

A Legendre Pseudospectral Method for Rapid Optimization of Launch Vehicle Trajectories

by

Jeremy Ryan Rea

B.S. Aeronautical and Astronautical Engineering
The Ohio State University, 1999

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of
Master of Science in Aeronautics and Astronautics
at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2001

© 2001 Jeremy Ryan Rea. All rights reserved.

The author hereby grants to MIT permission to reproduce and to distribute publicly
paper and electronic copies of this thesis document in whole or in part.

Author *Jeremy R. Rea*
Department of Aeronautics and Astronautics

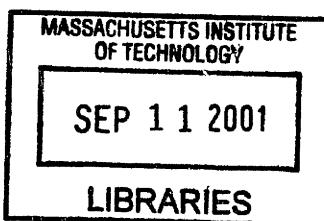
May 11, 2001

Certified by *M. Ross*
I. Michael Ross
Associate Professor of Aeronautics and Astronautics
Naval Postgraduate School
Thesis Supervisor

Certified by *Richard E. Phillips*
Richard E. Phillips
The Charles Stark Draper Laboratory, Inc.
Thesis Supervisor

Certified by *John J. Deyst*
John J. Deyst
Professor of Aeronautics and Astronautics, MIT
Thesis Supervisor

Accepted by *Wallace E. Vander Velde*
Wallace E. Vander Velde
Professor of Aeronautics and Astronautics, MIT
Chair, Committee on Graduate Students



[This page intentionally left blank.]

A Legendre Pseudospectral Method for Rapid Optimization of Launch Vehicle Trajectories

by

Jeremy Ryan Rea

Submitted to the Department of Aeronautics and Astronautics
on May 11, 2001, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

A Legendre Pseudospectral Method for launch vehicle trajectory optimization, proposed by Mike Ross and Fariba Fahroo of the Naval Postgraduate School, is presented and applied successfully to several launch problems. The method uses a Legendre pseudospectral differentiation matrix to discretize nonlinear differential equations (such as the equations of motion) into nonlinear algebraic equations. The equations are then posed in the form of a nonlinear optimization problem and solved numerically. The method is demonstrated to work very well with both continuous and discontinuous states and controls. The method is applied to the following launch problems: the single-stage and two-stage Goddard problem; a simplified, single-stage, two-dimensional launch problem; and a two-stage, three-dimensional launch problem. A technique for reducing the size of problems with second order differential equations is presented and applied. This technique is shown to increase the speed of convergence of some problems. A predictive guidance algorithm demonstrates the feasibility of using the method to find both open- and closed-loop controls for a single-stage, three-dimensional launch problem. The effects of different aerodynamic models between the optimizer and the "real" simulation are briefly analyzed. A number of issues that need to be resolved before the algorithm can be used in-flight for closed-loop guidance are described.

Thesis Supervisor: I. Michael Ross

Title: Associate Professor of Aeronautics and Astronautics, Naval Postgraduate School

Thesis Supervisor: Richard E. Phillips

Title: Staff, The Charles Stark Draper Laboratory, Inc.

Thesis Supervisor: John J. Deyst

Title: Professor of Aeronautics and Astronautics, MIT

[This page intentionally left blank.]

Acknowledgments

I would like to express my gratitude to all of those who have helped make this thesis a reality, and who made my time at MIT so rewarding.

Thanks to my advisors. Mike Ross, thanks for answering all of my questions whenever I asked. Also, thanks for introducing me to the wonderful world of pseudospectral methods! Dick Phillips, thanks for all of your help with boost guidance. Dr. John Deyst, thanks for helping me with my thesis from the MIT side. Also, thanks for sponsoring me on the doctoral qualifying exam.

I would like to thank Draper Lab for making this thesis possible by funding my second year of graduate study and giving me a great opportunity to work on an exciting topic. Thanks go to Tim Brand, Gregg Barton, Anil Rao, Chris Stoll, Tom Fill, Tom Thorvaldsen, David Geller, and the whole Education Office.

Thanks to the MIT Aero/Astro department for funding my first year of graduate study. Thanks to the administrative support staff, particularly Marie Stuppard. Thanks to Dr. David Miller and Dr. Ray Sedwick from the Space Systems Lab for giving me advice during my first year.

Thanks to all of the professors in the Ohio State Aero/Astro department. Go Buckeyes! Thanks to Rosemary Hill, former director of the Ohio State Cooperative Education and Internship Program. Thanks to NASA Johnson Space Center. I would especially like to thank Butch Cockrell in the Advanced Mission Design Branch for helping me make the Draper connection.

Thanks to all my friends who made MIT and Boston a great place to play. Dave, Ishtak, Zack, Andrew, the whole class of 16.89 Spring 2000, Paul Goulart (Thanks for helping me to pass the quals. I won't forget those late nights studying on the 5th floor...although I wish I could). Thanks to all the other 5th floor Draper Fellows: Raja, Tim, Andre.

A special thanks to my family: Mom and dad, thanks for being so supportive with all of my dreams. Matt, thanks for being a great brother. The best for last: Thanks Denise, for everything.

This thesis was prepared at The Charles Stark Draper Laboratory, Inc., under Independent Research and Development #13084, Launch-On-Demand Guidance.

Publication of this thesis does not constitute approval by Draper or the sponsoring agency of the findings or conclusions contained herein. It is published for the exchange and stimulation of ideas.

Permission is hereby granted by the author to the Massachusetts Institute of Technology to reproduce any or all of this thesis.

Jeremy R. Rea Jeremy R. Rea May 11, 2001

Contents

1	Introduction	19
1.1	Motivation	19
1.2	Overview	20
1.3	Research Objectives	23
1.4	Software and Hardware Specifications	23
1.4.1	Software	23
1.4.2	Computer Hardware	23
1.5	Approach and Outline	24
2	Pseudospectral Methods	27
2.1	Finite Difference Matrix	27
2.2	Pseudospectral Differentiation Matrix	31
3	The Legendre Pseudospectral Method	37
3.1	Example: The Single-Stage Goddard Problem	37
3.1.1	The Problem	37
3.1.2	Numerical Results	49
3.2	Staging	54
3.3	Example: The Two-Stage Goddard Problem	57
3.3.1	The Problem	57
3.3.2	Numerical Results	64
4	Coordinate System Comparison	69

4.1	Cartesian Coordinate System	70
4.2	Radial-Transverse Polar Coordinates	76
4.3	Normal-Tangential Polar Coordinates	82
4.4	Coordinate System Comparison Results	89
5	<i>D</i>² Order Reduction Method	97
5.1	<i>D</i> ² -Method	97
5.2	Application of <i>D</i> ² -Method to Two-Dimensional Coordinate Systems .	99
5.2.1	Cartesian Coordinates	99
5.2.2	Radial-Transverse Polar Coordinates	103
5.2.3	Normal-Tangential Polar Coordinates	107
5.3	Coordinate System Comparison Using the <i>D</i> ² -Method	107
5.4	Application of <i>D</i> ² -Method to Three-Dimensional Coordinate Systems	111
5.4.1	Cartesian Coordinates	111
5.4.2	Cylindrical Coordinates	112
5.4.3	Radial-Transverse Spherical Coordinates	113
5.4.4	Normal-Tangential Spherical Coordinates	115
5.5	Coordinate System Conclusions	116
6	Three-Dimensional Cartesian Model	119
6.1	Dynamic Constraints	120
6.1.1	Equations of Motion	120
6.1.2	Mass Flow Rate	121
6.1.3	Dynamic Constraints	122
6.2	Trajectory Constraints	124
6.2.1	Quaternion Normalization Constraint	124
6.2.2	Dynamic Pressure Constraint	124
6.2.3	Sensed Acceleration Constraint	124
6.2.4	Mass Change Constraint	125
6.3	Initial and Final Constraints	125
6.3.1	Initial Constraints	126

6.3.2	Final Altitude Constraint	126
6.3.3	Final Velocity Constraint	127
6.3.4	Final Flight Path Angle Constraint	127
6.3.5	Final Inclination Constraint	128
6.4	Staging Constraints	128
6.5	Optimization Vector	129
6.5.1	D -Method Optimization Vector	130
6.5.2	D^2 -Method Optimization Vector	131
6.5.3	Optimization Vector Bounds	131
6.6	Definition of Three Dimensional Model	132
6.6.1	D^2 -Method Velocity	132
6.6.2	Q Attitude Matrix and Euler Angles	133
6.6.3	Thrust	137
6.6.4	Gravity	138
6.6.5	Aerodynamic Force	139
6.6.6	Dynamic Pressure	140
6.6.7	Wind Relative Velocity	140
6.6.8	Body Velocity	141
6.6.9	Inertial Aerodynamic Coefficients	142
6.6.10	Mach Number	143
6.6.11	Aerodynamic Angles	143
6.6.12	Body Aerodynamic Coefficients	145
6.6.13	Atmosphere Model	148
7	Three Dimensional Trajectory Optimization	149
7.1	Vehicle Definition	149
7.2	Numerical Results	152
8	Guidance Concepts	161
8.1	Extracting Controls from the LGL Points	161
8.2	Open-Loop Guidance Demonstration	162

8.3	Closed-Loop Guidance Demonstration	166
8.4	Figures	169
8.5	Conclusions	178
8.5.1	Lessons Learned	178
8.5.2	Ideas for the Future	179
9	Conclusions	181
9.1	Summary	181
9.2	Conclusions	182
9.3	Future Work	182
A	Notes on Notation	185
B	Matrix Math	187
B.1	Matrix Derivatives	187
B.2	Chain Rule for Matrix Derivatives	188
B.3	Distributive Law for Matrices	190
C	Atmosphere Models	191
C.1	Exponential Atmosphere Model	191
C.2	Temperature-Based Atmosphere Model	192
D	Initial State Definition	193
E	Initial Guess	197
F	Coordinate System Comparison: Jacobian Derivation	203
F.1	Cartesian Coordinate System Jacobian	203
F.2	Radial-Transverse Polar Coordinate System Jacobian	209
F.3	Normal-Tangential Polar Coordinate System Jacobian	213
F.4	D^2 Cartesian Coordinate System Jacobian	216
F.5	D^2 Radial-Transverse Polar Coordinate System Jacobian	221

G Three Dimensional Cartesian Model: Jacobian Derivation	227
G.1 Dynamic Constraint Jacobian	227
G.2 Trajectory Constraint Jacobian	232
G.3 Initial and Final Constraint Jacobian	235
G.4 Staging Constraint Jacobian	240
G.5 Partial Derivatives	240
G.5.1 Partial Derivatives of D^2 -Method Velocity	240
G.5.2 Partial Derivatives of Q Attitude Matrix	241
G.5.3 Partial Derivatives of Thrust	242
G.5.4 Partial Derivatives of Gravity	244
G.5.5 Partial Derivatives of Aerodynamic Force	245
G.5.6 Partial Derivatives of Dynamic Pressure	246
G.5.7 Partial Derivatives of Relative Velocity Components and Mag- nitude	247
G.5.8 Partial Derivatives of Wind Relative Velocity in Body Coordinates	249
G.5.9 Partial Derivatives of Inertial Aerodynamic Coefficients	251
G.5.10 Partial Derivatives of Mach Number	253
G.5.11 Partial Derivatives of α and β	253
G.5.12 Partial Derivatives of Body Aerodynamic Coefficients	254
G.5.13 Partial Derivatives of Atmosphere Model	255
H Guidelines for Applying the Legendre Pseudospectral Method	257
References	259

[This page intentionally left blank.]

List of Figures

2-1 Position Data Taken at Constant Time Points	28
2-2 Position Data Taken at LGL Time Points	33
3-1 Single-Stage Goddard Problem – Time History of Altitude	52
3-2 Single-Stage Goddard Problem – Time History of Velocity	52
3-3 Single-Stage Goddard Problem – Time History of Mass	53
3-4 Single-Stage Goddard Problem – Time History of Thrust	53
3-5 Total Time Domain Divided Into Phases	54
3-6 Two-Stage Goddard Problem – Time History of Altitude	66
3-7 Two-Stage Goddard Problem – Time History of Velocity	66
3-8 Two-Stage Goddard Problem – Time History of Mass	67
3-9 Two-Stage Goddard Problem – Time History of Thrust	67
4-1 Cartesian Coordinates – Forces on a Point Mass	71
4-2 Radial-Transverse Polar Coordinates – Forces on a Point Mass	77
4-3 Normal-Tangential Polar Coordinates – Forces on a Point Mass . . .	83
4-4 Relation Between Normal-Tangential and Radial-Transverse Directions	83
4-5 Objective Function vs. Iterations, 20 points	91
4-6 CPU Time vs. Iterations, 20 points	91
4-7 Objective Function vs. Iterations, 40 points	92
4-8 CPU Time vs. Iterations, 40 points	92
4-9 Two-Dimensional Launch: Time History of Altitude	93
4-10 Two-Dimensional Launch: Trajectory	93
4-11 Two-Dimensional Launch: Time History of Velocity	94

4-12	Two-Dimensional Launch: Time History of Flight Path Angle	94
4-13	Two-Dimensional Launch: Time History of Mass	95
4-14	Two-Dimensional Launch: Time History of Thrust	95
4-15	Two-Dimensional Launch: Time History of Dynamic Pressure	96
4-16	Two-Dimensional Launch: Time History of Sensed Acceleration	96
5-1	Objective Function vs. Iterations, 20 points	109
5-2	CPU Time vs. Iterations, 20 points	109
5-3	Objective Function vs. Iterations, 40 points	110
5-4	CPU Time vs. Iterations, 40 points	110
6-1	Forces on a Vehicle in Powered Atmospheric Flight	120
6-2	Flight Path Angle and Zenith Angle (adapted from [2, p. 17])	127
6-3	Aerodynamic Angles	143
6-4	Right Spherical Triangle formed by Aerodynamic Angles	147
7-1	Three-Dimensional Launch: Time History of Altitude	155
7-2	Three-Dimensional Launch: Time History of Trajectory	155
7-3	Three-Dimensional Launch: Time History of Mach Number	156
7-4	Three-Dimensional Launch: Time History of Euler Angles	156
7-5	Three-Dimensional Launch: Time History of Aerodynamic Angles	157
7-6	Three-Dimensional Launch: Time History of Body Coefficients	157
7-7	Three-Dimensional Launch: Time History of Relative Flight Path Angle	158
7-8	Three-Dimensional Launch: Time History of Mass	158
7-9	Three-Dimensional Launch: Time History of Mass Flow Rate	159
7-10	Three-Dimensional Launch: Time History of Thrust	159
7-11	Three-Dimensional Launch: Time History of Dynamic Pressure	160
7-12	Three-Dimensional Launch: Time History of Sensed Acceleration	160
8-1	Case I: Altitude Comparison	170
8-2	Case I: Controls	170

8-3	Case I: Dynamic Pressure Comparison	171
8-4	Case I: Sensed Acceleration Comparison	171
8-5	Case II: Altitude Comparison	172
8-6	Case II: Controls	172
8-7	Case II: Dynamic Pressure Comparison	173
8-8	Case II: Sensed Acceleration Comparison	173
8-9	Case III: Altitude Comparison	174
8-10	Case III: Controls	174
8-11	Case III: Dynamic Pressure Comparison	175
8-12	Case III: Sensed Acceleration Comparison	175
8-13	Case IV: Altitude Comparison	176
8-14	Case IV: Controls	176
8-15	Case IV: Dynamic Pressure Comparison	177
8-16	Case IV: Sensed Acceleration Comparison	177
D-1	Topocentric-Horizon Launch Coordinate System (adapted from [2]) .	194
D-2	Relationship Between Launch Coordinates and Inertial Coordinates .	195

[This page intentionally left blank.]

List of Tables

1.1 Computer Hardware Specifications	24
3.1 Numerical Values Used for Single-Stage Goddard Problem	50
3.2 Numerical Values Used for Two-Stage Goddard Problem	65
4.1 Numerical Values Used for Coordinate System Comparison	89
7.1 Planet Definition	149
7.2 Mass Definition	150
7.3 Aerodynamic Definition	151
7.4 Rocket Engine Definition	151
7.5 Trajectory Constraint Definition	151
7.6 Initial Vehicle State	152
7.7 Final Target Orbit	152
8.1 Numerical Values Used for Guidance Simulation	163
8.2 Case I: Final Mass and Orbit Comparison	165
8.3 Case II: Final Mass and Orbit Comparison	166
8.4 Case III: Final Mass and Orbit Comparison	168
8.5 Case IV: Final Mass and Orbit Comparison	169

[This page intentionally left blank.]

Chapter 1

Introduction

1.1 Motivation

The flight of a typical launch vehicle begins at the surface of Earth, where the atmosphere is thickest, and ends in orbit around the Earth, where it is effectively in a vacuum. Current launch vehicles use both closed-loop and open-loop guidance schemes. Fuel optimal closed-loop guidance schemes have been used with wide success for the vacuum conditions above the atmosphere. These techniques are based mainly on the analytical methods of the calculus of variations. However, within the atmosphere, the effect of drag causes the system to be highly nonlinear, and so far no analytical optimal solution has been found for this regime.

During atmospheric flight, launch vehicles use an open-loop guidance scheme. The open-loop guidance is based on a reference trajectory that is designed before the launch. Over the years, techniques have been developed to find endo-atmospheric, fuel-optimal trajectories. The problem is that the tools are cumbersome and time consuming. In the case of the Space Shuttle, months of planning are required to create a trajectory. A problem with this method is that certain assumptions about the winds and weather must be made when the trajectories are developed. If a launch were to take place in spite of violations in these assumptions, it is possible that constraint violations would occur or that the fuel load would be inadequate. This can lead to costly launch delays.

In this situation, launch is delayed until conditions are once more within the margins that were used to plan the trajectory. There is a story of a Space Shuttle launch that was delayed because the winds were not high enough! If it were possible to generate trajectories minutes before launch, delays could be avoided.

If the conditions still allow a launch, it may be necessary for the vehicle to fly off of its pre-planned trajectory in order to avoid violating aerodynamic heating and structural load limits. The rest of the open-loop trajectory would then no longer be fuel-optimal, although the closed-loop exo-atmospheric guidance would partially compensate for the loss.

A rapid trajectory optimization scheme is desirable for many reasons. It could allow trajectories to be planned hours or minutes before launch, thus cutting down on delays due to uncertainties in the wind and weather. If a near real-time trajectory optimization scheme could be found, it could be used to reoptimize the trajectory in flight. This could be used as the basis for a closed-loop endo-atmospheric guidance scheme (i.e. closed-loop nonlinear control system), thus making the system more robust. It should also be noted that such a capability could be used for other vehicles that require closed-loop nonlinear optimal guidance and control. Rapid trajectory optimization could be used for reentry applications, air-to-air guidance, or even for underwater vehicles.

Robotic and manned missions to Mars would also benefit from rapid trajectory optimization. These missions will require an in-situ capability to create their own trajectories for closed-loop guidance in order to maximize safety margins during ascent from the Martian surface. This is especially true given the unknown characteristics of the Martian environment.

1.2 Overview

There are many methods for launch vehicle trajectory optimization. A short description of the main methods follows. For an in-depth discussion of current trajectory optimization techniques, see [3]. These methods can be grouped into two major

categories: indirect and direct methods [3].

In indirect methods, calculus of variations is used to explicitly solve for the optimality conditions of the optimal trajectory [3]. The problem is reduced to a nonlinear multipoint boundary value problem [3]. Indirect methods have two main drawbacks that preclude their use with complex problems. First, the radii of convergence for the problem is usually very small [8]. Second, it takes a lot of work to analytically derive the optimality conditions for a complex problem [8].

Direct methods convert the trajectory optimization problem into a parameter optimization problem. This results in a nonlinear programming problem that can be solved using any one of many nonlinear programming codes [3]. The conversion is done by discretizing the time domain into a set of subintervals. The end points of these intervals are called nodes. The parameters are the values of the states and controls at these nodes [8].

Currently, the two most widely used programs for numerical trajectory optimization are Program to Optimize Simulated Trajectories (POST) and Optimal Trajectories by Implicit Simulation/Sparse Optimal Control Software (OTIS/SOCS). POST was originally developed by NASA in the 1970's as a trajectory optimizer for the Space Shuttle. Many capabilities have been added since that time [4]. OTIS was developed by Boeing in 1987 for the United States Air Force. SOCS was added by NASA Glenn in 1997 [16]. Both of these methods use numerical approximations of differential equations. Some typical approximation schemes include trapezoidal, Euler, and Hermite-Simpson [8].

Pseudospectral methods have been used in the field of fluid dynamics for several years to solve partial differential equations [23]. Their use in optimal control theory for systems governed by ordinary differential equations is relatively new. Approximations of differential equations using pseudospectral methods are more accurate than those mentioned above. In fact, “they can often achieve ten digits of accuracy where a finite difference or finite element method would get two or three.” [23]. These methods, combined with high-powered computers and numerical optimization routines, should be well suited for rapid trajectory optimization.

The purpose of this research is to apply pseudospectral methods to the launch problem, for which rapid trajectory optimization is needed. In particular, the method used is called the Legendre Pseudospectral Method, created by Fariba Fahroo and Michael Ross of the Naval Postgraduate School. It falls into the direct method category described above. It combines pseudospectral methods with numerical nonlinear optimization. The time domain of the trajectory is discretized at a special set of points called the Legendre-Gauss-Lobatto points. The use of these points results in improved accuracy. The pseudospectral differentiation matrix then provides a numerical estimate of the derivatives at those points. The pseudospectral differentiation matrix transforms the equations of motion for a launch vehicle from nonlinear differential equations to nonlinear algebraic equations. These equations are then posed in the form of a nonlinear programming problem, and a numerical optimizer is used to solve the problem.

During the period of the study, Ross and Fahroo made two very important advances which make the method particularly appropriate to launch vehicle guidance [17]. The first of these is the reduction of the size of the optimization problem by returning to a description of the problem in terms of second order differential equations and a second order differentiation matrix. The second is the ability to handle both continuous and discontinuous states and controls. This is very useful when finding an optimal solution for a multistage launch vehicle, where the mass of the vehicle is discontinuous at the staging points. Both of these developments are used in this research.

It should be noted that the Legendre Pseudospectral Method is used in this research as a guidance algorithm only. This means that the control system of the vehicle is not modelled. For example, most launch vehicles have engines that can be gimballed in order to control the thrust direction. This control is not modelled. However, this is not to say that the control system can not be included in the optimization scheme; it can. Further, it would even be possible to use the optimization scheme to include vehicle design. However, for the purpose of this research, the method is only applied to trajectory optimization and guidance.

1.3 Research Objectives

This thesis investigates the following:

- The use of a Legendre pseudospectral method for rapid launch vehicle trajectory optimization, for both single-stage and multi-stage rockets.
- The possible use of such rapid trajectory optimization for launch vehicle guidance. This includes numerical examination of the robustness and convergence properties of the trajectory optimization scheme.

1.4 Software and Hardware Specifications

1.4.1 Software

MATLAB

All of the numerical work in this thesis is written using MATLAB, version 5.3.1.29215a (R11.1). This version was released on September 28, 1999. MATLAB is a product of The MathWorks, Inc.

NPSOL

The numerical optimizer that is used in this thesis is NPSOL. It was developed by Philip Gill at the University of California, Walter Murray and Michael Saunders at Stanford University, and Margaret Wright at Bell Laboratories [11]. NPSOL was originally written as a set of FORTRAN subroutines for nonlinear programming problems. For a detailed description of NPSOL, see [11]. A NPSOL mexfile is used as a MATLAB interface to the NPSOL optimization routines. This enables MATLAB to run the compiled FORTRAN routines as though they are MATLAB functions.

1.4.2 Computer Hardware

The Engineering Design and Computational Facility (EDCF) at The Charles Stark Draper Laboratory, Inc. is used for all work in this thesis. Only the computer labelled

DC6 is used. Table 1.1 lists the hardware specifications of DC6.

Manufacturer	Sun Microsystems
System Model	Enterprise 450 Model 4400
Main Memory	2.0 GB
Virtual Memory	4.7 GB
Operating System	SunOS 5.6
Number of CPU's	4
CPU Type	Sun UltraSPARC-II 400 MHz CPU

Table 1.1: Computer Hardware Specifications

1.5 Approach and Outline

In the succeeding chapters the Legendre Pseudospectral Method is described in its several aspects. The method is then applied to launch vehicle trajectory optimization problems of increasing complexity, culminating with the generation of an optimal three dimensional trajectory for a two-stage launch vehicle subject to constraints on thrust, dynamic pressure, and sensed acceleration.

In Chapter 2, the application and theory of the pseudospectral differentiation matrix is described. The pseudospectral differentiation matrix is described with an analogy to the finite difference matrix. Chapter 3 describes the application of the Legendre Pseudospectral Method to single- and multi-stage launch vehicles. The method is applied to the relatively simple Goddard problem to aid the reader in understanding the method. The method is then applied to a two-stage Goddard problem to show how it can handle discontinuities in states and controls (discontinuous mass in this case).

A short study is done to get some insight into how the choice of a coordinate system affects the numerical properties (i.e. robustness and speed of convergence) of the launch optimization problem. A simplified, two-dimensional launch problem is solved using several different coordinate systems. The results of this study are described in Chapter 4.

Chapter 5 discusses a method that reduces the size of the numerical optimization problem. The method uses a second order difference operator that reduces the size of the numerical optimization problem, thus increasing the speed of convergence of the algorithm. This method is applied to the coordinate systems of Chapter 4. Based on the relative merits of the two-dimensional coordinate systems compared in Chapter 4, a three-dimensional cartesian coordinate system is chosen for a more complex launch problem. Chapter 6 describes in detail this three dimensional launch model.

In Chapter 7, the Legendre Pseudospectral Method is applied to a realistic three dimensional, two-stage launch trajectory optimization problem. Chapter 8 presents some ideas and results for using the Legendre Pseudospectral Method for real-time, on-board guidance. The guidance commands resulting from the optimization are used in an open-loop launch vehicle simulation, and the predicted and actual trajectories are compared. Then, a closed-loop implementation is tried. The optimizer is used to re-optimize the trajectory in mid-flight, and a comparison is made between the open-loop and closed-loop implementations. Chapter 9 contains a summary of the findings of this research and presents some ideas for future investigation.

[This page intentionally left blank.]

Chapter 2

Pseudospectral Methods

2.1 Finite Difference Matrix

The thesis will begin with a simple explanation of a finite difference matrix. Assume that there is positional data for an object moving in a straight line. Assume that the data was taken at 6 times separated by equal time steps (see Figure 2-1)

Now, suppose that it is desired to find the velocity of the object using the positional data. The velocity is the time derivative of the position.

$$V = \frac{dx}{dt} \quad (2.1)$$

A finite difference formula can be used to estimate the velocity at each time point. For this example, a forward difference approximation is used for the first point, a central difference approximation for the middle points, and a backward difference approximation for the last point. These formulas are derived from the first two terms of a Taylor series expansion [15, p.234-239]. Note that these finite difference formulas are only approximations. The error is due to the truncation of the Taylor series expansion and is relative to the size of the time step (Δt).

Forward Difference [15, p. 238]

$$V_i = \frac{x_{i+1} - x_i}{\Delta t} + E$$

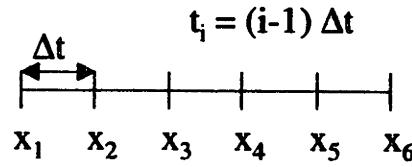


Figure 2-1: Position Data Taken at Constant Time Points

$$E \approx O(\Delta t) \quad (2.2)$$

Central Difference [15, p. 238]

$$V_i = \frac{x_{i+1} - x_{i-1}}{2\Delta t} + E$$

$$E \approx O(\Delta t^2) \quad (2.3)$$

Backward Difference [15, p. 238]

$$V_i = \frac{x_i - x_{i-1}}{\Delta t} + E$$

$$E \approx O(\Delta t) \quad (2.4)$$

The velocity at each time can be written as:

$$\begin{aligned} V_1 &\approx \frac{x_2 - x_1}{\Delta t} \\ V_2 &\approx \frac{x_3 - x_1}{2\Delta t} \\ V_3 &\approx \frac{x_4 - x_2}{2\Delta t} \\ V_4 &\approx \frac{x_5 - x_3}{2\Delta t} \\ V_5 &\approx \frac{x_6 - x_4}{2\Delta t} \\ V_6 &\approx \frac{x_6 - x_5}{\Delta t} \end{aligned} \quad (2.5)$$

These equations can also be rewritten in matrix form:

$$\frac{1}{\Delta t} \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & -\frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & -\frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & -\frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} \approx \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_6 \end{bmatrix} \quad (2.6)$$

This difference matrix is denoted as D_6 and will be referred to as the “ D -matrix.” The subscript “6” refers to the number of points used to find the difference matrix. The equation can be written in matrix-vector form:

$$D_6 \vec{x} \approx \vec{V} \quad (2.7)$$

The error can be reduced if more terms from the Taylor series expansion are added into the difference approximations. For example, if the first three terms of a Taylor series expansion are used, the forward, central, and backward difference approximations can be written as:

Forward Difference [15, p. 238]

$$V_i = \frac{-x_{i+2} + 4x_{i+1} - 3x_i}{2\Delta t} + E$$

$$E \approx O(\Delta t^2) \quad (2.8)$$

Central Difference [15, p. 238]

$$V_i = \frac{-x_{i+2} + 8x_{i+1} - 8x_{i-1} + x_{i-2}}{12\Delta t} + E$$

$$E \approx O(\Delta t^4) \quad (2.9)$$

Backward Difference [15, p. 238]

$$V_i = \frac{3x_i - 4x_{i-1} + x_{i-2}}{2\Delta t} + E$$

$$E \approx O(\Delta t^2) \quad (2.10)$$

Then the matrix equation is:

$$\frac{1}{\Delta t} \begin{bmatrix} -\frac{3}{2} & 2 & -\frac{1}{2} & 0 & 0 & 0 \\ 0 & -\frac{3}{2} & 2 & -\frac{1}{2} & 0 & 0 \\ \frac{1}{12} & -\frac{2}{3} & 0 & \frac{2}{3} & -\frac{1}{12} & 0 \\ 0 & \frac{1}{12} & -\frac{2}{3} & 0 & \frac{2}{3} & -\frac{1}{12} \\ 0 & 0 & \frac{1}{2} & -2 & \frac{3}{2} & 0 \\ 0 & 0 & 0 & \frac{1}{2} & -2 & \frac{3}{2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} \approx \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_6 \end{bmatrix} \quad (2.11)$$

As more and more terms are added in the difference approximations, the error is reduced.

The most important concept is the idea that a continuous differential equation can be approximated by a discrete set of algebraic equations. Given a column vector of position data at different time points, the velocity at those time points can be approximated using the difference matrix.

$$\frac{d}{dt}(x) = V \quad (2.12)$$

$$D_6 \vec{x} \approx \vec{V} \quad (2.13)$$

This equation can be extended to systems of equations with N points, where D_N is an $N \times N$ matrix that can be applied to a vector of length N .

2.2 Pseudospectral Differentiation Matrix

Here is an example of a pseudospectral differentiation matrix based on six Legendre-Gauss-Lobatto (LGL) points:

$$D_6 = \begin{bmatrix} -7.50 & 10.14 & -4.04 & 2.24 & -1.35 & 0.50 \\ -1.78 & 0 & 2.52 & -1.15 & 0.65 & -0.24 \\ 0.49 & -1.72 & 0 & 1.75 & -0.79 & 0.27 \\ -0.27 & 0.79 & -1.75 & 0 & 1.72 & -0.49 \\ 0.23 & -0.65 & 1.15 & -2.52 & 0 & 1.79 \\ -0.50 & 1.35 & -2.24 & 4.04 & -10.14 & 7.50 \end{bmatrix} \quad (2.14)$$

The pseudospectral D -matrix can be used to approximate derivatives in the same way as the finite difference D -matrix. However, the theory behind the pseudospectral differentiation matrix is quite different than that of the finite difference matrix. The main difference in the application is in how the points are spaced (they are no longer equispaced). The following sections will introduce the basic idea behind the theory of the pseudospectral differentiation matrix. They are not intended to be a complete discussion on the theory. For more details, see [23] or [10].

Lagrange Interpolation

Let there be a set of points. It is desired to find an interpolating function for the points (in other words, it is desired to curve-fit the points with a polynomial or a sum of polynomials). One method that can be used is Lagrange interpolation.

With Lagrange interpolation, polynomial functions are used to fit between the data points [6]. For a set of N data points, the Lagrange interpolation formula is [21]:

$$y(t) = \sum_{j=1}^N y_j \phi_j(t) \quad (2.15)$$

where

- $y(t)$ = the approximating polynomial
- y_j = the value of y at t_j
- $\phi_j(t)$ = the set of interpolating functions

The set of interpolating functions are polynomials of order $N - 1$ [21]:

$$\phi_j(t) = \frac{(t - t_1) \cdots (t - t_{j-1})(t - t_{j+1}) \cdots (t - t_N)}{(t_j - t_1) \cdots (t_j - t_{j-1})(t_j - t_{j+1}) \cdots (t_j - t_N)} \quad (2.16)$$

This can also be written as:

$$\phi_j(t) = \prod_{\substack{m=1 \\ m \neq j}}^N \frac{(t - t_m)}{(t_j - t_m)} \quad (2.17)$$

Consider when t is equal to any of the sample points t_i ($i = 1, 2, \dots$). The numerator of ϕ_j is zero except when $i = j$. The denominator normalizes the value of $\phi_j(t)$ to 1 when $i = j$. So, the following is true at the node points [8]:

$$\phi_j(t_i) = \delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \quad (2.18)$$

where δ_{ij} is the Kronecker delta.

Note that the approximation of equation 2.15 only applies to values of t between the t_i node points. By definition, the approximation is exactly equal to the data points at the values of t_i [8]. Also note that there is no reason that the data nodes must be evenly spaced. This is a method of finding a polynomial to approximate a set of arbitrarily spaced data points.

Optimal Node Spacing

As mentioned, the node spacing of the data points is arbitrary. It is possible to choose the node spacing that will give the best polynomial approximation. It turns out that using equispaced points is not the best node spacing [23]. Approximation theory

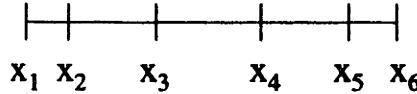


Figure 2-2: Position Data Taken at LGL Time Points

states that the optimal node spacing occurs when the nodes are roots of orthogonal polynomials such as Legendre or Tschebyscheff polynomials [8], [23], [10].

One such set of points is the set of Legendre-Gauss-Lobatto (LGL) points. The LGL points are defined as follows [7]:

$$\begin{aligned} t_1 &= -1 \\ t_j &= \text{roots of } L'_{N-1}(t) \quad (j=2..N-1) \\ t_N &= 1 \end{aligned}$$

where $L_{N-1}(t)$ is the Legendre polynomial of order $N - 1$.

These points lie in the range of [-1,1]. Note that these points must be computed numerically because there are no closed form equations [17]. An example of a set of six LGL points is given in Figure 2-2. Note how the points “bunch up” near the ends and “spread out” in the middle of the domain. This is a characteristic of the LGL points.

As mentioned, the LGL points exist in the range from -1 to 1. Another way of stating this would be to say that the pseudospectral differentiation matrix only operates over a time domain of -1 to 1. In this thesis, the differentiation matrix is used to approximate time derivatives. Therefore, the LGL points correspond to time points. The first LGL point is at time = -1, and the last point is at time = 1. Typically, rocket launches last on the order of 10 minutes. It is necessary to map the real time domain to the LGL time domain by the following equation [8]:

$$t_{LGL_i} = \frac{2(\tau_i - \tau_o) - (\tau_f - \tau_o)}{\tau_f - \tau_o} \quad (2.19)$$

where

- t_{LGL_i} = the ith LGL time point
- τ_i = the ith real time point
- τ_o = the initial real time point
- τ_f = the final real time point

The LGL time domain can be mapped to the real time domain by:

$$\tau_i = \frac{(\tau_f - \tau_o)t_{LGL_i} + (\tau_f + \tau_o)}{2} \quad (2.20)$$

Derivative Approximation

Now, the polynomial approximation can be used to estimate derivatives at the node points. It is desired to approximate the derivative of the function $y(t)$ with respect to t . The derivative of equation 2.15 can be used to estimate the derivative [8]:

$$\dot{y}(t) = \sum_{j=1}^N y_j \dot{\phi}_j(t) \quad (2.21)$$

Remember that only the derivatives at the node points are desired. So,

$$\dot{y}(t_i) = \sum_{j=1}^N y_j \dot{\phi}_j(t_i) = \sum_{j=1}^N D_{ij} y_j \quad (2.22)$$

This equation can now be written in matrix-vector form.

$$\vec{\dot{y}} = D_N \vec{y} \quad (2.23)$$

where

- \vec{y} = vector of N data points at the LGL nodes
- D_N = the NxN pseudospectral differentiation matrix

The elements of the D_N matrix are given by:

$$D_{ij} = \dot{\phi}_j(t_i) \quad (2.24)$$

Lagrange Interpolation with Legendre Polynomials

The Lagrange interpolating function is given in equation 2.17, repeated below.

$$\phi_j(t) = \prod_{\substack{m=1 \\ m \neq j}}^N \frac{(t - t_m)}{(t_j - t_m)} \quad (2.25)$$

This can be written as a function of the Legendre polynomials and the LGL points.

Define the following [7]:

$$w(t) = \prod_{m=1}^N (t - t_m) \quad (2.26)$$

Taking the derivative with respect to time and evaluating at t_j results in [7]:

$$w'(t_j) = (t_j - t_1) \cdots (t_j - t_{j-1}) (t_j - t_{j+1}) \cdots (t_j - t_N) = \prod_{\substack{m=1 \\ m \neq j}}^N (t_j - t_m) \quad (2.27)$$

So, the following is true [7]:

$$\phi_j(t) = \frac{w(t)}{(t - t_j)} \frac{1}{w'(t_j)} \quad (2.28)$$

Recall that the t_j (except for t_1 and t_N) are the roots of the derivative of the Legendre polynomial of order $N - 1$. Therefore, [7]

$$L'_{N-1}(t) = (t - t_2)(t - t_3) \cdots (t - t_{N-1}) \quad (2.29)$$

Equations 2.26 and 2.29 can be combined [7]:

$$\begin{aligned} w(t) &= \prod_{m=1}^N (t - t_m) = (t - t_1)(t - t_2) \cdots (t - t_{N-1})(t - t_N) \\ &= (t - t_1)L'_{N-1}(t)(t - t_N) \end{aligned} \quad (2.30)$$

Noting that $t_1 = -1$ and $t_N = 1$ results in [7]:

$$w(t) = (t^2 - 1) L'_{N-1}(t) \quad (2.31)$$

The Legendre polynomials are the eigenfunctions of [7]:

$$[(1 - t^2) L'_{N-1}]' + N(N - 1) L_{N-1}(t) = 0 \quad (2.32)$$

This can be rewritten as:

$$N(N - 1) L_{N-1}(t_j) = [(t^2 - 1) L'_{N-1}]'_{t=t_j} = \left[\prod_{m=1}^N (t - t_m) \right]'_{t=t_j} = w'(t_j) \quad (2.33)$$

This leads to [7]:

$$\phi_j(t) = \frac{w(t)}{(t - t_j)} \frac{1}{w'(t_j)} = \frac{(t^2 - 1) L'_{N-1}(t)}{N(N - 1) L_{N-1}(t_j)(t - t_j)} \quad (2.34)$$

The derivative of equation 2.34 can be used with equation 2.24 to find the elements of the D_N matrix. The result is [8]:

$$D_{ij} = \begin{cases} \frac{L_{N-1}(t_i)}{L_{N-1}(t_j)} \frac{1}{(t_i - t_j)} & i \neq j \\ -\frac{N(N-1)}{4} & i = j = 1 \\ \frac{N(N-1)}{4} & i = j = N \\ 0 & \text{otherwise} \end{cases} \quad (2.35)$$

Chapter 3

The Legendre Pseudospectral Method

The Legendre Pseudospectral Method used in this thesis was developed by Fariba Fahroo and Mike Ross of the Naval Postgraduate School. This method is a way of writing any problem modelled with differential equations as a problem with only algebraic constraints by use of the pseudospectral differentiation matrix. The problem is written numerically as a parameter optimization problem, where the states, controls, and times are manipulated by a numerical optimizer to achieve an optimal solution. The following section gives an example of how to use the Legendre Pseudospectral Method. Before continuing, it might be useful to review Appendices A and B.

3.1 Example: The Single-Stage Goddard Problem

3.1.1 The Problem

Find the optimal thrust profile that will maximize the final altitude of a single-stage launch vehicle when launched straight-up from the surface of the Earth. This is known as the Goddard problem. Assume the following:

1. The Earth is a perfect sphere.
2. The atmospheric density is exponential.

3. The drag coefficient of the vehicle is constant.

4. The specific impulse of the vehicle is constant.

Dynamic Equations

The following dynamic equations can be written to describe the problem:

$$\begin{aligned}\dot{R} &= V \\ \dot{V} &= \frac{T - D}{m} - g \\ \dot{m} &= -\frac{T}{g_o I_{sp}}\end{aligned}\tag{3.1}$$

where

- R = radial position from the center of the Earth
- V = velocity
- m = vehicle mass
- D = drag
- T = thrust
- g = gravitational acceleration (g_o is at the planet surface)
- I_{sp} = specific impulse of rocket engine

Recall from Chapter 2 that the pseudospectral difference operator only works over the domain $[-1,1]$ at the LGL points. Therefore, the time domain of the differential equations must be transformed from the real time domain to the LGL time domain. Recall equation 2.20, repeated below in a slightly different form:

$$\tau = \frac{(\tau_f - \tau_o)t_{LGL} + (\tau_f + \tau_o)}{2}\tag{3.2}$$

where τ is the real time, and t_{LGL} is the LGL time.

This relationship between τ and t_{LGL} can be used to transform the time domain of the derivatives. Taking the differential of both sides of equation 3.2 results in:

$$d\tau = \frac{(\tau_f - \tau_o)}{2} dt_{LGL} \quad (3.3)$$

The derivatives in the dynamic equations are of the form

$$\frac{dx}{d\tau} = y \quad (3.4)$$

Plugging equation 3.3 into equation 3.4 and rearranging results in:

$$\frac{2}{(\tau_f - \tau_o)} \frac{dx}{dt_{LGL}} = y \quad \text{or} \quad \frac{dx}{dt_{LGL}} = \frac{(\tau_f - \tau_o)}{2} y \quad (3.5)$$

So, in order to transform a derivative with respect to real time to a derivative with respect to LGL time, it is necessary to multiply the derivative times $\frac{2}{(\tau_f - \tau_o)}$. Equations 3.1 can now be transformed to the LGL time domain.

$$\begin{aligned} \dot{R} &= \frac{\tau_f - \tau_o}{2} [V] \\ \dot{V} &= \frac{\tau_f - \tau_o}{2} \left[\frac{T - D}{m} - g \right] \\ \dot{m} &= -\frac{\tau_f - \tau_o}{2} \left[\frac{T}{g_o I_{sp}} \right] \end{aligned} \quad (3.6)$$

Note that the variables in the real time domain and the variables in the LGL time domain are not the same. Strictly speaking, new notation should be used for the LGL time variables. However, in this thesis, the same notation is used for the variables whether in the LGL or real time domain. It will be clear from the context of the equations which domain is correct.

Now that the time domain has been transformed, the differential equations can be changed into algebraic equations by using the pseudospectral differentiation matrix.

$$\begin{aligned} D_N \vec{R} &= \left(\frac{\tau_f - \tau_o}{2} \right) \vec{V} \\ D_N \vec{V} &= \left(\frac{\tau_f - \tau_o}{2} \right) \left[\frac{\vec{T} - \vec{D}}{\vec{m}} - \vec{g} \right] \\ D_N \vec{m} &= \left(\frac{\tau_f - \tau_o}{2} \right) \left[-\frac{\vec{T}}{g_o I_{sp}} \right] \end{aligned} \quad (3.7)$$

where

- \vec{R} = vector of radial positions at the LGL points
- \vec{V} = vector of velocities at the LGL points
- \vec{m} = vector of masses at the LGL points
- \vec{T} = vector of thrusts at the LGL points
- \vec{D} = vector of drags at the LGL points
- \vec{g} = vector of gravitational accelerations at the LGL points
- τ_o = fixed initial time
- τ_f = free final time

Dynamic Constraints

The dynamic equations can be written as equality constraints for the nonlinear program solver. These constraints will be called dynamic constraints.

$$\vec{C}_{\dot{R}} = D_{NN} \vec{R} - \vec{V} = 0 \quad (3.8)$$

$$\vec{C}_{\dot{V}} = D_{NN} \vec{V} - \frac{\vec{T}}{\vec{m}} + \frac{\vec{D}}{\vec{m}} + \vec{g} = 0 \quad (3.9)$$

$$\vec{C}_{\dot{m}} = D_{NN} \vec{m} + \frac{\vec{T}}{g_o I_{sp}} = 0 \quad (3.10)$$

where

$$D_{NN} = \frac{2D_N}{(\tau_f - \tau_o)} \quad (3.11)$$

Note that equations 3.8, 3.9, and 3.10 are vectors of constraints. Each component of a constraint vector corresponds to a constraint at a LGL point. In this thesis, the constraint vectors are always column vectors. The constraint vectors are combined into the main constraint vector (\vec{C}).

$$\vec{C} = \begin{bmatrix} \vec{C}_R \\ \vec{C}_V \\ \vec{C}_m \end{bmatrix} \quad (3.12)$$

Constraint Vector Bounds

Upper and lower bounds must be placed on the constraint vector. In this very simple case, all of the constraints in the constraint vector are dynamic constraints. In this thesis, dynamic constraints will always be defined as equality constraints that must equal zero. Therefore, the upper and lower constraint bound vectors are zero vectors with the same length as the constraint vector.

$$\vec{B}_{L_C} = 0 \quad (3.13)$$

$$\vec{B}_{U_C} = 0 \quad (3.14)$$

Optimization Vector

The optimization vector defines the variables that the nonlinear program solver can manipulate. It is made up of the states and controls at each of the LGL points, as well as any free times in the problem. The optimization vector for this problem is:

$$\vec{x}_{opt} = \begin{bmatrix} \vec{R} \\ \vec{V} \\ \vec{m} \\ \vec{T} \\ \tau_f \end{bmatrix} \quad (3.15)$$

Note that the optimization vector has been written as a column vector. Also note that the states and controls have been written as column vectors. This is the general representation of these vectors. However, for Jacobian matrix computations, it is more convenient to think of the optimization vector, states, and controls as row vectors. This is discussed a bit later.

Optimization Vector Bounds

Upper and lower bounds must be placed on the values of the optimization vector.

- The initial radial position should be equal to the radius of the Earth (R_E). Also, the radial position should never go below the radius of the Earth. There is no upper limit for the radial position.
- The initial velocity should be equal to zero. The velocity is constrained to always be zero or positive. This insures that, during the time of interest, the vehicle only rises; it does not fall. There is no upper velocity limit.
- The initial mass must be equal to m_o . The mass can never go below $m_o - m_p$ where m_p is the amount of propellant. The mass can never be more than m_o .
- The thrust can be directed upward or downward. The lower thrust limit is $-T_{max}$ and the upper thrust limit is T_{max} , where T_{max} is the maximum thrust magnitude.
- The initial time is zero and the final time is free.

The bounds are given by:

Lower Bounds

$$\vec{b}_{l_R} = \begin{bmatrix} R_E & R_E & \cdots & R_E \end{bmatrix} \quad (3.16)$$

$$\vec{b}_{l_V} = \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix} \quad (3.17)$$

$$\vec{b}_{l_m} = \begin{bmatrix} m_o & m_o - m_p & \cdots & m_o - m_p \end{bmatrix} \quad (3.18)$$

$$\vec{b}_{l_T} = \begin{bmatrix} -T_{\max} & -T_{\max} & \cdots & -T_{\max} \end{bmatrix} \quad (3.19)$$

$$b_{l_{\tau_f}} = 0 \quad (3.20)$$

The lower bounds are combined into the lower bound vector.

$$\vec{B}_{L_x} = \left[\vec{b}_{l_R} \quad \vec{b}_{l_V} \quad \vec{b}_{l_m} \quad \vec{b}_{l_T} \quad b_{l_{\tau_f}} \right]^T \quad (3.21)$$

Upper Bounds

$$\vec{b}_{u_R} = \begin{bmatrix} R_E & \infty & \cdots & \infty \end{bmatrix} \quad (3.22)$$

$$\vec{b}_{u_V} = \begin{bmatrix} 0 & \infty & \cdots & \infty \end{bmatrix} \quad (3.23)$$

$$\vec{b}_{u_m} = \begin{bmatrix} m_o & m_o & \cdots & m_o \end{bmatrix} \quad (3.24)$$

$$\vec{b}_{u_T} = \begin{bmatrix} T_{\max} & T_{\max} & \cdots & T_{\max} \end{bmatrix} \quad (3.25)$$

$$b_{u_{\tau_f}} = \infty \quad (3.26)$$

The upper bounds are combined into the upper bound vector.

$$\vec{B}_{U_x} = \left[\begin{array}{ccccc} \vec{b}_{u_R} & \vec{b}_{u_V} & \vec{b}_{u_m} & \vec{b}_{u_T} & b_{u_{rf}} \end{array} \right]^T \quad (3.27)$$

Note that the lower and upper bound vectors are column vectors.

Nondimensionalization

It is desirable to have all of the variables in the optimization vector to be roughly the same order of magnitude. It is necessary to choose a set of dimensional quantities that can be used to nondimensionalize the variables of the problem. The following quantities are chosen as base values:

- Length will be measured in units of the radius of the planet (R_P).
- Velocity will be measured in units of the circular orbit velocity at the surface of the planet ($\sqrt{\frac{\mu}{R_P}}$). Note that μ is the gravitational parameter of the planet.
- Mass will be measured in units of the initial mass of the vehicle (m_o).

From these base units, nondimensionalization constants can be defined. In order to nondimensionalize a quantity, just multiply by the appropriate constant.

$$\tilde{n}_{length} = \frac{1}{R_P} \quad (3.28)$$

$$\tilde{n}_{velocity} = \sqrt{\frac{R_P}{\mu}} \quad (3.29)$$

$$\tilde{n}_{mass} = \frac{1}{m_o} \quad (3.30)$$

$$\tilde{n}_{time} = \frac{\tilde{n}_{length}}{\tilde{n}_{velocity}} \quad (3.31)$$

$$\tilde{n}_{force} = \frac{\tilde{n}_{mass}\tilde{n}_{length}}{\tilde{n}_{time}^2} \quad (3.32)$$

Once the values are nondimensionalized, new notation should be adopted. However, this will not be done in this thesis. **From now on, all values in this thesis**

should be considered to be nondimensionalized unless otherwise stated.

Nonlinear Programming Problem

The problem can now be posed as a nonlinear programming problem.

Minimize:

$$F = -R_{final}$$

Subject to:

$$\begin{aligned}\vec{B}_{L_x} &\leq \vec{x}_{opt} \leq \vec{B}_{U_x} \\ \vec{B}_{L_C} &\leq \vec{C} \leq \vec{B}_{U_C}\end{aligned}$$

This will maximize the final altitude while meeting the dynamic constraints at all of the LGL points.

Objective Gradient

The objective gradient is a vector of the partial derivatives of the objective function (F) with respect to the optimization vector. Most nonlinear program solvers can estimate the objective gradient numerically. However, the solver will run faster if an analytical objective gradient is given. The objective gradient is given by:

$$\vec{F}_{grad} = \frac{\partial F}{\partial \vec{x}_{opt}} \quad (3.33)$$

Note that F is a scalar and \vec{x}_{opt} is a column vector. Therefore, \vec{F}_{grad} is also a column vector. The objective gradient for this problem is very simple.

$$\vec{F}_{grad} = \left[\frac{\partial F}{\partial \vec{R}} \quad \frac{\partial F}{\partial \vec{V}} \quad \frac{\partial F}{\partial \vec{m}} \quad \frac{\partial F}{\partial \vec{T}} \quad \frac{\partial F}{\partial \tau_f} \right]^T \quad (3.34)$$

$$\frac{\partial F}{\partial \vec{R}} = \left[\begin{array}{ccccc} 0 & 0 & \cdots & -1 \end{array} \right] \quad (3.35)$$

$$\frac{\partial F}{\partial \vec{V}} = \left[\begin{array}{ccccc} 0 & 0 & \cdots & 0 \end{array} \right] \quad (3.36)$$

$$\frac{\partial F}{\partial \vec{m}} = \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix} \quad (3.37)$$

$$\frac{\partial F}{\partial \vec{T}} = \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix} \quad (3.38)$$

$$\frac{\partial F}{\partial \tau_f} = 0 \quad (3.39)$$

Note that \vec{R} , \vec{V} , \vec{m} , and \vec{T} are column vectors. However, when finding a derivative with respect to one of these variables, it is convenient to treat them as row vectors.

Jacobian Matrix

The Jacobian is a matrix of the partial derivatives of the constraints with respect to the optimization variables. Most nonlinear program solvers can estimate the Jacobian numerically. However, the solver will run much faster if an analytical Jacobian is given. The Jacobian is given by:

$$C_{Jac} = \frac{\partial \vec{C}}{\partial \vec{x}_{opt}} \quad (3.40)$$

Note that, as defined, the constraint vector and the optimization vector are both column vectors. However, **for purposes of computing the Jacobian matrix, the optimization vector should be treated as a row vector**. See Appendix B for a review of vector differentiation rules used in this thesis.

The Jacobian can be thought of as a combined series of block matrices that represent the partial derivatives of the constraints with respect to the optimization variables (states, controls, and free times). The Jacobian for this problem is rather straightforward to compute.

$$C_{Jac} = \frac{d\vec{C}}{d\vec{x}_{opt}} = \begin{bmatrix} \frac{\partial \vec{C}_R}{\partial R} & \frac{\partial \vec{C}_R}{\partial V} & \frac{\partial \vec{C}_R}{\partial \vec{m}} & \frac{\partial \vec{C}_R}{\partial \vec{T}} & \frac{\partial \vec{C}_R}{\partial \tau_f} \\ \frac{\partial \vec{C}_Y}{\partial R} & \frac{\partial \vec{C}_Y}{\partial V} & \frac{\partial \vec{C}_Y}{\partial \vec{m}} & \frac{\partial \vec{C}_Y}{\partial \vec{T}} & \frac{\partial \vec{C}_Y}{\partial \tau_f} \\ \frac{\partial \vec{C}_{\vec{m}}}{\partial R} & \frac{\partial \vec{C}_{\vec{m}}}{\partial V} & \frac{\partial \vec{C}_{\vec{m}}}{\partial \vec{m}} & \frac{\partial \vec{C}_{\vec{m}}}{\partial \vec{T}} & \frac{\partial \vec{C}_{\vec{m}}}{\partial \tau_f} \end{bmatrix} \quad (3.41)$$

Even though the partial derivatives for this problem are relatively easy to find, they are derived in the following sections to help familiarize the reader with the matrix derivative concepts and notation used in this thesis.

Partial Derivatives of the \vec{R} Constraint

Equation 3.8 is rewritten below for reference.

$$\vec{C}_{\vec{R}} = D_{NN}\vec{R} - \vec{V} = 0$$

The partial derivative matrices are:

$$\frac{\partial \vec{C}_{\vec{R}}}{\partial \vec{R}} = D_{NN} \quad (3.42)$$

$$\frac{\partial \vec{C}_{\vec{R}}}{\partial \vec{V}} = \langle -1 \rangle \quad (3.43)$$

$$\frac{\partial \vec{C}_{\vec{R}}}{\partial \vec{m}} = [0] \quad (3.44)$$

$$\frac{\partial \vec{C}_{\vec{R}}}{\partial \vec{T}} = [0] \quad (3.45)$$

$$\frac{\partial \vec{C}_{\vec{R}}}{\partial \tau_f} = \left[\begin{array}{c} -D_{NN}\vec{R} \\ (\tau_f - \tau_o) \end{array} \right] \quad (3.46)$$

To make sense of the derivative with respect to the final time, recall equation 3.11. Note that the partial derivative of a column vector with respect to a scalar is also a column vector. Therefore, $\frac{\partial \vec{C}_{\vec{R}}}{\partial \tau_f}$ is a column vector.

Partial Derivatives of the \vec{V} Constraint

Equation 3.9 is rewritten below for reference.

$$\vec{C}_{\vec{V}} = D_{NN}\vec{V} - \frac{\vec{T}}{\vec{m}} + \frac{\vec{D}}{\vec{m}} + \vec{g} = 0$$

The partial derivative matrices are:

$$\frac{\partial \vec{C}_V}{\partial \vec{R}} = \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}}{\partial \vec{R}} + \frac{\partial \vec{g}}{\partial \vec{R}} \quad (3.47)$$

$$\frac{\partial \vec{C}_V}{\partial V} = D_{NN} + \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}}{\partial V} \quad (3.48)$$

$$\frac{\partial \vec{C}_V}{\partial \vec{m}} = \left\langle \frac{\vec{T} - \vec{D}}{\vec{m}^2} \right\rangle \quad (3.49)$$

$$\frac{\partial \vec{C}_V}{\partial \vec{T}} = - \left\langle \frac{1}{\vec{m}} \right\rangle \quad (3.50)$$

$$\frac{\partial \vec{C}_V}{\partial \tau_f} = \left[\frac{-D_{NN}\vec{V}}{(\tau_f - \tau_o)} \right] \quad (3.51)$$

Partial Derivatives of the \vec{m} Constraint

Equation 3.10 is rewritten below for reference.

$$\vec{C}_{\vec{m}} = D_{NN}\vec{m} + \frac{\vec{T}}{g_o I_{sp}} = 0$$

The partial derivative matrices are:

$$\frac{\partial \vec{C}_{\vec{m}}}{\partial \vec{R}} = [0] \quad (3.52)$$

$$\frac{\partial \vec{C}_{\vec{m}}}{\partial V} = [0] \quad (3.53)$$

$$\frac{\partial \vec{C}_{\vec{m}}}{\partial \vec{m}} = D_{NN} \quad (3.54)$$

$$\frac{\partial \vec{C}_{\vec{m}}}{\partial \vec{T}} = \left\langle \frac{1}{g_o I_{sp}} \right\rangle \quad (3.55)$$

$$\frac{\partial \vec{C}_{\vec{m}}}{\partial \tau_f} = \left[\frac{-D_{NN}\vec{m}}{(\tau_f - \tau_o)} \right] \quad (3.56)$$

Partial Derivatives of Drag

The partial derivative matrices for drag with respect to altitude and velocity are required. The drag is given by:

$$\vec{D} = \frac{1}{2} \vec{\rho} \vec{V}^2 C_D A_{ref} \quad (3.57)$$

The partial derivative matrices are:

$$\frac{\partial \vec{D}}{\partial \vec{R}} = \left\langle \frac{1}{2} \vec{V}^2 C_D A_{ref} \right\rangle \frac{\partial \vec{\rho}}{\partial \vec{R}} \quad (3.58)$$

$$\frac{\partial \vec{D}}{\partial \vec{V}} = \left\langle \vec{\rho} \vec{V} C_D A_{ref} \right\rangle \quad (3.59)$$

The exponential atmospheric density model used for this example is defined in Appendix C.1.

Partial Derivative of Gravity

The partial derivative matrix for gravitational acceleration with respect to radial position is required. The gravitational acceleration is:

$$\vec{g} = \frac{\mu}{(\vec{R})^2} \quad (3.60)$$

The partial derivative matrix is:

$$\frac{\partial \vec{g}}{\partial \vec{R}} = \left\langle \frac{-2\mu}{\vec{R}^3} \right\rangle \quad (3.61)$$

3.1.2 Numerical Results

Table 3.1 shows the numerical parameters used to define the launch vehicle. Note that these parameters are not from any specific, real launch vehicle. However, the values are physically possible. Figures 3-1, 3-2, 3-3, and 3-4 show the time histories of altitude, velocity, mass, and thrust, respectively.

Initial Total Mass (kg)	5000
Initial Propellant Mass over Initial Total Mass	0.6
Drag Coefficient	0.2
Aerodynamic Reference Area (m^2)	10
Specific Impulse of Engine (s)	300
Maximum Thrust to Initial Weight Ratio	2

Table 3.1: Numerical Values Used for Single-Stage Goddard Problem

It should be noted that the optimal solution to the single-stage Goddard problem can be found analytically from the calculus of variations. In fact, it was first solved in 1951 [5]. It is a nonlinear problem with a singular arc [5]. The concept of a singular arc is beyond the scope of this thesis. For a description of singular arcs and the analytic solution to the single-stage Goddard problem, see [5, pp. 381-401] or [19]. The control (in this case the thrust) has a known switching structure with three phases [5],[19]:

- **Phase 1**

In the first phase, the thrust is set at full throttle. This is known as a “constrained” or “full” arc because it is riding the maximum thrust limit.

- **Phase 2**

The second phase involves the singular arc, where the thrust is throttled according to a nonlinear feedback law. This is called a “singular” arc. In this phase, the singular arc “corresponds to thrust slightly larger than drag plus weight so that the rocket is accelerating upward but not wasting fuel to overcome the very large drag it would have encounter if maximum thrust had been used” [5].

- **Phase 3**

The third phase is a coast to maximum altitude. This is known as a “zero” arc because the thrust is zero.

Of concern is the fact that the numerical solution to a problem with a singular arc can be complicated [5]. The control can be discontinuous when switching between a

constrained or zero arc and a singular arc [5]. In fact, the optimal solution to the Goddard problem has a discontinuity in thrust when switching from the full throttle arc to the singular arc, and again when switching to the zero thrust arc [5],[19]. These discontinuities can cause problems for numerical solutions.

From Figure 3-4, it can be seen that the thrust starts at its maximum value. It then attempts to follow the singular arc before shutting off. The numerical optimizer seems to have a problem finding the singular arc. This can be seen in the jagged thrust profile during the singular arc phase. As discussed, this behavior is not surprising given the singular nature of the Goddard problem. While this is an interesting problem, the purpose of this section is only to introduce the Legendre Pseudospectral Method. Therefore, this phenomenon will not be discussed further in this thesis. However, it should be noted that a method for modelling discontinuities in states and controls is discussed in the next section. This method could be used to numerically model the thrust discontinuities in the Goddard problem.

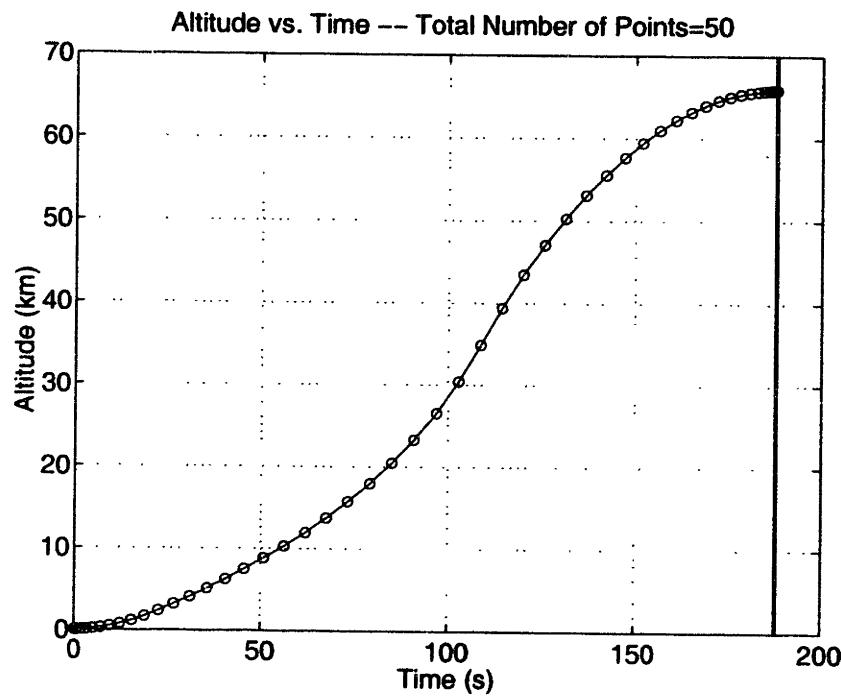


Figure 3-1: Single-Stage Goddard Problem – Time History of Altitude

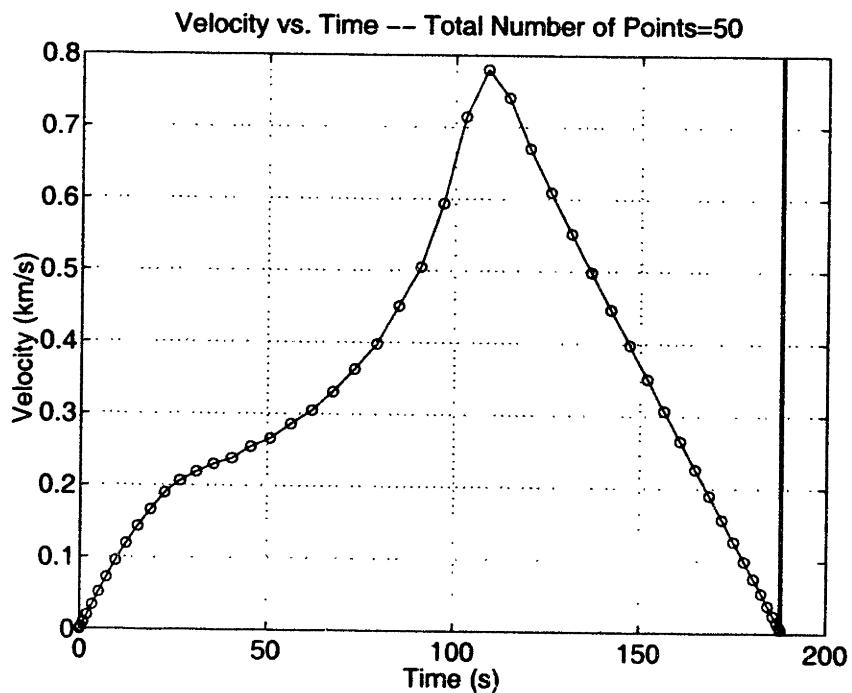


Figure 3-2: Single-Stage Goddard Problem – Time History of Velocity

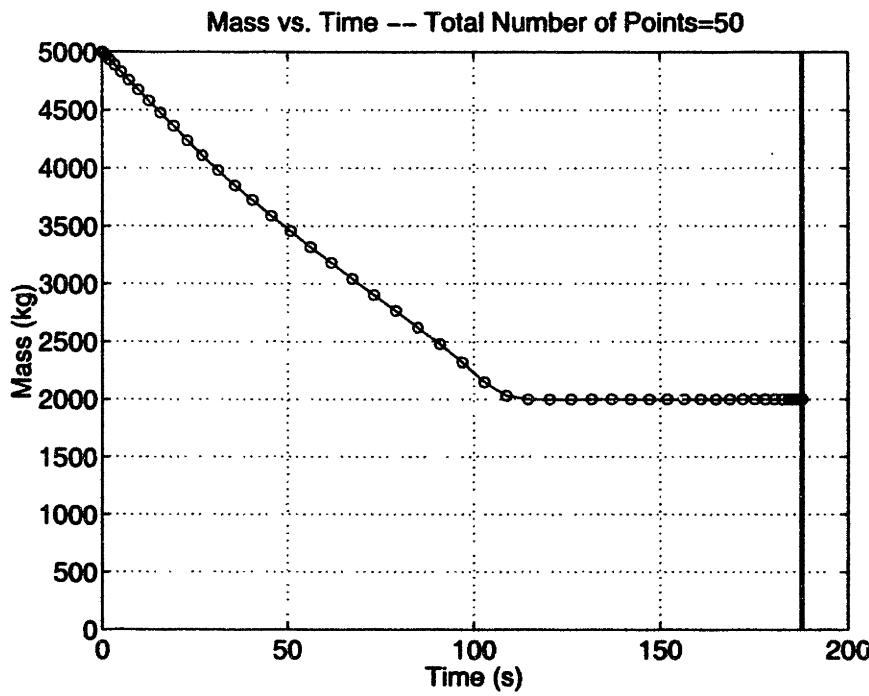


Figure 3-3: Single-Stage Goddard Problem – Time History of Mass

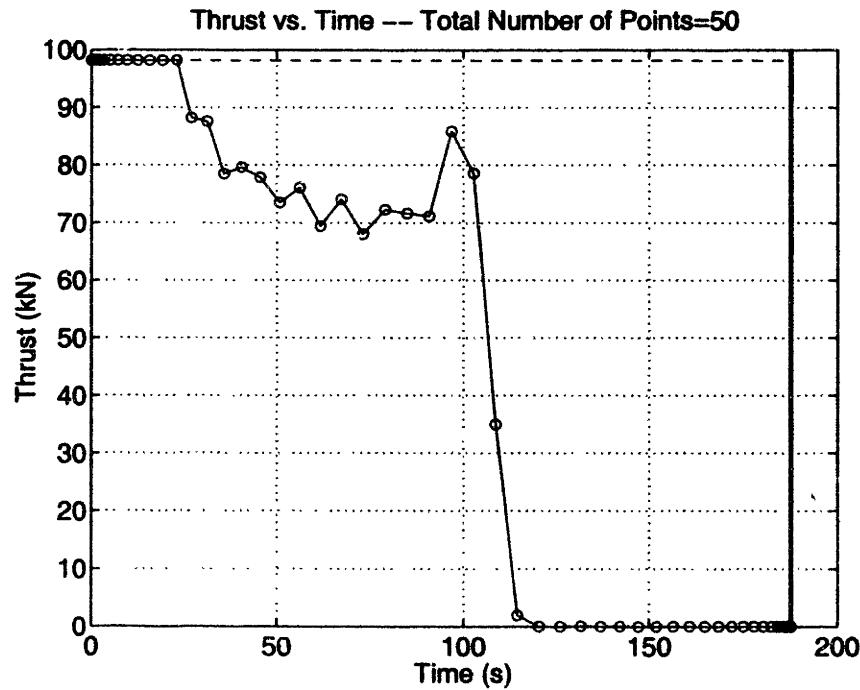


Figure 3-4: Single-Stage Goddard Problem – Time History of Thrust

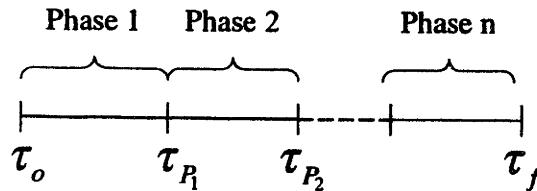


Figure 3-5: Total Time Domain Divided Into Phases

3.2 Staging

To date, there are no single-stage to orbit launch vehicles. Most vehicles have two or three stages. This causes a problem for the Legendre Pseudospectral Method described in the example above. While the D -matrix operates on discrete points, it requires that the function be continuous. The staged launch vehicle problem is discontinuous in mass. The following sections describe how the Legendre Pseudospectral Method can be applied to discontinuous systems. The concepts of phases, stages, and knots are introduced below.

