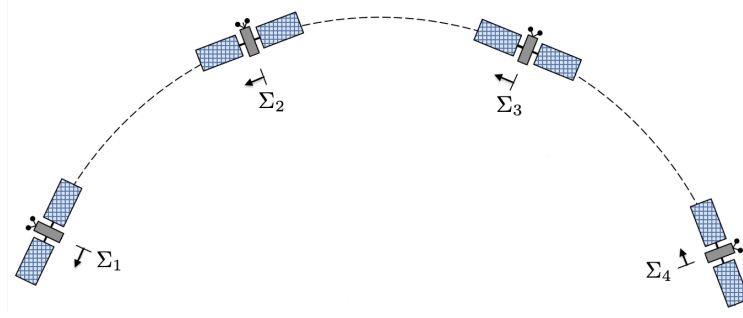
Figure 6.1: Constellation where each satellite shares state information with neighbors via communication links

sive with respect to proposed storage functions. A constellation coordination control law is introduced to interconnect the subsystems in a skew-symmetric coupling structure. The equilibrium-independent passivity property of each subsystem and the skew-symmetry of their interconnection enables us to prove the stability of the constellation at equilibrium.

Preliminaries

We use a compositional approach to certify the stability of a large system consisting of interconnected, dissipative subsystems. We briefly state results that extend the works in [56–58], which are used in a later section to prove stability of the constellation under a closed-loop formation and station-keeping control law. Consider the system Σ described by

$$\dot{x}(t) = f(t, x(t), u(t)) , \quad y(t) = h(t, x(t), u(t)) , \tag{6.1}$$

where $x(t) \in \mathbb{R}^{n_x}$ is the state, $u(t) \in \mathbb{R}^{n_u}$ is the input, and $y(t) \in \mathbb{R}^{n_y}$ is the output. Furthermore, suppose there exists a nonempty set $\mathcal{X} \subset \mathbb{R}^{n_x}$ where, for every $\bar{x} \in \mathcal{X}$, there exists a unique $\bar{u} \in \mathbb{R}^{n_u}$ satisfying $f(t, \bar{x}, \bar{u}) = 0$

Definition 1. The system (6.1) is *equilibrium independent dissipative* (EID) with supply rate $s(\cdot, \cdot)$ if there exist continuously differentiable functions $V : \mathbb{R} \times \mathbb{R}^{n_x} \times \mathcal{X} \mapsto \mathbb{R}$ and $\underline{V} : \mathbb{R}^{n_x} \times \mathcal{X} \mapsto \mathbb{R}$ satisfying the conditions

$$V(t, x, \bar{x}) \geq \underline{V}(x, \bar{x}) > 0 \quad \forall (x, \bar{x}) \text{ s.t. } x \neq \bar{x}, \tag{6.2a}$$

$$V(t, \bar{x}, \bar{x}) = 0, \quad \underline{V}(\bar{x}, \bar{x}) = 0, \tag{6.2b}$$

$$\dot{V}(t, x, \bar{x}) := \nabla_t V(t, x, \bar{x}) + \nabla_x V(t, x, \bar{x})^\top f(t, x, u) \leq s(u - \bar{u}, y - \bar{y}),$$

$$\forall (t, x, \bar{x}, u, \bar{u}) \in \mathbb{R} \times \mathbb{R}^{n_x} \times \mathcal{X} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_u} \text{ , where } \bar{y} = h(t, \bar{x}, \bar{u}). \tag{6.2c}$$

A system is *equilibrium-independent passive* (EIP) if it is EID with respect to the supply rate

$$s(u - \bar{u}, y - \bar{y}) = (u - \bar{u})^\top (y - \bar{y}) \tag{6.3}$$

and it is *output strictly equilibrium-independent passive* (OSEIP) if, for some $\epsilon > 0$, it is EID with respect to

$$s(u - \bar{u}, y - \bar{y}) = (u - \bar{u})^\top (y - \bar{y}) - \epsilon(y - \bar{y})^\top (y - \bar{y}) \ . \tag{6.4}$$

6.2 System Dynamics

Instead of creating a monolithic model of the constellation, we decompose it into subsystems and consider the interconnections between them. By characterizing the input-output properties of each individual subsystem and the interconnections that exist between them, we may certify stability and convergence properties of the constellation.

Planar Satellite Model

In our constellation, we refer to the constituent satellites as subsystems. Each satellite is under the influence of the gravitational pull from the central body, the thrust applied by the satellite, and natural perturbing forces (e.g., atmospheric drag, gravity from moons, solar radiation pressure). To model the motion of a satellite orbiting a planet, we start with the central-force problem (or restricted two-body problem) where we assume that the barycenter of the system is co-located with the center of a spherically, symmetric central body (i.e., the mass of the satellite is negligible). The satellite's motion can be described by the following second-order ordinary differential equation known as the fundamental orbital differential equation (FODE) with specific force perturbations [59]:

$$\ddot{\mathbf{r}} = -\frac{\mu}{\|\mathbf{r}\|_2^3} \mathbf{r} + \frac{1}{m} \boldsymbol{\tau} + \mathbf{a}_{perturb} \ , \tag{6.5}$$

where $\mathbf{r} \in \mathbb{R}^3$ is the position vector pointing from the center of the planet to the satellite, μ is the gravitational parameter of the central body (i.e., gravitational constant multiplied by the mass of the planet), m is the mass of the satellite, $\boldsymbol{\tau} \in \mathbb{R}^3$ is thrust, and $\mathbf{a}_{perturb} \in \mathbb{R}^3$ represents the specific forces due to perturbations.

It is well known that two-body motion in an inertial frame is planar. Since atmospheric drag acts against the direction of motion, a satellite under atmospheric drag remains in planar motion. Furthermore, if a satellite and the moons of a planet lie in the same plane (e.g., equatorial plane), then the gravitational perturbations from the moons may be approximated as planar. Hence, for certain examples, we may use a polar coordinate system to represent the satellite orbital kinematics in the plane:

$$\mathbf{r} = r\mathbf{e}_r \tag{6.6a}$$

$$\dot{\mathbf{r}} = \dot{r}\mathbf{e}_r + r\dot{\theta}\mathbf{e}_\theta \tag{6.6b}$$

$$\ddot{\mathbf{r}} = \left(\ddot{r} - r\dot{\theta}^2\right)\mathbf{e}_r + \left(2\dot{r}\dot{\theta} + r\ddot{\theta}\right)\mathbf{e}_\theta \ . \tag{6.6c}$$

We denote the magnitude of the radial position with r and the angular position with θ. We use $\mathbf{e}_r$ and $\mathbf{e}_\theta$ as the unit vectors in the radial and tangential directions of the orbital plane, respectively.

If we include the specific forces from the right-hand side of (6.5), we get the following model representing the ith satellite's motion in the radial and tangential directions, respectively:

$$\ddot{r}_i = r_i\dot{\theta}_i^2 - \frac{\mu}{r_i^2} + \frac{1}{m_i}\tau_{r,i} + (\vec{a}_{perturb,i})_r \tag{6.7a}$$

$$\ddot{\theta}_i = \frac{-2\dot{r}_i\dot{\theta}_i}{r_i} + \frac{1}{m_i r_i}\tau_{\theta,i} + \frac{1}{r_i}(\vec{a}_{perturb,i})_\theta \; . \tag{6.7b}$$

Finally, if we implement a change of variables so that $v := \dot{r}$ and $\omega := \dot{\theta}$, we get the following set of first-order differential equations to describe each satellite of the constellation

$$\dot{r}_i = v_i \tag{6.8a}$$

$$\dot{v}_i = r_i\omega_i^2 - \frac{\mu}{r_i^2} + \frac{1}{m_i}\tau_{r,i} \tag{6.8b}$$

$$\dot{\omega}_i = \frac{-2v_i\omega_i}{r_i} + \frac{1}{m_i r_i}\tau_{\theta,i} \; . \tag{6.8c}$$

Note that we exclude $\dot{\theta}_i = \omega_i$ from the set of equations. The θ state does not appear in the equations of motion (6.8), hence, it is not needed in our state feedback controller design. Furthermore, we omit the terms representing specific forces due to perturbations. Through an example simulation we will show that our state feedback controller based on the model described by (6.8) is robust to unmodeled disturbances that are present in the simulation model, described by (6.7).

Interconnections

We assume that only neighboring satellites may communicate with each other. The topology of this particular information exchange is illustrated by the undirected graph shown in Fig. 6.1. If the ith and j^{th} subsystems have access to relative state information, then the ith and j^{th} nodes of the graph are connected by a link $l = 1, \ldots, M$. Although the communication is assumed to be bidirectional, we assign an orientation to the graph by considering one of the nodes of a link to be the positive end. As a convention, we set the direction of a communication link to point in the direction of the orbital motion. Hence, the incidence matrix D of the graph is defined as:

$$D_{il} = \begin{cases} +1 & \text{if } i^{th} \text{ node is positive end of } l^{th} \text{ link} \\ -1 & \text{if } i^{th} \text{ node is negative end of } l^{th} \text{ link} \\ 0 & \text{otherwise.} \end{cases}$$

In this application, for a constellation with N satellites that only communicate with neighbors, the incidence matrix D is

$$
D = \begin{bmatrix} 1 & 0 & 0 \\ -1 & \ddots & 0 \\ 0 & \ddots & 1 \\ 0 & 0 & -1 \end{bmatrix} \in \mathbb{R}^{N \times M} \ ,
\tag{6.9}
$$

where $M := N - 1$. Note that we assume the 1st and N^{th} satellites do not communicate; hence, they do not share a communication link. All other satellites have two links each.

6.3 Control Strategy

We now describe an internal feedback control strategy for each satellite that renders a linear map between the input (to be designed with a simple state feedback law) and the output variable of interest. Subsequently, we add a *constellation coordination* term that regulates the relative angular spacing error between neighboring satellites.

Internal Feedback Control

For each subsystem, we propose the following thrust control laws in the radial and tangential directions:

$$
\tau_{r,i} = m_i \left(-r_i \omega_i^2 + \frac{\mu}{r_i^2} \right) - k_v(v_i - v_d) - k_r(r_i - r_d)
\tag{6.10a}
$$

$$
\tau_{\theta,i} = m_i \left(2v_i \omega_i - k_\omega(\omega_i - \omega_d) + \frac{r_i}{k_c} u_i \right),
\tag{6.10b}
$$

where r_d, v_d, and ω_d are the desired radius, radial velocity, and angular velocity for every satellite to maintain an areostationary orbit. The term u_i is a constellation coordination control law to be designed. The controller gains k_r, k_v, k_ω, $k_c > 0$ are discussed and chosen in the subsequent stability analysis and simulation results.

If we substitute the thrust control laws (6.10a)-(6.10b) into the equations of motion (6.8a)-(6.8c), the dynamics of each satellite, Σ_i for $i = 1, \ldots, N$, take the form of

$$
\dot{r}_i = v_i
\tag{6.11a}
$$

$$
\dot{v}_i = -k_v(v_i - v_d) - k_r(r_i - r_d)
\tag{6.11b}
$$

$$
\dot{\omega}_i = -\frac{k_\omega}{r_i}(\omega_i - \omega_d) + \frac{1}{k_c} u_i
\tag{6.11c}
$$

$$
z_i = \omega_i \ ,
\tag{6.11d}
$$

where the output variable z_i of interest is the angular velocity of the satellite. Note that we have transformed the radial dynamics (6.11a) - (6.11b) to be independent of the ω state.

Constellation Coordination Control

The subsystems are dynamically decoupled, however, we may coordinate their relative motion through a constellation coordination control law where we use feedback of local information from spatially neighboring subsystems. We assume that this local information is shared via inter-satellite communication links [60]- [61]. The links can be expressed as subsystems Λ_l for $l = 1, \ldots, M$:

$$\dot{\theta}_l^{rel} = e_l \tag{6.12a}$$

$$y_l = h_l(\theta_l^{rel}) \ , \tag{6.12b}$$

where e_l is the input and y_l is the output of each communication link. The subsystem Λ_l keeps track of a state $\theta_l^{rel} \in \mathbb{R}$ and outputs a signal of interest that is measured through the function $h_l : \mathbb{R} \mapsto \mathbb{R}$. We assume h_l is strictly increasing and onto, and that $\lim_{a \to \infty} h_l(a) = \infty$.

Let us refer to satellite inputs and outputs in compact form as $u := [u_1, \ldots, u_N]^\top$ and $z := [z_1, \ldots, z_N]^\top$, respectively. Similarly, we refer to the communication link inputs and outputs collectively as $e := [e_1, \ldots, e_M]^\top$ and $y := [y_1, \ldots, y_M]^\top$, respectively.
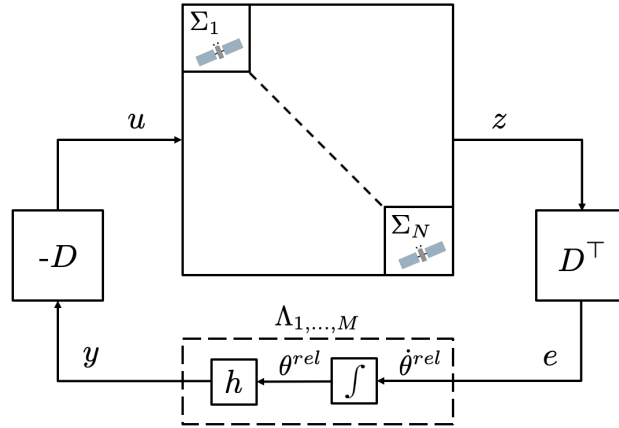
Figure 6.2: Interconnected system

We construct an interconnection between the satellites $\Sigma_1, \ldots, \Sigma_N$ and the communication links $\Lambda_1, \ldots, \Lambda_M$ as shown in Fig. 6.2 and define the following input-output mappings:

$$e := D^\top z = \begin{bmatrix} \omega_1 - \omega_2 \\ \omega_2 - \omega_3 \\ \vdots \\ \omega_{N-1} - \omega_N \end{bmatrix} \equiv \begin{bmatrix} \dot{\theta}_1^{rel} \\ \dot{\theta}_2^{rel} \\ \vdots \\ \dot{\theta}_M^{rel} \end{bmatrix} =: \dot{\theta}^{rel} \tag{6.13a}$$

$$u := -Dy = -D \begin{bmatrix} h_1(\theta_1^{rel}) \\ \vdots \\ h_M(\theta_M^{rel}) \end{bmatrix} = -Dh(\theta^{rel}) \ . \tag{6.13b}$$

Note that the input applied to the ith satellite,

$$u_i = -\sum_{l=1}^{M} D_{il} h_l(\theta_l^{rel}) \,, \tag{6.14}$$

is based only on local information since $D_{il} = 0$ when the ith subsystem does not have access to information on the lth communication link. Hence, we have a distributed control architecture where local controllers act on local information.

6.4 Stability Analysis

We first show the existence and uniqueness of an equilibrium point whose stability will be subsequently analyzed. At equilibrium, the right-hand sides of (6.11a), (6.11b), (6.11c) for all $i = 1, \ldots, N$, and (6.12a) for all $l = 1, \ldots, M$ must equal zero. The equilibrium states of the radial dynamics (6.11a)–(6.11b) may be found by inspection to be $(\bar{r}_i, \bar{v}_i) = (r_d, v_d) = (r_d, 0)$. For the right-hand side of (6.12a) to vanish, e_l must equal zero for $l = 1, \ldots M$. In other words,

$$\bar{e} = D^\top \bar{\omega} = \mathbf{0}. \tag{6.15}$$

By definition of D given in (6.9), we have $D^\top \mathbf{1} = 0$. Since $nullity(D^\top) = 1$, the span of $\mathbf{1}$ constitutes the entire null space of $D^\top$. Therefore, $\bar{\omega} = \omega_0 \mathbf{1}$ is the unique solution to (6.15), where ω_0 is the common angular velocity of all N satellites. That is, all satellites must have the same angular velocity. Finally, the right-hand side of (6.11c) must vanish:

$$-\frac{k_\omega}{r_i}(\omega_0 - \omega_d) + \frac{1}{k_c}\bar{u}_i = 0, \text{ for } i = 1, ..., N. \tag{6.16}$$

From (6.13b) and the fact that $\mathbf{1}^\top D = \mathbf{0}^\top$, we have $\sum_{i=1}^{N} u_i = \mathbf{1}^\top u = -\mathbf{1}^\top D h(\theta^{rel}) = 0$. Adding (6.16) from $i = 1$ to $i = N$ yields the following equation:

$$-(\omega_0 - \omega_d) \sum_{i=1}^{N} \frac{k_\omega}{r_i} = 0,$$

which requires that $\omega_0 = \omega_d$, and therefore $\bar{\omega} = \omega_d \mathbf{1}$. Substituting this value for ω_0 back into (6.16), we get

$$\bar{u}_i = -\sum_{l=1}^{M} D_{il} h_l(\bar{\theta}_l^{rel}) = 0 \text{ for } i = 1, ..., N, \tag{6.17}$$

which amounts to

$$h_1(\bar{\theta}_1^{rel}) = 0,$$
$$-h_{l-1}(\bar{\theta}_{l-1}^{rel}) + h_l(\bar{\theta}_l^{rel}) = 0, \ l = 2, ..., M, \tag{6.18}$$
$$-h_M(\bar{\theta}_M^{rel}) = 0.$$

A solution $\bar{\theta}_l^{rel}$ for $l = 1, \ldots, M$ exists and is unique since h_l is onto and strictly increasing. In summary, there exists a unique equilibrium point for a desired constellation given by $(\bar{r}_i, \bar{v}_i, \bar{\omega}_i) = (r_d, 0, \omega_d)$, $i = 1, \ldots, N$ and $\bar{\theta}_l^{rel}$, $l = 1, \ldots, M$ that satisfy (6.18). Furthermore, we note that $\omega_d = \sqrt{\mu/r_d^3}$ for a circular orbit at a given altitude.

We use a compositional approach to analyze the stability properties of the closed-loop constellation under our proposed internal feedback and coordination control laws. First, we show the stability of an equilibrium point for the radial component of each individual Σ_i subsystem (6.11a)–(6.11b). Second, we propose storage functions for each of the interconnected subsystems, comprised of the tangential component of the Σ_i subsystems (6.11c)–(6.11d), $i = 1, \ldots, N$ and the Λ_l subsystems (6.12), $l = 1, \ldots, M$, and certify that they are EID as defined in (6.2). We then use the storage functions to compose a Lyapunov function for the interconnected system.

For the radial component of the Σ_i subsystem (6.11a) - (6.11b), we choose k_r, k_v so that the closed-loop system is stable. We define $r_i^e = r_i - \bar{r}_i$, and $v_i^e = v_i - \bar{v}_i = v_i$, then (6.11a) and (6.11b) can be rewritten as

$$\begin{bmatrix} \dot{r}_i^e \\ \dot{v}_i^e \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k_r & -k_v \end{bmatrix} \begin{bmatrix} r_i^e \\ v_i^e \end{bmatrix}. \tag{6.19}$$

It can be verified that the equilibrium point $(\bar{r}_i, \bar{v}_i)$ of (6.11a)–(6.11b) is exponentially stable if and only if $k_r > 0$ and $k_v > 0$.

We now proceed to prove stability of the tangential component of the subsystems under the influence of both the internal feedback law (6.10b) and the constellation coordination law (6.13b). In the internal feedback law (6.10b), we utilize a positive parameter k_c to scale down the magnitude of the constellation coordination control input u_i. More specifically, we assume that k_c is a time-varying parameter:

$$k_c(t) \geq \underline{k}_c > 0, \ \dot{k}_c(t) \leq 0, \ \forall \, t \geq 0, \tag{6.20}$$

that decreases and converges to a positive limit $\underline{k}_c$.

We propose the following storage function for the ith subsystem:

$$S_i(t, \omega_i, \bar{\omega}_i) = \frac{k_c(t)}{2}(\omega_i - \bar{\omega}_i)^2 \ . \tag{6.21}$$

We can verify that $S_i(t, \omega_i, \bar{\omega}_i) \geq \frac{1}{2}\underline{k}_c(\omega_i - \bar{\omega}_i)^2 > 0$, for all $(\omega_i, \bar{\omega}_i)$ such that $\omega_i \neq \bar{\omega}_i$, and that $S_i(t, \bar{\omega}_i, \bar{\omega}_i) = 0$.

