NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

BOOLEAN XOR ENDPOINT CONSTRAINTS IN CONTINUOUS-TIME OPTIMAL CONTROL PROBLEMS

by

Elliott L. VonWeller

September 2020

Thesis Advisor: Mark Karpenko
Second Reader: Brian M. Wade

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2020	**3. REPORT TYPE AND DATES COVERED** Master's thesis

4. TITLE AND SUBTITLE BOOLEAN XOR ENDPOINT CONSTRAINTS IN CONTINUOUS-TIME OPTIMAL CONTROL PROBLEMS	**5. FUNDING NUMBERS**
6. AUTHOR(S) Elliott L. VonWeller	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.	**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

In continuous-time optimal control problems, constraints must be satisfied as a set of logical conjunctions. In many practical space missions, however, the end-point functions may contain disjunctions. This thesis presents an approach for handling these end-point function disjunctions as part of a single continuous-time trajectory optimization problem. The approach embeds continuous representations of discrete logic operators as part of the problem formulation in order to model the disjunctions. The application of this new concept is first analyzed and illustrated for a canonical double integrator model as a proxy for practical space flight applications. It is then shown how the approach can be used to efficiently allow an algorithm to choose the minimum effort or minimum time attitude rotation for a rigid spacecraft amongst a set of possible terminal attitudes.

14. SUBJECT TERMS costate, covector, DIDO, Hamiltonian, Lagrangian, optimal control	**15. NUMBER OF PAGES** 83
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89) Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

BOOLEAN XOR ENDPOINT CONSTRAINTS IN CONTINUOUS-TIME OPTIMAL CONTROL PROBLEMS

Elliott L. VonWeller
Lieutenant Commander, United States Navy
BS, Electrical Engineering, Portland State University, 2006
BS, Physics, Portland State University, 2006
MEng, Portland State University, 2007
MBA, Regis University, 2010
MS, Regis University, 2015

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ASTRONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 2020

Approved by: Mark Karpenko
Advisor

Brian M. Wade
Second Reader

Garth V. Hobson
Chair, Department of Mechanical and Aerospace Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

In continuous-time optimal control problems, constraints must be satisfied as a set of logical conjunctions. In many practical space missions, however, the end-point functions may contain disjunctions. This thesis presents an approach for handling these end-point function disjunctions as part of a single continuous-time trajectory optimization problem. The approach embeds continuous representations of discrete logic operators as part of the problem formulation in order to model the disjunctions. The application of this new concept is first analyzed and illustrated for a canonical double integrator model as a proxy for practical space flight applications. It is then shown how the approach can be used to efficiently allow an algorithm to choose the minimum effort or minimum time attitude rotation for a rigid spacecraft amongst a set of possible terminal attitudes.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

AND, OR	Boolean Operation
$e(x(t_f))$	End-point Function
$E(x(t_f))$	End-point Cost
$\bar{E}$	Endpoint Lagrangian
$f(\mathbf{x}, \mathbf{u})$	State Dynamics Vector
$F(\mathbf{x}, \mathbf{u})$	Running Cost
h, $\dot{h}$	Angular Momentum, Derivative
H	Hamiltonian
$\bar{H}$	Lagrangian of the Hamiltonian
I	Inertia Matrix
J	Cost Functional
q	Quaternion
u	Control
$\mathbf{x}$	State Vector
Z	Transformation Matrix
$\boldsymbol{\lambda}$	Costate Vector
$\boldsymbol{\mu}$	Covector
ω, $\dot{\omega}$	Body Angular Rate, Acceleration
Ω, $\dot{\Omega}$	Wheel Angular Rate, Acceleration

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgments

This has been a truly interesting, challenging and ultimately rewarding research endeavor. It would not have been possible without the guidance and insight I received from Dr. Mark Karpenko, Dr. Brian Wade and Dr. Mike Ross. I would like to express my sincere gratitude to Dr. Karpenko for his help in finding a topic to research that not only met the academic requirements, but held the promise of helping to redefine a large set of real-world problems. I would like to thank my cohort for their steadfast dedication to this program and their willingness to share their time and knowledge throughout. I would be remiss not to also acknowledge the support that I continue to receive from my family and friends, especially my daughters, Shilo and Hailey. My true source of pride and inspiration.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction

1.1 Motivation

Enabling spacecraft systems to more effectively plan and solve for minimum cost trajectory solutions can reduce the time and fuel required to maneuver between stations, increasing the number of potential collection events and fuel lifetime. Significant research has been done to develop autonomous trajectory and path planning solutions using Pontryagin's Principle [1]–[4]. In these solutions path constraints and end-point conditions must be satisfied together as a conjunction. For example, it is common practice to define a desired end-point location in *x, y and z* coordinates *AND* a final velocity in *x, y and z*, *AND* path constraints.

The purpose of this thesis work is ultimately to increase the mission value of on-orbit spacecraft. Many spacecraft missions are significantly limited by the ability to efficiently maneuver. Current solutions, to increase this efficiency, include a combination of simulation and human-in-the-loop commanding from the ground, often requiring multiple commands in sequence to achieve the desired end state. While significant work and research has been done to optimize and automate this process, expanding the current tools to include logical "OR" constraint conditions into the control and trajectory planning can significantly increase both spacecraft longevity and utility. A review of definitive texts on the subject including [1]–[4], and recent relevant work including [5]–[8] seem to validate that there is a gap in problem definition and solution development methodologies for problems with end-point disjunctions.

1.2 Thesis Objective and Scope

This thesis presents one methodology for implementing the Boolean exclusive or (*XOR*) operator into the end-point constraints for spacecraft slewing and pointing problems. This research first uses the two-dimensional double integrator model of a point mass as a simplified proxy for a spacecraft. A continuous representation of the XOR end-point condition

is then defined. Once this new end-point definition is inserted into the two-dimensional optimal control problem (OCP), Pontryagin's Minimization Principle is used to identify the necessary conditions for solution optimality, as well as any boundary condition needed to define an implementable boundary value problem (BVP). The problem is then solved and shown to meet the necessary conditions.

This new concept for a disjunctive end-point constraint is then implemented using the quaternion representation of a rigid body rotation. This is done by first defining two mathematical models (high and low fidelity) for spacecraft rotational dynamics and illustrating that these models well represent the actual systems they are modeling. Figure 1.1 shows a generic spacecraft in three-dimensional space, initially oriented so that the spacecraft body frame (red, green, blue) is aligned with the inertial frame (black).

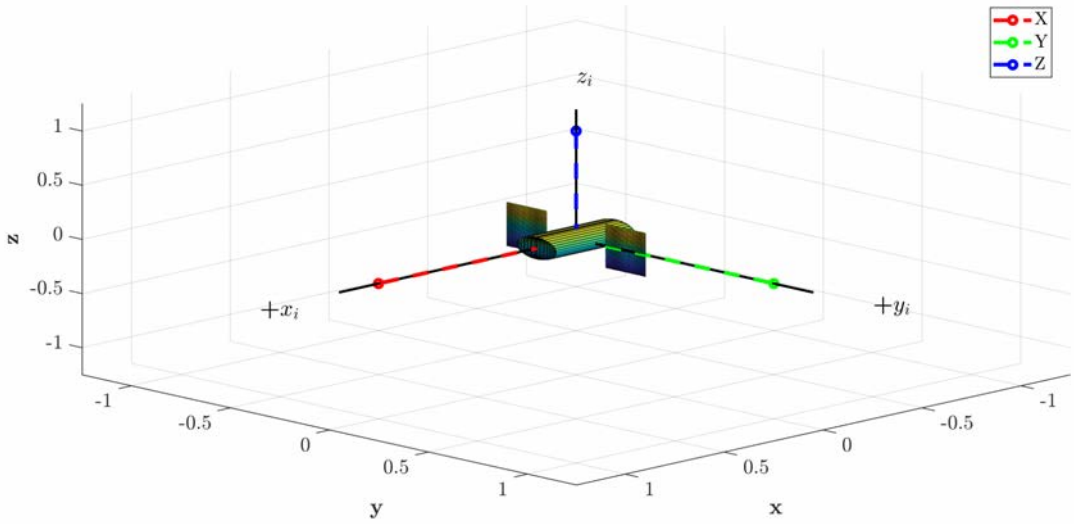


Figure 1.1: Initial spacecraft orientation in 3D space

The XOR end-point constraint is then implemented to autonomously select between two different rotations, represented by two final quaternions, shown in Figure 1.2. Pontryagin's Minimization Principle is again used to develop the necessary conditions, and the problem is solved and the solution shown to be optimal.

Current solutions would require solving for each end-point, effectively enumerating through the disjunction, and then selecting the minimum cost option. This thesis work presents a novel and unique solution to this problem.

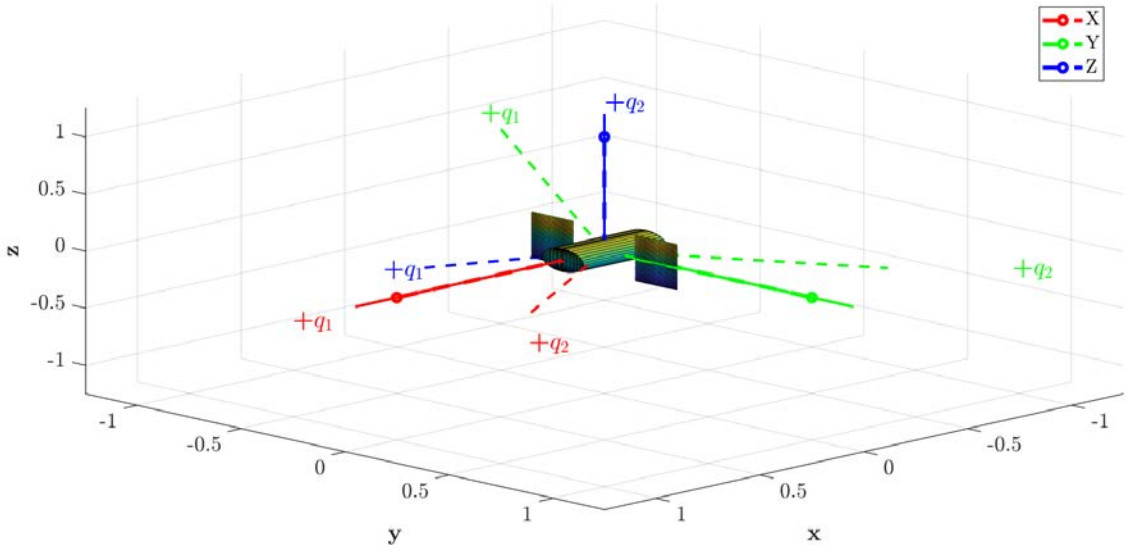


Figure 1.2: Spacecraft XOR reorientation problem

1.3 Thesis Outline

This thesis introduces the Lunar Reconnaissance Orbiter (LRO) mission as an example of a real world mission that can benefit from the introduction of disjuctive operators in optimal control. Chapter 3 introduces optimal control solution development using Pontryagin's Minimization Principle. Chapter 4 details the development of the new method for implementing the XOR end-point condition. Chapter 5 then solves the OCP using the two-dimensional double integrator problem to demonstrate that this new technique is valid. This new methodology is then implemented for a spacecraft rotation using a quaternion representation. Chapter 6 constructs the dynamic model for this analysis, and then develops and solves the OCP using the more complex quaternion rotation model, demonstrating that this new technique can be expanded out to more complex space systems problems. Finally, Chapter 7 outlines the thesis conclusions and presents areas for future work.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 2:
The Lunar Reconnaissance Orbiter Mission

2.1 LRO Mission

A timely example of how this research could impact real-world problems is the LRO mission. With NASA's goal of returning humans to the moon by 2024, scientists need to learn as much as possible about the lunar surface, which makes the mission of the LRO vital. Improving the LRO's collection efficiency would directly increase the amount of science information gathered during the resource constrained mission. The LRO is a robotic spacecraft that first entered lunar orbit on June 23, 2009, after making its way to space via an Alliance Atlas V rocket [9]. This started its Exploration Mission, focusing on furthering human existence in the solar system by mapping the moon.

NASA has extended the LRO's mission as it proved beyond capable of acquiring crucial data that addresses fundamental questions about lunar history and environment. Research questions included:

> Chronology/Bombardment, determining the bombardment history of the Moon from basin-forming events down to small, recent impacts; Crustal Evolution, investigating geological processes and their role in the evolution of the lunar crust and shallow lithosphere; Regolith Evolution, looking for evidence of processes that shaped the global lunar regolith in order to identify physical characteristics of the upper-most regolith layers; and Polar Volatiles, investigating volatile sources, sinks and transfer mechanisms with emphasis on the lunar polar regions. [10]

Using the above areas of research, scientists have used the LRO to draw important conclusions that will aid in future human habitation on the moon. In order to survive, colonists will need to have resources such as water readily available to them as cargo shuttles from Earth would be unable to keep up with the demand for necessary resources to sustain human life. Water maps and terrain maps created by the LRO will allow future settlers to pinpoint

hotspots for such resources and determine which regions would be of the greatest utility.

Notable findings thus far include the coldest temperatures measured in the entire solar system in the polar shadowed regions of the moon, mammoth hydrogen deposits in regions where temperature supports the existence of water ice, a detailed index of surface craters as well as relatively new small impact areas, relatively new volcanic ranges, and unexpectedly large amounts of volatile gasses released near the lunar surface [9]. With the data collected via images taken by the LRO, NASA and supporting organizations have developed a 3-D map of the moon's surface at 100-meter resolution with 98.2 percent coverage, excluding the polar shadow regions [11]. Shown in Figure 2.1 is an orthographic map of the Unified Geologic Map of the Moon that shows an updated topography of the lunar surface, developed by the USGS, NASA, and the Lunar Planetary Institute intended to be a reference for future moon missions [12].

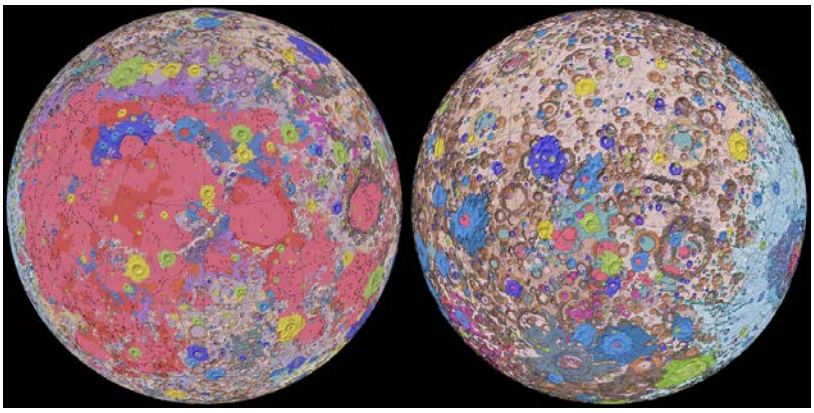


Figure 2.1: Geologic map of the moon. Source: [12]

2.2 Instruments

Such an in-depth mission requires highly-refined technology. The LRO has six highly sophisticated instruments in its charge, as well as one technology demonstration, to include the Cosmic Ray Telescope for the Effects of Radiation (CRaTER), Diviner Lunar Radiometer Experiment (DLRE), Lyman Alpha Mapping Project (LAMP), Lunar Exploration Neutron Detector (LEND), Lunar Orbiter Laser Altimeter (LOLA), Lunar Reconnaissance Orbiter Camera (LROC), and Mini-RF Technology Demonstration [13].

CRaTER assesses the radiation environment on the moon to determine potential hazards

to human life and aid in developing technology to provide protection against such hazards. CRaTER is constructed of tissue-like plastic and measures the galactic and solar cosmic ray radiation to observe and analyze the effects that such radiation would have on human biology [14] .

DLRE is the first instrument to measure the thermal emission from the moon's surface. The instrument has measured the emissions during daily periods of shadow and illumination as well as during seasonal changes in shadow and illumination, which has revealed the moon's extreme changes in temperature. Thermal imaging determines detailed temperature profiles on the moon, identifying cold traps and ice deposits that may endanger, or potentially benefit, human life. In addition to temperature hazards, DLRE also analyzes temperature differences to characterize topography, helping to identify potentially hazardous landing sites [15] .

LAMP uses ultraviolet light to locate pockets of water ice that likely lie in the polar shadow regions. It uses UV light produced by stars as well as hydrogen atoms that exist sparingly in the solar system. In addition to finding these pockets of water ice, LAMP is also on a quest to pioneer Lyman-Alpha Vision Assistance (LAVA). LAVA uses natural starlight and ultraviolet skyglow to see in the dark, ultimately allowing astronauts to work in the darkness of lunar night and in the lunar poles [16].

Using thermal, epithermal and high-energy neutron sensors, LEND searches for hydrogen distributions one to two meters beneath the lunar surface in order to identify potential regions of water ice on the moon's surface, particularly in the polar regions. The high resolution capability of LEND allows the detection of even the smallest distributions of hydrogen beneath the surface [17].

LOLA uses laser propagation and the subsequent range, scatter and return of the directed energy to determine the topographical profile of the moon. By identifying these elevation changes, scientists can make informed decisions regarding potential landing and exploration sites [18].

LROC consists of two narrow-angle cameras and one wide-angle camera that work together to "capture high resolution black and white images and moderate resolution multi-spectral images of the lunar surface" [10]. These cameras analyze data garnered via sunlight to

provide information on the moon's polar illumination conditions, thus helping to choose optimal landing sites.

Finally, the Mini-RF Technology Demonstration consists of two radar instruments launched into space, the second of which resides on the LRO. The instrument is a radar suite used to locate water ice on the lunar surface.

2.3 Mission Products

Though many of these instruments appear to accomplish similar goals, the methods of data collection they use produce significantly different results and information. Figure 2.2 provides images of the same lunar region, the South Lunar Pole, with very dissimilar results. Collectively, these instruments transmit 155 GB of data to Earth each day, all crucial to the future exploration and habitation of the moon.

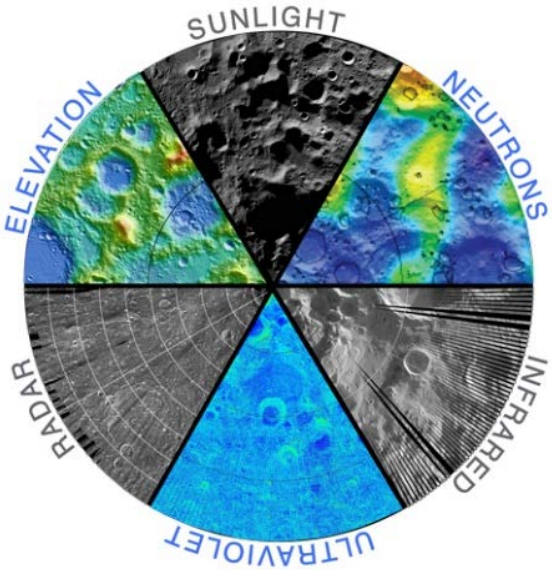


Figure 2.2: LRO sensor modes. Source: [13]

In addition to its primary mission mapping the moon, the LRO has been used to study the aftereffects of significant historical events, particularly lunar landings. Images of former Apollo landing sites have helped document the positions of many of the objects left on the moon from these missions, allowing scientists to confirm many assumptions made following the landings. The LRO first visited the site of the Apollo 11 landing, where humankind took

its first steps. The LRO captured images from just fifteen miles above the moon's surface, and the images clearly depict objects from the mission, such as the lunar module and experimental instruments that the astronauts assembled, shown in Figure 2.3. Due to its high resolution capability, the LRO images even captured the tracks created on the lunar surface as the astronauts completed the first space walk [19].