Phases

The total time of launch can be divided into phases. Figure 3-5 shows this conceptually. The phases are connected by knots. There are two types of knots:

- A soft knot is one where the states are continuous across the knot. The controls may or may not be continuous.
- A hard knot is one where one or more of the states are discontinuous across the knot.

Each phase has its own number of LGL points and associated states, controls, and beginning and end times . The initial time, final time, and knotting times may be free or fixed. Each phase has its own set of dynamic constraints (i.e. equations of motion, mass flow rate equation) and trajectory constraints (i.e. dynamic pressure limits). All phases have boundary conditions. Constraints on the initial points are initial conditions. Constraints on the final point are final conditions. Constraints across the knotting points are knotting conditions.

The dynamic and trajectory constraints in a phase are functions of the state variables, control variables, and beginning and end times *for that phase only*. It is important to note that the phase constraints depend only on the current phase variables, and are independent of the variables from the other phases. The constraints for a given phase can be totally determined given only the states, controls, initial time, and final time for that phase.

For each phase, define a row vector made up of the states and controls associated with that phase.

$$\vec{x}_{P_i} = \begin{bmatrix} \vec{x}_{states_{P_i}} & \vec{x}_{controls_{P_i}} \end{bmatrix} \quad (3.62)$$

Also, define a row vector made up of all the free times of the problem. If all of the times are free, the time vector is:

$$\vec{x}_{times} = \begin{bmatrix} \tau_o & \tau_{P_1} & \tau_{P_2} & \cdots & \tau_{P_n} & \tau_f \end{bmatrix} \quad (3.63)$$

The total optimization vector for the nonlinear programming problem is:

$$\vec{x}_{opt} = \begin{bmatrix} \vec{x}_{P_1} & \vec{x}_{P_2} & \cdots & \vec{x}_{P_n} & \vec{x}_{times} \end{bmatrix}^T \quad (3.64)$$

Each phase will have a column vector of dynamic and trajectory constraints associated with it. Note that it is not required that each phase have the same constraints. The dynamic and trajectory constraint vector for a phase is:

$$\vec{C}_{P_i} = \text{vector of dynamic and trajectory constraints for phase } i \quad (3.65)$$

In addition to the dynamic and trajectory constraints, there can be constraints on the initial and final points. There could even be constraints on midpoints. Most of the time, the constraints will be on boundary points. Therefore, all of these constraints will be combined into a column vector called $\vec{C}_{boundary}$.

$$\vec{C}_{boundary} = \text{vector of boundary and point constraints for phase } i \quad (3.66)$$

Finally, there are the knotting constraints. These are the continuity and/or discontinuity constraints across the knots. Knotting constraints have the general form of:

$$(\vec{x}_{states_{P_i}})_{initial} - (\vec{x}_{states_{P_{i-1}}})_{final}$$

For example, let there be two phases. It is desired to put a knotting condition on velocity so that the velocity is continuous across the knot joining the two phases. Assume that each phase has 10 LGL points. The knotting constraint would be:

$$(V_1)_{P_2} - (V_{10})_{P_1} = 0$$

In words, this constraint says that the final velocity in phase one must be equal to the first velocity in phase two.

It is also necessary to put knotting constraints on the knotting times. This is to guarantee that the knotting time from a previous phase will always be less than the final time of the current phase. Since time always moves forward, it must be insured that:

$$\tau_o \leq \tau_{P_1} \leq \dots \leq \tau_{P_n} \leq \tau_f$$

This is done by writing the following constraints:

$$\begin{aligned} \tau_{P_1} - \tau_o &\geq 0 \\ \tau_{P_2} - \tau_{P_1} &\geq 0 \\ &\vdots \\ \tau_{P_n} - \tau_{P_{n-1}} &\geq 0 \\ \tau_f - \tau_{P_n} &\geq 0 \end{aligned}$$

The knotting constraints are combined into a column vector.

$$\vec{C}_{knot} = \text{vector of knotting constraints} \quad (3.67)$$

Now, all of the constraints are combined into the total constraint vector.

$$\vec{C} = \begin{bmatrix} \vec{C}_{P_1} \\ \vdots \\ \vec{C}_{P_n} \\ \vec{C}_{\text{boundary}} \\ \vec{C}_{\text{knot}} \end{bmatrix} \quad (3.68)$$

The Jacobian matrix is

$$C_{\text{Jac}} = \frac{\partial \vec{C}}{\partial \vec{x}_{\text{opt}}} = \begin{bmatrix} \frac{\partial \vec{C}_{P_1}}{\partial \vec{x}_{P_1}} & 0 & \dots & 0 & \frac{\partial \vec{C}_{P_1}}{\partial \vec{x}_{\text{times}}} \\ 0 & \frac{\partial \vec{C}_{P_2}}{\partial \vec{x}_{P_2}} & 0 & \vdots & \frac{\partial \vec{C}_{P_2}}{\partial \vec{x}_{\text{times}}} \\ \vdots & 0 & \ddots & 0 & \vdots \\ 0 & \dots & 0 & \frac{\partial \vec{C}_{P_n}}{\partial \vec{x}_{P_n}} & \frac{\partial \vec{C}_{P_n}}{\partial \vec{x}_{\text{times}}} \\ \frac{\partial \vec{C}_{\text{boundary}}}{\partial \vec{x}_{P_1}} & \frac{\partial \vec{C}_{\text{boundary}}}{\partial \vec{x}_{P_2}} & \dots & \frac{\partial \vec{C}_{\text{boundary}}}{\partial \vec{x}_{P_n}} & \frac{\partial \vec{C}_{\text{boundary}}}{\partial \vec{x}_{\text{times}}} \\ \frac{\partial \vec{C}_{\text{knot}}}{\partial \vec{x}_{P_1}} & \frac{\partial \vec{C}_{\text{knot}}}{\partial \vec{x}_{P_2}} & \dots & \frac{\partial \vec{C}_{\text{knot}}}{\partial \vec{x}_{P_n}} & \frac{\partial \vec{C}_{\text{knot}}}{\partial \vec{x}_{\text{times}}} \end{bmatrix} \quad (3.69)$$

This Jacobian form was selected to maximize code reuse. When all of the phases have the same dynamic and trajectory constraints, the same MATLAB function can be used to compute the Jacobian block ($\frac{\partial \vec{C}_{P_i}}{\partial \vec{x}_{P_i}}$) for each of the phases.

The next section gives an example of how staging is accomplished using the Legendre Pseudospectral Method. Note that the staging example is only one specialized case of the use of phases. The Legendre Pseudospectral Method can be applied to other systems with discontinuities.

3.3 Example: The Two-Stage Goddard Problem

3.3.1 The Problem

Find the optimal thrust profile that will maximize the final altitude of a two-stage launch vehicle when launched straight-up from the surface of the Earth. This problem shares the same assumptions, dynamics, and constraints as the single-stage problem.

The total time domain is divided into two phases. In this case, the knot is the staging point. The radial position and velocity are continuous across the knot, but the mass is discontinuous as the first stage is dropped.

Dynamic Constraints

The dynamic constraints from the single-stage problem (equations 3.8, 3.9, and 3.10) apply to both phases of the two-stage problem. Now, there are two sets of dynamic constraint vectors, one for each phase. The constraint vectors are combined into the main constraint vector (\vec{C}).

$$\vec{C}_{dynamic} = \begin{bmatrix} \vec{C}_{\dot{R}_1} \\ \vec{C}_{\dot{V}_1} \\ \vec{C}_{\dot{m}_1} \\ \vec{C}_{\dot{R}_2} \\ \vec{C}_{\dot{V}_2} \\ \vec{C}_{\dot{m}_2} \end{bmatrix} \quad (3.70)$$

Knotting Constraints

The radial position and the velocity must be continuous at the staging point. The mass must be discontinuous. Also, the final time must be greater than the staging time. The following conditions must be met at the staging point:

- The first radial position in the second phase $[(R_2)_{n=1}]$ must equal the last radial position in the first stage $[(R_1)_{n=n_{LGL_1}}]$.
- The first velocity in the second phase $[(V_2)_{n=1}]$ must equal the last velocity in the first stage $[(V_1)_{n=n_{LGL_1}}]$.
- The first mass in the second phase $[(m_2)_{n=1}]$ must be less than the last mass in the first stage $[(m_1)_{n=n_{LGL_1}}]$.
- The final time $[\tau_f]$ must be greater than the staging time $[\tau_s]$.

The following knotting constraints can be written:

$$C_{knot_R} = (R_2)_{n=1} - (R_1)_{n=n_{LGL_1}} = 0 \quad (3.71)$$

$$C_{knot_V} = (V_2)_{n=1} - (V_1)_{n=n_{LGL_1}} = 0 \quad (3.72)$$

$$C_{knot_m} = (m_2)_{n=1} - (m_1)_{n=n_{LGL_1}} < 0 \quad (3.73)$$

$$C_{knot_\tau} = \tau_f - \tau_s > \varepsilon > 0 \quad (3.74)$$

where ε is a very small positive number.

Note that the knotting constraint on mass (equation 3.73) is actually redundant as the discontinuous mass constraint can be accomplished by proper choice of the upper and lower bounds on the optimization vector. Note also that the knotting constraint on the staging and final times (equation 3.74) uses ε to insure that the times τ_f and τ_s are not equal. If they are equal, a singularity will occur in the numerical model. The knotting constraints are combined into one column vector.

$$\vec{C}_{knot} = \begin{bmatrix} C_{knot_R} & C_{knot_V} & C_{knot_m} & C_{knot_\tau} \end{bmatrix}^T \quad (3.75)$$

Constraint Vector and Constraint Vector Bounds

The total constraint vector is:

$$\vec{C} = \begin{bmatrix} \vec{C}_{dynamic} \\ \vec{C}_{knot} \end{bmatrix} \quad (3.76)$$

The upper and lower bounds on the dynamic constraints are zero. The following row vectors describe the bounds:

$$\vec{b}_{l_{dynamic}} = 0 \quad (3.77)$$

$$\vec{b}_{u_{dynamic}} = 0 \quad (3.78)$$

The upper and lower bounds on the knotting constraints are:

$$\vec{b}_{l_{knot}} = \begin{bmatrix} 0 & 0 & -\infty & \epsilon \end{bmatrix} \quad (3.79)$$

$$\vec{b}_{u_{knot}} = \begin{bmatrix} 0 & 0 & 0 & \infty \end{bmatrix} \quad (3.80)$$

The total constraint bound vectors are:

$$\vec{B}_{L_C} = \left[\vec{b}_{l_{dynamic}} \quad \vec{b}_{l_{knot}} \right]^T \quad (3.81)$$

$$\vec{B}_{U_C} = \left[\vec{b}_{u_{dynamic}} \quad \vec{b}_{u_{knot}} \right]^T \quad (3.82)$$

Optimization Vector

The optimization vector is made up of the states and controls at each of the LGL points, as well as any free times in the problem. Note that for the staged problem, there are two sets of LGL points, one for each phase. There are states and controls associated with each phase. The number of LGL points in phases one and two (n_{LGL_1} , n_{LGL_2}) may or may not be equal. The optimization vector for this problem is:

$$\vec{x}_{opt} = \left[\vec{R}_1^T \quad \vec{V}_1^T \quad \vec{m}_1^T \quad \vec{T}_1^T \quad \vec{R}_2^T \quad \vec{V}_2^T \quad \vec{m}_2^T \quad \vec{T}_2^T \quad \tau_s \quad \tau_f \right]^T \quad (3.83)$$

where

- \vec{R}_i = vector of radial positions associated with phase i
- \vec{V}_i = vector of velocities associated with phase i
- \vec{m}_i = vector of vehicle masses associated with phase i
- \vec{T}_i = vector of thrusts associated with phase i
- τ_s = the staging time
- τ_f = the final time

Optimization Vector Bounds

Upper and lower bounds must be placed on the values of the optimization vector.

- The initial radial position should be equal to the radius of the Earth (R_E). Also, the radial position should never go below the radius of the Earth. There is no upper limit for the radial position.
- The initial velocity should be equal to zero. The velocity is constrained to always be zero or positive. This insures that, during the time of interest, the vehicle only rises; it does not fall. There is no upper velocity limit.
- The initial mass must be equal to $m_1 + m_2$, where m_i is the total mass of stage i. During phase 1, the mass can never go below $m_1(1 - f_{p_1}) + m_2$ where f_{p_1} is the fraction of the first stage mass that is propellant. Also during phase 1, the mass can never be more than $m_1 + m_2$. At the staging point, the mass must be equal to m_2 , since the total mass of the first stage has been dropped. During phase 1, the mass can never go below $m_2(1 - f_{p_2})$ where f_{p_2} is the fraction of the second stage mass that is propellant. Also during phase 2, the mass can never be more than m_2 .
- The thrust can be directed upward or downward. The lower thrust limit is $-T_{max,i}$ and the upper thrust limit is $T_{max,i}$, where $T_{max,i}$ is the maximum thrust magnitude. The subscript on the maximum thrust refers to the stage number. This is to emphasize that the maximum thrusts for different stages do not have to be equal.
- The initial time is zero. The staging and final times are free.

The bounds are given by:

Lower Bounds

$$\vec{b}_{l_{R_1}} = \begin{bmatrix} R_E & R_E & \cdots & R_E \end{bmatrix} \quad (3.84)$$

$$\vec{b}_{l_{V_1}} = \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix} \quad (3.85)$$

$$\vec{b}_{l_{m_1}} = \begin{bmatrix} m_1 + m_2 & m_1(1 - f_{p_1}) + m_2 & \cdots & m_1(1 - f_{p_1}) + m_2 \end{bmatrix} \quad (3.86)$$

$$\vec{b}_{l_{T_1}} = \begin{bmatrix} -T_{\max_1} & -T_{\max_1} & \cdots & -T_{\max_1} \end{bmatrix} \quad (3.87)$$

$$\vec{b}_{l_{R_2}} = \begin{bmatrix} R_E & R_E & \cdots & R_E \end{bmatrix} \quad (3.88)$$

$$\vec{b}_{l_{V_2}} = \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix} \quad (3.89)$$

$$\vec{b}_{l_{m_2}} = \begin{bmatrix} m_2 & m_2(1 - f_{p_2}) & \cdots & m_2(1 - f_{p_2}) \end{bmatrix} \quad (3.90)$$

$$\vec{b}_{l_{T_2}} = \begin{bmatrix} -T_{\max_2} & -T_{\max_2} & \cdots & -T_{\max_2} \end{bmatrix} \quad (3.91)$$

$$b_{l_{r_s}} = 0 \quad (3.92)$$

$$b_{l_{r_f}} = 0 \quad (3.93)$$

The lower bounds are combined into the lower bound vector:

$$\vec{B}_{L_x} = \left[\vec{b}_{l_{R_1}} \quad \vec{b}_{l_{V_1}} \quad \vec{b}_{l_{m_1}} \quad \vec{b}_{l_{T_1}} \quad \vec{b}_{l_{R_2}} \quad \vec{b}_{l_{V_2}} \quad \vec{b}_{l_{m_2}} \quad \vec{b}_{l_{T_2}} \quad b_{l_{r_s}} \quad b_{l_{r_f}} \right]^T \quad (3.94)$$

Upper Bounds

$$\vec{b}_{u_{R_1}} = \begin{bmatrix} R_E & \infty & \cdots & \infty \end{bmatrix} \quad (3.95)$$

$$\vec{b}_{u_{V_1}} = \begin{bmatrix} 0 & \infty & \cdots & \infty \end{bmatrix} \quad (3.96)$$

$$\vec{b}_{u_{m_1}} = \begin{bmatrix} m_1 + m_2 & m_1 + m_2 & \cdots & m_1 + m_2 \end{bmatrix} \quad (3.97)$$

$$\vec{b}_{u_{T_1}} = \begin{bmatrix} T_{\max_1} & T_{\max_1} & \cdots & T_{\max_1} \end{bmatrix} \quad (3.98)$$

$$\vec{b}_{u_{R_2}} = \begin{bmatrix} \infty & \infty & \cdots & \infty \end{bmatrix} \quad (3.99)$$

$$\vec{b}_{u_{V_2}} = \begin{bmatrix} \infty & \infty & \cdots & \infty \end{bmatrix} \quad (3.100)$$

$$\vec{b}_{u_{m_2}} = \begin{bmatrix} m_2 & m_2 & \cdots & m_2 \end{bmatrix} \quad (3.101)$$

$$\vec{b}_{u_{T_2}} = \begin{bmatrix} T_{\max_2} & T_{\max_2} & \cdots & T_{\max_2} \end{bmatrix} \quad (3.102)$$

$$b_{u_{r_s}} = \infty \quad (3.103)$$

$$b_{u_{r_f}} = \infty \quad (3.104)$$

The upper bounds are combined into the upper bound vector:

$$\vec{B}_{U_x} = \left[\vec{b}_{u_{R_1}} \quad \vec{b}_{u_{V_1}} \quad \vec{b}_{u_{m_1}} \quad \vec{b}_{u_{T_1}} \quad \vec{b}_{u_{R_2}} \quad \vec{b}_{u_{V_2}} \quad \vec{b}_{u_{m_2}} \quad \vec{b}_{u_{T_2}} \quad b_{u_{r_s}} \quad b_{u_{r_f}} \right]^T \quad (3.105)$$

Nonlinear Programming Problem

The problem can now be posed as a nonlinear programming problem.

Minimize:

$$F = -R_{final_2}$$

Subject to:

$$\begin{aligned} \vec{B}(L_x) &\leq \vec{x}_{opt} \leq \vec{B}(U_x) \\ \vec{B}(L_C) &\leq \vec{C} \leq \vec{B}(U_C) \end{aligned}$$

This will maximize the final altitude while meeting the dynamic constraints at all of the LGL points. Note that R_{final_2} is the last radial position in the second phase. The objective gradient is very easy to find. The Jacobian matrix is a little different than

before.

$$C_{Jac} = \begin{bmatrix} \frac{\partial \tilde{C}_{R_1}}{\partial R_1} & \frac{\partial \tilde{C}_{R_1}}{\partial V_1} & \frac{\partial \tilde{C}_{R_1}}{\partial \tilde{m}_1} & \frac{\partial \tilde{C}_{R_1}}{\partial \tilde{T}_1} & 0 & 0 & 0 & 0 & \frac{\partial \tilde{C}_{R_1}}{\partial \tau_s} & 0 \\ \frac{\partial \tilde{C}_{V_1}}{\partial R_1} & \frac{\partial \tilde{C}_{V_1}}{\partial V_1} & \frac{\partial \tilde{C}_{V_1}}{\partial \tilde{m}_1} & \frac{\partial \tilde{C}_{V_1}}{\partial \tilde{T}_1} & 0 & 0 & 0 & 0 & \frac{\partial \tilde{C}_{V_1}}{\partial \tau_s} & 0 \\ \frac{\partial \tilde{C}_{\dot{m}_1}}{\partial R_1} & \frac{\partial \tilde{C}_{\dot{m}_1}}{\partial V_1} & \frac{\partial \tilde{C}_{\dot{m}_1}}{\partial \tilde{m}_1} & \frac{\partial \tilde{C}_{\dot{m}_1}}{\partial \tilde{T}_1} & 0 & 0 & 0 & 0 & \frac{\partial \tilde{C}_{\dot{m}_1}}{\partial \tau_s} & 0 \\ 0 & 0 & 0 & 0 & \frac{\partial \tilde{C}_{R_2}}{\partial R_2} & \frac{\partial \tilde{C}_{R_2}}{\partial V_2} & \frac{\partial \tilde{C}_{R_2}}{\partial \tilde{m}_2} & \frac{\partial \tilde{C}_{R_2}}{\partial \tilde{T}_2} & \frac{\partial \tilde{C}_{R_2}}{\partial \tau_s} & \frac{\partial \tilde{C}_{R_2}}{\partial \tau_f} \\ 0 & 0 & 0 & 0 & \frac{\partial \tilde{C}_{V_2}}{\partial R_2} & \frac{\partial \tilde{C}_{V_2}}{\partial V_2} & \frac{\partial \tilde{C}_{V_2}}{\partial \tilde{m}_2} & \frac{\partial \tilde{C}_{V_2}}{\partial \tilde{T}_2} & \frac{\partial \tilde{C}_{V_2}}{\partial \tau_s} & \frac{\partial \tilde{C}_{V_2}}{\partial \tau_f} \\ 0 & 0 & 0 & 0 & \frac{\partial \tilde{C}_{\dot{m}_2}}{\partial R_2} & \frac{\partial \tilde{C}_{\dot{m}_2}}{\partial V_2} & \frac{\partial \tilde{C}_{\dot{m}_2}}{\partial \tilde{m}_2} & \frac{\partial \tilde{C}_{\dot{m}_2}}{\partial \tilde{T}_2} & \frac{\partial \tilde{C}_{\dot{m}_2}}{\partial \tau_s} & \frac{\partial \tilde{C}_{\dot{m}_2}}{\partial \tau_f} \\ \frac{\partial \tilde{C}_{knot}}{\partial R_1} & \frac{\partial \tilde{C}_{knot}}{\partial V_1} & \frac{\partial \tilde{C}_{knot}}{\partial \tilde{m}_1} & \frac{\partial \tilde{C}_{knot}}{\partial \tilde{T}_1} & \frac{\partial \tilde{C}_{knot}}{\partial R_2} & \frac{\partial \tilde{C}_{knot}}{\partial V_2} & \frac{\partial \tilde{C}_{knot}}{\partial \tilde{m}_2} & \frac{\partial \tilde{C}_{knot}}{\partial \tilde{T}_2} & \frac{\partial \tilde{C}_{knot}}{\partial \tau_s} & \frac{\partial \tilde{C}_{knot}}{\partial \tau_f} \end{bmatrix}$$

Most of these partial derivatives have already been defined for the single-stage problem. Note that the dynamic constraints of the first phase only interact with τ_o (which is not an optimization variable) and τ_s (which acts like the “final” time for the first phase). The partial derivatives of the constraints with respect to the final time have already been defined for the single stage problem.

The second phase dynamic constraints interact with τ_s (which is the “initial” time for the second phase) and τ_f (which is the final time for the second phase). The partial derivative with respect to the final time has already been defined. It is convenient that the partial derivative with respect to the “initial” time for a phase is equal to the negative of the partial derivative with respect to the final time for a phase.

So,

$$\frac{\partial \tilde{C}_{dynamic}}{\partial \tau_o} = -\frac{\partial \tilde{C}_{dynamic}}{\partial \tau_f} \quad (3.106)$$

To see that this is true, examine any of equations 3.46, 3.51, and 3.56.

3.3.2 Numerical Results

Table 3.2 shows the numerical parameters used to define the launch vehicle. Note that these parameters are not from any specific, real launch vehicle. However, the values are physically possible. Note that the same values for the aerodynamic reference area,

drag coefficient, and maximum thrust are used for both stages. Each stage has the same mass fraction of propellant.

Figures 3-6, 3-7, 3-8, and 3-9 show the time histories of altitude, velocity, mass, and thrust, respectively. The solid, vertical lines indicate the staging point and the final point, respectively. For these results, 20 points are used for the first stage, and 30 points are used for the second stage.

Note from figure 3-8 that the mass is discontinuous. This discontinuity is the staging point where the first stage is dropped. A large mass change between stages was chosen to emphasize the mass discontinuity. This is just one example of how discontinuities in states and controls can be modelled with the Legendre Pseudospectral Method.

Note from figure 3-9 that the numerical optimizer seems to have trouble tracking the singular arc that defines the optimal thrust throttling profile. As discussed for the single-stage Goddard problem, this is expected behavior for a numerical solution to a singular problem. See section 3.1.2 for the discussion on singular arcs and the Goddard problem.

Initial Total Mass (kg)	5000
Initial Mass of First Stage (kg)	3000
Initial Mass of Second Stage (kg)	2000
Initial Propellant Mass over Initial Total Mass	0.6
Drag Coefficient	0.2
Aerodynamic Reference Area (m^2)	10
Specific Impulse of Engine (s)	300
Maximum Thrust to Initial Weight Ratio	2

Table 3.2: Numerical Values Used for Two-Stage Goddard Problem

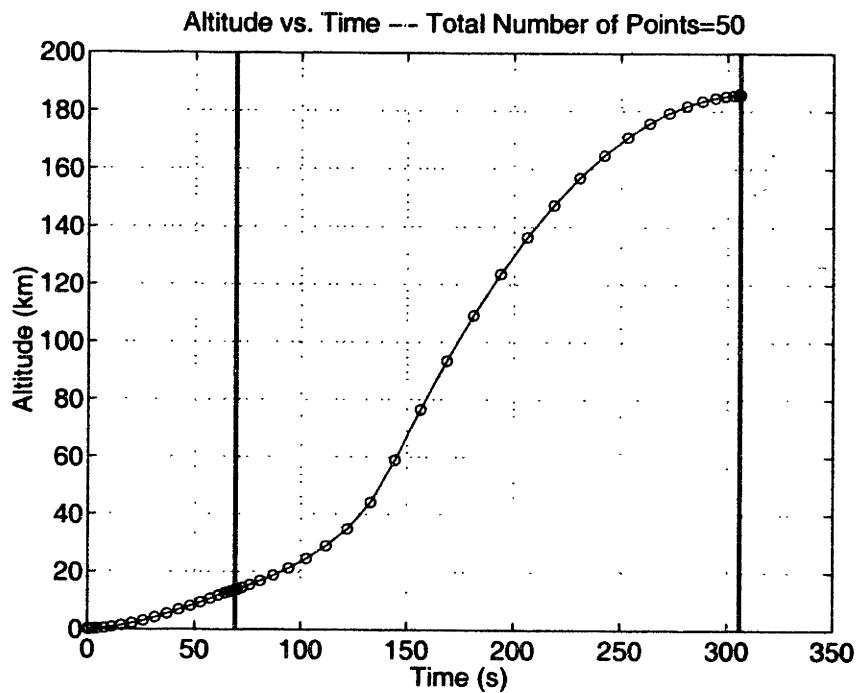


Figure 3-6: Two-Stage Goddard Problem – Time History of Altitude

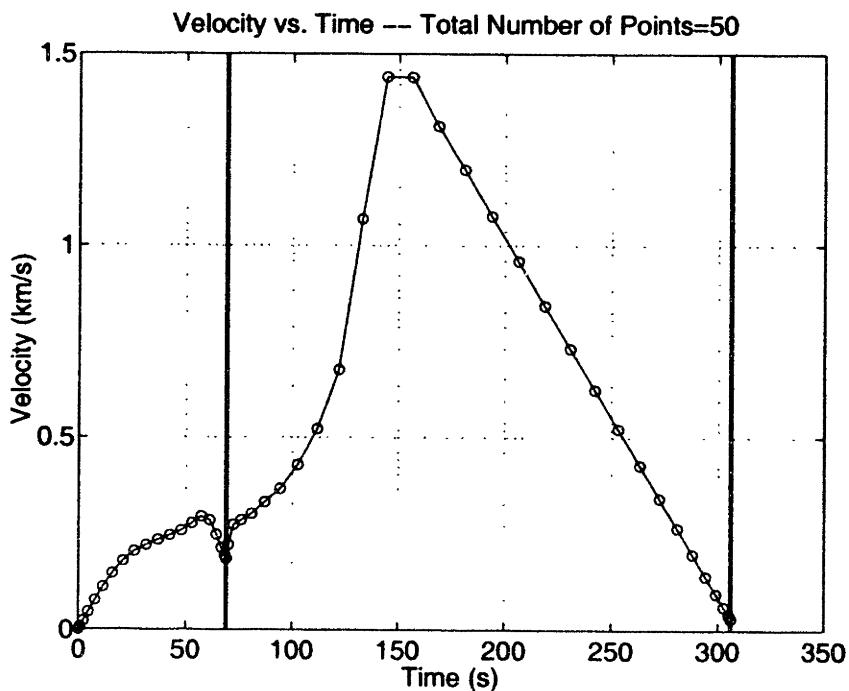


Figure 3-7: Two-Stage Goddard Problem – Time History of Velocity

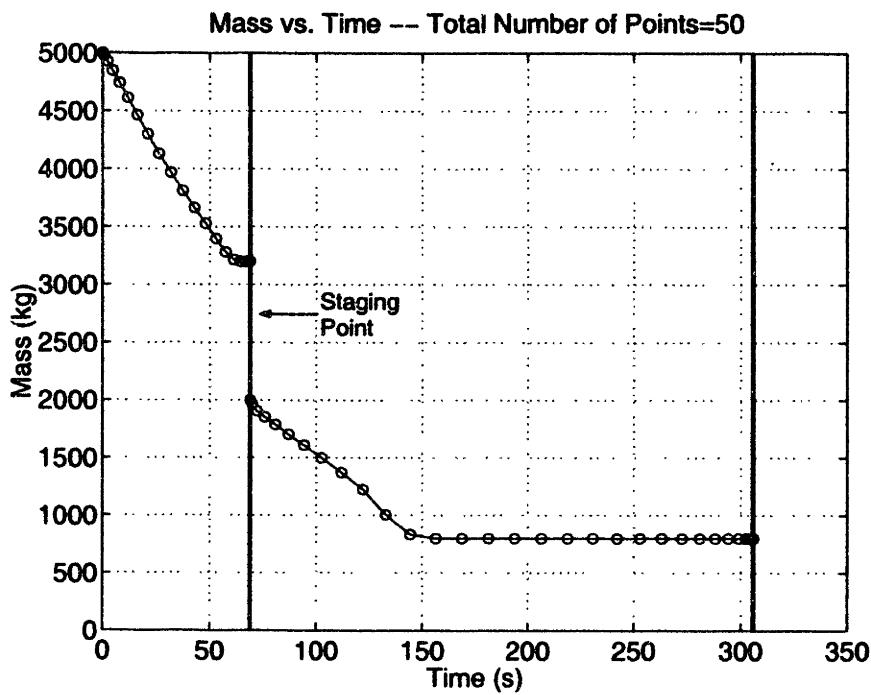


Figure 3-8: Two-Stage Goddard Problem – Time History of Mass

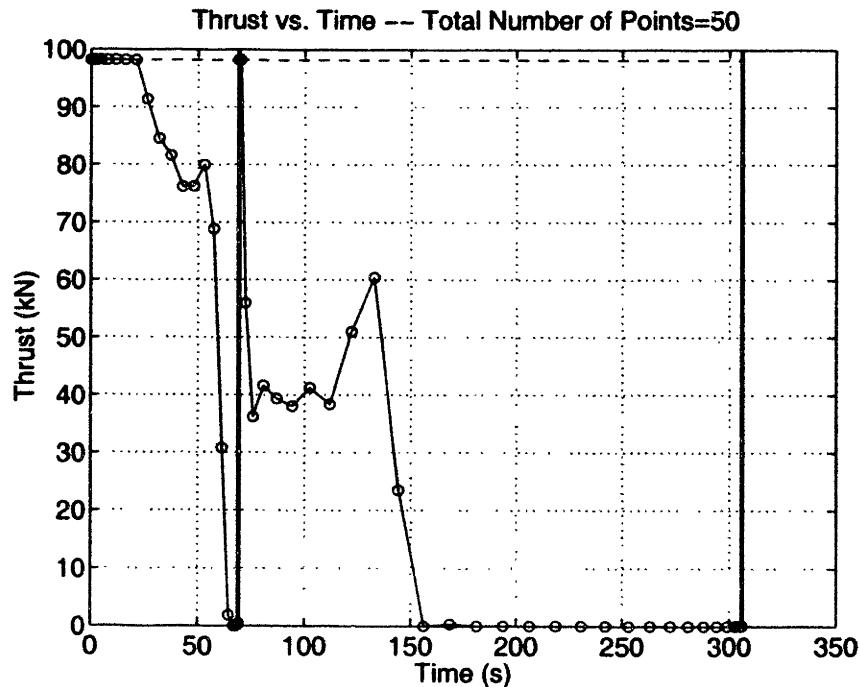


Figure 3-9: Two-Stage Goddard Problem – Time History of Thrust

[This page intentionally left blank.]

Chapter 4

Coordinate System Comparison

It is desired to find a good coordinate system for the problem of a rocket ascending from the surface of the Earth. What exactly is meant by a “good” coordinate system? Any coordinate system can be used to represent this problem. However, from a numerical perspective, it is possible that some coordinate systems will work better than others. The choice of a proper coordinate system can simplify the problem. In analytical terms, this could mean finding a coordinate system that has simpler expressions than other coordinate systems. In regards to nonlinear programming, the choice of proper coordinate system can result in fewer constraints and possibly even fewer optimization variables. For the problem at hand, speed of convergence and robustness are desired. Therefore, a “better” coordinate system is one that converges faster than others.

In order to make a comparison, a two-dimensional rocket ascent problem is posed in three different coordinate systems. Each coordinate system uses the same physical model. The coordinate systems are:

1. Cartesian
2. Radial-Transverse Polar
3. Normal-Tangential Polar

Simplified Test Problem

It is desired to model the flight of a rocket from the surface of the Earth to a circular target orbit. The objective is to minimize the amount of propellant used to achieve the target orbit. The objective function is:

$$F = \frac{m_o - m_f}{m_o} \quad (4.1)$$

where

m_f = final mass

m_o = initial mass

The following simplifying assumptions are made:

1. Two dimensional problem
2. Earth is a perfect sphere
3. Earth is non-rotating
4. The atmospheric density is an exponential function of altitude
5. The vehicle has a constant drag coefficient, with no lift coefficient
6. The rocket engine is perfectly expanded for all atmospheric pressures
7. The vehicle flies single stage to a circular target orbit

4.1 Cartesian Coordinate System

Equations of Motion

Figure 4-1 shows the relationship between the forces in relation to the x- and y-directions (\hat{e}_x and \hat{e}_y). The sum of all forces can be written along the x- and y-directions.

$$\Sigma F_x = T_x + D_x + mg_x = ma_x = m\dot{V}_x = m\ddot{R}_x \quad (4.2)$$

$$\Sigma F_y = T_y + D_y + mg_y = ma_y = m\dot{V}_y = m\ddot{R}_y \quad (4.3)$$

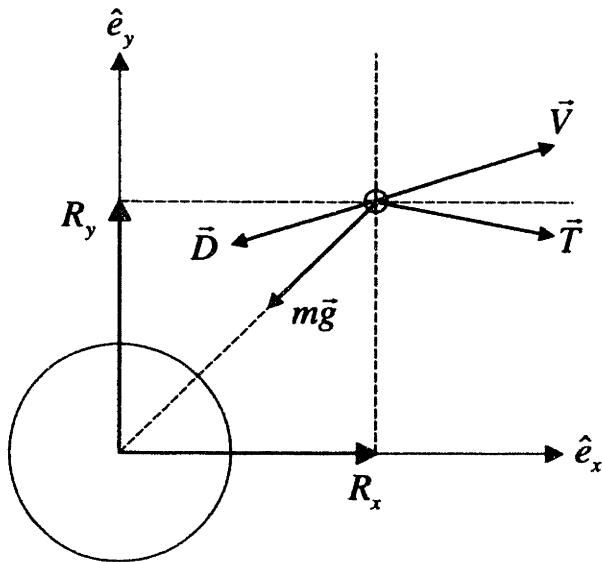


Figure 4-1: Cartesian Coordinates – Forces on a Point Mass

where

- T_x = X-thrust
- T_y = Y-thrust
- D_x = X-drag
- D_y = Y-drag
- g_x = X-gravity
- g_y = Y-gravity
- a_x = X-acceleration
- a_y = Y-acceleration
- V_x = X-velocity
- V_y = Y-velocity
- R_x = X-position
- R_y = Y-position
- m = mass

Thrust

For a perfectly expanded rocket engine, the thrust magnitude and mass flow rate are related by:

$$T = (T_x^2 + T_y^2)^{1/2} = -\dot{m}V_{exit} = -\dot{m}g_o I_{sp} \quad (4.4)$$

where

- T = thrust magnitude
- \dot{m} = mass flow rate
- V_{exit} = engine exit velocity
- g_o = gravitational acceleration at sea level
- I_{sp} = specific impulse of rocket engine

Dynamic Equations

The following dynamic equations can be written:

$$\dot{R}_x = V_x$$

$$\dot{R}_y = V_y$$

$$\begin{aligned}\dot{V}_x &= \frac{T_x}{m} + \frac{D_x}{m} + g_x \\ \dot{V}_y &= \frac{T_y}{m} + \frac{D_y}{m} + g_y \\ \dot{m} &= -\frac{(T_x^2 + T_y^2)^{1/2}}{g_o I_{sp}}\end{aligned} \quad (4.5)$$

Drag

The drag is defined in terms of the dynamic pressure, velocity, constant drag coefficient, and aerodynamic reference area. Note that the drag acts in the opposite direction of the velocity.

$$\begin{aligned}\vec{D}_x &= \frac{1}{2} \vec{\rho} (\vec{V}_x^2 + \vec{V}_y^2) C_D A_{ref} \frac{-\vec{V}_x}{(\vec{V}_x^2 + \vec{V}_y^2)^{1/2}} = -\frac{1}{2} \vec{\rho} \vec{V}_x (\vec{V}_x^2 + \vec{V}_y^2)^{1/2} C_D A_{ref} \\ \vec{D}_y &= \frac{1}{2} \vec{\rho} (\vec{V}_x^2 + \vec{V}_y^2) C_D A_{ref} \frac{-\vec{V}_y}{(\vec{V}_x^2 + \vec{V}_y^2)^{1/2}} = -\frac{1}{2} \vec{\rho} \vec{V}_y (\vec{V}_x^2 + \vec{V}_y^2)^{1/2} C_D A_{ref}\end{aligned} \quad (4.6)$$

Dynamic Pressure

The dynamic pressure is given by:

$$\vec{q} = \frac{1}{2} \vec{\rho} (\vec{V}_x^2 + \vec{V}_y^2) \quad (4.7)$$

Atmospheric Density

See Appendix C.1 for the exponential atmospheric density model used.

Gravity

The gravitational acceleration is:

$$\begin{aligned}\vec{g}_x &= \frac{\mu}{(\vec{R}_x^2 + \vec{R}_y^2)} \frac{-\vec{R}_x}{(\vec{R}_x^2 + \vec{R}_y^2)^{1/2}} = \frac{-\mu \vec{R}_x}{(\vec{R}_x^2 + \vec{R}_y^2)^{3/2}} \\ \vec{g}_y &= \frac{\mu}{(\vec{R}_x^2 + \vec{R}_y^2)} \frac{-\vec{R}_y}{(\vec{R}_x^2 + \vec{R}_y^2)^{1/2}} = \frac{-\mu \vec{R}_y}{(\vec{R}_x^2 + \vec{R}_y^2)^{3/2}}\end{aligned} \quad (4.8)$$

Trajectory Constraint Conditions

Limits are placed on the maximum thrust magnitude, maximum dynamic pressure, and sensed acceleration magnitude.

$$T_x^2 + T_y^2 \leq T_{\max}^2 \quad (4.9)$$

$$q \leq q_{\max} \quad (4.10)$$

$$a_{\text{sensed}}^2 \leq (a_{\text{sensed}})_{\max}^2 \quad (4.11)$$

The sensed acceleration is defined as:

$$\begin{aligned}\vec{a}_{\text{sensed}_x} &= \frac{\vec{T}_x + \vec{D}_x}{m} \\ \vec{a}_{\text{sensed}_y} &= \frac{\vec{T}_y + \vec{D}_y}{m} \\ (\vec{a}_{\text{sensed}})_{\text{mag}}^2 &= \vec{a}_{\text{sensed}_x}^2 + \vec{a}_{\text{sensed}_y}^2\end{aligned} \quad (4.12)$$

Burnout Constraint Conditions

It is desired to reach a circular target orbit with a radius of R_{mag_f} . The orbital speed for a circular orbit is [2]:

$$V_{circular} = \sqrt{\frac{\mu}{R}} \quad (4.13)$$

where

μ = gravitational parameter of the planet

R = radius of the circular orbit

The following burnout constraints are used:

- Constraint on the final radial position

$$C_{R_f} = R_{x_f}^2 + R_{y_f}^2 = R_{mag_f}^2 \quad (4.14)$$

- Constraint on the final velocity magnitude

$$C_{V_f} = V_{x_f}^2 + V_{y_f}^2 = V_{mag_f}^2 \quad (4.15)$$

- The final position and velocity vectors are constrained to be perpendicular.

$$C_{\vec{R}_f \cdot \vec{V}_f} = \vec{R}_f \cdot \vec{V}_f = R_{x_f} V_{x_f} + R_{y_f} V_{y_f} = 0 \quad (4.16)$$

Constraints

After transforming the time domain and using the D -matrix to change the derivatives into algebraic expressions, the constraints can be written as follows:

$$\vec{C}_{\vec{R}_x} = D_{NN} \vec{R}_x - \vec{V}_x = 0 \quad (4.17)$$

$$\vec{C}_{\vec{R}_y} = D_{NN} \vec{R}_y - \vec{V}_y = 0 \quad (4.18)$$

$$\vec{C}_{\vec{V}_x} = D_{NN} \vec{V}_x - \frac{\vec{T}_x}{\vec{m}} - \frac{\vec{D}_x}{\vec{m}} - \vec{g}_x = 0 \quad (4.19)$$

$$\vec{C}_{\vec{V}_y} = D_{NN} \vec{V}_y - \frac{\vec{T}_y}{\vec{m}} - \frac{\vec{D}_y}{\vec{m}} - \vec{g}_y = 0 \quad (4.20)$$

$$\vec{C}_{\vec{m}} = D_{NN} \vec{m} + \frac{(\vec{T}_x^2 + \vec{T}_y^2)^{1/2}}{g_o I_{sp}} = 0 \quad (4.21)$$

$$\vec{C}_T = \vec{T}_x^2 + \vec{T}_y^2 \leq T_{\max}^2 \quad (4.22)$$

$$\vec{C}_q = \vec{q} \leq q_{\max} \quad (4.23)$$

$$\vec{C}_a = \vec{a}_{mag_s}^2 \leq a_{\max_s}^2 \quad (4.24)$$

$$C_{R_f} = R_{x_f}^2 + R_{y_f}^2 = R_{mag_f}^2 \quad (4.25)$$

$$C_{V_f} = V_{x_f}^2 + V_{y_f}^2 = V_{mag_f}^2 \quad (4.26)$$

$$C_{\vec{R}, \vec{V}} = R_{x_f} V_{x_f} + R_{y_f} V_{y_f} = 0 \quad (4.27)$$

Optimization Vector

$$\vec{x}_{opt} = \begin{bmatrix} \vec{R}_x \\ \vec{R}_y \\ \vec{V}_x \\ \vec{V}_y \\ \vec{m} \\ \vec{T}_x \\ \vec{T}_y \\ \tau_f \end{bmatrix} \quad (4.28)$$

where τ_f is the final time.

Optimization Vector Bounds

The bounds on the optimization variables are:

$$\begin{aligned}
 -\infty &\leq \vec{R}_x \leq \infty \\
 -\infty &\leq \vec{R}_y \leq \infty \\
 -\infty &\leq \vec{V}_x \leq \infty \\
 -\infty &\leq \vec{V}_y \leq \infty \\
 (1-f_p)m_o &\leq \vec{m} \leq m_o \\
 -T_{\max} &\leq \vec{T}_x \leq T_{\max} \\
 -T_{\max} &\leq \vec{T}_y \leq T_{\max} \\
 0 &\leq \tau_f \leq \infty
 \end{aligned} \tag{4.29}$$

The exception to these bounds is at the initial point. The initial states and controls are defined (not allowed to vary) by setting the initial upper and lower bounds to the same values.

Jacobian

The Jacobian for the cartesian coordinate system is defined in Appendix F.1.

4.2 Radial-Transverse Polar Coordinates

Equations of Motion

Figure 4-2 shows the relationship between the forces in relation to the radial and transverse directions (\hat{e}_R and \hat{e}_θ). The radial direction is in the direction of a vector drawn from the planet center to the current position. The transverse direction is perpendicular to the radial direction. The angle θ is used to specify the angular position of the radial vector. It is measured from an arbitrary reference line. Note that the transverse direction is positive in the positive direction of theta. The sum of all forces can be written along the radial and transverse directions.

$$\Sigma F_R = T_R + D_R - mg = ma_R$$

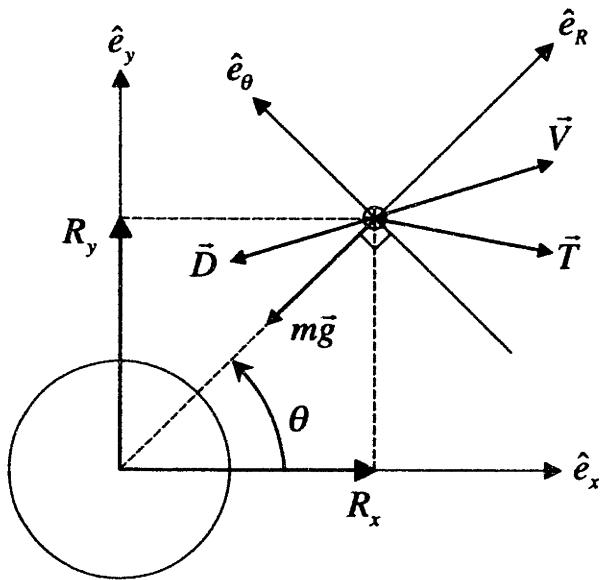


Figure 4-2: Radial-Transverse Polar Coordinates – Forces on a Point Mass

$$\Sigma F_\theta = T_\theta + D_\theta = m a_\theta \quad (4.30)$$

where

- T_R = radial thrust
- T_θ = transverse thrust
- D_R = radial drag
- D_θ = transverse drag
- g = gravity
- a_R = radial acceleration
- a_θ = transverse acceleration
- m = mass

Note that gravity now acts only in the negative radial direction. The velocities in the radial and transverse directions are [13, p. 68]

$$\begin{aligned} V_R &= \dot{R} \\ V_\theta &= R \dot{\theta} \end{aligned} \quad (4.31)$$

where

$$\begin{aligned} R &= \text{radial position} \\ \theta &= \text{angular position} \end{aligned}$$

The accelerations in the radial and transverse directions are given below [13, p. 68]:

$$\begin{aligned} a_R &= \ddot{R} - R\dot{\theta}^2 = \dot{V}_R - \frac{V_\theta^2}{R} \\ a_\theta &= R\ddot{\theta} + 2\dot{R}\dot{\theta} = \dot{V}_\theta + \frac{V_R V_\theta}{R} \end{aligned} \quad (4.32)$$

Equations 4.32 can be combined with equations 4.30:

$$\begin{aligned} T_R + D_R - mg &= m \left(\dot{V}_R - \frac{V_\theta^2}{R} \right) \\ T_\theta + D_\theta &= m \left(\dot{V}_\theta + \frac{V_R V_\theta}{R} \right) \end{aligned} \quad (4.33)$$

Thrust

For a perfectly expanded rocket engine, the thrust magnitude and mass flow rate are related by:

$$T = (T_R^2 + T_\theta^2)^{1/2} = -\dot{m}V_{exit} = -\dot{m}g_o I_{sp} \quad (4.34)$$

where

$$\begin{aligned} T &= \text{thrust magnitude} \\ \dot{m} &= \text{mass flow rate} \\ V_{exit} &= \text{engine exit velocity} \\ g_o &= \text{gravitational acceleration at sea level} \\ I_{sp} &= \text{specific impulse of rocket engine} \end{aligned}$$

Dynamic Equations

The following dynamic equations can be written:

$$\begin{aligned} \dot{R} &= V_R \\ \dot{\theta} &= \frac{V_\theta}{R} \\ \dot{V}_R &= \frac{T_R}{m} + \frac{D_R}{m} - g + \frac{V_\theta^2}{R} \end{aligned}$$

$$\begin{aligned}\dot{V}_\theta &= \frac{T_\theta}{m} + \frac{D_\theta}{m} - \frac{V_R V_\theta}{R} \\ \dot{m} &= -\frac{(T_R^2 + T_\theta^2)^{1/2}}{g_o I_{sp}}\end{aligned}\quad (4.35)$$

Drag

The drag is defined in terms of the dynamic pressure, velocity, constant drag coefficient, and aerodynamic reference area. Note that the drag acts in the opposite direction of the velocity:

$$\begin{aligned}\vec{D}_R &= \frac{1}{2} \vec{\rho} (\vec{V}_R^2 + \vec{V}_\theta^2) C_D A_{ref} \frac{-\vec{V}_R}{(\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2}} = -\frac{1}{2} \vec{\rho} \vec{V}_R (\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2} C_D A_{ref} \\ \vec{D}_\theta &= \frac{1}{2} \vec{\rho} (\vec{V}_R^2 + \vec{V}_\theta^2) C_D A_{ref} \frac{-\vec{V}_\theta}{(\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2}} = -\frac{1}{2} \vec{\rho} \vec{V}_\theta (\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2} C_D A_{ref}\end{aligned}\quad (4.36)$$

Dynamic Pressure

The dynamic pressure is given by:

$$\vec{q} = \frac{1}{2} \vec{\rho} (\vec{V}_R^2 + \vec{V}_\theta^2) \quad (4.37)$$

Atmospheric Density

See Appendix C.1 for the exponential atmospheric density model used.

Gravity

The magnitude of the gravitational acceleration is:

$$\vec{g} = \frac{\mu}{\vec{R}^2} \quad (4.38)$$

Trajectory Constraint Conditions

Limits are placed on the maximum thrust magnitude, maximum dynamic pressure, and sensed acceleration magnitude.

$$T_R^2 + T_\theta^2 \leq T_{\max}^2 \quad (4.39)$$

$$q \leq q_{\max} \quad (4.40)$$

$$a_{\text{sensed}}^2 \leq (a_{\text{sensed}})_{\max}^2 \quad (4.41)$$

The sensed acceleration is given by:

$$\begin{aligned}\vec{a}_{\text{sensed}_R} &= \frac{\vec{T}_R + \vec{D}_R}{\bar{m}} \\ \vec{a}_{\text{sensed}_\theta} &= \frac{\vec{T}_\theta + \vec{D}_\theta}{\bar{m}} \\ (\vec{a}_{\text{sensed}})_{\text{mag}}^2 &= \vec{a}_{\text{sensed}_R}^2 + \vec{a}_{\text{sensed}_\theta}^2\end{aligned}\quad (4.42)$$

Burnout Constraint Conditions

It is desired to reach a circular target orbit.

$$R_f = R_{f_{\text{target}}} \quad (4.43)$$

$$V_{R_f} = 0 \quad (4.44)$$

$$V_{\theta_f} = V_{\text{circ}_{\text{target}}} \quad (4.45)$$

Note that these burnout conditions are physically identical to those for the cartesian coordinates, but they are much simpler to write. In fact, these constraints are so simple that they can be written as simple bounds on the final state optimization variables.

Constraints

After transforming the time domain and using the D -matrix to change the derivatives into algebraic expressions, the constraints can be written as follows:

$$\vec{C}_{\dot{R}} = D_{NN} \vec{R} - \vec{V}_R = 0 \quad (4.46)$$

$$\vec{C}_\dot{\theta} = D_{NN} \vec{\theta} - \frac{\vec{V}_\theta}{\vec{R}} = 0 \quad (4.47)$$

$$\vec{C}_{\dot{V}_R} = D_{NN} \vec{V}_R - \frac{\vec{T}_R}{\bar{m}} - \frac{\vec{D}_R}{\bar{m}} + \vec{g} - \frac{\vec{V}_\theta^2}{\vec{R}} = 0 \quad (4.48)$$

$$\vec{C}_{\dot{V}_\theta} = D_{NN} \vec{V}_\theta - \frac{\vec{T}_\theta}{\bar{m}} - \frac{\vec{D}_\theta}{\bar{m}} + \frac{\vec{V}_R \vec{V}_\theta}{\vec{R}} = 0 \quad (4.49)$$

$$\vec{C}_{\dot{m}} = D_{NN}\vec{m} + \frac{(\vec{T}_R^2 + \vec{T}_\theta^2)^{1/2}}{g_o I_{sp}} = 0 \quad (4.50)$$

$$\vec{C}_{T_{\max}} = \vec{T}_R^2 + \vec{T}_\theta^2 \leq T_{\max}^2 \quad (4.51)$$

$$\vec{C}_q = \vec{q} \leq q_{\max} \quad (4.52)$$

$$\vec{C}_a = \vec{a}_{mag_s}^2 \leq a_{\max}^2, \quad (4.53)$$

Optimization Vector

$$\vec{x}_{opt} = \begin{bmatrix} \vec{R} \\ \vec{\theta} \\ \vec{V}_R \\ \vec{V}_\theta \\ \vec{m} \\ \vec{T}_R \\ \vec{T}_\theta \\ \tau_f \end{bmatrix} \quad (4.54)$$

where τ_f is the final time.

Optimization Vector Bounds

The bounds on the optimization variables are:

$$\begin{aligned}
 0 &\leq \vec{R} \leq \infty \\
 -\infty &\leq \vec{\theta} \leq \infty \\
 -\infty &\leq \vec{V}_R \leq \infty \\
 -\infty &\leq \vec{V}_\theta \leq \infty \\
 (1 - f_p)m_o &\leq \vec{m} \leq m_o \\
 -T_{\max} &\leq \vec{T}_R \leq T_{\max} \\
 -T_{\max} &\leq \vec{T}_T \leq T_{\max} \\
 0 &\leq \tau_f \leq \infty
 \end{aligned} \tag{4.55}$$

The exception to these bounds is at the initial point. The initial states and controls are defined (not allowed to vary) by setting the initial upper and lower bounds to the same values.

Jacobian

The Jacobian for the radial-transverse polar coordinate system is defined in Appendix F.2.

4.3 Normal-Tangential Polar Coordinates

Equations of Motion

Figure 4-3 shows the relationship between the forces in relation to the normal and tangential directions (\hat{e}_γ and \hat{e}_V). The tangential direction is in the same direction as the velocity. The normal direction is perpendicular to the velocity. As the figure shows, the flight path angle (γ) is used to relate the velocity direction to the local horizontal. The thrust angle (α) relates the thrust direction to the velocity direction. Figure 4-4 shows the relationship between the normal-tangential and radial-transverse directions. This figure is especially useful when determining the gravity components along the normal-tangential axes.

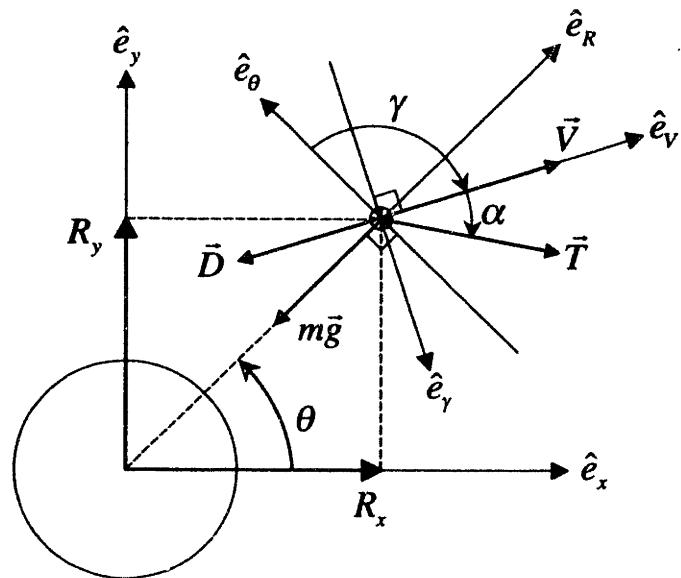


Figure 4-3: Normal-Tangential Polar Coordinates – Forces on a Point Mass

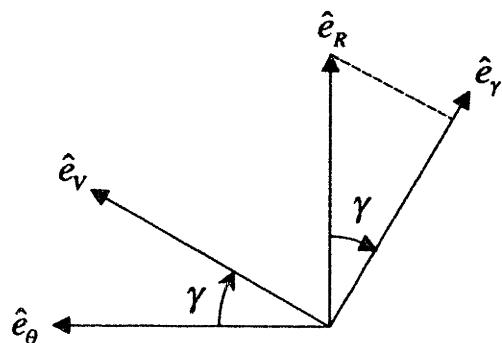


Figure 4-4: Relation Between Normal-Tangential and Radial-Transverse Directions

The sum of all forces can be written along the tangential and normal directions:

$$\begin{aligned}\Sigma F_V &= T \cos \alpha - D - mg \sin \gamma = ma_V \\ \Sigma F_\gamma &= T \sin \alpha - mg \cos \gamma = ma_\gamma\end{aligned}\quad (4.56)$$

where

- T = thrust magnitude
- α = thrust angle
- D = drag
- g = gravity
- γ = flight path angle
- a_V = tangential acceleration
- a_γ = normal acceleration
- m = mass

The accelerations in the normal and tangential directions are given below [13, p. 55]:

$$\begin{aligned}a_V &= \dot{V} \\ a_\gamma &= V \dot{\beta}\end{aligned}\quad (4.57)$$

where

$$\dot{\beta} = \text{angular direction change of normal and tangential directions}$$

It can be seen from Figures 4-3 and 4-4 that there are two components to the angular change in these directions. One is due to the rotation of the velocity vector relative to the radial-transverse frame. This change is $\dot{\gamma}$. The other change is due to the rotation of the radial-transverse frame. This change is $-\dot{\theta}$. The negative sign is necessary because, as defined, the theta rotation acts opposite to the gamma rotation. The normal acceleration can then be written as:

$$a_\gamma = V (\dot{\gamma} - \dot{\theta}) = V \left(\dot{\gamma} - \frac{V \cos \theta}{R} \right) \quad (4.58)$$

Equations 4.58 and 4.57 can be combined with equations 4.56:

$$\begin{aligned} T \cos \alpha - D - mg \sin \gamma &= m \dot{V} \\ T \sin \alpha - mg \cos \gamma &= mV \left(\dot{\gamma} - \frac{V \cos \theta}{R} \right) \end{aligned} \quad (4.59)$$

Thrust

For a perfectly expanded rocket engine, the thrust magnitude and mass flow rate are related by:

$$T = -\dot{m}V_{exit} = -\dot{m}g_o I_{sp} \quad (4.60)$$

where

- T = thrust magnitude
- \dot{m} = mass flow rate
- V_{exit} = engine exit velocity
- g_o = gravitational acceleration at sea level
- I_{sp} = specific impulse of rocket engine

Dynamic Equations

The following dynamic equations can be written:

$$\begin{aligned} \dot{R} &= V \sin \gamma \\ \dot{\theta} &= \frac{V \cos \gamma}{R} \\ \dot{V} &= \frac{T \cos \alpha}{m} - \frac{D}{m} - g \sin \gamma \\ \dot{\gamma} &= \frac{T \sin \alpha}{mV} - \frac{g \cos \gamma}{V} + \frac{V \cos \gamma}{R} \\ \dot{m} &= -\frac{T}{g_o I_{sp}} \end{aligned} \quad (4.61)$$

Drag

The magnitude of the drag is given by:

$$\vec{D} = \vec{q} C_D A_{ref} \quad (4.62)$$

Dynamic Pressure

The dynamic pressure is given by:

$$\vec{q} = \frac{1}{2} \vec{\rho} \vec{V}^2 \quad (4.63)$$

Atmospheric Density

See Appendix C.1 for the exponential atmospheric density model used.

Gravity

The magnitude of the gravitational acceleration is:

$$\vec{g} = \frac{\mu}{\vec{R}^2} \quad (4.64)$$

Trajectory Constraint Conditions

Limits are placed on the maximum thrust magnitude, maximum dynamic pressure, and sensed acceleration magnitude.

$$T \leq T_{max} \quad (4.65)$$

$$q \leq q_{max} \quad (4.66)$$

$$a_{sensed}^2 \leq (a_{sensed})_{max}^2 \quad (4.67)$$

Note that the thrust magnitude is now an optimization variable. This means that no extra constraint is necessary since it can be bounded directly.

The sensed acceleration is given by:

$$\begin{aligned}\vec{a}_{\text{sensed}_V} &= \frac{\vec{T} \cos \vec{\alpha} - \vec{D}}{\vec{m}} \\ \vec{a}_{\text{sensed}, \gamma} &= \frac{\vec{T} \sin \vec{\alpha}}{\vec{m}} \\ (\vec{a}_{\text{sensed}})_{\text{mag}}^2 &= \vec{a}_{\text{sensed}_V}^2 + \vec{a}_{\text{sensed}, \gamma}^2\end{aligned}\quad (4.68)$$

Burnout Constraint Conditions

It is desired to reach a circular target orbit.

$$R_f = R_{f_{\text{target}}} \quad (4.69)$$

$$V_f = V_{c_{\text{circ}}_{\text{target}}} \quad (4.70)$$

$$\gamma_f = 0 \quad (4.71)$$

Note that these burnout conditions are physically identical to those for the cartesian coordinates, but they are much simpler to write. In fact, these constraints are so simple that they can be written as simple bounds on the final state optimization variables.

Constraints

After transforming the time domain and using the D -matrix to change the derivatives into algebraic expressions, the constraints can be written as follows:

$$\vec{C}_{\dot{R}} = D_{NN} \vec{R} - \vec{V} \sin \vec{\gamma} = 0 \quad (4.72)$$

$$\vec{C}_{\dot{\theta}} = D_{NN} \vec{\theta} - \frac{\vec{V} \cos \vec{\gamma}}{\vec{R}} = 0 \quad (4.73)$$

$$\vec{C}_{\dot{V}} = D_{NN} \vec{V} - \frac{\vec{T} \cos \vec{\alpha}}{\vec{m}} + \frac{\vec{D}}{\vec{m}} + \vec{g} \sin \vec{\gamma} = 0 \quad (4.74)$$

$$\vec{C}_{\dot{\gamma}} = D_{NN} \vec{\gamma} - \frac{\vec{T} \sin \vec{\alpha}}{\vec{m} \vec{V}} + \frac{\vec{g} \cos \vec{\gamma}}{\vec{V}} - \frac{\vec{V} \cos \vec{\gamma}}{\vec{R}} = 0 \quad (4.75)$$

$$\vec{C}_m = D_{NN}\vec{m} + \frac{\vec{T}}{g_o I_{sp}} = 0 \quad (4.76)$$

$$\vec{C}_q = \vec{q} \leq q_{max} \quad (4.77)$$

$$\vec{C}_a = \vec{a}_{mag}^2 \leq a_{max}^2, \quad (4.78)$$

Optimization Vector

$$\vec{x}_{opt} = \begin{bmatrix} \vec{R} \\ \vec{\theta} \\ \vec{V} \\ \vec{\gamma} \\ \vec{m} \\ \vec{T} \\ \vec{\alpha} \\ \tau_f \end{bmatrix} \quad (4.79)$$

where τ_f is the final time.

Optimization Vector Bounds

The bounds on the optimization variables are:

$$\begin{aligned} 0 &\leq \vec{R} \leq \infty \\ -\infty &\leq \vec{\theta} \leq \infty \\ 0 &\leq \vec{V} \leq \infty \\ -\pi &\leq \vec{\gamma} \leq \pi \\ (1-f_p)m_o &\leq \vec{m} \leq m_o \\ 0 &\leq \vec{T} \leq T_{max} \\ -\pi &\leq \vec{\alpha} \leq \pi \\ 0 &\leq \tau_f \leq \infty \end{aligned} \quad (4.80)$$

The exception to these bounds is at the initial point. The initial states and controls

are defined (not allowed to vary) by setting the initial upper and lower bounds to the same values.

Jacobian

The Jacobian for the normal-tangential polar coordinate system is defined in Appendix F.3.

4.4 Coordinate System Comparison Results

Table 4.1 shows the numerical parameters used for the coordinate system comparison. Note that these parameters are not from any specific, real launch vehicle. However, the values are physically possible. Note that the simulation starts just prior to lift-off, with the rocket going straight up at maximum thrust. For a discussion of the initial guess generation for this problem, see Appendix E.

Initial Altitude (m)	10
Initial Velocity (m/s)	10
Initial Total Mass (kg)	100,000
Initial Propellant Mass over Initial Total Mass	0.99
Drag Coefficient	0.6
Aerodynamic Reference Area (m^2)	40
Specific Impulse of Engine (s)	300
Maximum Thrust to Initial Weight Ratio	1.5
Maximum Dynamic Pressure Limit (P_c)	14,000
Maximum Sensed Acceleration Magnitude (g 's)	3
Circular Target Orbit Altitude (km)	400

Table 4.1: Numerical Values Used for Coordinate System Comparison

Several cases were run in order to compare the relative merits of the different coordinate systems. Figures 4-5 and 4-7 show the objective function values versus number of iterations for solutions with 20 points and 40 points, respectively. Figures 4-6 and 4-8 show the elapsed CPU time versus number of iterations for solutions with 20 points and 40 points, respectively. This thesis is concerned with rapid trajectory optimization. Therefore, speed of convergence is more important than having the

“most” optimal solution. With this in mind, the plots can be analyzed to judge the relative performance of the coordinate systems.

- **Objective Function**

The purpose of the optimizer is to minimize the objective function. For the 20 point case, it is difficult to pick one coordinate system as the “best.” For fewer iterations, radial-transverse seems good, while normal-tangential is better for more than 50 iterations. For the 40 point case, radial-tangential is the best.

- **CPU Time**

For both the 20 point case and the 40 point case, the normal-tangential coordinate system seems to be the fastest.

Of the three coordinate systems analyzed, the normal-tangential coordinate system seems to be the most desirable, mainly due to its speed. However, Chapter 5 will introduce a technique that can be used to increase the speed of some of the coordinate systems.

Figures 4-9 through 4-16 show a sample solution for the problem of the coordinate comparison. These figures show no surprises and are included here for completeness. Note in figures 4-15 and 4-16 that the dynamic pressure and sensed acceleration limits are active on parts of the trajectory. This is a demonstration of how well the Legendre Pseudospectral Method can handle nonlinear inequality constraints.