If we take the derivative of the storage function we get

$$\dot{S}_i(t, \omega_i, \bar{\omega}_i) = k_c(t)(\omega_i - \bar{\omega}_i)\dot{\omega}_i + \frac{\dot{k}_c(t)}{2}(\omega_i - \bar{\omega}_i)^2$$

$$= k_c(t)(\omega_i - \bar{\omega}_i)\left(-\frac{k_\omega}{r_i}(\omega_i - \omega_d) + \frac{1}{k_c(t)}u_i\right) +$$

$$\frac{\dot{k}_c(t)}{2}(\omega_i - \bar{\omega}_i)^2 \tag{6.22}$$

$$= (u_i - \bar{u}_i)(\omega_i - \bar{\omega}_i) - \left(\frac{k_c(t)k_\omega}{r_i} - \frac{\dot{k}_c(t)}{2}\right)(\omega_i - \bar{\omega}_i)^2 \tag{6.23}$$

where we have used $\bar{\omega}_i = \omega_d$, $\bar{u}_i = -\sum_{l=1}^{M} D_{il}h_l(\bar{\theta}_l^{rel}) = 0$. We note that $r_i(t) > 0$, $\forall i = 1, ..., N$ is always satisfied (i.e., the radius is always positive). Hence, the storage function S_i, described by (6.21), certifies that the tangential component of the Σ_i subsystems (6.11c)–(6.11d), is OSEIP, as defined in (6.4).

For the links Λ_l, we propose

$$T_l(\theta_l^{rel}, \bar{\theta}_l^{rel}) = \int_{\bar{\theta}_l^{rel}}^{\theta_l^{rel}} \left(h_l(z) - h_l(\bar{\theta}_l^{rel})\right) dz. \tag{6.24}$$

Since h_l is strictly increasing, we can verify that $T_l(\theta_l^{rel}, \bar{\theta}_l^{rel}) > 0$ for all $\theta_l^{rel} \neq \bar{\theta}_l^{rel}$ and $T_l(\bar{\theta}_l^{rel}, \bar{\theta}_l^{rel}) = 0$.

If we take the derivative of the storage function we get

$$\dot{T}_l(\theta_l^{rel}, \bar{\theta}_l^{rel}) = \dot{\theta}_l^{rel} \left(h_l(\theta_l^{rel}) - h_l(\bar{\theta}_l^{rel})\right)$$
$$= (e_l - \bar{e}_l)(y_l - \bar{y}_l) \tag{6.25}$$

where we have used $\bar{e}_l = \sum_{i=1}^{N} D_{il}\bar{z}_i = \sum_{i=1}^{N} D_{il}\bar{\omega}_i = 0$ and $\bar{y}_l = h_l(\bar{\theta}_l^{rel})$. We note that the storage function T_l certifies that each communication link Λ_l is EIP as defined in (6.3).
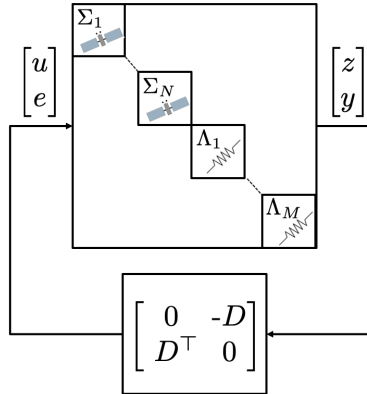
Figure 6.3: Interconnected system in canonical form

Now that we have shown that each of the subsystems is equilibrium-independent passive, we note that the interconnected system as shown in Fig. 6.2 may be brought into the canonical form of Fig. 6.3 where the upper block has the subsystems along its diagonal and the lower block contains a skew symmetric matrix. As shown in [56], since the

equilibrium-independent passive subsystems are coupled through a skew symmetric inter-connection matrix, an equilibrium point of the interconnected system, if it exists, is stable and the sum of the individual subsystems provides a Lyapunov function.

Let us sum the storage functions for all the Σ_i subsystems and Λ_l subsystems:

$$V(t, x, \bar{x}) = \sum_{i=1}^{N} S_i(t, \omega_i, \bar{\omega}_i) + \sum_{l=1}^{M} T_l(\theta_l^{rel}, \bar{\theta}_l^{rel}) \tag{6.26}$$

where we use $x := (\omega, \theta^{rel})$, $\bar{x} := (\bar{\omega}, \bar{\theta}^{rel})$. The time-varying Lyapunov function (6.26) can be lower and upper bounded:

$$\underline{V}(x, \bar{x}) \leq V(t, x, \bar{x}) \leq \bar{V}(x, \bar{x}) \ , \tag{6.27}$$

where

$$\underline{V}(x, \bar{x}) = \sum_{i=1}^{N} \frac{k_c}{2}(\omega_i - \bar{\omega}_i)^2 + \sum_{l=1}^{M} T_l(\theta_l^{rel}, \bar{\theta}_l^{rel}) \tag{6.28}$$

$$\bar{V}(x, \bar{x}) = \sum_{i=1}^{N} \frac{\bar{k}_c}{2}(\omega_i - \bar{\omega}_i)^2 + \sum_{l=1}^{M} T_l(\theta_l^{rel}, \bar{\theta}_l^{rel}) \tag{6.29}$$

and $\bar{k}_c := k_c(0) \geq k_c(t) \geq \underline{k}_c \ \forall t \geq 0$. We note that $\underline{V}(x, \bar{x})$ and $\bar{V}(x, \bar{x})$ are positive definite and radially unbounded.

If we take the time derivative of (6.26), we get:

$$\dot{V}(t, x, \bar{x}) = \sum_{i=1}^{N} \dot{S}_i + \sum_{l=1}^{M} \dot{T}_l$$

$$= \sum_{i=1}^{N} \left\{ (u_i - \bar{u}_i)(\omega_i - \bar{\omega}_i) - \left(\frac{k_c(t)k_\omega}{r_i} - \frac{\dot{k}_c(t)}{2} \right)(\omega_i - \bar{\omega}_i)^2 \right\}$$

$$+ \sum_{l=1}^{M} \left\{ (e_l - \bar{e}_l)(y_l - \bar{y}_l) \right\}.$$

Let us define $R := blkdiag(r_1, ..., r_N)$, and use $\bar{e} = D^\top \bar{\omega}$ in the expression above to get

$$= -k_c(t)k_\omega(\omega - \bar{\omega})^\top R^{-1}(\omega - \bar{\omega}) + \frac{\dot{k}_c(t)}{2}(\omega - \bar{\omega})^\top(\omega - \bar{\omega})$$

$$+ (\omega - \bar{\omega})^\top(u - \bar{u}) + (\omega - \bar{\omega})^\top D(y - \bar{y}).$$

Finally, if we use our constellation coordination control law (6.13b) and $\bar{u} = -D\bar{y}$, then

$$= -k_c(t)k_\omega(\omega - \bar{\omega})^\top R^{-1}(\omega - \bar{\omega}) + \frac{\dot{k}_c(t)}{2}(\omega - \bar{\omega})^\top(\omega - \bar{\omega}). \tag{6.30}$$

Note that the expression above is negative semi-definite. As a result, $(\bar{r}_i, \bar{v}_i, \bar{\omega}_i, \bar{\theta}_l^{rel}) = (r_d, 0, \omega_d, \bar{\theta}_l^{rel})$, for all Σ_i, $i = 1, \ldots N$ and all Λ_l, $l = 1, \ldots M$ is a stable equilibrium point of the interconnected system shown in Fig. 6.2, where $\bar{\theta}_l^{rel}$ satisfies equations (6.18).

Due to the time-varying parameters r and k_c, the interconnected constellation is a non-autonomous system for which the Lasalle-Krasovskii Invariance Principle is not applicable. Although we may not conclude asymptotic stability of an equilibrium, we may prove the weaker result [62] that ω_i, $i = 1, \ldots, N$ converges to the desired ω_d value. Physically, this signifies that the constellation will maintain a circular orbit.

As shown in [62], $x(t)$ is bounded by using (6.27) and the dynamics are locally Lipschitz in x and bounded in t, implying that $\dot{x}(t)$ is also bounded for all $t \geq 0$. Hence, $x(t)$ is uniformly continuous for $t \geq 0$. Define a negative semi-definite function

$$W(x) = -\underline{k}_c k_\omega (\omega - \bar{\omega})^\top R^{-1} (\omega - \bar{\omega}). \tag{6.31}$$

As a result, $W(\cdot)$ is uniformly continuous on the bounded domain of $x(t)$. From (6.30) we can verify that

$$\dot{V}(t, x(t), \bar{x}) \leq W(x(t)).$$

Integrating the expression over $[0, T]$, we get

$$V(T, x(T), \bar{x}) - V(0, x(0), \bar{x}) \leq \int_0^T W(x(t)) dt ,$$

which implies

$$-\int_0^\infty W(x(t)) dt \leq V(0, x(0), \bar{x}) < \infty.$$

Using Barbalat's Lemma, since $W(\cdot)$ is uniformly continuous and $\int_0^\infty W(x(t)) dt$ exists, $W(x(t)) \to 0$ as $t \to \infty$, which implies that $x(t)$ approaches $E = \{x : W(x) = 0\}$. In other words, $\omega_i(t) \to \bar{\omega}_i = \omega_d$.

6.5 Numerical Example

Consider a cluster of $N = 10$ satellites that have been batch deployed into a nearly-circular, equatorial, prograde orbit around the planet Mars at a desired altitude of approximately $17\,032$ km above the Martian surface. Assuming the equatorial radius of Mars is 3396.2 km, each satellite in this orbit has desired equilibrium states of $(r_d, v_d, \omega_d) = (r_d, 0, \sqrt{\mu/r_d^3})$ where $r_d = 20\,428.2$ km. This specific orbit, from the class of areosynchronous (i.e., Martian synchronous) orbits, is known as an areostationary orbit. Similar to satellites in geostationary orbit about Earth, the position of an areostationary satellite appears fixed in the sky relative to an observer on the surface of Mars. By equally spacing the 10 satellites within this orbit,

the resulting constellation may serve as a telecommunication network or navigation system for the exploration of Mars.

After deployment we assume the following initial conditions for all $i = 1, \ldots, N$ satellites: $r_i = (20\,428.0 \pm 0.1)\,\text{km}$, $v_i = (0 \pm 1) \times 10^{-8}\,\text{m s}^{-1}$, $\omega_i = (7.0879 \pm 0.0100) \times 10^{-5}\,\text{rad s}^{-1}$, $\theta_i = (0 \pm 5) \times 10^{-3}\,\text{rad}$. Note that the initial conditions prescribe nearly circular orbits. The angular position θ_i is measured with respect to a reference horizontal line in the orbital plane.

We assume each $m = 100\,\text{kg}$ satellite is equipped with a throttleable, continuous-thrust propulsion system with a maximum thrust of $\tau_{max} = 100\,\text{mN}$ in each of the radial and tangential directions of motion. In this example, we do not consider motion normal to the orbital plane. Solar electric propulsion systems, which use electricity generated by solar panels to accelerate propellant at high exhaust speeds, are capable of throttleable, continuous-thrust. Although electric propulsion systems have high specific impulse (i.e., they are fuel efficient), they have much weaker thrust compared to traditional chemical rockets. The NASA Evolutionary Xenon Thruster [63] is an example of a solar electric propulsion system with a maximum thrust of $236\,\text{mN}$. We expect that the state-of-the-art will continue to develop, allowing for even higher thrust magnitudes in the future, but we maintain a conservative thrust limit for this example.

In addition to the gravitational pull of Mars, we introduce perturbations due to the gravity of Mars' two moons. Since the inclinations of Phobos and Deimos with respect to Mars' equator are $1.093°$ and $0.930°$, respectively, we approximate their orbits as equatorial in this example. Note that since Phobos and Deimos have orbital eccentricities of 0.0151 and 0.0003, respectively, their orbits are nearly circular. We use the values of $9234.42\,\text{km}$ and $23\,455.50\,\text{km}$ for the radial distance of each moon's orbit at its respective periapsis. Finally, we use values of $\mu = 4.282\,837 \times 10^{13}\,\text{m}^3\,\text{s}^{-2}$, $\mu_{Phobos} = 7.161 \times 10^5\,\text{m}^3\,\text{s}^{-2}$, and $\mu_{Deimos} = 1.041 \times 10^5\,\text{m}^3\,\text{s}^{-2}$ for the standard gravitational parameter of Mars, Phobos, and Deimos, respectively. We find the specific force perturbation acting on each satellite by each moon, $\vec{a}_{p,i}$ (where $p = \{Phobos, Deimos\}$), by computing

$$\begin{bmatrix} (\vec{a}_{p,i})_r \\ (\vec{a}_{p,i})_\theta \end{bmatrix} = -\frac{\mu_p}{\|\vec{r}_{p,i}\|_2^3} \begin{bmatrix} \cos\theta_i & \sin\theta_i \\ -\sin\theta_i & \cos\theta_i \end{bmatrix} \vec{r}_{p,i} \;, \tag{6.32}$$

where $\vec{r}_{p,i}$, the expression for the relative position of the ith satellite with respect to the moon p in the Mars-centered inertial coordinate system, is

$$\vec{r}_{p,i} = \begin{bmatrix} r_i \cos\theta_i - r_p \cos\theta_p \\ r_i \sin\theta_i - r_m \sin\theta_p \end{bmatrix} \;. \tag{6.33a}$$

The radial and tangential components of the acceleration are found by rotating $\vec{r}_{p,i}$ by the appropriate rotation matrix.

The *mission objectives* are (1) spread out the initial cluster of satellites into an equally-spaced constellation, and (2) regulate the satellites' deviations from the desired areostationary orbit as well as their relative angular positions with respect to the desired spacings, in

the presence of unmodeled perturbations. We call these distinct phases of the mission as *formation* and *station-keeping*.

In the formation phase, we consider a generous formation time of $t_f = 355$ Martian days (Sols), or approximately 1 Earth year. Although the constellation may be acquired in less time, it may not be necessary. In various design proposals for manned missions to explore Mars [64], plans include an initial uncrewed cargo mission so that supplies and infrastructure are in place before the crewed missions arrive. We assume that a satellite constellation to serve as a telecommunications network would be launched in this initial mission. Given that subsequent crewed missions would require approximately two years to arrive, due to launch window constraints, 1 Earth year would provide sufficient time to deploy and test the satellite constellation before use by a crewed mission.

6.6   Results

We implement the thrust controls laws described by (6.10) where the formation control law u_i for all $i = 1, \ldots, N$ satellites is given by (6.14) and the interconnection between satellites is described by the incidence matrix D in (6.9). In this example, the measurement output from each of the communication links, $h_l(\theta_l^{rel})$, $l = 1, \ldots, M$, in (6.12b) is of the form:

$$h_l(\theta_l^{rel}) = \theta_l^{rel} - \theta_d^{rel}, \tag{6.34}$$

where $\theta_d^{rel} = \frac{2\pi}{N}$ represents the desired, equal angular spacing between neighboring satellites. The model (6.7) is used for simulation where the specific force perturbations due to Phobos and Deimos are included using (6.32).

To regulate the radial distance, radial velocity, and angular velocity of each satellite about the areostationary orbit, we use the gains $k_r = 1 \times 10^{-5}$, $k_v = 1 \times 10^{-4}$, and $k_w = 1 \times 10^4$. In the formation phase ($0 \le t \le t_f$), we use a time-varying constellation coordination gain

$$k_c(t) = (\bar{k}_c - \underline{k}_c) \exp(-\frac{c}{t_f}t) + \underline{k}_c \,, \tag{6.35}$$

where $\bar{k}_c > \underline{k}_c > 0$ and $c > 0$. We can simply calculate the time derivative of k_c as

$$\dot{k}_c(t) = -\frac{c}{t_f}(\bar{k}_c - \underline{k}_c) \exp(-c\frac{t}{t_f}) < 0, \ \forall \ t \ge 0 \,. \tag{6.36}$$

Note that the constellation coordination gain function, (6.35), satisfies the condition in (6.20) used for the stability analysis. For this example, we choose $\bar{k}_c = 1 \times 10^{11}$, $\underline{k}_c = 1 \times 10^9$, $c = 30$. Since the relative angle θ_l^{rel} is far from the desired relative angle θ_d^{rel} at the beginning of the formation phase, the magnitude of control input u_i derived with (6.34) is large. We initially need a large k_c to scale it down. As θ_l^{rel} converges to θ_d^{rel}, the magnitude of u_i decreases and we require less scaling. Therefore, the constantly decreasing parameter k_c allows the thrust commands $\tau_{r,i}$ and $\tau_{\theta,i}$ in (6.10) to stay within a reasonable range during
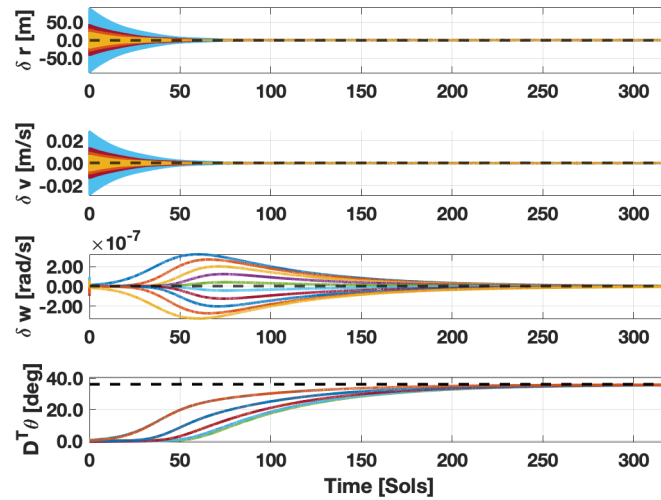
Figure 6.4: Relative states and angular spacing between neighboring satellites
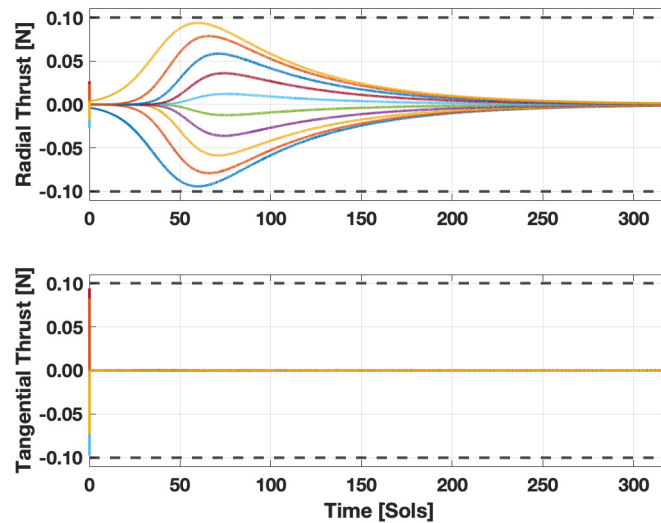
Figure 6.5: Radial and tangential thrust commands to each satellite during phasing

the formation phase. After formation, we enter the station-keeping phase where we use a constant value of $\underline{k}_c$.

The simulated states of each satellite are shown in the first three subplots of Fig. 6.4. Despite the perturbed initial conditions and the specific force perturbations due to Phobos and Deimos, each satellite regulates to the desired equilibrium point for an areostationary orbit (illustrated by the dotted lines). The fourth subplot of Fig. 6.4 shows that the angular
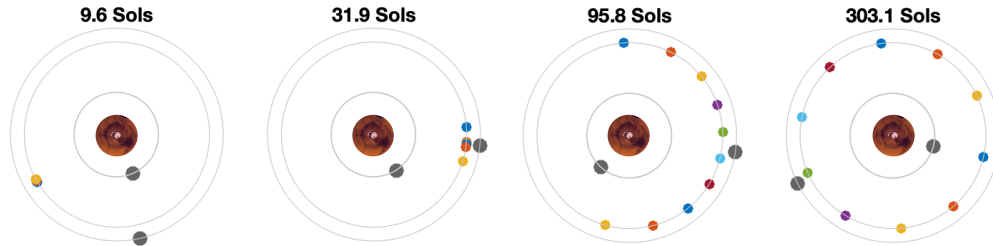
Figure 6.6: Stages of areostationary constellation phasing

spacing between each pair of satellites reaches the desired value of 36°. All angular spacings reach within a 0.5° tolerance of the desired value in 303.06 Sols (or approximately 311 solar Earth days).