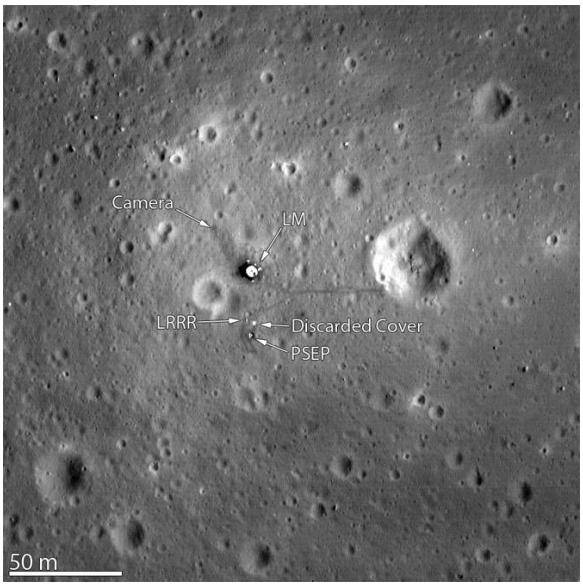


Figure 2.3: LRO image of Apollo 11 landing site. Source: [19]

Following the visitation of the Apollo 11 site, the LRO re-visited the sites of Apollo 12, 14, 15, and 16. During each of these visits, images from the LRO conveyed and confirmed details about the landing sites, including evidence of the first lunar rover's exploration of the Mons Hadley massif near the Apollo 15 landing site [20] and the previously elusive Apollo 16 booster rocket impact site [21]. Not only do these revisits help showcase prior lunar activity, but they also allow scientists to compare data discovered during the Apollo Missions to data shown in current images. These comparisons help scientists make further hypotheses regarding the lunar environment.

The Lunar Reconnaissance Orbiter plays a crucial role in future space exploration. Any errors in the LRO's operation can result in, at best case, loss of important data or, at worst case, incorrect data that could one day risk lives. There are many natural phenomena that threaten the operability and capability of the LRO, so it is key that we maximize its utility while not providing additional threats from technologic error.

From the instruments on the LRO, scientists have produced the most sophisticated and accurate model ever of the lunar surface. The LRO needs to continue producing reliable images to support the endeavor of space exploration and needs to do so as efficiently as possible, to gather as much scientific data as possible in its finite lifetime. The satellite now uses a gyro-less control mode that prevents some collection targets from being accessed, if the star trackers are occulted [22]. Further development of controllers using algorithms with the non-conjunctive functionality presented in this work could automatically determine which of the possible collections are accessible without the need for enumerating through and trying them all. This would significantly increase the rate at which collection events were planned and increase the total number of science mission achieved. The LRO mission is an excellent example of the need for developing procedures for autonomously handling disjunctive constraint sets. It is not the only space application that can benefit from a solution to this problem. Orbit transfers, landers, and mission planning are other examples where the ability to handle disjunctive constraints would be of benefit.

CHAPTER 3:
Pontryagin's Minimization Principle in Optimal Control

Optimal control problems are present in nearly every technical field. In these often complex problems, the goal is to develop a control solution that completes a task using minimum resource input. Intuitively, the optimal path is typically assumed to be a straight line. Though this can be the case for simple problems, "It has been shown for multi-body dynamic systems, like spacecraft antennas, that the optimal solution rarely follows a straight line because of the effects of nonlinearities and coupling between the various bodies" [6]. These problems are defined by their physical dynamics, written as a set of differential equations. The behavior of these systems is then further restricted by boundary conditions and constraints. Defined by Ross in his book, "a standard optimal control problem can be defined in terms of finding a dynamically feasible state-control function pair, $x(\cdot), u(\cdot)$, that transfers the state system, $\mathbf{x} \in \mathbb{R}^{N_x}$, from a given initial condition, $\mathbf{x}(t_o) = \mathbf{x}^o$, to a target condition $\mathbf{e}(\mathbf{x}_f, t_f) = 0$, while minimizing a given cost functional, J" [2]. Problem specifics vary widely with requirements and many other practical and engineering constraints, but irrespective of the details, the standard description of a continuous-time OCP is given as [2]:

$$\left.\begin{array}{ll} \mathbf{x} \in \mathbb{R}^{N_x} = (x_1, \ldots, x_{N_x}) & \mathbf{u} \in \mathbb{R}^{N_u} = (u_1, \ldots, u_{N_u}) \end{array}\right\} \textit{preamble}$$

$$\textit{Optimal Control Problem}: \begin{cases} \textit{Minimize}: & J[\mathbf{x}(\cdot), \mathbf{u}(\cdot), t_f] = & \\ & E[\mathbf{x}_o, \mathbf{x}_f, t_o, t_f] + \int_{t_o}^{t_f} F(\mathbf{x}(t), \mathbf{u}(t), t)dt & \} \textit{ cost} \\ \textit{Subject to}: & \dot{x} = f(\mathbf{x}(t), \mathbf{u}(t), t) & \} \mathcal{D} \\ & e^L \leq e(\mathbf{x}_o, \mathbf{x}_f, t_o, t_f) \leq e^U & \} \mathcal{E} \\ & \hbar^L \leq h(\mathbf{x}(t), \mathbf{u}(t), t) \leq \hbar^U & \} \mathcal{H} \end{cases}$$

The cost functional J is the summation of an end-point cost functional E and a running cost functional F. The inputs to this cost functional are the state variables ($\mathbf{x}$), the control variables ($\mathbf{u}$), time ($\mathbf{t}$) and the final time value (t_f).

The objective is to minimize J, subject to all of the linear and/or nonlinear constraints on the dynamics ($\mathcal{D}$), end-point events ($\mathcal{E}$), and path constraints ($\mathcal{H}$). For an optimal control solution, the constraints must be satisfied as a logical conjunction (see [23] for a discussion on the logical connectives). In other words, the logical expression $\mathcal{D} \wedge \mathcal{E} \wedge \mathcal{H}$ must be true for a feasible solution.

In order to use common solution techniques such as *collocation* or *shooting*, the OCP may need to be further developed into a BVP. The BVP can be developed using Pontryagin's Minimization Principle. While it does not solve the problem, Pontryagin's Principle provides all of the necessary conditions for a candidate solution to be optimal. These are defined by Ross [2] in his Theorem 2.1:

> **Theorem 2.1 (Pontryagin's Principle)** Given an optimal solution to Problem B, there exists an absolutely continuous covector function $\lambda(\cdot)$ and a covector ν that satisfy
>
> · the three Hamiltonian conditions;
>
> 1. Hamiltonian minimization condition
> 2. Hamiltonian value condition
> 3. Hamiltonian evolution equation,
>
> · the adjoint equations and
>
> · the transversality condition.

For a detailed discussion of the generalized steps in implementing Pontryagin's Principle to develop BVPs and necessary conditions for optimality in OCPs as defined above, see [2], [6], [24], [25]. Two specific implementations of these generalized steps are presented here for a 2D double integrator problem (Chapter 5) and for a 3D quaternion rotation (Chapter 6).

CHAPTER 4:
Development of the Boolean XOR End Constraint

In Chapter 3 the OCP was defined including a conjunction of problem dynamics ($\mathcal{D}$), end-point events ($\mathcal{E}$), and path constraints ($\mathcal{H}$). But in many aerospace problems, the event set $\mathcal{E}$ may be formed by a logical expression involving non-conjuctive operations including negations, disjunctions and/or exclusive-disjunctions across the end-point functions. Examples include selecting a feasible landing site from a set of alternatives, optimal reconfiguration of satellite formations, task selection and planning, or autonomous target engagement. An intuitive approach to solving such problems would be to enumerate across disjunctions. That is, first rewrite the original trajectory optimization problem as a collection of problems involving conjunctive constraints only. Then, solve the collection of problems to find the best solution.

This chapter presents a new approach for handling disjunctive end-point conditions as part of a single continuous-time trajectory optimization problem without the need to resort to a potentially computationally intensive brute-force enumeration. The idea is to employ a continuous representation of the otherwise discrete logic network (see, for example, [26]) as a means to model disjunctive operators over the entire admissible problem state-space. This new concept is first illustrated using the canonical double integrator model as a suitable abstraction for an aerospace system.

4.1 Continuous Representation of Discrete Logic Networks

An example logic network that can be found in many textbooks is shown in Figure 4.1. This logic network takes 3 inputs and implements the expression $(x_1 \vee x_2) \wedge (\bar{x}_2 \vee x_3)$ by using the 'AND', 'OR', and 'NOT' operations which are normally instantiated in terms of electronic gates. Each of these elementary logical operations can also be represented by an appropriately defined continuous surface (see [27], [28]). These surfaces can then be combined in much the same way as the electronic gates of Figure 4.1 to perform the needed logic operation [26].

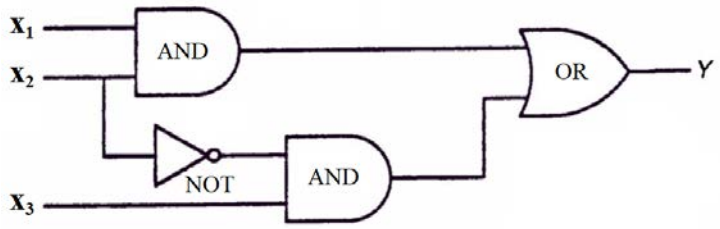


Figure 4.1: Example of a typical logic network

Amongst the various functional definitions that can be used to approximate discrete Boolean logic functions, radial basis functions (RBF) are perhaps one of the most simple,

$$\varphi(\mathbf{x}) = \sum_{i=1}^{N} a_i e^{-\beta||\mathbf{x}-\mathbf{c}_i||^2}$$

where N is the number of RBF elements needed to generate a continuous surface that adequately approximates the discrete logic; $\mathbf{c}_i$ locates the center of the surface peak i, and a_i and β are appropriately chosen surface weights. Parameter β controls the slope of the RBF defined surface to transition between the binary logic states. The example logic network in Figure 4.1 can be implemented using the RBF parameters shown in Table 4.1, the output of which is shown plotted in Figure 4.2.

Table 4.1: Radial basis function networks for basic logic functions.

operation	symbol	RBF parameters
AND		$\mathbf{a} = \{+1\},\ \mathbf{c} = \{(1,1)\}$ β = variable
OR		$\mathbf{a} = \{+1,+1,+1\},\ \mathbf{c} = \{(1,0),(1,0),(1,1)\}$ β = variable
XOR		$\mathbf{a} = \{+1,+1\},\ \mathbf{c} = \{(1,0),(1,0)\}$ β = variable
NOT		$\mathbf{a} = \{+1\},\ \mathbf{c} = \{0\}$ β = variable

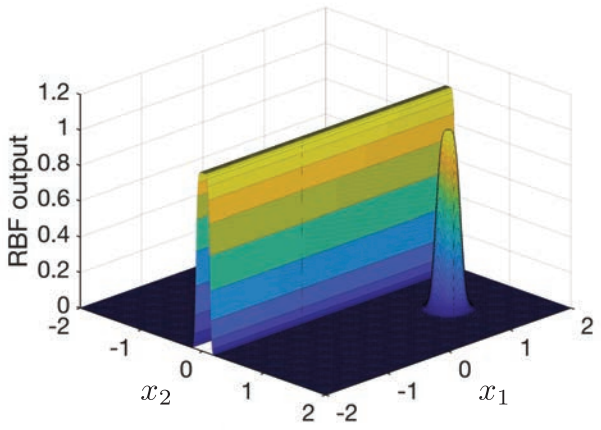


Figure 4.2: RBF representation of the network shown in Figure 4.1 for $x_3 = 1$ and $\beta = 12.5$.

4.2 Modeling a Two-Dimensional End-Point Disjunction

This section details the combination of a set of two or more terminal conditions (x_f,y_f) into a single value for use as a continuous representation of the discrete exclusive or (XOR) logic. In practice, end-point constraints of this type can be implemented in coded algorithms using *If* $\Rightarrow$ *Then or Case* logic. While this approach is easy to code, the results may be unpredictable because most optimal control solvers require a well-defined gradient.

Another approach is to train a neural network to approximate the XOR function:

$$e(x_f, y_f) = Net(x_f, y_f) = 0$$

To solve this problem, the neural network could be trained to solve an XOR problem (or any Boolean logic problem of any complexity), and the associated feed-forward equation of node weights, biases and activation functions can be extracted. This approach is implementable in a numerical solver because the gradients are the same as those needed for backpropogation during network training. Since backpropogation algorithms use differentiation, the extracted equations, while potentially complicated, can be differentiated to evaluate solution optimality.

For the XOR case, one such neural network based construction is simply the summation of two RBFs as described in the previous section. Each RBF produces a Gaussian bump

with a peak value of one at its center and that tails sharply to zero as the distance from the center increases. Summing two of these functions, each centered at a valid point, results in a desirable surface. Mathematically, the end-point disjunction for a set of two possible end-points ($(x_f, y_f) = (1, 0)$ XOR $(x_f, y_f) = (0, 1)$) can be modeled as:

$$f(x, y) = e^{\frac{-[(x)^2+(y-1)^2]}{2\sigma^2}} + e^{\frac{-[(x-1)^2+(y)^2]}{2\sigma^2}}$$

The surface has two RBF terms, one for each end-point. Inside each RBF is a summation of the squares of each term in the end-point. This being a two-dimensional problem, there are two terms in each RBF. This equation will expand for the more complicated quaternion (four-dimensional) case in Chapter 6.

This particular form of the error surface places gaussian bumps at the points (0,1) and (1,0) –the desired end-points– with the surface gradient controlled by σ. This allows for variation of the continuous representation to find the ideal compromise between approximating the discrete logic and the need for a smooth and continuous representation of the logic operator. Shown in Figure 4.3 is a representative surface using $\sigma = 0.3$.

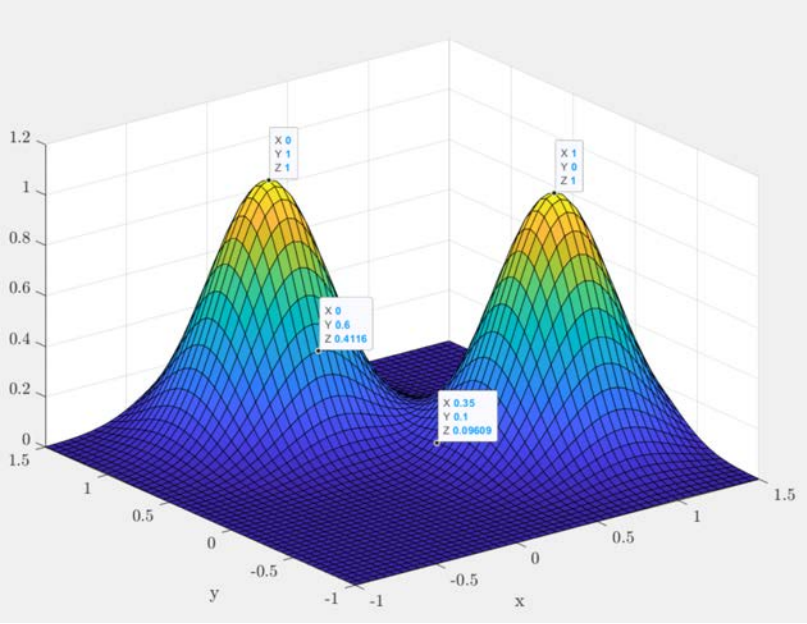


Figure 4.3: Error surface for end-points $(x_f, y_f) = (1, 0)$ XOR $(0, 1)$

CHAPTER 5:
Boolean XOR End Constraints in the Two-Dimensional Double Integrator Proxy Problem

This chapter describes how to insert the logical XOR end-point constraint into a two-dimensional double integrator continuous-time optimal control problem formulation with consideration for the requirements of Pontryagin's Minimization Principle. It then describes the process for implementing Pontryagin's Minimization Principle to develop a BVP and solves the problem.

5.1 Optimal Control Problem Definition and Analysis

An idealized body subject to Newtonian physics is randomly placed in a two-dimensional plane. This body is modeled using the ubiquitous double integrator where $\ddot{x} = u$; in this case $\ddot{x} = u_x$ and $\ddot{y} = u_y$. Consider the problem,shown in Figure 5.1, of finding the optimal control that moves the body from its random initial rest position: $x, y = (x_o, y_o)$ and $\dot{x}_o, \dot{y}_o = (0, 0)$ to rest at either valid end-point $x, y = [(x_{f_1}, y_{f_1}) \oplus (x_{f_2}, y_{f_2})]$ and $\dot{x}_f, \dot{y}_f = (0, 0)$ with minimum effort.

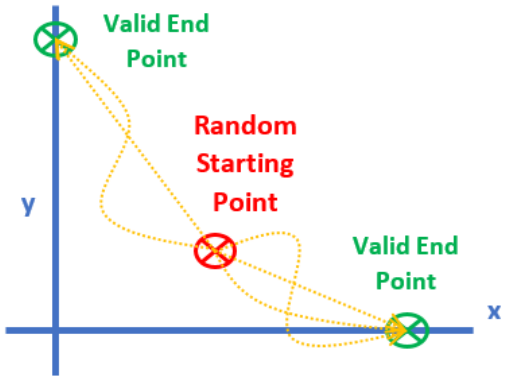


Figure 5.1: Two-dimensional end-point XOR problem

The desired 2-D double integrator system dynamics and required boundary conditions are then modeled as follows:

$$
\textit{System Dynamics} \begin{cases} \dot{x} &= \mathrm{v}_x \\ \dot{\mathrm{v}}_x &= u_x \\ \dot{y} &= \mathrm{v}_y \\ \dot{\mathrm{v}}_y &= u_y \end{cases} \qquad \textit{Boundary Conditions} \begin{cases} x_o, y_o &= (x^o, y^o) \\ \mathrm{v}_{x_o}, \mathrm{v}_{y_o} &= (0,0) \\ x_f, y_f &= P1 \oplus P2 \\ \mathrm{v}_{x_f}, \mathrm{v}_{y_f} &= (0,0) \end{cases}
$$

Optimal control of the double integrator is quite trivial (see [29]), but added complexity comes from the introduction of Boolean logic that defines a set of two acceptable boundary conditions. For initial conditions, the body is at rest at any random point in the 2-D space. The valid final position is defined as the exclusive or (XOR) of the two valid end-points. Using the RBF equation defined in the previous chapter and subtracting one to create a testable error equation results in the following end-point condition:

$$
e_{xor}(x_f, y_f) = e^{\frac{-[(x_f)^2+(y_f-1)^2]}{2\sigma^2}} + e^{\frac{-[(x_f-1)^2+(y_f)^2]}{2\sigma^2}} - 1 \tag{5.1}
$$

As the search algorithm drives the point mass to either of the valid end-points, the value returned by Equation 5.1 would decrease to approximately zero.