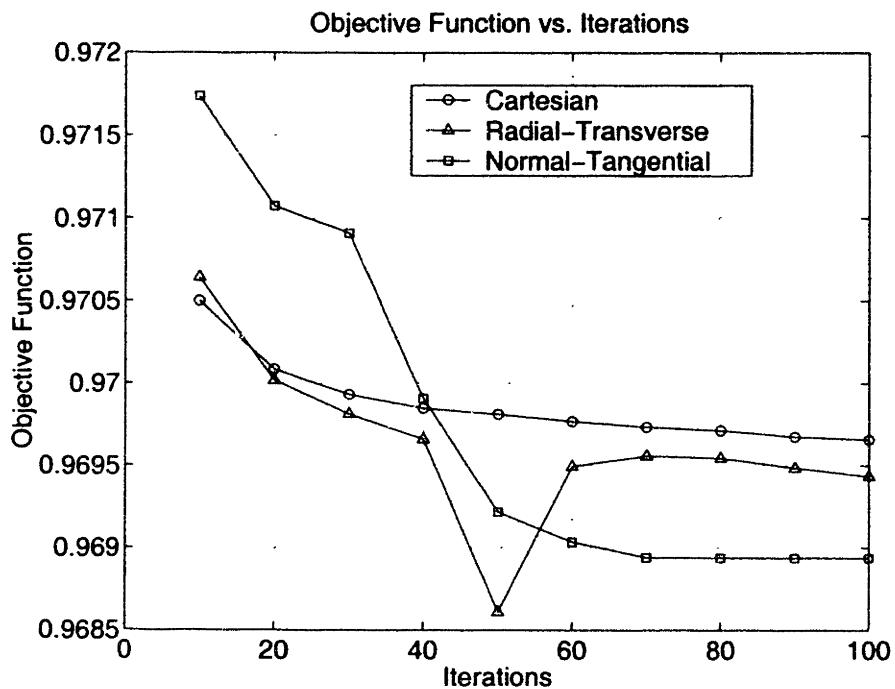


Figure 4-5: Objective Function vs. Iterations, 20 points

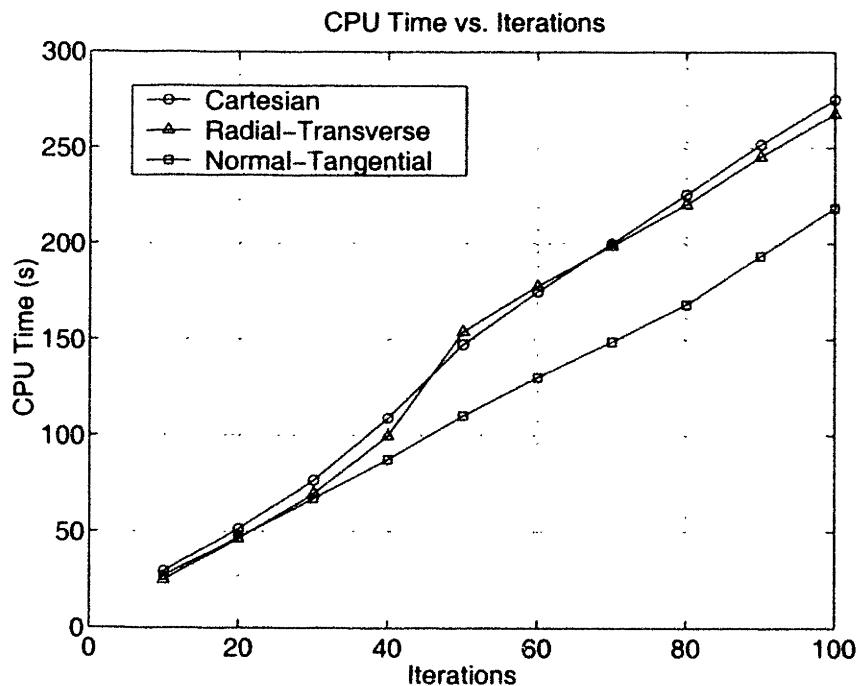


Figure 4-6: CPU Time vs. Iterations, 20 points

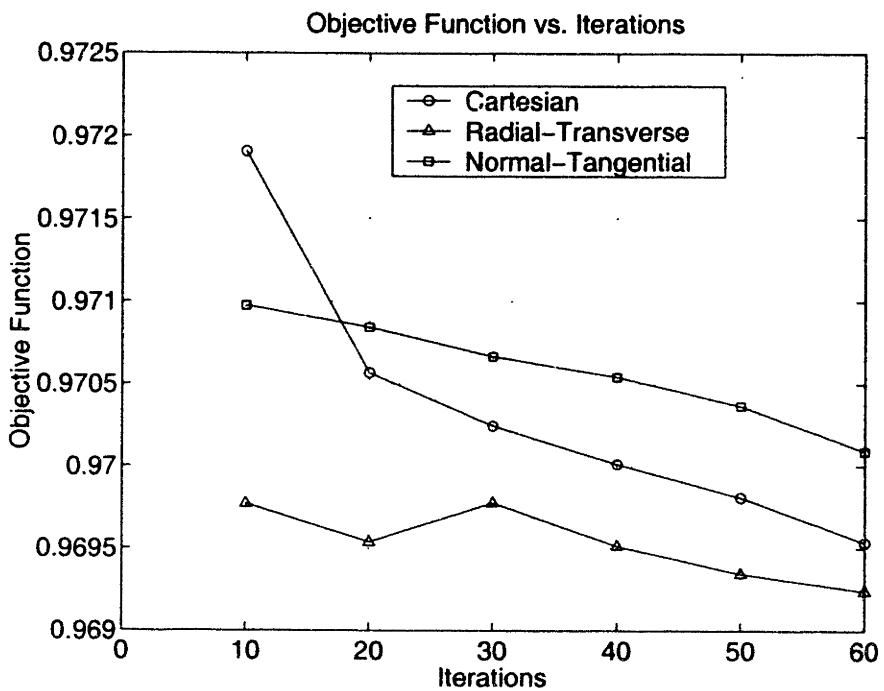


Figure 4-7: Objective Function vs. Iterations, 40 points

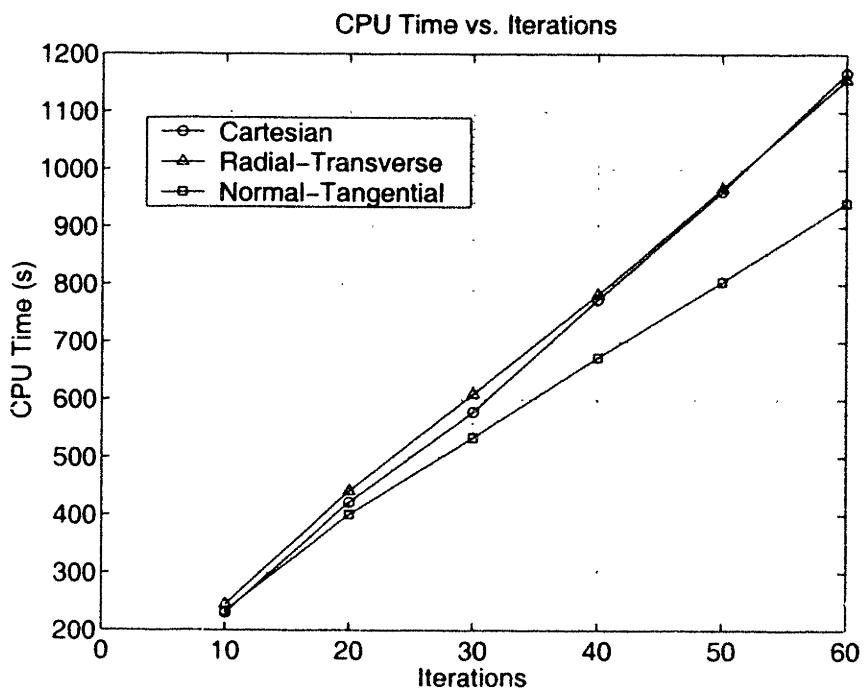


Figure 4-8: CPU Time vs. Iterations, 40 points

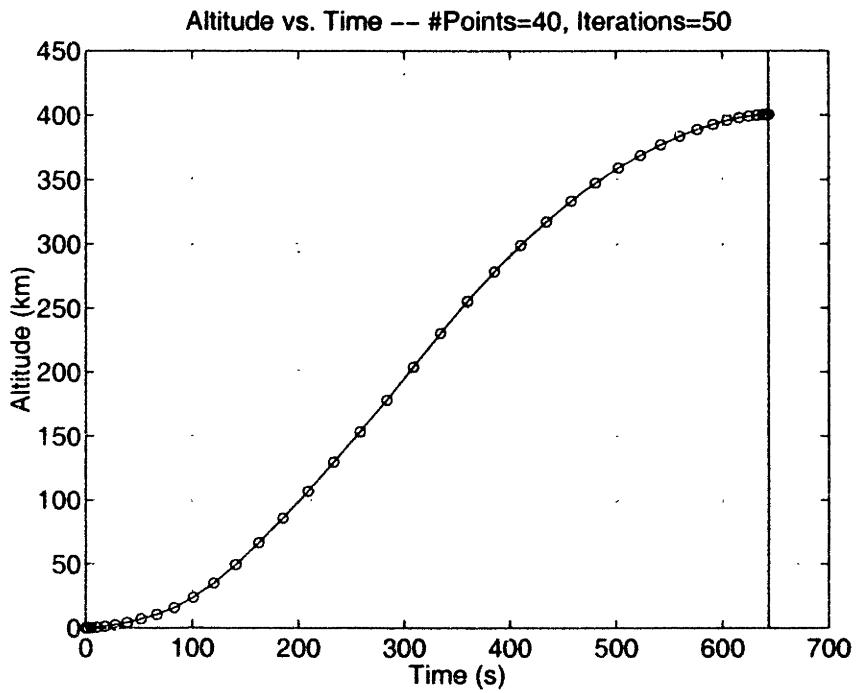


Figure 4-9: Two-Dimensional Launch: Time History of Altitude

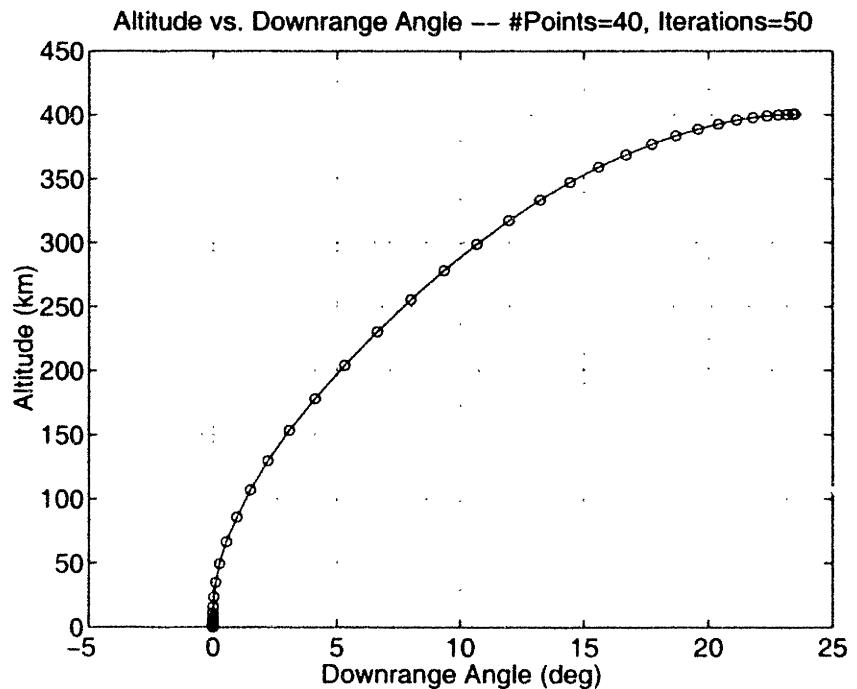


Figure 4-10: Two-Dimensional Launch: Trajectory

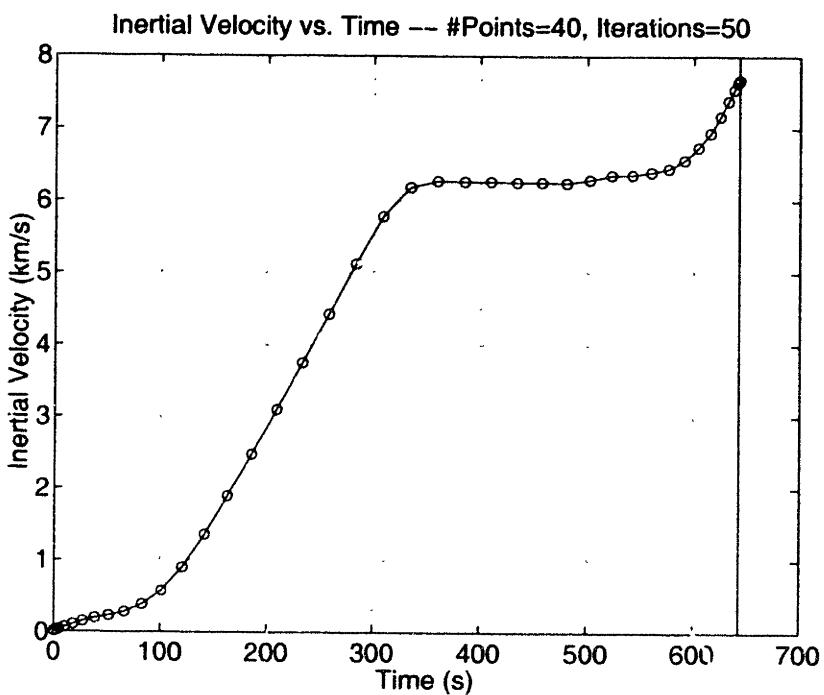


Figure 4-11: Two-Dimensional Launch: Time History of Velocity

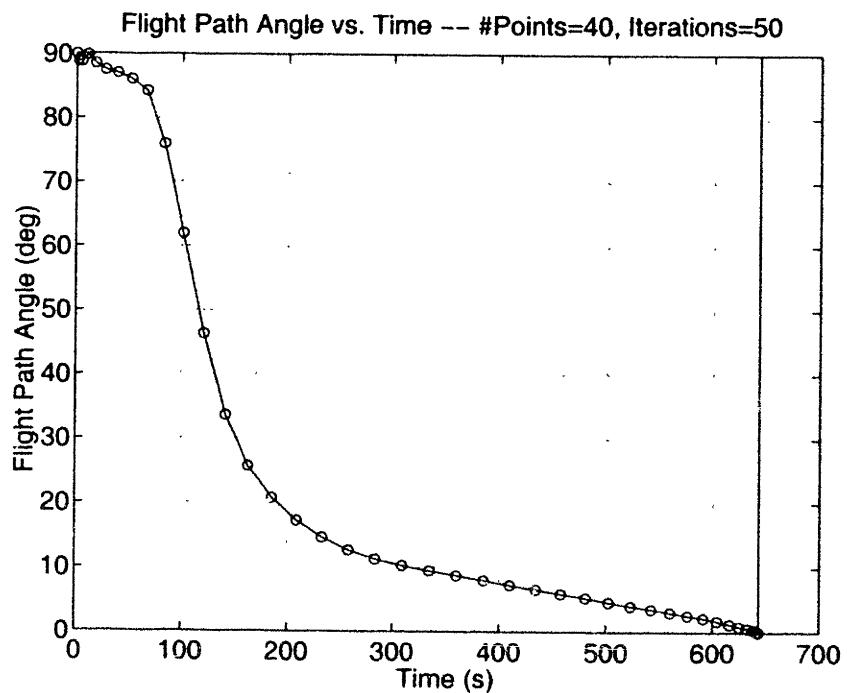


Figure 4-12: Two-Dimensional Launch: Time History of Flight Path Angle

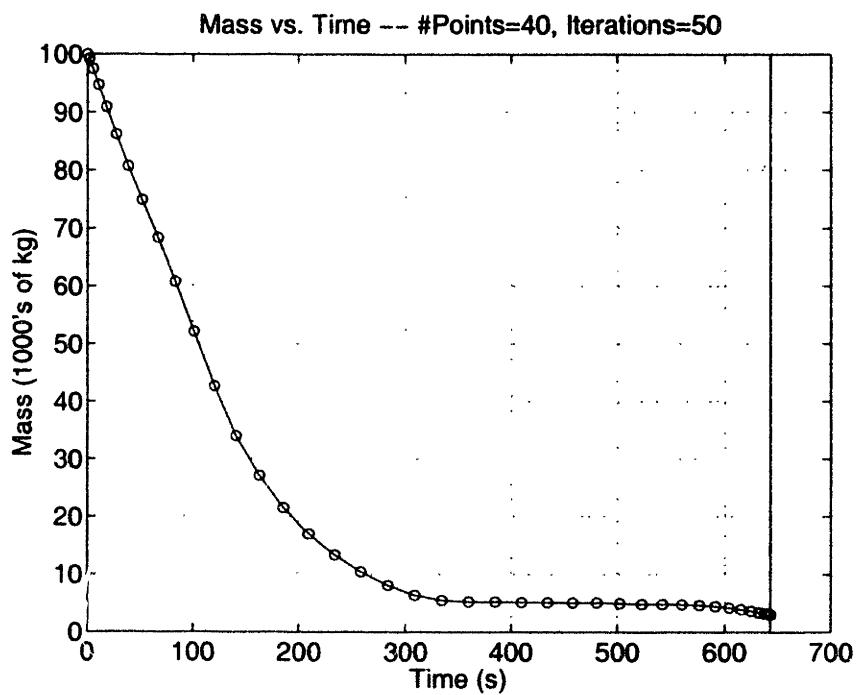


Figure 4-13: Two-Dimensional Launch: Time History of Mass

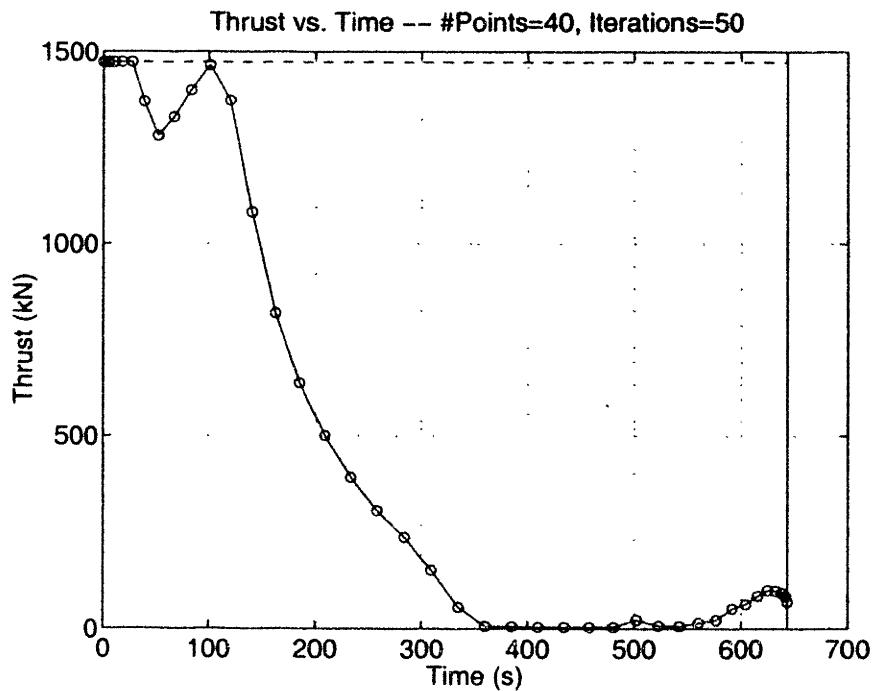


Figure 4-14: Two-Dimensional Launch: Time History of Thrust

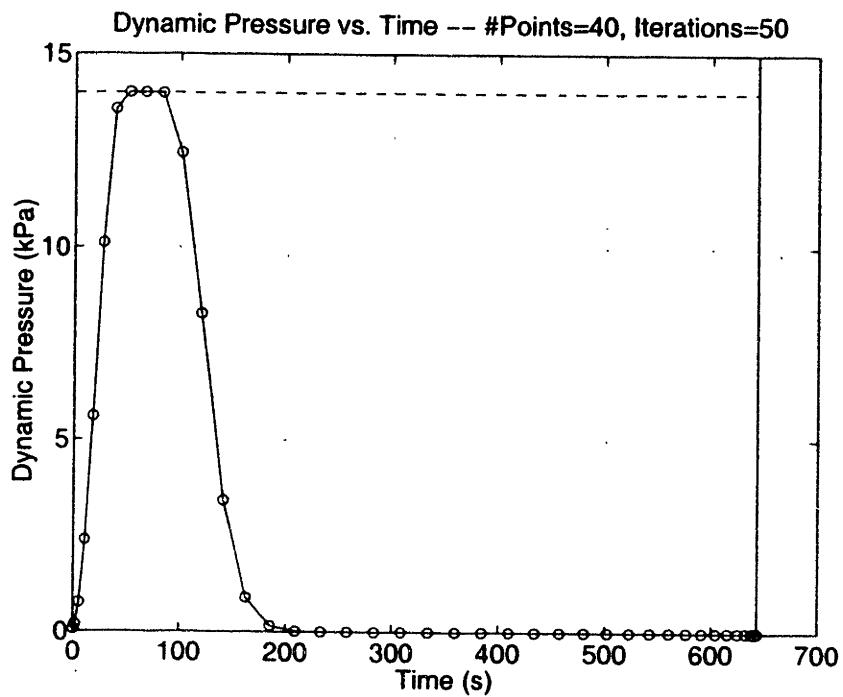


Figure 4-15: Two-Dimensional Launch: Time History of Dynamic Pressure

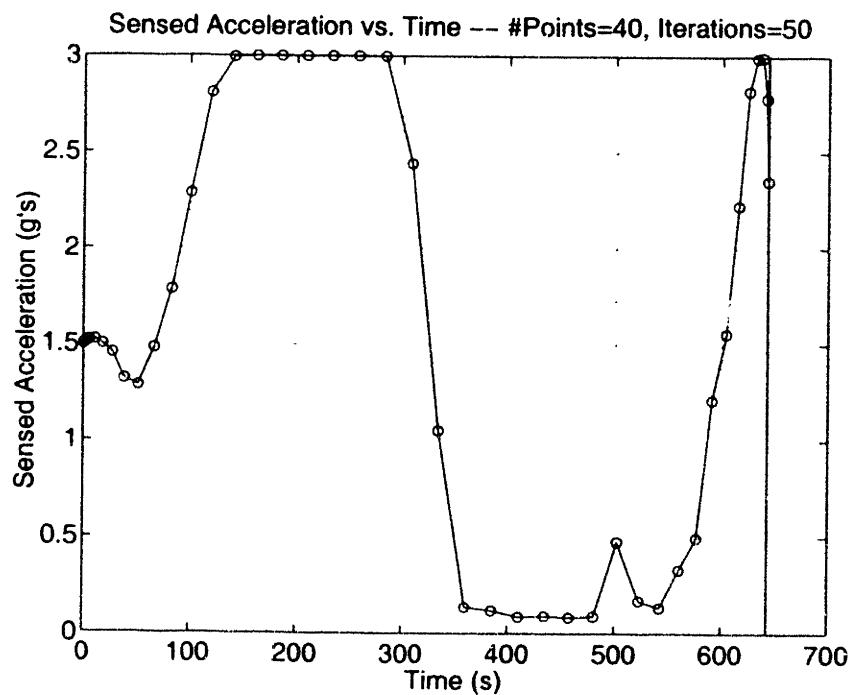


Figure 4-16: Two-Dimensional Launch: Time History of Sensed Acceleration

Chapter 5

D^2 Order Reduction Method

5.1 D^2 -Method

One way to speed up the convergence of the numerical optimizer is to reduce the order of the problem. The total length of the optimization vector should be reduced. Thus, the numerical optimizer will have fewer parameters to change in order to optimize the problem. One idea, proposed by Mike Ross, is the D^2 -Method. This method is useful when dealing with systems that have second-order differential equations. Let there be a system that can be modelled by the following set of differential equations:

$$\begin{aligned}\dot{x} &= V \\ \dot{V} &= a\end{aligned}\tag{5.1}$$

where a is a known function of time.

As shown in Chapter 2, the D -matrix can be used to write a discretization of the differential equations as a system of algebraic equations. After transforming the real time domain to the LGL time domain, the D -matrix can be applied to equations 5.1. The result is:

$$\begin{aligned} D_N \vec{x} &= \frac{(\tau_f - \tau_o)}{2} \vec{V} \\ D_N \vec{V} &= \frac{(\tau_f - \tau_o)}{2} \vec{a} \end{aligned} \quad (5.2)$$

These equations could be written as equality constraints in a nonlinear programming problem. The optimization vector for the problem would be:

$$\vec{x}_{opt} = \begin{bmatrix} \vec{x} \\ \vec{V} \end{bmatrix}$$

Rearranging equations 5.2 leads to:

$$\frac{2D_N}{(\tau_f - \tau_o)} \vec{x} = D_{NN} \vec{x} = \vec{V} \quad (5.3)$$

$$\frac{2D_N}{(\tau_f - \tau_o)} \vec{V} = D_{NN} \vec{V} = \vec{a} \quad (5.4)$$

Now, multiplying equation 5.3 by the D_{NN} matrix leads to:

$$D_{NN}^2 \vec{x} = D_{NN} \vec{V} \quad (5.5)$$

Equation 5.5 can be plugged into equation 5.4:

$$D_{NN} \vec{V} = D_{NN}^2 \vec{x} = \vec{a} \quad (5.6)$$

The nonlinear programming problem can now be rewritten with only one constraint as defined by equation 5.6. The optimization vector is reduced to:

$$\vec{x}_{opt} = [\vec{x}]$$

The vector \vec{V} is now exactly defined to be:

$$\vec{V} = D_{NN}\vec{x} \quad (5.7)$$

This alternate form is equivalent to writing the original equations 5.1 as:

$$\begin{aligned} \dot{x} &= V \\ \ddot{x} &= a \end{aligned} \quad (5.8)$$

This is the basic idea behind the D^2 -Method. The next section will look at its application with the coordinate systems from Chapter 4.

5.2 Application of D^2 -Method to Two-Dimensional Coordinate Systems

5.2.1 Cartesian Coordinates

From Chapter 4, the two-dimensional cartesian dynamic equations are:

$$\begin{aligned} \dot{R}_x &= V_x \\ \dot{R}_y &= V_y \\ \dot{V}_x &= \frac{T_x}{m} + \frac{D_x}{m} + g_x \\ \dot{V}_y &= \frac{T_y}{m} + \frac{D_y}{m} + g_y \\ \dot{m} &= -\frac{(T_x^2 + T_y^2)^{1/2}}{g_o I_{sp}} \end{aligned} \quad (5.9)$$

They can be rewritten as:

$$\dot{R}_x = V_x$$

$$\dot{R}_y = V_y$$

$$\begin{aligned}\ddot{R}_x &= \frac{T_x}{m} + \frac{D_x}{m} + g_x \\ \ddot{R}_y &= \frac{T_y}{m} + \frac{D_y}{m} + g_y \\ \dot{m} &= -\frac{(T_x^2 + T_y^2)^{1/2}}{g_o I_{sp}}\end{aligned}\tag{5.10}$$

Originally, there were 5 states (\vec{R}_x , \vec{R}_y , \vec{V}_x , \vec{V}_y , \vec{m}), 2 controls (T_x , T_y), and 1 final time (τ_f). The total size of the original optimization vector is:

$$size(\vec{x}_{opt}) = 7n_{LGL} + 1$$

where n_{LGL} is the number of LGL points.

Using the D^2 -Method, \vec{V}_x and \vec{V}_y can be cut from the optimization vector. Thus, the total size of the D^2 optimization vector is:

$$size(\vec{x}_{opt}) = 5n_{LGL} + 1$$

Constraints

The velocities are defined as:

$$\vec{V}_x = D_{NN} \vec{R}_x\tag{5.11}$$

$$\vec{V}_y = D_{NN} \vec{R}_y\tag{5.12}$$

The initial and final velocities are:

$$V_{x_0} = d_{NN_1} \vec{R}_x \quad (5.13)$$

$$V_{y_0} = d_{NN_1} \vec{R}_y \quad (5.14)$$

$$V_{x_f} = d_{NN_f} \vec{R}_x \quad (5.15)$$

$$V_{y_f} = d_{NN_f} \vec{R}_y \quad (5.16)$$

where d_{NN_1} is the first row and d_{NN_f} is the last row of the D_{NN} matrix.

The velocity magnitude is:

$$\vec{V}_{mag} = \sqrt{\vec{V}_x^2 + \vec{V}_y^2} \quad (5.17)$$

The constraints for the D^2 -method cartesian coordinates are very similar to those for the D -method cartesian coordinates. Recall that the initial velocities of the launch vehicle are set. It is necessary to add constraints for the initial velocities (equations 5.24 and 5.25). After transforming the time domain and using the D - and D^2 -matrices to change the derivatives into algebraic expressions, the constraints can be written as follows:

$$\vec{C}_{\vec{R}_x} = D_{NN}^2 \vec{R}_x - \frac{\vec{T}_x}{\vec{m}} - \frac{\vec{D}_x}{\vec{m}} - \vec{g}_x = 0 \quad (5.18)$$

$$\vec{C}_{\vec{R}_y} = D_{NN}^2 \vec{R}_y - \frac{\vec{T}_y}{\vec{m}} - \frac{\vec{D}_y}{\vec{m}} - \vec{g}_y = 0 \quad (5.19)$$

$$\vec{C}_{\vec{m}} = D_{NN} \vec{m} + \frac{(\vec{T}_x^2 + \vec{T}_y^2)^{1/2}}{g_o I_{sp}} = 0 \quad (5.20)$$

$$\vec{C}_T = \vec{T}_x^2 + \vec{T}_y^2 \leq T_{max}^2 \quad (5.21)$$

$$\vec{C}_q = \vec{q} \leq q_{max} \quad (5.22)$$

$$\vec{C}_a = \vec{a}_{mag}^2 \leq a_{max}^2 \quad (5.23)$$

$$C_{V_{xo}} = d_{NN_1} \vec{R}_x = V_{xo} \quad (5.24)$$

$$C_{V_{yo}} = d_{NN_1} \vec{R}_y = V_{yo} \quad (5.25)$$

$$C_{R_f} = R_{x_f}^2 + R_{y_f}^2 = R_{mag_f}^2 \quad (5.26)$$

$$C_{V_f} = V_{x_f}^2 + V_{y_f}^2 = V_{mag_f}^2 \quad (5.27)$$

$$C_{\tilde{R}\tilde{V}} = R_{x_f}V_{x_f} + R_{y_f}V_{y_f} = 0 \quad (5.28)$$

Optimization Vector

$$\vec{x}_{opt} = \begin{bmatrix} \vec{R}_x \\ \vec{R}_y \\ \vec{m} \\ \vec{T}_x \\ \vec{T}_y \\ \tau_f \end{bmatrix} \quad (5.29)$$

Optimization Vector Bounds

The bounds on the optimization variables are:

$$\begin{aligned} -\infty &\leq \vec{R}_x \leq \infty \\ -\infty &\leq \vec{R}_y \leq \infty \\ (1 - f_p)m_o &\leq \vec{m} \leq m_o \\ -T_{\max} &\leq \vec{T}_x \leq T_{\max} \\ -T_{\max} &\leq \vec{T}_y \leq T_{\max} \\ 0 &\leq \tau_f \leq \infty \end{aligned} \quad (5.30)$$

The exception to these bounds is at the initial point. The initial states and controls

are defined (not allowed to vary) by setting the initial upper and lower bounds to the same values.

Jacobian

The Jacobian for the D^2 cartesian coordinate system is defined in Appendix F.4.

5.2.2 Radial-Transverse Polar Coordinates

From Chapter 4, the two-dimensional radial-transverse polar dynamic equations are:

$$\begin{aligned}\dot{R} &= V_R \\ \dot{\theta} &= \frac{V_\theta}{R} \\ \dot{V}_R &= \frac{T_R}{m} + \frac{D_R}{m} - g + \frac{V_\theta^2}{R} \\ \dot{V}_\theta &= \frac{T_\theta}{m} + \frac{D_\theta}{m} - \frac{V_R V_\theta}{R} \\ \dot{m} &= -\frac{(T_R^2 + T_\theta^2)^{1/2}}{g_o I_{sp}}\end{aligned}\tag{5.31}$$

In order to rewrite the \dot{V}_θ equation, its derivation must be reexamined. Parts of equations 4.32 and 4.33 from Chapter 4 are rewritten below:

$$a_\theta = R\ddot{\theta} + 2\dot{R}\dot{\theta} = \dot{V}_\theta + \frac{V_R V_\theta}{R}\tag{5.32}$$

$$T_\theta + D_\theta = m a_\theta = m \left(R\ddot{\theta} + 2\dot{R}\dot{\theta} \right) = m \left(R\ddot{\theta} + \frac{2V_R V_\theta}{R} \right)\tag{5.33}$$

Equation 5.33 can be rewritten as:

$$\ddot{\theta} = \frac{T_\theta}{mR} + \frac{D_\theta}{mR} - \frac{2V_R V_\theta}{R^2}\tag{5.34}$$

The dynamic equations can now be written as:

$$\dot{R} = V_R$$

$$\dot{\theta}R = V_\theta$$

$$\begin{aligned}\ddot{R} &= \frac{T_R}{m} + \frac{D_R}{m} - g + \frac{V_\theta^2}{R} \\ \ddot{\theta} &= \frac{T_\theta}{mR} + \frac{D_\theta}{mR} - \frac{2V_R V_\theta}{R^2}\end{aligned}\tag{5.35}$$

$$\dot{m} = -\frac{(T_R^2 + T_\theta^2)^{1/2}}{g_o I_{sp}}\tag{5.36}$$

Using the D^2 -Method, V_R and V_θ can be cut from the optimization vector. This will lead to a reduction in the size of the optimization vector from

$$size(\vec{x}_{opt}) = 7n_{LGL} + 1$$

to

$$size(\vec{x}_{opt}) = 5n_{LGL} + 1$$

Constraints

The velocities are defined as:

$$\vec{V}_R = D_{NN} \vec{R}\tag{5.37}$$

$$\vec{V}_\theta = (D_{NN} \vec{\theta}) \vec{R}\tag{5.38}$$

The initial and final velocities are:

$$V_{R_0} = d_{NN_1} \vec{R}\tag{5.39}$$

$$V_{\theta_0} = (d_{NN_1} \vec{\theta}) R_1\tag{5.40}$$

$$V_{R_f} = d_{NN_f} \vec{R} \quad (5.41)$$

$$V_{\theta_f} = (d_{NN_f} \vec{\theta}) R_f \quad (5.42)$$

where d_{NN_1} is the first row and d_{NN_f} is the last row of the D_{NN} matrix.

The constraints for the D^2 -method radial-transverse coordinates are very similar to those for the D -method radial-transverse coordinates. Note that for the D^2 -method, it is necessary to add constraints for the initial and final velocities (equations 5.49 through 5.52). After transforming the time domain and using the D - and D^2 -matrices to change the derivatives into algebraic expressions, the constraints can be written as follows:

$$\vec{C}_{\vec{R}} = D_{NN}^2 \vec{R} - \frac{\vec{T}_R}{\vec{m}} - \frac{\vec{D}_R}{\vec{m}} + \vec{g} - \frac{\vec{V}_\theta^2}{\vec{R}} = 0 \quad (5.43)$$

$$\vec{C}_{\vec{\theta}} = D_{NN}^2 \vec{\theta} - \frac{\vec{T}_\theta}{\vec{m} \vec{R}} - \frac{\vec{D}_\theta}{\vec{m} \vec{R}} + \frac{2 \vec{V}_R \vec{V}_\theta}{\vec{R}^2} = 0 \quad (5.44)$$

$$\vec{C}_{\vec{m}} = D_{NN} \vec{m} + \frac{(\vec{T}_R^2 + \vec{T}_\theta^2)^{1/2}}{g_o I_{sp}} = 0 \quad (5.45)$$

$$\vec{C}_{T_{\max}} = \vec{T}_R^2 + \vec{T}_\theta^2 \leq T_{\max}^2 \quad (5.46)$$

$$\vec{C}_q = \vec{q} \leq q_{\max} \quad (5.47)$$

$$\vec{C}_a = \vec{a}_{mag_s}^2 \leq a_{\max_s}^2 \quad (5.48)$$

$$C_{V_{R_o}} = d_{NN_1} \vec{R} = V_{R_o} \quad (5.49)$$

$$C_{V_{\theta_o}} = (d_{NN_1} \vec{\theta}) R_1 = V_{\theta_o} \quad (5.50)$$

$$C_{V_{R_f}} = d_{NN_f} \vec{R} = V_{R_f} \quad (5.51)$$

$$C_{V_{\theta_f}} = \left(d_{NN_f} \vec{\theta} \right) R_f = V_{\theta_f} \quad (5.52)$$

Optimization Vector

$$\vec{x}_{opt} = \begin{bmatrix} \vec{R} \\ \vec{\theta} \\ \vec{m} \\ \vec{T}_R \\ \vec{T}_\theta \\ \tau_f \end{bmatrix} \quad (5.53)$$

Optimization Vector Bounds

The bounds on the optimization variables are:

$$\begin{aligned} 0 &\leq \vec{R} \leq \infty \\ -\infty &\leq \vec{\theta} \leq \infty \\ (1 - f_p)m_o &\leq \vec{m} \leq m_o \\ -T_{\max} &\leq \vec{T}_R \leq T_{\max} \\ -T_{\max} &\leq \vec{T}_\theta \leq T_{\max} \\ 0 &\leq \tau_f \leq \infty \end{aligned} \quad (5.54)$$

The exception to these bounds is at the initial and final points. The initial states and controls are defined (not allowed to vary) by setting the initial upper and lower bounds to the same values. Also note that the final radial position is controlled in the same way.

Jacobian

The Jacobian for the D^2 radial-transverse polar coordinate system is defined in Appendix F.5.

5.2.3 Normal-Tangential Polar Coordinates

From Chapter 4, the two-dimensional normal-tangential polar dynamic equations are:

$$\begin{aligned}
 \dot{R} &= V \sin \gamma \\
 \dot{\theta} &= \frac{V \cos \gamma}{R} \\
 \dot{V} &= \frac{T \cos \alpha}{m} - \frac{D}{m} - g \sin \gamma \\
 \dot{\gamma} &= \frac{T \sin \alpha}{mV} - \frac{g \cos \gamma}{V} + \frac{V \cos \gamma}{R} \\
 \dot{m} &= -\frac{T}{g_o I_{sp}}
 \end{aligned} \tag{5.55}$$

There is no way to use the D^2 -Method with this coordinate system. The size of the optimization vector is:

$$\text{size}(\vec{x}_{opt}) = 7n_{LGL} + 1$$

5.3 Coordinate System Comparison Using the D^2 -Method

The same numerical parameters (see Table 4.1) and initial guess generation (see Appendix E) were used for the coordinate systems using the D^2 -method. Figures 5-1 and 5-3 show the objective function values versus number of iterations for solutions with 20 points and 40 points, respectively. Figures 5-2 and 5-4 show the elapsed CPU time versus number of iterations for solutions with 20 points and 40 points, respectively. The comparison data from Chapter 4 is plotted along with the comparison data for the coordinate systems using the D^2 -method.

- **Objective Function**

For the 20 point case, the D^2 -method cartesian and radial-transverse systems do better than their D -method counterparts; however, the normal-tangential system is still better for iterations greater than 50. For 40 points, the D^2 -method does not seem to do any better or worse. The radial-transverse system still has the best objective performance for the 40 point case.

- **CPU Time**

In general, using the D^2 -method increases the speed of a given coordinate system. For the 20 point case, the D^2 cartesian model is slightly faster for fewer iterations. However, the normal-tangential system is still faster for larger numbers of iterations. For the 40 point case, both D^2 -systems are faster, with the D^2 cartesian system slightly faster than the D^2 radial-transverse system.

It is difficult to choose the “best” coordinate system based on this analysis. In general, using the D^2 -method increases the speed of a given problem. However, it is difficult to see a trend in how the D^2 -method affects the performance of the objective function. Also, this analysis may or may not have any relevance to choosing the “best” coordinate system with which to model a complex three-dimensional launch problem. However, these results were used in part for this very purpose. The next section discusses the selection.

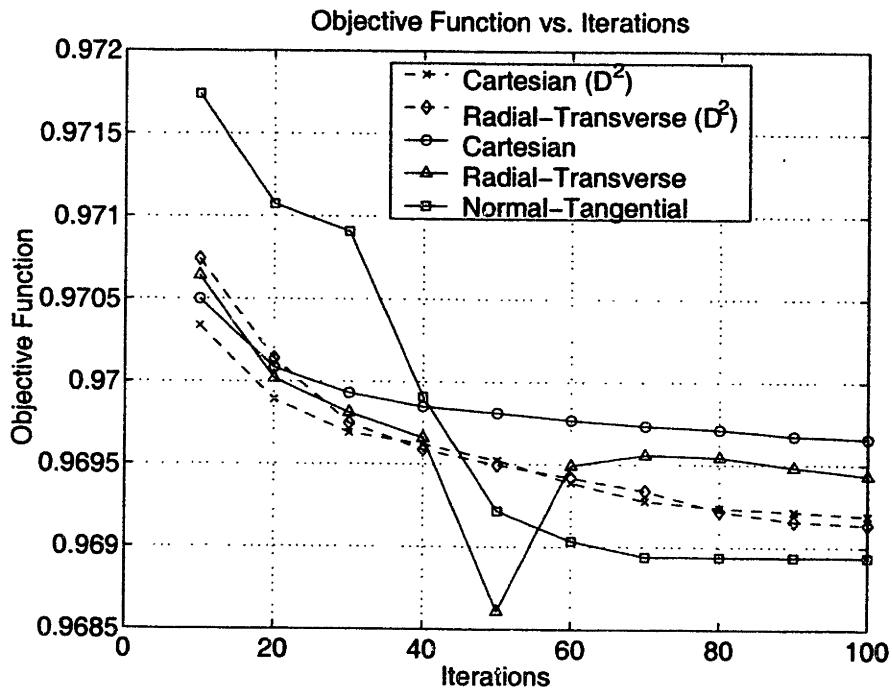


Figure 5-1: Objective Function vs. Iterations, 20 points

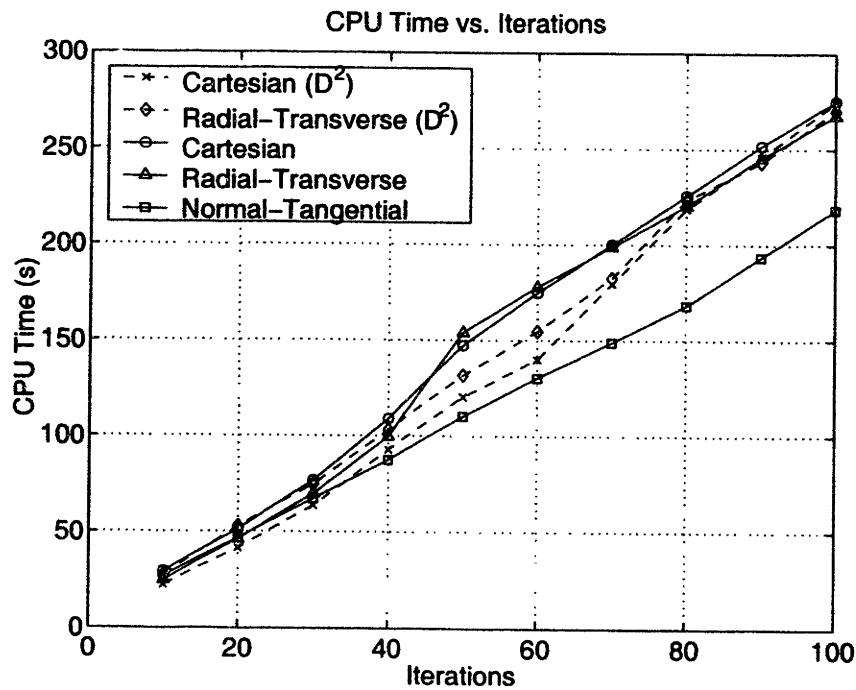


Figure 5-2: CPU Time vs. Iterations, 20 points

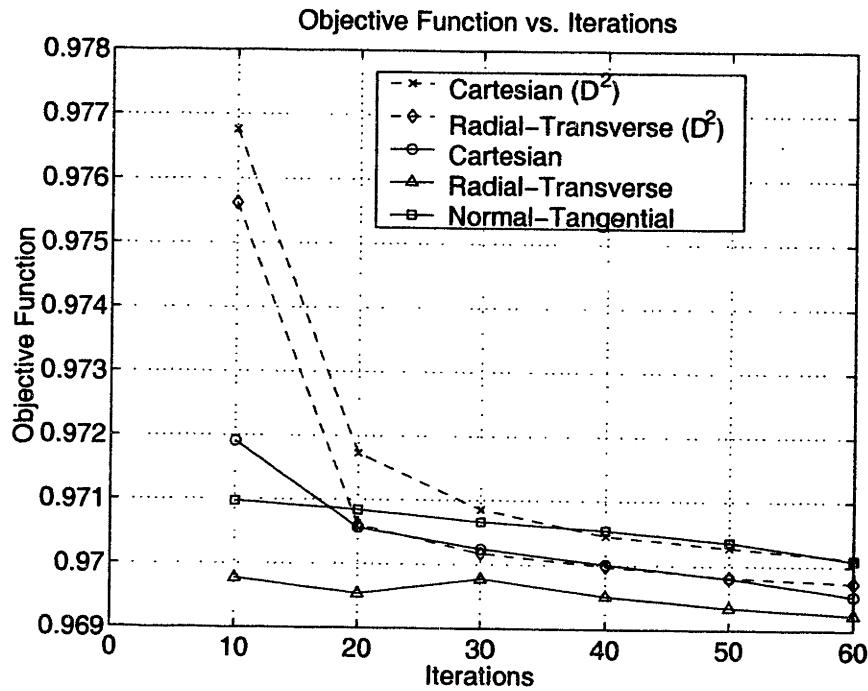


Figure 5-3: Objective Function vs. Iterations, 40 points

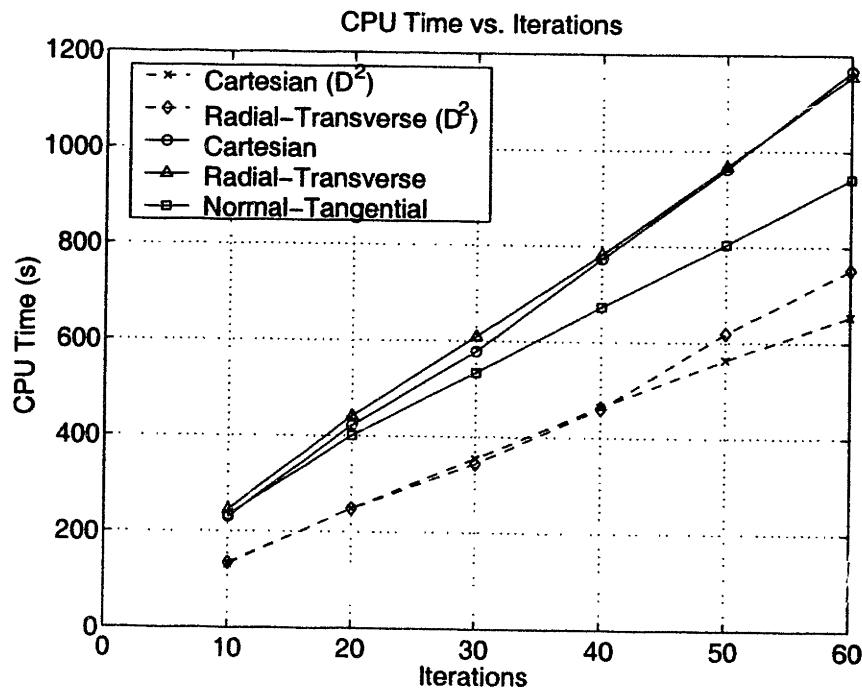


Figure 5-4: CPU Time vs. Iterations, 40 points

5.4 Application of D^2 -Method to Three-Dimensional Coordinate Systems

There are several variations possible for an inertial, three-dimensional coordinate system. These include:

- Cartesian Coordinates
- Cylindrical Coordinates
- Radial-Transverse Spherical Coordinates
- Normal-Tangential Spherical Coordinates

These systems are discussed in the following sections.

5.4.1 Cartesian Coordinates

The three-dimensional dynamic equations in cartesian coordinates are:

$$\dot{R}_x = V_x$$

$$\dot{R}_y = V_y$$

$$\dot{R}_z = V_z$$

$$\ddot{R}_x = \dot{V}_x = \frac{T_x}{m} + \frac{A_x}{m} + g_x$$

$$\ddot{R}_y = \dot{V}_y = \frac{T_y}{m} + \frac{A_y}{m} + g_y$$

$$\ddot{R}_z = \dot{V}_z = \frac{T_z}{m} + \frac{A_z}{m} + g_z$$

$$\dot{m} = -\frac{T}{g_o I_{sp}} \quad (5.56)$$

The optimization vector for the *D*-Method is:

$$\vec{x}_{opt} = \left[\vec{R}_x^T \ \vec{R}_y^T \ \vec{R}_z^T \ \vec{V}_x^T \ \vec{V}_y^T \ \vec{V}_z^T \ \vec{m}^T \ \vec{T}^T \ \vec{q}_1^T \ \vec{q}_2^T \ \vec{q}_3^T \ \vec{q}_4^T \ \tau_f \right]^T \quad (5.57)$$

where the \vec{q}_i are quaternion elements that define the attitude of the vehicle in inertial space. The size of this vector is:

$$size(\vec{x}_{opt}) = 12n_{LGL} + 1 \quad (5.58)$$

With the *D*²-Method, the velocities can be dropped from the optimization vector. The size of the optimization vector becomes:

$$size(\vec{x}_{opt}) = 9n_{LGL} + 1 \quad (5.59)$$

5.4.2 Cylindrical Coordinates

Cylindrical coordinates are very similar to the radial-transverse coordinate system described in Chapter 4. Only a Z-term is added along the axis of the reference cylinder. The three-dimensional dynamic equations in cylindrical coordinates are:

$$\dot{R} = V_R$$

$$\dot{\theta}R = V_\theta$$

$$\dot{Z} = V_Z$$

$$\ddot{R} = \frac{T_R}{m} + \frac{D_R}{m} - g_R + \frac{V_\theta^2}{R}$$

$$\ddot{\theta} = \frac{T_\theta}{mR} + \frac{D_\theta}{mR} - \frac{2V_R V_\theta}{R^2}$$

$$\ddot{Z} = \frac{T_Z}{m} + \frac{D_Z}{m} - g_Z$$

$$\dot{m} = -\frac{(T_R^2 + T_\theta^2 + T_Z^2)^{1/2}}{g_o I_{sp}} \quad (5.60)$$

The optimization vector for the *D*-Method is:

$$\vec{x}_{opt} = \left[\vec{R}^T \ \vec{\theta}^T \ \vec{Z}^T \ \vec{V}_R^T \ \vec{V}_\theta^T \ \vec{V}_Z^T \ \vec{m}^T \ \vec{T}^T \ \vec{q}_1^T \ \vec{q}_2^T \ \vec{q}_3^T \ \vec{q}_4^T \ \tau_f \right]^T \quad (5.61)$$

where the \vec{q}_i are quaternion elements that define the attitude of the vehicle in inertial space. The size of this vector is:

$$size(\vec{x}_{opt}) = 12n_{LGL} + 1 \quad (5.62)$$

With the *D*²-Method, the velocities can be dropped from the optimization vector. The size of the optimization vector becomes:

$$size(\vec{x}_{opt}) = 9n_{LGL} + 1 \quad (5.63)$$

5.4.3 Radial-Transverse Spherical Coordinates

The radial direction is in the direction of a vector drawn from the planet center to the current position. The theta direction is perpendicular to the radial direction and is positive in the direction of positive θ . The ϕ -direction is perpendicular to the radial direction and is positive in the direction of positive ϕ . The angles θ and ϕ are used to specify the angular position of the radial vector. They are measured from arbitrary reference planes. The sum of all forces can be written along the radial, θ , and ϕ directions.

$$\begin{aligned} \Sigma F_R &= T_R + D_R - mg = ma_R \\ \Sigma F_\theta &= T_\theta + D_\theta = ma_\theta \\ \Sigma F_\phi &= T_\phi + D_\phi = ma_\phi \end{aligned} \quad (5.64)$$

where

T_R	=	radial thrust
T_θ	=	theta thrust
T_ϕ	=	phi thrust
D_R	=	radial drag
D_θ	=	theta drag
D_ϕ	=	phi drag
g	=	gravity (only in negative radial direction)
a_R	=	radial acceleration
a_θ	=	theta acceleration
a_ϕ	=	phi acceleration
m	=	mass

The velocities in the radial, θ , and ϕ directions are [13, p. 83]:

$$\begin{aligned} V_R &= \dot{R} \\ V_\theta &= R\dot{\theta}\cos\phi \\ V_\phi &= R\dot{\phi} \end{aligned} \tag{5.65}$$

where

R	=	radial position
θ	=	theta angular position
ϕ	=	phi angular position

The accelerations in the radial, θ , and ϕ directions are given below [13, p. 83]:

$$\begin{aligned} a_R &= \ddot{R} - R\dot{\phi}^2 - R\dot{\theta}^2\cos^2\phi \\ a_\theta &= (2\dot{R}\dot{\theta} + R\ddot{\theta})\cos\phi - 2R\dot{\theta}\dot{\phi}\sin\phi \\ a_\phi &= 2\dot{R}\dot{\phi} + R\ddot{\phi} + R\dot{\theta}^2\sin\phi\cos\phi \end{aligned} \tag{5.66}$$

Equations 5.66 can be combined with equations 5.64:

$$\begin{aligned}
 T_R + D_R - mg &= m [\ddot{R} - R\dot{\phi}^2 - R\dot{\theta}^2 \cos^2 \phi] \\
 T_\theta + D_\theta &= m [(2\dot{R}\dot{\theta} + R\ddot{\theta}) \cos \phi - 2R\dot{\theta}\dot{\phi} \sin \phi] \\
 T_\phi + D_\phi &= m [2\dot{R}\dot{\phi} + R\ddot{\phi} + R\dot{\theta}^2 \sin \phi \cos \phi]
 \end{aligned} \tag{5.67}$$

These equations are very nonlinear, and it is not possible to write them in a form conducive to the Legendre Pseudospectral Method, for either the D - or D^2 -methods. The radial-transverse spherical coordinate system will not be considered further in this thesis.

5.4.4 Normal-Tangential Spherical Coordinates

Normal-tangential spherical coordinates are similar to the normal-tangential polar coordinate system described in Chapter 4. However, the spherical version is sufficiently different from the polar version that it can not be written intuitively from knowledge of the polar version. Therefore, the equations of motion will be quoted directly from reference [18]. The three-dimensional dynamic equations in normal-tangential spherical coordinates are:

$$\dot{R} = V \sin \gamma$$

$$\dot{\theta} = \frac{V \cos \gamma \cos \psi}{R}$$

$$\dot{\phi} = \frac{V \cos \gamma \sin \psi}{R}$$

$$\dot{V} = \frac{T \cos \alpha - D}{m} - g \sin \gamma$$

$$\dot{\psi} = \frac{(L + T \sin \alpha) \sin \delta}{m V \cos \gamma} - \frac{V}{R} \cos \gamma \cos \psi \tan \phi$$

$$\begin{aligned}\dot{\gamma} &= \frac{(L + T \sin \alpha) \cos \delta}{mV} - \frac{g \cos \gamma}{V} + \frac{V \cos \gamma}{R} \\ \dot{m} &= -\frac{T}{g_o I_{sp}}\end{aligned}\tag{5.68}$$

where

- R = radial position
- θ = theta angular position (longitude)
- ϕ = phi angular position (latitude)
- V = velocity magnitude
- ψ = heading angle
- γ = flight path angle
- α = angle of attack
- δ = bank angle

The optimization vector for the D -Method is:

$$\vec{x}_{opt} = \left[\vec{R}^T \ \vec{\theta}^T \ \vec{\phi}^T \ \vec{V}^T \ \vec{\psi}^T \ \vec{\gamma}^T \ \vec{m}^T \ \vec{T}^T \ \vec{\alpha}^T \ \vec{\delta}^T \ \tau_f \right]^T\tag{5.69}$$

The size of this vector is:

$$size(\vec{x}_{opt}) = 10n_{LGL} + 1\tag{5.70}$$

The dynamic equations do not directly involve second-order differential equations. It is not possible to use the D^2 -method with the normal-tangential spherical coordinate system.

5.5 Coordinate System Conclusions

Based on the results shown in section 5.3, the size of the optimization vector is important in determining the speed of the numerical problem. The cartesian and cylindrical coordinate systems have the smallest optimization vectors.

Cylindrical coordinates can be very useful for the launch problem. When a vehicle launches from the surface of the Earth directly into the plane of its target orbit,

cylindrical coordinates make natural sense. An example is a Shuttle launch from Florida into an orbit with an inclination of roughly 28 degrees. The Z-component becomes an out-of-plane component that must be nulled out by the optimizer. This means that when launching directly into the orbital plane, the problem becomes a two-dimensional problem with perturbations in a third dimension.

When not launching directly into the plane of the target orbit, cylindrical coordinates still make sense *as long as the plane of the target orbit is defined*. An example is a launch from South America toward a final goal of a geostationary orbit. The plane of the final orbit is defined. Another example would be a launch from a floating platform on the equator into an inclined orbit with a specified longitude of ascending node. Once again, the final orbit is defined.

However, if the final orbit is not defined, the cylindrical coordinate system loses its natural fit with the launch problem. Define the “polar plane” of the cylindrical coordinate system as the $R\theta$ -plane (i.e. zero Z-component). Assuming that the reference planes of the cylindrical coordinate system are set, it could occur that the final orbit of the optimal solution has components that are out of the polar plane. This could happen if a certain inclination is desired, but the longitude of the ascending node is free. In this case, in order to avoid components that are out of the polar plane of the coordinate system, pre-knowledge of the optimal target plane would be necessary. The polar plane would then need to be defined using this optimal target plane. This information may or may not be available, depending on the situation of the launch.

Another possibility for a case with specified final inclination but free longitude of ascending node would be to allow the optimizer to change the reference planes of the cylindrical coordinate system so that the optimal solution lies in the polar plane of the cylindrical coordinates. However, this would only add complexity to the optimization process, and might increase the time it takes to converge.

For the general launch problem, where the target orbit plane may or may not be defined, the cylindrical coordinate system seems to have no great advantage over the cartesian coordinate system. Also, the cartesian coordinate system is easier to

set-up and understand. For these reasons, cartesian coordinates are chosen for the development of a more accurate model for the flight of a launch vehicle. Chapter 6 discusses the development of this model.

Chapter 6

Three-Dimensional Cartesian Model

This chapter describes the model used to simulate the three-dimensional ascent of a launch vehicle from the surface of a planet to a target orbit. The model is described using both the D - and D^2 -methods. The two methods share many common parts. Each section in this chapter derives a part of the model for both methods. When the methods require different derivations, this is done. In sections 6.1 through 6.4, the constraints that define the three dimensional model in the nonlinear optimizer are described. In section 6.5, the optimization vector is defined. Due to the complexity of the three dimensional model, it is necessary to define it in terms of simple parts. This is done in section 6.6. The Jacobian matrix for the three dimensional model is derived in Appendix G.

The following assumptions are made for the three dimensional model:

1. The Legendre Pseudospectral Method is used with this model as a guidance algorithm that is decoupled from the control system of the vehicle. Thus, the commanded thrust and vehicle attitude are assumed to be achieved instantaneously. Note that the rotational equations of motion are not modelled.
2. The thrust vector is fixed with respect to the vehicle body frame. The thrust direction is controlled by changing the attitude of the vehicle.

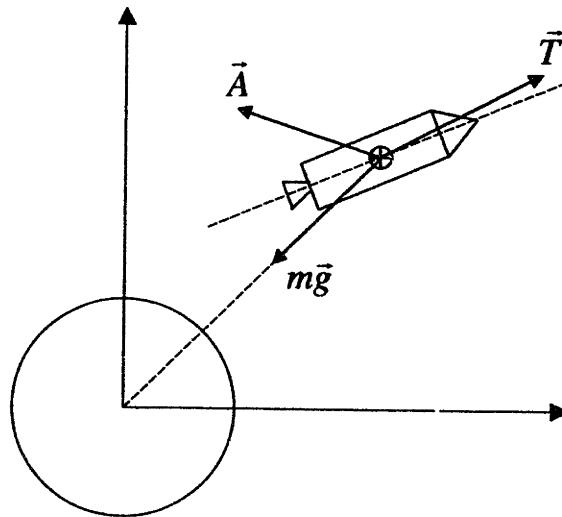


Figure 6-1: Forces on a Vehicle in Powered Atmospheric Flight

3. The rocket engine is perfectly expanded.
4. The launch vehicle has at least two stages.
5. The planet is a perfect sphere. Thus, the gravitational acceleration varies with the inverse of the square of the distance from the planet center.

6.1 Dynamic Constraints

6.1.1 Equations of Motion

Figure 6-1 shows the general forces on a vehicle in powered atmospheric flight. The general equations of motion for a vehicle travelling over a planet are:

$$\sum \vec{F} = \vec{T} + \vec{A} + m\vec{g} = m\ddot{\vec{R}} \quad (6.1)$$

where

- $\sum \vec{F}$ = sum of forces acting through the center of mass
- \vec{T} = thrust
- \vec{A} = aerodynamic forces
- m = mass
- \vec{g} = gravitational acceleration
- $\ddot{\vec{R}}$ = total acceleration

Equation 6.1 can be rewritten in two analytically identical forms.

- **D-Method**

This is the form of the equation that will be used for the *D*-method discussed in previous chapters.

$$\begin{aligned}\dot{\vec{R}} &= \dot{\vec{V}} \\ \dot{\vec{V}} &= \frac{\vec{T} + \vec{A}}{m} + \vec{g}\end{aligned}\tag{6.2}$$

- **D^2 -Method**

These equation will be used for the D^2 -method discussed in previous chapters.

$$\begin{aligned}\dot{\vec{R}} &= \dot{\vec{V}} \\ \ddot{\vec{R}} &= \frac{\vec{T} + \vec{A}}{m} + \vec{g}\end{aligned}\tag{6.3}$$

6.1.2 Mass Flow Rate

The thrust magnitude of the rocket is given by the following equation [1].

$$T = -\dot{m}V_{exit} + (p_\infty - p_{exit}) A_{exit}\tag{6.4}$$

where

T	=	thrust magnitude
\dot{m}	=	mass flow rate
V_{exit}	=	nozzle exit velocity
p_∞	=	atmospheric pressure
p_{exit}	=	nozzle exit pressure
A_{exit}	=	nozzle exit area

Assuming that the rocket nozzle is perfectly expanded (i.e. $p_{exit} = p_\infty$), the thrust magnitude is related to the mass flow rate by [1]

$$T = -\dot{m}V_{exit} = -\dot{m}g_o I_{sp} \quad (6.5)$$

where

g_o	=	gravity at the planet's surface
I_{sp}	=	specific impulse of rocket

Rearranging equation 6.5 results in:

$$\dot{m} = -\frac{T}{V_{exit}} \quad (6.6)$$

6.1.3 Dynamic Constraints

After transforming the time domain and using the D or D^2 matrix to change the derivatives into algebraic expressions, the dynamic constraints can be written as follows:

***D*-Method Dynamic Constraints**

$$\vec{C}_{\dot{R}_x} = D_{NN}\vec{R}_x - \vec{V}_x = 0 \quad (6.7)$$

$$\vec{C}_{\dot{R}_y} = D_{NN}\vec{R}_y - \vec{V}_y = 0 \quad (6.8)$$

$$\vec{C}_{\vec{R}_x} = D_{NN}\vec{R}_x - \vec{V}_z = 0 \quad (6.9)$$

$$\vec{C}_{\vec{V}_x} = D_{NN}\vec{V}_x - \frac{\vec{T}_x}{\vec{m}} - \frac{\vec{A}_x}{\vec{m}} - \vec{g}_x = 0 \quad (6.10)$$

$$\vec{C}_{\vec{V}_y} = D_{NN}\vec{V}_y - \frac{\vec{T}_y}{\vec{m}} - \frac{\vec{A}_y}{\vec{m}} - \vec{g}_y = 0 \quad (6.11)$$

$$\vec{C}_{\vec{V}_z} = D_{NN}\vec{V}_z - \frac{\vec{T}_z}{\vec{m}} - \frac{\vec{A}_z}{\vec{m}} - \vec{g}_z = 0 \quad (6.12)$$

$$\vec{C}_{\vec{m}} = D_{NN}\vec{m} + \frac{\vec{T}}{V_{exit}} = 0 \quad (6.13)$$

D^2 -Method Dynamic Constraints

Using the D^2 -Method, the velocities are defined directly in terms of the position vector by using the pseudospectral difference matrix.

$$\vec{V}_x = D_{NN}\vec{R}_x \quad \vec{V}_y = D_{NN}\vec{R}_y \quad \vec{V}_z = D_{NN}\vec{R}_z \quad (6.14)$$

The D^2 -method dynamic constraints are:

$$\vec{C}_{\vec{R}_x} = D_{NN}^2\vec{R}_x - \frac{\vec{T}_x}{\vec{m}} - \frac{\vec{A}_x}{\vec{m}} - \vec{g}_x = 0 \quad (6.15)$$

$$\vec{C}_{\vec{R}_y} = D_{NN}^2\vec{R}_y - \frac{\vec{T}_y}{\vec{m}} - \frac{\vec{A}_y}{\vec{m}} - \vec{g}_y = 0 \quad (6.16)$$

$$\vec{C}_{\vec{R}_z} = D_{NN}^2\vec{R}_z - \frac{\vec{T}_z}{\vec{m}} - \frac{\vec{A}_z}{\vec{m}} - \vec{g}_z = 0 \quad (6.17)$$

$$\vec{C}_{\vec{m}} = D_{NN}\vec{m} + \frac{\vec{T}}{V_{exit}} = 0 \quad (6.18)$$

Recall that

$$D_{NN} = \frac{2D_N}{(\tau_f - \tau_o)} \quad (6.19)$$

6.2 Trajectory Constraints

Remember that trajectory constraints apply at every LGL point, and they are typically inequality constraints.

6.2.1 Quaternion Normalization Constraint

As discussed in section 6.6.2, a normalization constraint is required for the quaternion elements. The constraint is:

$$\vec{C}_{qnorm} = (\vec{q}_1)^2 + (\vec{q}_2)^2 + (\vec{q}_3)^2 + (\vec{q}_4)^2 = 1 \quad (6.20)$$

6.2.2 Dynamic Pressure Constraint

$$\vec{C}_q = \vec{q} \leq q_{\max} \quad (6.21)$$

The dynamic pressure is defined later in the chapter.

6.2.3 Sensed Acceleration Constraint

The sensed acceleration is defined as the acceleration on the vehicle caused by all forces except gravitational force.

$$\vec{a}_{sensed} = \frac{\vec{T} + \vec{A}}{\vec{m}} = \vec{a}_{total} - \vec{g} = \dot{\vec{V}} - \vec{g} = \ddot{\vec{R}} - \vec{g} \quad (6.22)$$

Note that the sensed acceleration components can be written either in terms of thrust, aerodynamic forces, and mass, or just the velocity and/or position (with use of the D or D^2 matrices). Writing it in terms of just velocity and/or position simplifies the calculation of the Jacobian of the sensed acceleration constraint. The components of the sensed acceleration are:

- **D-Method**

$$\begin{aligned}
 (\vec{a}_{\text{sensed}})_x &= D_{NN} \vec{V}_x - \vec{g}_x \\
 (\vec{a}_{\text{sensed}})_y &= D_{NN} \vec{V}_y - \vec{g}_y \\
 (\vec{a}_{\text{sensed}})_z &= D_{NN} \vec{V}_z - \vec{g}_z
 \end{aligned} \tag{6.23}$$

- **D^2 -Method**

$$\begin{aligned}
 (\vec{a}_{\text{sensed}})_x &= D_{NN}^2 \vec{R}_x - \vec{g}_x \\
 (\vec{a}_{\text{sensed}})_y &= D_{NN}^2 \vec{R}_y - \vec{g}_y \\
 (\vec{a}_{\text{sensed}})_z &= D_{NN}^2 \vec{R}_z - \vec{g}_z
 \end{aligned} \tag{6.24}$$

The sensed acceleration constraint is:

$$\vec{C}_a = (\vec{a}_{\text{sensed}})_x^2 + (\vec{a}_{\text{sensed}})_y^2 + (\vec{a}_{\text{sensed}})_z^2 \leq a_{\max}^2 \tag{6.25}$$

6.2.4 Mass Change Constraint

It was noted that sometimes, when small numbers of LGL points were used, the mass flow rate would remain negative while the mass itself would slightly increase. To guarantee that the mass never increases, a mass change constraint is added.

$$(C_{\Delta m})_i = m_{i+1} - m_i \leq 0 \quad (i = 1 \dots n_{LGL} - 1) \tag{6.26}$$

Note that the length of this constraint is equal to $n_{LGL} - 1$, instead of n_{LGL} like the other trajectory constraints.

6.3 Initial and Final Constraints

These are constraints on the initial and final points of the trajectory. The initial state of the vehicle is assumed to be completely specified. For the launch problem,

the final constraints are essentially the burnout conditions that define the target orbit. For this model, the final altitude, velocity magnitude, flight path angle, and orbital inclination are constrained.

It is important to note that the initial constraints are functions of the initial phase variables only. Also, the final constraints are functions of the final phase variables only. This is important to remember when considering the partial derivatives of the initial and final constraints.

6.3.1 Initial Constraints

When using the D -method, it is possible to constrain the initial state by simply bounding the initial state variables in the optimization vector. Therefore, no additional constraints are required to set the initial state. This can also be done when using the D^2 -method, with the exception of the velocity. For the D^2 method, the velocity is not part of the optimization vector, and therefore can not be controlled by bounds on the optimization vector. Additional initial constraints are required.

D^2 -Method Initial Velocity Constraints

The initial velocity constraints for the D^2 -method are:

$$C_{V_x} = d_{NN_1} \vec{R}_x = V_x \quad (6.27)$$

$$C_{V_y} = d_{NN_1} \vec{R}_y = V_y \quad (6.28)$$

$$C_{V_z} = d_{NN_1} \vec{R}_z = V_z \quad (6.29)$$

where d_{NN_1} is the first row of the D_{NN} matrix corresponding to the initial time phase.