In Fig. 6.5, we plot the radial and tangential thrust inputs commanded by our feedback laws (6.10). We observe that the control histories remain within the maximum thrust value of 100 mN throughout the formation phase. We also note that, although the constellation coordination term appears in the tangential thrust control law, most of the control action occurs in the radial direction. This behavior signifies that the ω_i^2 term in the radial thrust law (6.10a) dominates the other terms. The controller exhibits the same strategy as traditional station-keeping methods where orbital phasing maneuvers (i.e., adjusting a satellite's position within an orbit) can be conducted by decreasing (increasing) the altitude of a spacecraft, causing it to speed up (slown down) in the tangential direction to gain (reduce) angular position.

Finally, we present Fig. 6.6, where the angular positions of the satellites are depicted at different times during the formation phase. The central red body represents Mars whereas the two gray bodies are the moons, Phobos and Deimos. We note that the orbit of the outer moon, Deimos, is very close to that of the areostationary orbit at a distance of approximately 3000 km. Despite the close proximity, the effect of the unmodeled gravitational perturbation is mitigated by the proposed control law. An animation of the formation phase is available at `https://youtu.be/-2y_IWRPuzU`.

6.7   Summary

We have presented a control strategy to coordinate a large number of satellites to not only acquire but also to maintain an equally-spaced constellation in areostationary orbit. The proposed distributed control law is implemented on each satellite using only local information from neighboring satellites. We proved that the closed-loop system, comprised of the satellites and communication links, is stable at equilibrium due to the equilibrium-independent passive property of each subsystem and the skew-symmetric coupling structure of their interconnections. We further proved that the angular velocities of each satellite converge to

the desired value necessary for a circular, areostationary orbit. We then demonstrated the efficacy of the formation and station-keeping control strategy in simulation.

Regarding the practical implementation of our approach to constellation formation and station-keeping, we note that although the proposed control strategy is not optimal (with respect to a minimum-formation-time or minimum-fuel objective), it is a simple, distributed, and computationally inexpensive approach that may be tuned to achieve specific mission constraints on time or fuel. Given the time and maximum thrust constraints of our example mission, our simulation results showed that the commanded thrust profiles are achievable with the current state-of-the-art in electric propulsion. We also note that the proposed strategy exhibits robustness to perturbed initial conditions and unmodeled disturbances. Future work will investigate delay robustness although we do not deem the communication delay between satellites to be significant relative to the slow time scales in which the constellation evolves in our example. If we assume that communication delay is proportional to inter-satellite link distance, the worst delay is when the areostationary constellation is completely acquired and the 10 satellites are equally spaced with a line-of-sight distance of $12\,625\,\mathrm{km}$ between each pair. Considering that the delay between a ground station and a geostationary satellite at an altitude of $36\,000\,\mathrm{km}$ is approximately a quarter of a second, we can deduce that the communication delay between our satellites will be relatively small compared to the time it takes a circular, areostationary orbit to be influenced by low-thrust propulsion or the time allowed in the constellation formation phase.

Chapter 7

Conclusion

The main contribution of this dissertation is the formulation of spacecraft attitude and orbit control problems relevant to the formation and operation of small spacecraft constellations. We have considered applications ranging from constellations in low Earth orbit and Mars areostationary orbit to rendezvous with other bodies in the Solar System. Our main tool is trajectory optimization, an approach that is appropriate for these spacecraft applications where we often seek solutions that are optimal (e.g., minimum-time, minimum-fuel) in the presence of mission or spacecraft constraints. Furthermore, while endoatmospheric launch or entry-decent-landing applications may be highly stochastic or uncertain, trajectory optimization techniques enjoy the relatively benign perturbations and high-fidelity dynamical models that exist for vehicles in outer space. We employ recent advances in sequential convex programming to efficiently handle the nonlinear dynamics and non-convex constraints that exist in our problem formulations. We have shown that the method scales well to multi-body systems with a large number of states and inputs. However, despite the benefits and advantages, we also come to realize that the approach is indeed a local optimization method that depends on the quality of an initial guess at the solution. We therefore conclude that it is prudent to warm start our approach with results gleaned from decades of research on similar but more fundamental problems and approaches.

The following topics are areas for future work:

- In our attitude trajectory optimization problems, we have assumed rigid body dynamics. The work may be extended to include models that consider the dynamics of propellant fuel slosh or flexible modes (i.e., due to antennas and solar panels) that affect agile attitude maneuvering.

- We have assumed the attitude control actuators (i.e., spinning rotors) to have single-integrator dynamics. We should address higher-fidelity models that may include non-linear friction and motor dynamics. We should also include other attitude control actuators such as reaction thrusters, control-moment gyroscopes, and magnetorquers.

- Our attitude problem formulations have included constraints on the angular velocity

state (i.e., slew rate), actuator state, and input torque. We may also consider non-convex constraints on the quaternion states. For example, we may impose "keep-out zones" that prohibit a spacecraft body axis from pointing in a certain direction.

- For orbit control, we have demonstrated the use of continuous low-thrust and solar radiation pressure. We may also include higher-fidelity models of the electric propulsion plant or the use of impulsive thrust via chemical rockets.

- We have considered the orbital perturbations of J_2 gravity, atmospheric drag, solar radiation pressure in the SCI frame, and continuous low-thrust. We should also include orbital perturbations due to third bodies (e.g., Moon, Sun, other planets) and solar radiation pressure in the ECI frame for high Earth orbit (HEO) applications.

- We may include attitude perturbations such as the moments due to atmospheric drag, solar radiation pressure, Earth's magnetic field, and gravity gradient.

- Since the attitude perturbations mentioned above depend on the orbital state of the spacecraft, we must address the coupling between the 3-DOF attitude motion of the spacecraft with its 3-DOF orbital motion. One approach is to sequentially solve the problems, i.e., solve an orbit trajectory optimization problem and use the resulting solution to estimate orbit-dependent attitude perturbations, representing them as forcing functions in the subsequent attitude trajectory optimization. In return, the solution of the attitude problem may then be used to better estimate attitude-dependent orbit perturbations. This process may be carried out iteratively until the solutions of both problems converge. Alternatively, we may solve a full 6-DOF problem that simultaneously addresses attitude and orbital motion.

- We recall that the low Earth orbit trajectory optimization problems resulted in high-frequency orbital motion over a short duration of several hours. On the other hand, the rendezvous problem using solar radiation pressure resulted in low-frequency orbital motion over a long duration of several years. We should also consider problems involving high-frequency orbital motion over long durations. For example, a low-thrust trajectory from low Earth orbit (LEO) to geo-synchronous orbit (GSO) using a geosynchronous transfer orbit (GTO) would require many revolutions over the course of months or years.

- Finally, future work should study problems involving multi-plane constellations, e.g., we may formulate a problem to convert a single-plane constellation into a multi-plane Walker constellation.

Appendix A

Mathematical Background

A.1 Derivatives

Jacobian

The first-order partial derivative (or *Jacobian*) matrix of a vector-valued, differentiable function $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$ evaluated at the point $\mathbf{x} \in \mathbb{R}^n$ can be denoted in the following ways:

$$D\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \frac{\partial \mathbf{f}}{\partial x_1}(\mathbf{x}) & \cdots & \frac{\partial \mathbf{f}}{\partial x_n}(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial \mathbf{x}}(\mathbf{x}) \\ \vdots \\ \frac{\partial f_m}{\partial \mathbf{x}}(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(\mathbf{x}) & \cdots & \frac{\partial f_1}{\partial x_n}(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1}(\mathbf{x}) & \cdots & \frac{\partial f_m}{\partial x_n}(\mathbf{x}) \end{bmatrix} \in \mathbb{R}^{m \times n}. \quad \text{(A.1)}$$

Gradient

Note that the Jacobian of a scalar-valued, differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ evaluated at the point $\mathbf{x} \in \mathbb{R}^n$ is a row vector:

$$Df(x) = \frac{\partial f}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial f}{\partial x_1}(\mathbf{x}) & \cdots & \frac{\partial f}{\partial x_n}(\mathbf{x}) \end{bmatrix} \in \mathbb{R}^{1 \times n}. \quad \text{(A.2)}$$

Taking its transpose results in a column vector called the *gradient* of the function f at $\mathbf{x}$:

$$\nabla f(\mathbf{x}) := Df(\mathbf{x})^\top \in \mathbb{R}^{n \times 1}. \quad \text{(A.3)}$$

Hessian

The second-order partial derivative (or *Hessian*) matrix of a scalar-valued, twice-differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ evaluated at the point $\mathbf{x} \in \mathbb{R}^n$ is a symmetric $n \times n$ matrix:

$$Hf(\mathbf{x}) = D^2 f(x) = D\nabla f(\mathbf{x})^\top = \nabla^2 f(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2 f}{(\partial x_1)^2}(\mathbf{x}) & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n}(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1}(\mathbf{x}) & \cdots & \frac{\partial^2 f}{(\partial x_n)^2}(\mathbf{x}) \end{bmatrix} \in \mathbb{R}^{n \times n}. \quad \text{(A.4)}$$

Multivariable Chain Rule

To compute the derivative of composite functions, we can use the multivariable chain rule. Let $f : \mathbb{R}^n \to \mathbb{R}$ so that $f = f(\mathbf{x}) = f(x_1, \ldots, x_n)$. Also let functions $g_1, \ldots g_n : \mathbb{R}^m \to \mathbb{R}$ so that $g_i(\mathbf{y}) = g_i(y_1, \ldots, y_m)$ for all $i = 1, \ldots, n$. We define $\mathbf{g} : \mathbb{R}^m \to \mathbb{R}^n$ by $\mathbf{g} = [g_1, \ldots, g_n]^\top$. A composite function $h : \mathbb{R}^m \to \mathbb{R}$ can be created by setting $x_i = g_i(\mathbf{y})$, so that:

$$h(\mathbf{y}) = f\left(\mathbf{g}(\mathbf{y})\right) = f\left(g_1(\mathbf{y}), \ldots, g_n(\mathbf{y})\right) \tag{A.5}$$

By the multivariable chain rule, we then have:

$$\frac{\partial h}{\partial y_k}(\mathbf{y}) = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}\left(\mathbf{g}(\mathbf{y})\right) \frac{\partial g_i}{\partial y_k}(\mathbf{y}) \quad \forall \; k = 1, \ldots, m \tag{A.6}$$

More generally, we can represent the derivative of a vector-valued composite function $\mathbf{h} : \mathbb{R}^m \to \mathbb{R}^p$ using the Jacobian matrix. Let $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^p$ and $\mathbf{g} : \mathbb{R}^m \to \mathbb{R}^n$ be differentiable functions. We define the composite function as $\mathbf{h} = \mathbf{f}(\mathbf{g}(\mathbf{y}))$ where $\mathbf{y} \in \mathbb{R}^m$. Then $\mathbf{h}$ is differentiable in $\mathbf{y}$ with the following Jacobian matrix evaluated at $\bar{\mathbf{y}}$:

$$D_{\mathbf{y}}\mathbf{h}(\bar{\mathbf{y}}) = D_{\mathbf{g}}\mathbf{f}(\mathbf{g}(\bar{\mathbf{y}}))D_{\mathbf{y}}\mathbf{g}(\bar{\mathbf{y}}) \tag{A.7}$$

A.2 Linear Approximation

The linear approximation of a vector-valued, differentiable function $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$ around a point $\bar{\mathbf{x}}$ is the first-order Taylor expansion about that point:

$$\mathbf{f}(\mathbf{x}) \approx \mathbf{f}(\bar{\mathbf{x}}) + D\mathbf{f}(\bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}}) \tag{A.8}$$

Let us consider a vector-valued, differentiable function $\mathbf{f} : \{\mathbb{R}^n, \mathbb{R}^m\} \to \mathbb{R}^n$ and linearize it about the point $\{\bar{\mathbf{x}}, \bar{\mathbf{u}}\}$:

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) \approx \mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) + D_{\mathbf{x}}\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})(\mathbf{x} - \bar{\mathbf{x}}) + D_{\mathbf{u}}\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})(\mathbf{u} - \bar{\mathbf{u}}) \tag{A.9}$$

If we define the following matrix-valued parameters $A := D_{\mathbf{x}}\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})$ and $B := D_{\mathbf{u}}\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})$, the expression above may be arranged into the following form:

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) \approx \mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) + A\mathbf{x} + B\mathbf{u} - (A\bar{\mathbf{x}} + B\bar{\mathbf{u}}) \tag{A.10}$$

Let us now define a vector-valued, differentiable function $\mathbf{F} : \{\mathbb{R}^n, \mathbb{R}^m, \mathbb{R}\} \to \mathbb{R}^n$:

$$\mathbf{F}(\mathbf{x}, \mathbf{u}, s) := s\mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{A.11}$$

and find its first-order Taylor expansion about the point $\{\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{s}\}$:

$$\begin{aligned}
\mathbf{F}(\mathbf{x}, \mathbf{u}, s) &\approx \mathbf{F}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{s}) + D_{\mathbf{x}}\mathbf{F}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{s})(\mathbf{x} - \bar{\mathbf{x}}) + D_{\mathbf{u}}\mathbf{F}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{s})(\mathbf{u} - \bar{\mathbf{u}}) + D_s\mathbf{F}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{s})(s - \bar{s}) \\
&= \bar{s}\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) + \bar{s}A\mathbf{x} + \bar{s}B\mathbf{u} - \bar{s}\left(A\bar{\mathbf{x}} + B\bar{\mathbf{u}}\right) + \mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})s - \bar{s}\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) \\
&= A_s\mathbf{x} + B_s\mathbf{u} + \mathbf{\Sigma}s + \boldsymbol{\xi}
\end{aligned} \tag{A.12}$$

where $A_s := \bar{s}A$, $B_s := \bar{s}B$, $\mathbf{\Sigma} := \mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})$, and $\boldsymbol{\xi} := -\bar{s}\left(A\bar{\mathbf{x}} + B\bar{\mathbf{u}}\right)$.

A.3   Convex Sets and Functions

In this section, we review the definitions of convex sets and functions [65].

Convex sets

A set $C \subseteq \mathbb{R}^n$ is convex if for all $a, b \in C$ and $0 \leq \theta \leq 1$, we have:

$$\theta a + (1 - \theta)b \in C \tag{A.13}$$

The geometric intuition is that if C is convex, then for all choices of $a, b \in C$, the line segment connecting a and b also lies in C.

Convex functions of one variable

A function $f : \mathbb{R} \to \mathbb{R}$ is called convex if

$$f\left(\theta x + (1 - \theta)y\right) \leq \theta f(x) + (1 - \theta)f(y) \tag{A.14}$$

for all $x, y \in \mathbb{R}$ and $0 \leq \theta \leq 1$. The geometric intuition is that if f is convex, then for all points x and y, the graph of f lies below the line segment connecting $f(x)$ and $f(y)$.

Equivalent characterizations of convexity are as follows:

- If $f : \mathbb{R} \to \mathbb{R}$ is continuously differentiable, then f is convex if and only if

$$f(x) + \frac{d}{dx}f(x)(y - x) \leq f(y) \tag{A.15}$$

  for all $x, y \in \mathbb{R}$. This condition implies that the graph of the convex function f lies above each of its tangent lines.

- If f is twice continuously differentiable, then f is convex if and only if

$$\frac{d^2}{dx^2}f(x) \geq 0 \tag{A.16}$$

  for all $x \in \mathbb{R}$.

Convex functions of multiple variables

A function $f : \mathbb{R}^n \to \mathbb{R}$ is called convex if

$$f\left(\theta \mathbf{x} + (1 - \theta)\mathbf{y}\right) \leq \theta f(\mathbf{x}) + (1 - \theta)f(\mathbf{y}) \tag{A.17}$$

for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and $0 \leq \theta \leq 1$.

Equivalent characterizations of multivariable convexity are as follows:

- If $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable, then f is convex if and only if

$$f(\mathbf{x}) + D_\mathbf{x} f(\mathbf{x})(\mathbf{y} - \mathbf{x}) \leq f(\mathbf{y}) \tag{A.18}$$

  for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$.

- If $f : \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable, then f is convex if and only if

$$D_\mathbf{x}^2 f(x) \succeq 0 \tag{A.19}$$

  for all $x \in \mathbb{R}$.

A.4 Skew-symmetric Matrices and Cross Product

A skew symmetric matrix is a square matrix whose transpose equals its negative:

$$A \text{ is skew-symmetric} \iff A^\top = -A \tag{A.20}$$

We define the skew-symmetric operator $[\,]^\times : \mathbb{R}^3 \mapsto \mathbb{R}^{3\times3}$ to conduct the following operation:

$$\boldsymbol{\omega}^\times = \begin{bmatrix} 0 & \text{-}\omega_z & \omega_y \\ \omega_z & 0 & \text{-}\omega_x \\ \text{-}\omega_y & \omega_x & 0 \end{bmatrix} \qquad \text{where} \quad \boldsymbol{\omega} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{A.21}$$

We note that the cross product of two vectors may be represented as the following matrix-vector multiplication:

$$\mathbf{a} \times \mathbf{b} = [\mathbf{a}]^\times \mathbf{b} \tag{A.22}$$

Appendix B

Attitude Problems

B.1 Derivation of Quaternion to Rotation Matrix Equation

We determine the relationship between the rotation matrix $^{\mathrm{I}}R^{\mathrm{B}}$ and $\mathbf{q}$ by evaluating the matrix multiplication of (2.67):

$$\left(\mathbf{q} \begin{bmatrix} [\mathbf{x}]^{\mathrm{B}} \\ 0 \end{bmatrix} \right) \mathbf{q}^+ = \begin{bmatrix} q_s^+ \mathrm{I} - (\mathbf{q}_v^+)^\times & \mathbf{q}_v^+ \\ -(\mathbf{q}_v^+)^\top & q_s^+ \end{bmatrix} \left(\begin{bmatrix} q_s \mathrm{I} + (\mathbf{q}_v)^\times & \mathbf{q}_v \\ -(\mathbf{q}_v)^\top & q_s \end{bmatrix} \begin{bmatrix} [\mathbf{x}]^{\mathrm{B}} \\ 0 \end{bmatrix} \right) \tag{B.1}$$

$$= \begin{bmatrix} q_s \mathrm{I} + (\mathbf{q}_v)^\times & -\mathbf{q}_v \\ (\mathbf{q}_v)^\top & q_s \end{bmatrix} \left(\begin{bmatrix} q_s \mathrm{I} + (\mathbf{q}_v)^\times & \mathbf{q}_v \\ -(\mathbf{q}_v)^\top & q_s \end{bmatrix} \begin{bmatrix} [\mathbf{x}]^{\mathrm{B}} \\ 0 \end{bmatrix} \right) \tag{B.2}$$

$$= \begin{bmatrix} (q_s \mathrm{I} + (\mathbf{q}_v)^\times)(q_s \mathrm{I} + (\mathbf{q}_v)^\times) + \mathbf{q}_v \mathbf{q}_v^\top & \mathbf{0}^{3\times1} \\ \mathbf{0}^{1\times3} & \mathbf{q}_v^\top \mathbf{q}_v + (q_s)^2 \end{bmatrix} \begin{bmatrix} [\mathbf{x}]^{\mathrm{B}} \\ 0 \end{bmatrix} \tag{B.3}$$

From above, we observe the relationship between the vector $\mathbf{x}$ coordinated in the inertial frame and coordinated in the body frame:

$$[\mathbf{x}]^{\mathrm{I}} = \left[(q_s \mathrm{I} + \mathbf{q}_v^\times)(q_s \mathrm{I} + \mathbf{q}_v^\times) + \mathbf{q}_v \mathbf{q}_v^\top \right] [\mathbf{x}]^{\mathrm{B}} \tag{B.4}$$

We define the rotation matrix as the following expression:

$$^{\mathrm{I}}R^{\mathrm{B}} := \left[(q_s \mathrm{I} + \mathbf{q}_v^\times)(q_s \mathrm{I} + \mathbf{q}_v^\times) + \mathbf{q}_v \mathbf{q}_v^\top \right] \tag{B.5}$$

$$= q_s^2 \mathrm{I} + 2q_s \mathbf{q}_v^\times + \mathbf{q}_v^\times \mathbf{q}_v^\times + \mathbf{q}_v \mathbf{q}_v^\top \tag{B.6}$$