The dynamic optimization problem is now sufficiently described to allow for formulation and analysis using Pontryagin's Principle. The dynamic optimization problem of interest is given in the standard form [2] as follows:

$$
OCP: \begin{cases} \textit{Minimize}: & J[\mathbf{x}(\cdot), \mathbf{u}(\cdot), t_f] &= \frac{1}{2}\int_{t_o}^{t_f} ({u_x}^2 + {u_y}^2)\, dt \\ \textit{Subject to}: & \dot{x}, \dot{y} &= v_x, v_y \\ & \dot{\mathrm{v}}_x, \dot{\mathrm{v}}_y &= u_x, u_y \\ & \mathrm{v}_{x_o}, \mathrm{v}_{y_o}, \mathrm{v}_{x_f}, \mathrm{v}_{y_f} &= (0,0,0,0) \\ & t_o, t_f &= (0,5) \\ & e^{\frac{-[(x_f)^2+(y_f-1)^2]}{2\sigma^2}} + e^{\frac{-[(x_f-1)^2+(y_f)^2]}{2\sigma^2}} - 1 &= 0 \end{cases}
$$

5.1.1 Developing the Necessary Conditions

Along with developing the necessary conditions for optimality, solving this problem using shooting or collocation requires the formation of a BVP. Part of this process involves identifying additional boundary conditions as needed to properly form the BVP. These boundary condition are developed using Pontryagin's Principle and the new XOR constraint. The Hamiltonian for the OCP is constructed [2].

$$H(\boldsymbol{\lambda}, \boldsymbol{x}, \boldsymbol{u}, t) = F(\boldsymbol{x}, \boldsymbol{u}, t) + \boldsymbol{\lambda}^T f(\boldsymbol{x}, \boldsymbol{u}) = \frac{{u_x}^2}{2} + \frac{{u_y}^2}{2} + \lambda_x \mathrm{v}_x + \lambda_{\mathrm{v}x} u_x + \lambda_y \mathrm{v}_y + \lambda_{\mathrm{v}y} u_y \quad (5.2)$$

Evaluating the adjoint equations (see [2]) by taking the partial derivative of the Hamiltonian (Eq. 5.2) with respect to each state variable describes the time variation of the costates.

$$\begin{bmatrix} \frac{\partial H}{\partial x} \\ \frac{\partial H}{\partial \mathrm{v}_x} \\ \frac{\partial H}{\partial y} \\ \frac{\partial H}{\partial \mathrm{v}_y} \end{bmatrix} = -\begin{bmatrix} \dot{\lambda}_x \\ \dot{\lambda}_{\mathrm{v}x} \\ \dot{\lambda}_y \\ \dot{\lambda}_{\mathrm{v}y} \end{bmatrix} = \begin{bmatrix} 0 \\ \lambda_x \\ 0 \\ \lambda_y \end{bmatrix} \quad \therefore \quad \begin{bmatrix} \lambda_x \; is \; constant \\ Slope \; of \; \lambda_{\mathrm{v}x} = -\lambda_x \\ \lambda_y \; is \; constant \\ Slope \; of \; \lambda_{\mathrm{v}y} = -\lambda_y \end{bmatrix}$$

Minimizing the Hamiltonian by taking the partial derivatives with respect to the controls defines the time-varying behavior of the controls, in this case equal to the opposite of their associated velocity costate.

$$\frac{\partial H}{\partial u_x} = u_x + \lambda_{\mathrm{v}_x} = 0 \therefore u_x = -\lambda_{\mathrm{v}_x} \qquad \frac{\partial H}{\partial u_y} = u_y + \lambda_{\mathrm{v}_y} = 0 \therefore u_y = -\lambda_{\mathrm{v}_y}$$

Next, construct the Endpoint Lagrangian as in [2].

$$\bar{E}(\boldsymbol{\nu}, \mathbf{x}(t_f)) := E(\mathbf{x}(t_f)) + \nu^T e(\mathbf{x}(t_f))$$

After inserting the terminal boundary conditions:

$$\mathrm{v}_{x_f} = \mathrm{v}_{y_f} = 0, \quad t_f = 5,$$

$$e_{xor}(x_f, y_f) = e^{\frac{-[(x_f)^2 + (y_f - 1)^2]}{2\sigma^2}} + e^{\frac{-[(x_f - 1)^2 + (y_f)^2]}{2\sigma^2}} - 1$$

the Endpoint Lagrangian becomes:

$$\begin{aligned}\bar{E}(\boldsymbol{\nu}, \boldsymbol{x_f}, t_f) = \nu_{v_x}(\mathrm{v}_{x_f} - 0) + \nu_{v_y}(\mathrm{v}_{y_f} - 0) + \nu_{t_f}(t_f - 5)\ldots \\ +\nu_e\left(e^{\frac{-[(x_f)^2+(y_f-1)^2]}{2\sigma^2}} + e^{\frac{-[(x_f-1)^2+(y_f)^2]}{2\sigma^2}} - 1\right)\end{aligned} \tag{5.3}$$

The Hamiltonian Value Condition can be applied to determine the final Hamiltonian value by taking the partial derivative of the Endpoint Lagrangian (Eq. 5.3) with respect to t_f.

$$H(t_f) = \frac{-\partial \bar{E}}{\partial t_f} = -\nu_{t_f}$$

The Hamiltonian should be equal to some unknown constant (ν_{t_f}) at t_f. Taking the partial derivative of the minimized Hamiltonian with respect to time gives insight into how the Hamiltonian will evolve.

$$\frac{d\mathcal{H}}{dt} = \frac{\partial H}{\partial t} = 0$$

These last two conditions show that the Hamiltonian should be constant over the entire trajectory.

Transversality analysis is used to determine the final values of the individual costates (λ), which are equal to the partial derivative of the Endpoint Lagrangian with respect to the final states.

$$\lambda_x(t_f) = \frac{\partial \bar{E}}{\partial x_f} = \frac{-\nu_e}{\sigma^2}\left((x_f)\overbrace{e^{\frac{-[(x_f)^2+(y_f-1)^2]}{2\sigma^2}}}^{A} + (x_f - 1)\overbrace{e^{\frac{-[(x_f-1)^2+(y_f)^2]}{2\sigma^2}}}^{B}]\right)$$

$$\lambda_y(t_f) = \frac{\partial \bar{E}}{\partial y_f} = \frac{-\nu_e}{\sigma^2}\left((y_f - 1)\overbrace{e^{\frac{-[(x_f)^2+(y_f-1)^2]}{2\sigma^2}}}^{A} + (y_f)\overbrace{e^{\frac{-[(x_f-1)^2+(y_f)^2]}{2\sigma^2}}}^{B}]\right)$$

By inspection, the position costates can be simplified. This is because for a given x_f, y_f the complicated fractional terms are the same in each derivative, so they can be rewritten as arbitrary constants A and B as shown:

$$\begin{aligned} \lambda_x(t_f) &= -\tfrac{\nu_e}{\sigma^2}\left(A * (x_f) + B * (x_f - 1)\right) \\ \lambda_y(t_f) &= -\tfrac{\nu_e}{\sigma^2}\left(A * (y_f - 1) + B * (y_f)\right) \end{aligned} \tag{5.4}$$

The final position costates are equal to some unknown constant value and are coupled due to the disjunctive end-point error function. The final velocity costate values are equal to some arbitrary constants.

$$\lambda_{\mathrm{v}_x}(t_f) = \frac{\partial \bar{E}}{\partial \mathrm{v}_{x_f}} = \nu_{x_f}$$

$$\lambda_{\mathrm{v}_y}(t_f) = \frac{\partial \bar{E}}{\partial \mathrm{v}_{y_f}} = \nu_{y_f}$$

The coupled nature of the position costates can be leveraged to generate the needed event condition for formulation of the BVP. Rearranging the position costate equations (5.4) to isolate the ν_e coefficient terms:

$$\frac{-\nu_e}{\sigma^2} = \frac{\lambda_x(t_f)}{((x_f)(A+B) - B}$$

$$\frac{-\nu_e}{\sigma^2} = \frac{\lambda_y(t_f)}{((y_f)(A+B) - A}$$

After substituting, cross multiplying and rearranging:

$$\lambda_x(t_f) \cdot [y_f(A+B) - A] - \lambda_y(t_f) \cdot [x_f(A+B) - B] = 0 \tag{5.5}$$

With the identification of an eighth event condition, the BVP is now sufficiently defined to allow coding and analysis via an algorithmic solver.

$$
XOR\ BVP \quad \begin{cases}
\dot{x}, \dot{y} & = & v_x, v_y \\
\dot{\mathrm{v}}_x, \dot{\mathrm{v}}_y & = & u_x, u_y \\
\dot{\lambda}_x, \dot{\lambda}_y & = & 0, 0 \\
\dot{\lambda}_{\mathrm{v}x}, \dot{\lambda}_{\mathrm{v}y} & = & \lambda_x, \lambda_y \\
t_o, t_f & = & (0, 5) \\
x_o, y_o & = & x^o, y^o \\
e^{\frac{-[(x_f)^2+(y_f-1)^2]}{2\sigma^2}} + e^{\frac{-[(x_f-1)^2+(y_f)^2]}{2\sigma^2}} - 1 & = & 0 \\
\mathrm{v}_{x_o}, \mathrm{v}_{y_o}, \mathrm{v}_{x_f}, \mathrm{v}_{y_f} & = & (0, 0, 0, 0) \\
\lambda_x(t_f) \cdot [y_f(A+B) - A] & = & \lambda_y(t_f) \cdot [x_f(A+B) - B]
\end{cases}
$$

5.1.2 Simulation Results

The solution to the XOR OCP – with an end-point disjunciton – is the minimum-effort state-control function pair $t \rightarrow (\mathbf{x}, \mathbf{u})$, which allows the point mass to be moved to the closer of the two possible end-points. In contrast to enumerating over the possibilities, the use of continuous representations of discrete logic operators allows the correct end-point to be chosen autonomously as part of the solution process. The inputs to the error surface function are the state variables x_f and y_f.

In order to test the new concept of using a continuous representation of the discrete end-point disjunction, a series of dynamic optimization problems were solved. The first problem used a randomly generated starting point that was significantly closer to the $(x_f, y_f) = (0, 1)$ end-point. Such a point was chosen because it is known that the minimum effort solution would be to move the double integrator system to the point (0,1). The OCP with the end-point disjunction was solved using DIDO [30]. Twenty nodes were used because the solution, when propagated through the system dynamics, resulted in the correct end-point conditions (x_f, y_f) when rounded at the third decimal point. From the trajectory shown in Figure 5.2 it is clear that the problem was properly solved with the selection of the closer end-point.

Verification that the calculated solution is optimal, per the procedure outlined by Ross [2], is done by plotting the necessary conditions associated with Pontryagin's Principle. Noting the

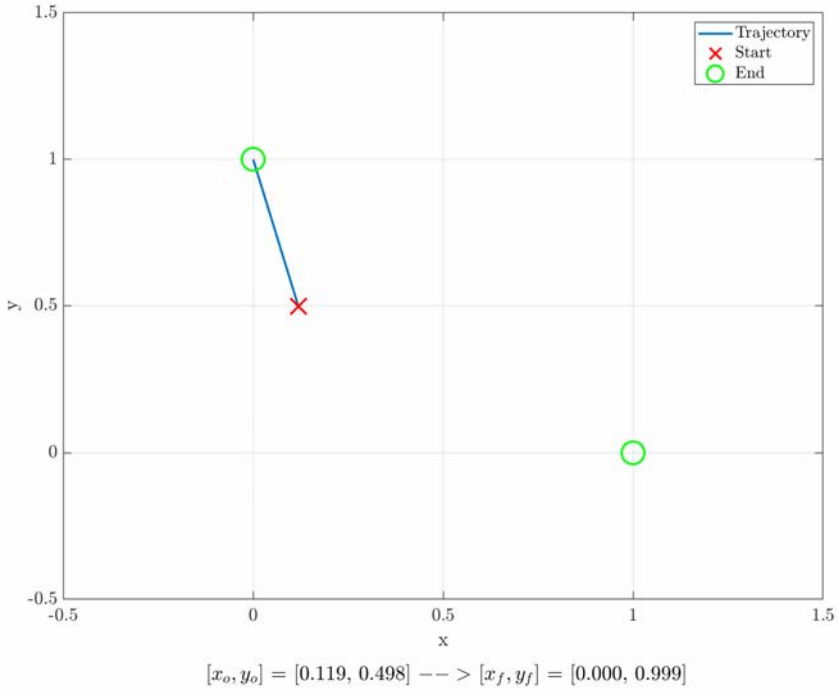


Figure 5.2: Example solution to the double integrator proxy problem showing that the correct solution was achieved.

plot scale, Figure 5.3 shows that the Hamiltonian, as expected, is approximately constant.

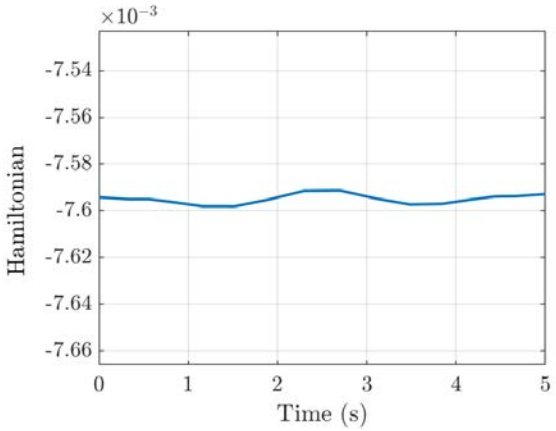
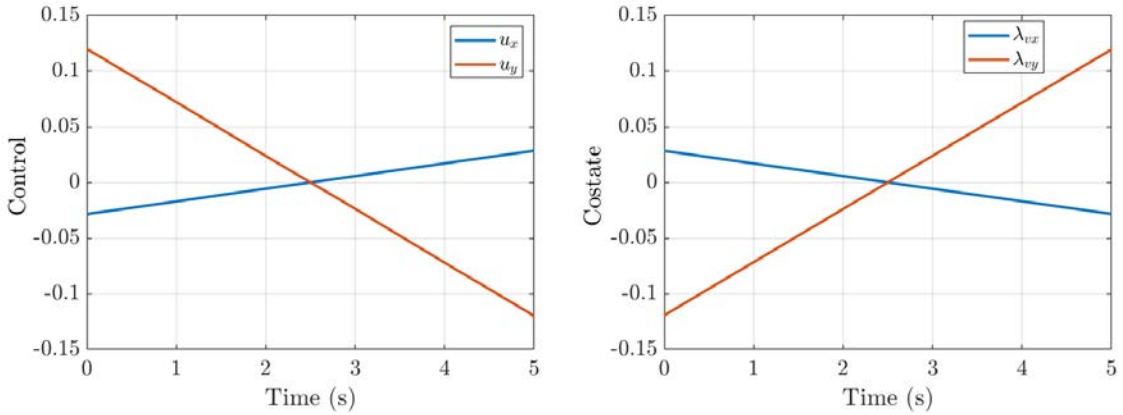


Figure 5.3: Constant Hamiltonian evolution as required for optimality of the double integrator proxy problem.

Shown in Figure 5.4, the minimization conditions, $u_x = -\lambda_{vx}$ and $u_y = -\lambda_{vy}$, are met.



(a) Control trajectories (b) Velocity costates

Figure 5.4: Control (u) and velocity costate plots validating the $u = -\lambda_v$ relationships for the double integrator proxy problem.

Figure 5.5 shows that the λ_x and λ_y costate trajectories are constant as expected and when compared with 5.4a they validate the $\dot{\lambda}_{vx} \approx -\lambda_x$ and $\dot{\lambda}_{vy} \approx -\lambda_y$ conditions.

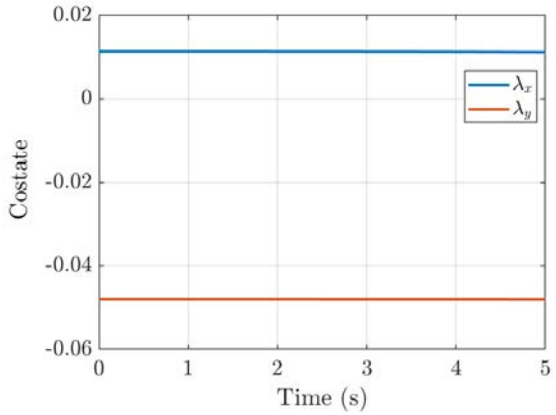


Figure 5.5: Constant position costates equal to the slope of their control (u) for the double integrator proxy problem.

The position transversality condition (Eq. 5.5), using the algorithm values for final position and position costate, is approximately equal to zero ($-2.3893E^{-07}$), therefore all of the necessary conditions for optimality established using Pontryagin's Principle [2] are met.

To determine if a solution to the other valid end-point could be generated, a second problem was solved with a randomly generated starting point that was significantly closer to the

other end-point $(x_f, y_f) = (1, 0)$. Examining Figure 5.6, it is again clear that the problem was properly solved with the selection of the closer end-point.

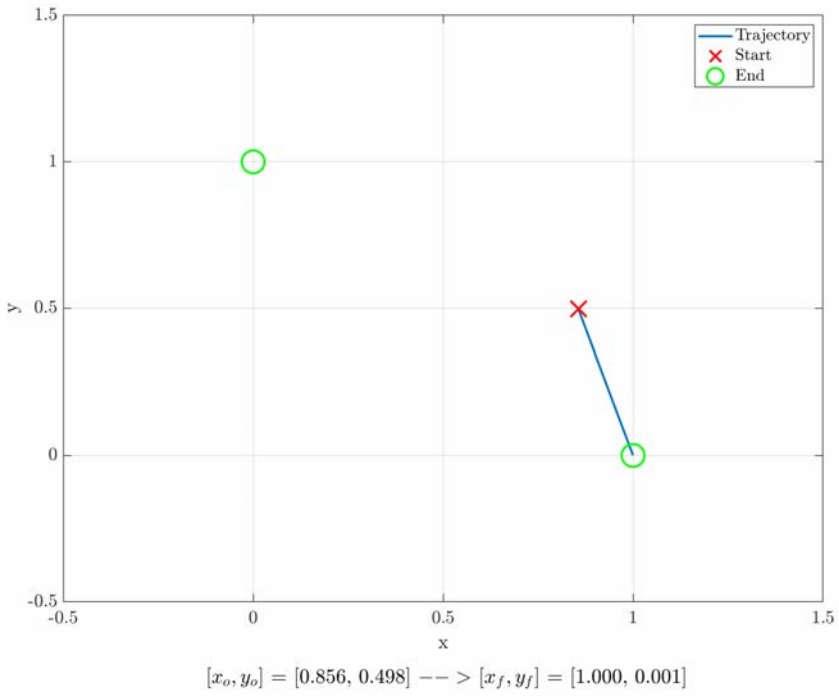


Figure 5.6: Example solution to the double integrator proxy problem showing that the correct solution was achieved.

Figure 5.7 again validates that the Hamiltonian is approximately constant.

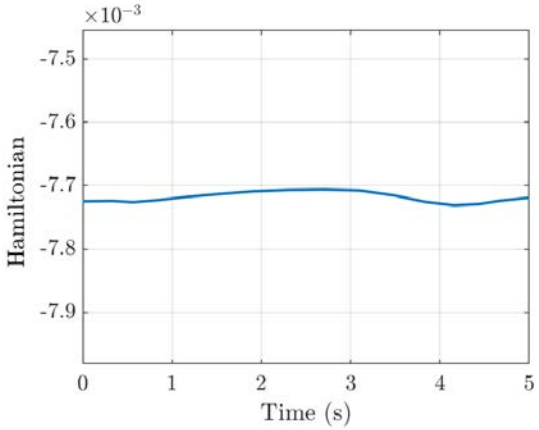
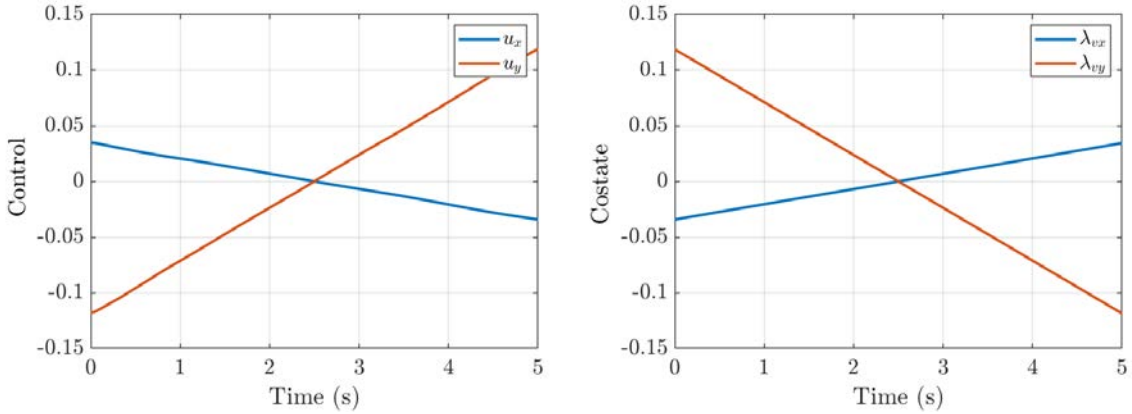


Figure 5.7: Constant Hamiltonian evolution as required for optimality of the double integrator proxy problem.

Shown in Figure 5.8, the minimization conditions, $u_x = -\lambda_{vx}$ and $u_y = -\lambda_{vy}$, are again met.