6.3.2 Final Altitude Constraint

This is a constraint on the final distance from the center of the planet.

$$C_{R_f} = R_{x_f}^2 + R_{y_f}^2 + R_{z_f}^2 = R_{mag_f}^2 \quad (6.30)$$

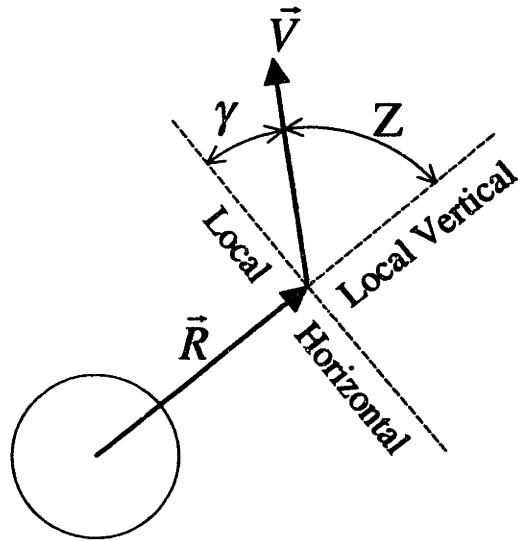


Figure 6-2: Flight Path Angle and Zenith Angle (adapted from [2, p. 17])

6.3.3 Final Velocity Constraint

This is a constraint on the final velocity magnitude.

$$C_{V_f} = V_{x_f}^2 + V_{y_f}^2 + V_{z_f}^2 = V_{mag_f}^2 \quad (6.31)$$

6.3.4 Final Flight Path Angle Constraint

The flight path angle (γ) is defined in figure 6-2. It is the angle that the velocity makes from the local horizontal plane. The zenith angle (Z) is also defined. From the figure, it can be seen that:

$$\vec{R}_f \cdot \vec{V}_f = R_{mag_f} V_{mag_f} \cos(Z) = R_{mag_f} V_{mag_f} \sin(\gamma_f) \quad (6.32)$$

The final flight path angle constraint is written as:

$$C_{\gamma_f} = R_{x_f} V_{x_f} + R_{y_f} V_{y_f} + R_{z_f} V_{z_f} = R_{mag_f} V_{mag_f} \sin(\gamma_f) \quad (6.33)$$

6.3.5 Final Inclination Constraint

The angular momentum vector (\vec{h}) of an orbit is perpendicular to the plane of the orbit. It is defined as [2, p. 17]

$$\vec{h} = \vec{R} \times \vec{V} = \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ R_x & R_y & R_z \\ V_x & V_y & V_z \end{vmatrix}$$

$$\vec{h} = (R_y V_z - R_z V_y) \hat{i} + (R_z V_x - R_x V_z) \hat{j} + (R_x V_y - R_y V_x) \hat{k} \quad (6.34)$$

The magnitude of the angular momentum of the orbit can be written as [2, p. 18]

$$h_{mag} = R_{mag} V_{mag} \sin(Z) = R_{mag} V_{mag} \cos(\gamma) \quad (6.35)$$

The orbital inclination is defined as the angle between the angular momentum vector and the inertial Z-axis, where the Z-axis is along the axis of rotation of the planet [2, p. 63].

$$\vec{h} \cdot \hat{k} = h_z = h_{mag} \cos(i) \quad (6.36)$$

The final orbital inclination constraint is written as:

$$C_{incl} = R_{x_f} V_{y_f} - R_{y_f} V_{x_f} = R_{mag_f} V_{mag_f} \cos(\gamma_f) \cos(i_f) \quad (6.37)$$

6.4 Staging Constraints

The vehicle has at least two stages. This requires the modelling of a discontinuity in mass when the first stage is dropped. As discussed in section 3.2, each stage will require a phase, with a hard knot between each of the phases. In this model, the staging constraints apply only to the state variables (position, velocity, and mass). Continuity constraints are applied to the position and velocity constraints across the knot. The knotting constraints for the knot between the first phase and the second

phase are:

$$C_{knot_{R_x}} = (R_{x_1})_{P_2} - (R_{x_{nLGL_1}})_{P_1} = 0 \quad (6.38)$$

$$C_{knot_{R_y}} = (R_{y_1})_{P_2} - (R_{y_{nLGL_1}})_{P_1} = 0 \quad (6.39)$$

$$C_{knot_{R_z}} = (R_{z_1})_{P_2} - (R_{z_{nLGL_1}})_{P_1} = 0 \quad (6.40)$$

$$C_{knot_{V_x}} = (V_{x_1})_{P_2} - (V_{x_{nLGL_1}})_{P_1} = 0 \quad (6.41)$$

$$C_{knot_{V_y}} = (V_{y_1})_{P_2} - (V_{y_{nLGL_1}})_{P_1} = 0 \quad (6.42)$$

$$C_{knot_{V_z}} = (V_{z_1})_{P_2} - (V_{z_{nLGL_1}})_{P_1} = 0 \quad (6.43)$$

The subscript 1 refers to the initial point in a phase. The subscript n_{LGL_i} refers to the final point in phase i (i.e. there are n_{LGL_i} points in phase i). The subscript P_i refers to phase i .

The discontinuity in mass at the staging point can be set by choosing proper bounds on the optimization vector. Thus, no additional knotting constraints are required for mass. Since the staging times are free, it is necessary to put knotting constraints on the staging times to make sure that time always moves forward.

$$\begin{aligned} \tau_{P_1} - \tau_o &\geq 0 \\ \tau_{P_2} - \tau_{P_1} &\geq 0 \\ &\vdots \\ \tau_{P_n} - \tau_{P_{n-1}} &\geq 0 \\ \tau_f - \tau_{P_n} &\geq 0 \end{aligned} \quad (6.44)$$

6.5 Optimization Vector

Assume that the launch vehicle has n stages. Each stage of the launch vehicle has a corresponding time phase. Since it is assumed that the launch vehicle has at least two stages, there will be at least two time phases. See section 3.2 for more details on the staging method. Define a row vector of states and controls in phase i . Call

this vector \vec{x}_{phase_i} . All of the constraints defined in the previous sections apply to all of the phases. Next, define a row vector of all the staging times and the final time. A staging time is defined as the time when stage i is dropped. Note that the initial time is not free, so it is not included in the optimization vector.

$$\vec{x}_{times} = \begin{bmatrix} \tau_{stage_1} & \tau_{stage_2} & \cdots & \tau_{stage_{n-1}} & \tau_f \end{bmatrix} \quad (6.45)$$

The total optimization vector is:

$$\vec{x}_{opt} = \left[\vec{x}_{phase_1} \quad \vec{x}_{phase_2} \quad \cdots \quad \vec{x}_{phase_n} \quad \vec{x}_{times} \right]^T \quad (6.46)$$

6.5.1 D-Method Optimization Vector

The vector of states and controls for one phase is:

$$\vec{x}_{phase_i} = \left[\vec{R}_x^T \quad \vec{R}_y^T \quad \vec{R}_z^T \quad \vec{V}_x^T \quad \vec{V}_y^T \quad \vec{V}_z^T \quad \vec{m}^T \quad \vec{T}^T \quad \vec{q}_1^T \quad \vec{q}_2^T \quad \vec{q}_3^T \quad \vec{q}_4^T \right]_i \quad (6.47)$$

where

- \vec{R}_x = vector of inertial X-positions at LGL time points
- \vec{R}_y = vector of inertial Y-positions at LGL time points
- \vec{R}_z = vector of inertial Z-positions at LGL time points
- \vec{V}_x = vector of inertial X-velocities at LGL time points
- \vec{V}_y = vector of inertial Y-velocities at LGL time points
- \vec{V}_z = vector of inertial Z-velocities at LGL time points
- \vec{m} = vector of masses at LGL time points
- \vec{T} = vector of thrust magnitudes at LGL time points
- \vec{q}_1 = vector of first quaternion elements at LGL time points
- \vec{q}_2 = vector of second quaternion elements at LGL time points
- \vec{q}_3 = vector of third quaternion elements at LGL time points
- \vec{q}_4 = vector of fourth quaternion elements at LGL time points
- τ_f = the free final time

6.5.2 D^2 -Method Optimization Vector

The vector of states and controls for one phase is:

$$\vec{x}_{phase_i} = \left[\vec{R}_x^T \quad \vec{R}_y^T \quad \vec{R}_z^T \quad \vec{m}^T \quad \vec{T}^T \quad \vec{q}_1^T \quad \vec{q}_2^T \quad \vec{q}_3^T \quad \vec{q}_4^T \right]_i \quad (6.48)$$

6.5.3 Optimization Vector Bounds

The lower and upper bounds on the optimization vector must fulfill the following requirements:

- The initial state is fixed.
- For any phase, the initial mass must be equal to:

$$\sum_{stage\#}^{n_P} m_i \quad (6.49)$$

where

$$\begin{aligned} m_i &= \text{the total mass of stage } i \\ n_P &= \text{the number of stages} \\ stage\# &= \text{the current stage} \end{aligned}$$

During phase i , the mass can never go above the initial mass for that stage.

Also, the mass can not go below

$$m_{stage\#} \left(1 - f_{P_{stage\#}} \right) + \sum_{(stage\#+1)}^{n_P} m_i \quad (6.50)$$

where

f_{P_i} = the fraction of the stage i mass that is propellant

- The thrust magnitude can not exceed the maximum thrust, T_{max} .
- The initial time is zero. The staging and final times are free.

The general bounds on the optimization variables are:

$$\begin{aligned}
-\infty &\leq \vec{R}_x \leq \infty \\
-\infty &\leq \vec{R}_y \leq \infty \\
-\infty &\leq \vec{R}_z \leq \infty \\
-\infty &\leq \vec{V}_x \leq \infty \\
-\infty &\leq \vec{V}_y \leq \infty \\
-\infty &\leq \vec{V}_z \leq \infty \\
m_{stage\#} (1 - f_{P_{stage\#}}) + \sum_{(stage\#+1)}^{n_p} m_i &\leq \vec{m} \leq \sum_{stage\#}^{n_p} m_i \\
0 &\leq \vec{T} \leq T_{max} \\
-1 &\leq \vec{q}_1 \leq 1 \\
-1 &\leq \vec{q}_2 \leq 1 \\
-1 &\leq \vec{q}_3 \leq 1 \\
-1 &\leq \vec{q}_4 \leq 1 \\
0 &\leq \tau_{stage\#} \leq \infty \\
0 &\leq \tau_f \leq \infty
\end{aligned} \tag{6.51}$$

The exception to these bounds is at the initial phase points. The initial (i.e. at time τ_0) states and controls are defined by setting the initial upper and lower bounds to the same values. The same thing is done for the mass at the staging points.

6.6 Definition of Three Dimensional Model

6.6.1 D^2 -Method Velocity

As stated in equations 6.14, for the D^2 -method, the velocities are defined to be exactly:

$$\vec{V}_x = D_{NN} \vec{R}_x \tag{6.52}$$

$$\vec{V}_y = D_{NN} \vec{R}_y \tag{6.53}$$

$$\vec{V}_z = D_{NN} \vec{R}_z \tag{6.54}$$

The velocity magnitude is:

$$\vec{V}_{mag} = \sqrt{\vec{V}_x^2 + \vec{V}_y^2 + \vec{V}_z^2} \quad (6.55)$$

Partial Derivatives of D^2 -Method Velocity

The partial derivatives for the D^2 -method velocity are derived in Appendix G.5.1.

6.6.2 Q Attitude Matrix and Euler Angles

The Q attitude matrix is defined by the four quaternion elements. The attitude matrix is defined as the transformation matrix from the vehicle body coordinates to the inertial coordinates. So,

$$\vec{r}_I = Q \vec{r}_B \quad (6.56)$$

where

\vec{r}_I = a vector in inertial coordinates

\vec{r}_B = a vector in vehicle body coordinates

Q Attitude Matrix

The attitude matrix can be formed from the four quaternion elements. The relationship is [20]:

$$Q = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_1q_2 - q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2q_3 + q_1q_4) \\ 2(q_1q_3 + q_2q_4) & 2(q_2q_3 - q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \quad (6.57)$$

Note that there is a Q matrix associated with each LGL point. A very useful property of the attitude matrix is that its columns are orthonormal. This means that the inverse of the attitude matrix is equal to the transpose of the attitude matrix.

$$Q^{-1} = Q^T \quad (6.58)$$

This is useful when transforming vectors from the inertial reference frame to the body reference frame.

The quaternion elements must satisfy the following normalization constraint [20]:

$$q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1 \quad (6.59)$$

It is possible to find the quaternion elements from the attitude matrix. Let the attitude matrix be written as:

$$Q = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} \\ Q_{21} & Q_{22} & Q_{23} \\ Q_{31} & Q_{32} & Q_{33} \end{bmatrix} \quad (6.60)$$

The following four sets of equations can be used to find the quaternion elements [20]:

Set 1

$$\begin{aligned} q_1 &= \pm \frac{1}{2} \sqrt{1 + Q_{11} - Q_{22} - Q_{33}} \\ q_2 &= \frac{1}{4q_1} (Q_{12} + Q_{21}) \\ q_3 &= \frac{1}{4q_1} (Q_{13} + Q_{31}) \\ q_4 &= \frac{1}{4q_1} (Q_{23} - Q_{32}) \end{aligned} \quad (6.61)$$

Set 2

$$\begin{aligned} q_2 &= \pm \frac{1}{2} \sqrt{1 - Q_{11} + Q_{22} - Q_{33}} \\ q_1 &= \frac{1}{4q_2} (Q_{21} + Q_{12}) \\ q_3 &= \frac{1}{4q_2} (Q_{23} + Q_{32}) \\ q_4 &= \frac{1}{4q_2} (Q_{31} - Q_{13}) \end{aligned} \quad (6.62)$$

Set 3

$$\begin{aligned} q_3 &= \pm \frac{1}{2} \sqrt{1 - Q_{11} - Q_{22} + Q_{33}} \\ q_1 &= \frac{1}{4q_3} (Q_{31} + Q_{13}) \\ q_2 &= \frac{1}{4q_3} (Q_{32} + Q_{23}) \\ q_4 &= \frac{1}{4q_3} (Q_{12} - Q_{21}) \end{aligned} \tag{6.63}$$

Set 4

$$\begin{aligned} q_4 &= \pm \frac{1}{2} \sqrt{1 + Q_{11} + Q_{22} + Q_{33}} \\ q_1 &= \frac{1}{4q_4} (Q_{23} - Q_{32}) \\ q_2 &= \frac{1}{4q_4} (Q_{31} - Q_{13}) \\ q_3 &= \frac{1}{4q_4} (Q_{12} - Q_{21}) \end{aligned} \tag{6.64}$$

Generally, it is best to use the set of equations that yields the largest term in the denominator. This helps with numerical accuracy. Note that there is a sign ambiguity. It turns out that the ambiguous sign is not important [20]. It can be chosen as either positive or negative.

Partial Derivatives of Q Attitude Matrix

The partial derivatives for the Q attitude matrix are derived in Appendix G.5.2

Euler Angles

Euler angles are not used directly in the simulation, but they can be helpful in visualizing the attitude of the vehicle in inertial space. The following Euler angles are used to visualize a transformation from the inertial coordinates to the body coordinates.

Ψ = rotation about inertial Z-axis

Θ = rotation about new Y-axis

Φ = rotation about new X-axis

Imagine that the body coordinates are initially aligned with the inertial coordinates. Three rotations of the body coordinates are used to put the body coordinates in the proper attitude with respect to the inertial coordinates. The order of the rotations is very important. The first rotation, Ψ , is about the inertial Z-axis (because the body Z-axis and inertial Z-axis are initially aligned). The second rotation, Θ , is about the new (from the first rotation) body Y-axis. The final rotation, Φ , is about the new (from the second rotation) body X-axis. The attitude transformation matrix from body to inertial coordinates is [24]:

$$A_{I/B} = \begin{bmatrix} \cos \Psi \cos \Theta & -\sin \Psi \cos \Phi + \cos \Psi \sin \Theta \sin \Phi & \sin \Psi \sin \Phi + \cos \Psi \sin \Theta \cos \Phi \\ \sin \Psi \cos \Theta & \cos \Psi \cos \Phi + \sin \Psi \sin \Theta \sin \Phi & -\cos \Psi \sin \Phi + \sin \Psi \sin \Theta \cos \Phi \\ -\sin \Theta & \cos \Theta \sin \Phi & \cos \Theta \cos \Phi \end{bmatrix} \quad (6.65)$$

This matrix transforms vectors from the body coordinates to the inertial coordinates.

So,

$$\vec{r}_I = A_{I/B} \vec{r}_B \quad (6.66)$$

It can be seen that $A_{I/B}$ is equivalent to the Q attitude matrix.

The Euler angles can be found from the attitude matrix. As before, let the attitude

matrix be written as:

$$A_{I/B} = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} \\ Q_{21} & Q_{22} & Q_{23} \\ Q_{31} & Q_{32} & Q_{33} \end{bmatrix} \quad (6.67)$$

The Euler angles can be found from:

$$\Psi = \tan^{-1} \left(\frac{Q_{21}}{Q_{11}} \right) \quad (6.68)$$

$$\Theta = \sin^{-1} (-Q_{31}) \quad (6.69)$$

$$\Phi = \tan^{-1} \left(\frac{Q_{32}}{Q_{33}} \right) \quad (6.70)$$

Care must be taken to preserve the correct quadrant of the Euler angles. In practice, the MATLAB command “atan2” is used for this purpose.

6.6.3 Thrust

The inertial thrust vector is defined by the thrust magnitude, the Q attitude matrix, and the thrust direction in the body coordinates. At one LGL point,

$$\vec{T}_I = T [Q] \hat{u}_{T_B} \quad (6.71)$$

where

\vec{T}_I = inertial thrust vector at one LGL point

T = thrust magnitude at one LGL point

$[Q]$ = Q attitude matrix at one LGL point

\hat{u}_{T_B} = unit thrust vector in body coordinates

For this model, the thrust direction is assumed to be constant in the body frame. However, it would be easy to add thrust gimbal by allowing the body thrust direction to vary. This could be accomplished by allowing the three components of the body thrust direction to be included in the optimization vector. Also, it might be possible to define two gimbal angles from which the body thrust direction could be defined. The two gimbal angles could then be added to the optimization vector.

At one LGL point, the components of the thrust can be found from:

$$\begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}_I = T [Q] \begin{bmatrix} u_{T_x} \\ u_{T_y} \\ u_{T_z} \end{bmatrix}_B \quad (6.72)$$

However, note that the Q matrix is different at each LGL point. The thrust components at each LGL point can be written in vector form as:

$$\begin{aligned} \vec{T}_x &= \vec{T} (\vec{Q}_{11} u_{T_x} + \vec{Q}_{12} u_{T_y} + \vec{Q}_{13} u_{T_z}) \\ \vec{T}_y &= \vec{T} (\vec{Q}_{21} u_{T_x} + \vec{Q}_{22} u_{T_y} + \vec{Q}_{23} u_{T_z}) \\ \vec{T}_z &= \vec{T} (\vec{Q}_{31} u_{T_x} + \vec{Q}_{32} u_{T_y} + \vec{Q}_{33} u_{T_z}) \end{aligned} \quad (6.73)$$

where the \vec{Q} 's are the elements of the Q matrix at the LGL points.

Partial Derivatives of Thrust

The partial derivatives for the thrust are derived in Appendix G.5.3

6.6.4 Gravity

For this problem, the planet is assumed to be perfectly spherical. Thus, the gravitational acceleration will be such that it follows Newton's law of gravitation with all of the planet's mass concentrated at the planet center. Also, the gravity will always act in the opposite direction of the position vector from the center of the planet to the vehicle. The gravitational acceleration at one LGL point is:

$$\vec{g} = \frac{\mu}{|\vec{r}|^2} \left(-\frac{\vec{r}}{|\vec{r}|} \right) \quad (6.74)$$

where μ is the gravitational parameter of the planet.

The components of the gravity along each axis at each LGL point are:

$$\vec{g}_x = \frac{-\mu \vec{R}_x}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{3/2}} \quad (6.75)$$

$$\vec{g}_y = \frac{-\mu \vec{R}_y}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{3/2}} \quad (6.76)$$

$$\vec{g}_z = \frac{-\mu \vec{R}_z}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{3/2}} \quad (6.77)$$

Partial Derivatives of Gravity

The partial derivatives for the gravity are derived in Appendix G.5.4

6.6.5 Aerodynamic Force

The aerodynamic force is defined in terms of the dynamic pressure, the inertial aerodynamic coefficients, and the reference area. The aerodynamic force can be written in vector notation (for one LGL point) :

$$\vec{A} = q \vec{C}_I A_{ref} \quad (6.78)$$

where

\vec{A} = aerodynamic forces in inertial coordinates

q = dynamic pressure

\vec{C}_I = vector of aerodynamic coefficients along inertial coordinates

A_{ref} = reference area of vehicle

This can also be written for all the LGL points:

$$\begin{aligned} \vec{A}_x &= \vec{q} \vec{C}_{I_x} A_{ref} \\ \vec{A}_y &= \vec{q} \vec{C}_{I_y} A_{ref} \\ \vec{A}_z &= \vec{q} \vec{C}_{I_z} A_{ref} \end{aligned} \quad (6.79)$$

Partial Derivatives of Aerodynamic Force

The partial derivatives for the aerodynamic force are derived in Appendix G.5.5

6.6.6 Dynamic Pressure

The dynamic pressure is given by:

$$\vec{q} = \frac{1}{2} \vec{\rho} \vec{V}_{rel}^2 \quad (6.80)$$

where

$\vec{\rho}$ = vector of atmospheric pressures at each LGL point

\vec{V}_{rel} = vector of wind relative velocity magnitudes at each LGL point

Partial Derivatives of Dynamic Pressure

The partial derivatives for the dynamic pressure are derived in Appendix G.5.6

6.6.7 Wind Relative Velocity

For this problem, wind is not modelled. This means that the atmosphere is assumed to be stationary with respect to the rotating Earth. The velocity of the atmosphere in Earth-centered inertial coordinates is then defined by:

$$(\vec{V}_{atmosphere})_I = \vec{\Omega}_P \times \vec{r}_I = \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ \Omega_x & \Omega_y & \Omega_z \\ R_x & R_y & R_z \end{vmatrix} = \begin{bmatrix} R_z\Omega_y - R_y\Omega_z \\ R_x\Omega_z - R_z\Omega_x \\ R_y\Omega_x - R_x\Omega_y \end{bmatrix} \quad (6.81)$$

where

$(\vec{V}_{atmosphere})_I$ = inertial velocity of atmosphere without wind

$\vec{\Omega}_P$ = inertial rotation rate of the planet

\vec{r}_I = inertial position vector at some point in the atmosphere

Relative Velocity

The velocity of the vehicle with respect to the atmosphere is called "wind relative velocity" or simply "relative velocity." It is given by:

$$\vec{V}_{rel} = \vec{V}_I - (\vec{V}_{atmosphere})_I \quad (6.82)$$

The components of the relative velocity at the LGL points are:

$$\vec{V}_{rel_x} = \vec{V}_x - \vec{R}_z \Omega_y + \vec{R}_y \Omega_z \quad (6.83)$$

$$\vec{V}_{rel_y} = \vec{V}_y - \vec{R}_x \Omega_z + \vec{R}_z \Omega_x \quad (6.84)$$

$$\vec{V}_{rel_z} = \vec{V}_z - \vec{R}_y \Omega_x + \vec{R}_x \Omega_y \quad (6.85)$$

Magnitude of Relative Velocity

The magnitude of the relative velocity is:

$$\vec{V}_{rel} = (\vec{V}_{rel_x}^2 + \vec{V}_{rel_y}^2 + \vec{V}_{rel_z}^2)^{1/2} \quad (6.86)$$

Partial Derivatives of Relative Velocity Components and Magnitude

The partial derivatives for the relative velocity components and magnitude are derived in Appendix G.5.7

6.6.8 Body Velocity

The wind relative velocity along the body axes can be found from the wind relative velocity in inertial coordinates and the attitude matrix (defined by the quaternion elements). From equation 6.58, it is known that:

$$Q^{-1} = Q^T \quad (6.87)$$

Converting the inertial wind relative velocity to wind relative velocity in body coordinates can be accomplished by using the transpose of the attitude matrix.

$$\begin{bmatrix} V_{rel_x} \\ V_{rel_y} \\ V_{rel_z} \end{bmatrix}_B = Q^T \begin{bmatrix} V_{rel_x} \\ V_{rel_y} \\ V_{rel_z} \end{bmatrix}_I \quad (6.88)$$

The wind relative velocity components in the body coordinates at each LGL point are given by:

$$\begin{aligned}\vec{V}_{B_x} &= \vec{Q}_{11}\vec{V}_{rel_x} + \vec{Q}_{21}\vec{V}_{rel_y} + \vec{Q}_{31}\vec{V}_{rel_z} \\ \vec{V}_{B_y} &= \vec{Q}_{12}\vec{V}_{rel_x} + \vec{Q}_{22}\vec{V}_{rel_y} + \vec{Q}_{32}\vec{V}_{rel_z} \\ \vec{V}_{B_z} &= \vec{Q}_{13}\vec{V}_{rel_x} + \vec{Q}_{23}\vec{V}_{rel_y} + \vec{Q}_{33}\vec{V}_{rel_z}\end{aligned}\quad (6.89)$$

Partial Derivatives of Body Velocity

The partial derivatives for the relative velocity components are derived in Appendix G.5.8

6.6.9 Inertial Aerodynamic Coefficients

The aerodynamic coefficients along the inertial axes can be found from the body aerodynamic coefficients and the attitude matrix.

$$\begin{bmatrix} C_{I_x} \\ C_{I_y} \\ C_{I_z} \end{bmatrix} = [Q] \begin{bmatrix} C_{B_x} \\ C_{B_y} \\ C_{B_z} \end{bmatrix}$$

The inertial aerodynamic coefficients at the LGL points are:

$$\begin{aligned}\vec{C}_{I_x} &= \vec{Q}_{11}\vec{C}_{B_x} + \vec{Q}_{12}\vec{C}_{B_y} + \vec{Q}_{13}\vec{C}_{B_z} \\ \vec{C}_{I_y} &= \vec{Q}_{21}\vec{C}_{B_x} + \vec{Q}_{22}\vec{C}_{B_y} + \vec{Q}_{23}\vec{C}_{B_z} \\ \vec{C}_{I_z} &= \vec{Q}_{31}\vec{C}_{B_x} + \vec{Q}_{32}\vec{C}_{B_y} + \vec{Q}_{33}\vec{C}_{B_z}\end{aligned}\quad (6.90)$$

Partial Derivatives of Inertial Aerodynamic Coefficients

The partial derivatives for the inertial aerodynamic coefficients are derived in Appendix G.5.9

6.6.10 Mach Number

The Mach number is the ratio of the wind relative velocity magnitude over the local atmospheric speed of sound.

$$\tilde{M} = \frac{\tilde{V}_{rel}}{\tilde{a}_\infty} \quad (6.91)$$

where

\tilde{V}_{rel} = vector of wind relative velocity magnitudes at the LGL points

\tilde{a}_∞ = vector of local atmospheric speeds of sound at the LGL points

Partial Derivatives of Mach Number

The partial derivatives for the Mach number are derived in Appendix G.5.10

6.6.11 Aerodynamic Angles

Figure 6-3 was adapted from [14]. It shows the relationship between the wind relative

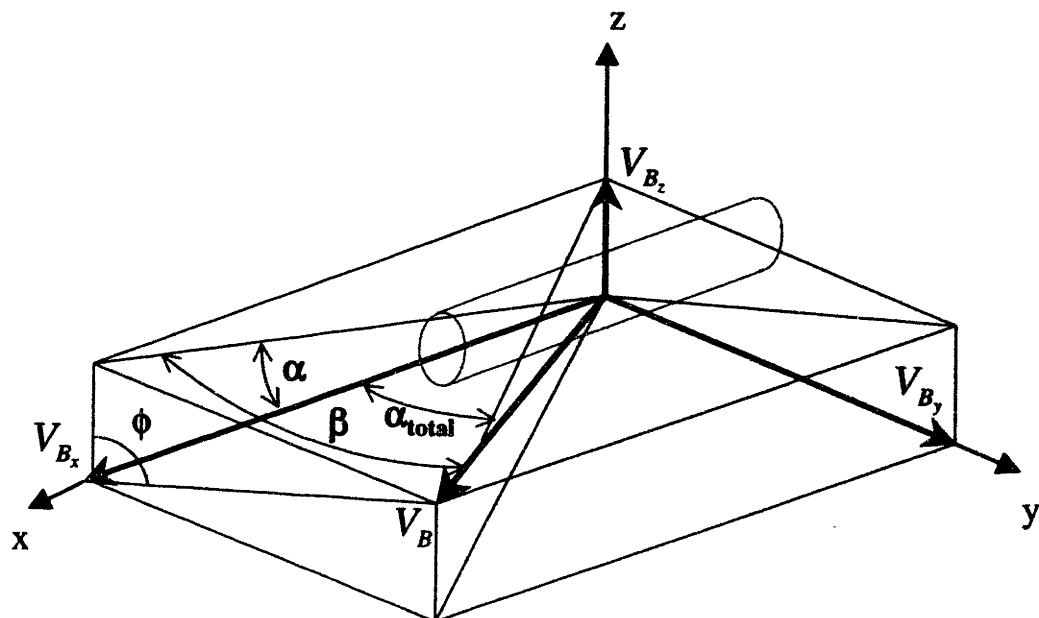


Figure 6-3: Aerodynamic Angles

Total Angle of Attack (α_{total})

The total angle of attack (α_{total}) is defined as the smallest angle between the velocity vector and the body X-axis. This angle lies in a plane formed by the body X-axis and the velocity vector. From Figure 6-3, it can be seen that:

$$\sin(\alpha_{total}) = \frac{\sqrt{V_{B_y}^2 + V_{B_z}^2}}{\sqrt{V_{B_x}^2 + V_{B_y}^2 + V_{B_z}^2}} \quad (6.92)$$

$$\cos(\alpha_{total}) = \frac{V_{B_x}}{\sqrt{V_{B_x}^2 + V_{B_y}^2 + V_{B_z}^2}} \quad (6.93)$$

$$\tan(\alpha_{total}) = \frac{\sqrt{V_{B_y}^2 + V_{B_z}^2}}{V_{B_x}} \quad (6.94)$$

Any of these expressions may be used to define the total angle of attack.

Angle of Attack (α)

The angle of attack (α) is defined as the angle from the body X-axis to the projection of the velocity onto the body XZ-plane. From Figure 6-3, it can be seen that:

$$\sin(\alpha) = \frac{V_{B_z}}{\sqrt{V_{B_x}^2 + V_{B_z}^2}} \quad (6.95)$$

$$\cos(\alpha) = \frac{V_{B_x}}{\sqrt{V_{B_x}^2 + V_{B_z}^2}} \quad (6.96)$$

$$\tan(\alpha) = \frac{V_{B_z}}{V_{B_x}} \quad (6.97)$$

Sideslip Angle (β)

The sideslip angle (β) is defined as the angle from the body XZ-plane to a plane perpendicular to the body XY-plane and parallel to the velocity vector. From Figure 6-3, it can be seen that:

$$\sin(\beta) = \frac{V_{B_y}}{\sqrt{V_{B_x}^2 + V_{B_y}^2 + V_{B_z}^2}} \quad (6.98)$$

$$\cos(\beta) = \frac{\sqrt{V_{B_x}^2 + V_{B_z}^2}}{\sqrt{V_{B_x}^2 + V_{B_y}^2 + V_{B_z}^2}} \quad (6.99)$$

$$\tan(\beta) = \frac{V_{B_y}}{\sqrt{V_{B_x}^2 + V_{B_z}^2}} \quad (6.100)$$

Roll Angle (ϕ)

The roll angle (ϕ) is defined as the angle from the body Z-axis to the projection of the velocity into the body YZ-plane. From Figure 6-3, it can be seen that:

$$\sin(\phi) = \frac{V_{B_y}}{\sqrt{V_{B_y}^2 + V_{B_z}^2}} \quad (6.101)$$

$$\cos(\phi) = \frac{V_{B_z}}{\sqrt{V_{B_y}^2 + V_{B_z}^2}} \quad (6.102)$$

$$\tan(\phi) = \frac{V_{B_y}}{V_{B_z}} \quad (6.103)$$

Partial Derivatives of Aerodynamic Angles

The partial derivatives of α and β are derived in Appendix G.5.11. Note that the partial derivatives of α_{total} and ϕ are not used in this thesis.

6.6.12 Body Aerodynamic Coefficients

The axial force coefficient (C_A) defines the force applied along the axis of an axisymmetric vehicle. The normal force coefficient (C_N) defines the force that is perpendicular to the axis of the vehicle and in the plane formed by the velocity vector and the body axis. C_A and C_N , along with the roll angle (ϕ) defined in section 6.6.11, can be used to write the aerodynamic force coefficients along the body axes:

$$\begin{aligned} C_{B_x} &= -C_A \\ C_{B_y} &= -C_N \sin(\phi) \\ C_{B_z} &= -C_N \cos(\phi) \end{aligned} \quad (6.104)$$

C_A and C_N are typically given as functions of the total angle of attack and Mach number. Also, this data is usually given in tabular form. Thus, it is necessary to interpolate when values between the data points are necessary. This can be a concern, especially when the derivatives of the tabular data are required.

From the definition of roll angle (equations 6.101, 6.102, and 6.103), it can be seen that the roll angle is undefined when V_{B_y} and V_{B_z} are zero. When this occurs, the velocity is aligned with the axis of the vehicle and both C_{B_y} and C_{B_z} are zero. So, the problem of an undefined roll angle at zero total angle of attack seems to go away. However, the partial derivatives of the body aerodynamic force coefficients are needed for the numerical optimizer. Assuming that the body velocities are all functions of some variable x , the derivative of the roll angle with respect to x is undefined when V_{B_y} and V_{B_z} are zero.

For this simulation, the following assumption is made:

Assumption: The vehicle travels such that some part of the velocity is always along the body x-axis. This means that the body x-velocity is never zero.

This is not a bad assumption. The body x-axis is chosen to be along the preferred direction of travel of the vehicle. However, this means that the simulation can not start with the vehicle at rest. This is not bad either, since most launch vehicles must initially launch straight up in order to clear their launch towers.

With this assumption, it can be seen from the equations in section 6.6.11 that α , β , and the derivatives of α and β will always be continuous. Given C_A and C_N as numerical functions of α_{total} and Mach number, it is possible to write C_{B_x} , C_{B_y} , and C_{B_z} as functions of α , β , and Mach number.

Spherical trigonometry can be used to find relationships for α_{total} and ϕ in terms of α and β . The spherical right triangle of figure 6-4 is extracted from figure 6-3. The following relationships hold between the angles [25, p. 468]:

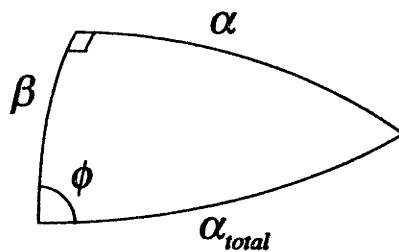


Figure 6-4: Right Spherical Triangle formed by Aerodynamic Angles

$$\cos(\alpha_{total}) = \cos(\alpha) \cos(\beta) \quad (6.105)$$

$$\tan(\phi) = \frac{\tan(\beta)}{\sin(\alpha)} \quad (6.106)$$

It is now possible to transform tabular data for C_A and C_N into tabular data for C_{B_x} , C_{B_y} and C_{B_z} . The method used in this thesis is as follows:

1. A grid of α and β points is set-up.
2. The α_{total} and ϕ that correspond to each α - β point is computed using equations 6.105 and 6.106.
3. For a given Mach number, interpolation is used to find the value of C_A and C_N at all the α - β points.
4. The interpolated values of C_A and C_N and the corresponding roll angle are used in equations 6.104 to calculate the values of C_{B_x} , C_{B_y} and C_{B_z} at the α - β points.

Simplified Body Force Coefficients

The body force coefficients can be simplified with the following assumptions:

1. C_{B_x} is constant
2. C_{B_y} is linearly dependent on β
3. C_{B_z} is linearly dependent on α

For an axisymmetric vehicle, the last two assumptions are not bad for small α and β . The first assumption is only good if α and β do not change. All three assumptions break down when the Mach number varies greatly, which happens during a launch. These simplified coefficients are used to help “jump-start” a solution. When optimizing a solution using tabular aerodynamic data, the numerical interpolation and calculation of the coefficient derivatives can slow down the process. The simplified coefficients can be used to find a good initial guess. This initial guess can then be used to find a solution using tabular aerodynamic data. This process is discussed further in Chapter 7. The simplified body force coefficients are:

$$\begin{aligned}\vec{C}_{B_x} &= \text{constant} \\ \vec{C}_{B_y} &= \left(\frac{dC_{B_y}}{d\beta} \right) \vec{\beta} \\ \vec{C}_{B_z} &= \left(\frac{dC_{B_z}}{d\alpha} \right) \vec{\alpha}\end{aligned}\tag{6.107}$$

where $\left(\frac{dC_{B_y}}{d\beta} \right)$ and $\left(\frac{dC_{B_z}}{d\alpha} \right)$ are constants.

Partial Derivatives of Body Aerodynamic Coefficients

The partial derivatives of the body aerodynamic coefficients are derived in Appendix G.5.12.

6.6.13 Atmosphere Model

The atmosphere model used for the three dimensional model is given in Appendix C.2. This atmosphere model gives the atmospheric density, pressure, temperature, and speed of sound as a function of altitude. The partial derivatives of atmospheric density, pressure, temperature, and speed of sound with respect to the optimization vector of the three dimensional cartesian model are derived in Appendix G.5.13.

Note that for the three dimensional model of this chapter, only the atmospheric density and speed of sound are required. However, the atmospheric pressure and temperature may be necessary if a heating rate constraint or backpressure effects on engine thrust are added to the model.

Chapter 7

Three Dimensional Trajectory Optimization

7.1 Vehicle Definition

A fictional launch vehicle loosely based on the Kistler K-1 launch vehicle is used for the trajectory optimization of this section. The K-1 is a reusable, two-stage launch vehicle being developed by the Kistler Aerospace Corporation. Although the first stage of the K-1 is designed to fly-back to the launch site, this fly-back portion is not modelled in this thesis. Thus, the vehicle is treated as a traditional launch vehicle with throw-away stages. See [12, pp. 175-189] for more information on the K-1. The numerical values used are described below.

- **Planet**

The vehicle is launched from the surface of the Earth. Table 7.1 shows the values used to define the planet.

Planet Radius (km) [22]	6371
Gravitational Parameter ($\frac{m^3}{s^2}$) [2]	3.986012×10^{14}
Sidereal Rotation Rate ($\frac{rad}{s}$)	7.2722×10^{-5}

Table 7.1: Planet Definition

Note that the sidereal rotation rate is estimated by dividing 2π by the number of seconds in 24 hours.

- **Mass**

The mass is defined in terms of the total mass of each stage, and the mass fraction of each stage that is propellant. Table 7.2 shows the mass values used for the numerical results.

First Stage	
Initial Total Mass (kg)	248950
Propellant Mass Fraction	0.8324
Second Stage	
Initial Total Mass (kg)	134040
Propellant Mass Fraction	0.88217

Table 7.2: Mass Definition

- **Aerodynamics**

Two models are used for the vehicle aerodynamic coefficients. One model uses tabular aerodynamic data for axial and normal force coefficients (C_A and C_N). The data is taken from a generic axisymmetric vehicle. The data is given as a function of Mach number and α_{total} . This data is converted into tabular data for the body force coefficients as a function of Mach number, α , and β . Linear interpolation is used to find the data for the desired conditions. The partial derivatives of the data are estimated numerically. See section 6.6.12 for a full discussion of this technique. The other model uses the simplified body coefficients from section 6.6.12. Both stages are assumed to have the same aerodynamic characteristics. Table 7.3 shows some of the aerodynamic parameters used for the numerical results.

Simplified Coefficient Definition	
C_{B_x}	-0.6
$\left(\frac{dC_{B_y}}{d\beta} \right)$	-4.0
$\left(\frac{dC_{B_z}}{da} \right)$	-4.0
Other Data	
Reference Area (m^2)	61

Table 7.3: Aerodynamic Definition

- **Rocket Engine Characteristics**

Propulsion is provided by a single large engine. The thrust direction is controlled by changing the attitude of the vehicle, but the control system required to achieve the desired vehicle attitude is not modelled. The maximum thrust level is defined by the initial thrust to weight ratio. The specific impulse of the engine is assumed to be constant. The engine is assumed to have full throttle capability. Both stages are assumed to have the same engine characteristics.

Table 7.4 shows the rocket engine characteristics.

Initial Thrust to Weight Ratio	1.2
Specific Impulse (s)	300
Throttle Capability	0% to 100%

Table 7.4: Rocket Engine Definition

- **Trajectory Constraints**

Constraints on the magnitude of the sensed acceleration and the dynamic pressure are applied everywhere on the trajectory (i.e. during both stages). Table 7.5 shows the numerical values for the trajectory constraints.

Sensed Acceleration Limit (g's)	4
Dynamic Pressure Limit (kPa)	15

Table 7.5: Trajectory Constraint Definition

- **Initial State**

Table 7.6 shows the initial state of the vehicle. Appendix D describes how these parameters are transformed into an initial inertial state.

Initial Inertial Latitude	28°
Initial Inertial Longitude	0°
Initial Altitude (m)	10
Initial Vertical Velocity ($\frac{m}{s}$)	10
Initial Elevation of Launch Vehicle	90°
Initial Azimuth of Launch Vehicle	90°

Table 7.6: Initial Vehicle State

Note that the vehicle is initialized with a small altitude and vertical velocity (i.e. the vehicle is initialized going straight up). This is necessary because there is a singularity in the model when the velocity is zero. Also, even though the initial azimuth is 90° (i.e. due east), this only points the vehicle body Z-axis in that direction. It does not force the vehicle to fly east.

7.2 Numerical Results

Target Orbit

The target orbit for the following plots is described in Table 7.7.

Final Altitude (km)	400
Final Eccentricity	0
Final Inclination	28°

Table 7.7: Final Target Orbit

Solution Steps for Finding the Optimal Trajectory

The D^2 -method (see Chapter 6) was used to find the optimal trajectory. The steps used to find this trajectory are outlined below:

1. A crude initial trajectory guess was found using the method described in Appendix E. The simplified body coefficients described in section 6.6.12 were used instead of the tabular aerodynamic data. The dynamic pressure and sensed acceleration constraints were removed. A good solution was found using 10 LGL points in each stage. The optimizer was only allowed to go through 100 iterations before termination.
2. A new solution was found using 20 LGL points in each stage. The solution from the previous step was used as a starting point. Cubic splines were used to interpolate the solution with 10 points per stage to an initial guess with 20 points per stage. Again, the optimizer was only allowed 100 iterations.
3. It was noted that the time span of the second stage was much larger than that of the first stage. Another solution was found with 20 LGL points in the first stage and 30 LGL points in the second stage.
4. The 20/30 LGL point solution was rerun with the dynamic pressure and sensed acceleration constraints added. The optimizer initially exited after several iterations without finding a feasible solution. However, it was found that repeatedly feeding the non-feasible solution back into the optimizer as an initial guess resulted in a feasible solution. Once a feasible solution was found, the optimizer was allowed to run for 100 iterations. A good solution was found.
5. The 20/30 LGL point solution with the trajectory constraints was rerun with the tabular aerodynamic coefficients used in place of the simplified body force coefficients. The optimizer was allowed to run for 100 iterations. After this, the solution was fed back into the optimizer for another 100 iterations in order to improve the solution.

Note that limiting the number of NPSOL iterations is a trade-off between getting a “nearer optimal” solution in more time or getting a “less than optimal” solution in less time. Many times, the final mass of a 200 iteration solution is not much better than that for a 100 iteration solution.

Figures 7-1 through 7-12 show various parameters of interest for the optimal trajectory. In Figure 7-2, it can be seen that the vehicle launches due east, taking advantage of the rotation of the Earth. The launch vehicle was not constrained to launch due east; the optimizer found this as part of the optimal solution.

Note from Figure 7-4 (Euler Angles), that the ϕ Euler angle is unconstrained. This is because the ϕ Euler angle is rotation about the body X-axis. Since the vehicle is axisymmetric, rotation about the body X-axis has no effect on the trajectory. In practice, it would be possible to specialize the model such that the roll angle of an axisymmetric vehicle is not part of the optimization vector. However, the model of Chapter 6 was generalized to apply to both axisymmetric and non-axisymmetric vehicles.

Figures 7-5 and 7-6 show the aerodynamic angles and coefficients, respectively. These plots only show the first 200 seconds of the flight. After 200 seconds, the vehicle is outside of the sensible atmosphere, and the aerodynamics are negligible. Note that between 25 and 60 seconds, the total angle of attack of the vehicle is held close to zero.

Figure 7-8 is of particular interest. It shows the time history of mass. The staging point can be clearly seen as a discontinuity in mass. This is another demonstration of the ability of the Legendre Pseudospectral Method to handle discontinuities in the states.

Figures 7-9 (Mass Flow Rate) and 7-10 (Thrust) are interesting when compared to Figures 7-11 (Dynamic Pressure) and 7-12 (Sensed Acceleration). Note that the thrust is throttled down when the vehicle approaches the maximum dynamic pressure limit. It is throttled down again when the vehicle approaches the maximum sensed acceleration limit. Also, note that the mass flow rate and thrust are discontinuous across the staging point.

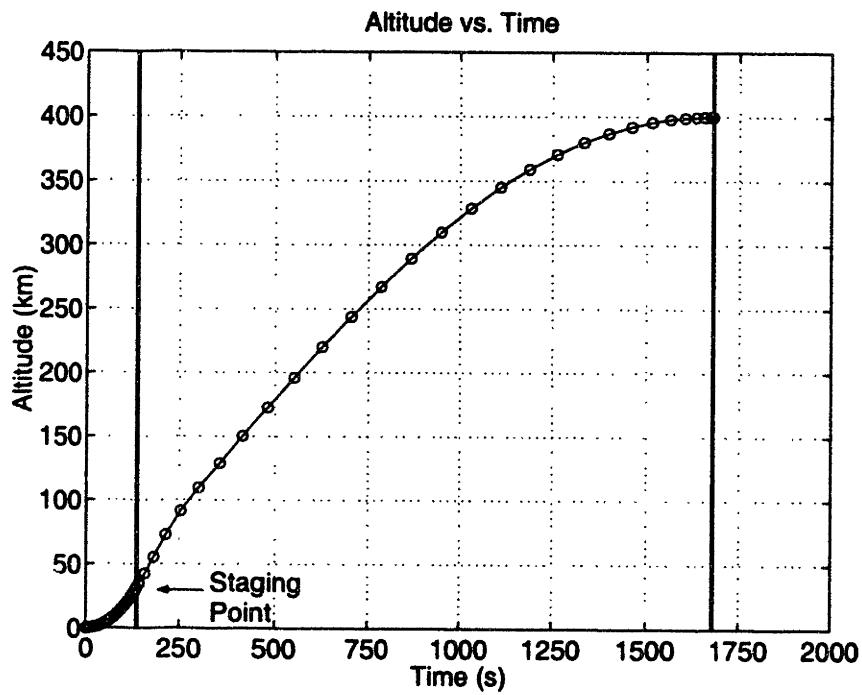


Figure 7-1: Three-Dimensional Launch: Time History of Altitude

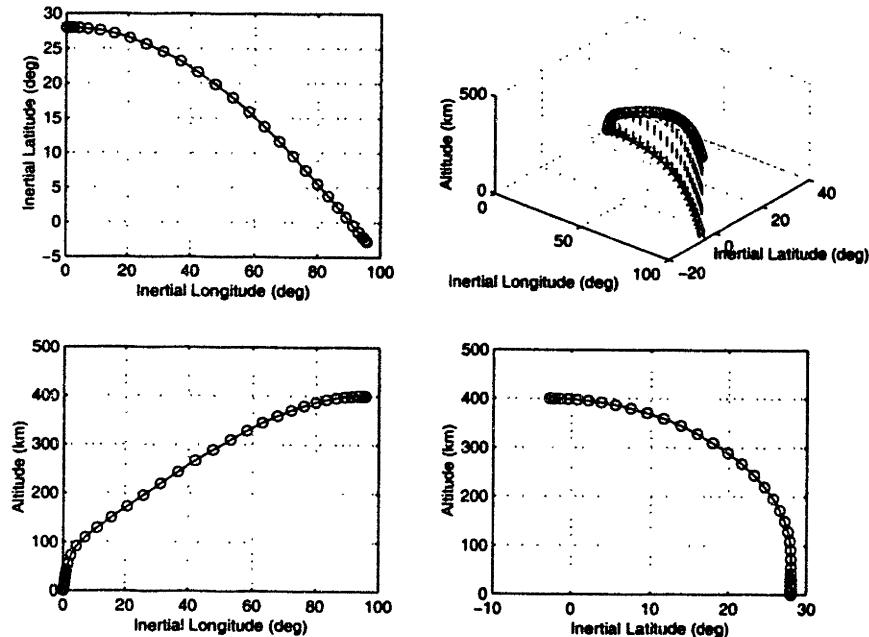


Figure 7-2: Three-Dimensional Launch: Time History of Trajectory

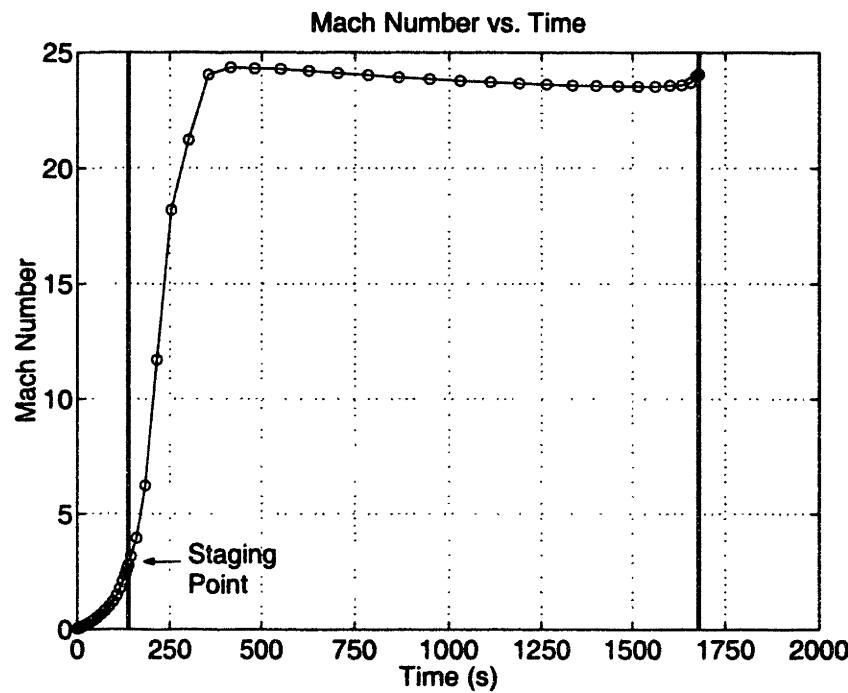


Figure 7-3: Three-Dimensional Launch: Time History of Mach Number

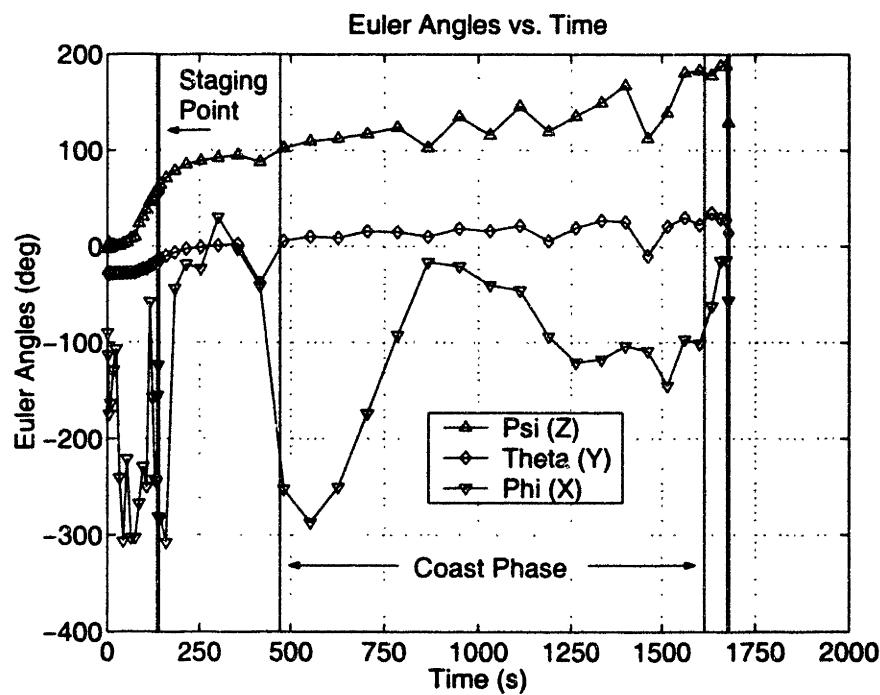


Figure 7-4: Three-Dimensional Launch: Time History of Euler Angles

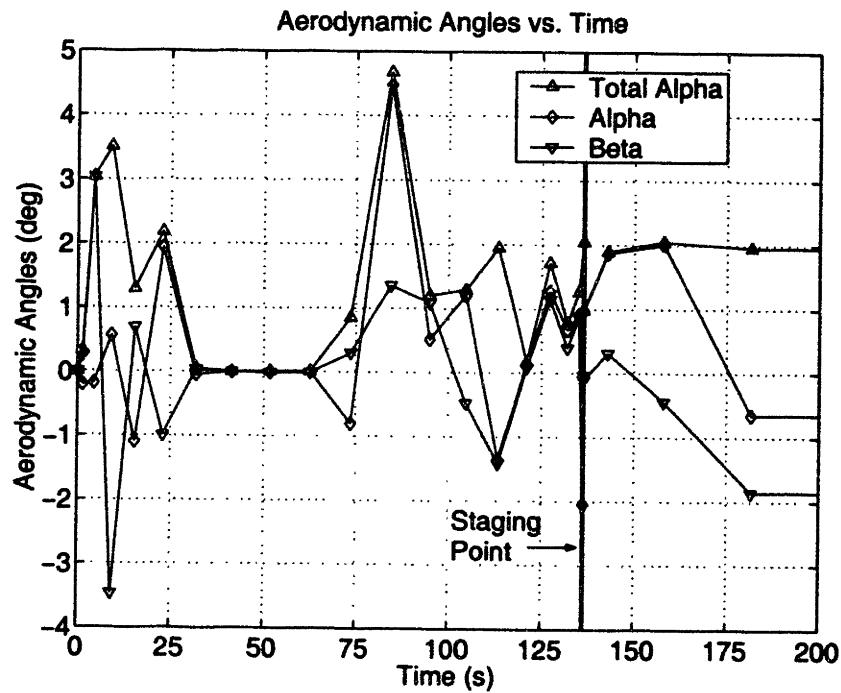


Figure 7-5: Three-Dimensional Launch: Time History of Aerodynamic Angles

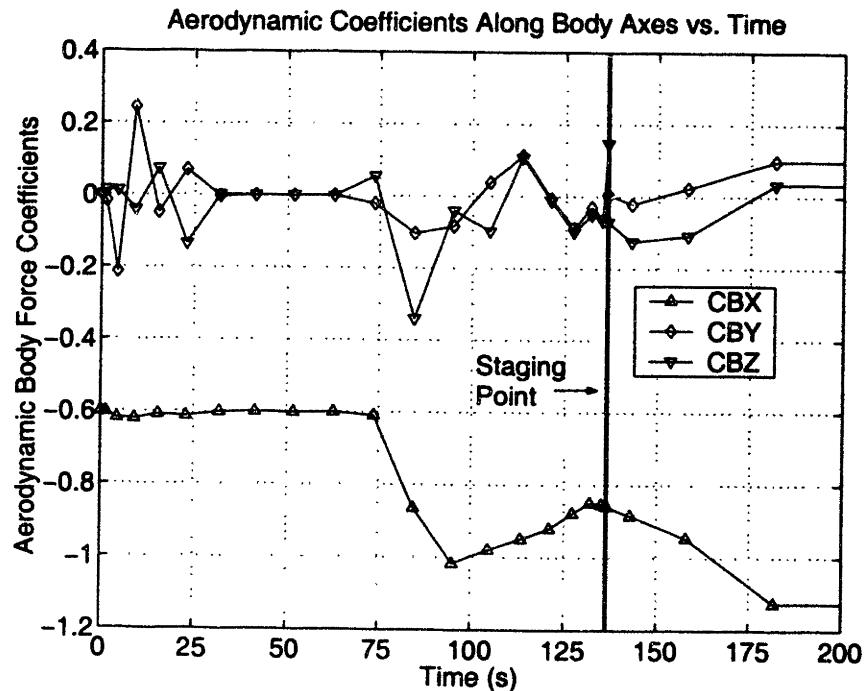


Figure 7-6: Three-Dimensional Launch: Time History of Body Coefficients

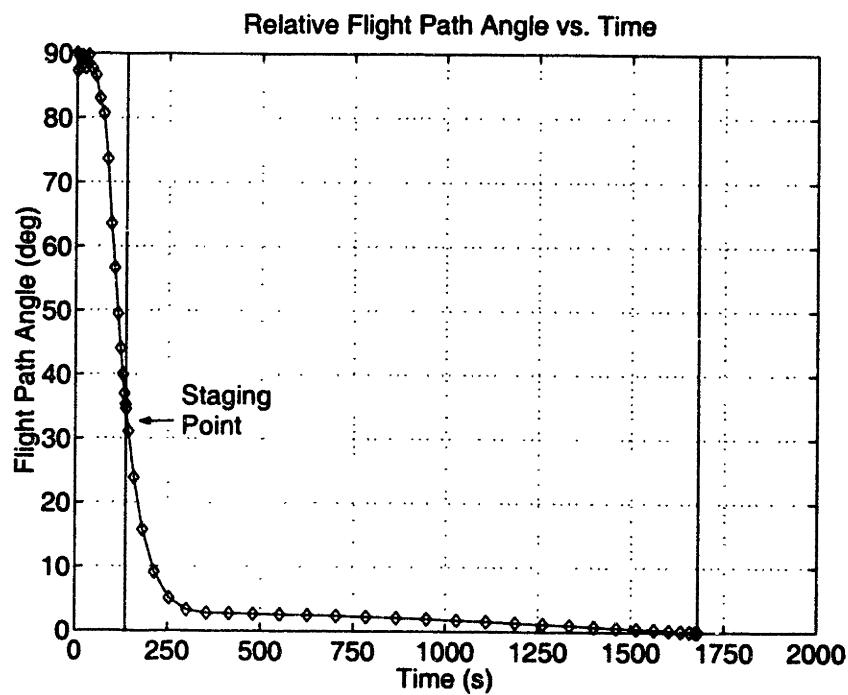


Figure 7-7: Three-Dimensional Launch: Time History of Relative Flight Path Angle

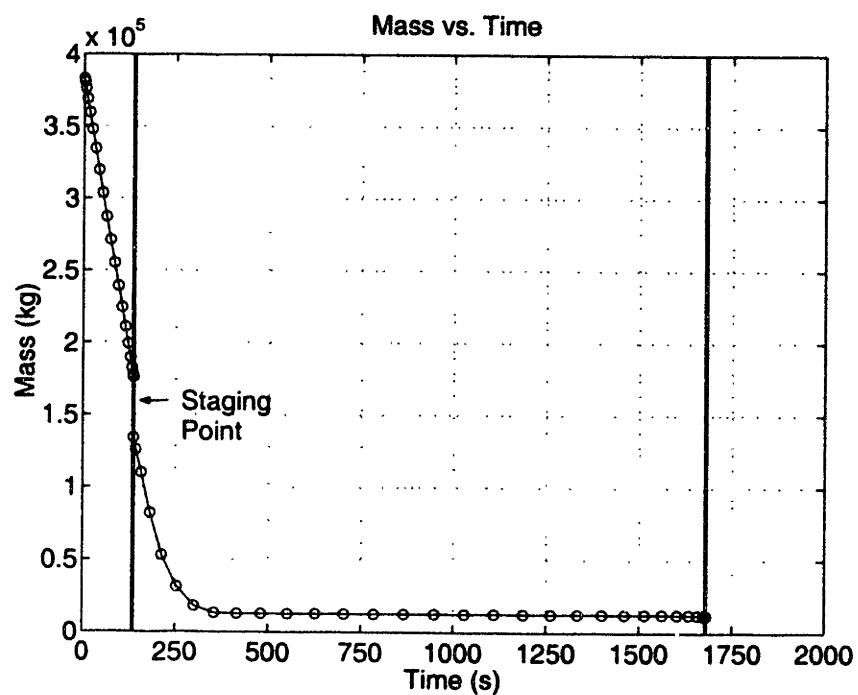


Figure 7-8: Three-Dimensional Launch: Time History of Mass

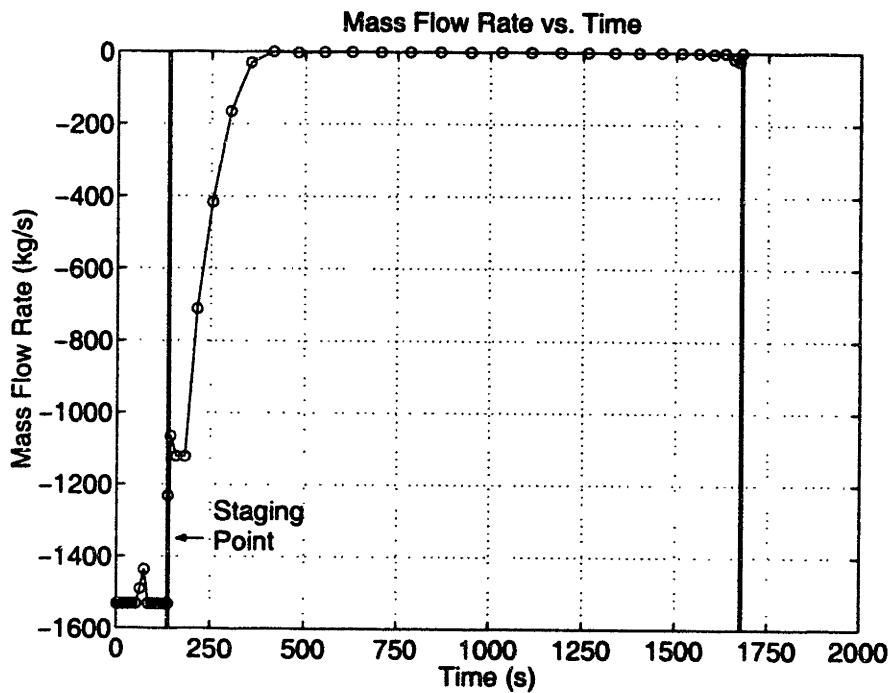


Figure 7-9: Three-Dimensional Launch: Time History of Mass Flow Rate

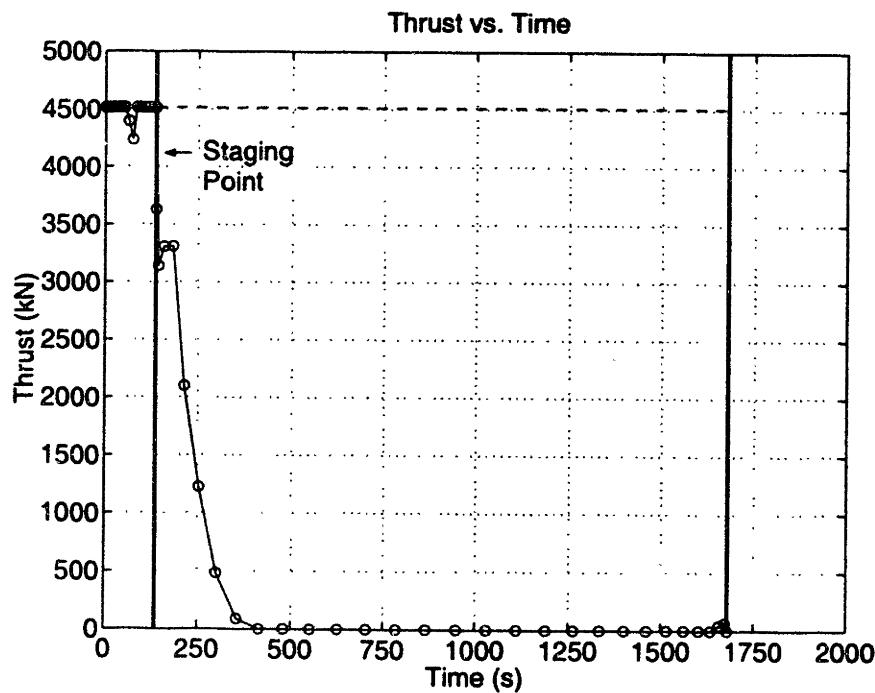


Figure 7-10: Three-Dimensional Launch: Time History of Thrust

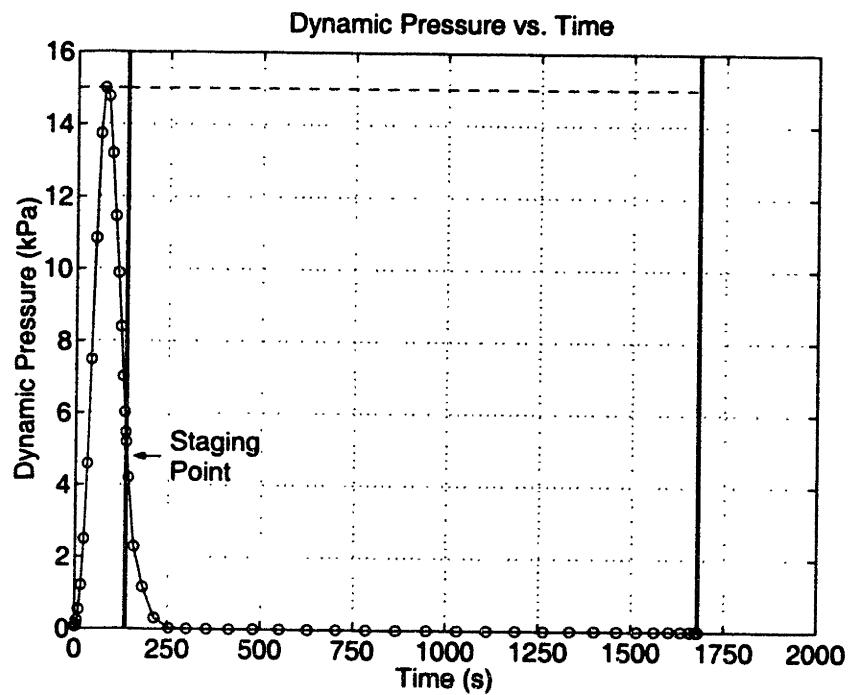


Figure 7-11: Three-Dimensional Launch: Time History of Dynamic Pressure

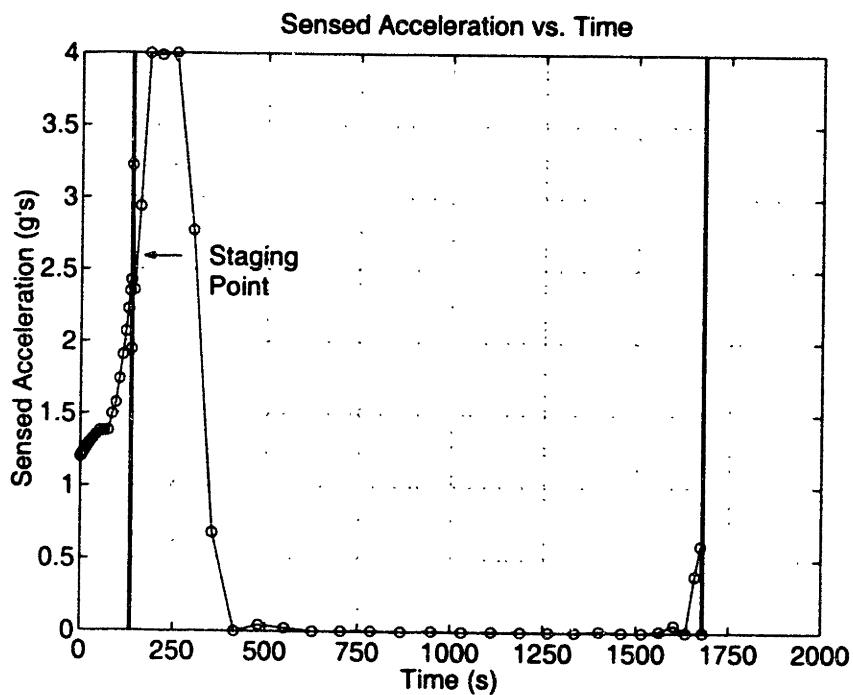


Figure 7-12: Three-Dimensional Launch: Time History of Sensed Acceleration

Chapter 8

Guidance Concepts

A possible use of the Legendre Pseudospectral Method of this thesis is for real-time, adaptive launch vehicle guidance. In this chapter, the guidance solution obtained by the Legendre Pseudospectral Method is used in a simulated launch vehicle. The actual performance is then compared with the predicted performance in two situations: i) when the models for the simulation and the optimizer are the same and, ii) when the optimization model is simpler than the simulation model. This latter situation is meant to reflect “real life” uncertainties in vehicle performance. The final mass fraction is compared to the predicted mass fraction and convergence times are noted.

8.1 Extracting Controls from the LGL Points

Using the Legendre Pseudospectral Method gives a state estimate and controls only at the LGL points. In order to increase the speed of convergence of the numerical optimizer, it is necessary to use the least number of points that accuracy requirements will allow. This results in a solution of lower fidelity. However, the time step of the integrator in the simulation is much smaller than the time step between the LGL points. It is necessary to estimate the controls at the simulation time points using the control information contained in the LGL points.

Interpolation is necessary to find the controls between the LGL points. Cubic spline interpolation is used in this thesis. Interpolation of the thrust magnitude with

this method works very well. However, a problem arises with the interpolation of the quaternions. Recall from section 6.6.2, equation 6.59, that the quaternion elements must meet the following normalization constraint:

$$q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1 \quad (8.1)$$

When interpolating the quaternion elements, the interpolated values very rarely meet this constraint. One solution is to renormalize the interpolated quaternion elements. However, this results in unsatisfactory performance. The renormalized quaternions do not point the vehicle in quite the right direction.

Another solution is to use the relationship between the quaternion elements and the Euler angles. The quaternions at the LGL points (which do meet the constraint of equation 8.1) are converted into inertial Euler angles. The particular Euler angles used in this thesis are described in section 6.6.2. The inertial Euler angles are then interpolated using cubic splines. The Euler angles are much more amenable to interpolation and do not have to meet any normalization constraint. The interpolated Euler angles are then converted back into quaternion elements. Due to the relationship between Euler angles and the quaternion elements, the quaternion elements computed from the interpolated Euler angles automatically meet the constraint of equation 8.1.