Noting the identity of $\mathbf{q}_v^\times \mathbf{q}_v^\times = \mathbf{q}_v \mathbf{q}_v^\top - (\mathbf{q}_v^\top \mathbf{q}_v)\mathrm{I}$, we replace the last term in the line above to get the following expression:

$$= q_s^2 \mathrm{I} + 2q_s \mathbf{q}_v^\times + 2\mathbf{q}_v^\times \mathbf{q}_v^\times + (\mathbf{q}_v^\top \mathbf{q}_v)\mathrm{I} \tag{B.7}$$

Since $\|\mathbf{q}\| = \mathbf{q}_v^\top \mathbf{q}_v + q_s^2 = 1 \implies \mathbf{q}_v^\top \mathbf{q}_v = (1 - q_s^2)$, we then have

$$= q_s^2 \mathrm{I} + 2q_s \mathbf{q}_v^\times + 2\mathbf{q}_v^\times \mathbf{q}_v^\times + (1 - q_s^2)\mathrm{I} \tag{B.8}$$

The final expression relating the rotation matrix and unit quaternion parameterizations is:

$$^{\mathrm{I}}R^{\mathrm{B}} = \mathrm{I} + 2\mathbf{q}_v^\times \left(q_s \mathrm{I} + \mathbf{q}_v^\times \right) \tag{B.9}$$

B.2 Derivation of Quaternion Time Derivative

The time derivative of the quaternion is:

$$\dot{\mathbf{q}}(t) := \frac{d}{dt}\mathbf{q}(t) = \lim_{\delta t \to 0} \frac{\mathbf{q}(t + \delta t) - \mathbf{q}(t)}{\delta t} \tag{B.10}$$

Let $\mathbf{q} = \mathbf{q}(t)$ and $\mathbf{q}^1 = \mathbf{q}(t + \delta t)$, then we have:

$$\dot{\mathbf{q}} := \frac{d}{dt}\mathbf{q}(t) = \lim_{\delta t \to 0} \frac{\mathbf{q}^1 - \mathbf{q}}{\delta t} \tag{B.11}$$

Since $\mathbf{q}^1$ may be expressed as a small rotation $\delta\mathbf{q}$ from $\mathbf{q}$, such that $\mathbf{q}^1 = \mathbf{q}\delta\mathbf{q}$:

$$\dot{\mathbf{q}} = \lim_{\delta t \to 0} \frac{\mathbf{q}\delta\mathbf{q} - \mathbf{q}}{\delta t} \tag{B.12}$$

The small rotation $\delta\mathbf{q}$ may be expressed with its corresponding Euler axis $\mathbf{d}$ and angle θ:

$$\dot{\mathbf{q}} = \lim_{\delta t \to 0} \frac{\mathbf{q} \begin{bmatrix} \mathbf{d}\sin(\frac{\delta\theta}{2}) \\ \cos(\frac{\delta\theta}{2}) \end{bmatrix} - \mathbf{q}}{\delta t} \tag{B.13}$$

For small rotations, the expression can be approximated as:

$$\dot{\mathbf{q}} = \lim_{\delta t \to 0} \frac{\mathbf{q} \begin{bmatrix} \mathbf{d}\frac{\delta\theta}{2} \\ 1 \end{bmatrix} - \mathbf{q}}{\delta t} \tag{B.14}$$

If we express the first term as a sum of quaternions and carry out the quaternion algebra, we get :

$$\dot{\mathbf{q}} = \lim_{\delta t \to 0} \frac{\left(\mathbf{q} \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} + \mathbf{q} \begin{bmatrix} \mathbf{d}\frac{\delta\theta}{2} \\ 0 \end{bmatrix} \right) - \mathbf{q}}{\delta t} \tag{B.15}$$

$$= \lim_{\delta t \to 0} \frac{\left(\mathbf{q} + \mathbf{q} \begin{bmatrix} \mathbf{d}\frac{\delta\theta}{2} \\ 0 \end{bmatrix} \right) - \mathbf{q}}{\delta t} \tag{B.16}$$

$$= \lim_{\delta t \to 0} \frac{\mathbf{q} \begin{bmatrix} \mathbf{d}\frac{\delta\theta}{2} \\ 0 \end{bmatrix}}{\delta t} \tag{B.17}$$

$$= \lim_{\delta t \to 0} \mathbf{q} \begin{bmatrix} \frac{1}{2}\mathbf{d}\frac{\delta\theta}{\delta t} \\ 0 \end{bmatrix} \tag{B.18}$$

If we assume a constant angular velocity $\boldsymbol{\omega} = \mathbf{d}\frac{\delta\theta}{\delta t}$ (of the body frame with respect to the inertial frame) during the infinitesimal time period of δt, we have the following expression for the time derivative of the quaternion, also known as the quaternion kinematics equation:

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q}\begin{bmatrix}\boldsymbol{\omega}\\0\end{bmatrix} \tag{B.19}$$

B.3 ADCS Actuator Configuration

The actuator Jacobian $A_r \in \mathbb{R}^{3\times n_r}$ describes a specific rotor configuration where the columns of A_r represent the axes of rotor rotation, and n_r is the number of rotors. Each rotor has a maximum torque and maximum momentum value, as prescribed by the rotor manufacturer. Given a particular direction $\mathbf{d}$, we can determine the total ADCS torque, composed of the individual contributions from each rotor, by solving the following optimization problem:

$$\underset{\boldsymbol{\tau_r}:\|\boldsymbol{\tau_r}\|_\infty \leq \tau_\mathbf{r}^{max}}{\text{maximize}} \quad \left(A_r\boldsymbol{\tau_r}\right)^\top \mathbf{d} \tag{B.20}$$

By rearranging the terms in the objective, we get:

$$\underset{\boldsymbol{\tau_r}:\|\boldsymbol{\tau_r}\|_\infty \leq \tau_\mathbf{r}^{max}}{\text{maximize}} \quad \left(A_r^\top\mathbf{d}\right)^\top \boldsymbol{\tau_r} \tag{B.21}$$

We let $\mathbf{z} := A_r^\top\mathbf{d}$:

$$\underset{\boldsymbol{\tau_r}:\|\boldsymbol{\tau_r}\|_\infty \leq \tau_\mathbf{r}^{max}}{\text{maximize}} \quad \mathbf{z}^\top \boldsymbol{\tau_r} \tag{B.22}$$

Due to the component-wise bound on the decision variables (i.e., $-\tau_\mathbf{r}^{max} \leq \tau_{\mathbf{r}_i} \leq +\tau_\mathbf{r}^{max}$), we note that the components of the solution will take on the following form:

$$\tau_{\mathbf{r}_i}^* = \begin{cases} +\tau_\mathbf{r}^{max}, & \text{if } z_i > 0 \\ -\tau_\mathbf{r}^{max}, & \text{if } z_i < 0 \end{cases} \tag{B.23}$$

The solution may be succinctly characterized as $\tau_{\mathbf{r}_i}^* = \text{sign}(z_i)\tau_\mathbf{r}^{max}$. When $z_i = 0$, this condition corresponds to a rotor axis that is orthogonal to the given direction $\mathbf{d}$. Any rotor torque about this axis does not contribute in the given direction and we may set $\tau_{\mathbf{r}_i} = 0$ since the decision variable is free.

Given n_r rotors, the optimal value of the total rotor torque maximization problem is then

$$\mathbf{z}^\top\boldsymbol{\tau_r}^* = \sum_{i=1}^{n_r} z_i\left(\text{sign}(z_i)\tau_\mathbf{r}^{max}\right) = \tau_\mathbf{r}^{max}\sum_{i=1}^{n_r}|z_i| = \tau_\mathbf{r}^{max}\|\mathbf{z}\|_1 = \tau_\mathbf{r}^{max}\|A_r^\top\mathbf{d}\|_1 \tag{B.24}$$

Using a dense set of unit vectors $\{\mathbf{d}_1, \mathbf{d}_2, \ldots\}$ that represent rotation axes in 3-dimensional space, we can then visualize the torque envelope for a particular actuator Jacobian and

maximum rotor torque value. The surface of the torque envelope represents the maximum total torque that can be achieved when the rotors work in unison. A similar rotor momentum envelope can also be determined, taking the same shape as the torque envelope but scaled by the ratio of max rotor momentum to max rotor torque in any given direction.

In Fig. B.1, we illustrate the torque envelope for a set of three rotors, each aligned with a body frame axis (i.e., red x-axis, green y-axis, blue z-axis). We note that the maximum torque that can be applied by the rotors is in the off-principal-axis directions.
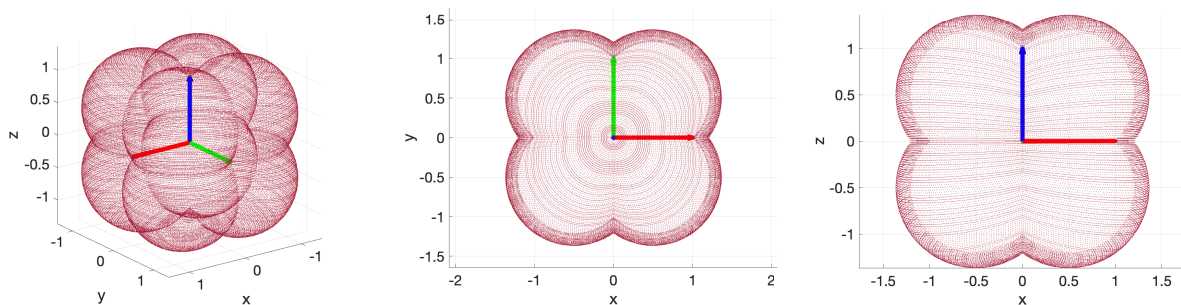
Figure B.1: Torque envelope for three-axis ADCS rotor configuration

Although three rotors aligned orthogonal to each other is sufficient for three-axis attitude control, it is common for spacecraft to have at least four rotors. In the case that one of the rotors fail, the spacecraft would still maintain three-axis control. A popular four-rotor configuration is the pyramid configuration shown in Fig. B.2. For this configuration we note that significant torque can be applied about the body x- and y-axes. Such a configuration is useful when, for example, an imaging instrument's field-of-view is aligned with the body z-axis and rotating the instrument about its line-of-sight is not critical to its operation.
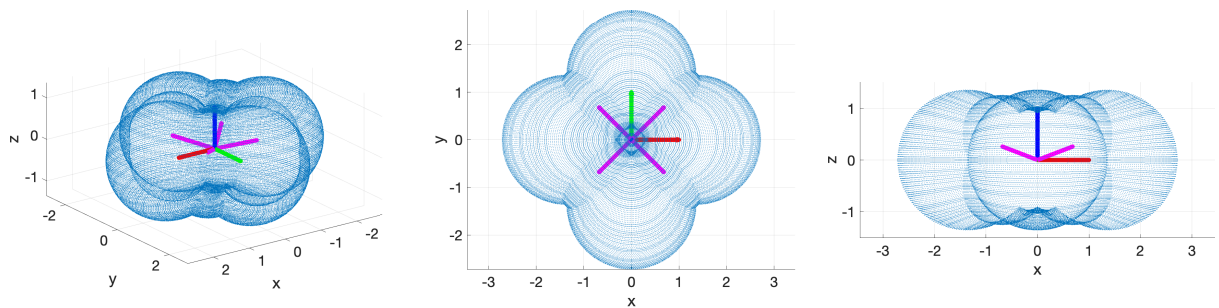
Figure B.2: Torque envelope for pyramid ADCS rotor configuration

In addition to redundancy and fault tolerance measures, a fourth rotor significantly increases the torque that can be applied in any given direction. In Fig. B.3, we superimpose the torque envelopes and note that the torque envelope of the pyramid configuration includes that of the three-axis configuration in almost entirely.
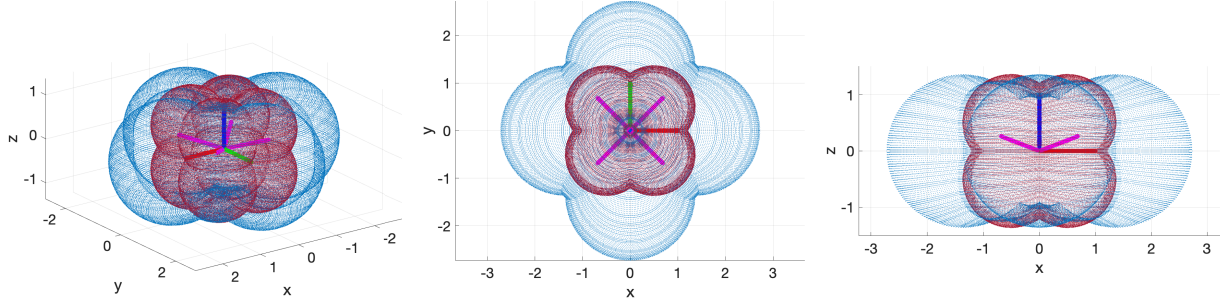
Figure B.3: Superimposed torque envelopes of three-axis and pyramid configurations

B.4 Linear Quadratic Trajectory Tracking Controller Design

We use the closed-form solution of the (unconstrained) finite-horizon, discrete-time LQR problem to design a time-varying, state feedback law that regulates deviation from desired trajectory in closed-loop simulation [43, 44, 66]. Due to the intrinsic unit-norm constraint on the unit quaternion, we convert the error-quaternion (i.e., error between desired quaternion state and the actual quaternion state) to the Euler axis and Angle representation with a transformation $\boldsymbol{\phi} = h(\bar{\mathbf{q}}^+\mathbf{q}) \in \mathbb{R}^3$ in our tracking controller implementation. Details on the transformation may be found in [6, 7, 39].

We use error variables $\boldsymbol{\phi}$, $\delta\boldsymbol{\omega} := (\boldsymbol{\omega} - \bar{\boldsymbol{\omega}})$ and $\delta\boldsymbol{\rho_r} := (\boldsymbol{\rho_r} - \bar{\boldsymbol{\rho_r}})$ to represent deviations from the desired trajectory:

$$\delta\mathbf{x}(t) := [\boldsymbol{\phi}(t)^\top \quad \delta\boldsymbol{\omega}(t)^\top \quad \delta\boldsymbol{\rho_r}(t)^\top]^\top \tag{B.25}$$

$$\delta\mathbf{u}(t) := \boldsymbol{\tau_r}(t) - \bar{\boldsymbol{\tau}}_\mathbf{r}(t) \tag{B.26}$$

The error system dynamics are:

$$\dot{\delta\mathbf{x}}(t) = \hat{A}(t)\delta\mathbf{x}(t) + \hat{B}(t)\delta\mathbf{u}(t) , \tag{B.27}$$

$$\hat{A}(t) := \begin{bmatrix} -[\bar{\boldsymbol{\omega}}(t)]^\times & \mathbf{I}^3 & \mathbf{0}^{3\times 4} \\ \mathbf{0}^{3\times 3} & -J^{-1}\left([\bar{\boldsymbol{\omega}}(t)]^\times J - [J\bar{\boldsymbol{\omega}}(t) + A_r\bar{\boldsymbol{\rho_r}}(t)]^\times\right) & -J^{-1}[\bar{\boldsymbol{\omega}}(t)]^\times A_r \\ \mathbf{0}^{4\times 3} & \mathbf{0}^{4\times 3} & \mathbf{0}^{4\times 4} \end{bmatrix} \tag{B.28}$$

$$\hat{B}(t) := \begin{bmatrix} \mathbf{0}^{3\times 4} \\ -J^{-1}A_r \\ \mathbf{I}^4 \end{bmatrix} \tag{B.29}$$

where $\mathbf{0}$ and I are null and identity matrices, respectively, and we use the skew-symmetric matrix operator $[\ ]^\times$ described in Sec A.4. In the model above, note that we have approximated the error-quaternion kinematics using the Euler axis and angle representation.

Assuming a sample time of $\Delta t = \frac{t_f}{K-1}$, we discretize the error dynamics with a zero-order hold on $\delta\mathbf{u}$. The attitude trajectory tracking problem is formulated as:

$$\underset{\{\delta\mathbf{x}_k, \delta\mathbf{u}_k\}_{k=1}^K}{\text{minimize}} \quad \delta\mathbf{x}_K^\top Q_K \delta\mathbf{x}_K + \sum_{k=1}^{K-1} \left\{ \delta\mathbf{x}_k^\top Q_k \delta\mathbf{x}_k + \delta\mathbf{u}_k^\top R \delta\mathbf{u}_k \right\} \tag{B.30}$$

$$\text{s.t.} \quad \delta\mathbf{x}_{k+1} = \hat{A}_k \delta\mathbf{x}_k + \hat{B}_k \delta\mathbf{u}_k \quad \forall\, k = 1, \ldots, K-1 \tag{B.31}$$

with weight matrices designed so that state trajectory deviations at the observation points can be penalized more heavily using a scaling term α:

$$Q_k := \begin{cases} Q, & k \notin \mathbb{K} \\ \alpha \cdot Q, & k \in \mathbb{K} \end{cases} \qquad \text{where } Q \succeq 0, \ R \succ 0, \alpha > 1. \tag{B.32}$$

The problem formulation admits a unique closed-form solution with feedback control law $\{\delta\mathbf{u}_k := -\mathbf{K}_k \delta\mathbf{x}_k\}_{k=1}^{K-1}$, where:

$$\mathbf{K}_k = (R_k + \hat{B}_k^\top P_{k+1} B_k)^{-1} \hat{B}_k^\top P_{k+1} \hat{A}_k \tag{B.33}$$

and P_k is found recursively from $P_K = Q_K$ using the following discrete-time dynamic Riccati equation:

$$P_k = \mathbf{K}_k^\top R_k \mathbf{K}_k + (\hat{A}_k - \hat{B}_k \mathbf{K}_k)^\top P_{k+1} (\hat{A}_k - \hat{B}_k \mathbf{K}_k) + Q_k. \tag{B.34}$$

B.5   ADCS Constraints and Performance Metrics

We define standard metrics to measure Attitude Determination and Control System (ADCS) performance in achieving a desired attitude trajectory. The metrics may be used to formulate performance constraints in trajectory optimization problem formulations.