(a) Control trajectories (b) Velocity costates

Figure 5.8: Control (u) and velocity costate plots validating the $u = -\lambda_v$ relationships for the double integrator proxy problem.

Examining Figures 5.8a and 5.9, the λ_x and λ_y costate trajectories are again constant as expected, with $\dot{\lambda}_{vx} \approx -\lambda_x$ and $\dot{\lambda}_{vy} \approx -\lambda_y$.

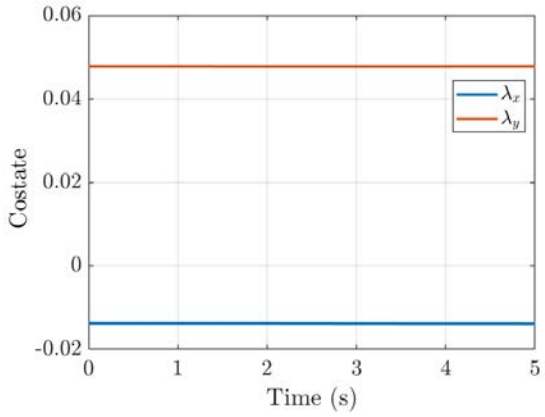


Figure 5.9: Constant position costates equal to the slope of their control (u) for the double integrator proxy problem.

The position transversality condition (Eq. 5.5) was again calculated and is approximately equal to zero ($7.7299E^{-07}$). All of the necessary conditions for optimality established using Pontryagin's Principle [2] are again met.

Defining which end-point is correct for given initial conditions is as simple as evaluating the Euclidean distance between the starting point and each end-point, and then selecting the shorter distance. In this case, that analysis divides the Cartesian plane along the diagonal line $y = x$, along which the distance between the two end-points in equal and propagating to either end-point results in an optimal solution.

To illustrate how the correct solution can be found irrespective of the starting point, the simulation was repeated, generating random starting points throughout the input domain space. Figure 5.10 shows that for each initial position, the proposed OCP formulation that utilized a continuous surface to model the end-point disjunction made it possible to find the minimum effort end-point and generate a control trajectory solution that meets all necessary conditions for optimality.

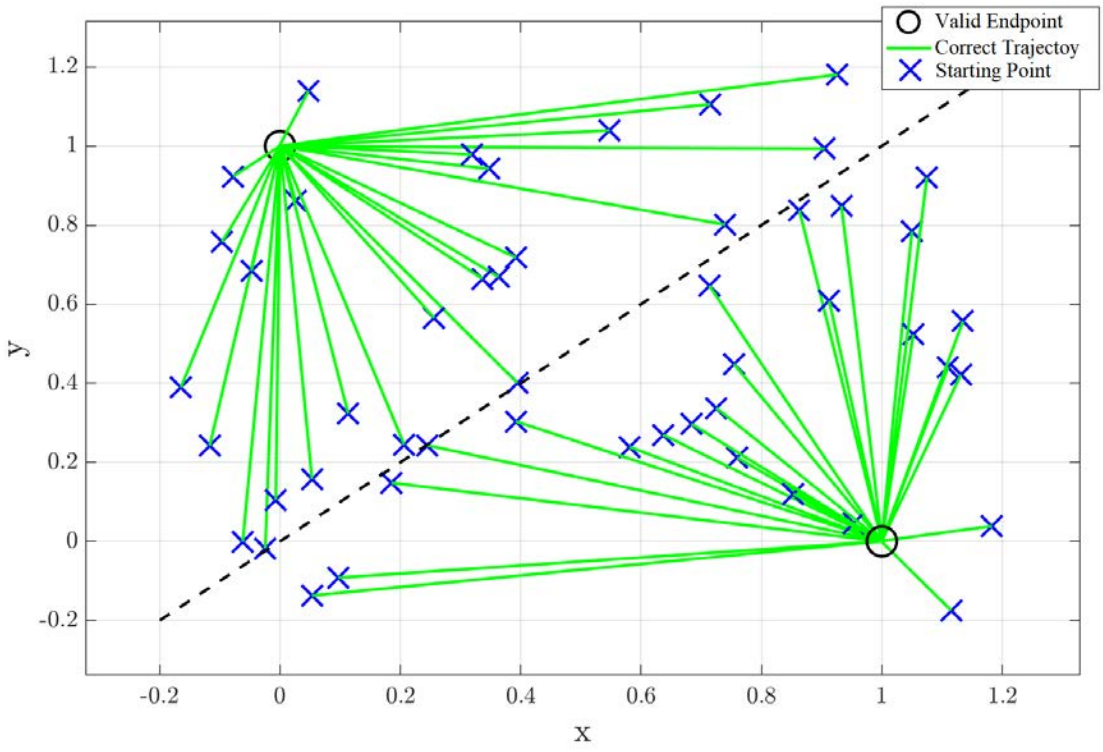


Figure 5.10: Trajectory mapping illustrating the ability to select the correct end-point by modeling the end-point disjunction as a continuous function.

5.2 Summary

The simulation results presented in this chapter support the conclusion that this approach for implementing the exclusive disjunction (XOR) end-point constraint works as desired. In each case the coded algorithm was able to use the RBF defined error surface to drive the point mass to the optimal, minimum cost, end-point. The next logical step is to implement this end-point error surface in a more complex problem.

CHAPTER 6:
Boolean XOR End Constraints in the Quaternion Reorientation Problem

This chapter expands on the use of the end-point error surface for implementing functional logic in more complex optimal control problems. This is done by defining a dynamics model for spacecraft quaternion-based rotation, validating that model, and commanding rotations with the non-conjunctive end-point constraints. Control constraints limit four reaction wheel torques (AND), while an end-point event condition allows the final quaternion to be either of two valid options (XOR). The problem is then fully analyzed using Pontryagin's Principle to identify the missing boundary condition and necessary conditions for optimality, and then coded in an algorithmic solver. It is then shown that the algorithm can independently identify the minimum time or minimum effort rotation and calculate an optimal control trajectory to rotate the model to that final quaternion.

For this analysis the spacecraft body frame is initially aligned with the inertial frame. Control constraints on the torque applied to the reaction wheels effectively slow the spacecraft rotation and are intuitive to understand but difficult to visualize graphically. To begin to develop a valid solution for this scenario, a quaternion-based model for rigid-body spacecraft rotation is developed. This model is defined for spacecraft with four reaction wheels in the typical pyramid configuration aligned along the spacecraft yaw axis and uses differential equations for rotational dynamics and attitude kinematics. A series of tests are then presented to validate that the model well describes the rotational system. The entire constraint set is defined and Pontryagin's Principle is then used to develop the necessary conditions for optimality. A coded algorithm is then used to determine optimal control trajectories for different combinations of rotations using the entire set constraints. The results are then analyzed to determine if the necessary conditions for optimality are met.

6.1 Quaternion Model Development and Validation

This section outlines the development of the spacecraft dynamics model using differential equations for attitude kinematics and rotational dynamics as defined in [8], [31]. This

is done using quaternions to define the spacecraft attitude with the following differential equation [8], [31], [32], such that the time rate of change of the quaternion is given as

$$\dot{\mathbf{q}} = \mathbf{Q}(\omega)\mathbf{q} \tag{6.1}$$

$\mathbf{Q}(\omega)$ is the anti-symmetric matrix of spacecraft angular rates shown below.

$$\mathbf{Q}(\omega) = \frac{1}{2}\begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix}$$

Euler's equations for rotational motion are, [8] [31]

$$\mathbf{I}_{sc}\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega}_{sc} \times (\mathbf{I}_{sc}\boldsymbol{\omega} + {}^{B}\mathbf{h}_{w}) - \boldsymbol{\tau}_{rw}^{B}$$

where $\mathbf{I}_{sc}$ is the spacecraft inertia matrix, $\dot{\boldsymbol{\omega}}$ is the spacecraft angular acceleration and $\boldsymbol{\omega}$ is the spacecraft angular velocity. Additionally, $\mathbf{h}$ is the reaction wheel angular momentum and $\boldsymbol{\tau}_{rw}^{B}$ is the reaction wheel control torque in the body-fixed coordinate frame, in a system with no external disturbance torques, henceforth referred to as spacecraft torque ($\boldsymbol{\tau}_{sc}$). All are expressed in vector form. Rearranging for spacecraft angular acceleration:

$$\dot{\boldsymbol{\omega}} = -\mathbf{I}_{sc}^{-1}[\boldsymbol{\omega} \times (\mathbf{I}_{sc}\boldsymbol{\omega} + \mathbf{h}) - \boldsymbol{\tau}_{sc}]$$

Since angular momentum is conserved for a zero net-bias system, the only torques on the body are those from the reaction wheels such that $\tau_{sc} = -\mathbf{Z}\tau_{rw}$, and the spacecraft angular acceleration simplifies to:

$$\dot{\boldsymbol{\omega}} = -\mathbf{I}_{sc}^{-1}\mathbf{Z}\tau_{rw} \tag{6.2}$$

Since $\mathbf{h} = \mathbf{I}\boldsymbol{\omega}$, the body-fixed reaction wheel angular momentum can be defined by multiplying the reaction wheel angular rate ($\boldsymbol{\Omega}$) and wheel inertia vector and then transforming that product into the body-fixed frame.

$${}^{B}\mathbf{h}_{w} = \mathbf{Z}(\mathbf{I}_{w}\boldsymbol{\Omega})$$

where $\mathbf{Z}$ is the matrix that rotates the reaction wheel values into the body-fixed coordinate system. Using the conservation of angular momentum, the derivative of the wheel angular momentum is equal to the sum of the torques on the body.

$$ {}^{B}\dot{\mathbf{h}}_w = \frac{d({}^{B}\mathbf{h}_w)}{dt} = \mathbf{Z}\boldsymbol{I}_w\dot{\boldsymbol{\Omega}}_w = -\boldsymbol{\tau}_{sc} = \mathbf{Z}\boldsymbol{\tau}_w $$

Solving for the reaction wheel angular acceleration:

$$ \mathbf{Z}\boldsymbol{I}_w\dot{\boldsymbol{\Omega}}_w = \mathbf{Z}\boldsymbol{\tau}_w $$

$$ \dot{\boldsymbol{\Omega}}_w = \boldsymbol{I}_w^{-1}\boldsymbol{\tau}_w \tag{6.3} $$

As discussed earlier four reaction wheel torques are modeled in a typical pyramid configuration aligned with the yaw axis as in [8], [31] using the following $\boldsymbol{Z}$ matrix to apply a torque to the model,

$$ \mathbf{Z} = \begin{bmatrix} 0.8192 & 0.8192 & -0.8192 & -0.8192 \\ 0.4056 & -0.4056 & 0.4056 & -0.4056 \\ 0.4056 & 0.4056 & 0.4056 & 0.4056 \end{bmatrix} $$

and the simplified inertia properties in the $\boldsymbol{I}$ matrix:

$$ \mathbf{I}_{sc} = \begin{bmatrix} I_{XX} & 0 & 0 \\ 0 & I_{YY} & 0 \\ 0 & 0 & I_{ZZ} \end{bmatrix} \; kg \cdot m^2 $$

The model is then defined by the state vector (Eqs. 6.1 - 6.3) and the control variable.

$$ \mathbf{x} = \begin{bmatrix} \dot{q} & \dot{\omega} & \dot{\boldsymbol{\Omega}} \end{bmatrix}^T \in \mathbb{R}^{11} $$

$$ \mathbf{u} = \begin{bmatrix} \boldsymbol{\tau}_{rw} \end{bmatrix}^T \in \mathbb{R}^4 $$

Assuming a zero-net bias system would allow for the use of the simpler form for the spacecraft angular acceleration ($\dot{\boldsymbol{\omega}}$), but for this relatively simple analysis the full version is

used. The model dynamics definition becomes:

$$\textit{Problem Dynamics} \begin{cases} \dot{\mathbf{q}} = \mathbf{Q}(\omega)\mathbf{q} \\ \dot{\boldsymbol{\omega}} = -\mathbf{I}_{sc}^{-1}[\boldsymbol{\omega} \times (\mathbf{I}_{sc}\boldsymbol{\omega} + \mathbf{h}) - \boldsymbol{\tau}_{sc}] \\ \dot{\boldsymbol{\Omega}} = \mathbf{I}_{w}^{-1}\mathbf{u} \end{cases}$$

6.1.1 Model Validation

To test the spacecraft rotational model, control torques (Figure 6.1) were defined and the system was propagated using Matlab's *ODE45* numerical integrator. Figure 6.2a shows, as expected, that any constant torque applied around an axis produced a linear angular rotation rate.

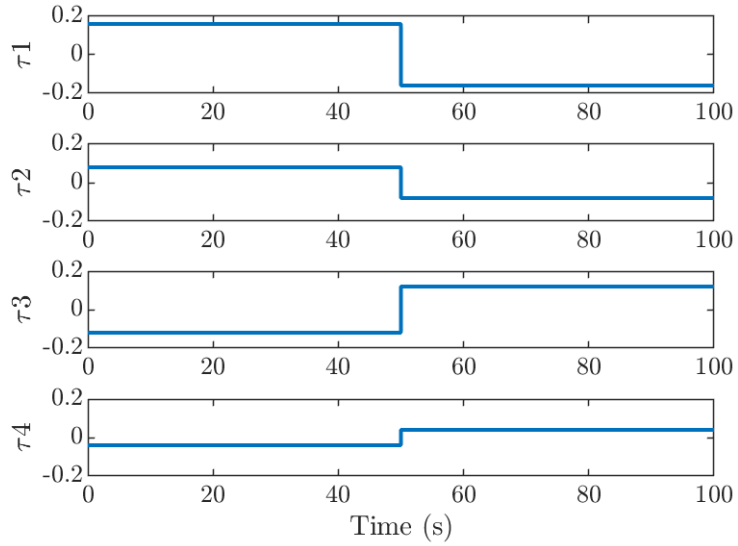
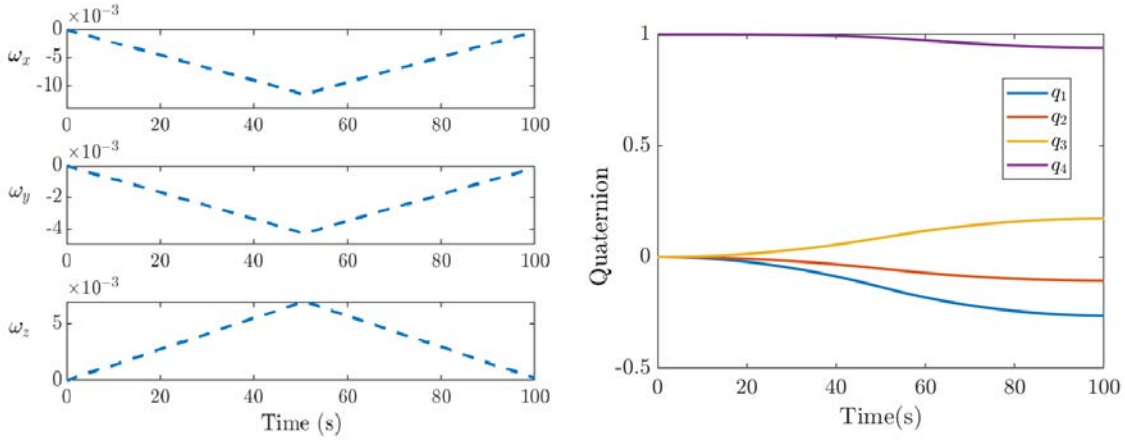


Figure 6.1: Quaternion rotation model - test control u = $\boldsymbol{\tau}$

Figures 6.2a and 6.2b show that a positive spacecraft angular rate causes the corresponding quaternion to move in the positive direction, and vice-versa.

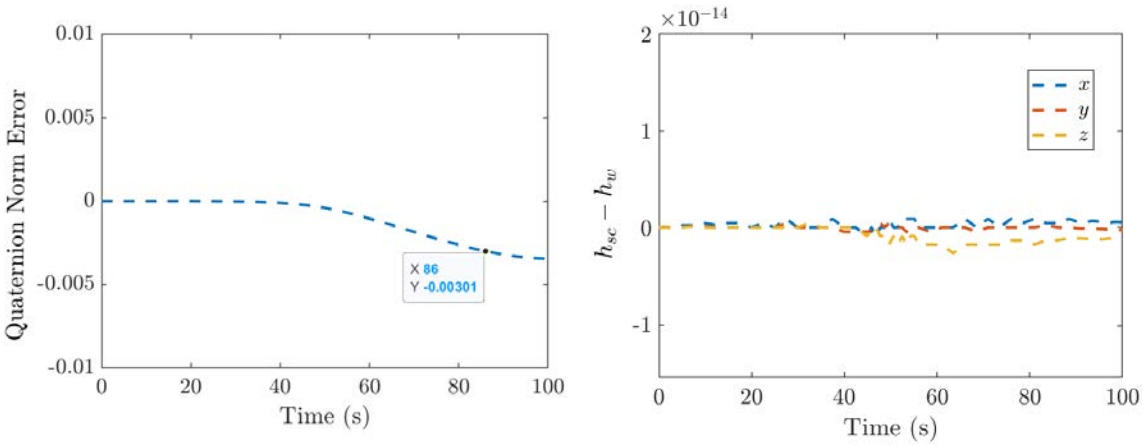
As always, changes to q_4 should maintain the norm of the quaternion very close to one. Figure 6.3a validates that that is in fact the case as the norm error is very close to zero. Plotting the difference between the spacecraft and the reaction wheel angular momentum in



(a) Spacecraft angular rate (rad/s) (b) Quaternion evolution

Figure 6.2: Quaternion rotation model - spacecraft dynamic response

each coordinate direction validates that angular momentum was conserved (Figure 6.3b).



(a) Quaternion norm error (b) Conservation of angular momentum

Figure 6.3: Quaternion rotation model - quaternion evolution and error norm

The validation arguments support the conclusion that the model behaves as expected to simulate a rigid body rotating in space. Angular momentum bias is zero, so the simpler form for angular acceleration would be a good candidate for resource-intensive simulations. This model can now be used in solving constrained optimization problems. The next step is to identify the necessary conditions for optimality.