8.2 Open-Loop Guidance Demonstration

One interesting question to ask is:

If the controls are extracted from the LGL points and integrated forward in the simulation, will the estimated state at the LGL points match the “true” state from the simulation?

With the model of Chapter 6, the states are position (R_x , R_y , R_z), velocity (V_x , V_y , V_z), and mass (m). The controls are thrust magnitude (T) and inertial vehicle

attitude, defined by the quaternion elements (q_1, q_2, q_3, q_4). The states and controls are related by the dynamic equations (equations 6.2 and 6.6), repeated below.

$$\dot{\vec{R}} = \vec{V}$$

$$\dot{\vec{V}} = \frac{\vec{T} + \vec{A}}{m} + \vec{g} \quad (8.2)$$

$$\dot{m} = -\frac{T}{V_{exit}}$$

A simulation can be easily set-up by numerically integrating the dynamic equations. All that is needed is an initial state and the controls. The initial state is then integrated forward in time.

In order to answer the stated question, a simulation is set-up using the MATLAB command “ode45” as the integrator for the dynamic equations (equations 8.2). A single-stage vehicle similar to the one described in Chapter 7 is used. The initial total mass of this vehicle is the same as the vehicle of Chapter 7. Table 8.1 shows the numerical values used for the simulation. Unless otherwise noted, all other values (i.e. initial state, engine characteristics, constraints, etc.) are the same as for the vehicle of Chapter 7.

Initial Total Mass (kg)	382,990
Initial Propellant Mass Fraction	0.99
Circular Target Orbit Altitude (km)	400
Target Orbit Inclination (degrees)	28

Table 8.1: Numerical Values Used for Guidance Simulation

The controls are extracted from the LGL points as discussed in the previous section. The controls are then flown “open-loop,” with no in-flight corrections. Case I shows a solution where both the optimizer and the simulation use the same vehicle model. Case II shows a solution where the optimizer uses the simplified aerodynamic coefficients while the simulator uses the tabular aerodynamic coefficients (see section 6.6.12). The plots for Case I and Case II show both the actual solution from the

simulation and the predicted solution from the optimizer. The solid lines are the values from the simulation. The points with no lines are the values at the LGL points from the optimizer.

Case I - Open-Loop: Simple Aerodynamic Model for Both Optimizer and Simulator

An optimal solution was found for the defined single-stage launch problem using 20 LGL points. The simplified aerodynamic coefficients of section 6.6.12 were used in the optimizer model to find the optimal solution. The controls are extracted from the optimal solution (as discussed in the previous section) and used as open-loop guidance commands. The simulation is then used to integrate the equations of motion forward to the final time of the optimal solution. The same simplified aerodynamic coefficients are used in the simulation model. The simulated trajectory closely matches the predicted trajectory.

Figures 8-1 through 8-4 show the results (all figures are at the end of the chapter in section 8.4). Figure 8-1, a plot of the time history of altitude, gives an idea of how well the open-loop trajectory matches the predicted trajectory.

Figure 8-2 shows the controls that were used. The thrust and two Euler angles, ψ and θ , are plotted. The Euler angle corresponding to roll about the vehicle X-axis, ϕ , is not plotted. Since the modelled vehicle is axisymmetric, the roll angle does not affect the trajectory in any way. Note that the numerical optimizer found that there is a zero-thrust coast phase of the trajectory. During the coast phase, the Euler angles appear to oscillate. This oscillation is not important for two reasons. First, there is no thrust, so the attitude of the vehicle is not controlling the thrust direction. Second, the vehicle is outside of the sensible atmosphere during the coast phase. Therefore, there are no aerodynamic forces to minimize by changing the vehicle attitude. So, the attitude of the vehicle does not affect the trajectory during the coast phase. In practice, these oscillating attitude commands would be ignored; the guidance system would command the vehicle to either hold an inertial attitude or slowly rotate to the new ignition attitude.

Figures 8-3 and 8-4 show the dynamic pressure and sensed acceleration time histories, respectively. It is interesting that the simulated trajectory passes through the LGL points on the maximum dynamic pressure limit, but that it violates the constraint between those points.

Table 8.2 shows a comparison between the predicted and actual final mass and orbit. Note that the actual final mass fraction is better than the predicted mass fraction, even after an impulsive trim burn is modelled to meet the target velocity. Of course, the simulated trajectory violates the dynamic pressure constraint. Had this constraint been imposed in the simulation, the simulated final mass fraction would probably not exceed the predicted final mass fraction.

Final Orbit	Predicted (LGL Solution)	Actual (Simulation)
Final Altitude (km)	400	390.5578
Final Velocity (km/s)	7.6726	7.6475
Final Eccentricity	0	0.0086711
Final Inclination (degrees)	28	28.0020
Final Mass	Predicted (LGL Solution)	Actual (Simulation)
Final Mass Fraction	0.042077	0.042482
Mass Fraction After Impulsive Trim Burn to Match Target Velocity = 0.042121		

Table 8.2: Case I: Final Mass and Orbit Comparison

Case II - Open-Loop: Simple Aerodynamics in Optimizer, Complex Aerodynamics in Simulator

The same initial optimal solution used in Case I is used for Case II. However, for Case II, the simulation uses the tabular aerodynamic coefficients. The simulated trajectory does not match the predicted trajectory as well as for Case I. This makes sense because the aerodynamic coefficients for the two models are different.

Figures 8-5 through 8-8 show the results. Figure 8-5 shows the time history of altitude. The dramatic difference between the simulated and predicted altitude profiles shows how much the aerodynamic models affect the trajectory. Figure 8-2 shows the controls that were used. These controls are exactly the same as those for Case I.

Remember from the discussion of the Case I solution that the Euler angle oscillations during the coast phase are not important and can be ignored. Figures 8-3 and 8-4 show the dynamic pressure and sensed acceleration time histories, respectively. Table 8.3 shows a comparison between the predicted and actual final mass and orbit. Note that the actual final mass fraction is better than the predicted mass fraction, but the final simulated altitude is far below the desired altitude. However, the vehicle does come close to the desired target velocity.

Final Orbit	Predicted (LGL Solution)	Actual (Simulation)
Final Altitude (km)	400	245.2063
Final Velocity (km/s)	7.6726	7.6619
Final Eccentricity	0	0.047711
Final Inclination (degrees)	28	28.0021
Final Mass	Predicted (LGL Solution)	Actual (Simulation)
Final Mass Fraction	0.042077	0.042473
Mass Fraction After Impulsive Trim Burn to Match Target Velocity = 0.042319		

Table 8.3: Case II: Final Mass and Orbit Comparison

8.3 Closed-Loop Guidance Demonstration

For the closed-loop guidance demonstration, it is assumed that the numerical optimizer can instantaneously provide a new set of guidance commands. However, it is also assumed that the guidance commands can only be provided at 50 second intervals. The demonstration includes an initial pre-flight convergence of the optimal solution. The simulation takes the controls from this initial solution and uses them for 50 seconds. After 50 seconds, the numerical optimizer is called again to reoptimize the trajectory from the current state. The controls from this new trajectory are used for another 50 seconds. This process repeats until the time-to-go of the flight is less than 50 seconds. The rest of the flight is then flown open-loop. Cases III and IV are the closed-loop versions of Cases I and II, respectively. The plots for Case III and Case IV show only the solution from the simulation (the LGL points from the optimizer are not shown).

Case III – Closed-Loop: Simple Aerodynamic Model for Both Optimizer and Simulator

As expected, closed-loop guidance comes much closer to meeting the desired orbit. The results are good in terms of the desired target orbit being reached. Figures 8-9 through 8-12 show the altitude, guidance commands, dynamic pressure, and sensed acceleration, respectively.

Only the first 50 seconds of the closed-loop trajectory use the same controls as before (see Figure 8-10). Thereafter, the controls reflect the closed-loop response to being off the nominal trajectory. As before, there is a zero-thrust coast phase during which the Euler angles oscillate. The oscillations are not as pronounced as before because of the closed-loop nature of the solution. There are also some small burns during the coast phase (these burns can be seen better in the plot of sensed acceleration, Figure 8-12). The guidance algorithm was implemented with a maximum limit of 10 iterations. However, this iteration limit is not high enough for the optimizer to converge to the “true” optimal solution (i.e. no small burns, only a pure coast). In practice, the iteration limit would allow the optimizer to converge to an optimal solution with a pure coast. Note that during the small burns, the attitude of the vehicle becomes relevant, and the Euler angle oscillation does not occur. See the discussion of the Case I solution for why the Euler angle oscillations during the zero-thrust coast phase can be ignored.

Figures 8-11 and 8-12 show the dynamic pressure and sensed acceleration time histories, respectively. Although the trajectory is reoptimized every 50 seconds, the controls are flown open-loop between these reoptimizations. Thus, the constraints are still violated for part of the trajectory, but not to the extent that they are in Case I.

Table 8.4 shows a comparison between the predicted and actual final mass and orbit. The final orbit is very close to the desired orbit. Again, the final mass fraction is higher than predicted. This probably occurs for two reasons. First, the predicted solution only uses 20 LGL points. Thus, it is not perfectly accurate. However, the trajectory is repeatedly reoptimized in-flight, allowing the solution to get closer to

the true optimal solution. Second, the constraints are violated; this allows a more fuel-efficient trajectory to orbit.

For Case III, the average reoptimization time was 74 seconds (CPU time). However, the optimizer was limited to only 10 iterations. If the optimizer were allowed more iterations, the solution would most likely be better, but each reoptimization would take longer. While 74 seconds is not close to real-time, this result holds promise for a near real-time numerical optimization capability.

One approach to real-time implementation is to have enough computer throughput to run guidance at such a rate that it stays converged in one iteration. Of course, how well the guidance stays converged depends on the accuracy of the model used in the optimization. A more accurate model may stay converged longer but would take longer to evaluate. A trade must be made to minimize the throughput requirements.

Final Orbit	Predicted (LGL Solution)	Actual (Simulation)
Final Altitude (km)	400	400.0006
Final Velocity (km/s)	7.6726	7.6725
Final Eccentricity	0	1.5356e-05
Final Inclination (degrees)	28	28.0000
Final Mass	Predicted (LGL Solution)	Actual (Simulation)
Final Mass Fraction	0.042077	0.042176
Mass Fraction After Impulsive Trim Burn to Match Target Velocity = 0.042175		
Final Time	Predicted (LGL Solution)	Actual (Simulation)
Final Time (s)	956	1496

Table 8.4: Case III: Final Mass and Orbit Comparison

Case IV – Closed-Loop: Simple Aerodynamics in Optimizer, Complex Aerodynamics in Simulator

Case IV is similar to Case II, except that the closed-loop guidance is used. The results are good in terms of the desired target orbit being reached. Figures 8-13 through 8-16 show the results.

Figure 8-13 shows the time history of altitude. Figure 8-14 shows the controls that were used. As in the other cases, there is a zero-thrust coast phase. However, as

for Case III, it can be seen that there are small burns during the coast phase (these burns can be seen better in the plot of sensed acceleration, Figure 8-16). Since the optimizer is commanding a series of burns, the vehicle attitude is now relevant, thus the Euler angle oscillations do not occur (see the discussions of the Case I and Case III solutions). Figures 8-15 and 8-16 show the dynamic pressure and sensed acceleration time histories, respectively. Note that the same constraint violations occur as in the other cases. Again, the closed-loop nature of the algorithm reduces the violations.

Table 8.5 shows a comparison between the predicted and actual final mass and orbit. As for Case III, the final orbit is very close to the desired orbit. This is a very good result for Case IV considering that the optimizer and the simulator used different aerodynamic models. Note that for this case, the actual final mass fraction is less than the predicted mass fraction. This is due to the higher drag losses that were not modelled in the predicted solution.

For Case IV, the average reoptimization time was 93 seconds (CPU time). This average time is higher than for Case III due probably to the different aerodynamic models between the optimizer and the simulator.

Final Orbit	Predicted (LGL Solution)	Actual (Simulation)
Final Altitude (km)	400	399.9926
Final Velocity (km/s)	7.6726	7.6122
Final Eccentricity	0	0.015698
Final Inclination (degrees)	28	28.0048
Final Mass	Predicted (LGL Solution)	Actual (Simulation)
Final Mass Fraction	0.042077	0.041062
Mass Fraction After Impulsive Trim Burn to Match Target Velocity = 0.040228		
Final Time	Predicted (LGL Solution)	Actual (Simulation)
Final Time (s)	956	1720

Table 8.5: Case IV: Final Mass and Orbit Comparison

8.4 Figures

The graphical results for this chapter are shown on the following pages.

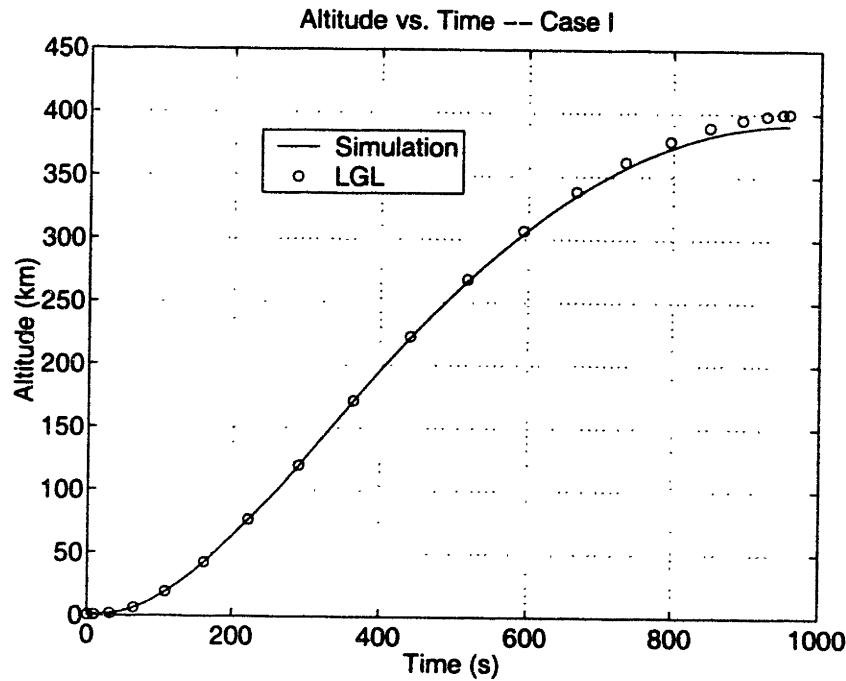


Figure 8-1: Case I: Altitude Comparison

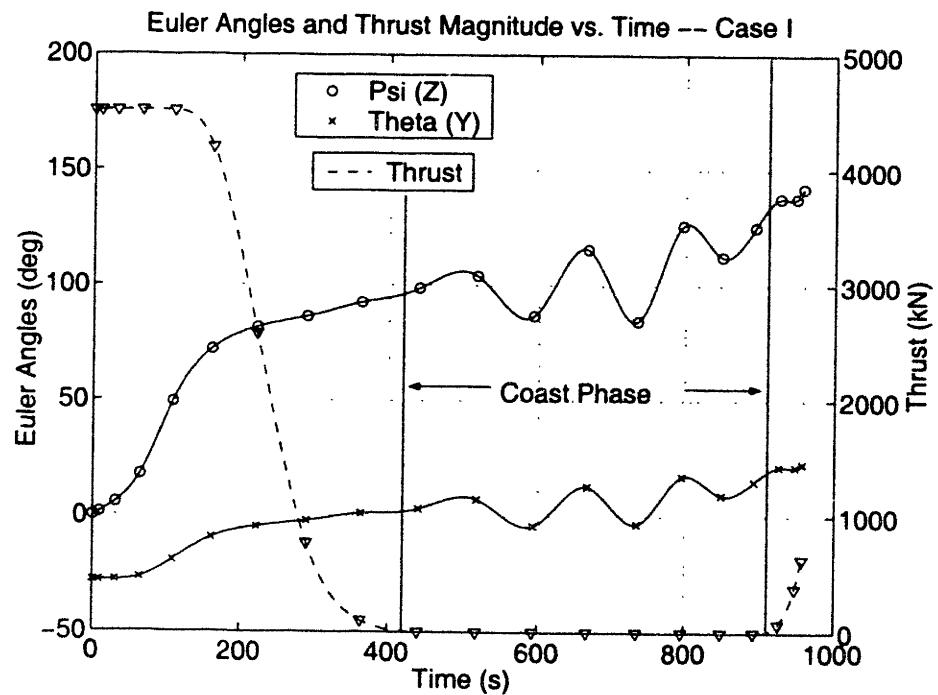


Figure 8-2: Case I: Controls

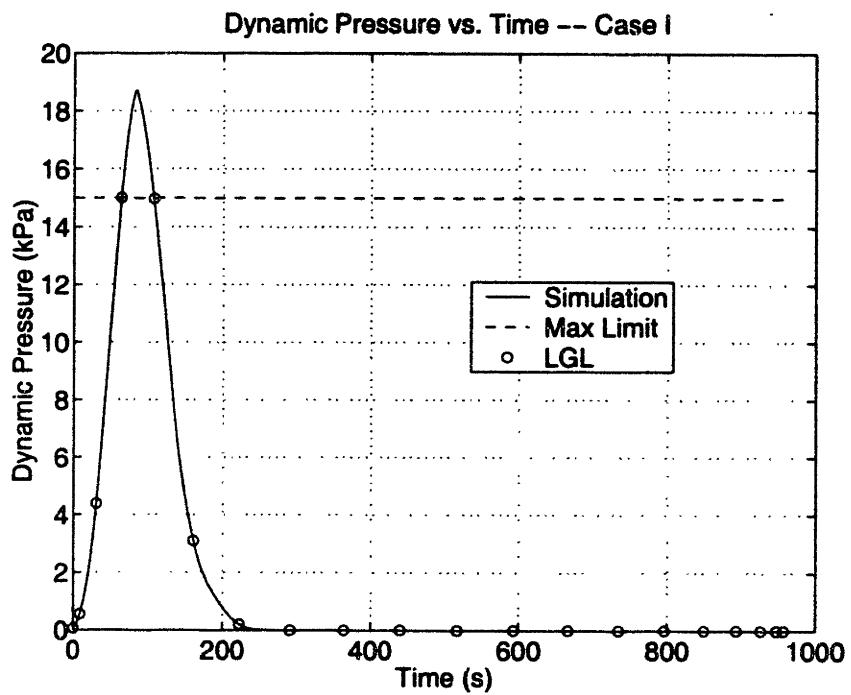


Figure 8-3: Case I: Dynamic Pressure Comparison

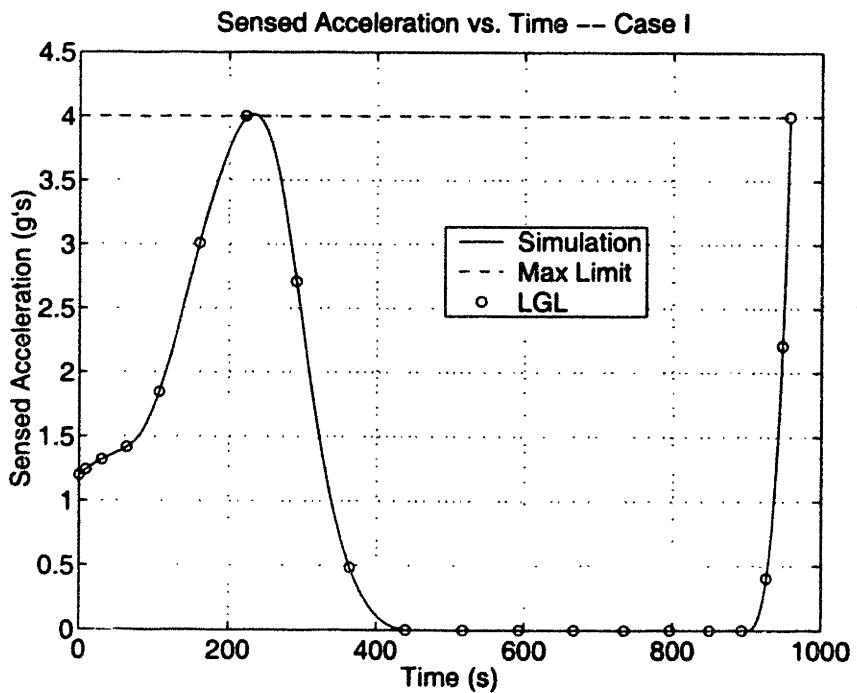


Figure 8-4: Case I: Sensed Acceleration Comparison

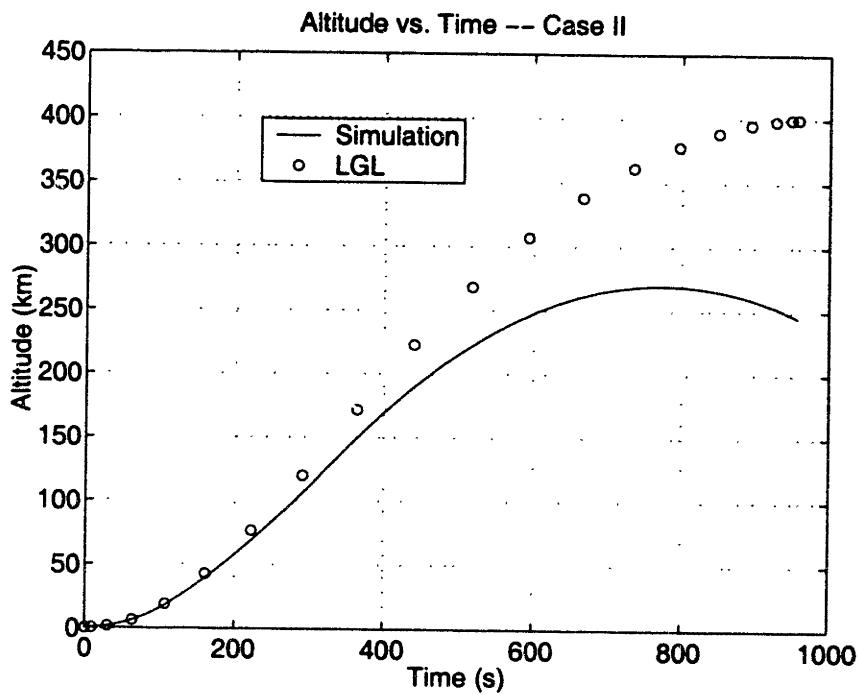


Figure 8-5: Case II: Altitude Comparison

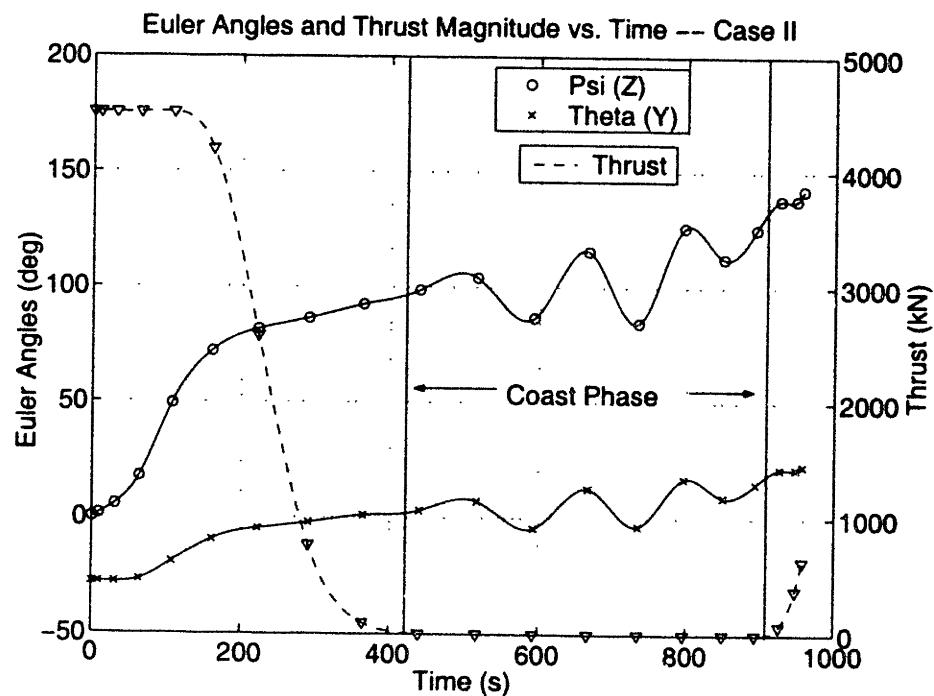


Figure 8-6: Case II: Controls

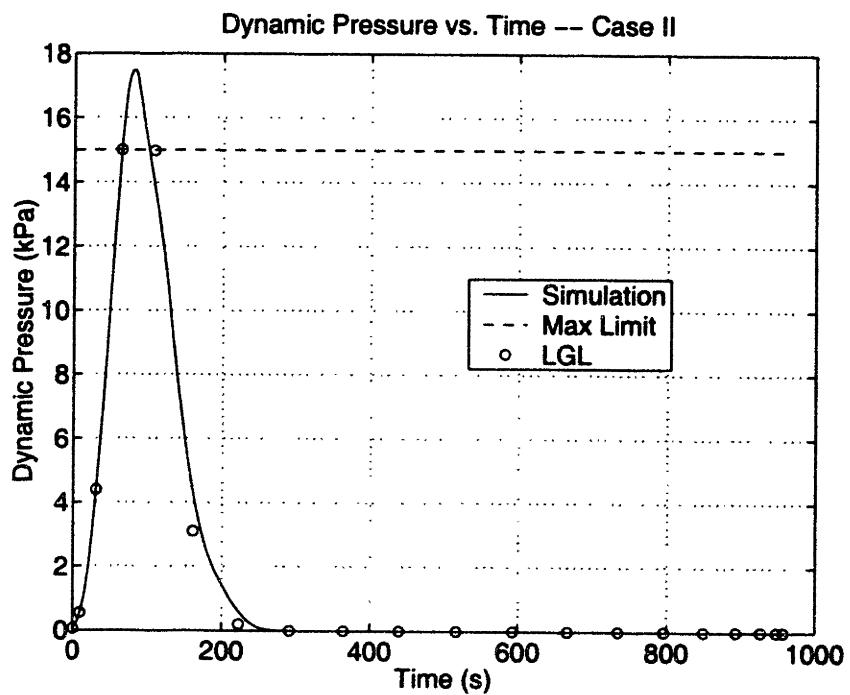


Figure 8-7: Case II: Dynamic Pressure Comparison

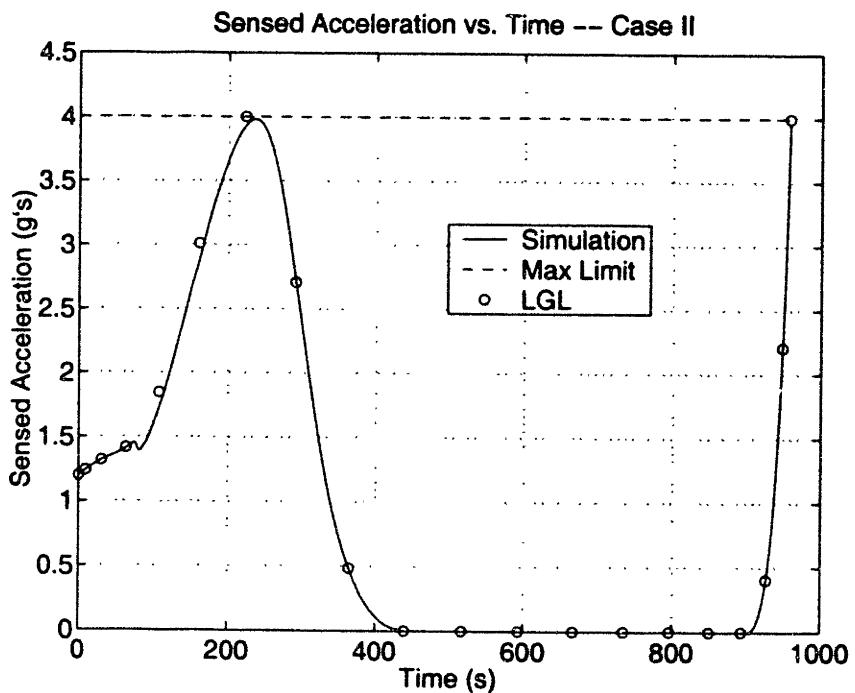


Figure 8-8: Case II: Sensed Acceleration Comparison

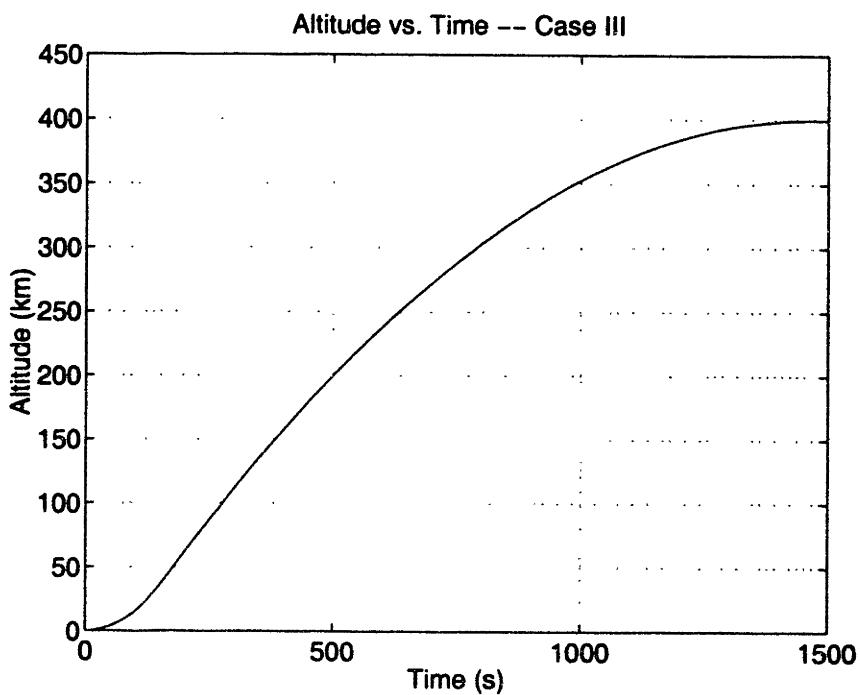


Figure 8-9: Case III: Altitude Comparison

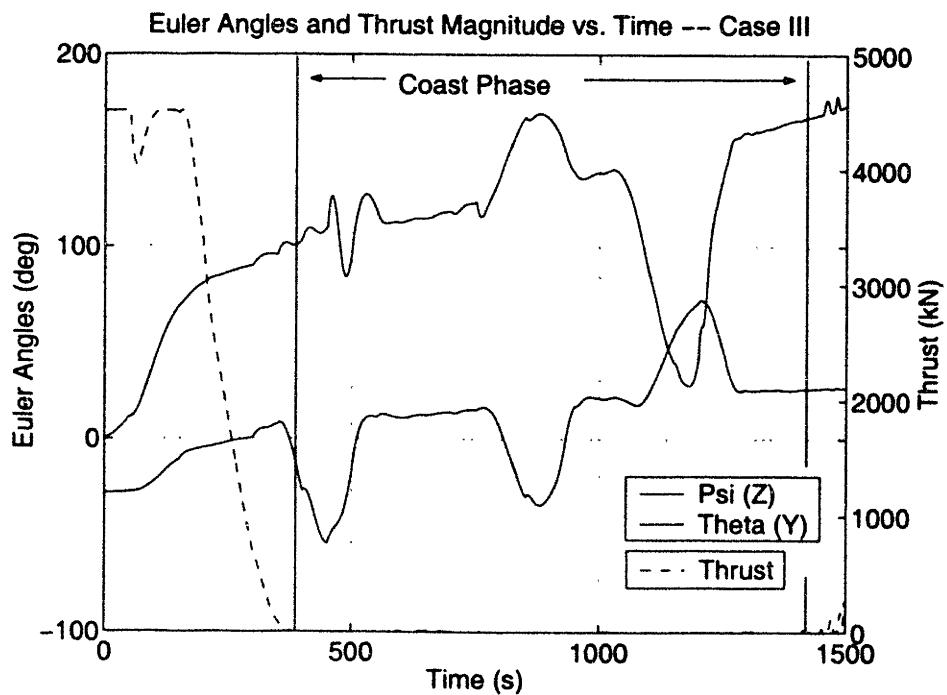


Figure 8-10: Case III: Controls

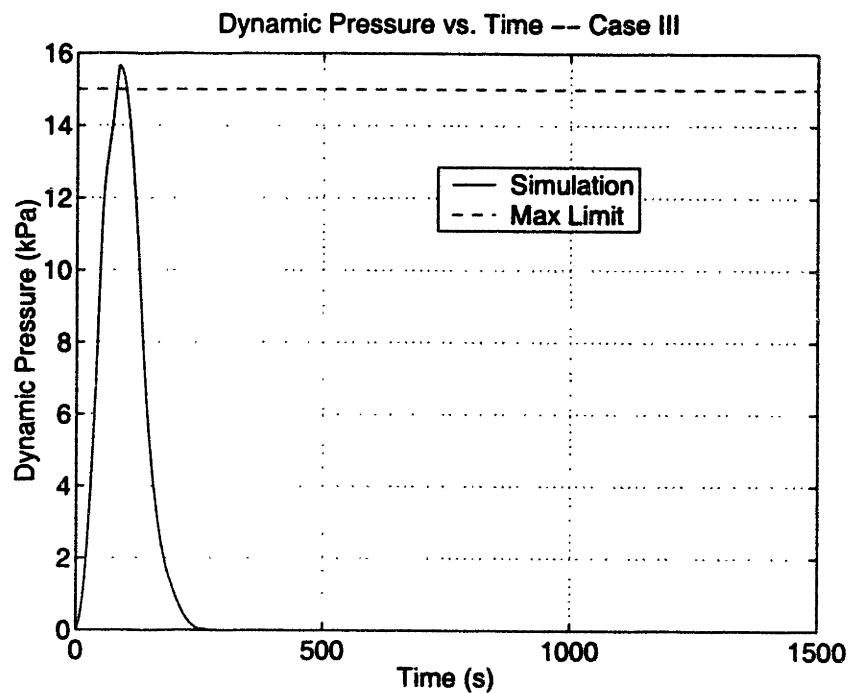


Figure 8-11: Case III: Dynamic Pressure Comparison

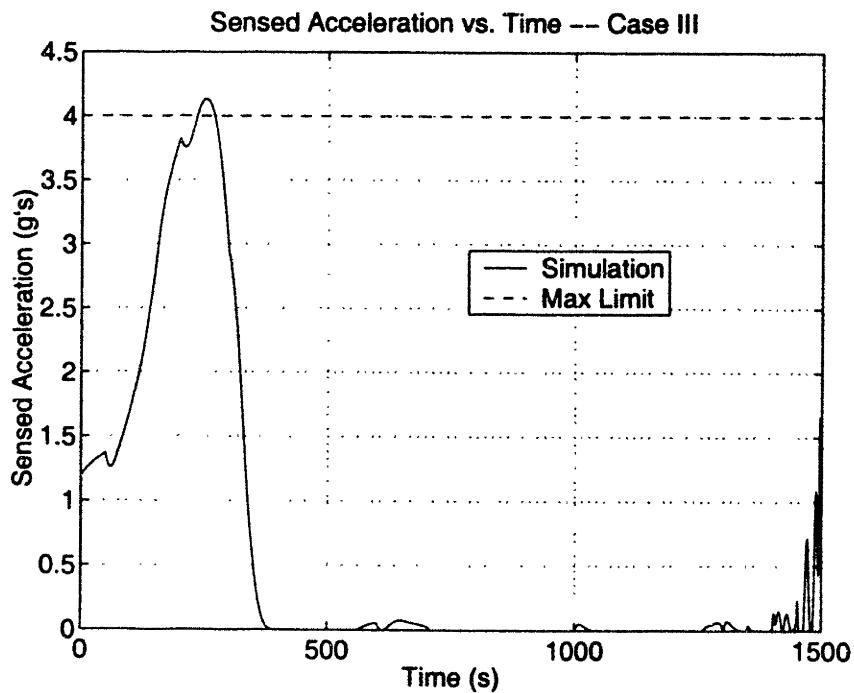


Figure 8-12: Case III: Sensed Acceleration Comparison

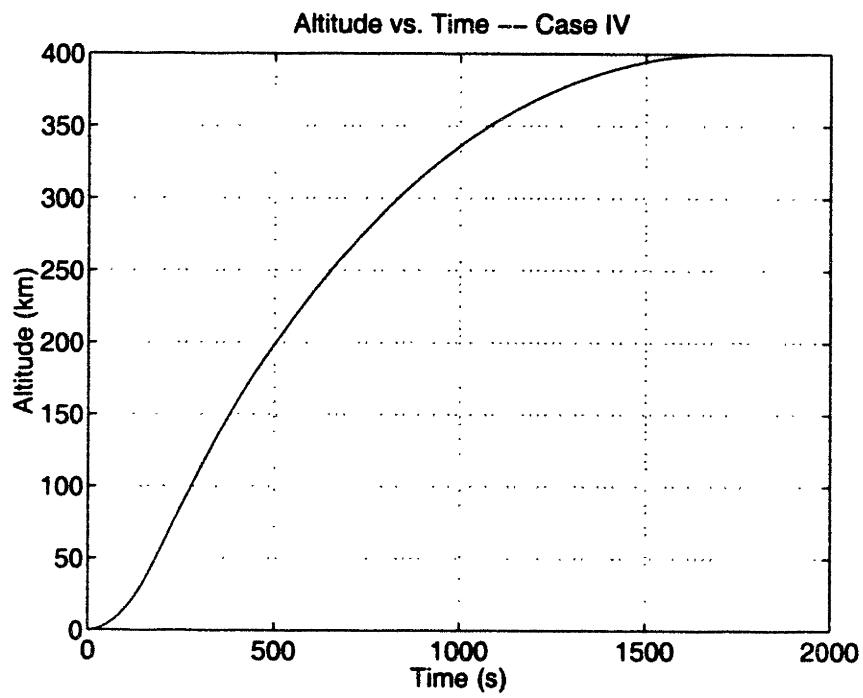


Figure 8-13: Case IV: Altitude Comparison

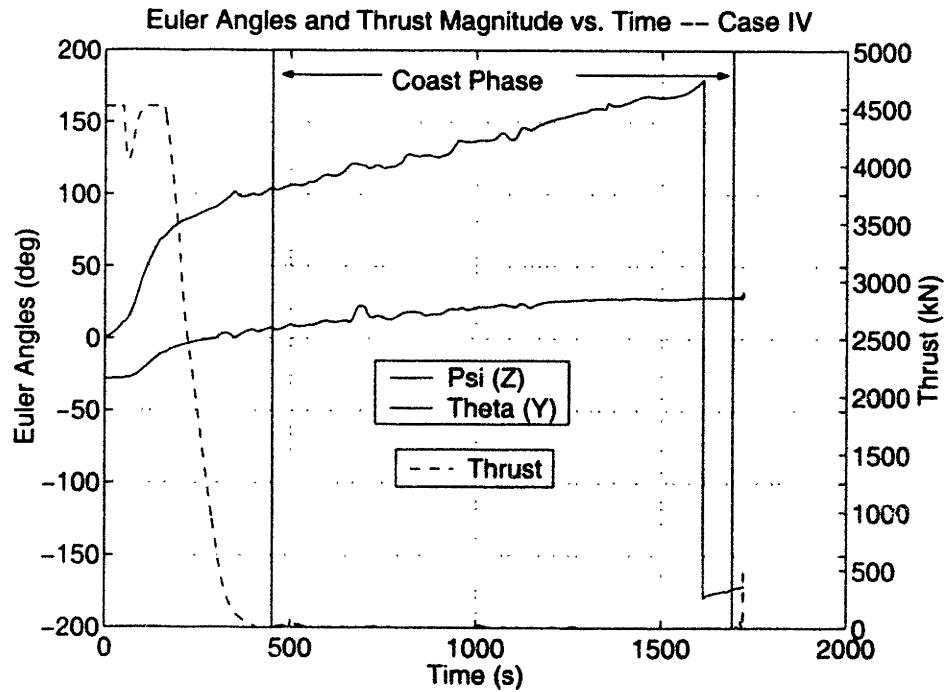


Figure 8-14: Case IV: Controls

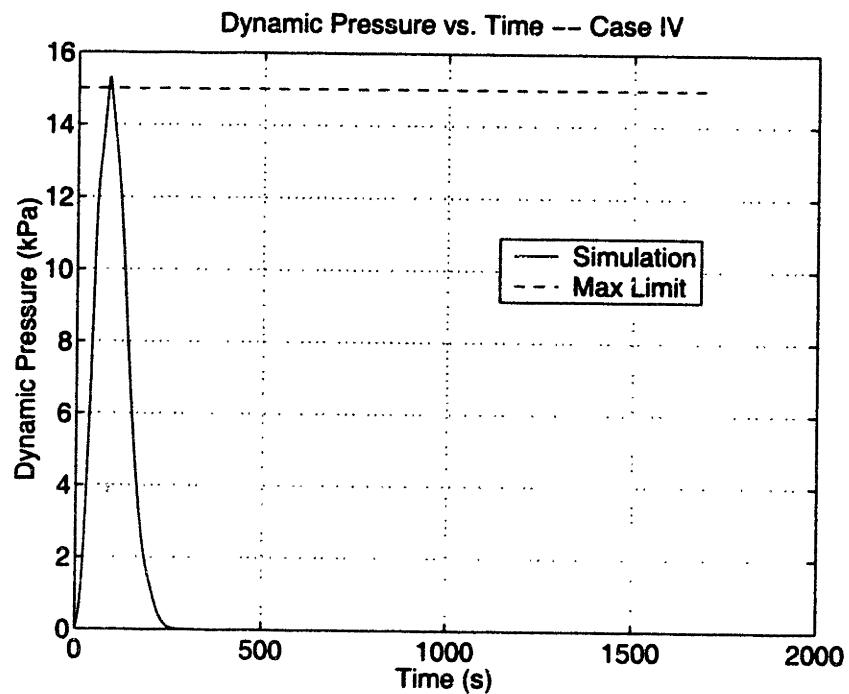


Figure 8-15: Case IV: Dynamic Pressure Comparison

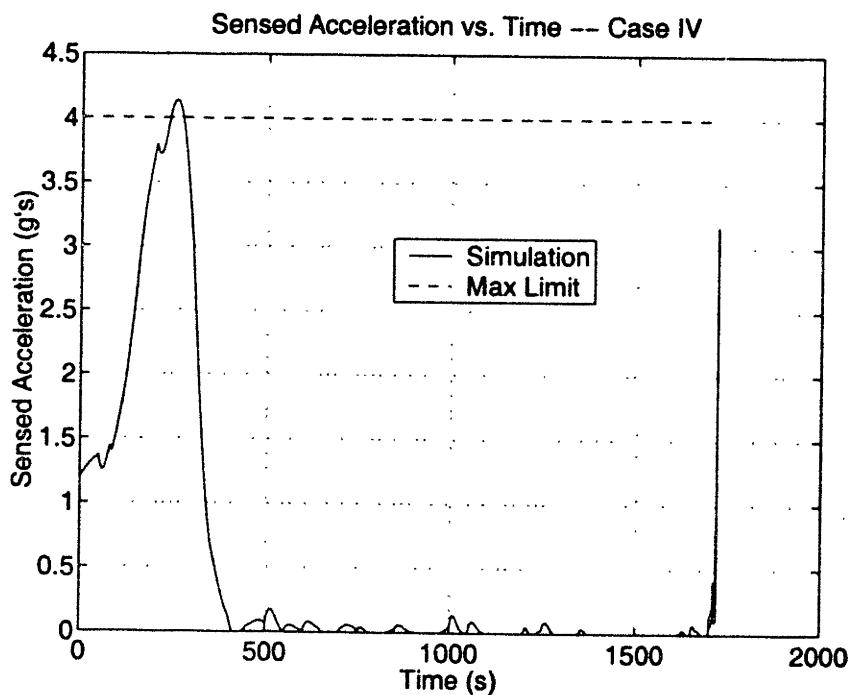


Figure 8-16: Case IV: Sensed Acceleration Comparison

8.5 Conclusions

This closed-loop guidance demonstration shows that the idea of reoptimizing a trajectory during flight can lead to improved performance over flying an open-loop trajectory. This is no surprise. It is nonetheless satisfying to see the substantial performance in meeting the end conditions. However, issues such as the time to convergence and the robustness of the optimizer still need to be investigated before a real-time algorithm is possible.

8.5.1 Lessons Learned

Some lessons learned:

- The choice of the 50 second guidance cycle was arbitrary. A smaller guidance cycle would likely lead to a better solution. The following quandary was noted during the closed loop guidance demonstration: As noted, the guidance commands were flown open-loop for 50 seconds. At some point during that time, the vehicle (between LGL points) violated the dynamic pressure constraint. If the constraints were in violation at the point of reoptimization, the optimizer would correctly observe that there was no feasible trajectory (i.e. the current state violated the constraint). This problem does not occur for the sensed acceleration constraint because the acceleration can be immediately reduced by throttling. The dynamic pressure constraint depends on velocity, which is the integral of acceleration. This can not be controlled instantaneously; planning is required. For this demonstration, the immediate solution was to relax the constraint when necessary. It is apparent that running shorter guidance cycles (less than 50 seconds) will reduce or eliminate the occurrence of this problem. Using more LGL points would also reduce this problem by giving solutions of higher fidelity. However, this would increase the computation time.
- Another problem was noted with the sensed acceleration constraint. It was initially written, as described in Section 6.2.3, in terms of the D^2 -matrix and

the gravitational acceleration. However, use of the D^2 -matrix sometimes caused a problem with this constraint, leading to very large partial derivatives for the sensed acceleration constraint. This caused the optimizer to declare a non-feasible solution. This problem was resolved by rewriting the sensed acceleration constraint directly in terms of the thrust, aerodynamic forces, and mass (see equation 6.22).

8.5.2 Ideas for the Future

This section is a discussion of ideas for using the method of this thesis for closed-loop guidance for the launch problem. Some of the ideas are simple; some are not.

- Use the lowest number of points that give a reasonable guidance solution. This will increase the speed of convergence of the numerical optimizer.
- Try to concentrate the LGL points where they are really needed. This could be done by adding soft knots to concentrate the points in certain regions of the trajectory. For example, more points are needed for the atmospheric flight to guarantee that all the constraints are met (i.e. dynamic pressure limit). Fewer points are needed for the coast phase.
- Use a soft knot to concentrate most of the points in the first 10, 20, or 30 seconds of the trajectory. This will give more fidelity for the controls in the immediate future when they will be used. Fewer points could be used for the rest of the trajectory. This would allow the entire trajectory to shape the immediate controls, while giving very good controls for the immediate guidance cycle. Think of the initial time as time zero, even as the vehicle moves forward in time. Then, the “final” time is just the time-to-go until the target is achieved. Put a soft knot at a set time, such as a point 20 seconds from the current time (time zero in this case). Let the concentration of points in this phase be high, maybe 20 points for 20 seconds. The fidelity of this solution will be very good. The rest of the trajectory can be modelled with fewer points. In this way, a

very good solution for the immediate future is available. This solution would still be influenced by the total trajectory.

- Use the numerical solution only within the atmosphere. Once outside the atmosphere, use the known near-optimal bilinear tangent guidance, as implemented by the Space Shuttle program. This method is closed-loop and near optimal, but only for exo-atmospheric flight. Some method for tying the analytic solution to the numerical solution would be required.
- It is known that part of the trajectory includes an exo-atmospheric coast. A phase could be added where this coast is built-in. This phase would only need two points. Let the initial burnout be the initial point of the phase. The final point of the phase could be determined by using the laws of orbital dynamics. In this way, no LGL points would need to be used for the coast section of the flight.
- Different stages with different dynamics could be added. For example, during the coast phase the vehicle is outside of the atmosphere, and the mass is constant. The problem could be simplified during the coast phase by dropping the aerodynamic forces and hard coding a constant mass and zero thrust. Removing variables such as these would reduce the order of the problem.
- Try running a case with many points. Intuition says this answer must be close to the true optimal since it will be of higher fidelity. Then, run the closed-loop guidance case (with only 20 points) with no perturbations between the guidance and simulation models. Run it with a small guidance cycle time (~ 10 seconds). Compare the closed-loop trajectory against the open-loop trajectory. Then, run a third case where the simulation model has perturbations (i.e. thrust is affected by atmospheric pressure, aerodynamic coefficients are off, attitude pointing has an error, etc.). Compare this trajectory against the open-loop trajectory. In this way, it can be determined how using fewer LGL points for the closed-loop solution compares with a very good open-loop solution with many LGL points.

Chapter 9

Conclusions

9.1 Summary

The Legendre Pseudospectral Method proposed by Mike Ross and Fariba Fahroo of the Postgraduate Naval School was applied successfully to several launch problems. The method was demonstrated to work for continuous and discontinuous states and controls. The method was applied to the single-stage and two-stage Goddard problem; a simplified, single-stage, two-dimensional launch problem; and a two-stage, three-dimensional launch problem. A guidance demonstration was completed to show the feasibility of using the method to find both open- and closed-loop controls for a single-stage, three-dimensional launch problem. The effects of different aerodynamic models between the optimizer and the “real” simulation were briefly analyzed. The results show that the optimizer has the potential to be used in-flight for closed-loop optimal control. However, issues surrounding time of convergence and solution robustness must still be resolved.

Both research objectives discussed in Chapter 1 were met. The Legendre Pseudospectral Method was applied to single- and multi-staged launch vehicle trajectory optimization. Also, the possible use of the method for closed-loop optimal control of launch vehicles was demonstrated.

9.2 Conclusions

The Legendre Pseudospectral Method used in this thesis works very well for many problems. It works well for all of the problems discussed in this thesis. It appears that the Legendre Pseudospectral Method could be applied to any problem that can be described by differential equations, including those that have discontinuities in states and controls.

9.3 Future Work

This research has created a number of interesting topics for further research. Many improvements to the implementation of the Legendre Pseudospectral Method were envisioned but not completed due to time and budget constraints. Some unresolved issues, as well as some ideas for improving the current application of the method, are listed below.

- How would an endo-atmospheric, closed-loop optimal guidance system affect the design and operation of rocket systems? Some questions to investigate are:
 - Can closed-loop fuel optimal endo-atmospheric guidance make it possible to reduce propellant margins, thus making it possible to add payload?
 - How will a completely closed-loop guidance scheme affect the safety margins of the launch vehicle (i.e. will the rocket be safer to fly)? Can the vehicle be made more robust with these methods?
 - How will such a capability affect launch windows (i.e. will they be bigger)?
 - How much throughput is required of the computer systems running such rapid trajectory optimization schemes? How much throughput is available from current flight computers?
- With the Legendre Pseudospectral Method, it is possible to compute the costates of the problem at the LGL points. Although this was not done in this thesis,

it could be very useful for checking the optimality of the solution. See [8] for more information.

- As stated, NPSOL was the numerical optimizer used in this thesis. However, a different numerical optimizer may have properties that make it more desirable for the launch problem. For example, SNOPT is a numerical optimizer that uses sparse matrix techniques. The Jacobian matrix of the launch problem is sparse. Taking advantage of sparse matrix techniques will speed convergence of the code.
- The tabular aerodynamic data was used with success. However, the linear interpolation slowed the optimization somewhat. A program called “lqlspine,” written by Hans Seywald at Analytical Mechanics Associates, Inc. (AMA) and Dan Moerder at NASA Langley, was investigated for this thesis, but not implemented due to time and budget constraints. It is a curve-fitting/interpolating program that may improve the use of tabular data. In particular, it has the potential to increase the accuracy of the estimated partial derivatives of the aerodynamic tabular data.
- In the research of this thesis, the NPSOL option of “cold starts” were used. There is another option called “warm starts.” See [11] for more information. This option was not used at all in this thesis, but it has the potential for speeding up convergence.
- In the application of this thesis, MATLAB scripts were used to code everything. Recoding the program in C or FORTRAN will yield a faster code. Compiling the existing MATLAB code may also increase the speed.
- It may be worthwhile to make another comparison between the possible coordinate systems for the launch problem. There may be better methods for comparison.
- A Kistler K-1 type vehicle was modelled in the example. However, the first stage of the K-1 is designed to fly back to the launch site. This was not modelled in

this thesis. Downrange constraints on the first stage would definitely affect the entire optimal trajectory (i.e. The first stage can't fly too far downrange if it is to return to the launch site). However, the fly-back portion of the first stage flight could be included in the problem by extension of the patching method described in Chapter 3.

Appendix A

Notes on Notation

1. Scalar notation for multiplication and division is used to indicate term-by-term operations on vectors.

$$\vec{m}\vec{g}_x = \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ \vdots \\ m_n \end{bmatrix} \begin{bmatrix} g_{x_1} \\ g_{x_2} \\ g_{x_3} \\ \vdots \\ g_{x_n} \end{bmatrix} = \begin{bmatrix} m_1 g_{x_1} \\ m_2 g_{x_2} \\ m_3 g_{x_3} \\ \vdots \\ m_n g_{x_n} \end{bmatrix}$$

2. In many Jacobian calculations, diagonal matrices are necessary. The notation for a diagonal matrix is $\langle \rangle$. Example:

$$\langle -1 \rangle = \begin{bmatrix} -1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 0 & \cdots & 0 \\ 0 & 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & -1 \end{bmatrix}$$

3. The optimization vector \vec{x}_{opt} is usually treated as a column vector. However, when finding the Jacobian matrix, the optimization vector is treated as a row vector.

4. Values in square brackets are matrices. Sometimes, values in square brackets may represent row or column “matrices,” depending on the context. For example, $[0]$ is a matrix of zeros. The size of the matrix should be apparent from the context.

Appendix B

Matrix Math

Matrix manipulation is very important in the calculation of the analytic Jacobian. This appendix serves as a short “course” on how to represent partial derivatives of vectors with respect to other vectors.

B.1 Matrix Derivatives

Let there be a column vector of length m .

$$\vec{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_m \end{bmatrix} \quad (\text{B.1})$$

It is desired to take the derivative of this vector with respect to a row vector of length n .

$$\vec{x} = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \end{bmatrix} \quad (\text{B.2})$$

The result is a matrix.

$$\frac{d\vec{y}}{d\vec{x}} = \begin{bmatrix} \frac{dy_1}{dx_1} & \frac{dy_1}{dx_2} & \frac{dy_1}{dx_3} & \cdots & \frac{dy_1}{dx_n} \\ \frac{dy_2}{dx_1} & \frac{dy_2}{dx_2} & \frac{dy_2}{dx_3} & \cdots & \frac{dy_2}{dx_n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{dy_m}{dx_1} & \frac{dy_m}{dx_2} & \frac{dy_m}{dx_3} & \cdots & \frac{dy_m}{dx_n} \end{bmatrix} \quad (\text{B.3})$$

B.2 Chain Rule for Matrix Derivatives

Let there be two column vectors, both of length m .

$$\vec{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix} \quad \vec{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \quad (\text{B.4})$$

Define another column vector as:

$$\vec{y} = \vec{a}^T \vec{b} = \begin{bmatrix} a_1 & a_2 & \cdots & a_m \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} = \begin{bmatrix} a_1 b_1 \\ a_2 b_2 \\ \vdots \\ a_m b_m \end{bmatrix} \quad (\text{B.5})$$

Using the notation described in Appendix A, \vec{y} can also be written as:

$$\vec{y} = \vec{a}\vec{b} \quad (\text{B.6})$$

This new notation is adopted because it is less cluttered. It also allows an easier analogy between scalar and matrix differentiation.

Assume that both \vec{a} and \vec{b} are functions of the n -length row vector \vec{x} . The chain

rule must be used to find the derivative of \vec{y} with respect to \vec{x} .

$$\frac{d\vec{y}}{d\vec{x}} = \begin{bmatrix} \frac{da_1}{dx_1} b_1 + a_1 \frac{db_1}{dx_1} & \frac{da_1}{dx_2} b_1 + a_1 \frac{db_1}{dx_2} & \cdots & \frac{da_1}{dx_n} b_1 + a_1 \frac{db_1}{dx_n} \\ \frac{da_2}{dx_1} b_2 + a_2 \frac{db_2}{dx_1} & \frac{da_2}{dx_2} b_2 + a_2 \frac{db_2}{dx_2} & \cdots & \frac{da_2}{dx_n} b_2 + a_2 \frac{db_2}{dx_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{da_m}{dx_1} b_m + a_m \frac{db_m}{dx_1} & \frac{da_m}{dx_2} b_m + a_m \frac{db_m}{dx_2} & \cdots & \frac{da_m}{dx_n} b_m + a_m \frac{db_m}{dx_n} \end{bmatrix} \quad (\text{B.7})$$

This matrix can be rewritten as the sum of two matrices:

$$\frac{d\vec{y}}{d\vec{x}} = \begin{bmatrix} b_1 \frac{da_1}{dx_1} & b_1 \frac{da_1}{dx_2} & \cdots & b_1 \frac{da_1}{dx_n} \\ b_2 \frac{da_2}{dx_1} & b_2 \frac{da_2}{dx_2} & \cdots & b_2 \frac{da_2}{dx_n} \\ \vdots & \vdots & \ddots & \vdots \\ b_m \frac{da_m}{dx_1} & b_m \frac{da_m}{dx_2} & \cdots & b_m \frac{da_m}{dx_n} \end{bmatrix} + \begin{bmatrix} a_1 \frac{db_1}{dx_1} & a_1 \frac{db_1}{dx_2} & \cdots & a_1 \frac{db_1}{dx_n} \\ a_2 \frac{db_2}{dx_1} & a_2 \frac{db_2}{dx_2} & \cdots & a_2 \frac{db_2}{dx_n} \\ \vdots & \vdots & \ddots & \vdots \\ a_m \frac{db_m}{dx_1} & a_m \frac{db_m}{dx_2} & \cdots & a_m \frac{db_m}{dx_n} \end{bmatrix} \quad (\text{B.8})$$

Now, these matrices can be rewritten in the following way:

$$\begin{aligned} \frac{d\vec{y}}{d\vec{x}} &= \begin{bmatrix} b_1 & 0 & 0 & \cdots & 0 \\ 0 & b_2 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & b_m \end{bmatrix} \begin{bmatrix} \frac{da_1}{dx_1} & \frac{da_1}{dx_2} & \frac{da_1}{dx_3} & \cdots & \frac{da_1}{dx_n} \\ \frac{da_2}{dx_1} & \frac{da_2}{dx_2} & \frac{da_2}{dx_3} & \cdots & \frac{da_2}{dx_n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{da_m}{dx_1} & \frac{da_m}{dx_2} & \frac{da_m}{dx_3} & \cdots & \frac{da_m}{dx_n} \end{bmatrix} \\ &+ \begin{bmatrix} a_1 & 0 & 0 & \cdots & 0 \\ 0 & a_2 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_m \end{bmatrix} \begin{bmatrix} \frac{db_1}{dx_1} & \frac{db_1}{dx_2} & \frac{db_1}{dx_3} & \cdots & \frac{db_1}{dx_n} \\ \frac{db_2}{dx_1} & \frac{db_2}{dx_2} & \frac{db_2}{dx_3} & \cdots & \frac{db_2}{dx_n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{db_m}{dx_1} & \frac{db_m}{dx_2} & \frac{db_m}{dx_3} & \cdots & \frac{db_m}{dx_n} \end{bmatrix} \end{aligned} \quad (\text{B.9})$$

The notation defined in Appendix A can be used to write the above matrix equation in the following form:

$$\frac{d\vec{y}}{d\vec{x}} = \langle \vec{b} \rangle \frac{d\vec{a}}{d\vec{x}} + \langle \vec{a} \rangle \frac{d\vec{b}}{d\vec{x}} \quad (\text{B.10})$$

The similarity between the matrix chain rule and the scalar chain rule can be seen. The familiar scalar version of the chain rule can be used on vector equations, as long as the diagonal matrix is on the left of the derivative matrix.

B.3 Distributive Law for Matrices

The distributive law for multiplication and addition with scalars also holds for matrices [9, p. 15].

$$\begin{aligned}(A + B)C &= AC + BC \\ C(A + B) &= CA + CB\end{aligned}\tag{B.11}$$

Note that a right side matrix multiplication is generally not equivalent to a left side matrix multiplication.

Appendix C

Atmosphere Models

C.1 Exponential Atmosphere Model

The atmospheric density can be modelled as:

$$\tilde{\rho} = \rho_o e^{-\beta(\bar{R}-R_P)} \quad (\text{C.1})$$

where

- R = radial distance from the center of the planet
 R_P = radius of the planet
 ρ_o = atmospheric density at the surface of the planet
 β = atmospheric scale factor

The partial derivative matrix of the atmospheric density with respect to radial position is given by:

$$\frac{\partial \tilde{\rho}}{\partial \bar{R}} = \left\langle -\beta \rho_o e^{-\beta(\bar{R}-R_P)} \right\rangle \quad (\text{C.2})$$

The atmospheric density at the surface of the Earth is [1]:

$$\rho_o = 1.225 \text{ kg/m}^3 \quad (\text{C.3})$$

The parameter β is the inverse of the atmospheric scale height (h_{scale}). The scale height is [22]:

$$h_{scale} = 8500 \text{ m} \quad (\text{C.4})$$

$$\beta = \frac{1}{h_{scale}} = \frac{1}{8500} \text{ m}^{-1} \quad (\text{C.5})$$

C.2 Temperature-Based Atmosphere Model

The temperature-based atmosphere model is taken from [1, pp. 74-78].

Appendix D

Initial State Definition

The initial state of the launch vehicle is defined in terms of the topocentric-horizon launch coordinate system. These coordinates lie on the surface of the planet at the launch point. The X_L -axis points south, the Y_L -axis points east, and the Z_L -axis points directly up. Figure D-1 shows the relationship between the launch coordinates and the body coordinates. It is assumed that the body X_B -axis nominally points in the direction of travel. It is also assumed that the body Z_B -axis is the axis that nominally points “down.” However, this is only important for a non-axisymmetric vehicle. The initial position, velocity, and attitude of the vehicle are completely specified by the following parameters:

λ_o = initial inertial latitude

L_o = initial inertial longitude

\vec{R}_{L_o} = initial position in launch coordinates

\vec{V}_{L_o} = initial velocity in launch coordinates

E_L = initial elevation of launch vehicle

A_Z = initial azimuth of launch vehicle

From Figure D-1, it can be seen that the relationship between the launch coordi-

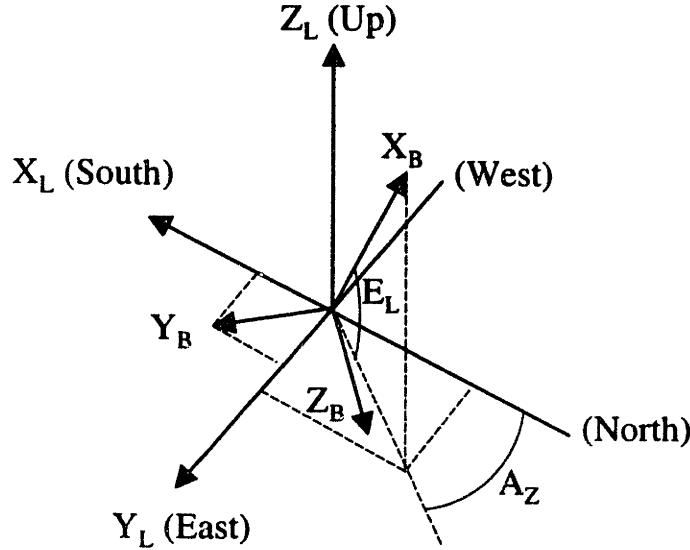


Figure D-1: Topocentric-Horizon Launch Coordinate System (adapted from [2])

nates and the body coordinates is:

$$\begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix} = \begin{bmatrix} -\cos(E_L)\cos(A_Z) & \sin(A_Z) & -\sin(E_L)\cos(A_Z) \\ \cos(E_L)\sin(A_Z) & \cos(A_Z) & \sin(E_L)\sin(A_Z) \\ \sin(E_L) & 0 & -\cos(E_L) \end{bmatrix} \begin{bmatrix} X_B \\ Y_B \\ Z_B \end{bmatrix} \quad (\text{D.1})$$

Define this matrix as $A_{L/B}$. In vector-matrix notation this becomes

$$\vec{R}_L = A_{L/B} \vec{R}_B \quad (\text{D.2})$$

The relationship between the launch coordinates and the inertial coordinates is shown in Figure D-2. It can be seen that the relationship between the inertial coordinates and launch coordinates is:

$$\begin{bmatrix} X_I \\ Y_I \\ Z_I \end{bmatrix} = \begin{bmatrix} \sin \lambda_I \cos L_I & -\sin L_I & \cos \lambda_I \cos L_I \\ \sin \lambda_I \sin L_I & \cos L_I & \cos \lambda_I \sin L_I \\ -\cos \lambda_I & 0 & \sin \lambda_I \end{bmatrix} \begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix} \quad (\text{D.3})$$

Define this matrix as $A_{I/L}$. Note that the launch coordinate system is fixed to the

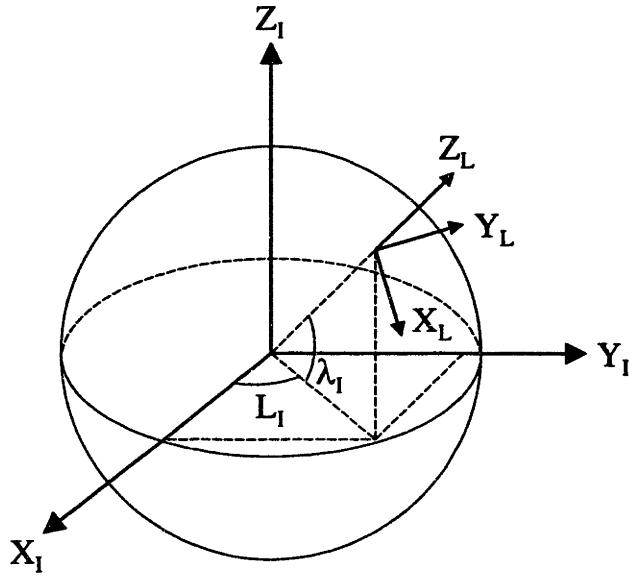


Figure D-2: Relationship Between Launch Coordinates and Inertial Coordinates

surface of the planet. When converting the position of a vehicle from launch coordinates to inertial coordinates, the radius of the planet (R_P) must be added to the Z_L -component of the position.

$$\vec{R}_I = A_{I/L} (\vec{R}_L + R_P \hat{k}_L) \quad (D.4)$$

When converting the vehicle velocity from launch coordinates to inertial coordinates, the velocity due to the rotation of the planet must be added.

$$\vec{V}_I = A_{I/L} \vec{V}_L + \vec{\omega}_I \times \vec{R}_I \quad (D.5)$$

where $\vec{\omega}_I$ is the rotation rate of the planet in inertial coordinates.

The attitude matrix defining the relationship between the inertial coordinates and the body coordinates is:

$$A_{I/B} = A_{I/L} A_{L/B} \quad (D.6)$$

This matrix defines the initial inertial attitude of the vehicle. The equations from section 6.6.2 can be used to find the initial quaternion elements from $A_{I/B}$.

[This page intentionally left blank.]

Appendix E

Initial Guess

It is necessary to give the nonlinear program solver an initial guess for the optimization vector. Usually, the closer the guess is to the optimal solution, the faster the nonlinear program solver will converge. So, it is desired to have an initial guess that is close to the optimal guess. For problems that are being run for the first time, this can be a problem, especially for very complex problems. However, the launch problem has been solved. Therefore, some idea of what the solution will look like is available. This can be used to aid in initial guess generation.

A fast method of generating initial guesses is desired. One method would be to make a guess for the controls (in the case of the launch problem, the controls are the thrust magnitude/direction and the vehicle attitude), and iterate the equations of motion forward. However, integration of the equations of motion can take some time.

A method that works very well and extremely fast is presented in this appendix. This method works very well for the numerical optimization problem presented in this thesis. The general idea is to use N -th order polynomials (power series) in time to estimate the flight trajectory.

$$y(t) = \sum_{i=1}^{N+1} C_i t^{i-1} = C_1 + C_2 t + C_3 t^2 + \dots + C_{N+1} t^N \quad (\text{E.1})$$

In particular, a third order polynomial works very well for this problem ($N=3$).