Body angular speed ("slew rate")

$$\|\boldsymbol{\omega}(t)\|_2 < \omega^{max} \qquad\qquad \forall\, t \in [t_0, t_f] \tag{B.35}$$

Rotor momentum

$$\|\boldsymbol{\rho}_{\mathbf{r}}(t)\|_\infty < \rho_{\mathbf{r}}^{max} \qquad\qquad \forall\, t \in [t_0, t_f] \tag{B.36}$$

Rotor torque

$$\|\boldsymbol{\tau}_{\mathbf{r}}(t)\|_\infty < \tau_{\mathbf{r}}^{max} \qquad\qquad \forall\, t \in [t_0, t_f] \tag{B.37}$$

Instantaneous power drawn by ADCS

$$\sum_{i=1}^{4} \left| \tau_{\mathbf{r}_i}(t) \cdot \frac{1}{J_r} \rho_{\mathbf{r}_i}(t) \right| < P_{max} \qquad \qquad \forall \, t \in [t_0, t_f] \qquad \text{(B.38)}$$

Energy consumption by ADCS

$$\int_{t_0}^{t_f} \left\{ \sum_{i=1}^{4} \left| \tau_{\mathbf{r}_i}(t) \cdot \frac{1}{J_r} \rho_{\mathbf{r}_i}(t) \right| \right\} dt < E_{max} \qquad \text{(B.39)}$$

Maximum point-wise deviation between actual and desired quaternion

$$\max_{k \in \mathbb{K}} \left\| \bar{\mathbf{q}}_k^+ \mathbf{q}_k - \mathbf{q}^{\mathrm{I}} \right\|_2 \leq q_e^{max} \qquad \text{(B.40)}$$

Average deviation between actual and desired quaternion

$$\frac{1}{|\mathbb{K}|} \sum_{k \in \mathbb{K}} \left\| \bar{\mathbf{q}}_k^+ \mathbf{q}_k - \mathbf{q}^{\mathrm{I}} \right\|_2 \leq q_e^{avg} \qquad \text{(B.41)}$$

Maximum point-wise deviation between actual and desired slew rate

$$\max_{k \in \mathbb{K}} \left\| \boldsymbol{\omega}_k - \bar{\boldsymbol{\omega}}_k \right\|_2 \leq \omega_e^{max} \qquad \text{(B.42)}$$

Average deviation between actual and desired slew rate

$$\frac{1}{|\mathbb{K}|} \sum_{k \in \mathbb{K}} \left\| \boldsymbol{\omega}_k - \bar{\boldsymbol{\omega}}_k \right\|_2 \leq \omega_e^{avg} \qquad \text{(B.43)}$$

B.6 Attitude Motion of a Gyrostat

State vector

$$\mathbf{x} = [\mathbf{q}^\top \ \ \boldsymbol{\omega}^\top \ \ \boldsymbol{\rho_r}^\top]^\top = [q_1 \ \ q_2 \ \ q_3 \ \ q_4 \ \ \omega_x \ \ \omega_y \ \ \omega_z \ \ \rho_{r_1} \ \ \rho_{r_2} \ \ \rho_{r_3} \ \ \rho_{r_4}]^\top \qquad (B.44)$$

Input vector

$$\mathbf{u} = \boldsymbol{\tau_r} = \begin{bmatrix} \tau_{r_1} & \tau_{r_2} & \tau_{r_3} & \tau_{r_4} \end{bmatrix}^\top \qquad (B.45)$$

Time derivative of state vector (i.e., the dynamics)

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$$

$$= \begin{bmatrix} \mathbf{f_q}\left(\mathbf{q}(t), \boldsymbol{\omega}(t)\right) \\ \mathbf{f_\omega}\left(\boldsymbol{\omega}(t), \boldsymbol{\rho_r}(t), \boldsymbol{\tau_r}(t)\right) \\ \mathbf{f_{\rho_r}}\left(\boldsymbol{\tau_r}(t)\right) \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\mathbf{q}(t)\begin{bmatrix}\boldsymbol{\omega}(t)\\0\end{bmatrix} \\ -J^{-1}\left(\boldsymbol{\omega}(t) \times (J\boldsymbol{\omega}(t) + A_r\boldsymbol{\rho_r}(t)) + A_r\boldsymbol{\tau_r}(t)\right) \\ \boldsymbol{\tau_r}(t) \end{bmatrix} \qquad (B.46)$$

Dynamics linearized about a nominal trajectory

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) \approx \mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) + D_\mathbf{x}\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})(\mathbf{x} - \bar{\mathbf{x}}) + D_\mathbf{u}\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})(\mathbf{u} - \bar{\mathbf{u}}) \qquad (B.47)$$

The partial of $\mathbf{f}$ with respect to $\mathbf{x}$, evaluated at the point $\{\bar{\mathbf{x}}, \bar{\mathbf{u}}\}$, is:

$$D_\mathbf{x}\mathbf{f}\left(\bar{\mathbf{x}}, \bar{\mathbf{u}}\right) = \begin{bmatrix} D_\mathbf{q}\mathbf{f_q}\left(\bar{\mathbf{q}}, \bar{\boldsymbol{\omega}}\right) & D_\omega\mathbf{f_q}\left(\bar{\mathbf{q}}, \bar{\boldsymbol{\omega}}\right) & 0^{4\times4} \\ 0^{3\times4} & D_\omega\mathbf{f_\omega}\left(\bar{\boldsymbol{\omega}}, \bar{\boldsymbol{\rho_r}}, \bar{\boldsymbol{\tau_r}}\right) & D_{\rho_r}\mathbf{f_\omega}\left(\bar{\boldsymbol{\omega}}, \bar{\boldsymbol{\rho_r}}, \bar{\boldsymbol{\tau_r}}\right) \\ 0^{4\times4} & 0^{4\times3} & 0^{4\times4} \end{bmatrix} \qquad (B.48)$$

and the partial of $\mathbf{f}$ with respect to $\mathbf{u}$, evaluated at the point $\{\bar{\mathbf{x}}, \bar{\mathbf{u}}\}$, is:

$$D_\mathbf{u}\mathbf{f}\left(\bar{\mathbf{x}}, \bar{\mathbf{u}}\right) = \begin{bmatrix} 0^{4\times4} \\ D_{\tau_r}\mathbf{f_\omega}\left(\bar{\boldsymbol{\omega}}, \bar{\boldsymbol{\rho_r}}, \bar{\boldsymbol{\tau_r}}\right) \\ I^4 \end{bmatrix} \qquad (B.49)$$

The partial of the quaternion kinematics with respect to the quaternion vector is:

$$D_\mathbf{q}\mathbf{f_q}(\mathbf{q}, \boldsymbol{\omega}) = \frac{1}{2}\begin{bmatrix} (0^{3\times3} - \boldsymbol{\omega}^\times) & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^\top & 0 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix} \qquad (B.50)$$

The partial of the quaternion kinematics with respect to the angular velocity vector is:

$$D_{\boldsymbol{\omega}}\mathbf{f_q}(\mathbf{q}, \boldsymbol{\omega}) = \frac{1}{2}\begin{bmatrix} q_s I + (\mathbf{q}_v)^\times \\ -(\mathbf{q}_v)^\top \end{bmatrix} = \frac{1}{2}\begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & -q_1 \\ -q_2 & q_1 & q_4 \\ -q_1 & -q_2 & -q_3 \end{bmatrix} \tag{B.51}$$

The partial of the gyrostat dynamics with respect to the angular velocity vector is:

$$D_{\boldsymbol{\omega}}\mathbf{f}_{\boldsymbol{\omega}}\left(\boldsymbol{\omega}, \boldsymbol{\rho_r}, \boldsymbol{\tau_r}\right) = -J^{-1}\left(\boldsymbol{\omega}^\times J - [J\boldsymbol{\omega} + A_r\boldsymbol{\rho_r}]^\times\right) \tag{B.52}$$

The partial of the gyrostat dynamics with respect to the rotor momenta vector is:

$$D_{\boldsymbol{\rho_r}}\mathbf{f}_{\boldsymbol{\omega}}\left(\boldsymbol{\omega}, \boldsymbol{\rho_r}, \boldsymbol{\tau_r}\right) = -J^{-1}\boldsymbol{\omega}^\times A_r \tag{B.53}$$

The partial of the gyrostat dynamics with respect to the rotor torque vector is:

$$D_{\boldsymbol{\tau_r}}\mathbf{f}_{\boldsymbol{\omega}}\left(\boldsymbol{\omega}, \boldsymbol{\rho_r}, \boldsymbol{\tau_r}\right) = -J^{-1}A_r \tag{B.54}$$

Appendix C

Orbital Problems

C.1 Partial Derivatives of Orbital Elements, Directions, Speeds, and Accelerations

Angular momentum vector

The partial derivative of the angular momentum vector with respect to the radius and velocity vectors are:

$$D_{\mathbf{r}}\mathbf{h}(\mathbf{r}, \mathbf{v}) = \frac{\partial}{\partial \mathbf{r}} \left(-\mathbf{v} \times \mathbf{r} \right) = -\mathbf{v} \times \mathrm{I} = -\left[\mathbf{v} \right]^{\times} \tag{C.1}$$

$$D_{\mathbf{v}}\mathbf{h}(\mathbf{r}, \mathbf{v}) = \frac{\partial}{\partial \mathbf{v}} \left(\mathbf{r} \times \mathbf{v} \right) = \mathbf{r} \times \mathrm{I} = \left[\mathbf{r} \right]^{\times} \tag{C.2}$$

where, for example, $[\mathbf{r}]^{\times}$ is the skew-symmetric matrix corresponding to the vector $\mathbf{r}$.

Eccentricity vector

The partial derivative of the eccentricity vector with respect to the radius vector is:

$$D_{\mathbf{r}}\mathbf{e}(\mathbf{r}, \mathbf{v}) = D_{\mathbf{r}} \left(\frac{\mathbf{v} \times \mathbf{h}(\mathbf{r}, \mathbf{v})}{\mu} - \hat{\mathbf{r}}(\mathbf{r}) \right) \tag{C.3}$$

$$= \frac{1}{\mu} \mathbf{v} \times D_{\mathbf{r}}\mathbf{h}(\mathbf{r}, \mathbf{v}) - D_{\mathbf{r}}\hat{\mathbf{r}}(\mathbf{r}) \tag{C.4}$$

$$= \frac{1}{\mu} [\mathbf{v}]^{\times} D_{\mathbf{r}}\mathbf{h}(\mathbf{r}, \mathbf{v}) - D_{\mathbf{r}}\hat{\mathbf{r}}(\mathbf{r}) \tag{C.5}$$

and its partial with respect to the velocity vector is:

$$D_{\mathbf{v}}\mathbf{e}(\mathbf{r}, \mathbf{v}) = D_{\mathbf{v}}\left(\frac{\mathbf{v} \times \mathbf{h}(\mathbf{r}, \mathbf{v})}{\mu} - \hat{\mathbf{r}}(\mathbf{r})\right) \tag{C.6}$$

$$= \frac{1}{\mu}\left(D_{\mathbf{v}}\mathbf{v} \times \mathbf{h}(\mathbf{r}, \mathbf{v}) + \mathbf{v} \times D_{\mathbf{v}}\mathbf{h}(\mathbf{r}, \mathbf{v})\right) \tag{C.7}$$

$$= \frac{1}{\mu}\left(-\mathbf{h}(\mathbf{r}, \mathbf{v}) \times D_{\mathbf{v}}\mathbf{v} + \mathbf{v} \times D_{\mathbf{v}}\mathbf{h}(\mathbf{r}, \mathbf{v})\right) \tag{C.8}$$

$$= \frac{1}{\mu}\left(-[\mathbf{h}(\mathbf{r}, \mathbf{v})]^{\times}\mathbf{I} + [\mathbf{v}]^{\times}D_{\mathbf{v}}\mathbf{h}(\mathbf{r}, \mathbf{v})\right) \tag{C.9}$$

Node vector

The partial derivative of the node vector with respect to the radius and velocity vectors are:

$$D_{\mathbf{r}}\mathbf{n}_A(\mathbf{r}, \mathbf{v}) = [\hat{\mathbf{k}}]^{\times}D_{\mathbf{r}}\mathbf{h}(\mathbf{r}, \mathbf{v}) \tag{C.10}$$

$$D_{\mathbf{v}}\mathbf{n}_A(\mathbf{r}, \mathbf{v}) = [\hat{\mathbf{k}}]^{\times}D_{\mathbf{v}}\mathbf{h}(\mathbf{r}, \mathbf{v}) \tag{C.11}$$

Unit vectors

The partial derivative of the unit vector pointing in the direction of the radius vector, with respect to the radius vector, is:

$$D_{\mathbf{r}}\hat{\mathbf{r}}(\mathbf{r}) = \frac{\partial}{\partial\mathbf{r}}\left(\mathbf{r}\|\mathbf{r}\|^{-1}\right) = \|\mathbf{r}\|^{-1}\mathbf{I} - \|\mathbf{r}\|^{-3}\mathbf{r}\mathbf{r}^{\top} \tag{C.12}$$

The partial derivative of the unit vector pointing in the direction of the velocity vector, with respect to the velocity vector is:

$$D_{\mathbf{v}}\hat{\mathbf{v}}(\mathbf{v}) = \frac{\partial}{\partial\mathbf{v}}\left(\mathbf{v}\|\mathbf{v}\|^{-1}\right) = \|\mathbf{v}\|^{-1}\mathbf{I} - \|\mathbf{v}\|^{-3}\mathbf{v}\mathbf{v}^{\top} \tag{C.13}$$

The partial derivative of the unit vector pointing in the direction of the angular momentum vector, with respect to the angular momentum vector is:

$$D_{\mathbf{h}}\hat{\mathbf{h}}(\mathbf{h}) = \frac{\partial}{\partial\mathbf{h}}\left(\mathbf{h}\|\mathbf{h}\|^{-1}\right) = \|\mathbf{h}\|^{-1}\mathbf{I} - \|\mathbf{h}\|^{-3}\mathbf{h}\mathbf{h}^{\top} \tag{C.14}$$

The partial derivative of the unit vector pointing in the direction of the eccentricity vector, with respect to the eccentricity vector is:

$$D_{\mathbf{e}}\hat{\mathbf{e}}(\mathbf{e}) = \frac{\partial}{\partial\mathbf{e}}\left(\mathbf{e}\|\mathbf{e}\|^{-1}\right) = \|\mathbf{e}\|^{-1}\mathbf{I} - \|\mathbf{e}\|^{-3}\mathbf{e}\mathbf{e}^{\top} \tag{C.15}$$

The partial derivative of the unit vector pointing in the direction of the node vector, with respect to the node vector is:

$$D_{\mathbf{n}_A}\hat{\mathbf{n}}_A(\mathbf{n}_A) = \frac{\partial}{\partial\mathbf{n}_A}\left(\mathbf{n}_A\|\mathbf{n}_A\|^{-1}\right) = \|\mathbf{n}_A\|^{-1}\mathbf{I} - \|\mathbf{n}_A\|^{-3}\mathbf{n}_A(\mathbf{n}_A)^{\top} \tag{C.16}$$

With the expressions above, we may compute the partial derivative of the unit vector pointing in the direction of the angular momentum vector, with respect to the radius vector, using the chain rule:

$$D_{\mathbf{r}}\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v}) = D_{\mathbf{h}}\hat{\mathbf{h}}\left(\mathbf{h}(\mathbf{r}, \mathbf{v})\right) D_{\mathbf{r}}\mathbf{h}(\mathbf{r}, \mathbf{v}) \tag{C.17}$$

$$= \left(\|\mathbf{h}(\mathbf{r}, \mathbf{v})\|^{-1}\mathrm{I} - \|\mathbf{h}(\mathbf{r}, \mathbf{v})\|^{-3}\mathbf{h}(\mathbf{r}, \mathbf{v})\mathbf{h}(\mathbf{r}, \mathbf{v})^{\top}\right)\left(-[\mathbf{v}]^{\times}\right) \tag{C.18}$$

and the partial derivative with respect to the velocity vector:

$$D_{\mathbf{v}}\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v}) = D_{\mathbf{h}}\hat{\mathbf{h}}(\mathbf{h}\left(\mathbf{r}, \mathbf{v}\right)) D_{\mathbf{v}}\mathbf{h}(\mathbf{r}, \mathbf{v}) \tag{C.19}$$

$$= \left(\|\mathbf{h}(\mathbf{r}, \mathbf{v})\|^{-1}\mathrm{I} - \|\mathbf{h}(\mathbf{r}, \mathbf{v})\|^{-3}\mathbf{h}(\mathbf{r}, \mathbf{v})\mathbf{h}(\mathbf{r}, \mathbf{v})^{\top}\right)\left([\mathbf{r}]^{\times}\right) \tag{C.20}$$

To verify these partial derivatives, we can derive their expressions explicitly. We first consider the partial of the angular momentum unit vector with respect to the radius vector:

$$\frac{\partial\hat{\mathbf{h}}}{\partial\mathbf{r}} = \frac{\partial}{\partial\mathbf{r}}\left(\frac{\mathbf{r}\times\mathbf{v}}{\|\mathbf{r}\times\mathbf{v}\|}\right) \tag{C.21}$$

$$= \|\mathbf{r}\times\mathbf{v}\|^{-1}\left(-\mathbf{v}\times\mathrm{I}\right) + (\mathbf{r}\times\mathbf{v})\left(-\|\mathbf{r}\times\mathbf{v}\|^{-2}\frac{(\mathbf{r}\times\mathbf{v})^{\top}}{\|\mathbf{r}\times\mathbf{v}\|}\left(-\mathbf{v}\times\mathrm{I}\right)\right) \tag{C.22}$$

$$= -\|\mathbf{r}\times\mathbf{v}\|^{-1}[\mathbf{v}]^{\times} + \|\mathbf{r}\times\mathbf{v}\|^{-3}(\mathbf{r}\times\mathbf{v})(\mathbf{r}\times\mathbf{v})^{\top}[\mathbf{v}]^{\times} \tag{C.23}$$

$$= \left(\|\mathbf{r}\times\mathbf{v}\|^{-1}\mathrm{I} - \|\mathbf{r}\times\mathbf{v}\|^{-3}(\mathbf{r}\times\mathbf{v})(\mathbf{r}\times\mathbf{v})^{\top}\right)\left(-[\mathbf{v}]^{\times}\right) \tag{C.24}$$

$$= \left(\|\mathbf{h}\|^{-1}\mathrm{I} - \|\mathbf{h}\|^{-3}\mathbf{h}\mathbf{h}^{\top}\right)\left(-[\mathbf{v}]^{\times}\right) \tag{C.25}$$

$$= \frac{\partial\hat{\mathbf{h}}}{\partial\mathbf{h}}\frac{\partial\mathbf{h}}{\partial\mathbf{r}} \tag{C.26}$$

Similarly, the partial of the angular momentum unit vector with respect to the velocity vector may also be explicitly verified.

The partial derivative of the unit vector pointing in the orbit tangential direction, with respect to the radius vector, is:

$$D_{\mathbf{r}}\hat{\mathbf{t}}(\mathbf{r}, \mathbf{v}) = D_{\mathbf{r}}\left(\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v})\times\hat{\mathbf{r}}(\mathbf{r})\right) \tag{C.27}$$

$$= \left(D_{\mathbf{r}}\hat{\mathbf{h}}\left(\mathbf{r}, \mathbf{v}\right)\times\hat{\mathbf{r}}(\mathbf{r})\right) + \left(\hat{\mathbf{h}}\left(\mathbf{r}, \mathbf{v}\right)\times D_{\mathbf{r}}\hat{\mathbf{r}}(\mathbf{r})\right) \tag{C.28}$$

$$= \left(-\hat{\mathbf{r}}(\mathbf{r})\times D_{\mathbf{r}}\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v})\right) + \left(\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v})\times D_{\mathbf{r}}\hat{\mathbf{r}}(\mathbf{r})\right) \tag{C.29}$$

$$= \left(-[\hat{\mathbf{r}}(\mathbf{r})]^{\times} D_{\mathbf{r}}\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v})\right) + \left(\left[\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v})\right]^{\times} D_{\mathbf{r}}\hat{\mathbf{r}}(\mathbf{r})\right) \tag{C.30}$$

and the partial with respect to the velocity vector is:

$$D_\mathbf{v}\hat{\mathbf{t}}(\mathbf{r}, \mathbf{v}) = D_\mathbf{v}\left(\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v}) \times \hat{\mathbf{r}}(\mathbf{r})\right) \tag{C.31}$$

$$= \left(D_\mathbf{v}\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v}) \times \hat{\mathbf{r}}(\mathbf{r})\right) + \left(\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v}) \times D_\mathbf{v}\hat{\mathbf{r}}(\mathbf{r})\right) \tag{C.32}$$

$$= \left(-\hat{\mathbf{r}}(\mathbf{r}) \times D_\mathbf{v}\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v})\right) + \left(\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v}) \times 0^{3\times3}\right) \tag{C.33}$$

$$= -\left[\hat{\mathbf{r}}(\mathbf{r})\right]^\times D_\mathbf{v}\hat{\mathbf{h}}(\mathbf{r}, \mathbf{v}) \tag{C.34}$$

We may compute the partial derivatives of the unit vector pointing in the direction of the orbital eccentricity vector:

$$D_\mathbf{r}\hat{\mathbf{e}}(\mathbf{e}(\mathbf{r}, \mathbf{v})) = D_\mathbf{e}\hat{\mathbf{e}}(\mathbf{e}(\mathbf{r}, \mathbf{v})) D_\mathbf{r}\mathbf{e}(\mathbf{r}, \mathbf{v}) \tag{C.35}$$

$$D_\mathbf{v}\hat{\mathbf{e}}(\mathbf{e}(\mathbf{r}, \mathbf{v})) = D_\mathbf{e}\hat{\mathbf{e}}(\mathbf{e}(\mathbf{r}, \mathbf{v})) D_\mathbf{v}\mathbf{e}(\mathbf{r}, \mathbf{v}) \tag{C.36}$$