6.1.2 Optimal Control Problem Definition and Analysis

The last requirement to finalize the quaternion rotation OCP is to define the XOR end-point constraint.

$$\boldsymbol{q}^f = \begin{bmatrix} q_{f_1}(1) \\ q_{f_1}(2) \\ q_{f_1}(3) \\ q_{f_1}(4) \end{bmatrix} \quad \oplus \quad \begin{bmatrix} q_{f_2}(1) \\ q_{f_2}(2) \\ q_{f_2}(3) \\ q_{f_2}(4) \end{bmatrix}$$

Expanding the RBF surface discussed in Chapter 4 to a four-dimension end-point (four terms in each RBF), and again two valid end-points (two RBFs), the end-point error constraint (Eq. 6.4) becomes:

$$f(\boldsymbol{q}^f) = \begin{cases} e^{\frac{-[(q^f(1)-q_{f_1}(1))^2+(q^f(2)-q_{f_1}(2))^2+(q^f(3)-q_{f_1}(3))^2+(q^f(4)-q_{f_1}(4))^2]}{2\sigma^2}} \dots \\ + e^{\frac{-[(q^f(1)-q_{f_2}(1))^2+(q^f(2)-q_{f_2}(2))^2+(q^f(3)-q_{f_2}(3))^2+(q^f(4)-q_{f_2}(4))^2]}{2\sigma^2}} - 1 \end{cases} \tag{6.4}$$

6.2 The Minimum Effort Problem

Performing a full analysis via Pontryagin's Principle is now possible using the quaternion XOR end-point constraint. A minimum effort cost functional is defined: [2]

$$J[\mathbf{x}(\cdot), \mathbf{u}(\cdot), t_f] = E(x_f) + \int_{t_o}^{t_f} F(x,u)dt = \frac{1}{2}\int_0^{t_f} ({u_1}^2 + {u_2}^2 + {u_3}^2 + {u_4}^2)\, dt$$

with the end-point constraints:

$$\boldsymbol{\omega}_o = \boldsymbol{\omega}_f = (0,0,0), \qquad \boldsymbol{\Omega}_o = \boldsymbol{\Omega}_f = (0,0,0,0), \qquad \boldsymbol{q}_o = (0,0,0,1)$$

$$e_{xor}(\boldsymbol{q}^f) = e^{\frac{-[(q^f(1)-q_{f_1}(1))^2+(q^f(2)-q_{f_1}(2))^2+(q^f(3)-q_{f_1}(3))^2+(q^f(4)-q_{f_1}(4))^2]}{2\sigma^2}} \dots + e^{\frac{-[(q^f(1)-q_{f_2}(1))^2+(q^f(2)-q_{f_2}(2))^2+(q^f(3)-q_{f_2}(3))^2+(q^f(4)-q_{f_2}(4))^2]}{2\sigma^2}} - 1 = 0$$

The Lagrangian of the Hamiltonian is constructed using the generic form (Eqn. 6.5). [2]

$$\bar{H}(\boldsymbol{\mu}, \boldsymbol{\lambda}, \boldsymbol{x}, \boldsymbol{u}, t) := H(\boldsymbol{\lambda}, \boldsymbol{x}, \boldsymbol{u}, t) + \mu^T h(\boldsymbol{x}, \boldsymbol{u}, t) = F(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{\lambda}^T f(\boldsymbol{x}, \boldsymbol{u}) + \mu^T \hbar(\boldsymbol{x}, \boldsymbol{u}, t) \tag{6.5}$$

$$
\begin{aligned}
\bar{H}(\boldsymbol{\mu}, \boldsymbol{\lambda}, \boldsymbol{x}, \boldsymbol{u}, t) = \quad & \tfrac{{u_1}^2}{2} + \tfrac{{u_2}^2}{2} + \tfrac{{u_3}^2}{2} + \tfrac{{u_4}^2}{2} \ldots \\
& + \tfrac{1}{2}[\lambda_{q_1}(q_4\omega_x - q_3\omega_y + q_2\omega_z) + \lambda_{q_2}(q_3\omega_x + q_4\omega_y - q_1\omega_z)] \ldots \\
& + \tfrac{1}{2}[\lambda_{q_3}(-q_2\omega_x + q_1\omega_y + q_4\omega_z) + \lambda_{q_4}(-q_1\omega_x - q_2\omega_y - q_3\omega_z)] \ldots \\
& - \lambda_{\omega_x} I_{sc_{11}}^{-1}[Z_{11}u_1 + Z_{12}u_2 + Z_{13}u_3 + Z_{14}u_4] \ldots \\
& - \lambda_{\omega_y} I_{sc_{22}}^{-1}[Z_{21}u_1 + Z_{22}u_3 + Z_{23}u_3 + Z_{24}u_4] \ldots \\
& - \lambda_{\omega_z} I_{sc_{33}}^{-1}[Z_{31}u_1 + Z_{32}u_2 + Z_{33}u_3 + Z_{34}u_4] \ldots \\
& + \lambda_{\Omega_1} I_{w_{11}}^{-1} u_1 + \lambda_{\Omega_2} I_{w_{22}}^{-1} u_2 + \lambda_{\Omega_3} I_{w_{33}}^{-1} u_3 + \lambda_{\Omega_4} I_{w_{44}}^{-1} u_4 \ldots \\
& + \mu_1 u_1 + \mu_2 u_2 + \mu_3 u_3 + \mu_4 u_4
\end{aligned}
\tag{6.6}
$$

Evaluating the adjoint conditions by taking the partial derivative of the Lagrangian of the Hamiltonian (eqn. 6.6) with respect to the states describes the time varying costate behavior.

$$
\frac{\partial \bar{H}}{\partial q_1} = -\dot{\lambda_{q_1}} = \frac{1}{2}[-\lambda_{q_2}\omega_z + \lambda_{q_3}\omega_y - \lambda_{q_4}\omega_x] \qquad \frac{\partial \bar{H}}{\partial q_3} = -\dot{\lambda_{q_3}} = \frac{1}{2}[\lambda_{q_1}\omega_y + \lambda_{q_2}\omega_x - \lambda_{q_4}\omega_z]
$$

$$
\frac{\partial \bar{H}}{\partial q_2} = -\dot{\lambda_{q_2}} = \frac{1}{2}[\lambda_{q_1}\omega_z - \lambda_{q_3}\omega_x - \lambda_{q_4}\omega_y] \qquad \frac{\partial \bar{H}}{\partial q_4} = -\dot{\lambda_{q_4}} = \frac{1}{2}[\lambda_{q_1}\omega_x + \lambda_{q_2}\omega_y + \lambda_{q_3}\omega_z]
$$

$$
\frac{\partial \bar{H}}{\partial \omega_x} = -\dot{\lambda_{\omega_x}} = \frac{1}{2}[\lambda_{q_1}q_4 + \lambda_{q_2}q_3 - \lambda_{q_3}q_2 - \lambda_{q_4}q_1]
$$

$$
\frac{\partial \bar{H}}{\partial \omega_y} = -\dot{\lambda_{\omega_y}} = \frac{1}{2}[-\lambda_{q_1}q_3 + \lambda_{q_2}q_4 + \lambda_{q_3}q_1 - \lambda_{q_4}q_2]
$$

$$
\frac{\partial \bar{H}}{\partial \omega_z} = -\dot{\lambda_{\omega_z}} = \frac{1}{2}[\lambda_{q_1}q_2 - \lambda_{q_2}q_1 + \lambda_{q_3}q_4 - \lambda_{q_4}q_3]
$$

$$
\frac{\partial \bar{H}}{\partial \Omega_1} = -\dot{\lambda_{\Omega_1}} = 0 \qquad \frac{\partial \bar{H}}{\partial \Omega_2} = -\dot{\lambda_{\Omega_2}} = 0 \qquad \frac{\partial \bar{H}}{\partial \Omega_3} = -\dot{\lambda_{\Omega_3}} = 0 \qquad \frac{\partial \bar{H}}{\partial \Omega_4} = -\dot{\lambda_{\Omega_4}} = 0
$$

Often these equations are helpful in validating results, but not always. In this case the analysis has provided four easily testable, necessary conditions for the $\dot{\lambda_\Omega}$ behavior, namely that the values for λ_{Ω_i} are all constants.

For the Lagrangian of the Hamiltonian to be minimized, both the stationarity and complementarity conditions must be met: [2]

$$
Stationarity\ Condition: \quad \frac{\partial \bar{H}}{\partial u_i} = 0 \tag{6.7}
$$

$$\begin{matrix} Control\ \ u=\tau \\ Complementarity: \\ Condition \end{matrix} \quad \mu_{1-4} \begin{cases} \leq 0 & if & u_i = -0.16 \\ = 0 & if & -0.16 < u_i < 0.16 \\ \geq 0 & if & u_i = 0.16 \end{cases} \tag{6.8}$$

These conditions are referred to as Karush–Kuhn–Tucker (KKT) conditions [2], [8], [31], and they generate two additional sets of conditions that can be easily tested.

First, the stationarity condition can be graphed to show that the Lagrangian of the Hamiltonian is constant over the trajectory. This is done by verifying that the derivative with respect to each control variable is mathematically zero over the trajectory.

$$\begin{aligned} \frac{\partial \bar{H}}{\partial u_1} &= u_1 - \lambda_{\omega_x} I_{sc_{11}}^{-1} Z_{11} - \lambda_{\omega_y} I_{sc_{22}}^{-1} Z_{21} - \lambda_{\omega_z} I_{sc_{33}}^{-1} Z_{31} + \lambda_{\Omega_1} I_{w_{11}}^{-1} + \mu_1 = 0 \\ \frac{\partial \bar{H}}{\partial u_2} &= u_2 - \lambda_{\omega_x} I_{sc_{11}}^{-1} Z_{12} - \lambda_{\omega_y} I_{sc_{22}}^{-1} Z_{22} - \lambda_{\omega_z} I_{sc_{33}}^{-1} Z_{32} + \lambda_{\Omega_2} I_{w_{11}}^{-1} + \mu_2 = 0 \\ \frac{\partial \bar{H}}{\partial u_3} &= u_3 - \lambda_{\omega_x} I_{sc_{11}}^{-1} Z_{13} - \lambda_{\omega_y} I_{sc_{22}}^{-1} Z_{23} - \lambda_{\omega_z} I_{sc_{33}}^{-1} Z_{33} + \lambda_{\Omega_3} I_{w_{11}}^{-1} + \mu_3 = 0 \\ \frac{\partial \bar{H}}{\partial u_4} &= u_4 - \lambda_{\omega_x} I_{sc_{11}}^{-1} Z_{14} - \lambda_{\omega_y} I_{sc_{22}}^{-1} Z_{24} - \lambda_{\omega_z} I_{sc_{33}}^{-1} Z_{34} + \lambda_{\Omega_4} I_{w_{11}}^{-1} + \mu_4 = 0 \end{aligned} \tag{6.9}$$

The complementarity condition shows a testable relationship between the control constraint and its μ covector. If the control is at the upper limit, the covector should be positive, and if the control is at the lower limit, the covector should be negative. If the control is anywhere in between the limits, then the constraint is not active and the covector should be zero. Graphing the two together should show whether this relationship holds.

Evaluating the partial derivative of the Endpoint Lagrangian (Eqn. 6.10) with respect to the initial and final times identifies the expected initial and final value of the Hamiltonian.

$$\bar{E}(\boldsymbol{\nu}, \mathbf{x}(t_f)) := E(\mathbf{x}(t_f)) + \nu^T e(\mathbf{x}(t_f)) \tag{6.10}$$

$$\begin{aligned} \bar{E}(\boldsymbol{\nu}, \boldsymbol{x_f}, t_f) = \nu_e(e_{xor}) + \nu_{\omega_x}\omega_{x_f} + \nu_{\omega_y}\omega_{y_f} + \nu_{\omega_z}\omega_{z_f} \ldots \\ + \nu_{\Omega_1}\Omega_{1_f} + \nu_{\Omega_2}\Omega_{2_f} + \nu_{\Omega_3}\Omega_{3_f} + \nu_{\Omega_4}\Omega_{4_f} + \nu_{t_f}(t_f - t^f) \end{aligned} \tag{6.11}$$

$$\frac{-\partial \bar{E}}{\partial t_f} = H(t_f) = -\nu_{t_f}$$

Taking the partial derivative of the Hamiltonian with respect to time:

$$\frac{d\mathcal{H}}{dt} = \frac{\partial H}{\partial t} = 0$$

These last two steps show that the Hamiltonian should be equal to some arbitrary constant over the entire trajectory.

Transversality analysis is done to determine the final values of the individual costates, which are equal to the derivative of the Endpoint Lagrangian with respect to the final state.

$$\lambda_{q_i}(t_f) = \frac{-\nu_e}{\sigma^2}\left(\begin{array}{l} [q^f(i) - q_{f_1}(i)]\, e^{\overbrace{\frac{-[(q^f(1)-q_{f_1}(1))^2+(q^f(2)-q_{f_1}(2))^2+(q^f(3)-q_{f_1}(3))^2+(q^f(4)-q_{f_1}(4))^2]}{2\sigma^2}}^{A}} \\ +[q^f(i) - q_{f_2}(i)]\, e^{\overbrace{\frac{-[(q^f(1)-q_{f_2}(1))^2+(q^f(2)-q_{f_2}(2))^2+(q^f(3)-q_{f_2}(3))^2+(q^f(4)-q_{f_2}(4))^2]}{2\sigma^2}}^{B}} \end{array}\right)$$

For brevity, each quaternion costate equation is expressed in terms of A and B, since for any combination of final quaternions, A and B are some constant value:

$$\lambda_{q_i}(t_f) = -\frac{\nu_e}{\sigma^2}\left(A[q^f(i) - q_{f_1}(i)] + B[q^f(i) - q_{f_2}(i)]\right) \tag{6.12}$$

Rearranging (6.12) for the ν_e coefficient terms:

$$\frac{-\nu_e}{\sigma^2} = \frac{\lambda_{q_i}(t_f)}{A[q^f(i) - q_{f_1}(i)] + B[q^f(i) - q_{f_2}(i)]}$$

After substituting and rearranging, six new boundary conditions are developed:

$$\left.\begin{array}{l} \lambda_{q_i}(t_f) \cdot \overbrace{\left(A[q^f(i) - q_{f_1}(i)] + B[q^f(i) - q_{f_2}(i)]\right)}^{C} \\ = \lambda_{q_j}(t_f) \cdot \overbrace{\left(A[q^f(j) - q_{f_1}(j)] + B[q^f(j) - q_{f_2}(j)]\right)}^{D} \end{array}\right\} \quad \forall\, i, j = [1,2,3,4],\ i \neq j \tag{6.13}$$

Transversality analysis for the velocity costates is straightforward.

$$\lambda_\omega(t_f) = \frac{\partial \bar{E}}{\partial \omega^f} = \nu_\omega$$

$$\lambda_\Omega(t_f) = \frac{\partial \bar{E}}{\partial \Omega^f} = \nu_\Omega$$

The final velocity costate values are equal to some arbitrary constant which provides no useful information.

The BVP and the necessary conditions for solution optimality are now fully developed and the problem can be coded into an algorithmic solver.

$$OCP: \begin{cases} Minimize: & J[\mathbf{x}(\cdot), \mathbf{u}(\cdot), t_f] & = & \frac{1}{2}\int_0^{t_f} (\sum_{n=1}^{4} {u_n}^2)\, dt \\ Subject\ to: & \dot{\mathbf{q}} & = & \mathbf{Q}(\omega)\mathbf{q} \\ & \dot{\boldsymbol{\omega}} & = & -\mathbf{I}_{sc}^{-1}[\boldsymbol{\omega} \times (\mathbf{I}_{sc}\boldsymbol{\omega} + \mathbf{h}) - \boldsymbol{\tau}_{sc}] \\ & \dot{\boldsymbol{\Omega}} & = & \mathbf{I}_w^{-1}\mathbf{u} \\ & \boldsymbol{q}^o & = & [0,0,0,1]^T \\ & e_{xor}(\boldsymbol{q}^f) & = & 0 \\ & \boldsymbol{\omega}_o, \boldsymbol{\omega}_f & = & (0,0,0), (0,0,0) \\ & \boldsymbol{\Omega}_o, \boldsymbol{\Omega}_f & = & (0,0,0,0), (0,0,0,0) \end{cases}$$

6.2.1 Minimum Effort Results

Once the minimum effort problem was fully defined and coded in an algorithmic solver, two different scenarios were presented and analyzed. Each minimum effort scenario was bounded to make the rotation in 300 seconds. Before using the algorithm to solve each XOR problem, the individual rotations were solved to identify baselines for cost, control trajectory and solution accuracy. This was done using the same dynamics model but with the standard $2N_x$ end-point event conditions:

$$
\begin{aligned}
\boldsymbol{q}^o, \boldsymbol{q}^f &= (0,0,0,1), (\boldsymbol{q}_f(1), \boldsymbol{q}_f(2), \boldsymbol{q}_f(3), \boldsymbol{q}_f(4)) \\
\boldsymbol{\omega}_o, \boldsymbol{\omega}_f &= (0,0,0), (0,0,0) \\
\boldsymbol{\Omega}_o, \boldsymbol{\Omega}_f &= (0,0,0,0), (0,0,0,0)
\end{aligned}
$$

Scenario 1: Angle Discrimination

This scenario tested the algorithm's ability to discriminate between two different magnitude rotations independent of any inertia tensor bias. It involves two end-point quaternions representing different magnitude rotations about different axes. To simplify the analysis, a spherically symmetric spacecraft body is assumed using a scalar inertia tensor matrix.

$$
q_f = \begin{bmatrix} sin(100/2) \\ 0 \\ 0 \\ cos(100/2) \end{bmatrix} \oplus \begin{bmatrix} 0 \\ sin(40/2) \\ 0 \\ cos(40/2) \end{bmatrix} \quad and \quad \mathbf{I}_{sc} = \begin{bmatrix} 800 & 0 & 0 \\ 0 & 800 & 0 \\ 0 & 0 & 800 \end{bmatrix} kg \cdot m^2
$$

The first quaternion represents a 100° rotation about the roll axis. Initial orientation and commanded rotation trajectories are shown in figures 6.4 and 6.5. Note that the control trajectory commands a pure rotation about the roll axis with a cost of 0.1628 kg^2m^2s. In this analysis, the kg^2m^2s have no actual context but are used as a relative measure.

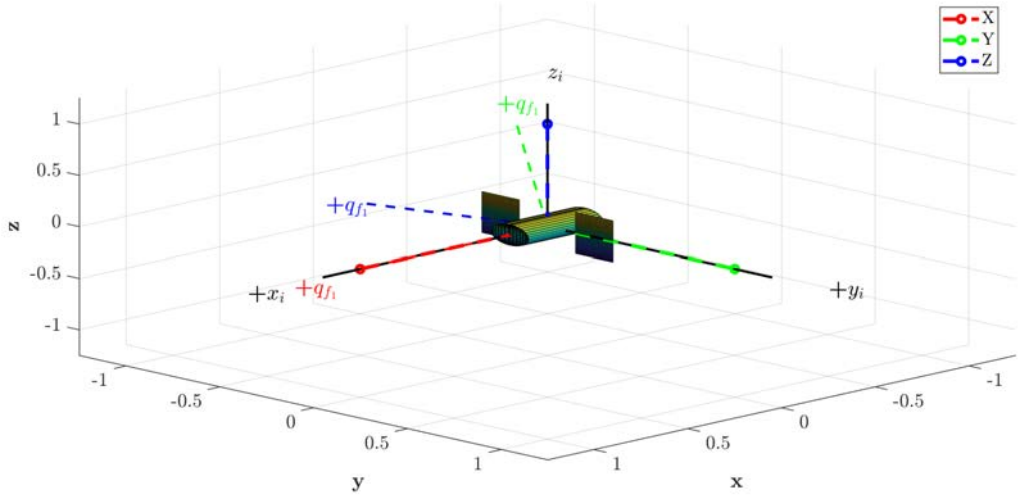


Figure 6.4: Quaternion minimum effort angle discrimination, initial orientation for q_1 rotation

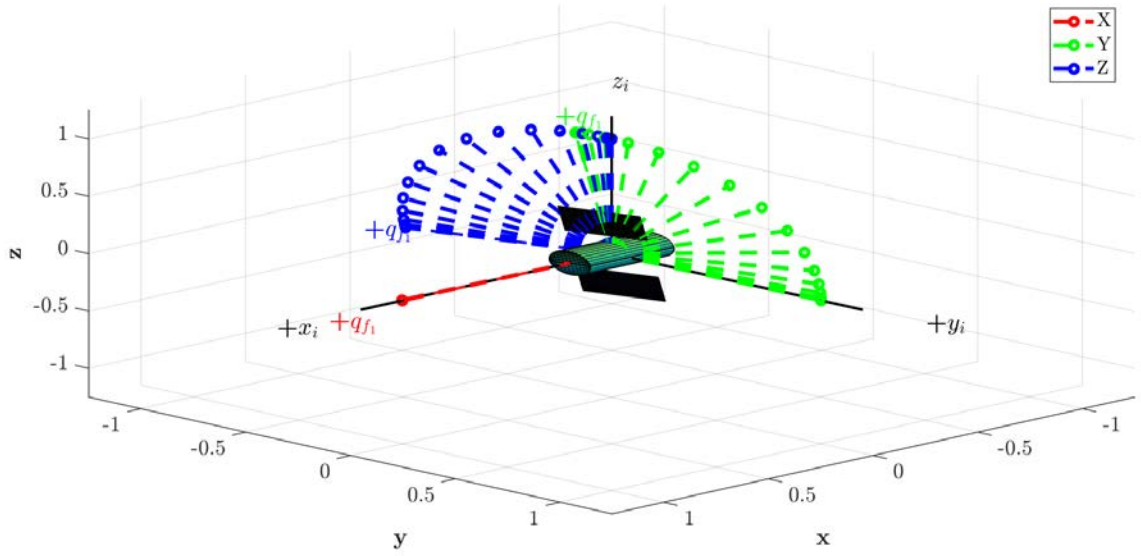


Figure 6.5: Quaternion minimum effort angle discrimination, simple rotation to q_1

The second quaternion option represents a 40° rotation about the pitch axis. Initial orientation and commanded rotation trajectories are again shown in Figures 6.6 and 6.7. Again the optimal control trajectory commands a pure rotation about the pitch axis, this time at a much lower control cost of 0.1055 kg^2m^2s.