With a third order polynomial, the initial value, final value, initial first derivative, and final first derivative can be directly controlled. Let there be a third order polynomial that models the time history of the value y .

$$y(t) = C_1 + C_2 t + C_3 t^2 + C_4 t^3 \quad (\text{E.2})$$

The following conditions must hold:

- At the initial time, t_o

$$y(t_o) = y_o \quad (\text{E.3})$$

$$\dot{y}(t_o) = \dot{y}_o \quad (\text{E.4})$$

- At the final time, t_f

$$y(t_f) = y_f \quad (\text{E.5})$$

$$\dot{y}(t_f) = \dot{y}_f \quad (\text{E.6})$$

It is desired to find the coefficients of the polynomial (C_i) that will give the desired initial and final conditions. The following four equations can be written:

$$y_o = y(t_o) = C_1 + C_2 t_o + C_3 t_o^2 + C_4 t_o^3 \quad (\text{E.7})$$

$$\dot{y}_o = \dot{y}(t_o) = C_2 + 2C_3 t_o + 3C_4 t_o^2 \quad (\text{E.8})$$

$$y_f = y(t_f) = C_1 + C_2 t_f + C_3 t_f^2 + C_4 t_f^3 \quad (\text{E.9})$$

$$\dot{y}_f = \dot{y}(t_f) = C_2 + 2C_3 t_f + 3C_4 t_f^2 \quad (\text{E.10})$$

These equations can be rewritten in matrix form.

$$\begin{bmatrix} 1 & t_o & t_o^2 & t_o^3 \\ 0 & 1 & 2t_o & 3t_o^2 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix} = \begin{bmatrix} y_o \\ \dot{y}_o \\ y_f \\ \dot{y}_f \end{bmatrix} \quad (\text{E.11})$$

Given y_o , \dot{y}_o , y_f , \dot{y}_f , t_o and t_f , the coefficients can be easily found by solving the matrix equation:

$$\begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix} = \begin{bmatrix} 1 & t_o & t_o^2 & t_o^3 \\ 0 & 1 & 2t_o & 3t_o^2 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix}^{-1} \begin{bmatrix} y_o \\ \dot{y}_o \\ y_f \\ \dot{y}_f \end{bmatrix} \quad (\text{E.12})$$

Any suitable method can be used to invert the matrix.

For a two dimensional launch problem, the trajectory can be represented by polynomial equations. Equation E.12 can be used to find the coefficients of the polynomials. The method used in this thesis uses an inertial polar coordinate system (not fixed to the planet). Three polynomials are used: one for altitude, one for angular position, and one for vehicle mass.

For launching from the surface of a planet into a circular orbit, the following initial and final conditions must hold.

- Altitude (h)

$$h_o = 0 \quad (\text{E.13})$$

$$h_f = \text{desired final altitude} \quad (\text{E.14})$$

$$\dot{h}_o = 0 \quad (\text{E.15})$$

$$\dot{h}_f = 0 \quad (\text{E.16})$$

- Angular Position (θ)

$$\theta_o = 0 \quad (\text{E.17})$$

$$\theta_f = \text{final angular position guess} \quad (\text{E.18})$$

$$\dot{\theta}_o = \Omega_P \quad (\text{E.19})$$

$$\dot{\theta}_f = \frac{V_C}{R_P + h_f} \quad (\text{E.20})$$

where R_P is the planet radius, Ω_P is the angular rotation rate of the planet, and V_C is the circular orbit velocity [2]:

$$V_C = \sqrt{\frac{\mu}{R_P + h_f}}$$

where μ is the gravitational parameter of the planet.

- Mass (m)

Assume that the initial mass (m_o) of the vehicle and how much propellant is available (m_p) is known. For purposes of making an initial guess, assume that all of the available propellant is used. Also, assume that the initial mass flow rate is such that the initial thrust is at a maximum and that this value is known.

$$m_o = \text{initial mass} \quad (\text{E.21})$$

$$m_f = m_o - m_p \quad (\text{E.22})$$

$$\dot{m}_o = \dot{m}_{max} \quad (\text{E.23})$$

$$\dot{m}_f = 0 \quad (\text{E.24})$$

Say that a guess for the radial-transverse polar coordinate system from Chapter 4 is desired. As an initial guess, the polynomial equations for the radial position (R), angular position (θ), and mass of the vehicle (m) are available. A reasonable guess for the final time is also known. It only remains to find the thrusts in the radial and transverse directions.

Note that by differentiating the polynomial equations, the inertial velocities and accelerations in the radial and transverse directions can be found. Equations 5.36 define the dynamic equations for the radial-transverse polar coordinate system. They are rewritten below for reference.

$$\dot{R} = V_R$$

$$\dot{\theta} R = V_\theta$$

$$\begin{aligned}\ddot{R} &= \frac{T_R}{m} + \frac{D_R}{m} - g + \frac{V_\theta^2}{R} \\ \ddot{\theta} &= \frac{T_\theta}{mR} + \frac{D_\theta}{mR} - \frac{2V_R V_\theta}{R^2}\end{aligned}\quad (\text{E.25})$$

$$\dot{m} = -\frac{(T_R^2 + T_\theta^2)^{1/2}}{g_o I_{sp}} \quad (\text{E.26})$$

The drag in the radial and transverse directions are given by equation 4.36, repeated below.

$$\begin{aligned}\vec{D}_R &= \frac{1}{2} \vec{\rho} (\vec{V}_R^2 + \vec{V}_\theta^2) C_D A_{ref} \frac{-\vec{V}_R}{(\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2}} = -\frac{1}{2} \vec{\rho} \vec{V}_R (\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2} C_D A_{ref} \\ \vec{D}_\theta &= \frac{1}{2} \vec{\rho} (\vec{V}_R^2 + \vec{V}_\theta^2) C_D A_{ref} \frac{-\vec{V}_\theta}{(\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2}} = -\frac{1}{2} \vec{\rho} \vec{V}_\theta (\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2} C_D A_{ref}\end{aligned}\quad (\text{E.27})$$

The gravity is defined from equation 4.38, repeated below.

$$\vec{g} = \frac{\mu}{\vec{R}^2} \quad (\text{E.28})$$

It is possible to solve the dynamic equations for the values of T_R and T_θ . Note that the equation for \dot{m} is not used for this solution, and that it may not even be satisfied. However, for an initial guess, this is quite acceptable. This method is used to find estimates for the states and controls at each LGL point. This method can also be extended to the normal-tangential and cartesian coordinate systems. It is only

necessary to make the proper conversion from the radial-transverse coordinate system of the polynomials. Similarly, this method can be extended to three dimensions by adding a fourth polynomial to model a third dimension (either spherical or cylindrical coordinates would work).

Appendix F

Coordinate System Comparison: Jacobian Derivation

F.1 Cartesian Coordinate System Jacobian

The Jacobian for the two dimensional, cartesian coordinate system from section 4.1 is defined below. The Jacobian matrix is:

$$C_{Jac} = \frac{d\vec{C}}{d\vec{x}_{opt}} = \begin{bmatrix} \frac{\partial \vec{C}_{R_x}}{\partial R_x} & \frac{\partial \vec{C}_{R_x}}{\partial R_y} & \frac{\partial \vec{C}_{R_x}}{\partial V_x} & \frac{\partial \vec{C}_{R_x}}{\partial V_y} & \frac{\partial \vec{C}_{R_x}}{\partial \bar{m}} & \frac{\partial \vec{C}_{R_x}}{\partial T_x} & \frac{\partial \vec{C}_{R_x}}{\partial T_y} & \frac{\partial \vec{C}_{R_x}}{\partial \tau_f} \\ \frac{\partial \vec{C}_{R_y}}{\partial R_x} & \frac{\partial \vec{C}_{R_y}}{\partial R_y} & \frac{\partial \vec{C}_{R_y}}{\partial V_x} & \frac{\partial \vec{C}_{R_y}}{\partial V_y} & \frac{\partial \vec{C}_{R_y}}{\partial \bar{m}} & \frac{\partial \vec{C}_{R_y}}{\partial T_x} & \frac{\partial \vec{C}_{R_y}}{\partial T_y} & \frac{\partial \vec{C}_{R_y}}{\partial \tau_f} \\ \frac{\partial \vec{C}_{V_x}}{\partial R_x} & \frac{\partial \vec{C}_{V_x}}{\partial R_y} & \frac{\partial \vec{C}_{V_x}}{\partial V_x} & \frac{\partial \vec{C}_{V_x}}{\partial V_y} & \frac{\partial \vec{C}_{V_x}}{\partial \bar{m}} & \frac{\partial \vec{C}_{V_x}}{\partial T_x} & \frac{\partial \vec{C}_{V_x}}{\partial T_y} & \frac{\partial \vec{C}_{V_x}}{\partial \tau_f} \\ \frac{\partial \vec{C}_{V_y}}{\partial R_x} & \frac{\partial \vec{C}_{V_y}}{\partial R_y} & \frac{\partial \vec{C}_{V_y}}{\partial V_x} & \frac{\partial \vec{C}_{V_y}}{\partial V_y} & \frac{\partial \vec{C}_{V_y}}{\partial \bar{m}} & \frac{\partial \vec{C}_{V_y}}{\partial T_x} & \frac{\partial \vec{C}_{V_y}}{\partial T_y} & \frac{\partial \vec{C}_{V_y}}{\partial \tau_f} \\ \frac{\partial \vec{C}_{\bar{m}}}{\partial R_x} & \frac{\partial \vec{C}_{\bar{m}}}{\partial R_y} & \frac{\partial \vec{C}_{\bar{m}}}{\partial V_x} & \frac{\partial \vec{C}_{\bar{m}}}{\partial V_y} & \frac{\partial \vec{C}_{\bar{m}}}{\partial \bar{m}} & \frac{\partial \vec{C}_{\bar{m}}}{\partial T_x} & \frac{\partial \vec{C}_{\bar{m}}}{\partial T_y} & \frac{\partial \vec{C}_{\bar{m}}}{\partial \tau_f} \\ \frac{\partial \vec{C}_T}{\partial R_x} & \frac{\partial \vec{C}_T}{\partial R_y} & \frac{\partial \vec{C}_T}{\partial V_x} & \frac{\partial \vec{C}_T}{\partial V_y} & \frac{\partial \vec{C}_T}{\partial \bar{m}} & \frac{\partial \vec{C}_T}{\partial T_x} & \frac{\partial \vec{C}_T}{\partial T_y} & \frac{\partial \vec{C}_T}{\partial \tau_f} \\ \frac{\partial \vec{C}_a}{\partial R_x} & \frac{\partial \vec{C}_a}{\partial R_y} & \frac{\partial \vec{C}_a}{\partial V_x} & \frac{\partial \vec{C}_a}{\partial V_y} & \frac{\partial \vec{C}_a}{\partial \bar{m}} & \frac{\partial \vec{C}_a}{\partial T_x} & \frac{\partial \vec{C}_a}{\partial T_y} & \frac{\partial \vec{C}_a}{\partial \tau_f} \\ \frac{\partial C_{R_f}}{\partial R_x} & \frac{\partial C_{R_f}}{\partial R_y} & \frac{\partial C_{R_f}}{\partial V_x} & \frac{\partial C_{R_f}}{\partial V_y} & \frac{\partial C_{R_f}}{\partial \bar{m}} & \frac{\partial C_{R_f}}{\partial T_x} & \frac{\partial C_{R_f}}{\partial T_y} & \frac{\partial C_{R_f}}{\partial \tau_f} \\ \frac{\partial C_{V_f}}{\partial R_x} & \frac{\partial C_{V_f}}{\partial R_y} & \frac{\partial C_{V_f}}{\partial V_x} & \frac{\partial C_{V_f}}{\partial V_y} & \frac{\partial C_{V_f}}{\partial \bar{m}} & \frac{\partial C_{V_f}}{\partial T_x} & \frac{\partial C_{V_f}}{\partial T_y} & \frac{\partial C_{V_f}}{\partial \tau_f} \\ \frac{\partial C_{\bar{m},v}}{\partial R_x} & \frac{\partial C_{\bar{m},v}}{\partial R_y} & \frac{\partial C_{\bar{m},v}}{\partial V_x} & \frac{\partial C_{\bar{m},v}}{\partial V_y} & \frac{\partial C_{\bar{m},v}}{\partial \bar{m}} & \frac{\partial C_{\bar{m},v}}{\partial T_x} & \frac{\partial C_{\bar{m},v}}{\partial T_y} & \frac{\partial C_{\bar{m},v}}{\partial \tau_f} \end{bmatrix}$$

Partial Derivatives of \dot{R}_x Constraint

$$\begin{aligned}
\frac{\partial \tilde{C}_{\dot{R}_x}}{\partial \vec{R}_x} &= D_{NN} & \frac{\partial \tilde{C}_{\dot{R}_x}}{\partial \vec{m}} &= [0] \\
\frac{\partial \tilde{C}_{\dot{R}_x}}{\partial \vec{R}_y} &= [0] & \frac{\partial \tilde{C}_{\dot{R}_x}}{\partial \vec{T}_x} &= [0] \\
\frac{\partial \tilde{C}_{\dot{R}_x}}{\partial \vec{V}_x} &= \langle -1 \rangle & \frac{\partial \tilde{C}_{\dot{R}_x}}{\partial \vec{T}_y} &= [0] \\
\frac{\partial \tilde{C}_{\dot{R}_x}}{\partial \vec{V}_y} &= [0] & \frac{\partial \tilde{C}_{\dot{R}_x}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{R}_x \right]
\end{aligned} \tag{F.1}$$

Partial Derivatives of \dot{R}_y Constraint

$$\begin{aligned}
\frac{\partial \tilde{C}_{\dot{R}_y}}{\partial \vec{R}_x} &= [0] & \frac{\partial \tilde{C}_{\dot{R}_y}}{\partial \vec{m}} &= [0] \\
\frac{\partial \tilde{C}_{\dot{R}_y}}{\partial \vec{R}_y} &= D_{NN} & \frac{\partial \tilde{C}_{\dot{R}_y}}{\partial \vec{T}_x} &= [0] \\
\frac{\partial \tilde{C}_{\dot{R}_y}}{\partial \vec{V}_x} &= [0] & \frac{\partial \tilde{C}_{\dot{R}_y}}{\partial \vec{T}_y} &= [0] \\
\frac{\partial \tilde{C}_{\dot{R}_y}}{\partial \vec{V}_y} &= \langle -1 \rangle & \frac{\partial \tilde{C}_{\dot{R}_y}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{R}_x \right]
\end{aligned} \tag{F.2}$$

Partial Derivatives of \dot{V}_x Constraint

$$\begin{aligned}
\frac{\partial \tilde{C}_{\dot{V}_x}}{\partial \vec{R}_x} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_x}{\partial \vec{R}_x} - \frac{\partial \vec{g}_x}{\partial \vec{R}_x} & \frac{\partial \tilde{C}_{\dot{V}_x}}{\partial \vec{m}} &= \left\langle \frac{\vec{T}_x + \vec{D}_x}{\vec{m}^2} \right\rangle \\
\frac{\partial \tilde{C}_{\dot{V}_x}}{\partial \vec{R}_y} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_x}{\partial \vec{R}_y} - \frac{\partial \vec{g}_x}{\partial \vec{R}_y} & \frac{\partial \tilde{C}_{\dot{V}_x}}{\partial \vec{T}_x} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \\
\frac{\partial \tilde{C}_{\dot{V}_x}}{\partial \vec{V}_x} &= D_{NN} - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_x}{\partial \vec{V}_x} & \frac{\partial \tilde{C}_{\dot{V}_x}}{\partial \vec{T}_y} &= [0] \\
\frac{\partial \tilde{C}_{\dot{V}_x}}{\partial \vec{V}_y} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_x}{\partial \vec{V}_y} & \frac{\partial \tilde{C}_{\dot{V}_x}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{V}_x \right]
\end{aligned} \tag{F.3}$$

Partial Derivatives of \dot{V}_y Constraint

$$\begin{aligned}
\frac{\partial \tilde{C}_{\dot{V}_y}}{\partial \vec{R}_x} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_y}{\partial \vec{R}_x} - \frac{\partial \vec{g}_y}{\partial \vec{R}_x} & \frac{\partial \tilde{C}_{\dot{V}_y}}{\partial \vec{m}} &= \left\langle \frac{\vec{T}_y + \vec{D}_y}{\vec{m}^2} \right\rangle \\
\frac{\partial \tilde{C}_{\dot{V}_y}}{\partial \vec{R}_y} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_y}{\partial \vec{R}_y} - \frac{\partial \vec{g}_y}{\partial \vec{R}_y} & \frac{\partial \tilde{C}_{\dot{V}_y}}{\partial \vec{T}_x} &= [0] \\
\frac{\partial \tilde{C}_{\dot{V}_y}}{\partial \vec{V}_x} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_y}{\partial \vec{V}_x} & \frac{\partial \tilde{C}_{\dot{V}_y}}{\partial \vec{T}_y} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \\
\frac{\partial \tilde{C}_{\dot{V}_y}}{\partial \vec{V}_y} &= D_{NN} - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_y}{\partial \vec{V}_y} & \frac{\partial \tilde{C}_{\dot{V}_y}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{V}_y \right]
\end{aligned} \tag{F.4}$$

Partial Derivatives of \dot{m} Constraint

$$\begin{aligned}
 \frac{\partial \tilde{C}_{\dot{m}}}{\partial R_x} &= [0] & \frac{\partial \tilde{C}_{\dot{m}}}{\partial \tilde{m}} &= D_{NN} \\
 \frac{\partial \tilde{C}_{\dot{m}}}{\partial R_y} &= [0] & \frac{\partial \tilde{C}_{\dot{m}}}{\partial T_x} &= \frac{1}{g_o I_{sp}} \left\langle \frac{\vec{T}_x}{\sqrt{\vec{T}_x^2 + \vec{T}_y^2}} \right\rangle \\
 \frac{\partial \tilde{C}_{\dot{m}}}{\partial V_x} &= [0] & \frac{\partial \tilde{C}_{\dot{m}}}{\partial T_y} &= \frac{1}{g_o I_{sp}} \left\langle \frac{\vec{T}_y}{\sqrt{\vec{T}_x^2 + \vec{T}_y^2}} \right\rangle \\
 \frac{\partial \tilde{C}_{\dot{m}}}{\partial V_y} &= [0] & \frac{\partial \tilde{C}_{\dot{m}}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \tilde{m} \right]
 \end{aligned} \tag{F.5}$$

Partial Derivatives of Thrust Magnitude Constraint

$$\begin{aligned}
 \frac{\partial \tilde{C}_T}{\partial R_x} &= [0] & \frac{\partial \tilde{C}_T}{\partial \tilde{m}} &= [0] \\
 \frac{\partial \tilde{C}_T}{\partial R_y} &= [0] & \frac{\partial \tilde{C}_T}{\partial T_x} &= \langle 2\vec{T}_x \rangle \\
 \frac{\partial \tilde{C}_T}{\partial V_x} &= [0] & \frac{\partial \tilde{C}_T}{\partial T_y} &= \langle 2\vec{T}_y \rangle \\
 \frac{\partial \tilde{C}_T}{\partial V_y} &= [0] & \frac{\partial \tilde{C}_T}{\partial \tau_f} &= [0]
 \end{aligned} \tag{F.6}$$

Partial Derivatives of Dynamic Pressure Constraint

The dynamic pressure is given by:

$$\vec{q} = \frac{1}{2} \vec{\rho} (\vec{V}_x^2 + \vec{V}_y^2) \tag{F.7}$$

$$\begin{aligned}
 \frac{\partial \tilde{C}_q}{\partial R_x} &= \frac{\partial \tilde{q}}{\partial R_x} = \frac{1}{2} \langle (\vec{V}_x^2 + \vec{V}_y^2) \rangle \frac{\partial \tilde{\rho}}{\partial R_x} & \frac{\partial \tilde{C}_q}{\partial \tilde{m}} &= \frac{\partial \tilde{q}}{\partial \tilde{m}} = [0] \\
 \frac{\partial \tilde{C}_q}{\partial R_y} &= \frac{\partial \tilde{q}}{\partial R_y} = \frac{1}{2} \langle (\vec{V}_x^2 + \vec{V}_y^2) \rangle \frac{\partial \tilde{\rho}}{\partial R_y} & \frac{\partial \tilde{C}_q}{\partial T_x} &= \frac{\partial \tilde{q}}{\partial T_x} = [0] \\
 \frac{\partial \tilde{C}_q}{\partial V_x} &= \frac{\partial \tilde{q}}{\partial V_x} = \langle \tilde{\rho} \vec{V}_x \rangle & \frac{\partial \tilde{C}_q}{\partial T_y} &= \frac{\partial \tilde{q}}{\partial T_y} = [0] \\
 \frac{\partial \tilde{C}_q}{\partial V_y} &= \frac{\partial \tilde{q}}{\partial V_y} = \langle \tilde{\rho} \vec{V}_y \rangle & \frac{\partial \tilde{C}_q}{\partial \tau_f} &= \frac{\partial \tilde{q}}{\partial \tau_f} = [0]
 \end{aligned} \tag{F.8}$$

Partial Derivatives of Sensed Acceleration Constraint

The sensed acceleration is defined as:

$$\begin{aligned}
 \vec{a}_{sensed_x} &= \frac{\vec{T}_x + \vec{D}_x}{\tilde{m}} \\
 \vec{a}_{sensed_y} &= \frac{\vec{T}_y + \vec{D}_y}{\tilde{m}} \\
 (\vec{a}_{sensed})_{mag}^2 &= \vec{a}_{sensed_x}^2 + \vec{a}_{sensed_y}^2
 \end{aligned} \tag{F.9}$$

$$\begin{aligned}
\frac{\partial \tilde{C}_a}{\partial \vec{R}_x} &= 2 \left\langle \frac{1}{\tilde{m}} \right\rangle \left[\langle \vec{a}_{sensed_x} \rangle \frac{\partial \vec{D}_x}{\partial \vec{R}_x} + \langle \vec{a}_{sensed_y} \rangle \frac{\partial \vec{D}_y}{\partial \vec{R}_x} \right] \\
\frac{\partial \tilde{C}_a}{\partial \vec{R}_y} &= 2 \left\langle \frac{1}{\tilde{m}} \right\rangle \left[\langle \vec{a}_{sensed_x} \rangle \frac{\partial \vec{D}_x}{\partial \vec{R}_y} + \langle \vec{a}_{sensed_y} \rangle \frac{\partial \vec{D}_y}{\partial \vec{R}_y} \right] \\
\frac{\partial \tilde{C}_a}{\partial \vec{V}_x} &= 2 \left\langle \frac{1}{\tilde{m}} \right\rangle \left[\langle \vec{a}_{sensed_x} \rangle \frac{\partial \vec{D}_x}{\partial \vec{V}_x} + \langle \vec{a}_{sensed_y} \rangle \frac{\partial \vec{D}_y}{\partial \vec{V}_x} \right] \\
\frac{\partial \tilde{C}_a}{\partial \vec{V}_y} &= 2 \left\langle \frac{1}{\tilde{m}} \right\rangle \left[\langle \vec{a}_{sensed_x} \rangle \frac{\partial \vec{D}_x}{\partial \vec{V}_y} + \langle \vec{a}_{sensed_y} \rangle \frac{\partial \vec{D}_y}{\partial \vec{V}_y} \right] \\
\frac{\partial \tilde{C}_a}{\partial \vec{m}} &= -2 \left\langle \frac{\vec{a}_{sensed_x}^2 + \vec{a}_{sensed_y}^2}{\tilde{m}} \right\rangle \\
\frac{\partial \tilde{C}_a}{\partial \vec{T}_x} &= 2 \left\langle \frac{\vec{a}_{sensed_x}}{\tilde{m}} \right\rangle \\
\frac{\partial \tilde{C}_a}{\partial \vec{T}_y} &= 2 \left\langle \frac{\vec{a}_{sensed_y}}{\tilde{m}} \right\rangle \\
\frac{\partial \tilde{C}_a}{\partial \tau_f} &= [0]
\end{aligned} \tag{F.10}$$

Partial Derivatives of Final Position Magnitude Constraint

$$\begin{aligned}
\frac{\partial C_{R_f}}{\partial \vec{R}_x} &= \begin{bmatrix} 0 & 0 & \cdots & 2R_{x_n LGL} \end{bmatrix} & \frac{\partial C_{R_f}}{\partial \vec{m}} &= [0] \\
\frac{\partial C_{R_f}}{\partial \vec{R}_y} &= \begin{bmatrix} 0 & 0 & \cdots & 2R_{y_n LGL} \end{bmatrix} & \frac{\partial C_{R_f}}{\partial \vec{T}_x} &= [0] \\
\frac{\partial C_{R_f}}{\partial \vec{V}_x} &= [0] & \frac{\partial C_{R_f}}{\partial \vec{T}_y} &= [0] \\
\frac{\partial C_{R_f}}{\partial \vec{V}_y} &= [0] & \frac{\partial C_{R_f}}{\partial \tau_f} &= [0]
\end{aligned} \tag{F.11}$$

Partial Derivatives of Final Velocity Magnitude Constraint

$$\begin{aligned}
\frac{\partial C_{V_f}}{\partial \vec{R}_x} &= [0] & \frac{\partial C_{V_f}}{\partial \vec{m}} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{R}_y} &= [0] & \frac{\partial C_{V_f}}{\partial \vec{T}_x} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{V}_x} &= \begin{bmatrix} 0 & 0 & \cdots & 2V_{x_n LGL} \end{bmatrix} & \frac{\partial C_{V_f}}{\partial \vec{T}_y} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{V}_y} &= \begin{bmatrix} 0 & 0 & \cdots & 2V_{y_n LGL} \end{bmatrix} & \frac{\partial C_{V_f}}{\partial \tau_f} &= [0]
\end{aligned} \tag{F.12}$$

Partial Derivatives of $\vec{R}_f \cdot \vec{V}_f$ Constraint

$$\begin{aligned}
\frac{\partial C_{\vec{R} \cdot \vec{V}}}{\partial \vec{R}_x} &= \begin{bmatrix} 0 & 0 & \cdots & V_{x_n LGL} \end{bmatrix} & \frac{\partial C_{\vec{R} \cdot \vec{V}}}{\partial \vec{m}} &= [0] \\
\frac{\partial C_{\vec{R} \cdot \vec{V}}}{\partial \vec{R}_y} &= \begin{bmatrix} 0 & 0 & \cdots & V_{y_n LGL} \end{bmatrix} & \frac{\partial C_{\vec{R} \cdot \vec{V}}}{\partial \vec{T}_x} &= [0] \\
\frac{\partial C_{\vec{R} \cdot \vec{V}}}{\partial \vec{V}_x} &= \begin{bmatrix} 0 & 0 & \cdots & R_{x_n LGL} \end{bmatrix} & \frac{\partial C_{\vec{R} \cdot \vec{V}}}{\partial \vec{T}_y} &= [0] \\
\frac{\partial C_{\vec{R} \cdot \vec{V}}}{\partial \vec{V}_y} &= \begin{bmatrix} 0 & 0 & \cdots & R_{y_n LGL} \end{bmatrix} & \frac{\partial C_{\vec{R} \cdot \vec{V}}}{\partial \tau_f} &= 0
\end{aligned} \tag{F.13}$$

Partial Derivatives of Drag

$$\begin{aligned}\vec{D}_x &= \frac{1}{2} \tilde{\rho} (\vec{V}_x^2 + \vec{V}_y^2) C_D A_{ref} \frac{-\vec{V}_x}{(\vec{V}_x^2 + \vec{V}_y^2)^{1/2}} = -\frac{1}{2} \tilde{\rho} \vec{V}_x (\vec{V}_x^2 + \vec{V}_y^2)^{1/2} C_D A_{ref} \\ \vec{D}_y &= \frac{1}{2} \tilde{\rho} (\vec{V}_x^2 + \vec{V}_y^2) C_D A_{ref} \frac{-\vec{V}_y}{(\vec{V}_x^2 + \vec{V}_y^2)^{1/2}} = -\frac{1}{2} \tilde{\rho} \vec{V}_y (\vec{V}_x^2 + \vec{V}_y^2)^{1/2} C_D A_{ref}\end{aligned}\quad (\text{F.14})$$

$$\begin{aligned}\frac{\partial \vec{D}_x}{\partial \vec{R}_x} &= -\frac{1}{2} \left\langle \vec{V}_x (\vec{V}_x^2 + \vec{V}_y^2)^{1/2} \right\rangle \frac{\partial \tilde{\rho}}{\partial \vec{R}_x} C_D A_{ref} & \frac{\partial \vec{D}_x}{\partial \vec{m}} &= [0] \\ \frac{\partial \vec{D}_x}{\partial \vec{R}_y} &= -\frac{1}{2} \left\langle \vec{V}_x (\vec{V}_x^2 + \vec{V}_y^2)^{1/2} \right\rangle \frac{\partial \tilde{\rho}}{\partial \vec{R}_y} C_D A_{ref} & \frac{\partial \vec{D}_x}{\partial \vec{T}_x} &= [0] \\ \frac{\partial \vec{D}_x}{\partial \vec{V}_x} &= -\frac{1}{2} \langle \tilde{\rho} \rangle \left[\left\langle (\vec{V}_x^2 + \vec{V}_y^2)^{1/2} \right\rangle + \left\langle \frac{\vec{V}_x^2}{(\vec{V}_x^2 + \vec{V}_y^2)^{1/2}} \right\rangle \right] C_D A_{ref} & \frac{\partial \vec{D}_x}{\partial \vec{T}_y} &= [0] \\ \frac{\partial \vec{D}_x}{\partial \vec{V}_y} &= -\frac{1}{2} \langle \tilde{\rho} \rangle \left\langle \frac{\vec{V}_x \vec{V}_y}{(\vec{V}_x^2 + \vec{V}_y^2)^{1/2}} \right\rangle C_D A_{ref} & \frac{\partial \vec{D}_x}{\partial \tau_f} &= [0]\end{aligned}\quad (\text{F.15})$$

$$\begin{aligned}\frac{\partial \vec{D}_y}{\partial \vec{R}_x} &= -\frac{1}{2} \left\langle \vec{V}_y (\vec{V}_x^2 + \vec{V}_y^2)^{1/2} \right\rangle \frac{\partial \tilde{\rho}}{\partial \vec{R}_x} C_D A_{ref} & \frac{\partial \vec{D}_y}{\partial \vec{m}} &= [0] \\ \frac{\partial \vec{D}_y}{\partial \vec{R}_y} &= -\frac{1}{2} \left\langle \vec{V}_y (\vec{V}_x^2 + \vec{V}_y^2)^{1/2} \right\rangle \frac{\partial \tilde{\rho}}{\partial \vec{R}_y} C_D A_{ref} & \frac{\partial \vec{D}_y}{\partial \vec{T}_x} &= [0] \\ \frac{\partial \vec{D}_y}{\partial \vec{V}_x} &= -\frac{1}{2} \langle \tilde{\rho} \rangle \left\langle \frac{\vec{V}_x \vec{V}_y}{(\vec{V}_x^2 + \vec{V}_y^2)^{1/2}} \right\rangle C_D A_{ref} & \frac{\partial \vec{D}_y}{\partial \vec{T}_y} &= [0] \\ \frac{\partial \vec{D}_y}{\partial \vec{V}_y} &= -\frac{1}{2} \langle \tilde{\rho} \rangle \left[\left\langle (\vec{V}_x^2 + \vec{V}_y^2)^{1/2} \right\rangle + \left\langle \frac{\vec{V}_y^2}{(\vec{V}_x^2 + \vec{V}_y^2)^{1/2}} \right\rangle \right] C_D A_{ref} & \frac{\partial \vec{D}_y}{\partial \tau_f} &= [0]\end{aligned}\quad (\text{F.16})$$

Partial Derivatives of Atmospheric Density

See Appendix C.1 for the exponential atmospheric density model used. Note that:

$$\vec{R} = \sqrt{\vec{R}_x^2 + \vec{R}_y^2} \quad (\text{F.17})$$

$$\frac{\partial \vec{R}}{\partial \vec{R}_x} = \left\langle \frac{\vec{R}_x}{\sqrt{\vec{R}_x^2 + \vec{R}_y^2}} \right\rangle \quad \frac{\partial \vec{R}}{\partial \vec{R}_y} = \left\langle \frac{\vec{R}_y}{\sqrt{\vec{R}_x^2 + \vec{R}_y^2}} \right\rangle \quad (\text{F.18})$$

Therefore,

$$\begin{aligned}\frac{\partial \tilde{\rho}}{\partial \vec{R}_x} &= \frac{\partial \tilde{\rho}}{\partial \vec{R}} \frac{\partial \vec{R}}{\partial \vec{R}_x} \\ \frac{\partial \tilde{\rho}}{\partial \vec{R}_y} &= \frac{\partial \tilde{\rho}}{\partial \vec{R}} \frac{\partial \vec{R}}{\partial \vec{R}_y}\end{aligned}\quad (\text{F.19})$$

Partial Derivatives of Gravity

The gravitational acceleration is:

$$\begin{aligned}\vec{g}_x &= \frac{\mu}{(\bar{R}_x^2 + \bar{R}_y^2)} \frac{-\bar{R}_x}{(\bar{R}_x^2 + \bar{R}_y^2)^{1/2}} = \frac{-\mu \bar{R}_x}{(\bar{R}_x^2 + \bar{R}_y^2)^{3/2}} \\ \vec{g}_y &= \frac{\mu}{(\bar{R}_x^2 + \bar{R}_y^2)} \frac{-\bar{R}_y}{(\bar{R}_x^2 + \bar{R}_y^2)^{1/2}} = \frac{-\mu \bar{R}_y}{(\bar{R}_x^2 + \bar{R}_y^2)^{3/2}}\end{aligned}\quad (\text{F.20})$$

$$\begin{aligned}\frac{\partial \vec{g}_x}{\partial \bar{R}_x} &= \left\langle \frac{-\mu}{(\bar{R}_x^2 + \bar{R}_y^2)^{3/2}} \right\rangle + \left\langle \frac{3\mu \bar{R}_x^2}{(\bar{R}_x^2 + \bar{R}_y^2)^{5/2}} \right\rangle \quad \frac{\partial \vec{g}_x}{\partial m} = [0] \\ \frac{\partial \vec{g}_x}{\partial \bar{R}_y} &= \left\langle \frac{3\mu \bar{R}_x \bar{R}_y}{(\bar{R}_x^2 + \bar{R}_y^2)^{5/2}} \right\rangle \quad \frac{\partial \vec{g}_x}{\partial T_x} = [0] \\ \frac{\partial \vec{g}_x}{\partial V_x} &= [0] \quad \frac{\partial \vec{g}_x}{\partial T_y} = [0] \\ \frac{\partial \vec{g}_x}{\partial V_y} &= [0] \quad \frac{\partial \vec{g}_x}{\partial r_f} = [0]\end{aligned}\quad (\text{F.21})$$

$$\begin{aligned}\frac{\partial \vec{g}_y}{\partial \bar{R}_x} &= \left\langle \frac{3\mu \bar{R}_x \bar{R}_y}{(\bar{R}_x^2 + \bar{R}_y^2)^{5/2}} \right\rangle \quad \frac{\partial \vec{g}_y}{\partial m} = [0] \\ \frac{\partial \vec{g}_y}{\partial \bar{R}_y} &= \left\langle \frac{-\mu}{(\bar{R}_x^2 + \bar{R}_y^2)^{3/2}} \right\rangle + \left\langle \frac{3\mu \bar{R}_y^2}{(\bar{R}_x^2 + \bar{R}_y^2)^{5/2}} \right\rangle \quad \frac{\partial \vec{g}_y}{\partial T_x} = [0] \\ \frac{\partial \vec{g}_y}{\partial V_x} &= [0] \quad \frac{\partial \vec{g}_y}{\partial T_y} = [0] \\ \frac{\partial \vec{g}_y}{\partial V_y} &= [0] \quad \frac{\partial \vec{g}_y}{\partial r_f} = [0]\end{aligned}\quad (\text{F.22})$$

F.2 Radial-Transverse Polar Coordinate System Jacobian

The Jacobian for the two dimensional, radial-transverse polar coordinate system from section 4.2 is defined below. The Jacobian matrix is:

$$C_{Jac} = \frac{d\vec{C}}{d\vec{x}_{opt}} = \begin{bmatrix} \frac{\partial \vec{C}_R}{\partial R} & \frac{\partial \vec{C}_R}{\partial \theta} & \frac{\partial \vec{C}_R}{\partial V_R} & \frac{\partial \vec{C}_R}{\partial V_\theta} & \frac{\partial \vec{C}_R}{\partial \bar{m}} & \frac{\partial \vec{C}_R}{\partial \bar{T}_R} & \frac{\partial \vec{C}_R}{\partial \bar{T}_\theta} & \frac{\partial \vec{C}_R}{\partial \tau_f} \\ \frac{\partial \vec{C}_\phi}{\partial R} & \frac{\partial \vec{C}_\phi}{\partial \theta} & \frac{\partial \vec{C}_\phi}{\partial V_R} & \frac{\partial \vec{C}_\phi}{\partial V_\theta} & \frac{\partial \vec{C}_\phi}{\partial \bar{m}} & \frac{\partial \vec{C}_\phi}{\partial \bar{T}_R} & \frac{\partial \vec{C}_\phi}{\partial \bar{T}_\theta} & \frac{\partial \vec{C}_\phi}{\partial \tau_f} \\ \frac{\partial \vec{C}_{V_R}}{\partial R} & \frac{\partial \vec{C}_{V_R}}{\partial \theta} & \frac{\partial \vec{C}_{V_R}}{\partial V_R} & \frac{\partial \vec{C}_{V_R}}{\partial V_\theta} & \frac{\partial \vec{C}_{V_R}}{\partial \bar{m}} & \frac{\partial \vec{C}_{V_R}}{\partial \bar{T}_R} & \frac{\partial \vec{C}_{V_R}}{\partial \bar{T}_\theta} & \frac{\partial \vec{C}_{V_R}}{\partial \tau_f} \\ \frac{\partial \vec{C}_{V_\theta}}{\partial R} & \frac{\partial \vec{C}_{V_\theta}}{\partial \theta} & \frac{\partial \vec{C}_{V_\theta}}{\partial V_R} & \frac{\partial \vec{C}_{V_\theta}}{\partial V_\theta} & \frac{\partial \vec{C}_{V_\theta}}{\partial \bar{m}} & \frac{\partial \vec{C}_{V_\theta}}{\partial \bar{T}_R} & \frac{\partial \vec{C}_{V_\theta}}{\partial \bar{T}_\theta} & \frac{\partial \vec{C}_{V_\theta}}{\partial \tau_f} \\ \frac{\partial \vec{C}_{\bar{m}}}{\partial R} & \frac{\partial \vec{C}_{\bar{m}}}{\partial \theta} & \frac{\partial \vec{C}_{\bar{m}}}{\partial V_R} & \frac{\partial \vec{C}_{\bar{m}}}{\partial V_\theta} & \frac{\partial \vec{C}_{\bar{m}}}{\partial \bar{m}} & \frac{\partial \vec{C}_{\bar{m}}}{\partial \bar{T}_R} & \frac{\partial \vec{C}_{\bar{m}}}{\partial \bar{T}_\theta} & \frac{\partial \vec{C}_{\bar{m}}}{\partial \tau_f} \\ \frac{\partial \vec{C}_T}{\partial R} & \frac{\partial \vec{C}_T}{\partial \theta} & \frac{\partial \vec{C}_T}{\partial V_R} & \frac{\partial \vec{C}_T}{\partial V_\theta} & \frac{\partial \vec{C}_T}{\partial \bar{m}} & \frac{\partial \vec{C}_T}{\partial \bar{T}_R} & \frac{\partial \vec{C}_T}{\partial \bar{T}_\theta} & \frac{\partial \vec{C}_T}{\partial \tau_f} \\ \frac{\partial \vec{C}_q}{\partial R} & \frac{\partial \vec{C}_q}{\partial \theta} & \frac{\partial \vec{C}_q}{\partial V_R} & \frac{\partial \vec{C}_q}{\partial V_\theta} & \frac{\partial \vec{C}_q}{\partial \bar{m}} & \frac{\partial \vec{C}_q}{\partial \bar{T}_R} & \frac{\partial \vec{C}_q}{\partial \bar{T}_\theta} & \frac{\partial \vec{C}_q}{\partial \tau_f} \\ \frac{\partial \vec{C}_a}{\partial R} & \frac{\partial \vec{C}_a}{\partial \theta} & \frac{\partial \vec{C}_a}{\partial V_R} & \frac{\partial \vec{C}_a}{\partial V_\theta} & \frac{\partial \vec{C}_a}{\partial \bar{m}} & \frac{\partial \vec{C}_a}{\partial \bar{T}_R} & \frac{\partial \vec{C}_a}{\partial \bar{T}_\theta} & \frac{\partial \vec{C}_a}{\partial \tau_f} \end{bmatrix}$$

Partial Derivatives of \dot{R} Constraint

$$\begin{aligned} \frac{\partial \vec{C}_R}{\partial R} &= D_{NN} & \frac{\partial \vec{C}_R}{\partial \bar{m}} &= [0] \\ \frac{\partial \vec{C}_R}{\partial \theta} &= [0] & \frac{\partial \vec{C}_R}{\partial \bar{T}_R} &= [0] \\ \frac{\partial \vec{C}_R}{\partial V_R} &= \langle -1 \rangle & \frac{\partial \vec{C}_R}{\partial \bar{T}_\theta} &= [0] \\ \frac{\partial \vec{C}_R}{\partial V_\theta} &= [0] & \frac{\partial \vec{C}_R}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{R} \right] \end{aligned} \tag{F.23}$$

Partial Derivatives of $\dot{\theta}$ Constraint

$$\begin{aligned} \frac{\partial \vec{C}_\phi}{\partial R} &= \left\langle \frac{\vec{V}_\theta}{\vec{R}^2} \right\rangle & \frac{\partial \vec{C}_\phi}{\partial \bar{m}} &= [0] \\ \frac{\partial \vec{C}_\phi}{\partial \theta} &= D_{NN} & \frac{\partial \vec{C}_\phi}{\partial \bar{T}_R} &= [0] \\ \frac{\partial \vec{C}_\phi}{\partial V_R} &= [0] & \frac{\partial \vec{C}_\phi}{\partial \bar{T}_\theta} &= [0] \\ \frac{\partial \vec{C}_\phi}{\partial V_\theta} &= \left\langle \frac{-1}{\vec{R}} \right\rangle & \frac{\partial \vec{C}_\phi}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{\theta} \right] \end{aligned} \tag{F.24}$$

Partial Derivatives of \dot{V}_R Constraint

$$\begin{aligned}
\frac{\partial \vec{C}_{\dot{V}_R}}{\partial \vec{R}} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_R}{\partial \vec{R}} + \frac{\partial \vec{g}}{\partial \vec{R}} + \frac{\vec{V}_\theta^2}{\vec{R}^2} & \frac{\partial \vec{C}_{\dot{V}_R}}{\partial \vec{m}} &= \left\langle \frac{\vec{T}_R + \vec{D}_R}{\vec{m}^2} \right\rangle \\
\frac{\partial \vec{C}_{\dot{V}_R}}{\partial \theta} &= [0] & \frac{\partial \vec{C}_{\dot{V}_R}}{\partial \vec{T}_R} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \\
\frac{\partial \vec{C}_{\dot{V}_R}}{\partial \vec{V}_R} &= D_{NN} - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_R}{\partial \vec{V}_R} & \frac{\partial \vec{C}_{\dot{V}_R}}{\partial \vec{T}_\theta} &= [0] \\
\frac{\partial \vec{C}_{\dot{V}_R}}{\partial \vec{V}_\theta} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_R}{\partial \vec{V}_\theta} - 2 \left\langle \frac{\vec{V}_\theta}{\vec{R}} \right\rangle & \frac{\partial \vec{C}_{\dot{V}_R}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{V}_R \right]
\end{aligned} \tag{F.25}$$

Partial Derivatives of \dot{V}_θ Constraint

$$\begin{aligned}
\frac{\partial \vec{C}_{\dot{V}_\theta}}{\partial \vec{R}} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_\theta}{\partial \vec{R}} - \frac{\vec{V}_R \vec{V}_\theta}{\vec{R}^2} & \frac{\partial \vec{C}_{\dot{V}_\theta}}{\partial \vec{m}} &= \left\langle \frac{\vec{T}_\theta + \vec{D}_\theta}{\vec{m}^2} \right\rangle \\
\frac{\partial \vec{C}_{\dot{V}_\theta}}{\partial \theta} &= [0] & \frac{\partial \vec{C}_{\dot{V}_\theta}}{\partial \vec{T}_R} &= [0] \\
\frac{\partial \vec{C}_{\dot{V}_\theta}}{\partial \vec{V}_R} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_\theta}{\partial \vec{V}_R} + \left\langle \frac{\vec{V}_\theta}{\vec{R}} \right\rangle & \frac{\partial \vec{C}_{\dot{V}_\theta}}{\partial \vec{T}_\theta} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \\
\frac{\partial \vec{C}_{\dot{V}_\theta}}{\partial \vec{V}_\theta} &= D_{NN} - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{D}_\theta}{\partial \vec{V}_\theta} + \left\langle \frac{\vec{V}_R}{\vec{R}} \right\rangle & \frac{\partial \vec{C}_{\dot{V}_\theta}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{V}_\theta \right]
\end{aligned} \tag{F.26}$$

Partial Derivatives of \dot{m} Constraint

$$\begin{aligned}
\frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{R}} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{m}} &= D_{NN} \\
\frac{\partial \vec{C}_{\dot{m}}}{\partial \theta} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{T}_R} &= \frac{1}{g_o I_{sp}} \left\langle \frac{\vec{T}_R}{\sqrt{\vec{T}_R^2 + \vec{T}_\theta^2}} \right\rangle \\
\frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{V}_R} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{T}_\theta} &= \frac{1}{g_o I_{sp}} \left\langle \frac{\vec{T}_\theta}{\sqrt{\vec{T}_R^2 + \vec{T}_\theta^2}} \right\rangle \\
\frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{V}_\theta} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{m} \right]
\end{aligned} \tag{F.27}$$

Partial Derivatives of Thrust Magnitude Constraint

$$\begin{aligned}
\frac{\partial \vec{C}_T}{\partial \vec{R}} &= [0] & \frac{\partial \vec{C}_T}{\partial \vec{m}} &= [0] \\
\frac{\partial \vec{C}_T}{\partial \theta} &= [0] & \frac{\partial \vec{C}_T}{\partial \vec{T}_R} &= \left\langle 2 \vec{T}_R \right\rangle \\
\frac{\partial \vec{C}_T}{\partial \vec{V}_R} &= [0] & \frac{\partial \vec{C}_T}{\partial \vec{T}_\theta} &= \left\langle 2 \vec{T}_\theta \right\rangle \\
\frac{\partial \vec{C}_T}{\partial \vec{V}_\theta} &= [0] & \frac{\partial \vec{C}_T}{\partial \tau_f} &= [0]
\end{aligned} \tag{F.28}$$

Partial Derivatives of Dynamic Pressure Constraint

The dynamic pressure is given by:

$$\vec{q} = \frac{1}{2} \vec{\rho} (\vec{V}_R^2 + \vec{V}_\theta^2) \tag{F.29}$$

$$\begin{aligned}
\frac{\partial \tilde{C}_q}{\partial R} &= \frac{\partial \tilde{q}}{\partial R} = \frac{1}{2} \left\langle (\vec{V}_R^2 + \vec{V}_\theta^2) \right\rangle \frac{\partial \tilde{\rho}}{\partial R} & \frac{\partial \tilde{C}_q}{\partial \tilde{m}} &= \frac{\partial \tilde{q}}{\partial \tilde{m}} = [0] \\
\frac{\partial \tilde{C}_q}{\partial \theta} &= \frac{\partial \tilde{q}}{\partial \theta} = [0] & \frac{\partial \tilde{C}_q}{\partial \tilde{T}_R} &= \frac{\partial \tilde{q}}{\partial \tilde{T}_R} = [0] \\
\frac{\partial \tilde{C}_q}{\partial \vec{V}_R} &= \frac{\partial \tilde{q}}{\partial \vec{V}_R} = \left\langle \tilde{\rho} \vec{V}_R \right\rangle & \frac{\partial \tilde{C}_q}{\partial \tilde{T}_\theta} &= \frac{\partial \tilde{q}}{\partial \tilde{T}_\theta} = [0] \\
\frac{\partial \tilde{C}_q}{\partial \vec{V}_\theta} &= \frac{\partial \tilde{q}}{\partial \vec{V}_\theta} = \left\langle \tilde{\rho} \vec{V}_\theta \right\rangle & \frac{\partial \tilde{C}_q}{\partial \tau_f} &= \frac{\partial \tilde{q}}{\partial \tau_f} = [0]
\end{aligned} \tag{F.30}$$

Partial Derivatives of Sensed Acceleration Constraint

The sensed acceleration is given by:

$$\begin{aligned}
\vec{a}_{sensed_R} &= \frac{\vec{T}_R + \vec{D}_R}{\tilde{m}} \\
\vec{a}_{sensed_\theta} &= \frac{\vec{T}_\theta + \vec{D}_\theta}{\tilde{m}} \\
(\vec{a}_{sensed})_{mag}^2 &= \vec{a}_{sensed_R}^2 + \vec{a}_{sensed_\theta}^2
\end{aligned} \tag{F.31}$$

$$\begin{aligned}
\frac{\partial \tilde{C}_a}{\partial R} &= 2 \left\langle \frac{1}{\tilde{m}} \right\rangle \left[\langle \vec{a}_{sensed_R} \rangle \frac{\partial \vec{D}_R}{\partial R} + \langle \vec{a}_{sensed_\theta} \rangle \frac{\partial \vec{D}_\theta}{\partial R} \right] & \frac{\partial \tilde{C}_a}{\partial \tilde{m}} &= -2 \left\langle \frac{\vec{a}_{sensed_R}^2 + \vec{a}_{sensed_\theta}^2}{\tilde{m}} \right\rangle \\
\frac{\partial \tilde{C}_a}{\partial \theta} &= [0] & \frac{\partial \tilde{C}_a}{\partial \tilde{T}_R} &= 2 \left\langle \frac{\vec{a}_{sensed_R}}{\tilde{m}} \right\rangle \\
\frac{\partial \tilde{C}_a}{\partial \vec{V}_R} &= 2 \left\langle \frac{1}{\tilde{m}} \right\rangle \left[\langle \vec{a}_{sensed_R} \rangle \frac{\partial \vec{D}_R}{\partial \vec{V}_R} + \langle \vec{a}_{sensed_\theta} \rangle \frac{\partial \vec{D}_\theta}{\partial \vec{V}_R} \right] & \frac{\partial \tilde{C}_a}{\partial \tilde{T}_\theta} &= 2 \left\langle \frac{\vec{a}_{sensed_\theta}}{\tilde{m}} \right\rangle \\
\frac{\partial \tilde{C}_a}{\partial \vec{V}_\theta} &= 2 \left\langle \frac{1}{\tilde{m}} \right\rangle \left[\langle \vec{a}_{sensed_R} \rangle \frac{\partial \vec{D}_R}{\partial \vec{V}_\theta} + \langle \vec{a}_{sensed_\theta} \rangle \frac{\partial \vec{D}_\theta}{\partial \vec{V}_\theta} \right] & \frac{\partial \tilde{C}_a}{\partial \tau_f} &= [0]
\end{aligned} \tag{F.32}$$

Partial Derivatives of Drag

The drag is defined in equation 4.36, repeated below:

$$\begin{aligned}
\vec{D}_R &= \frac{1}{2} \tilde{\rho} \left(\vec{V}_R^2 + \vec{V}_\theta^2 \right) C_D A_{ref} \frac{-\vec{V}_R}{(\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2}} = -\frac{1}{2} \tilde{\rho} \vec{V}_R \left(\vec{V}_R^2 + \vec{V}_\theta^2 \right)^{1/2} C_D A_{ref} \\
\vec{D}_\theta &= \frac{1}{2} \tilde{\rho} \left(\vec{V}_R^2 + \vec{V}_\theta^2 \right) C_D A_{ref} \frac{-\vec{V}_\theta}{(\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2}} = -\frac{1}{2} \tilde{\rho} \vec{V}_\theta \left(\vec{V}_R^2 + \vec{V}_\theta^2 \right)^{1/2} C_D A_{ref}
\end{aligned} \tag{F.33}$$

$$\begin{aligned}
\frac{\partial \vec{D}_R}{\partial R} &= -\frac{1}{2} \left\langle \vec{V}_R \left(\vec{V}_R^2 + \vec{V}_\theta^2 \right)^{1/2} \right\rangle \frac{\partial \tilde{\rho}}{\partial R} C_D A_{ref} & \frac{\partial \vec{D}_R}{\partial \tilde{m}} &= [0] \\
\frac{\partial \vec{D}_R}{\partial \theta} &= [0] & \frac{\partial \vec{D}_R}{\partial \tilde{T}_R} &= [0] \\
\frac{\partial \vec{D}_R}{\partial \vec{V}_R} &= -\frac{1}{2} \langle \tilde{\rho} \rangle \left[\left\langle \left(\vec{V}_R^2 + \vec{V}_\theta^2 \right)^{1/2} \right\rangle + \left\langle \frac{\vec{V}_R^2}{(\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2}} \right\rangle \right] C_D A_{ref} & \frac{\partial \vec{D}_R}{\partial \tilde{T}_\theta} &= [0] \\
\frac{\partial \vec{D}_R}{\partial \vec{V}_\theta} &= -\frac{1}{2} \langle \tilde{\rho} \rangle \left\langle \frac{\vec{V}_R \vec{V}_\theta}{(\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2}} \right\rangle C_D A_{ref} & \frac{\partial \vec{D}_R}{\partial \tau_f} &= [0]
\end{aligned} \tag{F.34}$$

$$\begin{aligned}
\frac{\partial \tilde{D}_\theta}{\partial R} &= -\frac{1}{2} \left\langle \vec{V}_\theta \left(\vec{V}_R^2 + \vec{V}_\theta^2 \right)^{1/2} \right\rangle \frac{\partial \tilde{\rho}}{\partial R} & \frac{\partial \tilde{D}_\theta}{\partial \tilde{m}} &= [0] \\
\frac{\partial \tilde{D}_\theta}{\partial \theta} &= [0] & \frac{\partial \tilde{D}_\theta}{\partial \tilde{T}_R} &= [0] \\
\frac{\partial \tilde{D}_\theta}{\partial \tilde{V}_R} &= -\frac{1}{2} \langle \tilde{\rho} \rangle \left\langle \frac{\vec{V}_R \vec{V}_\theta}{(\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2}} \right\rangle & \frac{\partial \tilde{D}_\theta}{\partial \tilde{T}_\theta} &= [0] \\
\frac{\partial \tilde{D}_\theta}{\partial \tilde{V}_\theta} &= -\frac{1}{2} \langle \tilde{\rho} \rangle \left[\left\langle \left(\vec{V}_R^2 + \vec{V}_\theta^2 \right)^{1/2} \right\rangle + \left\langle \frac{\vec{V}_\theta^2}{(\vec{V}_R^2 + \vec{V}_\theta^2)^{1/2}} \right\rangle \right] & \frac{\partial \tilde{D}_\theta}{\partial \tau_f} &= [0]
\end{aligned} \tag{F.35}$$

Partial Derivatives of Atmospheric Density

See Appendix C.1 for the exponential atmospheric density model used. The partial derivative given in the appendix is used for this model.

Partial Derivatives of Gravity

The magnitude of the gravitational acceleration is given by equation 4.38, repeated below:

$$\vec{g} = \frac{\mu}{\tilde{R}^2} \tag{F.36}$$

$$\frac{\partial \vec{g}}{\partial \tilde{R}} = \left\langle \frac{-2\mu}{\tilde{R}^3} \right\rangle \tag{F.37}$$

F.3 Normal-Tangential Polar Coordinate System Jacobian

The Jacobian for the two dimensional, normal-tangential polar coordinate system from section 4.3 is defined below. The Jacobian matrix is:

$$C_{Jac} = \frac{d\vec{C}}{d\vec{x}_{opt}} = \begin{bmatrix} \frac{\partial \vec{C}_R}{\partial R} & \frac{\partial \vec{C}_R}{\partial \theta} & \frac{\partial \vec{C}_R}{\partial V} & \frac{\partial \vec{C}_R}{\partial \bar{\gamma}} & \frac{\partial \vec{C}_R}{\partial \bar{m}} & \frac{\partial \vec{C}_R}{\partial \bar{T}} & \frac{\partial \vec{C}_R}{\partial \bar{\alpha}} & \frac{\partial \vec{C}_R}{\partial \tau_f} \\ \frac{\partial \vec{C}_\dot{\theta}}{\partial R} & \frac{\partial \vec{C}_\dot{\theta}}{\partial \theta} & \frac{\partial \vec{C}_\dot{\theta}}{\partial V} & \frac{\partial \vec{C}_\dot{\theta}}{\partial \bar{\gamma}} & \frac{\partial \vec{C}_\dot{\theta}}{\partial \bar{m}} & \frac{\partial \vec{C}_\dot{\theta}}{\partial \bar{T}} & \frac{\partial \vec{C}_\dot{\theta}}{\partial \bar{\alpha}} & \frac{\partial \vec{C}_\dot{\theta}}{\partial \tau_f} \\ \frac{\partial \vec{C}_V}{\partial R} & \frac{\partial \vec{C}_V}{\partial \theta} & \frac{\partial \vec{C}_V}{\partial V} & \frac{\partial \vec{C}_V}{\partial \bar{\gamma}} & \frac{\partial \vec{C}_V}{\partial \bar{m}} & \frac{\partial \vec{C}_V}{\partial \bar{T}} & \frac{\partial \vec{C}_V}{\partial \bar{\alpha}} & \frac{\partial \vec{C}_V}{\partial \tau_f} \\ \frac{\partial \vec{C}_\dot{\gamma}}{\partial R} & \frac{\partial \vec{C}_\dot{\gamma}}{\partial \theta} & \frac{\partial \vec{C}_\dot{\gamma}}{\partial V} & \frac{\partial \vec{C}_\dot{\gamma}}{\partial \bar{\gamma}} & \frac{\partial \vec{C}_\dot{\gamma}}{\partial \bar{m}} & \frac{\partial \vec{C}_\dot{\gamma}}{\partial \bar{T}} & \frac{\partial \vec{C}_\dot{\gamma}}{\partial \bar{\alpha}} & \frac{\partial \vec{C}_\dot{\gamma}}{\partial \tau_f} \\ \frac{\partial \vec{C}_\dot{m}}{\partial R} & \frac{\partial \vec{C}_\dot{m}}{\partial \theta} & \frac{\partial \vec{C}_\dot{m}}{\partial V} & \frac{\partial \vec{C}_\dot{m}}{\partial \bar{\gamma}} & \frac{\partial \vec{C}_\dot{m}}{\partial \bar{m}} & \frac{\partial \vec{C}_\dot{m}}{\partial \bar{T}} & \frac{\partial \vec{C}_\dot{m}}{\partial \bar{\alpha}} & \frac{\partial \vec{C}_\dot{m}}{\partial \tau_f} \\ \frac{\partial \vec{C}_q}{\partial R} & \frac{\partial \vec{C}_q}{\partial \theta} & \frac{\partial \vec{C}_q}{\partial V} & \frac{\partial \vec{C}_q}{\partial \bar{\gamma}} & \frac{\partial \vec{C}_q}{\partial \bar{m}} & \frac{\partial \vec{C}_q}{\partial \bar{T}} & \frac{\partial \vec{C}_q}{\partial \bar{\alpha}} & \frac{\partial \vec{C}_q}{\partial \tau_f} \\ \frac{\partial \vec{C}_a}{\partial R} & \frac{\partial \vec{C}_a}{\partial \theta} & \frac{\partial \vec{C}_a}{\partial V} & \frac{\partial \vec{C}_a}{\partial \bar{\gamma}} & \frac{\partial \vec{C}_a}{\partial \bar{m}} & \frac{\partial \vec{C}_a}{\partial \bar{T}} & \frac{\partial \vec{C}_a}{\partial \bar{\alpha}} & \frac{\partial \vec{C}_a}{\partial \tau_f} \end{bmatrix}$$

Partial Derivatives of \dot{R} Constraint

$$\begin{aligned} \frac{\partial \vec{C}_R}{\partial R} &= D_{NN} & \frac{\partial \vec{C}_R}{\partial \bar{m}} &= [0] \\ \frac{\partial \vec{C}_R}{\partial \theta} &= [0] & \frac{\partial \vec{C}_R}{\partial \bar{T}} &= [0] \\ \frac{\partial \vec{C}_R}{\partial V} &= \langle -\sin \bar{\gamma} \rangle & \frac{\partial \vec{C}_R}{\partial \bar{\alpha}} &= [0] \\ \frac{\partial \vec{C}_R}{\partial \bar{\gamma}} &= \langle -\vec{V} \cos \bar{\gamma} \rangle & \frac{\partial \vec{C}_R}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{R} \right] \end{aligned} \quad (F.38)$$

Partial Derivatives of $\dot{\theta}$ Constraint

$$\begin{aligned} \frac{\partial \vec{C}_\dot{\theta}}{\partial R} &= \left\langle \frac{\vec{V} \cos \bar{\gamma}}{R^2} \right\rangle & \frac{\partial \vec{C}_\dot{\theta}}{\partial \bar{m}} &= [0] \\ \frac{\partial \vec{C}_\dot{\theta}}{\partial \theta} &= D_{NN} & \frac{\partial \vec{C}_\dot{\theta}}{\partial \bar{T}} &= [0] \\ \frac{\partial \vec{C}_\dot{\theta}}{\partial V} &= \left\langle -\frac{\cos \bar{\gamma}}{R} \right\rangle & \frac{\partial \vec{C}_\dot{\theta}}{\partial \bar{\alpha}} &= [0] \\ \frac{\partial \vec{C}_\dot{\theta}}{\partial \bar{\gamma}} &= \left\langle \frac{\vec{V} \sin \bar{\gamma}}{R} \right\rangle & \frac{\partial \vec{C}_\dot{\theta}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \bar{\theta} \right] \end{aligned} \quad (F.39)$$

Partial Derivatives of \dot{V} Constraint

$$\begin{aligned}
 \frac{\partial \tilde{C}_V}{\partial R} &= \left\langle \frac{1}{\bar{m}} \right\rangle \frac{\partial \tilde{D}}{\partial R} + \langle \sin \tilde{\gamma} \rangle \frac{\partial \tilde{g}}{\partial R} & \frac{\partial \tilde{C}_V}{\partial \bar{m}} &= \left\langle \frac{\tilde{T} \cos \tilde{\alpha} - \tilde{D}}{\bar{m}^2} \right\rangle \\
 \frac{\partial \tilde{C}_V}{\partial \theta} &= [0] & \frac{\partial \tilde{C}_V}{\partial T} &= \left\langle \frac{-\cos \tilde{\alpha}}{\bar{m}} \right\rangle \\
 \frac{\partial \tilde{C}_V}{\partial V} &= D_{NN} + \left\langle \frac{1}{\bar{m}} \right\rangle \frac{\partial \tilde{D}}{\partial V} & \frac{\partial \tilde{C}_V}{\partial \tilde{\alpha}} &= \left\langle \frac{\tilde{T} \sin \tilde{\alpha}}{\bar{m}} \right\rangle \\
 \frac{\partial \tilde{C}_V}{\partial \tilde{\gamma}} &= \langle \tilde{g} \cos \tilde{\gamma} \rangle & \frac{\partial \tilde{C}_V}{\partial \tau_f} &= \left\langle \frac{-D_{NN}}{(\tau_f - \tau_o)} \tilde{V} \right\rangle
 \end{aligned} \tag{F.40}$$

Partial Derivatives of $\dot{\gamma}$ Constraint

$$\begin{aligned}
 \frac{\partial \tilde{C}_{\dot{\gamma}}}{\partial R} &= \left\langle \frac{\cos \tilde{\gamma}}{\tilde{V}} \right\rangle \frac{\partial \tilde{g}}{\partial R} + \left\langle \frac{\tilde{V} \cos \tilde{\gamma}}{\tilde{R}^2} \right\rangle & \frac{\partial \tilde{C}_{\dot{\gamma}}}{\partial \bar{m}} &= \left\langle \frac{\tilde{T} \sin \tilde{\alpha}}{\bar{m}^2 \tilde{V}} \right\rangle \\
 \frac{\partial \tilde{C}_{\dot{\gamma}}}{\partial \theta} &= [0] & \frac{\partial \tilde{C}_{\dot{\gamma}}}{\partial T} &= \left\langle \frac{-\sin \tilde{\alpha}}{\bar{m} \tilde{V}} \right\rangle \\
 \frac{\partial \tilde{C}_{\dot{\gamma}}}{\partial V} &= \left\langle \frac{\tilde{T} \sin \tilde{\alpha}}{\bar{m} \tilde{V}^2} - \frac{\tilde{g} \cos \tilde{\gamma}}{\tilde{V}^2} - \frac{\cos \tilde{\gamma}}{\tilde{R}} \right\rangle & \frac{\partial \tilde{C}_{\dot{\gamma}}}{\partial \tilde{\alpha}} &= \left\langle \frac{-\tilde{T} \cos \tilde{\alpha}}{\bar{m} \tilde{V}} \right\rangle \\
 \frac{\partial \tilde{C}_{\dot{\gamma}}}{\partial \tilde{\gamma}} &= D_{NN} - \left\langle \frac{\tilde{g} \sin \tilde{\gamma}}{\tilde{V}} - \frac{\tilde{V} \sin \tilde{\gamma}}{\tilde{R}} \right\rangle & \frac{\partial \tilde{C}_{\dot{\gamma}}}{\partial \bar{m}} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \tilde{\gamma} \right]
 \end{aligned} \tag{F.41}$$

Partial Derivatives of \dot{m} Constraint

$$\begin{aligned}
 \frac{\partial \tilde{C}_{\dot{m}}}{\partial R} &= [0] & \frac{\partial \tilde{C}_{\dot{m}}}{\partial \bar{m}} &= D_{NN} \\
 \frac{\partial \tilde{C}_{\dot{m}}}{\partial \theta} &= [0] & \frac{\partial \tilde{C}_{\dot{m}}}{\partial T} &= \left\langle \frac{1}{g_o I_{sp}} \right\rangle \\
 \frac{\partial \tilde{C}_{\dot{m}}}{\partial V} &= [0] & \frac{\partial \tilde{C}_{\dot{m}}}{\partial \tilde{\alpha}} &= [0] \\
 \frac{\partial \tilde{C}_{\dot{m}}}{\partial \tilde{\gamma}} &= [0] & \frac{\partial \tilde{C}_{\dot{m}}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \bar{m} \right]
 \end{aligned} \tag{F.42}$$

Partial Derivatives of Dynamic Pressure Constraint

The dynamic pressure is given by:

$$\tilde{q} = \frac{1}{2} \tilde{\rho} \tilde{V}^2 \tag{F.43}$$

$$\begin{aligned}
 \frac{\partial \tilde{C}_q}{\partial R} &= \frac{\partial \tilde{q}}{\partial R} = \left\langle \frac{1}{2} \tilde{V}^2 \right\rangle \frac{\partial \tilde{\rho}}{\partial R} & \frac{\partial \tilde{C}_q}{\partial \bar{m}} &= \frac{\partial \tilde{q}}{\partial \bar{m}} = [0] \\
 \frac{\partial \tilde{C}_q}{\partial \theta} &= \frac{\partial \tilde{q}}{\partial \theta} = [0] & \frac{\partial \tilde{C}_q}{\partial T} &= \frac{\partial \tilde{q}}{\partial T} = [0] \\
 \frac{\partial \tilde{C}_q}{\partial V} &= \frac{\partial \tilde{q}}{\partial V} = \left\langle \tilde{\rho} \tilde{V} \right\rangle & \frac{\partial \tilde{C}_q}{\partial \tilde{\alpha}} &= \frac{\partial \tilde{q}}{\partial \tilde{\alpha}} = [0] \\
 \frac{\partial \tilde{C}_q}{\partial \tilde{\gamma}} &= \frac{\partial \tilde{q}}{\partial \tilde{\gamma}} = [0] & \frac{\partial \tilde{C}_q}{\partial \tau_f} &= \frac{\partial \tilde{q}}{\partial \tau_f} = [0]
 \end{aligned} \tag{F.44}$$

Partial Derivatives of Sensed Acceleration Constraint

The sensed acceleration is given by:

$$\begin{aligned}\vec{a}_{\text{sensed}_V} &= \frac{\vec{T} \cos \vec{\alpha} - \vec{D}}{m} \\ \vec{a}_{\text{sensed}, \gamma} &= \frac{\vec{T} \sin \vec{\alpha}}{m} \\ (\vec{a}_{\text{sensed}})^2_{\text{mag}} &= \vec{a}_{\text{sensed}_V}^2 + \vec{a}_{\text{sensed}, \gamma}^2\end{aligned}\quad (\text{F.45})$$

$$\begin{aligned}\frac{\partial \vec{C}_a}{\partial R} &= 2 \left\langle -\frac{\vec{a}_{\text{sensed}_V}}{m} \right\rangle \frac{\partial \vec{D}}{\partial R} & \frac{\partial \vec{C}_a}{\partial m} &= -2 \left\langle \frac{\vec{a}_{\text{sensed}_V}^2 + \vec{a}_{\text{sensed}, \gamma}^2}{m} \right\rangle \\ \frac{\partial \vec{C}_a}{\partial \theta} &= [0] & \frac{\partial \vec{C}_a}{\partial T} &= 2 \left\langle \vec{a}_{\text{sensed}_V} \frac{\cos \vec{\alpha}}{m} + \vec{a}_{\text{sensed}, \gamma} \frac{\sin \vec{\alpha}}{m} \right\rangle \\ \frac{\partial \vec{C}_a}{\partial V} &= 2 \left\langle -\frac{\vec{a}_{\text{sensed}_V}}{m} \right\rangle \frac{\partial \vec{D}}{\partial V} & \frac{\partial \vec{C}_a}{\partial \vec{\alpha}} &= 2 \left\langle -\vec{a}_{\text{sensed}_V} \frac{\vec{T} \sin \vec{\alpha}}{m} + \vec{a}_{\text{sensed}, \gamma} \frac{\vec{T} \cos \vec{\alpha}}{m} \right\rangle \\ \frac{\partial \vec{C}_a}{\partial \gamma} &= [0] & \frac{\partial \vec{C}_a}{\partial \tau_f} &= [0]\end{aligned}\quad (\text{F.46})$$

Partial Derivatives of Drag

The magnitude of the drag is given by:

$$\vec{D} = \vec{q} C_D A_{\text{ref}} \quad (\text{F.47})$$

$$\begin{aligned}\frac{\partial \vec{D}}{\partial R} &= \frac{\partial \vec{q}}{\partial R} C_D A_{\text{ref}} & \frac{\partial \vec{D}}{\partial m} &= [0] \\ \frac{\partial \vec{D}}{\partial \theta} &= [0] & \frac{\partial \vec{D}}{\partial T} &= [0] \\ \frac{\partial \vec{D}}{\partial V} &= \frac{\partial \vec{q}}{\partial V} C_D A_{\text{ref}} & \frac{\partial \vec{D}}{\partial \vec{\alpha}} &= [0] \\ \frac{\partial \vec{D}}{\partial \gamma} &= [0] & \frac{\partial \vec{D}}{\partial \tau_f} &= [0]\end{aligned}\quad (\text{F.48})$$

The partial derivatives of the dynamic pressure (\vec{q}) are defined in equations F.44.