We may also find the partial derivatives of the unit vector pointing in the direction of the orbital node vector:

$$D_\mathbf{r}\hat{\mathbf{n}}_A(\mathbf{r}, \mathbf{v}) = D_{\mathbf{n}_A}\hat{\mathbf{n}}_A(\mathbf{n}_A(\mathbf{r}, \mathbf{v})) D_\mathbf{r}\mathbf{n}_A(\mathbf{r}, \mathbf{v}) \tag{C.37}$$

$$D_\mathbf{v}\hat{\mathbf{n}}_A(\mathbf{r}, \mathbf{v}) = D_{\mathbf{n}_A}\hat{\mathbf{n}}_A(\mathbf{n}_A(\mathbf{r}, \mathbf{v})) D_\mathbf{v}\mathbf{n}_A(\mathbf{r}, \mathbf{v}) \tag{C.38}$$

Orbital elements

Given the state vector of a spacecraft, semi-major axis of its orbit can be found with the following expression:

$$a = a(\mathbf{r}, \mathbf{v}) = \left(\frac{2}{\|\mathbf{r}\|} - \frac{\|\mathbf{v}\|^2}{\mu}\right)^{-1} \tag{C.39}$$

The partial derivatives of the semi-major axis with respect to the radius and velocity vectors are:

$$D_\mathbf{r}a(\mathbf{r}, \mathbf{v}) = -\left(\frac{2}{\|\mathbf{r}\|} - \frac{\|\mathbf{v}\|^2}{\mu}\right)^{-2}\left(-2\|\mathbf{r}\|^{-3}\mathbf{r}^\top\right) \tag{C.40}$$

$$D_\mathbf{v}a(\mathbf{r}, \mathbf{v}) = -\left(\frac{2}{\|\mathbf{r}\|} - \frac{\|\mathbf{v}\|^2}{\mu}\right)^{-2}\left(-\frac{2}{\mu}\mathbf{v}^\top\right) \tag{C.41}$$

The partial derivative of the orbit eccentricity $e = \|\mathbf{e}(\mathbf{r}, \mathbf{v})\|$ with respect to the eccentricity vector is:

$$D_\mathbf{e}e(\mathbf{e}(\mathbf{r}, \mathbf{v})) = \left(\frac{\partial}{\partial\mathbf{e}}\|\mathbf{e}(\mathbf{r}, \mathbf{v})\|\right) \tag{C.42}$$

$$= \frac{\mathbf{e}(\mathbf{r}, \mathbf{v})^\top}{\|\mathbf{e}(\mathbf{r}, \mathbf{v})\|} \tag{C.43}$$

and its partials with respect to the radius and velocity vectors are:

$$D_{\mathbf{r}}e\left(\mathbf{e}(\mathbf{r}, \mathbf{v})\right) = D_{\mathbf{e}}e\left(\mathbf{e}(\mathbf{r}, \mathbf{v})\right) D_{\mathbf{r}}\mathbf{e}(\mathbf{r}, \mathbf{v}) \tag{C.44}$$

$$D_{\mathbf{v}}e\left(\mathbf{e}(\mathbf{r}, \mathbf{v})\right) = D_{\mathbf{e}}e\left(\mathbf{e}(\mathbf{r}, \mathbf{v})\right) D_{\mathbf{v}}\mathbf{e}(\mathbf{r}, \mathbf{v}) \tag{C.45}$$

We recall the eccentricity of the orbit may also be computed with the semi-major axis a and semi-latus rectum p:

$$e = e\left(p(\mathbf{r}, \mathbf{v}), a(\mathbf{r}, \mathbf{v})\right) = \sqrt{1 - \frac{p(\mathbf{r}, \mathbf{v})}{a(\mathbf{r}, \mathbf{v})}} \quad \text{where} \quad p = p(\mathbf{h}) = p(\mathbf{r}, \mathbf{v}) = \frac{\|\mathbf{h}(\mathbf{r}, \mathbf{v})\|^2}{\mu} \tag{C.46}$$

The partial of the semi-latus rectum with respect to the angular momentum vector is:

$$D_{\mathbf{h}}p(\mathbf{h}) = \frac{1}{\mu}\frac{\partial}{\partial\mathbf{h}}\|\mathbf{h}\|^2 = \frac{2}{\mu}\mathbf{h}^\top \tag{C.47}$$

and its partials with respect to the radius and velocity vectors are:

$$D_{\mathbf{r}}p(\mathbf{r}, \mathbf{v}) = D_{\mathbf{h}}p(\mathbf{r}, \mathbf{v})D_{\mathbf{r}}\mathbf{h}(\mathbf{r}, \mathbf{v}) \tag{C.48}$$

$$D_{\mathbf{v}}p(\mathbf{r}, \mathbf{v}) = D_{\mathbf{h}}p(\mathbf{r}, \mathbf{v})D_{\mathbf{v}}\mathbf{h}(\mathbf{r}, \mathbf{v}) \tag{C.49}$$

The partials of the eccentricity with respect to the semi-latus rectum and semi-major axis are:

$$D_p e(p, a) = \frac{1}{2}\left(1 - \frac{p}{a}\right)^{-\frac{1}{2}}\left(-a^{-1}\right) \tag{C.50}$$

$$D_a e(p, a) = \frac{1}{2}\left(1 - \frac{p}{a}\right)^{-\frac{1}{2}}\left(pa^{-2}\right) \tag{C.51}$$

Finally, the alternative expressions for the partials of the eccentricity with respect to the radius and velocity vectors are:

$$D_{\mathbf{r}}e(\mathbf{r}, \mathbf{v}) = D_p e\left(p(\mathbf{r}, \mathbf{v}), a(\mathbf{r}, \mathbf{v})\right) D_{\mathbf{r}}p(\mathbf{r}, \mathbf{v}) + D_a e\left(p(\mathbf{r}, \mathbf{v}), a(\mathbf{r}, \mathbf{v})\right) D_{\mathbf{r}}a(\mathbf{r}, \mathbf{v}) \tag{C.52}$$

$$D_{\mathbf{v}}e(\mathbf{r}, \mathbf{v}) = D_p e\left(p(\mathbf{r}, \mathbf{v}), a(\mathbf{r}, \mathbf{v})\right) D_{\mathbf{v}}p(\mathbf{r}, \mathbf{v}) + D_a e\left(p(\mathbf{r}, \mathbf{v}), a(\mathbf{r}, \mathbf{v})\right) D_{\mathbf{v}}a(\mathbf{r}, \mathbf{v}) \tag{C.53}$$

Note that with these expressions, we do not explicitly use the eccentricity vector $\mathbf{e}$.

The orbital inclination with respect to the central's body equatorial plane is:

$$i = i(\mathbf{h}(\mathbf{r}, \mathbf{v})) = \arccos\left(\frac{h_z(\mathbf{r}, \mathbf{v})}{\|\mathbf{h}(\mathbf{r}, \mathbf{v})\|}\right) \tag{C.54}$$

The partial derivative of orbit inclination with respect to the angular momentum vector is:

$$D_{\mathbf{h}}i(\mathbf{h}) = -\frac{1}{\sqrt{1 - (h_z\|\mathbf{h}\|^{-1})^2}}\left(\|\mathbf{h}\|^{-1}\frac{\partial}{\partial\mathbf{h}}h_z + h_z\frac{\partial}{\partial\mathbf{h}}\|\mathbf{h}\|^{-1}\right) \tag{C.55}$$

where:

$$\frac{\partial}{\partial \mathbf{h}} h_z = [0 \quad 0 \quad 1]^\top \tag{C.56}$$

$$\frac{\partial}{\partial \mathbf{h}} \|\mathbf{h}\|^{-1} = -\|\mathbf{h}\|^{-3} \mathbf{h}^\top \tag{C.57}$$

The partial derivatives of orbit inclination with respect to radius and velocity vectors are:

$$D_{\mathbf{r}} i(\mathbf{r}, \mathbf{v}) = D_{\mathbf{h}} i \left(\mathbf{h}(\mathbf{r}, \mathbf{v}) \right) D_{\mathbf{r}} \mathbf{h}(\mathbf{r}, \mathbf{v}) \tag{C.58}$$

$$D_{\mathbf{v}} i(\mathbf{r}, \mathbf{v}) = D_{\mathbf{h}} i \left(\mathbf{h}(\mathbf{r}, \mathbf{v}) \right) D_{\mathbf{v}} \mathbf{h}(\mathbf{r}, \mathbf{v}) \tag{C.59}$$

Orbital speeds

The partial derivatives of the orbital radial speed, with respect to the radius and velocity vectors are:

$$D_{\mathbf{r}} v_R(\mathbf{r}, \mathbf{v}) = D_{\mathbf{r}} \left(\mathbf{v}^\top \hat{\mathbf{r}} (\mathbf{r}) \right) = \mathbf{v}^\top D_{\mathbf{r}} \hat{\mathbf{r}} (\mathbf{r}) \tag{C.60}$$

$$D_{\mathbf{v}} v_R(\mathbf{r}, \mathbf{v}) = D_{\mathbf{v}} \left(\mathbf{v}^\top \hat{\mathbf{r}} (\mathbf{r}) \right) = \hat{\mathbf{r}} (\mathbf{r})^\top D_{\mathbf{v}} \mathbf{v} = \hat{\mathbf{r}} (\mathbf{r})^\top I \tag{C.61}$$

The partial derivatives of the orbital tangential speed, with respect to the radius and velocity vectors are:

$$D_{\mathbf{r}} v_T(\mathbf{r}, \mathbf{v}) = D_{\mathbf{r}} \left(\mathbf{v}^\top \hat{\mathbf{t}} (\mathbf{r}, \mathbf{v}) \right) = \mathbf{v}^\top D_{\mathbf{r}} \hat{\mathbf{t}} (\mathbf{r}, \mathbf{v}) \tag{C.62}$$

$$D_{\mathbf{v}} v_T(\mathbf{r}, \mathbf{v}) = D_{\mathbf{v}} \left(\mathbf{v}^\top \hat{\mathbf{t}} (\mathbf{r}, \mathbf{v}) \right) = \hat{\mathbf{t}} (\mathbf{r}, \mathbf{v})^\top I + \mathbf{v}^\top D_{\mathbf{v}} \hat{\mathbf{t}} (\mathbf{r}, \mathbf{v}) \tag{C.63}$$

The partial derivatives of the orbital normal speed, with respect to the radius and velocity vectors are:

$$D_{\mathbf{r}} v_N(\mathbf{r}, \mathbf{v}) = D_{\mathbf{r}} \left(\mathbf{v}^\top \hat{\mathbf{h}} (\mathbf{r}, \mathbf{v}) \right) = \mathbf{v}^\top D_{\mathbf{r}} \hat{\mathbf{h}} (\mathbf{r}, \mathbf{v}) \tag{C.64}$$

$$D_{\mathbf{v}} v_N(\mathbf{r}, \mathbf{v}) = D_{\mathbf{v}} \left(\mathbf{v}^\top \hat{\mathbf{h}} (\mathbf{r}, \mathbf{v}) \right) = \hat{\mathbf{h}} (\mathbf{r}, \mathbf{v})^\top I + \mathbf{v}^\top D_{\mathbf{v}} \hat{\mathbf{h}} (\mathbf{r}, \mathbf{v}) \tag{C.65}$$

The partial derivative of orbit circular speed, with respect to the radius vector is:

$$D_{\mathbf{r}} v_C(\mathbf{r}) = \frac{\partial}{\partial \mathbf{r}} \left(\mu \|\mathbf{r}\|^{-1} \right)^{\frac{1}{2}} = -\frac{1}{2} \mu^{\frac{1}{2}} \|\mathbf{r}\|^{-\frac{5}{2}} \mathbf{r}^\top \tag{C.66}$$

Acceleration due to central gravity

Recall the acceleration due to central body gravity is:

$$\mathbf{a_G}(\mathbf{r}) = -\frac{\mu}{\|\mathbf{r}\|^3} \mathbf{r} \tag{C.67}$$

The partial derivative of the acceleration due to central body gravity, with respect to the radius vector is:

$$D_{\mathbf{r}} \mathbf{a_G}(\mathbf{r}) = -\frac{\mu}{\|\mathbf{r}\|^3} I + \frac{3\mu}{\|\mathbf{r}\|^5} \mathbf{r} \mathbf{r}^\top \tag{C.68}$$

Acceleration due to J2 gravity

Recall the acceleration due to J2 gravity in an Earth-Centered Inertial (ECI) frame:

$$\mathbf{a_{J2}}(\mathbf{r}) = k_{J2} \frac{1}{\|\mathbf{r}\|^5} G_{J2}(\mathbf{r})\mathbf{r} \tag{C.69}$$

where:

$$k_{J2} := 1.5 J_2 \mu r_E^2 \tag{C.70}$$

$$G_{J2}(\mathbf{r}) := \mathrm{diag}\left(\left[5\left(\frac{r_z}{\|\mathbf{r}\|}\right)^2 - 1, \quad 5\left(\frac{r_z}{\|\mathbf{r}\|}\right)^2 - 1, \quad 5\left(\frac{r_z}{\|\mathbf{r}\|}\right)^2 - 3 \right] \right) \tag{C.71}$$

Its partial derivative with respect to the radius vector is:

$$D_{\mathbf{r}}\mathbf{a_{J2}}(\mathbf{r}) = k_{J2}G_{J2}(\mathbf{r})\mathbf{r}\left(-5\frac{\mathbf{r}^\top}{\|\mathbf{r}\|^7}\right) + \frac{k_{J2}}{\|\mathbf{r}\|^5} \begin{bmatrix} r_x \frac{\partial}{\partial \mathbf{r}}\left(5\left(\frac{r_z}{\|\mathbf{r}\|}\right)^2 - 1\right) \\ r_y \frac{\partial}{\partial \mathbf{r}}\left(5\left(\frac{r_z}{\|\mathbf{r}\|}\right)^2 - 1\right) \\ r_z \frac{\partial}{\partial \mathbf{r}}\left(5\left(\frac{r_z}{\|\mathbf{r}\|}\right)^2 - 3\right) \end{bmatrix} + \frac{k_{J2}}{\|\mathbf{r}\|^5}G_{J2}(\mathbf{r})I \tag{C.72}$$

where the partial derivative of the recurring term (where c is a scalar constant), with respect to the radius vector, is:

$$\frac{\partial}{\partial \mathbf{r}}\left(5\left(\frac{r_z}{\|\mathbf{r}\|}\right)^2 - c\right) = 5r_z^2\left(-2\frac{\mathbf{r}^\top}{\|\mathbf{r}\|^4}\right) + \frac{5}{\|\mathbf{r}\|^2}\begin{bmatrix} 0 & 0 & 2r_z \end{bmatrix} \tag{C.73}$$

Acceleration due to atmospheric drag

Recall the acceleration due to atmospheric drag in an Earth-Centered Inertial (ECI) frame:

$$\mathbf{a_D}(\mathbf{r}, \mathbf{v}, m) = -\frac{1}{2}\frac{\rho}{m}C_D S\|\mathbf{v}\|\mathbf{v} \tag{C.74}$$

Its partial derivatives, with respect to the radius vector, velocity vector, and mass are:

$$D_{\mathbf{r}}\mathbf{a_D}(\mathbf{r}, \mathbf{v}, m) = -\frac{1}{2m}C_D S\|\mathbf{v}\|\mathbf{v}\left(\frac{\partial \rho}{\partial \|\mathbf{r}\|}\frac{\mathbf{r}^\top}{\|\mathbf{r}\|}\right) \tag{C.75}$$

$$D_{\mathbf{v}}\mathbf{a_D}(\mathbf{r}, \mathbf{v}, m) = -\frac{1}{2}\frac{\rho}{m}C_D S\left(\|\mathbf{v}\|I + \frac{1}{\|\mathbf{v}\|}\mathbf{v}\mathbf{v}^\top\right) \tag{C.76}$$

$$D_m\mathbf{a_D}(\mathbf{r}, \mathbf{v}, m) = +\frac{1}{2}\frac{\rho}{m^2}C_D S\|\mathbf{v}\|\mathbf{v} \tag{C.77}$$

Acceleration due to thrust

Recall the acceleration due to thrust:

$$\mathbf{a_T}(m, \mathbf{T}) = \frac{\mathbf{T}}{m} \tag{C.78}$$

The partial derivatives of this acceleration with respect to mass and the thrust vector are:

$$D_m \mathbf{a_T}(m, \mathbf{T}) = -\frac{\mathbf{T}}{m^2} \tag{C.79}$$

$$D_\mathbf{T} \mathbf{a_T}(m, \mathbf{T}) = \frac{1}{m} \mathbf{I}^3 \tag{C.80}$$

Acceleration due to solar radiation pressure in SCI frame

Recall the acceleration due to solar radiation pressure in a Sun-Centered Inertial (SCI) frame:

$$\mathbf{a_S}(\mathbf{r}, \mathbf{v}, \alpha, \beta) = a_c \left(\frac{r_{AU}}{\|\mathbf{r}\|}\right)^2 \begin{bmatrix} \hat{\mathbf{r}} & \hat{\mathbf{t}} & \hat{\mathbf{h}} \end{bmatrix} \begin{bmatrix} \cos^3 \alpha \\ \sin \alpha \cos^2 \alpha \sin \beta \\ \sin \alpha \cos^2 \alpha \cos \beta \end{bmatrix} \tag{C.81}$$

The partial of the SRP acceleration with respect to the radius vector is:

$$D_\mathbf{r} \mathbf{a_S}(\mathbf{r}, \mathbf{v}, \alpha, \beta) = \ldots$$

$$a_c \left(\frac{r_{AU}}{\|\mathbf{r}\|}\right)^2 \left((\cos^3 \alpha) D_\mathbf{r} \hat{\mathbf{r}}(\mathbf{r}) + (\sin \alpha \cos^2 \alpha \sin \beta) D_\mathbf{r} \hat{\mathbf{t}}(\mathbf{r}, \mathbf{v}) + (\sin \alpha \cos^2 \alpha \cos \beta) D_\mathbf{r} \hat{\mathbf{h}}(\mathbf{r}, \mathbf{v}) \right)$$

$$+ a_c r_{AU}^2 \begin{bmatrix} \hat{\mathbf{r}} & \hat{\mathbf{t}} & \hat{\mathbf{h}} \end{bmatrix} \begin{bmatrix} \cos^3 \alpha \\ \sin \alpha \cos^2 \alpha \sin \beta \\ \sin \alpha \cos^2 \alpha \cos \beta \end{bmatrix} \frac{\partial}{\partial \mathbf{r}} \left(\frac{1}{\|\mathbf{r}\|^2}\right) \tag{C.82}$$

where

$$\frac{\partial}{\partial \mathbf{r}} \left(\frac{1}{\|\mathbf{r}\|^2}\right) = -2 \frac{\mathbf{r}^\top}{\|\mathbf{r}\|^4} \tag{C.83}$$

The partial of the SRP acceleration with respect to the velocity vector is:

$$D_\mathbf{v} \mathbf{a_S}(\mathbf{r}, \mathbf{v}, \alpha, \beta) = \ldots$$

$$a_c \left(\frac{r_{AU}}{\|\mathbf{r}\|}\right)^2 \left((\cos^3 \alpha) D_\mathbf{v} \hat{\mathbf{r}}(\mathbf{r}) + (\sin \alpha \cos^2 \alpha \sin \beta) D_\mathbf{v} \hat{\mathbf{t}}(\mathbf{r}, \mathbf{v}) + (\sin \alpha \cos^2 \alpha \cos \beta) D_\mathbf{v} \hat{\mathbf{h}}(\mathbf{r}, \mathbf{v}) \right)$$

$$\tag{C.84}$$

The partial of the SRP acceleration with respect to the cone angle is:

$$D_\alpha \mathbf{a_S}(\mathbf{r}, \mathbf{v}, \alpha, \beta) = a_c \left(\frac{r_{AU}}{\|\mathbf{r}\|}\right)^2 \begin{bmatrix} \hat{\mathbf{r}} & \hat{\mathbf{t}} & \hat{\mathbf{h}} \end{bmatrix} \begin{bmatrix} -3 \sin \alpha \cos^2 \alpha \\ \cos \alpha \sin \beta (\cos^2 \alpha - 2 \sin^2 \alpha) \\ \cos \alpha \cos \beta (\cos^2 \alpha - 2 \sin^2 \alpha) \end{bmatrix} \tag{C.85}$$

The partial of the SRP acceleration with respect to the clock angle is:

$$D_\beta \mathbf{a_S}(\mathbf{r}, \mathbf{v}, \alpha, \beta) = a_c \left(\frac{r_{AU}}{\|\mathbf{r}\|} \right)^2 \begin{bmatrix} \hat{\mathbf{r}} & \hat{\mathbf{t}} & \hat{\mathbf{h}} \end{bmatrix} \begin{bmatrix} 0 \\ \sin\alpha \cos^2\alpha \cos\beta \\ -\sin\alpha \cos^2\alpha \sin\beta \end{bmatrix} \tag{C.86}$$