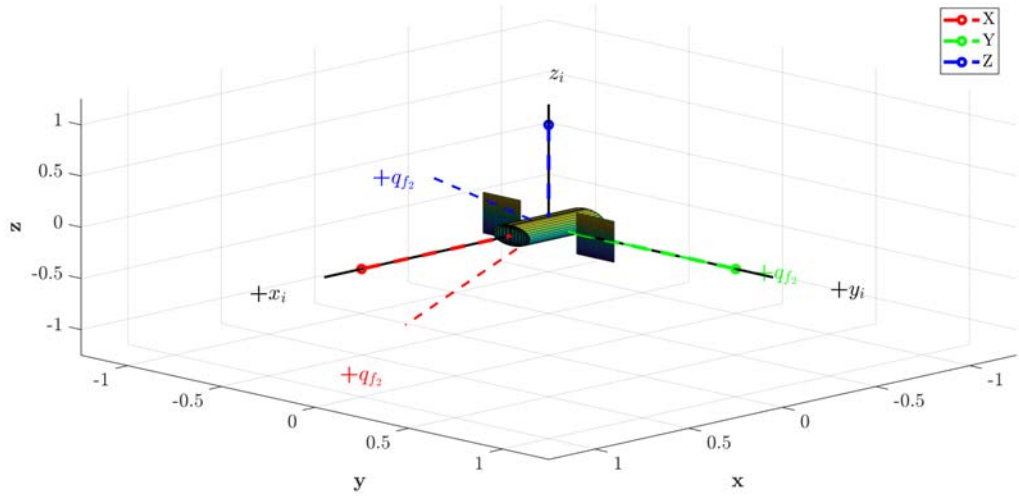


Figure 6.6: Quaternion minimum effort angle discrimination, initial orientation for q_2 rotation

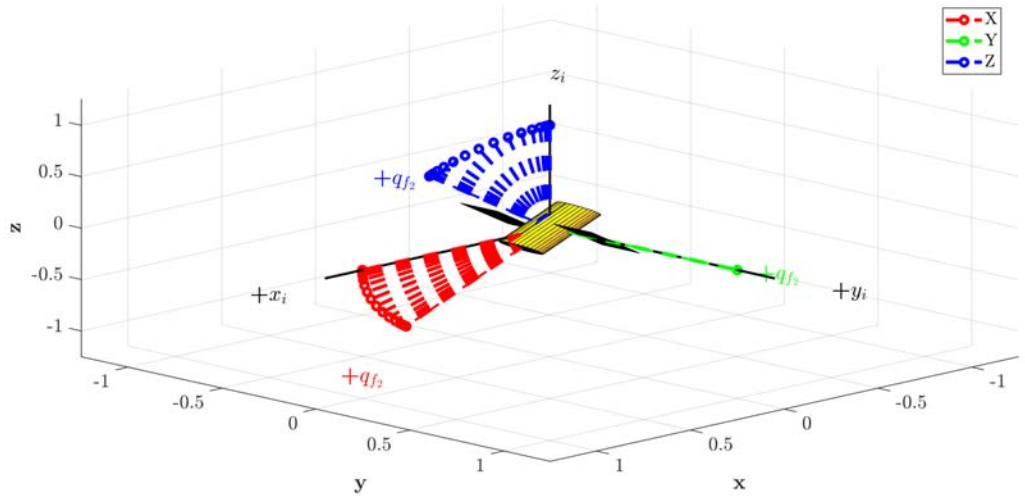


Figure 6.7: Quaternion minimum effort angle discrimination, simple rotation to q_2

Once these rotations were calculated, they were analyzed for optimality. Single end-point optimality analysis has been demonstrated in previous works (see [24]). That analysis is omitted here and instead a complete analysis of the XOR end-point case is presented.

The minimum effort, angle discrimination, XOR problem was then solved. Figure 6.8 shows the initial spacecraft orientation with both possible end-point quaternions (q_1 and q_2). Figure 6.9 shows the resulting rotation. As desired, the algorithm correctly chose the minimum effort rotation (to q_2) at a very similar control cost of 0.1054 kg^2m^2s.

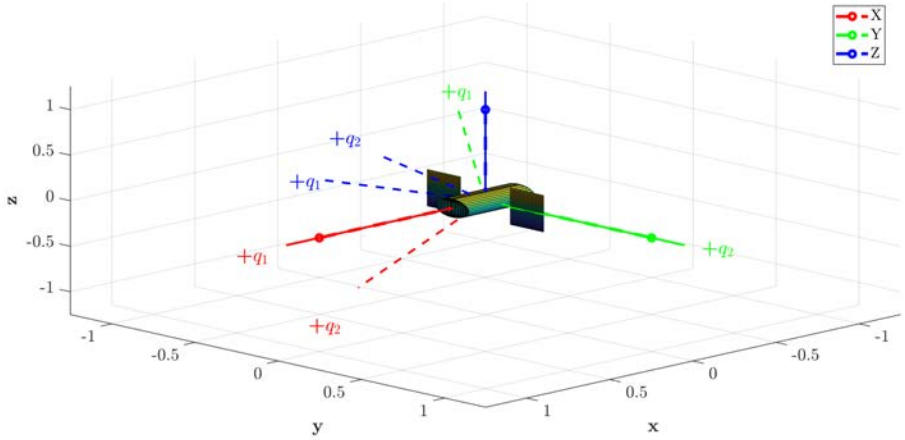


Figure 6.8: Quaternion minimum effort angle discrimination, XOR rotation initial orientation

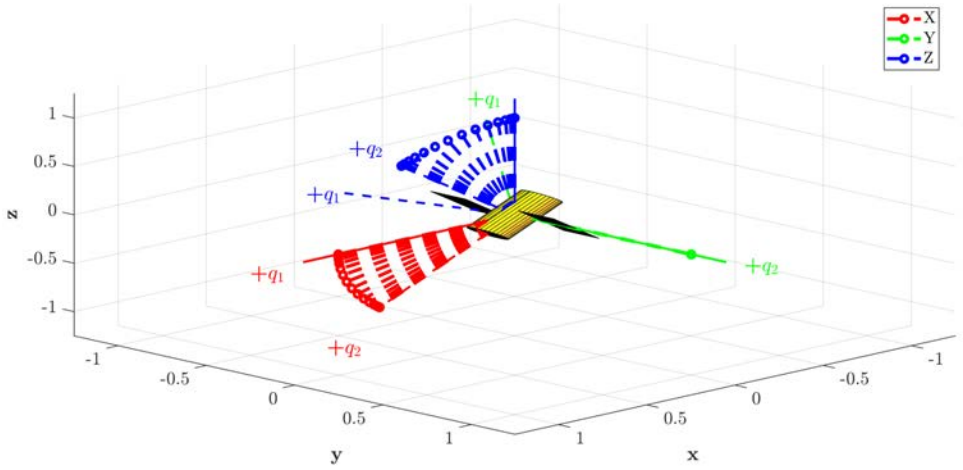
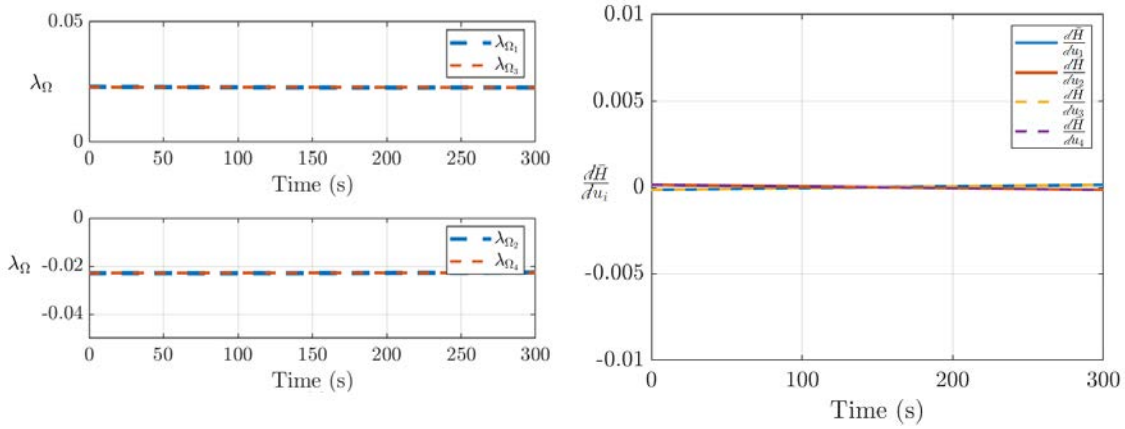


Figure 6.9: Quaternion minimum effort angle discrimination, XOR rotation

It is interesting to note that there is a small difference in control cost between the two solutions. This is due to the difference in final end-point error and is discussed later.

Optimality Validation

Figure 6.10 shows that the adjoint necessary conditions are met as each λ_ω is constant ($\dot{\lambda}_\omega = 0$). The derivative of the Lagrangian of the Hamiltonian is approximately constant with respect to each control, validating the stationarity condition (Figure 6.10b).



(a) Costates (b) Stationarity condition

Figure 6.10: Quaternion minimum effort angle discrimination, necessary conditions

The complementarity conditions (Figure 6.11), while met, are uninteresting as the constraint never activates. This is clear because the control costates (μ_τ) never diverge from zero.

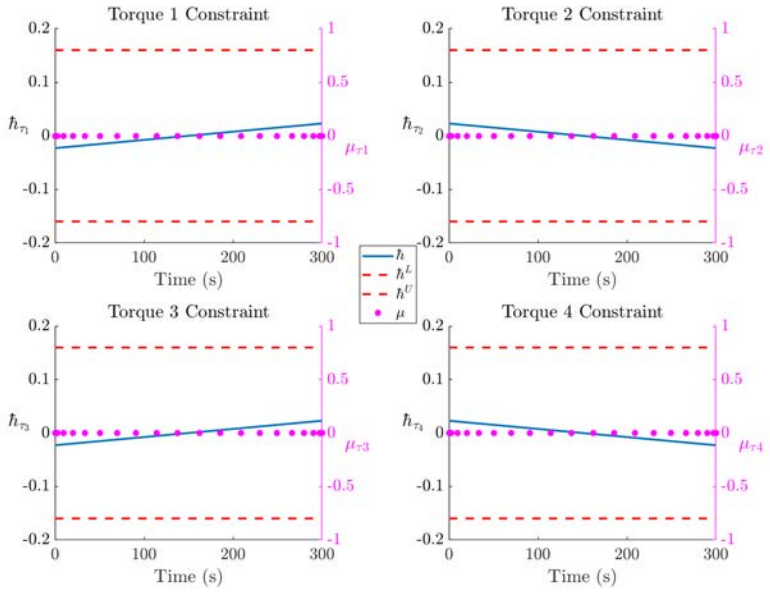


Figure 6.11: Quaternion minimum effort angle discrimination, complementarity condition

Figure 6.12 verifies that the Hamiltonian is constant throughout the trajectory.

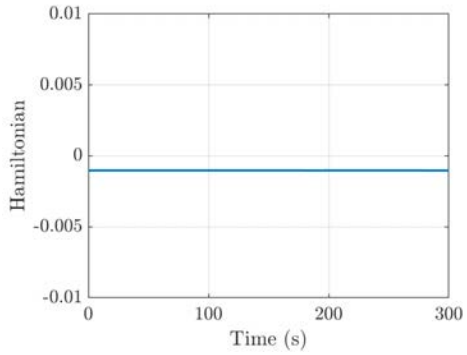


Figure 6.12: Quaternion minimum effort angle discrimination, Hamiltonian

Lastly, the transversality conditions were verified by calculating the four versions of equation 6.13. In each case the difference was very close to zero, so all necessary conditions for this to be an optimal solution are met.

$$
\begin{aligned}
\lambda_{q_1}(t_f) \cdot C - \lambda_{q_2}(t_f) \cdot D &= 1.3059E^{-12} \\
\lambda_{q_2}(t_f) \cdot C - \lambda_{q_3}(t_f) \cdot D &= -1.2910E^{-12} \\
\lambda_{q_3}(t_f) \cdot C - \lambda_{q_4}(t_f) \cdot D &= 3.1584E^{-12}
\end{aligned}
$$

Scenario 2: Inertia Tensor Discrimination

The second scenario tested the algorithm's ability to discriminate between rotations of the same magnitude accounting for a non-symmetric inertia tensor. This case involves two possible end-point quaternions representing 40° rotations about different axes. A simplified, non-symmetric spacecraft body is assumed using a diagonal inertia tensor matrix.

$$
q_f = \begin{bmatrix} sin(40/2) \\ 0 \\ 0 \\ cos(40/2) \end{bmatrix} \oplus \begin{bmatrix} 0 \\ sin(40/2) \\ 0 \\ cos(40/2) \end{bmatrix} \quad and \quad \mathbf{I}_{sc} = \begin{bmatrix} 712.5 & 0 & 0 \\ 0 & 949.5 & 0 \\ 0 & 0 & 871.5 \end{bmatrix} kg \cdot m^2
$$

The first quaternion option now represents a 40° rotation about the roll axis. The commanded rotation trajectories are shown in figure 6.13. Note that the calculated control trajectory, as expected, commands a pure rotation about the roll axis with a cost of 0.0211 kg^2m^2s.

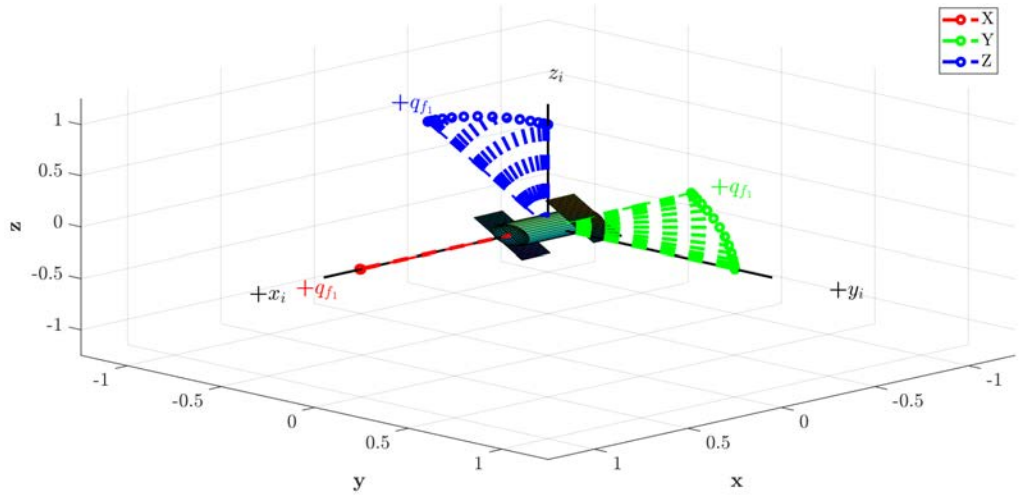


Figure 6.13: Quaternion minimum effort inertia discrimination, simple rotation to q_1

As in Scenario 1, the second quaternion option represents a 40° rotation about the pitch

axis. The commanded rotation trajectories are again shown in figure 6.14. Again the optimal control trajectory commands a pure rotation about the pitch axis, this time at a much higher control cost of 0.1496 kg^2m^2s.

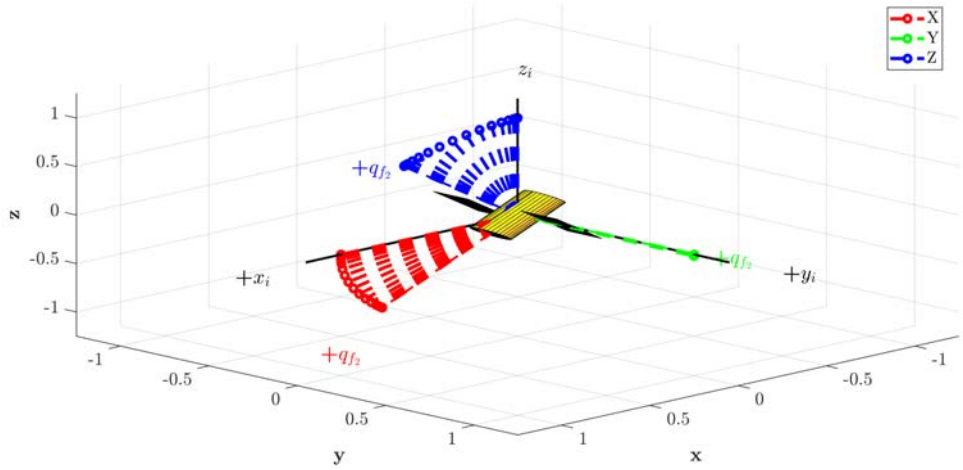


Figure 6.14: Quaternion minimum effort inertia discrimination, simple rotation to q_2

The minimum effort, inertia tensor discrimination, XOR problem was then solved. The algorithm correctly chose the minimum effort rotation again at a very similar cost to the single end-point rotation problem of 0.0205 kg^2m^2s.