Partial Derivatives of Atmospheric Density

See Appendix C.1 for the exponential atmospheric density model used. The partial derivative given in the appendix is used for this model.

Partial Derivatives of Gravity

The magnitude of the gravitational acceleration is:

$$\vec{g} = \frac{\mu}{\vec{R}^2} \quad (\text{F.49})$$

$$\frac{\partial \vec{g}}{\partial \vec{R}} = \left\langle \frac{-2\mu}{\vec{R}^3} \right\rangle \quad (\text{F.50})$$

F.4 D^2 Cartesian Coordinate System Jacobian

The Jacobian for the two dimensional, D^2 cartesian coordinate system from section 5.2.1 is defined below. The Jacobian matrix is:

$$C_{Jac} = \frac{d\vec{C}}{d\vec{x}_{opt}} = \begin{bmatrix} \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{R}_x} & \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{R}_y} & \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{m}} & \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{T}_x} & \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{T}_y} & \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \tau_f} \\ \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{R}_x} & \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{R}_y} & \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{m}} & \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{T}_x} & \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{T}_y} & \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \tau_f} \\ \frac{\partial \vec{C}_{\vec{m}}}{\partial \vec{R}_x} & \frac{\partial \vec{C}_{\vec{m}}}{\partial \vec{R}_y} & \frac{\partial \vec{C}_{\vec{m}}}{\partial \vec{m}} & \frac{\partial \vec{C}_{\vec{m}}}{\partial \vec{T}_x} & \frac{\partial \vec{C}_{\vec{m}}}{\partial \vec{T}_y} & \frac{\partial \vec{C}_{\vec{m}}}{\partial \tau_f} \\ \frac{\partial \vec{C}_T}{\partial \vec{R}_x} & \frac{\partial \vec{C}_T}{\partial \vec{R}_y} & \frac{\partial \vec{C}_T}{\partial \vec{m}} & \frac{\partial \vec{C}_T}{\partial \vec{T}_x} & \frac{\partial \vec{C}_T}{\partial \vec{T}_y} & \frac{\partial \vec{C}_T}{\partial \tau_f} \\ \frac{\partial \vec{C}_q}{\partial \vec{R}_x} & \frac{\partial \vec{C}_q}{\partial \vec{R}_y} & \frac{\partial \vec{C}_q}{\partial \vec{m}} & \frac{\partial \vec{C}_q}{\partial \vec{T}_x} & \frac{\partial \vec{C}_q}{\partial \vec{T}_y} & \frac{\partial \vec{C}_q}{\partial \tau_f} \\ \frac{\partial \vec{C}_a}{\partial \vec{R}_x} & \frac{\partial \vec{C}_a}{\partial \vec{R}_y} & \frac{\partial \vec{C}_a}{\partial \vec{m}} & \frac{\partial \vec{C}_a}{\partial \vec{T}_x} & \frac{\partial \vec{C}_a}{\partial \vec{T}_y} & \frac{\partial \vec{C}_a}{\partial \tau_f} \\ \frac{\partial C_{V_xo}}{\partial \vec{R}_x} & \frac{\partial C_{V_xo}}{\partial \vec{R}_y} & \frac{\partial C_{V_xo}}{\partial \vec{m}} & \frac{\partial C_{V_xo}}{\partial \vec{T}_x} & \frac{\partial C_{V_xo}}{\partial \vec{T}_y} & \frac{\partial C_{V_xo}}{\partial \tau_f} \\ \frac{\partial C_{V_yo}}{\partial \vec{R}_x} & \frac{\partial C_{V_yo}}{\partial \vec{R}_y} & \frac{\partial C_{V_yo}}{\partial \vec{m}} & \frac{\partial C_{V_yo}}{\partial \vec{T}_x} & \frac{\partial C_{V_yo}}{\partial \vec{T}_y} & \frac{\partial C_{V_yo}}{\partial \tau_f} \\ \frac{\partial C_{R_f}}{\partial \vec{R}_x} & \frac{\partial C_{R_f}}{\partial \vec{R}_y} & \frac{\partial C_{R_f}}{\partial \vec{m}} & \frac{\partial C_{R_f}}{\partial \vec{T}_x} & \frac{\partial C_{R_f}}{\partial \vec{T}_y} & \frac{\partial C_{R_f}}{\partial \tau_f} \\ \frac{\partial C_{V_f}}{\partial \vec{R}_x} & \frac{\partial C_{V_f}}{\partial \vec{R}_y} & \frac{\partial C_{V_f}}{\partial \vec{m}} & \frac{\partial C_{V_f}}{\partial \vec{T}_x} & \frac{\partial C_{V_f}}{\partial \vec{T}_y} & \frac{\partial C_{V_f}}{\partial \tau_f} \\ \frac{\partial C_{R,V}}{\partial \vec{R}_x} & \frac{\partial C_{R,V}}{\partial \vec{R}_y} & \frac{\partial C_{R,V}}{\partial \vec{m}} & \frac{\partial C_{R,V}}{\partial \vec{T}_x} & \frac{\partial C_{R,V}}{\partial \vec{T}_y} & \frac{\partial C_{R,V}}{\partial \tau_f} \end{bmatrix}$$

Partial Derivatives of Velocities and Velocity Magnitude

The partial derivatives of the velocity with respect to R_x , R_y and τ_f are necessary.

$$\begin{aligned}\frac{\partial \vec{V}_x}{\partial R_x} &= D_{NN} & \frac{\partial \vec{V}_x}{\partial \tau_f} &= \frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{R}_x \\ \frac{\partial \vec{V}_y}{\partial R_y} &= D_{NN} & \frac{\partial \vec{V}_y}{\partial \tau_f} &= \frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{R}_y\end{aligned}\quad (\text{F.51})$$

The velocity magnitude is:

$$\vec{V}_{mag} = \sqrt{\vec{V}_x^2 + \vec{V}_y^2} \quad (\text{F.52})$$

The partial derivatives of the velocity magnitude are:

$$\begin{aligned}\frac{\partial \vec{V}_{mag}}{\partial R_x} &= \left\langle \frac{\vec{V}_x}{\vec{V}_{mag}} \right\rangle \frac{\partial \vec{V}_x}{\partial R_x} \\ \frac{\partial \vec{V}_{mag}}{\partial R_y} &= \left\langle \frac{\vec{V}_y}{\vec{V}_{mag}} \right\rangle \frac{\partial \vec{V}_y}{\partial R_y} \\ \frac{\partial \vec{V}_{mag}}{\partial \tau_f} &= \frac{1}{\vec{V}_{mag}} \left[\vec{V}_x \frac{\partial \vec{V}_x}{\partial \tau_f} + \vec{V}_y \frac{\partial \vec{V}_y}{\partial \tau_f} \right]\end{aligned}\quad (\text{F.53})$$

Partial Derivatives of \ddot{R}_x Constraint

$$\begin{aligned}\frac{\partial \vec{C}_{\ddot{R}_x}}{\partial R_x} &= D_{NN}^2 - \left\langle \frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{D}_x}{\partial R_x} - \frac{\partial \vec{g}_x}{\partial R_x} & \frac{\partial \vec{C}_{\ddot{R}_x}}{\partial \vec{T}_x} &= - \left\langle \frac{1}{\bar{m}} \right\rangle \\ \frac{\partial \vec{C}_{\ddot{R}_x}}{\partial R_y} &= - \left\langle \frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{D}_x}{\partial R_y} - \frac{\partial \vec{g}_x}{\partial R_y} & \frac{\partial \vec{C}_{\ddot{R}_x}}{\partial \vec{T}_y} &= [0] \\ \frac{\partial \vec{C}_{\ddot{R}_x}}{\partial \bar{m}} &= \left\langle \frac{\vec{T}_x + \vec{D}_x}{\bar{m}^2} \right\rangle & \frac{\partial \vec{C}_{\ddot{R}_x}}{\partial \tau_f} &= \left[\frac{-2D_{NN}^2}{(\tau_f - \tau_o)} \vec{R}_x - \frac{1}{\bar{m}} \frac{\partial \vec{D}_x}{\partial \tau_f} \right]\end{aligned}\quad (\text{F.54})$$

Partial Derivatives of \ddot{R}_y Constraint

$$\begin{aligned}\frac{\partial \vec{C}_{\ddot{R}_y}}{\partial R_x} &= - \left\langle \frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{D}_y}{\partial R_x} - \frac{\partial \vec{g}_y}{\partial R_x} & \frac{\partial \vec{C}_{\ddot{R}_y}}{\partial \vec{T}_x} &= [0] \\ \frac{\partial \vec{C}_{\ddot{R}_y}}{\partial R_y} &= D_{NN}^2 - \left\langle \frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{D}_y}{\partial R_y} - \frac{\partial \vec{g}_y}{\partial R_y} & \frac{\partial \vec{C}_{\ddot{R}_y}}{\partial \vec{T}_y} &= - \left\langle \frac{1}{\bar{m}} \right\rangle \\ \frac{\partial \vec{C}_{\ddot{R}_y}}{\partial \bar{m}} &= \left\langle \frac{\vec{T}_y + \vec{D}_y}{\bar{m}^2} \right\rangle & \frac{\partial \vec{C}_{\ddot{R}_y}}{\partial \tau_f} &= \left[\frac{-2D_{NN}^2}{(\tau_f - \tau_o)} \vec{R}_y - \frac{1}{\bar{m}} \frac{\partial \vec{D}_y}{\partial \tau_f} \right]\end{aligned}\quad (\text{F.55})$$

Partial Derivatives of \dot{m} Constraint

$$\begin{aligned}\frac{\partial \tilde{C}_{\dot{m}}}{\partial R_x} &= [0] & \frac{\partial \tilde{C}_{\dot{m}}}{\partial T_x} &= \frac{1}{g_0 I_{sp}} \left\langle \frac{\vec{T}_x}{\sqrt{\vec{T}_x^2 + \vec{T}_y^2}} \right\rangle \\ \frac{\partial \tilde{C}_{\dot{m}}}{\partial R_y} &= [0] & \frac{\partial \tilde{C}_{\dot{m}}}{\partial T_y} &= \frac{1}{g_0 I_{sp}} \left\langle \frac{\vec{T}_y}{\sqrt{\vec{T}_x^2 + \vec{T}_y^2}} \right\rangle \\ \frac{\partial \tilde{C}_{\dot{m}}}{\partial \dot{m}} &= D_{NN} & \frac{\partial \tilde{C}_{\dot{m}}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \dot{m} \right]\end{aligned}\quad (\text{F.56})$$

Partial Derivatives of Thrust Magnitude Constraint

$$\begin{aligned}\frac{\partial \tilde{C}_T}{\partial R_x} &= [0] & \frac{\partial \tilde{C}_T}{\partial T_x} &= \left\langle 2\vec{T}_x \right\rangle \\ \frac{\partial \tilde{C}_T}{\partial R_y} &= [0] & \frac{\partial \tilde{C}_T}{\partial T_y} &= \left\langle 2\vec{T}_y \right\rangle \\ \frac{\partial \tilde{C}_T}{\partial \dot{m}} &= [0] & \frac{\partial \tilde{C}_T}{\partial \tau_f} &= [0]\end{aligned}\quad (\text{F.57})$$

Partial Derivatives of Dynamic Pressure Constraint

The dynamic pressure is given by:

$$\tilde{q} = \frac{1}{2} \tilde{\rho} \vec{V}_{mag}^2 \quad (\text{F.58})$$

The partial derivatives are:

$$\begin{aligned}\frac{\partial \tilde{C}_q}{\partial R_x} &= \frac{\partial \tilde{q}}{\partial R_x} = \left\langle \frac{1}{2} \vec{V}_{mag}^2 \right\rangle \frac{\partial \tilde{\rho}}{\partial R_x} + \left\langle \tilde{\rho} \vec{V}_{mag} \right\rangle \frac{\partial \vec{V}_{mag}}{\partial R_x} \\ \frac{\partial \tilde{C}_q}{\partial R_y} &= \frac{\partial \tilde{q}}{\partial R_y} = \left\langle \frac{1}{2} \vec{V}_{mag}^2 \right\rangle \frac{\partial \tilde{\rho}}{\partial R_y} + \left\langle \tilde{\rho} \vec{V}_{mag} \right\rangle \frac{\partial \vec{V}_{mag}}{\partial R_y} \\ \frac{\partial \tilde{C}_q}{\partial \tau_f} &= \frac{\partial \tilde{q}}{\partial \tau_f} = \left[\tilde{\rho} \vec{V}_{mag} \frac{\partial \vec{V}_{mag}}{\partial \tau_f} \right]\end{aligned}\quad (\text{F.59})$$

All other derivatives are zero.

Partial Derivatives of Sensed Acceleration Constraint

The sensed acceleration is defined as:

$$\begin{aligned}\vec{a}_{sensed_x} &= \frac{\vec{T}_x + \vec{D}_x}{\dot{m}} \\ \vec{a}_{sensed_y} &= \frac{\vec{T}_y + \vec{D}_y}{\dot{m}} \\ (\vec{a}_{sensed})_{mag}^2 &= \vec{a}_{sensed_x}^2 + \vec{a}_{sensed_y}^2\end{aligned}\quad (\text{F.60})$$

$$\begin{aligned}\frac{\partial \tilde{C}_a}{\partial \vec{R}_x} &= 2 \left\langle \frac{1}{\vec{m}} \right\rangle \left[\langle \vec{a}_{sensed_x} \rangle \frac{\partial \vec{D}_x}{\partial \vec{R}_x} + \langle \vec{a}_{sensed_y} \rangle \frac{\partial \vec{D}_y}{\partial \vec{R}_x} \right] & \frac{\partial \tilde{C}_a}{\partial T_x} &= 2 \left\langle \frac{\vec{a}_{sensed_x}}{\vec{m}} \right\rangle \\ \frac{\partial \tilde{C}_a}{\partial \vec{R}_y} &= 2 \left\langle \frac{1}{\vec{m}} \right\rangle \left[\langle \vec{a}_{sensed_x} \rangle \frac{\partial \vec{D}_x}{\partial \vec{R}_y} + \langle \vec{a}_{sensed_y} \rangle \frac{\partial \vec{D}_y}{\partial \vec{R}_y} \right] & \frac{\partial \tilde{C}_a}{\partial T_y} &= 2 \left\langle \frac{\vec{a}_{sensed_y}}{\vec{m}} \right\rangle \\ \frac{\partial \tilde{C}_a}{\partial \vec{m}} &= -2 \left\langle \frac{\vec{a}_{sensed_x}^2 + \vec{a}_{sensed_y}^2}{\vec{m}} \right\rangle & \frac{\partial \tilde{C}_a}{\partial \tau_f} &= [0]\end{aligned}\quad (\text{F.61})$$

Partial Derivatives of Initial Velocity Constraints

$$\begin{aligned}\frac{\partial C_{V_{x_0}}}{\partial \vec{R}_x} &= [d_{NN_1}] & \frac{\partial C_{V_{x_0}}}{\partial \vec{T}_x} &= [0] \\ \frac{\partial C_{V_{x_0}}}{\partial \vec{R}_y} &= [0] & \frac{\partial C_{V_{x_0}}}{\partial \vec{T}_y} &= [0] \\ \frac{\partial C_{V_{x_0}}}{\partial \vec{m}} &= [0] & \frac{\partial C_{V_{x_0}}}{\partial \tau_f} &= \frac{-d_{NN_1}}{(\tau_f - \tau_o)} \vec{R}_x\end{aligned}\quad (\text{F.62})$$

$$\begin{aligned}\frac{\partial C_{V_{y_0}}}{\partial \vec{R}_x} &= [0] & \frac{\partial C_{V_{y_0}}}{\partial \vec{T}_x} &= [0] \\ \frac{\partial C_{V_{y_0}}}{\partial \vec{R}_y} &= [d_{NN_1}] & \frac{\partial C_{V_{y_0}}}{\partial \vec{T}_y} &= [0] \\ \frac{\partial C_{V_{y_0}}}{\partial \vec{m}} &= [0] & \frac{\partial C_{V_{y_0}}}{\partial \tau_f} &= \frac{-d_{NN_1}}{(\tau_f - \tau_o)} \vec{R}_y\end{aligned}\quad (\text{F.63})$$

where d_{NN_1} is the first row of the D_{NN} matrix. Note that it is equivalent to a row vector.

Partial Derivatives of Final Position Magnitude Constraint

$$\begin{aligned}\frac{\partial C_{R_{x_f}}}{\partial \vec{R}_x} &= \begin{bmatrix} 0 & 0 & \dots & 2R_{x_n LGL} \end{bmatrix} & \frac{\partial C_{R_{x_f}}}{\partial \vec{m}} &= [0] \\ \frac{\partial C_{R_{x_f}}}{\partial \vec{R}_y} &= \begin{bmatrix} 0 & 0 & \dots & 2R_{y_n LGL} \end{bmatrix} & \frac{\partial C_{R_{x_f}}}{\partial \vec{T}_x} &= [0] \\ \frac{\partial C_{R_{x_f}}}{\partial \vec{V}_x} &= [0] & \frac{\partial C_{R_{x_f}}}{\partial \vec{T}_y} &= [0] \\ \frac{\partial C_{R_{x_f}}}{\partial \vec{V}_y} &= [0] & \frac{\partial C_{R_{x_f}}}{\partial \tau_f} &= [0]\end{aligned}\quad (\text{F.64})$$

Partial Derivatives of Final Velocity Magnitude Constraint

$$\begin{aligned}\frac{\partial C_{V_{x_f}}}{\partial \vec{R}_x} &= [2V_{x_f} d_{NN_f}] & \frac{\partial C_{V_{x_f}}}{\partial \vec{T}_x} &= [0] \\ \frac{\partial C_{V_{x_f}}}{\partial \vec{R}_y} &= [2V_{y_f} d_{NN_f}] & \frac{\partial C_{V_{x_f}}}{\partial \vec{T}_y} &= [0] \\ \frac{\partial C_{V_{x_f}}}{\partial \vec{m}} &= [0] & \frac{\partial C_{V_{x_f}}}{\partial \tau_f} &= \frac{-2}{(\tau_f - \tau_o)} [V_{x_f} d_{NN_f} \vec{R}_x + V_{y_f} d_{NN_f} \vec{R}_y]\end{aligned}\quad (\text{F.65})$$

where d_{NN_f} is the last row of the D_{NN} matrix. Note that it is equivalent to a row vector.

Partial Derivatives of $\vec{R}_f \cdot \vec{V}_f$ Constraint

$$\begin{aligned}\frac{\partial C_{\vec{R}_f \cdot \vec{V}_f}}{\partial \vec{R}_x} &= \begin{bmatrix} R_{x_f} d_{NN_{f,1}} & R_{x_f} d_{NN_{f,2}} & \cdots & R_{x_f} d_{NN_{f,n-1}} & V_{x_f} + R_{x_f} d_{NN_{f,n}} \end{bmatrix} \\ \frac{\partial C_{\vec{R}_f \cdot \vec{V}_f}}{\partial \vec{R}_y} &= \begin{bmatrix} R_{y_f} d_{NN_{f,1}} & R_{y_f} d_{NN_{f,2}} & \cdots & R_{y_f} d_{NN_{f,n-1}} & V_{y_f} + R_{y_f} d_{NN_{f,n}} \end{bmatrix} \\ \frac{\partial C_{\vec{R}_f \cdot \vec{V}_f}}{\partial \vec{m}} &= [0] \\ \frac{\partial C_{\vec{R}_f \cdot \vec{V}_f}}{\partial \vec{T}_x} &= [0] \\ \frac{\partial C_{\vec{R}_f \cdot \vec{V}_f}}{\partial \vec{T}_y} &= [0] \\ \frac{\partial C_{\vec{R}_f \cdot \vec{V}_f}}{\partial \tau_f} &= -\frac{1}{(\tau_f - \tau_o)} [R_{x_f} d_{NN_f} \vec{R}_x + R_{y_f} d_{NN_f} \vec{R}_y]\end{aligned}\tag{F.66}$$

where $d_{NN_{f,i}}$ is the i th element of the last row of the D_{NN} matrix.

Partial Derivatives of Drag

$$\begin{aligned}\vec{D}_x &= \frac{1}{2} \vec{\rho} \vec{V}_{mag}^2 C_D A_{ref} \frac{-\vec{V}_x}{\vec{V}_{mag}} = -\frac{1}{2} \vec{\rho} \vec{V}_x \vec{V}_{mag} C_D A_{ref} \\ \vec{D}_y &= \frac{1}{2} \vec{\rho} \vec{V}_{mag}^2 C_D A_{ref} \frac{-\vec{V}_y}{\vec{V}_{mag}} = -\frac{1}{2} \vec{\rho} \vec{V}_y \vec{V}_{mag} C_D A_{ref}\end{aligned}\tag{F.67}$$

$$\begin{aligned}\frac{\partial \vec{D}_x}{\partial \vec{R}_x} &= -\frac{1}{2} C_D A_{ref} \left[\langle \vec{V}_x \vec{V}_{mag} \rangle \frac{\partial \vec{\rho}}{\partial \vec{R}_x} + \langle \vec{\rho} \vec{V}_{mag} \rangle \frac{\partial \vec{V}_x}{\partial \vec{R}_x} + \langle \vec{\rho} \vec{V}_x \rangle \frac{\partial \vec{V}_{mag}}{\partial \vec{R}_x} \right] \\ \frac{\partial \vec{D}_x}{\partial \vec{R}_y} &= -\frac{1}{2} C_D A_{ref} \left[\langle \vec{V}_x \vec{V}_{mag} \rangle \frac{\partial \vec{\rho}}{\partial \vec{R}_y} + \langle \vec{\rho} \vec{V}_x \rangle \frac{\partial \vec{V}_{mag}}{\partial \vec{R}_y} \right] \\ \frac{\partial \vec{D}_x}{\partial \tau_f} &= -\frac{1}{2} C_D A_{ref} \left[\vec{\rho} \vec{V}_{mag} \frac{\partial \vec{V}_x}{\partial \tau_f} + \vec{\rho} \vec{V}_x \frac{\partial \vec{V}_{mag}}{\partial \tau_f} \right]\end{aligned}\tag{F.68}$$

$$\begin{aligned}\frac{\partial \vec{D}_y}{\partial \vec{R}_x} &= -\frac{1}{2} C_D A_{ref} \left[\langle \vec{V}_y \vec{V}_{mag} \rangle \frac{\partial \vec{\rho}}{\partial \vec{R}_x} + \langle \vec{\rho} \vec{V}_y \rangle \frac{\partial \vec{V}_{mag}}{\partial \vec{R}_x} \right] \\ \frac{\partial \vec{D}_y}{\partial \vec{R}_y} &= -\frac{1}{2} C_D A_{ref} \left[\langle \vec{V}_y \vec{V}_{mag} \rangle \frac{\partial \vec{\rho}}{\partial \vec{R}_y} + \langle \vec{\rho} \vec{V}_{mag} \rangle \frac{\partial \vec{V}_y}{\partial \vec{R}_x} + \langle \vec{\rho} \vec{V}_y \rangle \frac{\partial \vec{V}_{mag}}{\partial \vec{R}_y} \right] \\ \frac{\partial \vec{D}_y}{\partial \tau_f} &= -\frac{1}{2} C_D A_{ref} \left[\vec{\rho} \vec{V}_{mag} \frac{\partial \vec{V}_y}{\partial \tau_f} + \vec{\rho} \vec{V}_y \frac{\partial \vec{V}_{mag}}{\partial \tau_f} \right]\end{aligned}\tag{F.69}$$

All other partial derivatives are zero.

Partial Derivatives of Atmospheric Density

The atmospheric density and associated partial derivatives are the same as those defined for the D -method cartesian coordinates in Appendix F.1.

Partial Derivatives of Gravity

The gravitational acceleration and associated partial derivatives are the same as those defined for the D -method cartesian coordinates in Appendix F.1.

F.5 D^2 Radial-Transverse Polar Coordinate System Jacobian

The Jacobian for the two dimensional, D^2 radial-transverse polar coordinate system from section 5.2.2 is defined below. The Jacobian matrix is:

$$C_{Jac} = \frac{d\vec{C}}{d\vec{x}_{opt}} = \begin{bmatrix} \frac{\partial \vec{C}_R}{\partial R} & \frac{\partial \vec{C}_R}{\partial \theta} & \frac{\partial \vec{C}_R}{\partial m} & \frac{\partial \vec{C}_R}{\partial \vec{T}_R} & \frac{\partial \vec{C}_R}{\partial \vec{T}_\theta} & \frac{\partial \vec{C}_R}{\partial \vec{T}_f} \\ \frac{\partial \vec{C}_\theta}{\partial R} & \frac{\partial \vec{C}_\theta}{\partial \theta} & \frac{\partial \vec{C}_\theta}{\partial m} & \frac{\partial \vec{C}_\theta}{\partial \vec{T}_R} & \frac{\partial \vec{C}_\theta}{\partial \vec{T}_\theta} & \frac{\partial \vec{C}_\theta}{\partial \vec{T}_f} \\ \frac{\partial \vec{C}_m}{\partial R} & \frac{\partial \vec{C}_m}{\partial \theta} & \frac{\partial \vec{C}_m}{\partial m} & \frac{\partial \vec{C}_m}{\partial \vec{T}_R} & \frac{\partial \vec{C}_m}{\partial \vec{T}_\theta} & \frac{\partial \vec{C}_m}{\partial \vec{T}_f} \\ \frac{\partial \vec{C}_T}{\partial R} & \frac{\partial \vec{C}_T}{\partial \theta} & \frac{\partial \vec{C}_T}{\partial m} & \frac{\partial \vec{C}_T}{\partial \vec{T}_R} & \frac{\partial \vec{C}_T}{\partial \vec{T}_\theta} & \frac{\partial \vec{C}_T}{\partial \vec{T}_f} \\ \frac{\partial \vec{C}_q}{\partial R} & \frac{\partial \vec{C}_q}{\partial \theta} & \frac{\partial \vec{C}_q}{\partial m} & \frac{\partial \vec{C}_q}{\partial \vec{T}_R} & \frac{\partial \vec{C}_q}{\partial \vec{T}_\theta} & \frac{\partial \vec{C}_q}{\partial \vec{T}_f} \\ \frac{\partial C_{V_{R_o}}}{\partial R} & \frac{\partial C_{V_{R_o}}}{\partial \theta} & \frac{\partial C_{V_{R_o}}}{\partial m} & \frac{\partial C_{V_{R_o}}}{\partial \vec{T}_R} & \frac{\partial C_{V_{R_o}}}{\partial \vec{T}_\theta} & \frac{\partial C_{V_{R_o}}}{\partial \vec{T}_f} \\ \frac{\partial C_{V_{\theta_o}}}{\partial R} & \frac{\partial C_{V_{\theta_o}}}{\partial \theta} & \frac{\partial C_{V_{\theta_o}}}{\partial m} & \frac{\partial C_{V_{\theta_o}}}{\partial \vec{T}_R} & \frac{\partial C_{V_{\theta_o}}}{\partial \vec{T}_\theta} & \frac{\partial C_{V_{\theta_o}}}{\partial \vec{T}_f} \\ \frac{\partial C_{V_{R_f}}}{\partial R} & \frac{\partial C_{V_{R_f}}}{\partial \theta} & \frac{\partial C_{V_{R_f}}}{\partial m} & \frac{\partial C_{V_{R_f}}}{\partial \vec{T}_R} & \frac{\partial C_{V_{R_f}}}{\partial \vec{T}_\theta} & \frac{\partial C_{V_{R_f}}}{\partial \vec{T}_f} \\ \frac{\partial C_{V_{\theta_f}}}{\partial R} & \frac{\partial C_{V_{\theta_f}}}{\partial \theta} & \frac{\partial C_{V_{\theta_f}}}{\partial m} & \frac{\partial C_{V_{\theta_f}}}{\partial \vec{T}_R} & \frac{\partial C_{V_{\theta_f}}}{\partial \vec{T}_\theta} & \frac{\partial C_{V_{\theta_f}}}{\partial \vec{T}_f} \end{bmatrix}$$

Partial Derivatives of Velocities and Velocity Magnitude

The partial derivatives of the velocity with respect to R , θ and τ_f are necessary.

$$\begin{aligned}\frac{\partial \vec{V}_R}{\partial R} &= D_{NN} & \frac{\partial \vec{V}_\theta}{\partial R} &= \left\langle D_{NN} \vec{\theta} \right\rangle \\ \frac{\partial \vec{V}_R}{\partial \theta} &= [0] & \frac{\partial \vec{V}_\theta}{\partial \theta} &= \left\langle \vec{R} \right\rangle D_{NN} \\ \frac{\partial \vec{V}_R}{\partial \tau_f} &= \frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{R} & \frac{\partial \vec{V}_\theta}{\partial \tau_f} &= \left(\frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{\theta} \right) \vec{R}\end{aligned}\quad (\text{F.70})$$

The velocity magnitude is:

$$\vec{V}_{mag} = \sqrt{\vec{V}_R^2 + \vec{V}_\theta^2} \quad (\text{F.71})$$

The partial derivatives of the velocity magnitude are:

$$\begin{aligned}\frac{\partial \vec{V}_{mag}}{\partial R} &= \left\langle \frac{\vec{V}_R}{\vec{V}_{mag}} \right\rangle \frac{\partial \vec{V}_R}{\partial R} + \left\langle \frac{\vec{V}_\theta}{\vec{V}_{mag}} \right\rangle \frac{\partial \vec{V}_\theta}{\partial R} \\ \frac{\partial \vec{V}_{mag}}{\partial \theta} &= \left\langle \frac{\vec{V}_\theta}{\vec{V}_{mag}} \right\rangle \frac{\partial \vec{V}_\theta}{\partial \theta} \\ \frac{\partial \vec{V}_{mag}}{\partial \tau_f} &= \frac{1}{\vec{V}_{mag}} \left(\vec{V}_R \frac{\partial \vec{V}_R}{\partial \tau_f} + \vec{V}_\theta \frac{\partial \vec{V}_\theta}{\partial \tau_f} \right)\end{aligned}\quad (\text{F.72})$$

Partial Derivatives of \ddot{R} Constraint

$$\begin{aligned}\frac{\partial \vec{C}_{\ddot{R}}}{\partial \ddot{R}} &= D_{NN}^2 - \left\langle \frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{D}_R}{\partial \ddot{R}} + \frac{\partial \vec{g}}{\partial \ddot{R}} + \left\langle \frac{\vec{V}_\theta^2}{\ddot{R}} \right\rangle - \left\langle \frac{2\vec{V}_\theta}{\ddot{R}} \right\rangle \frac{\partial \vec{V}_\theta}{\partial \ddot{R}} \\ \frac{\partial \vec{C}_{\ddot{R}}}{\partial \theta} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{D}_R}{\partial \theta} - \left\langle \frac{2\vec{V}_\theta}{\ddot{R}} \right\rangle \frac{\partial \vec{V}_\theta}{\partial \theta} \\ \frac{\partial \vec{C}_{\ddot{R}}}{\partial \bar{m}} &= \left\langle \frac{\vec{T}_R + \vec{T}_\theta}{\bar{m}^2} \right\rangle \\ \frac{\partial \vec{C}_{\ddot{R}}}{\partial \vec{T}_R} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \\ \frac{\partial \vec{C}_{\ddot{R}}}{\partial \vec{T}_\theta} &= [0] \\ \frac{\partial \vec{C}_{\ddot{R}}}{\partial \tau_f} &= \frac{-2D_{NN}^2}{(\tau_f - \tau_o)} \vec{R} - \frac{1}{\bar{m}} \frac{\partial \vec{D}_R}{\partial \tau_f} - \frac{2\vec{V}_\theta}{\ddot{R}} \frac{\partial \vec{V}_\theta}{\partial \tau_f}\end{aligned}\quad (\text{F.73})$$

Partial Derivatives of $\ddot{\theta}$ Constraint

$$\begin{aligned}
\frac{\partial \tilde{C}_{\dot{\theta}}}{\partial R} &= \left\langle \frac{\vec{T}_\theta + \vec{D}_\theta}{\bar{m}R^2} \right\rangle - \left\langle \frac{4\vec{V}_R \vec{V}_\theta}{R^3} \right\rangle - \left\langle \frac{1}{\bar{m}R} \right\rangle \frac{\partial \vec{D}_\theta}{\partial R} + \left\langle \frac{2\vec{V}_\theta}{R^2} \right\rangle \frac{\partial \vec{V}_R}{\partial R} + \left\langle \frac{2\vec{V}_R}{R^2} \right\rangle \frac{\partial \vec{V}_\theta}{\partial R} \\
\frac{\partial \tilde{C}_{\dot{\theta}}}{\partial \theta} &= D_{NN}^2 - \left\langle \frac{1}{\bar{m}R} \right\rangle \frac{\partial \vec{D}_\theta}{\partial \theta} + \left\langle \frac{2\vec{V}_R}{R^2} \right\rangle \frac{\partial \vec{V}_\theta}{\partial \theta} \\
\frac{\partial \tilde{C}_{\dot{\theta}}}{\partial \bar{m}} &= \left\langle \frac{\vec{T}_\theta + \vec{D}_\theta}{\bar{m}^2 R} \right\rangle \\
\frac{\partial \tilde{C}_{\dot{\theta}}}{\partial T_R} &= [0] \\
\frac{\partial \tilde{C}_{\dot{\theta}}}{\partial T_\theta} &= \left\langle -\frac{1}{\bar{m}R} \right\rangle \\
\frac{\partial \tilde{C}_{\dot{\theta}}}{\partial \tau_f} &= \frac{-2D_{NN}^2}{(\tau_f - \tau_o)} \vec{\theta} - \frac{1}{\bar{m}R} \frac{\partial \vec{D}_\theta}{\partial \tau_f} + \frac{2\vec{V}_\theta}{R^2} \frac{\partial \vec{V}_R}{\partial \tau_f} + \frac{2\vec{V}_R}{R^2} \frac{\partial \vec{V}_\theta}{\partial \tau_f}
\end{aligned} \tag{F.74}$$

Partial Derivatives of \dot{m} Constraint

$$\begin{aligned}
\frac{\partial \tilde{C}_{\dot{m}}}{\partial R} &= [0] & \frac{\partial \tilde{C}_{\dot{m}}}{\partial T_R} &= \frac{1}{g_o I_{sp}} \left\langle \frac{\vec{T}_R}{\sqrt{\vec{T}_R^2 + \vec{T}_\theta^2}} \right\rangle \\
\frac{\partial \tilde{C}_{\dot{m}}}{\partial \theta} &= [0] & \frac{\partial \tilde{C}_{\dot{m}}}{\partial T_\theta} &= \frac{1}{g_o I_{sp}} \left\langle \frac{\vec{T}_\theta}{\sqrt{\vec{T}_R^2 + \vec{T}_\theta^2}} \right\rangle \\
\frac{\partial \tilde{C}_{\dot{m}}}{\partial \bar{m}} &= D_{NN} & \frac{\partial \tilde{C}_{\dot{m}}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \bar{m} \right]
\end{aligned} \tag{F.75}$$

Partial Derivatives of Thrust Magnitude Constraint

$$\begin{aligned}
\frac{\partial \tilde{C}_T}{\partial R} &= [0] & \frac{\partial \tilde{C}_T}{\partial T_R} &= \left\langle 2\vec{T}_R \right\rangle \\
\frac{\partial \tilde{C}_T}{\partial \theta} &= [0] & \frac{\partial \tilde{C}_T}{\partial T_\theta} &= \left\langle 2\vec{T}_\theta \right\rangle \\
\frac{\partial \tilde{C}_T}{\partial \bar{m}} &= [0] & \frac{\partial \tilde{C}_T}{\partial \tau_f} &= [0]
\end{aligned} \tag{F.76}$$

Partial Derivatives of Dynamic Pressure Constraint

The dynamic pressure is given by:

$$\vec{q} = \frac{1}{2} \vec{\rho} \vec{V}_{mag}^2 \tag{F.77}$$

$$\begin{aligned}
\frac{\partial \tilde{C}_q}{\partial R} &= \frac{\partial \vec{q}}{\partial R} = \left\langle \frac{1}{2} \vec{V}_{mag}^2 \right\rangle \frac{\partial \vec{\rho}}{\partial R} + \left\langle \vec{\rho} \vec{V}_{mag} \right\rangle \frac{\partial \vec{V}_{mag}}{\partial R} \\
\frac{\partial \tilde{C}_q}{\partial \theta} &= \frac{\partial \vec{q}}{\partial \theta} = \left\langle \vec{\rho} \vec{V}_{mag} \right\rangle \frac{\partial \vec{V}_{mag}}{\partial \theta} \\
\frac{\partial \tilde{C}_q}{\partial \tau_f} &= \frac{\partial \vec{q}}{\partial \tau_f} = \vec{\rho} \vec{V}_{mag} \frac{\partial \vec{V}_{mag}}{\partial \tau_f}
\end{aligned} \tag{F.78}$$

All other partial derivatives are zero.

Partial Derivatives of Sensed Acceleration Constraint

The sensed acceleration is given by:

$$\begin{aligned}\vec{a}_{\text{sensed}_R} &= \frac{\vec{T}_R + \vec{D}_R}{\bar{m}} \\ \vec{a}_{\text{sensed}_\theta} &= \frac{\vec{T}_\theta + \vec{D}_\theta}{\bar{m}} \\ (\vec{a}_{\text{sensed}})^2_{\text{mag}} &= \vec{a}_{\text{sensed}_R}^2 + \vec{a}_{\text{sensed}_\theta}^2\end{aligned}\tag{F.79}$$

$$\begin{aligned}\frac{\partial \vec{C}_a}{\partial R} &= 2 \left\langle \frac{1}{\bar{m}} \right\rangle \left[\langle \vec{a}_{\text{sensed}_R} \rangle \frac{\partial \vec{D}_R}{\partial R} + \langle \vec{a}_{\text{sensed}_\theta} \rangle \frac{\partial \vec{D}_\theta}{\partial R} \right] \\ \frac{\partial \vec{C}_a}{\partial \theta} &= 2 \left\langle \frac{1}{\bar{m}} \right\rangle \left[\langle \vec{a}_{\text{sensed}_R} \rangle \frac{\partial \vec{D}_R}{\partial \theta} + \langle \vec{a}_{\text{sensed}_\theta} \rangle \frac{\partial \vec{D}_\theta}{\partial \theta} \right] \\ \frac{\partial \vec{C}_a}{\partial \bar{m}} &= -2 \left\langle \frac{\vec{a}_{\text{sensed}_R}^2 + \vec{a}_{\text{sensed}_\theta}^2}{\bar{m}} \right\rangle \\ \frac{\partial \vec{C}_a}{\partial \vec{T}_R} &= 2 \left\langle \frac{\vec{a}_{\text{sensed}_R}}{\bar{m}} \right\rangle \\ \frac{\partial \vec{C}_a}{\partial \vec{T}_\theta} &= 2 \left\langle \frac{\vec{a}_{\text{sensed}_\theta}}{\bar{m}} \right\rangle \\ \frac{\partial \vec{C}_a}{\partial \vec{\tau}_f} &= \frac{2}{\bar{m}} \left(\vec{a}_{\text{sensed}_R} \frac{\partial \vec{D}_R}{\partial \vec{\tau}_f} + \vec{a}_{\text{sensed}_\theta} \frac{\partial \vec{D}_\theta}{\partial \vec{\tau}_f} \right)\end{aligned}\tag{F.80}$$

Partial Derivatives of Initial Velocity Constraints

Partial derivatives for $C_{V_{R_o}}$ constraint:

$$\begin{aligned}\frac{\partial C_{V_{R_o}}}{\partial \vec{R}} &= [d_{NN_1}] & \frac{\partial C_{V_{R_o}}}{\partial \vec{T}_R} &= [0] \\ \frac{\partial C_{V_{R_o}}}{\partial \theta} &= [0] & \frac{\partial C_{V_{R_o}}}{\partial \vec{T}_\theta} &= [0] \\ \frac{\partial C_{V_{R_o}}}{\partial \bar{m}} &= [0] & \frac{\partial C_{V_{R_o}}}{\partial \vec{\tau}_f} &= \frac{-d_{NN_1}}{(\vec{\tau}_f - \vec{\tau}_o)} \vec{R}\end{aligned}\tag{F.81}$$

Partial derivatives for $C_{V_{\theta_o}}$ constraint:

$$\begin{aligned}\frac{\partial C_{V_{\theta_o}}}{\partial \vec{R}} &= \begin{bmatrix} 0 & \dots & 0 & d_{NN_1} \vec{\theta} \end{bmatrix} & \frac{\partial C_{V_{\theta_o}}}{\partial \vec{T}_R} &= [0] \\ \frac{\partial C_{V_{\theta_o}}}{\partial \theta} &= [d_{NN_1} R_1] & \frac{\partial C_{V_{\theta_o}}}{\partial \vec{T}_\theta} &= [0] \\ \frac{\partial C_{V_{\theta_o}}}{\partial \bar{m}} &= [0] & \frac{\partial C_{V_{\theta_o}}}{\partial \vec{\tau}_f} &= \left(\frac{-d_{NN_1}}{(\vec{\tau}_f - \vec{\tau}_o)} \vec{\theta} \right) R_1\end{aligned}\tag{F.82}$$

where d_{NN_1} is the first row of the D_{NN} matrix. Note that it is equivalent to a row vector.

Partial Derivatives of Final Velocity Constraints

Partial derivatives for $C_{V_{R_f}}$ constraint:

$$\begin{aligned}\frac{\partial C_{V_{R_f}}}{\partial \vec{R}} &= [d_{NN_f}] & \frac{\partial C_{V_{R_f}}}{\partial \vec{T}_R} &= [0] \\ \frac{\partial C_{V_{R_f}}}{\partial \vec{\theta}} &= [0] & \frac{\partial C_{V_{R_f}}}{\partial \vec{T}_\theta} &= [0] \\ \frac{\partial C_{V_{R_f}}}{\partial \vec{m}} &= [0] & \frac{\partial C_{V_{R_f}}}{\partial \tau_f} &= \frac{-d_{NN_f}}{(\tau_f - \tau_o)} \vec{R}\end{aligned}\quad (\text{F.83})$$

Partial derivatives for C_{V_θ} constraint:

$$\begin{aligned}\frac{\partial C_{V_\theta}}{\partial \vec{R}} &= \begin{bmatrix} 0 & \dots & 0 & d_{NN_f} \vec{\theta} \end{bmatrix} & \frac{\partial C_{V_{R_f}}}{\partial \vec{T}_R} &= [0] \\ \frac{\partial C_{V_\theta}}{\partial \vec{\theta}} &= [d_{NN_f} R_f] & \frac{\partial C_{V_{R_f}}}{\partial \vec{T}_\theta} &= [0] \\ \frac{\partial C_{V_\theta}}{\partial \vec{m}} &= [0] & \frac{\partial C_{V_{R_f}}}{\partial \tau_f} &= \left(\frac{-d_{NN_f}}{(\tau_f - \tau_o)} \vec{\theta} \right) R_f\end{aligned}\quad (\text{F.84})$$

where d_{NN_f} is the last row of the D_{NN} matrix. Note that it is equivalent to a row vector.

Partial Derivatives of Drag

The drag is defined below:

$$\begin{aligned}\vec{D}_R &= \frac{1}{2} \vec{\rho} \vec{V}_{mag}^2 C_D A_{ref} \frac{-\vec{V}_R}{\vec{V}_{mag}} = -\frac{1}{2} \vec{\rho} \vec{V}_R \vec{V}_{mag} C_D A_{ref} \\ \vec{D}_\theta &= \frac{1}{2} \vec{\rho} \vec{V}_{mag}^2 C_D A_{ref} \frac{-\vec{V}_\theta}{\vec{V}_{mag}} = -\frac{1}{2} \vec{\rho} \vec{V}_\theta \vec{V}_{mag} C_D A_{ref}\end{aligned}\quad (\text{F.85})$$

$$\begin{aligned}\frac{\partial \vec{D}_R}{\partial \vec{R}} &= -\frac{1}{2} C_D A_{ref} \left[\langle \vec{V}_R \vec{V}_{mag} \rangle \frac{\partial \vec{\rho}}{\partial \vec{R}} + \langle \vec{\rho} \vec{V}_{mag} \rangle \frac{\partial \vec{V}_R}{\partial \vec{R}} + \langle \vec{\rho} \vec{V}_R \rangle \frac{\partial \vec{V}_{mag}}{\partial \vec{R}} \right] \\ \frac{\partial \vec{D}_R}{\partial \vec{\theta}} &= -\frac{1}{2} C_D A_{ref} \left[\langle \vec{\rho} \vec{V}_R \rangle \frac{\partial \vec{V}_{mag}}{\partial \vec{\theta}} \right] \\ \frac{\partial \vec{D}_R}{\partial \tau_f} &= -\frac{1}{2} C_D A_{ref} \left[\vec{\rho} \vec{V}_{mag} \frac{\partial \vec{V}_R}{\partial \tau_f} + \vec{\rho} \vec{V}_R \frac{\partial \vec{V}_{mag}}{\partial \tau_f} \right]\end{aligned}\quad (\text{F.86})$$

$$\begin{aligned}
 \frac{\partial \tilde{D}_\theta}{\partial R} &= -\frac{1}{2} C_D A_{ref} \left[\langle \vec{V}_\theta \vec{V}_{mag} \rangle \frac{\partial \tilde{\rho}}{\partial R} + \langle \tilde{\rho} \vec{V}_{mag} \rangle \frac{\partial \vec{V}_\theta}{\partial R} + \langle \tilde{\rho} \vec{V}_\theta \rangle \frac{\partial \vec{V}_{mag}}{\partial R} \right] \\
 \frac{\partial \tilde{D}_\theta}{\partial \theta} &= -\frac{1}{2} C_D A_{ref} \left[\langle \tilde{\rho} \vec{V}_{mag} \rangle \frac{\partial \vec{V}_\theta}{\partial \theta} \right] + \frac{1}{2} C_D A_{ref} \left[\langle \tilde{\rho} \vec{V}_\theta \rangle \frac{\partial \vec{V}_{mag}}{\partial \theta} \right] \\
 \frac{\partial \tilde{D}_\theta}{\partial \tau_f} &= -\frac{1}{2} C_D A_{ref} \left[\tilde{\rho} \vec{V}_{mag} \frac{\partial \vec{V}_\theta}{\partial \tau_f} + \tilde{\rho} \vec{V}_\theta \frac{\partial \vec{V}_{mag}}{\partial \tau_f} \right]
 \end{aligned} \tag{F.87}$$

All other partial derivatives are zero.

Partial Derivatives of Atmospheric Density

The atmospheric density and associated partial derivatives are the same as those defined for the *D*-method radial-transverse coordinates in Appendix F.2.

Partial Derivatives of Gravity

The gravitational acceleration and associated partial derivatives are the same as those defined for the *D*-method radial-transverse coordinates in Appendix F.2.

Appendix G

Three Dimensional Cartesian Model: Jacobian Derivation

The analytic Jacobian for each constraint in the three dimensional model must be found in order to speed up the nonlinear program solver. The constraints for the three dimensional model are defined in Chapter 6. Recall that the Jacobian matrix is defined by:

$$C_{Jac} = \frac{\partial \vec{C}}{\partial \vec{x}_{opt}} \quad (G.1)$$

G.1 Dynamic Constraint Jacobian

D-Method

- Partial Derivatives of \dot{R}_x Constraint

$$\begin{aligned} \frac{\partial \vec{C}_{\dot{R}_x}}{\partial \vec{R}_x} &= D_{NN} & \frac{\partial \vec{C}_{\dot{R}_x}}{\partial \vec{T}} &= [0] \\ \frac{\partial \vec{C}_{\dot{R}_x}}{\partial \vec{R}_y} &= [0] & \frac{\partial \vec{C}_{\dot{R}_x}}{\partial \vec{q}_1} &= [0] \\ \frac{\partial \vec{C}_{\dot{R}_x}}{\partial \vec{R}_z} &= [0] & \frac{\partial \vec{C}_{\dot{R}_x}}{\partial \vec{q}_2} &= [0] \\ \frac{\partial \vec{C}_{\dot{R}_x}}{\partial \vec{V}_x} &= \langle -1 \rangle & \frac{\partial \vec{C}_{\dot{R}_x}}{\partial \vec{q}_3} &= [0] \end{aligned} \quad (G.2)$$

$$\begin{aligned}
\frac{\partial \vec{C}_{\dot{R}_x}}{\partial \vec{V}_y} &= [0] & \frac{\partial \vec{C}_{\dot{R}_x}}{\partial \vec{q}_4} &= [0] \\
\frac{\partial \vec{C}_{\dot{R}_x}}{\partial \vec{V}_z} &= [0] & \frac{\partial \vec{C}_{\dot{R}_x}}{\partial \tau_f} &= \left[\frac{-2D_{NN}}{(\tau_f - \tau_o)} \vec{R}_x \right] \\
\frac{\partial \vec{C}_{\dot{R}_x}}{\partial \vec{m}} &= [0]
\end{aligned}$$

- Partial Derivatives of \dot{R}_y Constraint

$$\begin{aligned}
\frac{\partial \vec{C}_{\dot{R}_y}}{\partial \vec{R}_x} &= [0] & \frac{\partial \vec{C}_{\dot{R}_y}}{\partial \vec{T}} &= [0] \\
\frac{\partial \vec{C}_{\dot{R}_y}}{\partial \vec{R}_y} &= D_{NN} & \frac{\partial \vec{C}_{\dot{R}_y}}{\partial \vec{q}_1} &= [0] \\
\frac{\partial \vec{C}_{\dot{R}_y}}{\partial \vec{R}_z} &= [0] & \frac{\partial \vec{C}_{\dot{R}_y}}{\partial \vec{q}_2} &= [0] \\
\frac{\partial \vec{C}_{\dot{R}_y}}{\partial \vec{V}_x} &= [0] & \frac{\partial \vec{C}_{\dot{R}_y}}{\partial \vec{q}_3} &= [0] \\
\frac{\partial \vec{C}_{\dot{R}_y}}{\partial \vec{V}_y} &= \langle -1 \rangle & \frac{\partial \vec{C}_{\dot{R}_y}}{\partial \vec{q}_4} &= [0] \\
\frac{\partial \vec{C}_{\dot{R}_y}}{\partial \vec{V}_z} &= [0] & \frac{\partial \vec{C}_{\dot{R}_y}}{\partial \tau_f} &= \left[\frac{-2D_{NN}}{(\tau_f - \tau_o)} \vec{R}_y \right] \\
\frac{\partial \vec{C}_{\dot{R}_y}}{\partial \vec{m}} &= [0]
\end{aligned} \tag{G.3}$$

- Partial Derivatives of \dot{R}_z Constraint

$$\begin{aligned}
\frac{\partial \vec{C}_{\dot{R}_z}}{\partial \vec{R}_x} &= [0] & \frac{\partial \vec{C}_{\dot{R}_z}}{\partial \vec{T}} &= [0] \\
\frac{\partial \vec{C}_{\dot{R}_z}}{\partial \vec{R}_y} &= [0] & \frac{\partial \vec{C}_{\dot{R}_z}}{\partial \vec{q}_1} &= [0] \\
\frac{\partial \vec{C}_{\dot{R}_z}}{\partial \vec{R}_z} &= D_{NN} & \frac{\partial \vec{C}_{\dot{R}_z}}{\partial \vec{q}_2} &= [0] \\
\frac{\partial \vec{C}_{\dot{R}_z}}{\partial \vec{V}_x} &= [0] & \frac{\partial \vec{C}_{\dot{R}_z}}{\partial \vec{q}_3} &= [0] \\
\frac{\partial \vec{C}_{\dot{R}_z}}{\partial \vec{V}_y} &= [0] & \frac{\partial \vec{C}_{\dot{R}_z}}{\partial \vec{q}_4} &= [0]
\end{aligned} \tag{G.4}$$

$$\begin{aligned}\frac{\partial \vec{C}_{\dot{R}_z}}{\partial \vec{V}_z} &= \langle -1 \rangle & \frac{\partial \vec{C}_{\dot{R}_z}}{\partial \tau_f} &= \left[\frac{-2D_{NN}}{(\tau_f - \tau_o)} \vec{R}_z \right] \\ \frac{\partial \vec{C}_{\dot{R}_z}}{\partial \vec{m}} &= [0]\end{aligned}$$

- Partial Derivatives of \dot{V}_x Constraint

$$\begin{aligned}\frac{\partial \vec{C}_{\dot{V}_x}}{\partial \vec{R}_x} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_x}{\partial \vec{R}_x} - \frac{\partial \vec{q}_x}{\partial \vec{R}_x} & \frac{\partial \vec{C}_{\dot{V}_x}}{\partial \vec{T}} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{T}_x}{\partial \vec{T}} \\ \frac{\partial \vec{C}_{\dot{V}_x}}{\partial \vec{R}_y} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_x}{\partial \vec{R}_y} - \frac{\partial \vec{q}_x}{\partial \vec{R}_y} & \frac{\partial \vec{C}_{\dot{V}_x}}{\partial \vec{q}_1} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_x}{\partial \vec{q}_1} + \frac{\partial \vec{A}_x}{\partial \vec{q}_1} \right] \\ \frac{\partial \vec{C}_{\dot{V}_x}}{\partial \vec{R}_z} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_x}{\partial \vec{R}_z} - \frac{\partial \vec{q}_x}{\partial \vec{R}_z} & \frac{\partial \vec{C}_{\dot{V}_x}}{\partial \vec{q}_2} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_x}{\partial \vec{q}_2} + \frac{\partial \vec{A}_x}{\partial \vec{q}_2} \right] \\ \frac{\partial \vec{C}_{\dot{V}_x}}{\partial \vec{V}_x} &= D_{NN} + \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_x}{\partial \vec{V}_x} & \frac{\partial \vec{C}_{\dot{V}_x}}{\partial \vec{q}_3} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_x}{\partial \vec{q}_3} + \frac{\partial \vec{A}_x}{\partial \vec{q}_3} \right] \\ \frac{\partial \vec{C}_{\dot{V}_x}}{\partial \vec{V}_y} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_x}{\partial \vec{V}_y} & \frac{\partial \vec{C}_{\dot{V}_x}}{\partial \vec{q}_4} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_x}{\partial \vec{q}_4} + \frac{\partial \vec{A}_x}{\partial \vec{q}_4} \right] \\ \frac{\partial \vec{C}_{\dot{V}_x}}{\partial \vec{V}_z} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_x}{\partial \vec{V}_z} & \frac{\partial \vec{C}_{\dot{V}_x}}{\partial \tau_f} &= \left[\frac{-2D_{NN}}{(\tau_f - \tau_o)} \vec{V}_x \right] \\ \frac{\partial \vec{C}_{\dot{V}_x}}{\partial \vec{m}} &= \left\langle \frac{\vec{T}_x + \vec{A}_x}{\vec{m}^2} \right\rangle\end{aligned}\tag{G.5}$$

- Partial Derivatives of \dot{V}_y Constraint

$$\begin{aligned}\frac{\partial \vec{C}_{\dot{V}_y}}{\partial \vec{R}_x} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_y}{\partial \vec{R}_x} - \frac{\partial \vec{q}_y}{\partial \vec{R}_x} & \frac{\partial \vec{C}_{\dot{V}_y}}{\partial \vec{T}} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{T}_y}{\partial \vec{T}} \\ \frac{\partial \vec{C}_{\dot{V}_y}}{\partial \vec{R}_y} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_y}{\partial \vec{R}_y} - \frac{\partial \vec{q}_y}{\partial \vec{R}_y} & \frac{\partial \vec{C}_{\dot{V}_y}}{\partial \vec{q}_1} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_y}{\partial \vec{q}_1} + \frac{\partial \vec{A}_y}{\partial \vec{q}_1} \right] \\ \frac{\partial \vec{C}_{\dot{V}_y}}{\partial \vec{R}_z} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_y}{\partial \vec{R}_z} - \frac{\partial \vec{q}_y}{\partial \vec{R}_z} & \frac{\partial \vec{C}_{\dot{V}_y}}{\partial \vec{q}_2} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_y}{\partial \vec{q}_2} + \frac{\partial \vec{A}_y}{\partial \vec{q}_2} \right] \\ \frac{\partial \vec{C}_{\dot{V}_y}}{\partial \vec{V}_x} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_y}{\partial \vec{V}_x} & \frac{\partial \vec{C}_{\dot{V}_y}}{\partial \vec{q}_3} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_y}{\partial \vec{q}_3} + \frac{\partial \vec{A}_y}{\partial \vec{q}_3} \right] \\ \frac{\partial \vec{C}_{\dot{V}_y}}{\partial \vec{V}_y} &= D_{NN} + \left\langle -\frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_y}{\partial \vec{V}_y} & \frac{\partial \vec{C}_{\dot{V}_y}}{\partial \vec{q}_4} &= \left\langle -\frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_y}{\partial \vec{q}_4} + \frac{\partial \vec{A}_y}{\partial \vec{q}_4} \right]\end{aligned}\tag{G.6}$$

$$\begin{aligned}\frac{\partial \vec{C}_{\dot{V}_y}}{\partial \vec{V}_z} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{A}_y}{\partial V_z} & \frac{\partial \vec{C}_{\dot{V}_y}}{\partial \tau_f} &= \left[\frac{-2D_{NN}}{(\tau_f - \tau_o)} \vec{V}_y \right] \\ \frac{\partial \vec{C}_{\dot{V}_y}}{\partial \bar{m}} &= \left\langle \frac{\vec{T}_y + \vec{A}_y}{\bar{m}^2} \right\rangle\end{aligned}$$

- Partial Derivatives of \dot{V}_z Constraint

$$\begin{aligned}\frac{\partial \vec{C}_{\dot{V}_z}}{\partial \vec{R}_x} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{A}_z}{\partial R_x} - \frac{\partial \vec{q}_z}{\partial \vec{R}_x} & \frac{\partial \vec{C}_{\dot{V}_z}}{\partial \vec{T}} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \frac{\partial T_z}{\partial \vec{T}} \\ \frac{\partial \vec{C}_{\dot{V}_z}}{\partial \vec{R}_y} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{A}_z}{\partial R_y} - \frac{\partial \vec{q}_z}{\partial \vec{R}_y} & \frac{\partial \vec{C}_{\dot{V}_z}}{\partial \vec{q}_1} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \left[\frac{\partial \vec{T}_z}{\partial \vec{q}_1} + \frac{\partial \vec{A}_z}{\partial \vec{q}_1} \right] \\ \frac{\partial \vec{C}_{\dot{V}_z}}{\partial \vec{R}_z} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{A}_z}{\partial R_z} - \frac{\partial \vec{q}_z}{\partial \vec{R}_z} & \frac{\partial \vec{C}_{\dot{V}_z}}{\partial \vec{q}_2} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \left[\frac{\partial \vec{T}_z}{\partial \vec{q}_2} + \frac{\partial \vec{A}_z}{\partial \vec{q}_2} \right] \\ \frac{\partial \vec{C}_{\dot{V}_z}}{\partial \vec{V}_x} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{A}_z}{\partial V_x} & \frac{\partial \vec{C}_{\dot{V}_z}}{\partial \vec{q}_3} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \left[\frac{\partial \vec{T}_z}{\partial \vec{q}_3} + \frac{\partial \vec{A}_z}{\partial \vec{q}_3} \right] \quad (G.7) \\ \frac{\partial \vec{C}_{\dot{V}_z}}{\partial \vec{V}_y} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{A}_z}{\partial V_y} & \frac{\partial \vec{C}_{\dot{V}_z}}{\partial \vec{q}_4} &= \left\langle -\frac{1}{\bar{m}} \right\rangle \left[\frac{\partial \vec{T}_z}{\partial \vec{q}_4} + \frac{\partial \vec{A}_z}{\partial \vec{q}_4} \right] \\ \frac{\partial \vec{C}_{\dot{V}_z}}{\partial \vec{V}_z} &= D_{NN} + \left\langle -\frac{1}{\bar{m}} \right\rangle \frac{\partial \vec{A}_z}{\partial V_z} & \frac{\partial \vec{C}_{\dot{V}_z}}{\partial \tau_f} &= \left[\frac{-2D_{NN}}{(\tau_f - \tau_o)} \vec{V}_z \right] \\ \frac{\partial \vec{C}_{\dot{V}_z}}{\partial \bar{m}} &= \left\langle \frac{\vec{T}_z + \vec{A}_z}{\bar{m}^2} \right\rangle\end{aligned}$$

- Partial Derivatives of \dot{m} Constraint

$$\begin{aligned}\frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{R}_x} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{T}} &= \left\langle \frac{\vec{T}}{V_{exit}} \right\rangle \\ \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{R}_y} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{q}_1} &= [0] \\ \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{R}_y} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{q}_2} &= [0] \\ \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{V}_x} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{q}_3} &= [0] \quad (G.8) \\ \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{V}_y} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{q}_4} &= [0] \\ \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{V}_y} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \bar{m} \right]\end{aligned}$$

$$\frac{\partial \vec{C}_{\vec{R}_m}}{\partial \vec{m}} = D_{NN}$$

D^2 -Method

- Partial Derivatives of \ddot{R}_x Constraint

$$\begin{aligned}
 \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{R}_x} &= D_{NN}^2 - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_x}{\partial \vec{R}_x} - \frac{\partial \vec{g}_x}{\partial \vec{R}_x} & \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{q}_1} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_x}{\partial \vec{q}_1} + \frac{\partial \vec{A}_x}{\partial \vec{q}_1} \right] \\
 \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{R}_y} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_x}{\partial \vec{R}_y} - \frac{\partial \vec{g}_x}{\partial \vec{R}_y} & \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{q}_2} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_x}{\partial \vec{q}_2} + \frac{\partial \vec{A}_x}{\partial \vec{q}_2} \right] \\
 \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{R}_z} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_x}{\partial \vec{R}_z} - \frac{\partial \vec{g}_x}{\partial \vec{R}_z} & \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{q}_3} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_x}{\partial \vec{q}_3} + \frac{\partial \vec{A}_x}{\partial \vec{q}_3} \right] \quad (G.9) \\
 \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{m}} &= \left\langle \frac{\vec{T}_x + \vec{A}_x}{\vec{m}^2} \right\rangle & \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{q}_4} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_x}{\partial \vec{q}_4} + \frac{\partial \vec{A}_x}{\partial \vec{q}_4} \right] \\
 \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{T}} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{T}_x}{\partial \vec{T}} & \frac{\partial \vec{C}_{\vec{R}_x}}{\partial \tau_f} &= \left[\frac{-2D_{NN}^2}{(\tau_f - \tau_o)} \vec{R}_x - \frac{1}{\vec{m}} \frac{\partial \vec{A}_x}{\partial \tau_f} \right]
 \end{aligned}$$

- Partial Derivatives of \ddot{R}_y Constraint

$$\begin{aligned}
 \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{R}_x} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_y}{\partial \vec{R}_x} - \frac{\partial \vec{g}_y}{\partial \vec{R}_x} & \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{q}_1} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_y}{\partial \vec{q}_1} + \frac{\partial \vec{A}_y}{\partial \vec{q}_1} \right] \\
 \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{R}_y} &= D_{NN}^2 - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_y}{\partial \vec{R}_y} - \frac{\partial \vec{g}_y}{\partial \vec{R}_y} & \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{q}_2} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_y}{\partial \vec{q}_2} + \frac{\partial \vec{A}_y}{\partial \vec{q}_2} \right] \\
 \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{R}_z} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_y}{\partial \vec{R}_z} - \frac{\partial \vec{g}_y}{\partial \vec{R}_z} & \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{q}_3} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_y}{\partial \vec{q}_3} + \frac{\partial \vec{A}_y}{\partial \vec{q}_3} \right] \quad (G.10) \\
 \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{m}} &= \left\langle \frac{\vec{T}_y + \vec{A}_y}{\vec{m}^2} \right\rangle & \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{q}_4} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_y}{\partial \vec{q}_4} + \frac{\partial \vec{A}_y}{\partial \vec{q}_4} \right] \\
 \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \vec{T}} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{T}_y}{\partial \vec{T}} & \frac{\partial \vec{C}_{\vec{R}_y}}{\partial \tau_f} &= \left[\frac{-2D_{NN}^2}{(\tau_f - \tau_o)} \vec{R}_y - \frac{1}{\vec{m}} \frac{\partial \vec{A}_y}{\partial \tau_f} \right]
 \end{aligned}$$

- Partial Derivatives of \ddot{R}_z Constraint

$$\begin{aligned}
 \frac{\partial \vec{C}_{\vec{R}_z}}{\partial \vec{R}_x} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_z}{\partial \vec{R}_x} - \frac{\partial \vec{g}_z}{\partial \vec{R}_x} & \frac{\partial \vec{C}_{\vec{R}_z}}{\partial \vec{q}_1} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_z}{\partial \vec{q}_1} + \frac{\partial \vec{A}_z}{\partial \vec{q}_1} \right]
 \end{aligned}$$

$$\begin{aligned}
\frac{\partial \vec{C}_{\vec{R}_x}}{\partial \vec{R}_y} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_x}{\partial \vec{R}_y} - \frac{\partial \vec{q}_x}{\partial \vec{R}_y} & \frac{\partial \vec{C}_{\vec{R}_z}}{\partial \vec{q}_2} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_z}{\partial \vec{q}_2} + \frac{\partial \vec{A}_z}{\partial \vec{q}_2} \right] \\
\frac{\partial \vec{C}_{\vec{R}_z}}{\partial \vec{R}_z} &= D_{NN}^2 - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{A}_z}{\partial \vec{R}_z} - \frac{\partial \vec{q}_z}{\partial \vec{R}_z} & \frac{\partial \vec{C}_{\vec{R}_z}}{\partial \vec{q}_3} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_z}{\partial \vec{q}_3} + \frac{\partial \vec{A}_z}{\partial \vec{q}_3} \right] \quad (\text{G.11}) \\
\frac{\partial \vec{C}_{\vec{R}_z}}{\partial \vec{m}} &= \left\langle \frac{\vec{T}_z + \vec{A}_z}{\vec{m}^2} \right\rangle & \frac{\partial \vec{C}_{\vec{R}_z}}{\partial \vec{q}_4} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \left[\frac{\partial \vec{T}_z}{\partial \vec{q}_4} + \frac{\partial \vec{A}_z}{\partial \vec{q}_4} \right] \\
\frac{\partial \vec{C}_{\vec{R}_z}}{\partial \vec{T}} &= - \left\langle \frac{1}{\vec{m}} \right\rangle \frac{\partial \vec{T}}{\partial \vec{T}} & \frac{\partial \vec{C}_{\vec{R}_z}}{\partial \tau_f} &= \left[\frac{-2D_{NN}^2}{(\tau_f - \tau_o)} \vec{R}_z - \frac{1}{\vec{m}} \frac{\partial \vec{A}_z}{\partial \tau_f} \right]
\end{aligned}$$

- Partial Derivatives of \dot{m} Constraint

$$\begin{aligned}
\frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{R}_x} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{q}_1} &= [0] \\
\frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{R}_y} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{q}_2} &= [0] \\
\frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{R}_z} &= [0] & \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{q}_3} &= [0] \quad (\text{G.12}) \\
\frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{m}} &= D_{NN} & \frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{q}_4} &= [0] \\
\frac{\partial \vec{C}_{\dot{m}}}{\partial \vec{T}} &= \left\langle \frac{\vec{T}}{V_{exit}} \right\rangle & \frac{\partial \vec{C}_{\dot{m}}}{\partial \tau_f} &= \left[\frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{m} \right]
\end{aligned}$$

G.2 Trajectory Constraint Jacobian

D-Method

- Partial Derivatives of Quaternion Normalization Constraint (\vec{C}_{qnorm})

$$\begin{aligned}
\frac{\partial \vec{C}_{qnorm}}{\partial \vec{q}_1} &= \langle 2\vec{q}_1 \rangle & \frac{\partial \vec{C}_{qnorm}}{\partial \vec{q}_3} &= \langle 2\vec{q}_2 \rangle \\
\frac{\partial \vec{C}_{qnorm}}{\partial \vec{q}_2} &= \langle 2\vec{q}_2 \rangle & \frac{\partial \vec{C}_{qnorm}}{\partial \vec{q}_4} &= \langle 2\vec{q}_4 \rangle \quad (\text{G.13})
\end{aligned}$$

All other elements are zero.