C.2   Approximation of Orbital Constraints

Bounds on radial distance

Lower and upper bounds on the radial distance ensure that the spacecraft operates within a given altitude range:

$$r_{min} \leq \|\mathbf{r}(t)\|_2 \leq r_{max} \qquad\qquad \forall \, t \in [t_0, t_f] \tag{C.87}$$

The upper bound on the radial distance is a convex constraint, but we note that the lower bound is not convex. We linearize the lower bound constraint about a trajectory $\bar{\mathbf{r}}$ to get the following approximation:

$$r_{min} \leq \|\bar{\mathbf{r}}(t)\|_2 + D_{\mathbf{r}}\|\bar{\mathbf{r}}(t)\|_2 \left(\mathbf{r}(t) - \bar{\mathbf{r}}(t)\right) \qquad\qquad \forall \, t \in [t_0, t_f]$$

$$r_{min} \leq \|\bar{\mathbf{r}}(t)\|_2 + \left(\frac{\bar{\mathbf{r}}(t)^\top}{\|\bar{\mathbf{r}}(t)\|_2} \right) \left(\mathbf{r}(t) - \bar{\mathbf{r}}(t)\right) \qquad\qquad \forall \, t \in [t_0, t_f]$$

$$r_{min} \leq \|\bar{\mathbf{r}}(t)\|_2 + \frac{\bar{\mathbf{r}}(t)^\top}{\|\bar{\mathbf{r}}(t)\|_2}\mathbf{r}(t) - \frac{\bar{\mathbf{r}}(t)^\top}{\|\bar{\mathbf{r}}(t)\|_2}\bar{\mathbf{r}}(t) \qquad\qquad \forall \, t \in [t_0, t_f]$$

$$r_{min} \leq \frac{\bar{\mathbf{r}}(t)^\top}{\|\bar{\mathbf{r}}(t)\|_2}\mathbf{r}(t) \qquad\qquad \forall \, t \in [t_0, t_f] \tag{C.88}$$

The bounds on the radial distance are now represented by the following two constraints:

$$r_{min} \leq \frac{\bar{\mathbf{r}}(t)^\top}{\|\bar{\mathbf{r}}(t)\|_2}\mathbf{r}(t) \quad , \quad \|\mathbf{r}(t)\|_2 \leq r_{max} \qquad \forall \, t \in [t_0, t_f] \tag{C.89}$$

Bounds on thrust magnitude

Lower and upper bounds on the thrust magnitude may be specified by the spacecraft's propulsion system:

$$T_{min} \leq \|\mathbf{T}(t_f)\|_2 \leq T_{max} \qquad \forall \, t \in [t_0, t_f] \tag{C.90}$$

We also linearize the lower bound constraint about a trajectory $\bar{\mathbf{T}}$ to get the following two constraints:

$$T_{min} \leq \frac{\bar{\mathbf{T}}(t)^\top}{\|\bar{\mathbf{T}}(t)\|_2}\mathbf{T}(t) \quad , \quad \|\mathbf{T}(t)\|_2 \leq T_{max} \qquad \forall \, t \in [t_0, t_f] \tag{C.91}$$

Constraint on radial distance

The following equality constraint on the final radial distance:

$$\|\mathbf{r}(t_f)\|_2 = r^{des} \tag{C.92}$$

may be approximated with the following inequality constraints:

$$r^{des} - \epsilon_r \leq \|\mathbf{r}(t_f)\|_2 \leq r^{des} + \epsilon_r \tag{C.93}$$

where ϵ_r is a small positive scalar. We further approximate the lower bound with a convex constraint by linearizing about a trajectory $\bar{\mathbf{r}}$:

$$\left(r^{des} - \epsilon_r\right) \leq \frac{\bar{\mathbf{r}}(t_f)^\top}{\|\bar{\mathbf{r}}(t_f)\|_2}\mathbf{r}(t_f) \quad , \quad \|\mathbf{r}(t_f)\|_2 \leq \left(r^{des} + \epsilon_r\right) \tag{C.94}$$

Constraint on semi-major axis

The following equality constraint on the final semi-major axis:

$$a\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) = a^{des} \tag{C.95}$$

may be approximated with the following two inequality constraints:

$$a^{des} - \epsilon_a \leq a\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) \leq a^{des} + \epsilon_a \tag{C.96}$$

where ϵ_a is a small positive scalar. We further approximate the non-convex constraints by linearizing about a trajectory $\{\bar{\mathbf{r}}, \bar{\mathbf{v}}\}$:

$$a^{des} - \epsilon_a \leq$$
$$a\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) + \begin{bmatrix} D_\mathbf{r} a\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) & D_\mathbf{v} a\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) \end{bmatrix} \left(\begin{bmatrix} \mathbf{r}(t_f) \\ \mathbf{v}(t_f) \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{r}}(t_f) \\ \bar{\mathbf{v}}(t_f) \end{bmatrix} \right) \tag{C.97}$$
$$\leq a^{des} + \epsilon_a$$

Constraint on eccentricity

The following equality constraint on the final eccentricity:

$$e\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) = e^{des} \tag{C.98}$$

may be approximated with the following two inequality constraints:

$$\max(0, e^{des} - \epsilon_e) \leq e\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) \leq e^{des} + \epsilon_e \tag{C.99}$$

where ϵ_e is a small positive scalar. We further approximate the non-convex constraints by linearizing about a trajectory $\{\bar{\mathbf{r}}, \bar{\mathbf{v}}\}$:

$$\max(0, e^{des} - \epsilon_e) \leq$$
$$e\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) + \begin{bmatrix} D_\mathbf{r} e\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) & D_\mathbf{v} e\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) \end{bmatrix} \left(\begin{bmatrix} \mathbf{r}(t_f) \\ \mathbf{v}(t_f) \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{r}}(t_f) \\ \bar{\mathbf{v}}(t_f) \end{bmatrix} \right) \tag{C.100}$$
$$\leq e^{des} + \epsilon_e$$

Constraint on radial speed

The constraint on the final radial orbital speed:

$$v_R(\mathbf{r}(t_f), \mathbf{v}(t_f)) = 0 \tag{C.101}$$

may be approximated with the following expression where ϵ_{v_R} is some small value:

$$|v_R(\mathbf{r}(t_f), \mathbf{v}(t_f))| \leq \epsilon_{v_R} \tag{C.102}$$

This constraint is also linearized about $\bar{\mathbf{r}}$ and $\bar{\mathbf{v}}$:

$$\left| v_R(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)) + \left[D_\mathbf{r} v_R\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) \quad D_\mathbf{v} v_R\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) \right] \left(\begin{bmatrix} \mathbf{r}(t_f) \\ \mathbf{v}(t_f) \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{r}}(t_f) \\ \bar{\mathbf{v}}(t_f) \end{bmatrix} \right) \right| \leq \epsilon_{v_R} \tag{C.103}$$

Constraint on normal speed

A similar constraint on the normal speed:

$$v_N(\mathbf{r}(t_f), \mathbf{v}(t_f)) = 0 \tag{C.104}$$

is approximated with:

$$|v_N(\mathbf{r}(t_f), \mathbf{v}(t_f))| \leq \epsilon_{v_N} \tag{C.105}$$

and then linearized about a trajectory:

$$\left| v_N(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)) + \left[D_\mathbf{r} v_N\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) \quad D_\mathbf{v} v_N\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) \right] \left(\begin{bmatrix} \mathbf{r}(t_f) \\ \mathbf{v}(t_f) \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{r}}(t_f) \\ \bar{\mathbf{v}}(t_f) \end{bmatrix} \right) \right| \leq \epsilon_{v_N} \tag{C.106}$$

Constraint on tangential speed

We define a constraint on the final tangential speed to equal the speed for a circular orbit at the final altitude:

$$v_T\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) = v_C\left(\mathbf{r}(t_f)\right) \tag{C.107}$$

The constraint may be approximated so that the tangential speed is within an ϵ_{v_T} interval of the desired value:

$$\left(v_C\left(\mathbf{r}(t_f)\right) - \epsilon_{v_T}\right) \leq v_T\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) \leq \left(v_C\left(\mathbf{r}(t_f)\right) + \epsilon_{v_T}\right) \tag{C.108}$$

The corresponding convex approximation is:

$$\left(v_C\left(\bar{\mathbf{r}}(t_f)\right) + D_\mathbf{r} v_C\left(\bar{\mathbf{r}}(t_f)\right)\left(\mathbf{r}(t_f) - \bar{\mathbf{r}}(t_f)\right) - \epsilon_{v_T}\right) \leq$$

$$v_T\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) + \left[D_\mathbf{r} v_T\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) \quad D_\mathbf{v} v_T\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) \right] \left(\begin{bmatrix} \mathbf{r}(t_f) \\ \mathbf{v}(t_f) \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{r}}(t_f) \\ \bar{\mathbf{v}}(t_f) \end{bmatrix} \right) \tag{C.109}$$

$$\leq \left(v_C\left(\bar{\mathbf{r}}(t_f)\right) + D_\mathbf{r} v_C\left(\bar{\mathbf{r}}(t_f)\right)\left(\mathbf{r}(t_f) - \bar{\mathbf{r}}(t_f)\right) + \epsilon_{v_T}\right)$$

Constraint on orbital specific angular momentum unit vector

We desire the specific angular momentum unit vector to be aligned with a desired direction:

$$\hat{\mathbf{h}}^{des} \cdot \hat{\mathbf{h}}\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) = 1 \tag{C.110}$$

A relaxation is to constrain the specific angular momentum vector to lie within a cone about the desired direction:

$$\hat{\mathbf{h}}^{des} \cdot \hat{\mathbf{h}}\left(\mathbf{r}(t_f), \mathbf{v}(t_f)\right) \geq \cos(\epsilon_h) \tag{C.111}$$

Since the constraint is not convex in $\{\mathbf{r}, \mathbf{v}\}$, we linearize it to get the following approximation:

$$\hat{\mathbf{h}}^{des} \cdot \left(\hat{\mathbf{h}}\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) + \left[D_{\mathbf{r}}\hat{\mathbf{h}}\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right) \quad D_{\mathbf{v}}\hat{\mathbf{h}}\left(\bar{\mathbf{r}}(t_f), \bar{\mathbf{v}}(t_f)\right)\right] \left(\begin{bmatrix} \mathbf{r}(t_f) \\ \mathbf{v}(t_f) \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{r}}(t_f) \\ \bar{\mathbf{v}}(t_f) \end{bmatrix} \right) \right)$$
$$\geq \cos(\epsilon_h) \tag{C.112}$$

Constraint on relative angular spacing

The final angular spacing between neighboring pairs of satellites can be specified by the following constraint:

$$\hat{\mathbf{r}}\left(\mathbf{r}_{i+1}(t_f)\right)^{\top} \hat{\mathbf{r}}\left(\mathbf{r}_i(t_f)\right) = \cos\left(\theta_i^{des}\right) \qquad \forall\ i = 1, \ldots, N-1 \tag{C.113}$$

For equal angular spacing, we can set the desired spacing value as:

$$\theta_i^{des} = \frac{2\pi}{N} \qquad \forall\ i = 1, \ldots, N-1 \tag{C.114}$$

We can approximate (C.113) with the following set of constraints:

$$\hat{\mathbf{r}}\left(\bar{\mathbf{r}}_{i+1}(t_f)\right)^{\top} \hat{\mathbf{r}}\left(\bar{\mathbf{r}}_i(t_f)\right) +$$
$$\left[\hat{\mathbf{r}}\left(\bar{\mathbf{r}}_{i+1}(t_f)\right)^{\top} D_{\mathbf{r}}\hat{\mathbf{r}}\left(\bar{\mathbf{r}}_i(t_f)\right) \quad \hat{\mathbf{r}}\left(\bar{\mathbf{r}}_i(t_f)\right)^{\top} D_{\mathbf{r}}\hat{\mathbf{r}}\left(\bar{\mathbf{r}}_{i+1}(t_f)\right) \right] \left(\begin{bmatrix} \mathbf{r}_i(t_f) \\ \mathbf{r}_{i+1}(t_f) \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{r}}_i(t_f) \\ \bar{\mathbf{r}}_{i+1}(t_f) \end{bmatrix} \right)$$
$$= \cos\left(\theta_i^{des}\right) + d_i \qquad \forall\ i = 1, \ldots, N-1 \tag{C.115}$$

where $\mathbf{d} = [d_1, \ldots, d_{N-1}]^{\top}$ are slack variables penalized in the objective function with a large weight $w_d > 0$:

$$\underset{\mathbf{x}, \mathbf{u}, t_f, \mathbf{d}}{\text{minimize}} \quad J_{obj}(\mathbf{x}, \mathbf{u}, t_f) + w_d \|\mathbf{d}\|_1 \tag{C.116}$$

C.3 Orbital Motion in ECI frame using Continuous Low Thrust Propulsion

State vector

$$\mathbf{x} = [\mathbf{r}^\top \ \ \mathbf{v}^\top \ \ m]^\top = [r_x \ \ r_y \ \ r_z \ \ v_x \ \ v_y \ \ v_z \ \ m]^\top \tag{C.117}$$

Input vector

$$\mathbf{u} = \mathbf{T} = [T_x \ \ T_y \ \ T_z]^\top \tag{C.118}$$

Time derivative of state vector (i.e., the dynamics)

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \tag{C.119}$$

$$= \begin{bmatrix} \mathbf{f_r}\,(\mathbf{v}(t)) \\ \mathbf{f_v}\,(\mathbf{r}(t), \mathbf{v}(t), m(t), \mathbf{T}(t)) \\ \mathbf{f}_m(\mathbf{T}(t)) \end{bmatrix} \tag{C.120}$$

$$= \begin{bmatrix} \mathbf{v}(t) \\ \mathbf{a_G}(\mathbf{r}(t)) + \mathbf{a_{J2}}(\mathbf{r}(t)) + \mathbf{a_D}(\mathbf{r}(t), \mathbf{v}(t), m(t)) + \mathbf{a_T}(m(t), \mathbf{T}(t)) \\ -\frac{\|\mathbf{T}(t)\|}{g_0 I_{sp}} \end{bmatrix} \tag{C.121}$$

Dynamics linearized about a nominal trajectory

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) \approx \mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) + D_\mathbf{x}\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})(\mathbf{x} - \bar{\mathbf{x}}) + D_\mathbf{u}\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})(\mathbf{u} - \bar{\mathbf{u}}) \tag{C.122}$$

The partial of $\mathbf{f}$ with respect to $\mathbf{x}$, evaluated at the point $\{\bar{\mathbf{x}}, \bar{\mathbf{u}}\}$, is:

$$D_\mathbf{x}\mathbf{f}\,(\bar{\mathbf{x}}, \bar{\mathbf{u}}) =$$

$$\begin{bmatrix} 0^{3\times3} & I^3 & 0^{3\times1} \\ [D_\mathbf{r}\mathbf{a_G}(\bar{\mathbf{r}}) + D_\mathbf{r}\mathbf{a_{J2}}(\bar{\mathbf{r}}) + D_\mathbf{r}\mathbf{a_D}(\bar{\mathbf{r}}, \bar{\mathbf{v}}, \bar{m})] & D_\mathbf{v}\mathbf{a_D}(\bar{\mathbf{r}}, \bar{\mathbf{v}}, \bar{m}) & [D_m\mathbf{a_D}(\bar{\mathbf{r}}, \bar{\mathbf{v}}, \bar{m}) + D_m\mathbf{a_T}(\bar{m}, \bar{\mathbf{T}})] \\ 0^{1\times3} & 0^{1\times3} & 0 \end{bmatrix} \tag{C.123}$$

and the partial of $\mathbf{f}$ with respect to $\mathbf{u}$, evaluated at the point $\{\bar{\mathbf{x}}, \bar{\mathbf{u}}\}$, is:

$$D_\mathbf{u}\mathbf{f}\,(\bar{\mathbf{x}}, \bar{\mathbf{u}}) = \begin{bmatrix} 0^{3\times3} \\ D_\mathbf{T}\mathbf{a_T}(\bar{m}, \bar{\mathbf{T}}) \\ D_\mathbf{T}\mathbf{f}_m(\bar{\mathbf{T}}) \end{bmatrix} \tag{C.124}$$

The partial of the mass dynamics with respect to thrust, evaluated at the point $\bar{\mathbf{T}}$, is:

$$D_\mathbf{T}\mathbf{f}_m(\mathbf{T}) = -\frac{\mathbf{T}^\top}{g_0 I_{sp}\|\mathbf{T}\|} \tag{C.125}$$

C.4   Derivation of Acceleration due to Solar Radiation Pressure

We derive the relationships between the solar sail's cone and clock angles with respect to the spacecraft's RTN frame. Using these angles, we may express the acceleration due to solar radiation pressure in the RTN frame. A rotation is used to coordinate the SRP acceleration vector in the Sun-centered inertial frame for use in the equations of orbital motion.
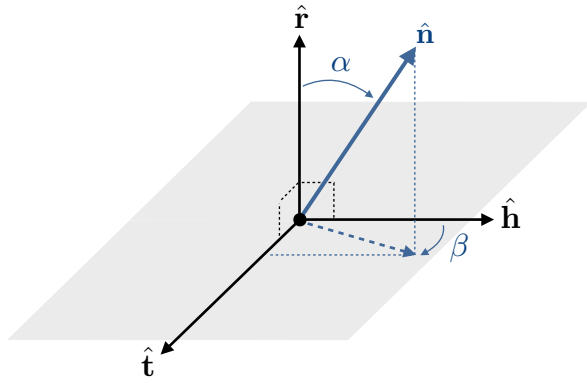

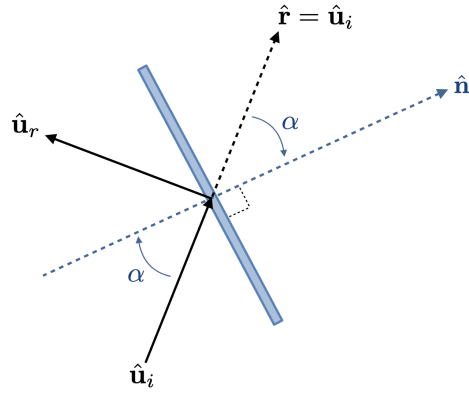Figure C.1: Cone and clock angles

Figure C.2: Cone angle w.r.t. SRP direction

Referring to Fig. C.1, the angle between the spacecraft's radial unit vector $\hat{\mathbf{r}}$ and the solar sail normal unit vector $\hat{\mathbf{n}}$ is called the cone angle α, uniquely defined as follows:

$$\hat{\mathbf{n}} \cdot \hat{\mathbf{r}} = \cos \alpha \quad , \qquad 0 \leq \alpha \leq \frac{\pi}{2} \tag{C.126}$$

We define the vector $\mathbf{p}$ as the projection of the solar sail normal unit vector $\hat{\mathbf{n}}$ onto the plane spanned by the $\hat{\mathbf{t}}$ and $\hat{\mathbf{h}}$ unit vectors, and note its length is

$$\|\mathbf{p}\| = \|\hat{\mathbf{n}}\| \sin \alpha \tag{C.127}$$

We now define the clock angle β as the angle between the orbital angular momentum unit vector $\hat{\mathbf{h}}$ and the projected vector $\mathbf{p}$. As a convention, we measure positive angular displacement towards the orbital tangential unit vector $\hat{\mathbf{t}}$ over the following range:

$$0 \leq \beta \leq 2\pi \tag{C.128}$$

We allow the clock angle to be non-unique only at the end points of the given range.