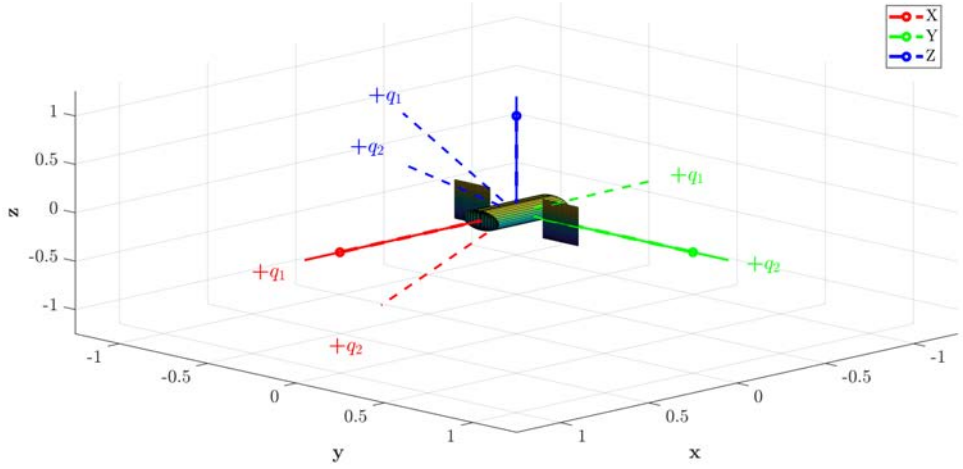


Figure 6.15: Quaternion minimum effort inertia discrimination, XOR rotation initial orientation

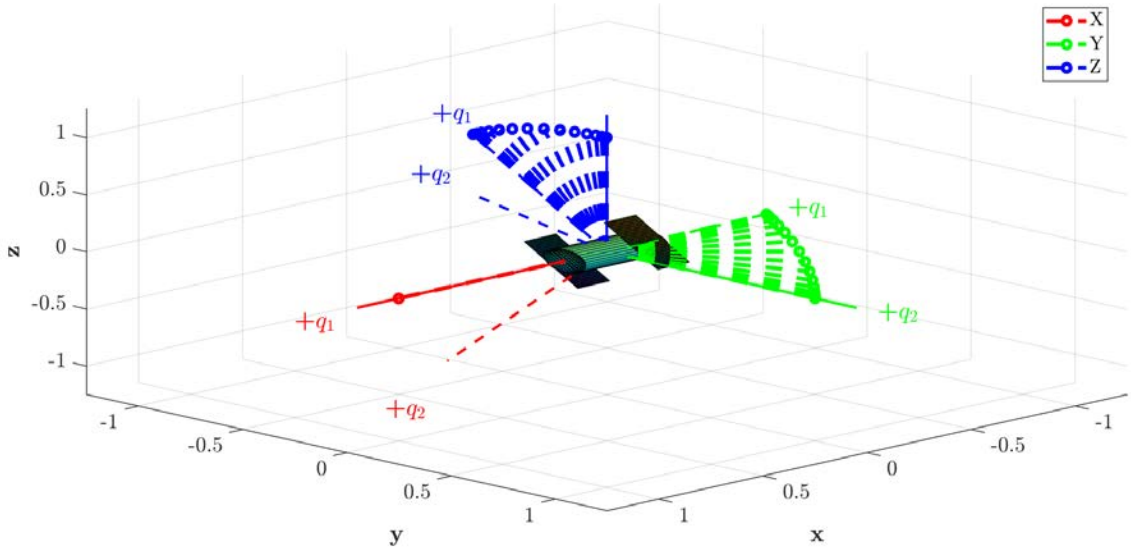
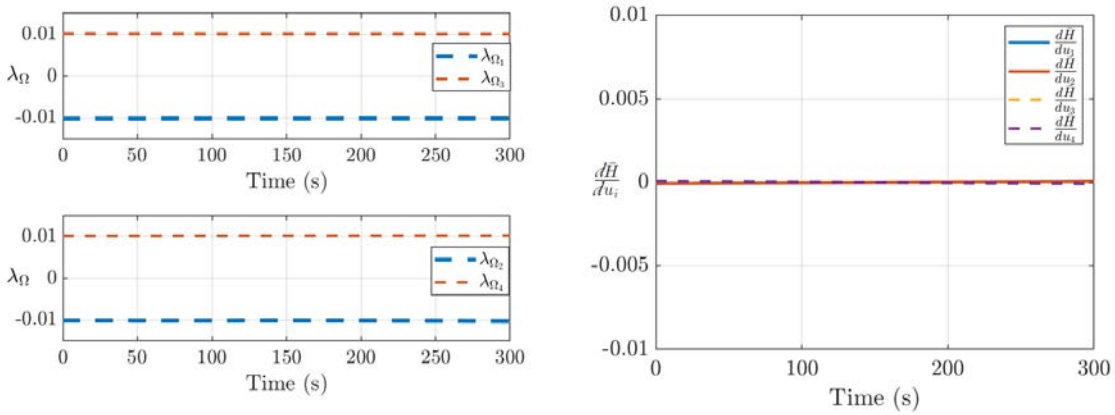


Figure 6.16: Quaternion minimum effort inertia tensor discrimination, XOR rotation

Optimality Validation

Figures 6.17 and 6.18 show that the adjoint necessary conditions, stationarity, and constancy of the Hamiltonian are all met. The torque constraints remain inactive so the complementarity conditions are again uninteresting.



(a) Costates

(b) Stationarity condition

Figure 6.17: Quaternion minimum effort inertia tensor discrimination, necessary Conditions

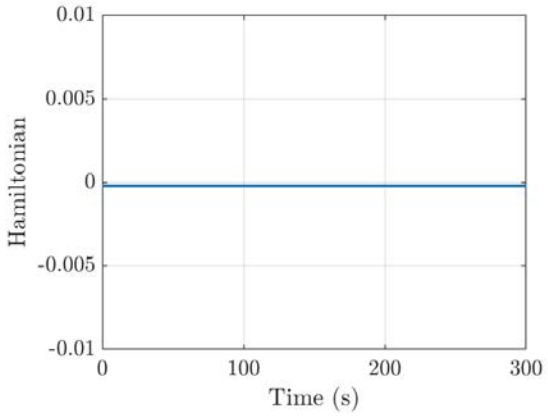


Figure 6.18: Quaternion minimum effort inertia tensor discrimination, Hamiltonian

The transversality conditions were again verified to be very close to zero.

$$\begin{aligned}\lambda_{q_1}(t_f)\cdot C - \lambda_{q_2}(t_f)\cdot D &= 5.3693E^{-08}\\ \lambda_{q_2}(t_f)\cdot C - \lambda_{q_3}(t_f)\cdot D &= 2.9692E^{-08}\\ \lambda_{q_3}(t_f)\cdot C - \lambda_{q_4}(t_f)\cdot D &= -4.9017E^{-08}\end{aligned}$$

Again all necessary conditions for this to be an optimal solution are met.

6.3 The Minimum Time Problem

This section presents the analysis of the same XOR problem, now for a minimum time cost functional [2]

$$J[\mathbf{x}(\cdot),\mathbf{u}(\cdot),t_f] = E(x_f) + \int_{t_o}^{t_f} F(x,u)dt = t_f \tag{6.14}$$

with the same end-point constraints:

$$\boldsymbol{\omega}_o = \boldsymbol{\omega}_f = (0,0,0), \qquad \boldsymbol{\Omega}_o = \boldsymbol{\Omega}_f = (0,0,0,0), \qquad \boldsymbol{q}_o = (0,0,0,1)$$

$$e_{xor}(\boldsymbol{q}^f) = e^{\frac{-[(q^f(1)-q_{f_1}(1))^2+(q^f(2)-q_{f_1}(2))^2+(q^f(3)-q_{f_1}(3))^2+(q^f(4)-q_{f_1}(4))^2]}{2\sigma^2}} \ldots + e^{\frac{-[(q^f(1)-q_{f_2}(1))^2+(q^f(2)-q_{f_2}(2))^2+(q^f(3)-q_{f_2}(3))^2+(q^f(4)-q_{f_2}(4))^2]}{2\sigma^2}} - 1 = 0$$

Using equation 6.5, noting the loss of the u^2 terms, the Lagrangian of the Hamiltonian is now linear in control [2], thus we expect a band-bang type control.

$$\begin{aligned}\bar{H}(\boldsymbol{\mu},\boldsymbol{\lambda},\boldsymbol{x},\boldsymbol{u},t) = \quad & \tfrac{1}{2}[\lambda_{q_1}(q_4\omega_x - q_3\omega_y + q_2\omega_z) + \lambda_{q_2}(q_3\omega_x + q_4\omega_y - q_1\omega_z)]... \\ & +\tfrac{1}{2}[\lambda_{q_3}(-q_2\omega_x + q_1\omega_y + q_4\omega_z) + \lambda_{q_4}(-q_1\omega_x - q_2\omega_y - q_3\omega_z)]... \\ & -\lambda_{\omega_x} I_{sc_{11}}^{-1}[Z_{11}u_1 + Z_{12}u_2 + Z_{13}u_3 + Z_{14}u_4]... \\ & -\lambda_{\omega_y} I_{sc_{22}}^{-1}[Z_{21}u_1 + Z_{22}u_3 + Z_{23}u_3 + Z_{24}u_4]... \\ & -\lambda_{\omega_z} I_{sc_{33}}^{-1}[Z_{31}u_1 + Z_{32}u_2 + Z_{33}u_3 + Z_{34}u_4]... \\ & \lambda_{\Omega_1} I_{w_{11}}^{-1}u_1 + \lambda_{\Omega_2} I_{w_{22}}^{-1}u_2 + \lambda_{\Omega_3} I_{w_{33}}^{-1}u_3 + \lambda_{\Omega_4} I_{w_{44}}^{-1}u_4... \\ & +\mu_1u_1 + \mu_2u_2 + \mu_3u_3 + \mu_4u_4\end{aligned} \tag{6.15}$$

Evaluating the adjoint conditions results in the same testable conditions as before. Likewise, the same KKT conditions (stationarity Eq. 6.7 and complementarity 6.8) must be met [2].

The stationarity condition equations no longer contain the control term (u).

$$\begin{aligned}\frac{\partial \bar{H}}{\partial u_1} &= -\lambda_{\omega_x} I_{sc_{11}}^{-1} Z_{11} - \lambda_{\omega_y} I_{sc_{22}}^{-1} Z_{21} - \lambda_{\omega_z} I_{sc_{33}}^{-1} Z_{31} + \lambda_{\Omega_1} I_{w_{11}}^{-1} + \mu_1 = 0 \\ \frac{\partial \bar{H}}{\partial u_2} &= -\lambda_{\omega_x} I_{sc_{11}}^{-1} Z_{12} - \lambda_{\omega_y} I_{sc_{22}}^{-1} Z_{22} - \lambda_{\omega_z} I_{sc_{33}}^{-1} Z_{32} + \lambda_{\Omega_2} I_{w_{11}}^{-1} + \mu_2 = 0 \\ \frac{\partial \bar{H}}{\partial u_3} &= -\lambda_{\omega_x} I_{sc_{11}}^{-1} Z_{13} - \lambda_{\omega_y} I_{sc_{22}}^{-1} Z_{23} - \lambda_{\omega_z} I_{sc_{33}}^{-1} Z_{33} + \lambda_{\Omega_3} I_{w_{11}}^{-1} + \mu_3 = 0 \\ \frac{\partial \bar{H}}{\partial u_4} &= -\lambda_{\omega_x} I_{sc_{11}}^{-1} Z_{14} - \lambda_{\omega_y} I_{sc_{22}}^{-1} Z_{24} - \lambda_{\omega_z} I_{sc_{33}}^{-1} Z_{34} + \lambda_{\Omega_4} I_{w_{11}}^{-1} + \mu_4 = 0\end{aligned} \tag{6.16}$$

The complementarity condition plots should still show the relationship between the control constraint and its μ covector. Unlike in the previous case, for the minimum time problem the constraints should become active.

The Endpoint Lagrangian now includes a t_f term.

$$\bar{E}(\boldsymbol{\nu}, \mathbf{x}(t_f)) := E(\mathbf{x}(t_f)) + \nu^T e(\mathbf{x}(t_f)) \tag{6.17}$$

$$\begin{aligned}\bar{E}(\boldsymbol{\nu},\boldsymbol{x_f},t_f) = t_f + \nu_e(e_{xor}) + \nu_{\omega_x}\omega_{x_f} + \nu_{\omega_y}\omega_{y_f} + \nu_{\omega_z}\omega_{z_f}... \\ +\nu_{\Omega_1}\Omega_{1_f} + \nu_{\Omega_2}\Omega_{2_f} + \nu_{\Omega_3}\Omega_{3_f} + \nu_{\Omega_4}\Omega_{4_f}\end{aligned} \tag{6.18}$$

$$\frac{-\partial \bar{E}}{\partial t_f} = H(t_f) = -1; \quad and \quad \frac{d\mathcal{H}}{dt} = \frac{\partial H}{\partial t} = 0$$

These last two equations now show that the Hamiltonian should be equal to negative one and constant over the entire trajectory. Transversality analysis is unchanged with the same quaternion costate behaviors (6.13).

The OCP and the necessary conditions for solution optimality are again fully developed and the problem can be coded into an algorithmic solver.

$$
OCP: \begin{cases}
Minimize: & J[\mathbf{x}(\cdot), \mathbf{u}(\cdot), t_f] &=& t_f \\
Subject\ to: & \dot{\mathbf{q}} &=& \mathbf{Q}(\omega)\mathbf{q} \\
& \dot{\boldsymbol{\omega}} &=& -\mathbf{I}_{sc}^{-1}[\boldsymbol{\omega} \times (\mathbf{I}_{sc}\boldsymbol{\omega} + \mathbf{h}) - \boldsymbol{\tau}_{sc}] \\
& \dot{\boldsymbol{\Omega}} &=& \mathbf{I}_w^{-1}\mathbf{u} \\
& \boldsymbol{q}^o &=& [0,0,0,1]^T \\
& e_{xor}(\boldsymbol{q}^f) &=& 0 \\
& \boldsymbol{\omega}_o, \boldsymbol{\omega}_f &=& (0,0,0), (0,0,0) \\
& \boldsymbol{\Omega}_o, \boldsymbol{\Omega}_f &=& (0,0,0,0), (0,0,0,0)
\end{cases}
$$

6.3.1 Minimum Time Results

This scenario tested the algorithm's ability to solve for a minimum time rotation by choosing between different magnitude rotations. This test again involves two end-point quaternions representing different magnitude rotations about different axes.

$$
q_f = \begin{bmatrix} sin(120/2) \\ 0 \\ 0 \\ cos(120/2) \end{bmatrix} \oplus \begin{bmatrix} 0 \\ 0 \\ sin(30/2) \\ cos(30/2) \end{bmatrix} \quad and \quad \mathbf{I}_{sc} = \begin{bmatrix} 712.5 & 0 & 0 \\ 0 & 949.5 & 0 \\ 0 & 0 & 871.5 \end{bmatrix} kg \cdot m^2
$$

The first quaternion represents a 120° rotation about the roll axis. Initial orientation and commanded rotations are shown in figures 6.19 and 6.20. Note that the control trajectory commands a pure rotation about the roll axis with a rotation time of 107.40 seconds.

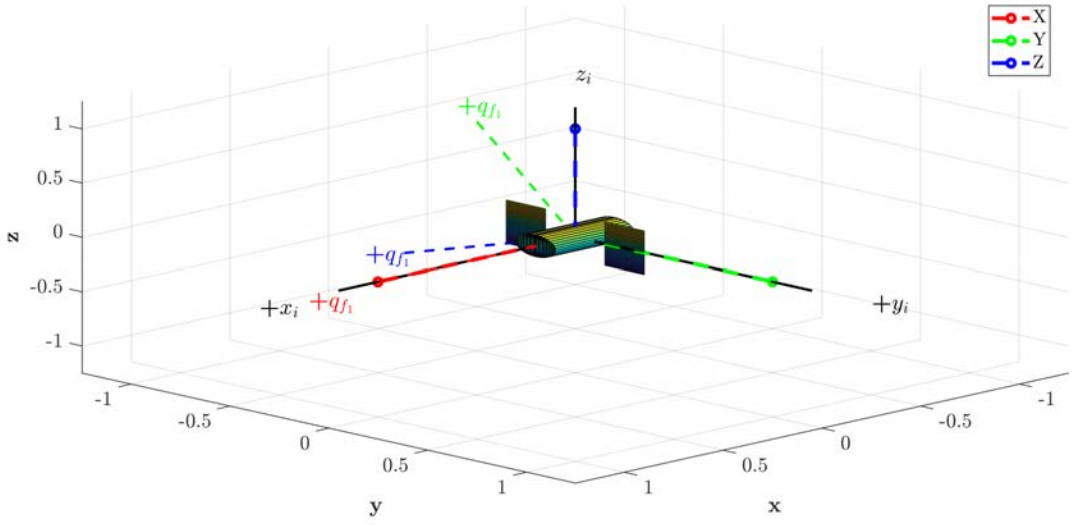


Figure 6.19: Quaternion minimum time, initial orientation for q_1 rotation

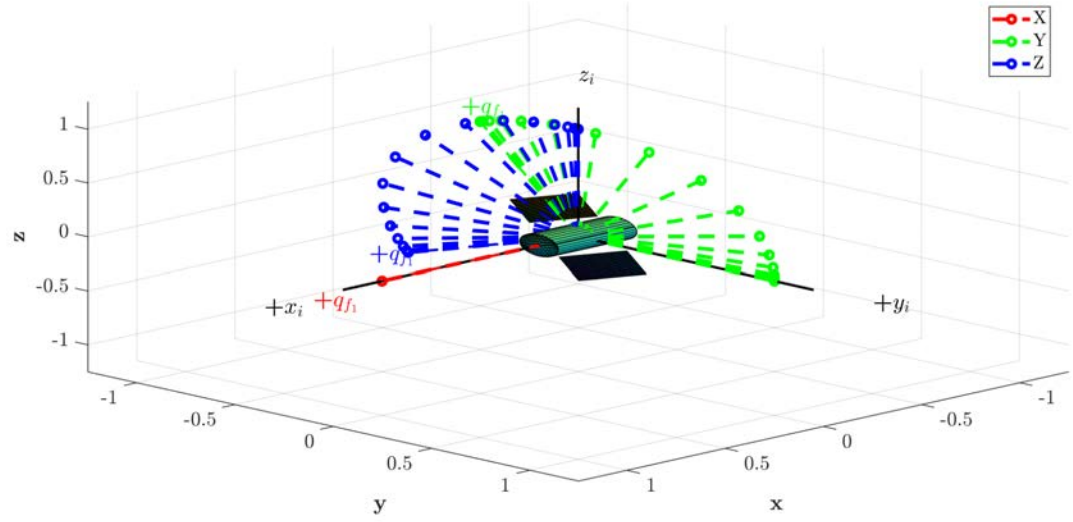


Figure 6.20: Quaternion minimum time, simple rotation to q_1

The second quaternion represents a 30° rotation about the yaw axis. Initial orientation and commanded rotation trajectories are again shown in figures 6.21 and 6.22. Again the optimal control trajectory commands a pure rotation about the pitch axis with a rotation time of 84.37 seconds.

As in the previous, minimum effort case, the baseline control trajectories and rotations were

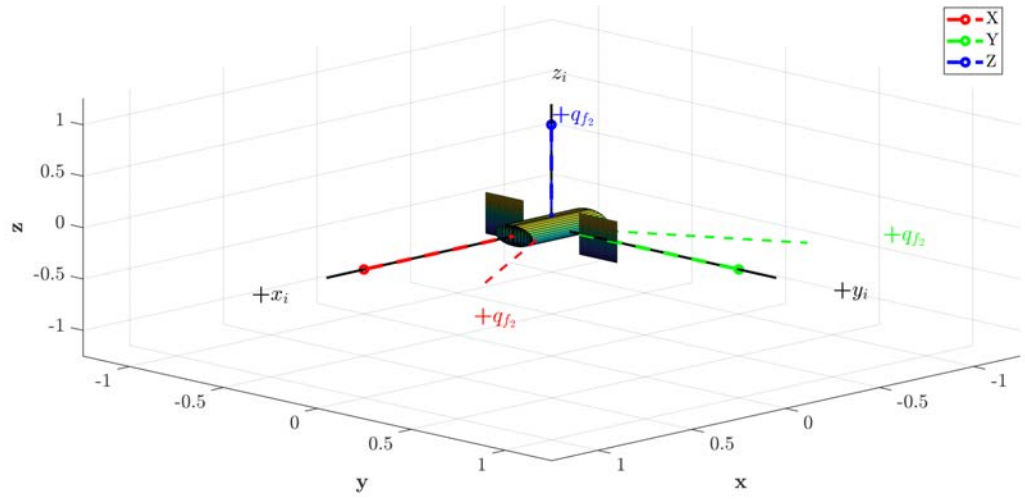


Figure 6.21: Quaternion minimum time, initial orientation for q_2 rotation

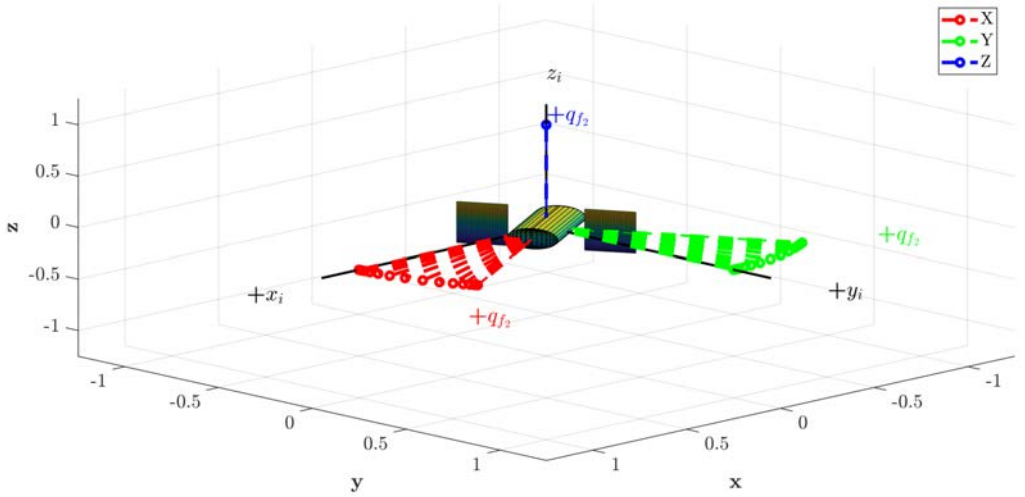


Figure 6.22: Quaternion minimum time, simple rotation to q_2

calculated and then verified to meet all necessary conditions for optimality. That analysis is again omitted in favor of a complete optimality analysis of the XOR problem.