We may now express the solar sail normal vector in the spacecraft's orbital RTN frame using the cone and clock angles:

$$[\hat{\mathbf{n}}]^{\text{RTN}} = \begin{bmatrix} \|\hat{\mathbf{n}}\| \cos \alpha \\ \|\mathbf{p}\| \sin \beta \\ \|\mathbf{p}\| \cos \beta \end{bmatrix} = \begin{bmatrix} \|\hat{\mathbf{n}}\| \cos \alpha \\ \|\hat{\mathbf{n}}\| \sin \alpha \sin \beta \\ \|\hat{\mathbf{n}}\| \sin \alpha \cos \beta \end{bmatrix} = \begin{bmatrix} \cos \alpha \\ \sin \alpha \sin \beta \\ \sin \alpha \cos \beta \end{bmatrix} \tag{C.129}$$

In Fig. C.2, we illustrate the direction of incident and reflected solar radiation pressure on the solar sail surface. The pressure causes forces on the spacecraft proportional to the incident area A_i of the sail.

$$\mathbf{f_i} = PA_i\hat{\mathbf{u}}_\mathbf{i} \tag{C.130}$$
$$\mathbf{f_r} = -PA_i\hat{\mathbf{u}}_\mathbf{r} \tag{C.131}$$

The total force due to solar radiation pressure is the sum of the incident and reflected forces:

$$\mathbf{f_S} = \mathbf{f_i} + \mathbf{f_r} \tag{C.132}$$
$$= PA_i(\hat{\mathbf{u}}_\mathbf{i} - \hat{\mathbf{u}}_\mathbf{r}) \tag{C.133}$$

Using the Householder transformation:

$$= PA_i(2\hat{\mathbf{n}}\hat{\mathbf{n}}^\top\hat{\mathbf{u}}_\mathbf{i}) \tag{C.134}$$

We then express the incident area as a function of the reference area A, the solar sail normal vector, and the orbit radial unit vector:

$$= P(A\hat{\mathbf{n}} \cdot \hat{\mathbf{u}}_\mathbf{i})(2\hat{\mathbf{n}}\hat{\mathbf{n}}^\top\hat{\mathbf{u}}_\mathbf{i}) \tag{C.135}$$

Rearrange terms, we get:

$$= 2PA(\hat{\mathbf{n}} \cdot \hat{\mathbf{u}}_\mathbf{i})^2\hat{\mathbf{n}} \tag{C.136}$$

By evaluating the dot product, we have:

$$\mathbf{f_S} = 2PA(\cos\alpha)^2\hat{\mathbf{n}} \tag{C.137}$$

Now that we have expression for the force due to solar radiation pressure, we define the corresponding acceleration as:

$$\mathbf{a_S} = \frac{2PA}{m}\left(\cos^2\alpha\right)\hat{\mathbf{n}} \tag{C.138}$$

The magnitude of the solar radiation pressure is:

$$P(\mathbf{r}) = \frac{G_{SC}}{c}\left(\frac{r_{AU}}{\|\mathbf{r}\|}\right)^2 \tag{C.139}$$

where G_{SC} is the solar constant at 1AU, c is the speed of light, and r_{AU} is the length of one Astronautical unit (i.e., the distance from the Sun to the Earth). We may then express the acceleration term as:

$$\mathbf{a_S} = a_c\left(\frac{r_{AU}}{\|\mathbf{r}\|}\right)^2\left(\cos^2\alpha\right)\hat{\mathbf{n}} \tag{C.140}$$

where the characteristic acceleration is defined as:

$$a_c := \frac{2G_{SC}A}{mc} \tag{C.141}$$

Coordinated in the RTN frame, the acceleration due to solar radiation pressure is:

$$[\mathbf{a_S}]^{\text{RTN}} = a_c \left(\frac{r_{AU}}{\|\mathbf{r}\|}\right)^2 \left(\cos^2 \alpha\right) [\hat{\mathbf{n}}]^{\text{RTN}} \tag{C.142}$$

$$= a_c \left(\frac{r_{AU}}{\|\mathbf{r}\|}\right)^2 \begin{bmatrix} \cos^3 \alpha \\ \sin \alpha \cos^2 \alpha \sin \beta \\ \sin \alpha \cos^2 \alpha \cos \beta \end{bmatrix} \tag{C.143}$$

To express the SRP acceleration vector in the inertial frame, we use the following relationship:

$$\mathbf{a_S} := [\mathbf{a_S}]^{\text{Inertial}} = \begin{bmatrix} \hat{\mathbf{r}} & \hat{\mathbf{t}} & \hat{\mathbf{h}} \end{bmatrix} [\mathbf{a_S}]^{\text{RTN}} \tag{C.144}$$

We have derived the expression for the acceleration due to solar radiation pressure, coordinated in the Sun-centered inertial frame:

$$\mathbf{a_S}(\mathbf{r}, \mathbf{v}, \alpha, \beta) = a_c \left(\frac{r_{AU}}{\|\mathbf{r}\|}\right)^2 \begin{bmatrix} \hat{\mathbf{r}} & \hat{\mathbf{t}} & \hat{\mathbf{h}} \end{bmatrix} \begin{bmatrix} \cos^3 \alpha \\ \sin \alpha \cos^2 \alpha \sin \beta \\ \sin \alpha \cos^2 \alpha \cos \beta \end{bmatrix} \tag{C.145}$$

C.5 Orbital Motion in SCI frame using Solar Radiation Pressure

State vector

$$\mathbf{x} = [\mathbf{r}^\top \quad \mathbf{v}^\top]^\top = [r_x \quad r_y \quad r_z \quad v_x \quad v_y \quad v_z]^\top \tag{C.146}$$

Input vector

$$\mathbf{u} = [\alpha \quad \beta]^\top \tag{C.147}$$

Time derivative of state vector (i.e., the dynamics)

$$\begin{aligned}
\dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \\
&= \begin{bmatrix} \mathbf{f_r}\left(\mathbf{v}(t)\right) \\ \mathbf{f_v}\left(\mathbf{r}(t), \mathbf{v}(t), \alpha(t), \beta(t)\right) \end{bmatrix} = \begin{bmatrix} \mathbf{v}(t) \\ \mathbf{a_G}(\mathbf{r}(t)) + \mathbf{a_S}(\mathbf{r}(t), \mathbf{v}(t), \alpha(t), \beta(t)) \end{bmatrix}
\end{aligned} \tag{C.148}$$

Dynamics linearized about a nominal trajectory

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) \approx \mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) + D_{\mathbf{x}}\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})(\mathbf{x} - \bar{\mathbf{x}}) + D_{\mathbf{u}}\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})(\mathbf{u} - \bar{\mathbf{u}}) \tag{C.149}$$

The partial of $\mathbf{f}$ with respect to $\mathbf{x}$, evaluated at the point $\{\bar{\mathbf{x}}, \bar{\mathbf{u}}\}$, is:

$$D_{\mathbf{x}}\mathbf{f}\left(\bar{\mathbf{x}}, \bar{\mathbf{u}}\right) = \begin{bmatrix} 0^{3\times3} & I^3 \\ D_{\mathbf{r}}\mathbf{a_G}(\bar{\mathbf{r}}) + D_{\mathbf{r}}\mathbf{a_S}(\bar{\mathbf{r}}, \bar{\mathbf{v}}, \bar{\alpha}, \bar{\beta}) & D_{\mathbf{v}}\mathbf{a_S}(\bar{\mathbf{r}}, \bar{\mathbf{v}}, \bar{\alpha}, \bar{\beta}) \end{bmatrix} \tag{C.150}$$

and the partial of $\mathbf{f}$ with respect to $\mathbf{u}$, evaluated at the point $\{\bar{\mathbf{x}}, \bar{\mathbf{u}}\}$, is:

$$D_{\mathbf{u}}\mathbf{f}\left(\bar{\mathbf{x}}, \bar{\mathbf{u}}\right) = \begin{bmatrix} 0^{3\times1} & 0^{3\times1} \\ D_{\alpha}\mathbf{a_S}(\bar{\mathbf{r}}, \bar{\mathbf{v}}, \bar{\alpha}, \bar{\beta}) & D_{\beta}\mathbf{a_S}(\bar{\mathbf{r}}, \bar{\mathbf{v}}, \bar{\alpha}, \bar{\beta}) \end{bmatrix} \tag{C.151}$$

Bibliography

[1] Union of Concerned Scientists, "UCS Satellite Database," January 2021. URL `https://ucsusa.org/resources/satellite-database`.

[2] Euroconsult, "Prospects for the small satellite market research report, 5th edition," February 2020. URL `https://digital-platform.euroconsult-ec.com/product/prospects-for-the-small-satellite-market/`.

[3] I. Harris and W. Priester, "Theoretical models for the solar-cycle variation of the upper atmosphere," *J. Geophys. Res.*, vol. 67, no. 12, pp. 4585–4591, 1962.

[4] L. G. Jacchia, "Thermospheric Temperature, Density, and Composition: New Models," *SAO Special Report*, vol. 375, Mar. 1977.

[5] P. M. Mehta, A. Walker, E. Lawrence, R. Linares, D. Higdon, and J. Koller, "Modeling satellite drag coefficients with response surfaces," *Advances in Space Research*, vol. 54, no. 8, pp. 1590–1607, 2014.

[6] J. Wertz, *Spacecraft Attitude Determination and Control*. Astrophysics and Space Science Library, Netherlands: Springer, 1978.

[7] F. L. Markley and J. L. Crassidis, *Fundamentals of spacecraft attitude determination and control*. Space technology library, New York, NY: Springer, 2014.

[8] B. Wie, H. Weiss, and A. Arapostathis, "Quarternion feedback regulator for spacecraft eigenaxis rotations," *Journal of Guidance, Control, and Dynamics*, vol. 12, no. 3, pp. 375–380, 1989.

[9] B. Wie and J. Lu, "Feedback control logic for spacecraft eigenaxis rotations under slew rate and control constraints," *Journal of Guidance, Control, and Dynamics*, vol. 18, no. 6, pp. 1372–1379, 1995.

[10] J. T. Betts, "Survey of numerical methods for trajectory optimization," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 193–207, 1998.

[11] M. Kelly, "An introduction to trajectory optimization: How to do your own direct collocation," *SIAM Review*, vol. 59, no. 4, pp. 849–904, 2017.

[12] I. Ross, *A Primer on Pontryagin's Principle in Optimal Control: Second Edition.* Collegiate Publishers, 2015.

[13] J. T. Betts, *Practical Methods for Optimal Control Using Nonlinear Programming.* Philadelphia, PA: Society for Industrial and Applied Mathematics, third ed., 2020.

[14] J. Nocedal and S. J. Wright, *Numerical Optimization.* New York, NY, USA: Springer, second ed., 2006.

[15] M. Szmuk, U. Eren, and B. Acikmese, "Successive convexification for Mars 6-DOF powered descent landing guidance," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, 2017.

[16] M. Szmuk and B. Acikmese, "Successive convexification for 6-DOF Mars rocket powered landing with free-final-time," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, 2018.

[17] D. Dueri, Y. Mao, Z. Mian, J. Ding, and B. Açikmeşe, "Trajectory optimization with inter-sample obstacle avoidance via successive convexification," in *Proceedings of the IEEE 56th Annual Conference on Decision and Control (CDC)*, pp. 1150–1156, 2017.

[18] Y. Mao, M. Szmuk, X. Xu, and B. Acikmese, "Successive convexification: A superlinearly convergent algorithm for non-convex optimal control problems," *arXiv:1804.06539 [math.OC]*, 2019.

[19] M. Szmuk, T. P. Reynolds, and B. Açıkmeşe, "Successive convexification for real-time six-degree-of-freedom powered descent guidance with state-triggered constraints," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 8, pp. 1399–1413, 2020.

[20] T. Reynolds, D. Malyuta, M. Mesbahi, B. Acikmese, and J. M. Carson, "A real-time algorithm for non-convex powered descent guidance," in *Proceedings of the AIAA Scitech Forum*, 2019.

[21] I. M. Ross, Q. Gong, M. Karpenko, and R. J. Proulx, "Scaling and balancing for high-performance computation of optimal controls," *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 10, pp. 2086–2097, 2018.

[22] D. G. Hull, "Conversion of optimal control problems into parameter optimization problems," *Journal of Guidance, Control, and Dynamics*, vol. 20, no. 1, pp. 57–60, 1997.

[23] D. Malyuta, T. Reynolds, M. Szmuk, M. Mesbahi, B. Acikmese, and J. M. Carson, "Discretization performance and accuracy analysis for the rocket powered descent guidance problem," in *Proceedings of the AIAA Scitech Forum*, 2019.

[24] M. Szmuk, *Successive Convexification & High Performance Feedback Control for Agile Flight.* PhD thesis, University of Washington, 2019.

[25] S. Nag, A. S. Li, and J. H. Merrick, "Scheduling algorithms for rapid imaging using agile cubesat constellations," *Advances in Space Research*, vol. 61, no. 3, pp. 891–913, 2018.

[26] S. Nag, A. S. Li, V. Ravindra, M. S. Net, K.-M. Cheung, R. Lammers, and B. Bledsoe, "Autonomous scheduling of agile spacecraft constellations with delay tolerant networking for reactive imaging," in *Proceedings of the International Conference on Planning and Scheduling SPARK Workshop*, 2019.

[27] S. Nag, M. S. Net, A. Li, and V. Ravindra, "Designing a disruption tolerant network for reactive spacecraft constellations," in *Proceedings of ASCEND*, 2020.

[28] S. Nag, M. Moghaddam, D. Selva, J. Frank, V. Ravindra, R. Levinson, A. Azemati, A. Aguilar, A. Li, and R. Akbar, "Distributed spacecraft with heuristic intelligence to enable logistical decisions," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, 2020.

[29] V. Shah, V. Vittaldev, L. Stepan, and C. Foster, "Scheduling the world's largest earth-observing fleet of medium-resolution imaging satellites," in *Proceedings of the International Workshops on Planning and Scheduling for Space*, 2019.

[30] M. Harris, "Tech giants race to build orbital internet [news]," *IEEE Spectrum*, vol. 55, no. 6, pp. 10–11, 2018.

[31] A. S. Li and J. Mason, "Optimal utility of satellite constellation separation with differential drag," in *Proceedings of the AIAA/AAS Astrodynamics Specialist Conference*, 2014.

[32] C. Foster, J. Mason, V. Vittaldev, L. Leung, V. Beukelaers, L. Stepan, and R. Zimmerman, "Constellation phasing with differential drag on planet labs satellites," *Journal of Spacecraft and Rockets*, vol. 55, no. 2, pp. 473–483, 2018.

[33] E. Sin, M. Arcak, and A. Packard, "Small satellite constellation separation using linear programming based differential drag commands," in *Proceedings of the American Control Conference*, pp. 4951–4956, 2018.

[34] E. Sin, H. Yin, and M. Arcak, "Passivity-based distributed acquisition and station-keeping control of a satellite constellation in areostationary orbit," in *Proceedings of the ASME Dynamic Systems and Control Conference*, 2020.

[35] J. T. . Y. Wen and K. Kreutz-Delgado, "The attitude control problem," *IEEE Transactions on Automatic Control*, vol. 36, no. 10, pp. 1148–1162, 1991.

[36] K. D. Bilimoria and B. Wie, "Time-optimal three-axis reorientation of a rigid spacecraft," *Journal of Guidance, Control, and Dynamics*, vol. 16, no. 3, pp. 446–452, 1993.

[37] B. Wie, D. Bailey, and C. Heiberg, "Rapid multitarget acquisition and pointing control of agile spacecraft," *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 1, pp. 96–104, 2002.

[38] E. Sin, M. Arcak, S. Nag, V. Ravindra, A. Li, and R. Levinson, "Attitude trajectory optimization for agile satellites in autonomous remote sensing constellations," in *AIAA Scitech Forum*, 2020.

[39] P. Hughes, *Spacecraft Attitude Dynamics*. Mineola, NY, USA: Dover Publications, 1986.

[40] V. Ryswyk, "Planet announces 50cm skysat imagery, tasking dashboard and up to 12x revisit," *Planet Pulse*, June 2020.

[41] K. Shoemake, "Animating rotation with quaternion curves," in *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '85, (New York, NY, USA), p. 245–254, Association for Computing Machinery, 1985.

[42] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1." `http://cvxr.com/cvx`, Mar. 2014.

[43] M. Athans and P. L. Falb, *Optimal Control: An Introduction to the Theory and its Applications*. Lincoln Laboratory Publications, New York: McGraw-Hill Inc, 1966.

[44] A. E. Bryson Jr. and Y. C. Ho, *Applied Optimal Control: Optimization, Estimation, and Control*. New York, NY, USA: Taylor & Francis, first ed., 1975.

[45] National Aeronautics and Space Administration, "NASA Earth Observatory: Earth Observing-1 (EO-1)," November 2000. URL `https://earthobservatory.nasa.gov/features/EO1/eo1_2.php`.

[46] U. Walter, *Astronautics: The Physics of Space Flight*. Physics textbook, Wiley, 2012.

[47] B. Conway, *Spacecraft Trajectory Optimization*. Cambridge Aerospace Series, Cambridge University Press, 2010.

[48] W. McClain and D. Vallado, *Fundamentals of Astrodynamics and Applications*. Space Technology Library, Springer Netherlands, 2001.

[49] C. McInnes, *Solar Sailing: Technology, Dynamics and Mission Applications*. Astronomy and Planetary Sciences, Springer, 2004.

[50] C. Foster, H. Hallam, and J. Mason, "Orbit determination and differential-drag control of planet labs cubesat constellations," *arXiv:1509.03270 [physics.space-ph]*, 2015.

[51] Space Exploration Technologies Corp., "SpaceX Starlink Deployment," May 2021. URL `https://www.spacex.com/launches/`.

[52] T. Lafleur and N. Apffel, "Low-earth-orbit constellation phasing using miniaturized low-thrust propulsion systems," *Journal of Spacecraft and Rockets Article in Advance*, 2021.

[53] M. Arcak, "Passivity as a design tool for group coordination," *IEEE Transactions on Automatic Control*, vol. 52, no. 8, pp. 1380–1390, 2007.

[54] H. Bai, M. Arcak, and J. Wen, *Cooperative control design. A systematic, passivity-based approach.* New York, NY, USA: Springer Communications and Control Engineering Series, 2011.

[55] R. Reyes-Báez, A. van der Schaft, and B. Jayawardhana, "Passivity based distributed tracking control of networked euler-lagrange systems," in *IFAC-PapersOnLine*, pp. 136–141, 2018. 7th IFAC Workshop on Distributed Estimation and Control in Networked Systems, 2018.

[56] M. Arcak, C. Meissen, and A. Packard, *Networks of Dissipative Systems: Compositional Certification of Stability, Performance, and Safety.* Switzerland: Springer Publishing Company, Incorporated, 1st ed., 2016.

[57] G. H. Hines, M. Arcak, and A. K. Packard, "Equilibrium-independent passivity: A new definition and numerical certification," *Automatica*, vol. 47, no. 9, pp. 1949–1956, 2011.

[58] M. Bürger, D. Zelazo, and F. Allgöwer, "Duality and network theory in passivity-based cooperative control," *Automatica*, vol. 50, no. 8, pp. 2051–2061, 2014.

[59] R. R. Bate, D. D. Mueller, and J. E. White, *Fundamentals of Astrodynamics.* New York: Dover Publications, 1971.

[60] R. Radhakrishnan, W. W. Edmonson, F. Afghah, R. M. Rodriguez-Osorio, F. Pinto, and S. C. Burleigh, "Survey of inter-satellite communication for small satellite systems: Physical layer to network layer view," *IEEE Communications Surveys Tutorials*, vol. 18, no. 4, pp. 2442–2473, 2016.

[61] National Aeronautics and Space Administration, "Communications- state of the art of small spacecraft technology," November 2020. URL `https://www.nasa.gov/smallsat-institute/sst-soa`.

[62] H. K. Khalil, *Nonlinear Systems.* Upper Saddle River, NJ, USA: Prentice Hall, 2002.

[63] National Aeronautics and Space Administration, "NASA"s evolutionary xenon thruster (NEXT)," October 2020. URL `https://www1.grc.nasa.gov/space/sep/gridded-ion-thrusters-next-c/`.

[64] W. von Braun, "Manned Mars Landing Presentation to the Space Task Group," October 1969. URL `https://www.nasa.gov/sites/default/files/atoms/files/19690804_manned_mars_landing_presentation_to_the_space_task_group_by_dr._wernher_von_braun.pdf`.

[65] S. Boyd and L. Vandenberghe, *Convex Optimization.* USA: Cambridge University Press, 2004.

[66] B. Anderson and J. Moore, *Optimal Control: Linear Quadratic Methods.* Englewood Cliffs, NJ, USA: Prentice-Hall, 1990.