The XOR problem was then solved. Figure 6.23 shows the initial spacecraft orientation with both possible end-point quaternions (q_1 and q_2). Figure 6.24 shows the resulting rotation. As desired, the algorithm correctly chose the minimum time rotation (to q_2) at a very similar total rotation time of 84.30 seconds.

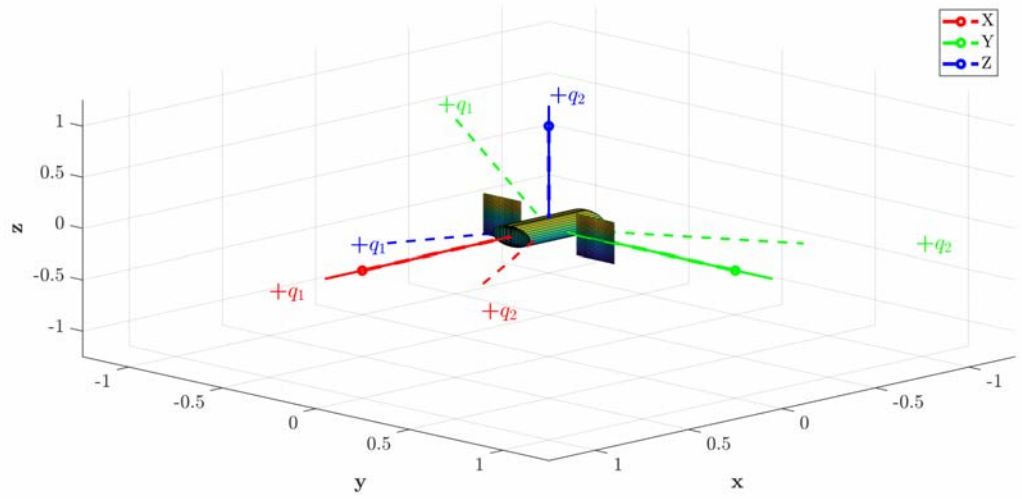


Figure 6.23: Quaternion minimum time, initial orientation for the XOR rotation using non-symmetric inertia tensor

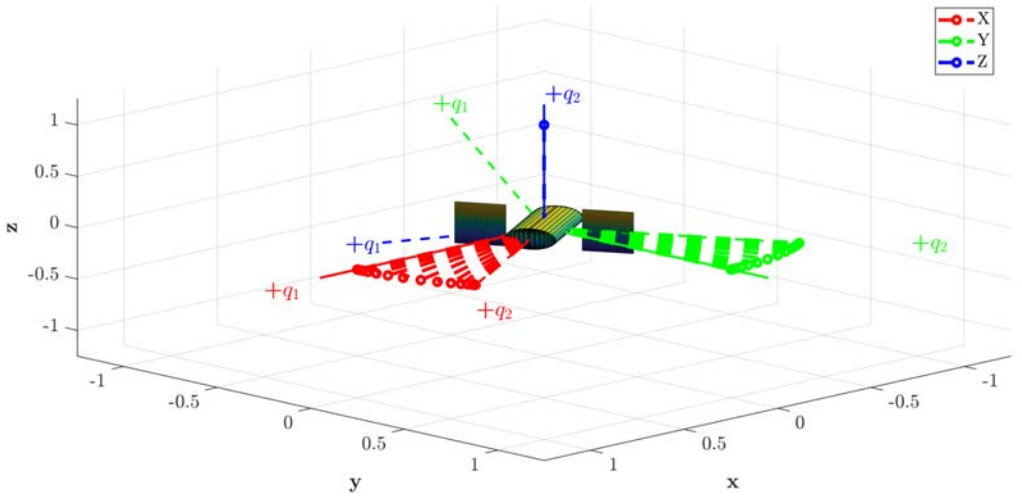
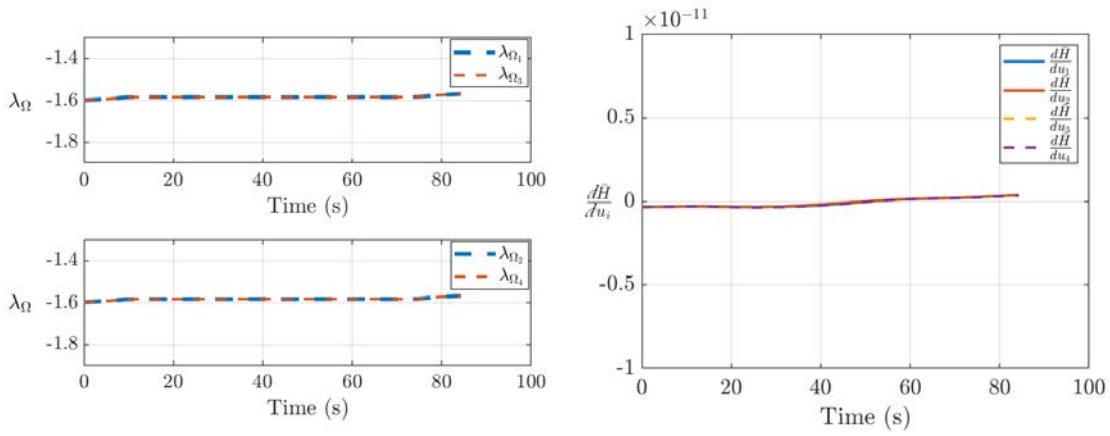


Figure 6.24: Quaternion minimum time, XOR rotation using non-symmetric inertia tensor

Optimality Validation

Figure 6.25 shows that the adjoint necessary conditions are met (each λ_Ω is approximately constant, $\dot{\lambda}_\Omega = 0$), and that the derivative of the Lagrangian of the Hamiltonian is approximately constant with respect to each control, validating the stationarity condition.

This complementarity condition says that when path constraint ($\hbar$) is at the upper constraint, the covector must be positive or zero. Inversely, if $\hbar$ is at the lower constraint the covector



(a) Costates (b) Stationarity condition

Figure 6.25: Quaternion minimum time, necessary conditions

must be negative or zero. If $\hbar$ is in between, the constraint is inactive and the covector should be zero. In each constraint case, shown in Figure 6.26, complementarity is met as expected for the minimum time bang-bang type control.

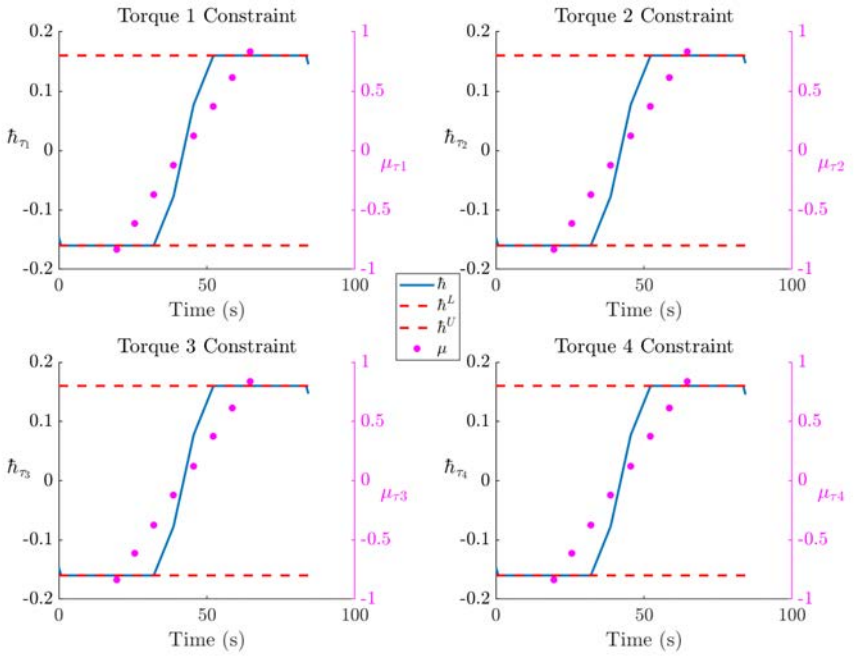


Figure 6.26: Quaternion minimum time, complementarity condition validation

Figure 6.27 verifies that the Hamiltonian is constant and approximately negative one throughout the trajectory, with the slight bump in the middle as commonly observed in numerical solutions to minimum time OCP.

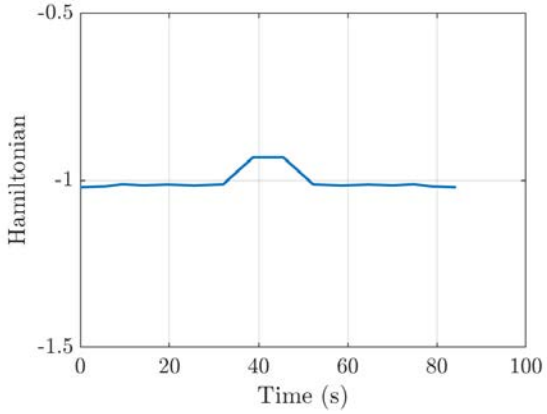


Figure 6.27: Quaternion minimum time, Hamiltonian

Lastly, the transversality conditions were verified to again be close to zero.

$$\begin{aligned} \lambda_{q_1}(t_f) \cdot C - \lambda_{q_2}(t_f) \cdot D &= -2.8820E^{-12} \\ \lambda_{q_2}(t_f) \cdot C - \lambda_{q_3}(t_f) \cdot D &= -0.0541 \\ \lambda_{q_3}(t_f) \cdot C - \lambda_{q_4}(t_f) \cdot D &= 0.0361 \end{aligned}$$

These transversality condition values are much higher than in the previous minimum effort problem. While larger, the final error is still small, and comparing the XOR trajectory to the simple rotation for the same end-point it is clear that the XOR solution is the same and therefore is evaluated as meeting all necessary conditions to be an optimal solution.

6.4 Summary

The simulation results presented in this chapter again support the conclusion that the newly developed approach for implementing the exclusive disjunction (XOR) end-point constraint can be applied for practical space problems –in this case an optimal spacecraft reorientation. In each case the coded algorithm was able to use the developed continuous error surface to choose the minimum cost rotation and develop a control solution to rotate the spacecraft to the correct final quaternion state, while meeting all the necessary conditions for optimality.

The goal of this Chapter was to implement a method for choosing between two reorientations. It did not specifically address how well the algorithm reached the final state. It is however important to note, as highlighted earlier, that there were very small differences between the calculated trajectories, and final cost, between the XOR rotations and the single end-point rotations. This was due to small differences in the final end-point error. The more complicated end-point definition reduced the final accuracy very slightly.

For the minimum time rotation the final quaternion target was

$$q_T = \begin{bmatrix} 0 \\ 0 \\ 0.258819 \\ 0.965925 \end{bmatrix}$$

while the single end-point rotation ended at

$$q_{SR} = \begin{bmatrix} 9.0058E^{-22} \\ -8.8416E^{-19} \\ 0.2588 \\ 0.9659 \end{bmatrix}$$

and the XOR rotation ended at

$$q_{XOR} = \begin{bmatrix} 71059E^{-8} \\ -1.4509E^{-7} \\ 0.2584 \\ 0.9660 \end{bmatrix}$$

These very small differences in the final end-point were, in the most significant case, at the fourth decimal. These differences resulted in correspondingly small differences in trajectory and final cost. While the difference is small, it is not zero. It is important to consider what degree of final position accuracy is required for a given implementation. For a satellite slewing maneuver, this solution is sufficient to be implemented in a planning architecture

to determine the minimum cost rotation. If it were to be implemented in an actual slewing controller, the XOR solution is accurate enough to rotate a spacecraft to the point that a feedback controller could drive the residual error to any needed requirement.

CHAPTER 7:
Conclusions and Future Work

7.1 Conclusions

This thesis outlined the need to include non-conjunctive constraints as part of continuous-time optimal control problems. Expanding the available logic connectives for OCPs beyond conjunctions, can enable spacecraft like the Lunar Reconnaissance Orbiter to more effectively plan and solve for minimum cost trajectory solutions. This would reduce the time and fuel required to maneuver between stations and increase the number of potential collection events, fuel lifetime, and total mission value. While significant research has been done to develop autonomous trajectory and path-planning solutions, this thesis expands on that body of work, detailing one method for implementing logical XOR end-point constraint conditions into the control and trajectory planning.

The two-dimensional double integrator model of a point mass is a reasonable, albeit simplified, proxy for a spacecraft. This initial piece of the thesis work effectively demonstrated that, in this idealized application, this new method can be used to effectively implement the XOR end-point constraint. Expanding the implementation of this new method to a more complex spacecraft reorientation model was an integral step in the development of this new optimal control tool. This work was able to effectively demonstrate that a trajectory optimization solver could autonomously determine which final quaternion represented the minimum cost rotation.

7.2 Future Work

This work should continue to evolve. Implementation in increasingly complex systems and eventually hardware is a logical and achievable goal. One interesting topic to explore going forward is the possibility of inserting parameters into the problem formulations (see [30]). It is likely that the use of end-point parameters could reduce the final error when using more complex end-point error functions.

Another avenue for future work to continue exploring the question of non-conjunctive

constraints is the use of non-conjunctive logic in trajectory path constraints. There is a subtle but important difference between end-point and path constraints that alters the way in which they are implemented. That is that end-point constraints are valid only for an instant in the trajectory, while path constraints must be true throughout.

Additionally, exploring logical operations other than OR/XOR could prove valuable. Developing a method for implementing the Boolean NAND (not AND) block would be particularly valuable. This is because in Boolean logic the NAND block has the property of functional completeness, meaning that it is possible to develop any truth table using only the NAND blocks. If a method for implementing the NAND block is developed, then it is possible that any Boolean truth table function could be implemented into continuous-time optimal control problems in the manner detailed in this thesis.

List of References

[1] L. S. Pontryagin, *Mathematical Theory of Optimal Processes*. Routledge, 2018.

[2] I. M. Ross, *A Primer on Pontryagin's Principle in Optimal Control*, 2nd ed. Collegiate Publishers, 2015.

[3] D. E. Kirk, *Optimal Control Theory: An Introduction*. Courier Corporation, 2004.

[4] A. E. Bryson, *Applied Optimal Control: Optimization, Estimation and Control*. Routledge, 2018.

[5] P. C. Calhoun and J. C. Garrick, "Observing mode attitude controller for the lunar reconnaissance orbiter," in *Proceedings of the 20th International Symposium on Space Flight Dynamics*, 2007.

[6] R. Arledge, "Implementation of optimal controls using conventional control systems," Master's thesis, Dept. of Mech. and Aerosp. Eng., Naval Postgraduate School, Monterey, CA, 2014.

[7] J. Kaufman, "Automated maneuver design and checkout for the Lunar Reconnaissance Orbiter," Master's thesis, Dept. of Mech. and Aerosp. Eng., Naval Postgraduate School, Monterey, CA, 2014.

[8] T. Lippman, "Enhancing the science collection capability of NASA's Lunar Reconnaissance Orbiter (LRO)," Master's thesis, Dept. of Mech. and Aerosp. Eng., Naval Postgraduate School, Monterey, CA, 2017.

[9] The LRO Mission. NASA. [Online]. Available: https://lunar.gsfc.nasa.gov/about.html. Accessed Apr. 20, 2020.

[10] LROC. [Online]. Available: https://www.lroc.asu.edu. Accessed Apr. 20, 2020.

[11] NASA Probe Beams Home Best Moon Map Ever. (2011, Nov. 18). SPACE.com. [Online]. Available: https://www.space.com/13666-moon-map-lunar-reconnaissance-orbiter.html. Accessed Apr. 20, 2020.

[12] USGS Releases First-Ever Comprehensive Geologic Map of the Moon. (2020). USGS. [Online]. Available: https://www.usgs.gov/news/usgs-releases-first-ever-comprehensive-geologic-map-moon. Accessed Aug. 19, 2020.

[13] The LRO Instruments. NASA. [Online]. Available: https://lunar.gsfc.nasa.gov/instruments.html. Accessed Apr. 20, 2020.

[14] Cosmic Ray Telescope for the Effects of Radiation. (2015). University of New Hampshire. [Online]. Available: http://crater.sr.unh.edu/. Accessed Apr. 20, 2020.

[15] D. Paige and B. Greenhagen, "Diviner lunar radiometer experiment extended mission results: Thermal, thermophysical and compositional properties," in *Lunar and Planetary Science Conference*, 2013, vol. 44, p. 2492.

[16] The Lyman Alpha Mapping Project: Seeing in the Dark. Southwest Research Institute (SwRI). [Online]. Available: https://www.boulder.swri.edu/lamp/. Accessed Apr. 20, 2020.

[17] I. Mitrofanov, A. Bartels, Y. Bobrovnitsky, W. Boynton, G. Chin, H. Enos, L. Evans, S. Floyd, J. Garvin, D. Golovin *et al.*, "Lunar Exploration Neutron Detector for the NASA Lunar Reconnaissance Orbiter," *Space science reviews*, vol. 150, no. 1-4, pp. 183–207, 2010.

[18] Lunar Orbiter Laser Altimeter. NASA Goddard. [Online]. Available: https://lola.gsfc.nasa.gov/. Accessed Apr. 20, 2020.

[19] Apollo 11: 'A Stark Beauty All Its Own'. NASA. [Online]. Available: https://www.nasa.gov/mission_pages/LRO/news/apollo-11.html. Accessed Apr. 20, 2020.

[20] Apollo 15: Interplanetary Mountaineers. NASA. [Online]. Available: https://www.nasa.gov/content/goddard/apollo-15-original-interplanetary-mountaineers. Accessed Apr. 20, 2020.

[21] LRO Finds Apollo 16 Booster Rocket Impact Site. (2019, Apr. 3). NASA. [Online]. Available: https://www.nasa.gov/image-feature/goddard/lro-finds-apollo-16-booster-rocket-impact-site. Accessed Apr. 20, 2020.

[22] J. Halverson, P. Calhoun, O. Hsu, J.-E. Dongmo, R. Besser, B. Ellis, R. De Hart, Y. Tedla, S. Rosney, and S. T. Snell, "Testing of the lunar reconnaissance orbiter attitude control system re-design without a gyro," in *42nd Annual AAS Guidance and Control Conference*. Breckenridge, CO: AAS 19-108, 2019.

[23] E. Mendelson, *Introduction to Mathematical Logic*. CRC Press, 2009.

[24] A. Fleming, P. Sekhavat, and I. M. Ross, "Minimum-time reorientation of a rigid body," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 1, pp. 160–170, 2010.

[25] A. Fleming, "Real-time optimal slew maneuver design and control," Master's thesis, Dept. of Mech. and Astro. Eng., Naval Postgraduate School, Monterey, CA, 2004.

[26] S. Shi, H. Chen, M. Zhang, and Y. Zhang, "Neural logic networks," *arXiv preprint arXiv:1910.08629*, 2019.

[27] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. Prentice-Hall, Inc., 2000.

[28] C. M. Bishop *et al.*, *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[29] M. Athans and P. Falb, *Optimal Control an Introduction to the Theory and Its Applications*. McGraw-Hill, 1966.

[30] I. M. Ross, "Enhancements to the DIDO optimal control toolbox," *arXiv preprint arXiv:2004.13112*, 2020.

[31] T. Lippman, J. Kaufman, and M. Karpenko, "Autonomous planning of constrained spacecraft reorientation maneuvers," in *AAS/AIAA Astrodynamics Specialist Conference*. Stevenson, WA: AAS 17-676, 2017.

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

THIS PAGE INTENTIONALLY LEFT BLANK

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California