- Partial Derivatives of Dynamic Pressure Constraint (\vec{C}_q)

$$\begin{aligned}
 \frac{\partial \vec{C}_q}{\partial \vec{R}_x} &= \frac{\partial \vec{q}}{\partial \vec{R}_x} & \frac{\partial C_q}{\partial \vec{T}} &= [0] \\
 \frac{\partial \vec{C}_q}{\partial \vec{R}_y} &= \frac{\partial \vec{q}}{\partial \vec{R}_y} & \frac{\partial C_q}{\partial \vec{q}_1} &= [0] \\
 \frac{\partial \vec{C}_q}{\partial \vec{R}_z} &= \frac{\partial \vec{q}}{\partial \vec{R}_z} & \frac{\partial C_q}{\partial \vec{q}_2} &= [0] \\
 \frac{\partial \vec{C}_q}{\partial \vec{V}_x} &= \frac{\partial \vec{q}}{\partial \vec{V}_x} & \frac{\partial C_q}{\partial \vec{q}_3} &= [0] \\
 \frac{\partial \vec{C}_q}{\partial \vec{V}_y} &= \frac{\partial \vec{q}}{\partial \vec{V}_y} & \frac{\partial C_q}{\partial \vec{q}_4} &= [0] \\
 \frac{\partial \vec{C}_q}{\partial \vec{V}_z} &= \frac{\partial \vec{q}}{\partial \vec{V}_z} & \frac{\partial C_q}{\partial \tau_f} &= [0] \\
 \frac{\partial \vec{C}_q}{\partial \vec{m}} &= [0]
 \end{aligned} \tag{G.14}$$

- Partial Derivatives of Sensed Acceleration Constraint (\vec{C}_a)

$$\begin{aligned}
 \frac{\partial \vec{C}_a}{\partial \vec{R}_x} &= -2 \left[\langle (\vec{a}_{sensed})_x \rangle \frac{\partial \vec{g}_x}{\partial \vec{R}_x} + \langle (\vec{a}_{sensed})_y \rangle \frac{\partial \vec{g}_y}{\partial \vec{R}_x} + \langle (\vec{a}_{sensed})_z \rangle \frac{\partial \vec{g}_z}{\partial \vec{R}_x} \right] \\
 \frac{\partial \vec{C}_a}{\partial \vec{R}_y} &= -2 \left[\langle (\vec{a}_{sensed})_x \rangle \frac{\partial \vec{g}_x}{\partial \vec{R}_y} + \langle (\vec{a}_{sensed})_y \rangle \frac{\partial \vec{g}_y}{\partial \vec{R}_y} + \langle (\vec{a}_{sensed})_z \rangle \frac{\partial \vec{g}_z}{\partial \vec{R}_y} \right] \\
 \frac{\partial \vec{C}_a}{\partial \vec{R}_z} &= -2 \left[\langle (\vec{a}_{sensed})_x \rangle \frac{\partial \vec{g}_x}{\partial \vec{R}_z} + \langle (\vec{a}_{sensed})_y \rangle \frac{\partial \vec{g}_y}{\partial \vec{R}_z} + \langle (\vec{a}_{sensed})_z \rangle \frac{\partial \vec{g}_z}{\partial \vec{R}_z} \right] \\
 \frac{\partial \vec{C}_a}{\partial \vec{V}_x} &= \langle 2(\vec{a}_{sensed})_x \rangle D_{NN} \\
 \frac{\partial \vec{C}_a}{\partial \vec{V}_y} &= \langle 2(\vec{a}_{sensed})_y \rangle D_{NN} \\
 \frac{\partial \vec{C}_a}{\partial \vec{V}_z} &= \langle 2(\vec{a}_{sensed})_z \rangle D_{NN} \\
 \frac{\partial \vec{C}_a}{\partial \vec{m}} &= [0] \\
 \frac{\partial \vec{C}_a}{\partial \vec{T}} &= [0] \\
 \frac{\partial \vec{C}_a}{\partial \vec{q}_1} &= [0] \\
 \frac{\partial \vec{C}_a}{\partial \vec{q}_2} &= [0] \\
 \frac{\partial \vec{C}_a}{\partial \vec{q}_3} &= [0] \\
 \frac{\partial \vec{C}_a}{\partial \vec{q}_4} &= [0]
 \end{aligned} \tag{G.15}$$

$$\frac{\partial \vec{C}_a}{\partial \tau_f} = \frac{-2}{\tau_f - \tau_0} \left[(\vec{a}_{sensed})_x (D_{NN} \vec{V}_x) + (\vec{a}_{sensed})_y (D_{NN} \vec{V}_y) + (\vec{a}_{sensed})_z (D_{NN} \vec{V}_z) \right]$$

- Partial Derivatives of Mass Change Constraint (\vec{C}_a)

$$\frac{\partial \vec{C}_{\Delta m}}{\partial \vec{m}} = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & -1 & 1 \end{bmatrix} \quad (G.16)$$

All other derivatives are zero. Note that this matrix is of size $n_{LGL} - 1$ by n_{LGL} since there are $n_{LGL} - 1$ mass change constraints but n_{LGL} masses (one mass for each LGL point).

D^2 -Method

- Partial Derivatives of Quaternion Normalization Constraint (\vec{C}_{qnorm})

The D^2 -method partial derivatives are the same as for the D -method.

- Partial Derivatives of Dynamic Pressure Constraint (\vec{C}_q)

$$\begin{aligned} \frac{\partial \vec{C}_q}{\partial \vec{R}_x} &= \frac{\partial \vec{q}}{\partial \vec{R}_x} & \frac{\partial \vec{C}_q}{\partial \vec{q}_1} &= [0] \\ \frac{\partial \vec{C}_q}{\partial \vec{R}_y} &= \frac{\partial \vec{q}}{\partial \vec{R}_y} & \frac{\partial \vec{C}_q}{\partial \vec{q}_2} &= [0] \\ \frac{\partial \vec{C}_q}{\partial \vec{R}_z} &= \frac{\partial \vec{q}}{\partial \vec{R}_z} & \frac{\partial \vec{C}_q}{\partial \vec{q}_3} &= [0] \\ \frac{\partial \vec{C}_q}{\partial \vec{m}} &= [0] & \frac{\partial \vec{C}_q}{\partial \vec{q}_4} &= [0] \\ \frac{\partial \vec{C}_q}{\partial \vec{T}} &= [0] & \frac{\partial \vec{C}_q}{\partial \tau_f} &= \left[\frac{\partial \vec{q}}{\partial \tau_f} \right] \end{aligned} \quad (G.17)$$

- Partial Derivatives of Sensed Acceleration Constraint (\vec{C}_a)

$$\frac{\partial \vec{C}_a}{\partial \vec{R}_x} = 2 \langle (\vec{a}_{sensed})_x \rangle \left[D_{NN}^2 - \frac{\partial \vec{g}_x}{\partial \vec{R}_x} \right] - 2 \langle (\vec{a}_{sensed})_y \rangle \frac{\partial \vec{g}_y}{\partial \vec{R}_x} - 2 \langle (\vec{a}_{sensed})_z \rangle \frac{\partial \vec{g}_z}{\partial \vec{R}_x}$$

$$\begin{aligned}
\frac{\partial \tilde{C}_a}{\partial R_y} &= 2 \langle (\tilde{a}_{sensed})_x \rangle \frac{\partial \tilde{g}_x}{\partial \tilde{R}_y} - 2 \langle (\tilde{a}_{sensed})_y \rangle \left[D_{NN}^2 - \frac{\partial \tilde{g}_y}{\partial \tilde{R}_y} \right] - 2 \langle (\tilde{a}_{sensed})_z \rangle \frac{\partial \tilde{g}_z}{\partial \tilde{R}_y} \\
\frac{\partial \tilde{C}_a}{\partial R_z} &= 2 \langle (\tilde{a}_{sensed})_x \rangle \frac{\partial \tilde{g}_x}{\partial \tilde{R}_z} - 2 \langle (\tilde{a}_{sensed})_y \rangle \frac{\partial \tilde{g}_y}{\partial \tilde{R}_z} - 2 \langle (\tilde{a}_{sensed})_z \rangle \left[D_{NN}^2 - \frac{\partial \tilde{g}_z}{\partial \tilde{R}_z} \right] \\
\frac{\partial \tilde{C}_a}{\partial \tilde{m}} &= [0] \\
\frac{\partial \tilde{C}_a}{\partial T} &= [0] \\
\frac{\partial \tilde{C}_a}{\partial \tilde{q}_1} &= [0] \\
\frac{\partial \tilde{C}_a}{\partial \tilde{q}_2} &= [0] \\
\frac{\partial \tilde{C}_a}{\partial \tilde{q}_3} &= [0] \\
\frac{\partial \tilde{C}_a}{\partial \tilde{q}_4} &= [0] \\
\frac{\partial \tilde{C}_a}{\partial \tau_f} &= \frac{-4}{(\tau_f - \tau_o)} \left[(\tilde{a}_{sensed})_x (D_{NN}^2 \tilde{R}_x) + (\tilde{a}_{sensed})_y (D_{NN}^2 \tilde{R}_y) + (\tilde{a}_{sensed})_z (D_{NN}^2 \tilde{R}_z) \right]
\end{aligned} \tag{G.18}$$

- **Partial Derivatives of Mass Change Constraint (\tilde{C}_a)**

The partial derivatives of the mass change constraint using the D^2 -method are the same as for the D -method.

G.3 Initial and Final Constraint Jacobian

Note that the initial and final constraints are scalar quantities, not vectors. This is because they only apply at the initial or final time point.

D-Method

Note that there are no initial constraints for the D -method.

- **Partial Derivatives of Final Altitude Constraint (C_{R_f})**

$$\begin{array}{ll}
\frac{\partial C_{R_f}}{\partial \tilde{R}_x} = \begin{bmatrix} 0 & 0 & \dots & 2R_{x_f} \end{bmatrix} & \frac{\partial C_{R_f}}{\partial \tilde{T}} = [0] \\
\frac{\partial C_{R_f}}{\partial \tilde{R}_y} = \begin{bmatrix} 0 & 0 & \dots & 2R_{y_f} \end{bmatrix} & \frac{\partial C_{R_f}}{\partial \tilde{q}_1} = [0] \\
\frac{\partial C_{R_f}}{\partial \tilde{R}_z} = \begin{bmatrix} 0 & 0 & \dots & 2R_{z_f} \end{bmatrix} & \frac{\partial C_{R_f}}{\partial \tilde{q}_2} = [0]
\end{array}$$

$$\begin{aligned}
\frac{\partial C_{R_f}}{\partial \vec{V}_x} &= [0] & \frac{\partial C_{R_f}}{\partial \vec{q}_3} &= [0] \\
\frac{\partial C_{R_f}}{\partial \vec{V}_y} &= [0] & \frac{\partial C_{R_f}}{\partial \vec{q}_4} &= [0] \\
\frac{\partial C_{R_f}}{\partial \vec{V}_z} &= [0] & \frac{\partial C_{R_f}}{\partial \tau_f} &= 0 \\
\frac{\partial C_{R_f}}{\partial \vec{m}} &= [0]
\end{aligned} \tag{G.19}$$

• Partial Derivatives of Final Velocity Constraint (C_{V_f})

$$\begin{aligned}
\frac{\partial C_{V_f}}{\partial \vec{R}_x} &= [0] & \frac{\partial C_{V_f}}{\partial \vec{T}} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{R}_y} &= [0] & \frac{\partial C_{V_f}}{\partial \vec{q}_1} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{R}_z} &= [0] & \frac{\partial C_{V_f}}{\partial \vec{q}_2} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{V}_x} &= \begin{bmatrix} 0 & 0 & \cdots & 2V_{x_f} \end{bmatrix} & \frac{\partial C_{V_f}}{\partial \vec{q}_3} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{V}_y} &= \begin{bmatrix} 0 & 0 & \cdots & 2V_{y_f} \end{bmatrix} & \frac{\partial C_{V_f}}{\partial \vec{q}_4} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{V}_z} &= \begin{bmatrix} 0 & 0 & \cdots & 2V_{z_f} \end{bmatrix} & \frac{\partial C_{V_f}}{\partial \tau_f} &= 0 \\
\frac{\partial C_{V_f}}{\partial \vec{m}} &= [0]
\end{aligned} \tag{G.20}$$

• Partial Derivatives of Final Flight Path Angle Constraint (C_{γ_f})

$$\begin{aligned}
\frac{\partial C_{\gamma_f}}{\partial \vec{R}_x} &= \begin{bmatrix} 0 & 0 & \cdots & V_{x_f} \end{bmatrix} & \frac{\partial C_{\gamma_f}}{\partial \vec{T}} &= [0] \\
\frac{\partial C_{\gamma_f}}{\partial \vec{R}_y} &= \begin{bmatrix} 0 & 0 & \cdots & V_{y_f} \end{bmatrix} & \frac{\partial C_{\gamma_f}}{\partial \vec{q}_1} &= [0] \\
\frac{\partial C_{\gamma_f}}{\partial \vec{R}_z} &= \begin{bmatrix} 0 & 0 & \cdots & V_{z_f} \end{bmatrix} & \frac{\partial C_{\gamma_f}}{\partial \vec{q}_2} &= [0] \\
\frac{\partial C_{\gamma_f}}{\partial \vec{V}_x} &= \begin{bmatrix} 0 & 0 & \cdots & R_{x_f} \end{bmatrix} & \frac{\partial C_{\gamma_f}}{\partial \vec{q}_3} &= [0] \\
\frac{\partial C_{\gamma_f}}{\partial \vec{V}_y} &= \begin{bmatrix} 0 & 0 & \cdots & R_{y_f} \end{bmatrix} & \frac{\partial C_{\gamma_f}}{\partial \vec{q}_4} &= [0]
\end{aligned} \tag{G.21}$$

$$\begin{aligned}\frac{\partial C_{\gamma_f}}{\partial \vec{V}_z} &= \begin{bmatrix} 0 & 0 & \cdots & R_{z_f} \end{bmatrix} & \frac{\partial C_{\gamma_f}}{\partial \tau_f} &= 0 \\ \frac{\partial C_{\gamma_f}}{\partial \vec{m}} &= [0]\end{aligned}$$

- Partial Derivatives of Final Inclination Constraint (C_{incl})

$$\begin{aligned}\frac{\partial C_{incl}}{\partial \vec{R}_x} &= \begin{bmatrix} 0 & 0 & \cdots & V_{y_f} \end{bmatrix} & \frac{\partial C_{incl}}{\partial \vec{T}} &= [0] \\ \frac{\partial C_{incl}}{\partial \vec{R}_y} &= \begin{bmatrix} 0 & 0 & \cdots & -V_{x_f} \end{bmatrix} & \frac{\partial C_{incl}}{\partial \vec{q}_1} &= [0] \\ \frac{\partial C_{incl}}{\partial \vec{R}_z} &= [0] & \frac{\partial C_{incl}}{\partial \vec{q}_2} &= [0] \\ \frac{\partial C_{incl}}{\partial \vec{V}_x} &= \begin{bmatrix} 0 & 0 & \cdots & -R_{y_f} \end{bmatrix} & \frac{\partial C_{incl}}{\partial \vec{q}_3} &= [0] \quad (\text{G.22}) \\ \frac{\partial C_{incl}}{\partial \vec{V}_y} &= \begin{bmatrix} 0 & 0 & \cdots & R_{x_f} \end{bmatrix} & \frac{\partial C_{incl}}{\partial \vec{q}_4} &= [0] \\ \frac{\partial C_{incl}}{\partial \vec{V}_z} &= [0] & \frac{\partial C_{incl}}{\partial \tau_f} &= 0 \\ \frac{\partial C_{incl}}{\partial \vec{m}} &= [0]\end{aligned}$$

D^2 -Method

- Partial Derivatives of Initial Velocity Constraints ($C_{V_{x_o}}$, $C_{V_{y_o}}$, $C_{V_{z_o}}$)

$$\begin{aligned}\frac{\partial \vec{C}_{V_{x_o}}}{\partial \vec{R}_x} &= [d_{NN_1}] & \frac{\partial \vec{C}_{V_{x_o}}}{\partial \tau_f} &= \frac{-d_{NN_1}}{(\tau_f - \tau_o)} \vec{R}_x \\ \frac{\partial \vec{C}_{V_{y_o}}}{\partial \vec{R}_y} &= [d_{NN_1}] & \frac{\partial \vec{C}_{V_{y_o}}}{\partial \tau_f} &= \frac{-d_{NN_1}}{(\tau_f - \tau_o)} \vec{R}_y \quad (\text{G.23}) \\ \frac{\partial \vec{C}_{V_{z_o}}}{\partial \vec{R}_z} &= [d_{NN_1}] & \frac{\partial \vec{C}_{V_{z_o}}}{\partial \tau_f} &= \frac{-d_{NN_1}}{(\tau_f - \tau_o)} \vec{R}_z\end{aligned}$$

where d_{NN_1} is the first row of the D_{NN} matrix corresponding to the initial time phase. Note that it is equivalent to a row vector.

All other partial derivatives are zero.

- Partial Derivatives of Final Altitude Constraint (C_{R_f})

$$\begin{aligned}
 \frac{\partial C_{R_f}}{\partial \vec{R}_x} &= \begin{bmatrix} 0 & 0 & \cdots & 2R_{x_f} \end{bmatrix} & \frac{\partial C_{R_f}}{\partial \vec{q}_1} &= [0] \\
 \frac{\partial C_{R_f}}{\partial \vec{R}_y} &= \begin{bmatrix} 0 & 0 & \cdots & 2R_{y_f} \end{bmatrix} & \frac{\partial C_{R_f}}{\partial \vec{q}_2} &= [0] \\
 \frac{\partial C_{R_f}}{\partial \vec{R}_z} &= \begin{bmatrix} 0 & 0 & \cdots & 2R_{z_f} \end{bmatrix} & \frac{\partial C_{R_f}}{\partial \vec{q}_3} &= [0] \\
 \frac{\partial C_{R_f}}{\partial \vec{m}} &= [0] & \frac{\partial C_{R_f}}{\partial \vec{q}_4} &= [0] \\
 \frac{\partial C_{R_f}}{\partial \vec{T}} &= [0] & \frac{\partial C_{R_f}}{\partial \tau_f} &= 0
 \end{aligned} \tag{G.24}$$

- Partial Derivatives of Final Velocity Constraint (C_{V_f})

$$\begin{aligned}
 \frac{\partial C_{V_f}}{\partial \vec{R}_x} &= [2V_{x_f} d_{NN_f}] & \frac{\partial C_{V_f}}{\partial \vec{q}_1} &= [0] \\
 \frac{\partial C_{V_f}}{\partial \vec{R}_y} &= [2V_{y_f} d_{NN_f}] & \frac{\partial C_{V_f}}{\partial \vec{q}_2} &= [0] \\
 \frac{\partial C_{V_f}}{\partial \vec{R}_z} &= [2V_{z_f} d_{NN_f}] & \frac{\partial C_{V_f}}{\partial \vec{q}_3} &= [0] \\
 \frac{\partial C_{V_f}}{\partial \vec{m}} &= [0] & \frac{\partial C_{V_f}}{\partial \vec{q}_4} &= [0] \\
 \frac{\partial C_{V_f}}{\partial \vec{T}} &= [0] & \frac{\partial C_{V_f}}{\partial \tau_f} &= \frac{-2}{(\tau_f - \tau_o)} (V_{x_f} d_{NN_f} \vec{R}_x + V_{y_f} d_{NN_f} \vec{R}_y + V_{z_f} d_{NN_f} \vec{R}_z)
 \end{aligned} \tag{G.25}$$

where d_{NN_f} is the last row of the D_{NN} matrix corresponding to the last time phase. Note that it is equivalent to a row vector.

- Partial Derivatives of Final Flight Path Angle Constraint (C_{γ_f})

$$\begin{aligned}
 \frac{\partial C_{\gamma_f}}{\partial \vec{R}_x} &= \begin{bmatrix} R_{x_f} d_{NN_{f,1}} & R_{x_f} d_{NN_{f,2}} & \cdots & R_{x_f} d_{NN_{f,n-1}} & V_{x_f} + R_{x_f} d_{NN_{f,n}} \end{bmatrix} \\
 \frac{\partial C_{\gamma_f}}{\partial \vec{R}_y} &= \begin{bmatrix} R_{y_f} d_{NN_{f,1}} & R_{y_f} d_{NN_{f,2}} & \cdots & R_{y_f} d_{NN_{f,n-1}} & V_{y_f} + R_{y_f} d_{NN_{f,n}} \end{bmatrix} \\
 \frac{\partial C_{\gamma_f}}{\partial \vec{R}_z} &= \begin{bmatrix} R_{z_f} d_{NN_{f,1}} & R_{z_f} d_{NN_{f,2}} & \cdots & R_{z_f} d_{NN_{f,n-1}} & V_{z_f} + R_{z_f} d_{NN_{f,n}} \end{bmatrix} \\
 \frac{\partial C_{\gamma_f}}{\partial \vec{m}} &= [0] \\
 \frac{\partial C_{\gamma_f}}{\partial \vec{T}} &= [0]
 \end{aligned} \tag{G.26}$$

$$\begin{aligned}
\frac{\partial C_{V_f}}{\partial \vec{q}_1} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{q}_2} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{q}_3} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{q}_4} &= [0] \\
\frac{\partial C_{V_f}}{\partial \tau_f} &= \frac{-1}{(\tau_f - \tau_o)} (R_{x_f} d_{NN_f} \vec{R}_x + R_{y_f} d_{NN_f} \vec{R}_y + R_{z_f} d_{NN_f} \vec{R}_z)
\end{aligned}$$

where $d_{NN_f,i}$ is the i th element of the last row of the D_{NN} matrix corresponding to the last time phase.

- Partial Derivatives of Final Inclination Constraint (C_{incl})

$$\begin{aligned}
\frac{\partial C_{V_f}}{\partial \vec{R}_x} &= \left[\begin{array}{cccc} -R_{y_f} d_{NN_f,1} & -R_{y_f} d_{NN_f,2} & \cdots & -R_{y_f} d_{NN_f,n-1} & V_{y_f} - R_{y_f} d_{NN_f,n} \end{array} \right] \\
\frac{\partial C_{V_f}}{\partial \vec{R}_y} &= \left[\begin{array}{cccc} R_{x_f} d_{NN_f,1} & R_{x_f} d_{NN_f,2} & \cdots & R_{x_f} d_{NN_f,n-1} & -V_{x_f} + R_{x_f} d_{NN_f,n} \end{array} \right] \\
\frac{\partial C_{V_f}}{\partial \vec{R}_z} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{m}} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{T}} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{q}_1} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{q}_2} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{q}_3} &= [0] \\
\frac{\partial C_{V_f}}{\partial \vec{q}_4} &= [0] \\
\frac{\partial C_{V_f}}{\partial \tau_f} &= \frac{-1}{(\tau_f - \tau_o)} (R_{x_f} d_{NN_f} \vec{R}_y - R_{y_f} d_{NN_f} \vec{R}_x)
\end{aligned} \tag{G.27}$$

where $d_{NN_f,i}$ is the i th element of the last row of the D_{NN} matrix corresponding to the last time phase.

G.4 Staging Constraint Jacobian

D-Method

The Jacobian corresponding to equations 6.39 through 6.44 is very simple. The Jacobian of each equation consists of a row vector of mostly zeros. Each row vector will have only two non-zero elements (1 and -1), corresponding to the states or times that are being knotted.

D^2 -Method

The Jacobian for the D^2 -method is slightly more complicated than for the D -method. The Jacobians of the position and time knotting constraints are the same as for the D -method. However, the velocity knotting constraints are different. The derivatives of the velocities with respect to the positions are row vectors. Thus, the 1 and -1 elements are replaced by row vectors corresponding to the appropriate position vectors. The partial derivatives of the initial and final velocities in a time phase are:

$$\begin{aligned} \frac{\partial V_{x_1}}{\partial \vec{R}_x} &= [d_{NN_1}] & \frac{\partial V_{x_{n_{LGL}}}}{\partial \vec{R}_x} &= [d_{NN_{n_{LGL}}}] \\ \frac{\partial V_{y_1}}{\partial \vec{R}_y} &= [d_{NN_1}] & \frac{\partial V_{y_{n_{LGL}}}}{\partial \vec{R}_y} &= [d_{NN_{n_{LGL}}}] \\ \frac{\partial V_{z_1}}{\partial \vec{R}_z} &= [d_{NN_1}] & \frac{\partial V_{z_{n_{LGL}}}}{\partial \vec{R}_z} &= [d_{NN_{n_{LGL}}}] \end{aligned} \quad (\text{G.18})$$

where d_{NN_1} and $d_{NN_{n_{LGL}}}$ are the first and last rows, respectively, of the D_{NN} matrix corresponding to the proper time phase.

G.5 Partial Derivatives

G.5.1 Partial Derivatives of D^2 -Method Velocity

The partial derivatives of the D^2 -method velocity defined in section 6.6.1 are derived in this Appendix. The partial derivatives of the velocities are:

$$\frac{\partial \vec{V}_x}{\partial \vec{R}_x} = D_{NN} \quad \frac{\partial \vec{V}_x}{\partial \tau_f} = \frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{R}_x$$

$$\begin{aligned}\frac{\partial \vec{V}_y}{\partial \vec{R}_y} &= D_{NN} & \frac{\partial \vec{V}_y}{\partial \tau_f} &= \frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{R}_y \\ \frac{\partial \vec{V}_z}{\partial \vec{R}_z} &= D_{NN} & \frac{\partial \vec{V}_z}{\partial \tau_f} &= \frac{-D_{NN}}{(\tau_f - \tau_o)} \vec{R}_z\end{aligned}\quad (\text{G.29})$$

The partial derivatives of the velocity magnitude are:

$$\begin{aligned}\frac{\partial \vec{V}_{mag}}{\partial \vec{R}_x} &= \left\langle \frac{\vec{V}_x}{\vec{V}_{mag}} \right\rangle \frac{\partial \vec{V}_x}{\partial \vec{R}_x} \\ \frac{\partial \vec{V}_{mag}}{\partial \vec{R}_y} &= \left\langle \frac{\vec{V}_y}{\vec{V}_{mag}} \right\rangle \frac{\partial \vec{V}_y}{\partial \vec{R}_y} \\ \frac{\partial \vec{V}_{mag}}{\partial \vec{R}_z} &= \left\langle \frac{\vec{V}_z}{\vec{V}_{mag}} \right\rangle \frac{\partial \vec{V}_z}{\partial \vec{R}_z} \\ \frac{\partial \vec{V}_{mag}}{\partial \tau_f} &= \frac{1}{\vec{V}_{mag}} \left[\frac{\partial \vec{V}_x}{\partial \tau_f} + \frac{\partial \vec{V}_y}{\partial \tau_f} + \frac{\partial \vec{V}_z}{\partial \tau_f} \right]\end{aligned}\quad (\text{G.30})$$

All other partial derivatives are zero.

G.5.2 Partial Derivatives of Q Attitude Matrix

The partial derivatives of the Q attitude matrix defined in section 6.6.2 are derived in this Appendix. The partial derivatives of the Q attitude matrix at one LGL point are:

$$\frac{\partial Q}{\partial q_1} = \begin{bmatrix} 2q_1 & 2q_2 & 2q_3 \\ 2q_2 & -2q_1 & 2q_4 \\ 2q_3 & -2q_4 & -2q_1 \end{bmatrix} \quad (\text{G.31})$$

$$\frac{\partial Q}{\partial q_2} = \begin{bmatrix} -2q_2 & 2q_1 & -2q_4 \\ 2q_1 & 2q_2 & 2q_3 \\ 2q_4 & 2q_3 & -2q_2 \end{bmatrix} \quad (\text{G.32})$$

$$\frac{\partial Q}{\partial q_3} = \begin{bmatrix} -2q_3 & 2q_4 & 2q_1 \\ -2q_4 & -2q_3 & 2q_2 \\ 2q_1 & 2q_2 & 2q_3 \end{bmatrix} \quad (\text{G.33})$$

$$\frac{\partial Q}{\partial q_4} = \begin{bmatrix} 2q_4 & 2q_3 & -2q_2 \\ -2q_3 & 2q_4 & 2q_1 \\ 2q_2 & -2q_1 & 2q_4 \end{bmatrix} \quad (\text{G.34})$$

Since there are a set of four quaternion elements at each LGL point, there is also an attitude matrix at each LGL point. This means that the elements in equation 6.60 are actually vectors. The partial derivatives of the attitude matrix elements are:

$$\begin{aligned} \frac{\partial \tilde{Q}_{11}}{\partial \vec{q}_1} &= \langle 2\vec{q}_1 \rangle & \frac{\partial \tilde{Q}_{12}}{\partial \vec{q}_1} &= \langle 2\vec{q}_2 \rangle & \frac{\partial \tilde{Q}_{13}}{\partial \vec{q}_1} &= \langle 2\vec{q}_3 \rangle \\ \frac{\partial \tilde{Q}_{21}}{\partial \vec{q}_1} &= \langle 2\vec{q}_2 \rangle & \frac{\partial \tilde{Q}_{22}}{\partial \vec{q}_1} &= \langle -2\vec{q}_1 \rangle & \frac{\partial \tilde{Q}_{23}}{\partial \vec{q}_1} &= \langle 2\vec{q}_4 \rangle \\ \frac{\partial \tilde{Q}_{31}}{\partial \vec{q}_1} &= \langle 2\vec{q}_3 \rangle & \frac{\partial \tilde{Q}_{32}}{\partial \vec{q}_1} &= \langle -2\vec{q}_4 \rangle & \frac{\partial \tilde{Q}_{33}}{\partial \vec{q}_1} &= \langle -2\vec{q}_1 \rangle \end{aligned} \quad (\text{G.35})$$

$$\begin{aligned} \frac{\partial \tilde{Q}_{11}}{\partial \vec{q}_2} &= \langle -2\vec{q}_2 \rangle & \frac{\partial \tilde{Q}_{12}}{\partial \vec{q}_2} &= \langle 2\vec{q}_1 \rangle & \frac{\partial \tilde{Q}_{13}}{\partial \vec{q}_2} &= \langle -2\vec{q}_4 \rangle \\ \frac{\partial \tilde{Q}_{21}}{\partial \vec{q}_2} &= \langle 2\vec{q}_1 \rangle & \frac{\partial \tilde{Q}_{22}}{\partial \vec{q}_2} &= \langle 2\vec{q}_2 \rangle & \frac{\partial \tilde{Q}_{23}}{\partial \vec{q}_2} &= \langle 2\vec{q}_3 \rangle \\ \frac{\partial \tilde{Q}_{31}}{\partial \vec{q}_2} &= \langle 2\vec{q}_4 \rangle & \frac{\partial \tilde{Q}_{32}}{\partial \vec{q}_2} &= \langle 2\vec{q}_3 \rangle & \frac{\partial \tilde{Q}_{33}}{\partial \vec{q}_2} &= \langle -2\vec{q}_2 \rangle \end{aligned} \quad (\text{G.36})$$

$$\begin{aligned} \frac{\partial \tilde{Q}_{11}}{\partial \vec{q}_3} &= \langle -2\vec{q}_3 \rangle & \frac{\partial \tilde{Q}_{12}}{\partial \vec{q}_3} &= \langle 2\vec{q}_4 \rangle & \frac{\partial \tilde{Q}_{13}}{\partial \vec{q}_3} &= \langle 2\vec{q}_1 \rangle \\ \frac{\partial \tilde{Q}_{21}}{\partial \vec{q}_3} &= \langle -2\vec{q}_4 \rangle & \frac{\partial \tilde{Q}_{22}}{\partial \vec{q}_3} &= \langle -2\vec{q}_3 \rangle & \frac{\partial \tilde{Q}_{23}}{\partial \vec{q}_3} &= \langle 2\vec{q}_2 \rangle \\ \frac{\partial \tilde{Q}_{31}}{\partial \vec{q}_3} &= \langle 2\vec{q}_1 \rangle & \frac{\partial \tilde{Q}_{32}}{\partial \vec{q}_3} &= \langle 2\vec{q}_2 \rangle & \frac{\partial \tilde{Q}_{33}}{\partial \vec{q}_3} &= \langle 2\vec{q}_3 \rangle \end{aligned} \quad (\text{G.37})$$

$$\begin{aligned} \frac{\partial \tilde{Q}_{11}}{\partial \vec{q}_4} &= \langle 2\vec{q}_4 \rangle & \frac{\partial \tilde{Q}_{12}}{\partial \vec{q}_4} &= \langle 2\vec{q}_3 \rangle & \frac{\partial \tilde{Q}_{13}}{\partial \vec{q}_4} &= \langle -2\vec{q}_2 \rangle \\ \frac{\partial \tilde{Q}_{21}}{\partial \vec{q}_4} &= \langle -2\vec{q}_3 \rangle & \frac{\partial \tilde{Q}_{22}}{\partial \vec{q}_4} &= \langle 2\vec{q}_4 \rangle & \frac{\partial \tilde{Q}_{23}}{\partial \vec{q}_4} &= \langle 2\vec{q}_1 \rangle \\ \frac{\partial \tilde{Q}_{31}}{\partial \vec{q}_4} &= \langle 2\vec{q}_2 \rangle & \frac{\partial \tilde{Q}_{32}}{\partial \vec{q}_4} &= \langle -2\vec{q}_1 \rangle & \frac{\partial \tilde{Q}_{33}}{\partial \vec{q}_4} &= \langle 2\vec{q}_4 \rangle \end{aligned} \quad (\text{G.38})$$

All other partial derivatives are zero.

G.5.3 Partial Derivatives of Thrust

The partial derivatives of the thrust defined in section 6.6.3 are derived in this Appendix. Note that they are the same for both the *D*- and *D*²-methods.

$$\begin{aligned}
\frac{\partial \vec{T}_x}{\partial \vec{T}} &= \langle \vec{Q}_{11} u_{T_x} + \vec{Q}_{12} u_{T_y} + \vec{Q}_{13} u_{T_z} \rangle \\
\frac{\partial \vec{T}_y}{\partial \vec{T}} &= \langle \vec{Q}_{21} u_{T_x} + \vec{Q}_{22} u_{T_y} + \vec{Q}_{23} u_{T_z} \rangle \\
\frac{\partial \vec{T}_z}{\partial \vec{T}} &= \langle \vec{Q}_{31} u_{T_x} + \vec{Q}_{32} u_{T_y} + \vec{Q}_{33} u_{T_z} \rangle
\end{aligned} \tag{G.39}$$

$$\begin{aligned}
\frac{\partial \vec{T}_x}{\partial \vec{q}_1} &= \langle \vec{T} \rangle \left[\langle u_{T_x} \rangle \frac{\vec{Q}_{11}}{\partial \vec{q}_1} + \langle u_{T_y} \rangle \frac{\vec{Q}_{12}}{\partial \vec{q}_1} + \langle u_{T_z} \rangle \frac{\vec{Q}_{13}}{\partial \vec{q}_1} \right] \\
\frac{\partial \vec{T}_y}{\partial \vec{q}_1} &= \langle \vec{T} \rangle \left[\langle u_{T_x} \rangle \frac{\vec{Q}_{21}}{\partial \vec{q}_1} + \langle u_{T_y} \rangle \frac{\vec{Q}_{22}}{\partial \vec{q}_1} + \langle u_{T_z} \rangle \frac{\vec{Q}_{23}}{\partial \vec{q}_1} \right] \\
\frac{\partial \vec{T}_z}{\partial \vec{q}_1} &= \langle \vec{T} \rangle \left[\langle u_{T_x} \rangle \frac{\vec{Q}_{31}}{\partial \vec{q}_1} + \langle u_{T_y} \rangle \frac{\vec{Q}_{32}}{\partial \vec{q}_1} + \langle u_{T_z} \rangle \frac{\vec{Q}_{33}}{\partial \vec{q}_1} \right]
\end{aligned} \tag{G.40}$$

$$\begin{aligned}
\frac{\partial \vec{T}_x}{\partial \vec{q}_2} &= \langle \vec{T} \rangle \left[\langle u_{T_x} \rangle \frac{\vec{Q}_{11}}{\partial \vec{q}_2} + \langle u_{T_y} \rangle \frac{\vec{Q}_{12}}{\partial \vec{q}_2} + \langle u_{T_z} \rangle \frac{\vec{Q}_{13}}{\partial \vec{q}_2} \right] \\
\frac{\partial \vec{T}_y}{\partial \vec{q}_2} &= \langle \vec{T} \rangle \left[\langle u_{T_x} \rangle \frac{\vec{Q}_{21}}{\partial \vec{q}_2} + \langle u_{T_y} \rangle \frac{\vec{Q}_{22}}{\partial \vec{q}_2} + \langle u_{T_z} \rangle \frac{\vec{Q}_{23}}{\partial \vec{q}_2} \right] \\
\frac{\partial \vec{T}_z}{\partial \vec{q}_2} &= \langle \vec{T} \rangle \left[\langle u_{T_x} \rangle \frac{\vec{Q}_{31}}{\partial \vec{q}_2} + \langle u_{T_y} \rangle \frac{\vec{Q}_{32}}{\partial \vec{q}_2} + \langle u_{T_z} \rangle \frac{\vec{Q}_{33}}{\partial \vec{q}_2} \right]
\end{aligned} \tag{G.41}$$

$$\begin{aligned}
\frac{\partial \vec{T}_x}{\partial \vec{q}_3} &= \langle \vec{T} \rangle \left[\langle u_{T_x} \rangle \frac{\vec{Q}_{11}}{\partial \vec{q}_3} + \langle u_{T_y} \rangle \frac{\vec{Q}_{12}}{\partial \vec{q}_3} + \langle u_{T_z} \rangle \frac{\vec{Q}_{13}}{\partial \vec{q}_4} \right] \\
\frac{\partial \vec{T}_y}{\partial \vec{q}_3} &= \langle \vec{T} \rangle \left[\langle u_{T_x} \rangle \frac{\vec{Q}_{21}}{\partial \vec{q}_3} + \langle u_{T_y} \rangle \frac{\vec{Q}_{22}}{\partial \vec{q}_3} + \langle u_{T_z} \rangle \frac{\vec{Q}_{23}}{\partial \vec{q}_4} \right] \\
\frac{\partial \vec{T}_z}{\partial \vec{q}_3} &= \langle \vec{T} \rangle \left[\langle u_{T_x} \rangle \frac{\vec{Q}_{31}}{\partial \vec{q}_3} + \langle u_{T_y} \rangle \frac{\vec{Q}_{32}}{\partial \vec{q}_3} + \langle u_{T_z} \rangle \frac{\vec{Q}_{33}}{\partial \vec{q}_4} \right]
\end{aligned} \tag{G.42}$$

$$\begin{aligned}
\frac{\partial \vec{T}_x}{\partial \vec{q}_4} &= \langle \vec{T} \rangle \left[\langle u_{T_x} \rangle \frac{\vec{Q}_{11}}{\partial \vec{q}_4} + \langle u_{T_y} \rangle \frac{\vec{Q}_{12}}{\partial \vec{q}_4} + \langle u_{T_z} \rangle \frac{\vec{Q}_{13}}{\partial \vec{q}_4} \right] \\
\frac{\partial \vec{T}_y}{\partial \vec{q}_4} &= \langle \vec{T} \rangle \left[\langle u_{T_x} \rangle \frac{\vec{Q}_{21}}{\partial \vec{q}_4} + \langle u_{T_y} \rangle \frac{\vec{Q}_{22}}{\partial \vec{q}_4} + \langle u_{T_z} \rangle \frac{\vec{Q}_{23}}{\partial \vec{q}_4} \right] \\
\frac{\partial \vec{T}_z}{\partial \vec{q}_4} &= \langle \vec{T} \rangle \left[\langle u_{T_x} \rangle \frac{\vec{Q}_{31}}{\partial \vec{q}_4} + \langle u_{T_y} \rangle \frac{\vec{Q}_{32}}{\partial \vec{q}_4} + \langle u_{T_z} \rangle \frac{\vec{Q}_{33}}{\partial \vec{q}_4} \right]
\end{aligned} \tag{G.43}$$

All other partial derivatives are zero.

G.5.4 Partial Derivatives of Gravity

The partial derivatives of the thrust defined in section 6.6.4 are derived in this Appendix. The partial derivatives of the gravity are the same for the D - and D^2 -methods.

- Partial Derivatives of \vec{g}_x

$$\begin{aligned}
\frac{\partial \vec{g}_x}{\partial \vec{R}_x} &= \left\langle \frac{-\mu}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{3/2}} + \frac{3\mu \vec{R}_x^2}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{5/2}} \right\rangle \\
\frac{\partial \vec{g}_x}{\partial \vec{R}_y} &= \left\langle \frac{3\mu \vec{R}_x \vec{R}_y}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{5/2}} \right\rangle \\
\frac{\partial \vec{g}_x}{\partial \vec{R}_z} &= \left\langle \frac{3\mu \vec{R}_x \vec{R}_z}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{5/2}} \right\rangle
\end{aligned} \tag{G.44}$$

- Partial Derivatives of \vec{g}_y

$$\begin{aligned}
\frac{\partial \vec{g}_y}{\partial \vec{R}_x} &= \left\langle \frac{3\mu \vec{R}_x \vec{R}_y}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{5/2}} \right\rangle \\
\frac{\partial \vec{g}_y}{\partial \vec{R}_y} &= \left\langle \frac{-\mu}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{3/2}} + \frac{3\mu \vec{R}_y^2}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{5/2}} \right\rangle \\
\frac{\partial \vec{g}_y}{\partial \vec{R}_z} &= \left\langle \frac{3\mu \vec{R}_y \vec{R}_z}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{5/2}} \right\rangle
\end{aligned} \tag{G.45}$$

- Partial Derivatives of \vec{g}_z

$$\begin{aligned}
 \frac{\partial \vec{g}_z}{\partial \vec{R}_x} &= \left\langle \frac{3\mu \vec{R}_x \vec{R}_z}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{5/2}} \right\rangle \\
 \frac{\partial \vec{g}_z}{\partial \vec{R}_y} &= \left\langle \frac{3\mu \vec{R}_y \vec{R}_z}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{5/2}} \right\rangle \\
 \frac{\partial \vec{g}_z}{\partial \vec{R}_z} &= \left\langle \frac{-\mu}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{3/2}} + \frac{3\mu \vec{R}_z^2}{(\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2)^{5/2}} \right\rangle
 \end{aligned} \tag{G.46}$$

All other partial derivatives are zero.

G.5.5 Partial Derivatives of Aerodynamic Force

The partial derivatives of the aerodynamic force defined in section 6.6.5 are derived in this Appendix. The partial derivatives for the D - and D^2 -methods are slightly different. In particular, note that the aerodynamic forces are dependent on the final time with the D^2 -method. This is because the velocities are defined in terms of the positions and the D_{NN} matrix, which has a dependence on the final time.

- D -Method Partial Derivatives

The form of the partial derivatives follows ($i=x,y,z$):

$$\begin{array}{ll}
 \frac{\partial \vec{A}_i}{\partial \vec{R}_x} = \left[\left\langle \vec{C}_{I_i} \right\rangle \frac{\partial \vec{q}}{\partial \vec{R}_x} + \langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{R}_x} \right] A_{ref} & \frac{\partial \vec{A}_i}{\partial \vec{T}} = [0] \\
 \frac{\partial \vec{A}_i}{\partial \vec{R}_y} = \left[\left\langle \vec{C}_{I_i} \right\rangle \frac{\partial \vec{q}}{\partial \vec{R}_y} + \langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{R}_y} \right] A_{ref} & \frac{\partial \vec{A}_i}{\partial \vec{q}_1} = \left[\langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{q}_1} \right] A_{ref} \\
 \frac{\partial \vec{A}_i}{\partial \vec{R}_z} = \left[\left\langle \vec{C}_{I_i} \right\rangle \frac{\partial \vec{q}}{\partial \vec{R}_z} + \langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{R}_z} \right] A_{ref} & \frac{\partial \vec{A}_i}{\partial \vec{q}_2} = \left[\langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{q}_2} \right] A_{ref} \\
 \frac{\partial \vec{A}_i}{\partial \vec{V}_x} = \left[\left\langle \vec{C}_{I_i} \right\rangle \frac{\partial \vec{q}}{\partial \vec{V}_x} + \langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{V}_x} \right] A_{ref} & \frac{\partial \vec{A}_i}{\partial \vec{q}_3} = \left[\langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{q}_3} \right] A_{ref} \\
 \frac{\partial \vec{A}_i}{\partial \vec{V}_y} = \left[\left\langle \vec{C}_{I_i} \right\rangle \frac{\partial \vec{q}}{\partial \vec{V}_y} + \langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{V}_y} \right] A_{ref} & \frac{\partial \vec{A}_i}{\partial \vec{q}_4} = \left[\langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{q}_4} \right] A_{ref}
 \end{array} \tag{G.47}$$

$$\begin{aligned}\frac{\partial \vec{A}_i}{\partial \vec{V}_z} &= \left[\langle \vec{C}_{I_i} \rangle \frac{\partial \vec{q}}{\partial \vec{V}_z} + \langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{V}_z} \right] A_{ref} & \frac{\partial \vec{A}_i}{\partial \tau_f} &= [0] \\ \frac{\partial \vec{A}_i}{\partial \vec{m}} &= [0]\end{aligned}$$

- **D^2 -Method Partial Derivatives**

The form of the partial derivatives follows ($i=x,y,z$):

$$\begin{aligned}\frac{\partial \vec{A}_i}{\partial \vec{R}_x} &= \left[\langle \vec{C}_{I_i} \rangle \frac{\partial \vec{q}}{\partial \vec{R}_x} + \langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{R}_x} \right] A_{ref} & \frac{\partial \vec{A}_i}{\partial \vec{q}_1} &= \left[\langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{q}_1} \right] A_{ref} \\ \frac{\partial \vec{A}_i}{\partial \vec{R}_y} &= \left[\langle \vec{C}_{I_i} \rangle \frac{\partial \vec{q}}{\partial \vec{R}_y} + \langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{R}_y} \right] A_{ref} & \frac{\partial \vec{A}_i}{\partial \vec{q}_2} &= \left[\langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{q}_2} \right] A_{ref} \\ \frac{\partial \vec{A}_i}{\partial \vec{R}_z} &= \left[\langle \vec{C}_{I_i} \rangle \frac{\partial \vec{q}}{\partial \vec{R}_z} + \langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{R}_z} \right] A_{ref} & \frac{\partial \vec{A}_i}{\partial \vec{q}_3} &= \left[\langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{q}_3} \right] A_{ref} \quad (\text{G.48}) \\ \frac{\partial \vec{A}_i}{\partial \vec{m}} &= [0] & \frac{\partial \vec{A}_i}{\partial \vec{q}_4} &= \left[\langle \vec{q} \rangle \frac{\partial \vec{C}_{I_i}}{\partial \vec{q}_4} \right] A_{ref} \\ \frac{\partial \vec{A}_i}{\partial \vec{T}} &= [0] & \frac{\partial \vec{A}_i}{\partial \tau_f} &= \left[\vec{C}_{I_i} \frac{\partial \vec{q}}{\partial \tau_f} + \vec{q} \frac{\partial \vec{C}_{I_i}}{\partial \tau_f} \right]\end{aligned}$$

G.5.6 Partial Derivatives of Dynamic Pressure

The partial derivatives of the dynamic pressure defined in section 6.6.6 are derived in this Appendix. The partial derivatives for the D - and D^2 -methods are slightly different.

- **D -Method Partial Derivatives**

$$\begin{aligned}\frac{\partial \vec{q}}{\partial \vec{R}_x} &= \left\langle \frac{1}{2} \vec{V}_{rel}^2 \right\rangle \frac{\partial \vec{p}}{\partial \vec{R}_x} + \left\langle \vec{\rho} \vec{V}_{rel} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial \vec{R}_x} & \frac{\partial \vec{q}}{\partial \vec{V}_x} &= \left\langle \vec{\rho} \vec{V}_{rel} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial \vec{V}_x} \\ \frac{\partial \vec{q}}{\partial \vec{R}_y} &= \left\langle \frac{1}{2} \vec{V}_{rel}^2 \right\rangle \frac{\partial \vec{p}}{\partial \vec{R}_y} + \left\langle \vec{\rho} \vec{V}_{rel} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial \vec{R}_y} & \frac{\partial \vec{q}}{\partial \vec{V}_y} &= \left\langle \vec{\rho} \vec{V}_{rel} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial \vec{V}_y} \quad (\text{G.49}) \\ \frac{\partial \vec{q}}{\partial \vec{R}_z} &= \left\langle \frac{1}{2} \vec{V}_{rel}^2 \right\rangle \frac{\partial \vec{p}}{\partial \vec{R}_z} + \left\langle \vec{\rho} \vec{V}_{rel} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial \vec{R}_z} & \frac{\partial \vec{q}}{\partial \vec{V}_z} &= \left\langle \vec{\rho} \vec{V}_{rel} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial \vec{V}_z}\end{aligned}$$

- **D^2 -Method Partial Derivatives**

$$\begin{aligned}
 \frac{\partial \vec{q}}{\partial R_x} &= \left\langle \frac{1}{2} \vec{V}_{rel}^2 \right\rangle \frac{\partial \vec{\rho}}{\partial \vec{R}_x} + \left\langle \vec{\rho} \vec{V}_{rel} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial \vec{R}_x} \\
 \frac{\partial \vec{q}}{\partial R_y} &= \left\langle \frac{1}{2} \vec{V}_{rel}^2 \right\rangle \frac{\partial \vec{\rho}}{\partial \vec{R}_y} + \left\langle \vec{\rho} \vec{V}_{rel} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial \vec{R}_y} \\
 \frac{\partial \vec{q}}{\partial R_z} &= \left\langle \frac{1}{2} \vec{V}_{rel}^2 \right\rangle \frac{\partial \vec{\rho}}{\partial \vec{R}_z} + \left\langle \vec{\rho} \vec{V}_{rel} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial \vec{R}_z} \\
 \frac{\partial \vec{q}}{\partial \tau_f} &= \left[\vec{\rho} \vec{V}_{rel} \frac{\partial \vec{V}_{rel}}{\partial \tau_f} \right]
 \end{aligned} \tag{G.50}$$

All other derivatives are zero.

G.5.7 Partial Derivatives of Relative Velocity Components and Magnitude

The partial derivatives of the relative velocity components and magnitude defined in section 6.6.7 are derived in this Appendix. Note that the partial derivatives for the D - and D^2 -methods are different.

Partial Derivatives of Relative Velocity Components

- **D -Method Partial Derivatives**

$$\begin{array}{lll}
 \frac{\partial \vec{V}_{rel,x}}{\partial R_x} = [0] & \frac{\partial \vec{V}_{rel,x}}{\partial R_y} = \langle \Omega_z \rangle & \frac{\partial \vec{V}_{rel,x}}{\partial R_z} = \langle -\Omega_y \rangle \\
 \frac{\partial \vec{V}_{rel,y}}{\partial R_x} = \langle -\Omega_z \rangle & \frac{\partial \vec{V}_{rel,y}}{\partial R_y} = [0] & \frac{\partial \vec{V}_{rel,y}}{\partial R_z} = \langle \Omega_x \rangle \\
 \frac{\partial \vec{V}_{rel,z}}{\partial R_x} = \langle \Omega_y \rangle & \frac{\partial \vec{V}_{rel,z}}{\partial R_y} = \langle -\Omega_x \rangle & \frac{\partial \vec{V}_{rel,z}}{\partial R_z} = [0] \\
 \frac{\partial \vec{V}_{rel,x}}{\partial \vec{V}_x} = \langle 1 \rangle & \frac{\partial \vec{V}_{rel,x}}{\partial \vec{V}_y} = [0] & \frac{\partial \vec{V}_{rel,x}}{\partial \vec{V}_z} = [0] \\
 \frac{\partial \vec{V}_{rel,y}}{\partial \vec{V}_x} = [0] & \frac{\partial \vec{V}_{rel,y}}{\partial \vec{V}_y} = \langle 1 \rangle & \frac{\partial \vec{V}_{rel,y}}{\partial \vec{V}_z} = [0] \\
 \frac{\partial \vec{V}_{rel,z}}{\partial \vec{V}_x} = [0] & \frac{\partial \vec{V}_{rel,z}}{\partial \vec{V}_y} = [0] & \frac{\partial \vec{V}_{rel,z}}{\partial \vec{V}_z} = \langle 1 \rangle
 \end{array} \tag{G.51}$$

- **D^2 -Method Partial Derivatives**

$$\begin{aligned}
 \frac{\partial \vec{V}_{relx}}{\partial R_x} &= \frac{\partial \vec{V}_x}{\partial R_x} & \frac{\partial \vec{V}_{relx}}{\partial R_y} &= \langle \Omega_z \rangle & \frac{\partial \vec{V}_{relx}}{\partial R_z} &= \langle -\Omega_y \rangle \\
 \frac{\partial \vec{V}_{rely}}{\partial R_x} &= \langle -\Omega_z \rangle & \frac{\partial \vec{V}_{rely}}{\partial R_y} &= \frac{\partial \vec{V}_y}{\partial R_y} & \frac{\partial \vec{V}_{rely}}{\partial R_z} &= \langle \Omega_x \rangle \\
 \frac{\partial \vec{V}_{relz}}{\partial R_x} &= \langle \Omega_y \rangle & \frac{\partial \vec{V}_{relz}}{\partial R_y} &= \langle -\Omega_x \rangle & \frac{\partial \vec{V}_{relz}}{\partial R_z} &= \frac{\partial \vec{V}_z}{\partial R_z} \\
 \frac{\partial \vec{V}_{relx}}{\partial r_f} &= \frac{\partial \vec{V}_x}{\partial r_f} & \frac{\partial \vec{V}_{rely}}{\partial r_f} &= \frac{\partial \vec{V}_y}{\partial r_f} & \frac{\partial \vec{V}_{relz}}{\partial r_f} &= \frac{\partial \vec{V}_z}{\partial r_f}
 \end{aligned} \tag{G.52}$$

All other derivatives are zero.

Partial Derivatives of Magnitude of Relative Velocity

- **D -Method Partial Derivatives**

$$\begin{aligned}
 \frac{\partial \vec{V}_{rel}}{\partial R_x} &= \left\langle \frac{1}{\vec{V}_{rel}} \right\rangle \left[\left\langle \vec{V}_{relx} \right\rangle \frac{\partial \vec{V}_{relx}}{\partial R_x} + \left\langle \vec{V}_{rely} \right\rangle \frac{\partial \vec{V}_{rely}}{\partial R_x} + \left\langle \vec{V}_{relz} \right\rangle \frac{\partial \vec{V}_{relz}}{\partial R_x} \right] \\
 \frac{\partial \vec{V}_{rel}}{\partial R_y} &= \left\langle \frac{1}{\vec{V}_{rel}} \right\rangle \left[\left\langle \vec{V}_{relx} \right\rangle \frac{\partial \vec{V}_{relx}}{\partial R_y} + \left\langle \vec{V}_{rely} \right\rangle \frac{\partial \vec{V}_{rely}}{\partial R_y} + \left\langle \vec{V}_{relz} \right\rangle \frac{\partial \vec{V}_{relz}}{\partial R_y} \right] \\
 \frac{\partial \vec{V}_{rel}}{\partial R_z} &= \left\langle \frac{1}{\vec{V}_{rel}} \right\rangle \left[\left\langle \vec{V}_{relx} \right\rangle \frac{\partial \vec{V}_{relx}}{\partial R_z} + \left\langle \vec{V}_{rely} \right\rangle \frac{\partial \vec{V}_{rely}}{\partial R_z} + \left\langle \vec{V}_{relz} \right\rangle \frac{\partial \vec{V}_{relz}}{\partial R_z} \right] \\
 \frac{\partial \vec{V}_{rel}}{\partial V_x} &= \left\langle \frac{1}{\vec{V}_{rel}} \right\rangle \left[\left\langle \vec{V}_{relx} \right\rangle \frac{\partial \vec{V}_{relx}}{\partial V_x} + \left\langle \vec{V}_{rely} \right\rangle \frac{\partial \vec{V}_{rely}}{\partial V_x} + \left\langle \vec{V}_{relz} \right\rangle \frac{\partial \vec{V}_{relz}}{\partial V_x} \right] \\
 \frac{\partial \vec{V}_{rel}}{\partial V_y} &= \left\langle \frac{1}{\vec{V}_{rel}} \right\rangle \left[\left\langle \vec{V}_{relx} \right\rangle \frac{\partial \vec{V}_{relx}}{\partial V_y} + \left\langle \vec{V}_{rely} \right\rangle \frac{\partial \vec{V}_{rely}}{\partial V_y} + \left\langle \vec{V}_{relz} \right\rangle \frac{\partial \vec{V}_{relz}}{\partial V_y} \right] \\
 \frac{\partial \vec{V}_{rel}}{\partial V_z} &= \left\langle \frac{1}{\vec{V}_{rel}} \right\rangle \left[\left\langle \vec{V}_{relx} \right\rangle \frac{\partial \vec{V}_{relx}}{\partial V_z} + \left\langle \vec{V}_{rely} \right\rangle \frac{\partial \vec{V}_{rely}}{\partial V_z} + \left\langle \vec{V}_{relz} \right\rangle \frac{\partial \vec{V}_{relz}}{\partial V_z} \right]
 \end{aligned} \tag{G.53}$$

- **D^2 -Method Partial Derivatives**

$$\begin{aligned}
\frac{\partial \vec{V}_{rel}}{\partial \vec{R}_x} &= \left\langle \frac{1}{\vec{V}_{rel}} \right\rangle \left[\left\langle \vec{V}_{rel_x} \right\rangle \frac{\partial \vec{V}_{rel_x}}{\partial \vec{R}_x} + \left\langle \vec{V}_{rel_y} \right\rangle \frac{\partial \vec{V}_{rel_y}}{\partial \vec{R}_x} + \left\langle \vec{V}_{rel_z} \right\rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{R}_x} \right] \\
\frac{\partial \vec{V}_{rel}}{\partial \vec{R}_y} &= \left\langle \frac{1}{\vec{V}_{rel}} \right\rangle \left[\left\langle \vec{V}_{rel_x} \right\rangle \frac{\partial \vec{V}_{rel_x}}{\partial \vec{R}_y} + \left\langle \vec{V}_{rel_y} \right\rangle \frac{\partial \vec{V}_{rel_y}}{\partial \vec{R}_y} + \left\langle \vec{V}_{rel_z} \right\rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{R}_y} \right] \\
\frac{\partial \vec{V}_{rel}}{\partial \vec{R}_z} &= \left\langle \frac{1}{\vec{V}_{rel}} \right\rangle \left[\left\langle \vec{V}_{rel_x} \right\rangle \frac{\partial \vec{V}_{rel_x}}{\partial \vec{R}_z} + \left\langle \vec{V}_{rel_y} \right\rangle \frac{\partial \vec{V}_{rel_y}}{\partial \vec{R}_z} + \left\langle \vec{V}_{rel_z} \right\rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{R}_z} \right] \\
\frac{\partial \vec{V}_{rel}}{\partial \tau_f} &= \frac{1}{\vec{V}_{rel}} \left[\vec{V}_{rel_x} \frac{\partial \vec{V}_{rel_x}}{\partial \tau_f} + \vec{V}_{rel_y} \frac{\partial \vec{V}_{rel_y}}{\partial \tau_f} + \vec{V}_{rel_z} \frac{\partial \vec{V}_{rel_z}}{\partial \tau_f} \right]
\end{aligned} \tag{G.54}$$

All other derivatives are zero.

G.5.8 Partial Derivatives of Wind Relative Velocity in Body Coordinates

The partial derivatives of the body velocity defined in section 6.6.8 are derived in this Appendix. The partial derivatives for the D - and D^2 -methods are different.

- **D -Method Partial Derivatives**

The partial derivatives for V_{B_x} are given. The partial derivatives of V_{B_y} and V_{B_z} have a similar form, except that the quaternion matrix elements are different. To save space, the partial derivatives of V_{B_y} and V_{B_z} are not written.

$$\begin{aligned}
\frac{\partial \vec{V}_{B_x}}{\partial \vec{R}_x} &= \left\langle \vec{Q}_{11} \right\rangle \frac{\partial \vec{V}_{rel_x}}{\partial \vec{R}_x} + \left\langle \vec{Q}_{21} \right\rangle \frac{\partial \vec{V}_{rel_y}}{\partial \vec{R}_x} + \left\langle \vec{Q}_{31} \right\rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{R}_x} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{R}_y} &= \left\langle \vec{Q}_{11} \right\rangle \frac{\partial \vec{V}_{rel_x}}{\partial \vec{R}_y} + \left\langle \vec{Q}_{21} \right\rangle \frac{\partial \vec{V}_{rel_y}}{\partial \vec{R}_y} + \left\langle \vec{Q}_{31} \right\rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{R}_y} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{R}_z} &= \left\langle \vec{Q}_{11} \right\rangle \frac{\partial \vec{V}_{rel_x}}{\partial \vec{R}_z} + \left\langle \vec{Q}_{21} \right\rangle \frac{\partial \vec{V}_{rel_y}}{\partial \vec{R}_z} + \left\langle \vec{Q}_{31} \right\rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{R}_z} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{V}_x} &= \left\langle \vec{Q}_{11} \right\rangle \frac{\partial \vec{V}_{rel_x}}{\partial \vec{V}_x} + \left\langle \vec{Q}_{21} \right\rangle \frac{\partial \vec{V}_{rel_y}}{\partial \vec{V}_x} + \left\langle \vec{Q}_{31} \right\rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{V}_x} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{V}_y} &= \left\langle \vec{Q}_{11} \right\rangle \frac{\partial \vec{V}_{rel_x}}{\partial \vec{V}_y} + \left\langle \vec{Q}_{21} \right\rangle \frac{\partial \vec{V}_{rel_y}}{\partial \vec{V}_y} + \left\langle \vec{Q}_{31} \right\rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{V}_y}
\end{aligned} \tag{G.55}$$

$$\begin{aligned}
\frac{\partial \vec{V}_{B_x}}{\partial \vec{V}_z} &= \langle \vec{Q}_{11} \rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{V}_z} + \langle \vec{Q}_{21} \rangle \frac{\partial \vec{V}_{rel_y}}{\partial \vec{V}_z} + \langle \vec{Q}_{31} \rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{V}_z} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{q}_1} &= \langle \vec{V}_{rel_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_1} + \langle \vec{V}_{rel_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_1} \langle \vec{V}_{rel_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_1} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{q}_2} &= \langle \vec{V}_{rel_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_2} + \langle \vec{V}_{rel_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_2} \langle \vec{V}_{rel_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_2} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{q}_3} &= \langle \vec{V}_{rel_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_3} + \langle \vec{V}_{rel_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_3} \langle \vec{V}_{rel_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_3} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{q}_4} &= \langle \vec{V}_{rel_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_4} + \langle \vec{V}_{rel_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_4} \langle \vec{V}_{rel_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_4}
\end{aligned}$$

- **D^2 -Method Partial Derivatives**

The partial derivatives for V_{B_x} are given. The partial derivatives of V_{B_y} and V_{B_z} have a similar form, except that the quaternion matrix elements are different. To save space, the partial derivatives of V_{B_y} and V_{B_z} are not written.

$$\begin{aligned}
\frac{\partial \vec{V}_{B_x}}{\partial \vec{R}_x} &= \langle \vec{Q}_{11} \rangle \frac{\partial \vec{V}_{rel_x}}{\partial \vec{R}_x} + \langle \vec{Q}_{21} \rangle \frac{\partial \vec{V}_{rel_y}}{\partial \vec{R}_x} + \langle \vec{Q}_{31} \rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{R}_x} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{R}_y} &= \langle \vec{Q}_{11} \rangle \frac{\partial \vec{V}_{rel_x}}{\partial \vec{R}_y} + \langle \vec{Q}_{21} \rangle \frac{\partial \vec{V}_{rel_y}}{\partial \vec{R}_y} + \langle \vec{Q}_{31} \rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{R}_y} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{R}_z} &= \langle \vec{Q}_{11} \rangle \frac{\partial \vec{V}_{rel_x}}{\partial \vec{R}_z} + \langle \vec{Q}_{21} \rangle \frac{\partial \vec{V}_{rel_y}}{\partial \vec{R}_z} + \langle \vec{Q}_{31} \rangle \frac{\partial \vec{V}_{rel_z}}{\partial \vec{R}_z} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{q}_1} &= \langle \vec{V}_{rel_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_1} + \langle \vec{V}_{rel_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_1} \langle \vec{V}_{rel_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_1} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{q}_2} &= \langle \vec{V}_{rel_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_2} + \langle \vec{V}_{rel_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_2} \langle \vec{V}_{rel_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_2} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{q}_3} &= \langle \vec{V}_{rel_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_3} + \langle \vec{V}_{rel_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_3} \langle \vec{V}_{rel_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_3} \\
\frac{\partial \vec{V}_{B_x}}{\partial \vec{q}_4} &= \langle \vec{V}_{rel_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_4} + \langle \vec{V}_{rel_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_4} \langle \vec{V}_{rel_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_4} \\
\frac{\partial \vec{V}_{B_x}}{\partial \tau_f} &= \left[\vec{Q}_{11} \frac{\partial \vec{V}_{rel_x}}{\partial \tau_f} + \vec{Q}_{21} \frac{\partial \vec{V}_{rel_y}}{\partial \tau_f} + \vec{Q}_{31} \frac{\partial \vec{V}_{rel_z}}{\partial \tau_f} \right]
\end{aligned} \tag{G.56}$$

All other derivatives are zero.

G.5.9 Partial Derivatives of Inertial Aerodynamic Coefficients

The partial derivatives of the inertial aerodynamic coefficients defined in section 6.6.9 are derived in this Appendix. The partial derivatives for the D - and D^2 -methods are different.

- **D -Method Partial Derivatives**

The partial derivatives for C_{I_x} are given. The partial derivatives of C_{I_y} and C_{I_z} have a similar form, except that the quaternion matrix elements are different. To save space, the partial derivatives of C_{I_y} and C_{I_z} are not written.

$$\begin{aligned}
 \frac{\partial \tilde{C}_{I_x}}{\partial \vec{R}_x} &= \langle \tilde{Q}_{11} \rangle \frac{\partial \tilde{C}_{B_x}}{\partial \vec{R}_x} + \langle \tilde{Q}_{21} \rangle \frac{\partial \tilde{C}_{B_y}}{\partial \vec{R}_x} + \langle \tilde{Q}_{31} \rangle \frac{\partial \tilde{C}_{B_z}}{\partial \vec{R}_x} \\
 \frac{\partial \tilde{C}_{I_x}}{\partial \vec{R}_y} &= \langle \tilde{Q}_{11} \rangle \frac{\partial \tilde{C}_{B_x}}{\partial \vec{R}_y} + \langle \tilde{Q}_{21} \rangle \frac{\partial \tilde{C}_{B_y}}{\partial \vec{R}_y} + \langle \tilde{Q}_{31} \rangle \frac{\partial \tilde{C}_{B_z}}{\partial \vec{R}_y} \\
 \frac{\partial \tilde{C}_{I_x}}{\partial \vec{R}_z} &= \langle \tilde{Q}_{11} \rangle \frac{\partial \tilde{C}_{B_x}}{\partial \vec{R}_z} + \langle \tilde{Q}_{21} \rangle \frac{\partial \tilde{C}_{B_y}}{\partial \vec{R}_z} + \langle \tilde{Q}_{31} \rangle \frac{\partial \tilde{C}_{B_z}}{\partial \vec{R}_z} \\
 \frac{\partial \tilde{C}_{I_x}}{\partial \vec{V}_x} &= \langle \tilde{Q}_{11} \rangle \frac{\partial \tilde{C}_{B_x}}{\partial \vec{V}_x} + \langle \tilde{Q}_{21} \rangle \frac{\partial \tilde{C}_{B_y}}{\partial \vec{V}_x} + \langle \tilde{Q}_{31} \rangle \frac{\partial \tilde{C}_{B_z}}{\partial \vec{V}_x} \\
 \frac{\partial \tilde{C}_{I_x}}{\partial \vec{V}_y} &= \langle \tilde{Q}_{11} \rangle \frac{\partial \tilde{C}_{B_x}}{\partial \vec{V}_y} + \langle \tilde{Q}_{21} \rangle \frac{\partial \tilde{C}_{B_y}}{\partial \vec{V}_y} + \langle \tilde{Q}_{31} \rangle \frac{\partial \tilde{C}_{B_z}}{\partial \vec{V}_y} \\
 \frac{\partial \tilde{C}_{I_x}}{\partial \vec{V}_z} &= \langle \tilde{Q}_{11} \rangle \frac{\partial \tilde{C}_{B_x}}{\partial \vec{V}_z} + \langle \tilde{Q}_{21} \rangle \frac{\partial \tilde{C}_{B_y}}{\partial \vec{V}_z} + \langle \tilde{Q}_{31} \rangle \frac{\partial \tilde{C}_{B_z}}{\partial \vec{V}_z} \\
 \frac{\partial \tilde{C}_{I_x}}{\partial \vec{q}_1} &= \langle \tilde{Q}_{11} \rangle \frac{\partial \tilde{C}_{B_x}}{\partial \vec{q}_1} + \langle \tilde{Q}_{21} \rangle \frac{\partial \tilde{C}_{B_y}}{\partial \vec{q}_1} + \langle \tilde{Q}_{31} \rangle \frac{\partial \tilde{C}_{B_z}}{\partial \vec{q}_1} \\
 &\quad + \langle \tilde{C}_{B_x} \rangle \frac{\partial \tilde{Q}_{11}}{\partial \vec{q}_1} + \langle \tilde{C}_{B_y} \rangle \frac{\partial \tilde{Q}_{21}}{\partial \vec{q}_1} \langle \tilde{C}_{B_z} \rangle \frac{\partial \tilde{Q}_{31}}{\partial \vec{q}_1} \\
 \frac{\partial \tilde{C}_{I_x}}{\partial \vec{q}_2} &= \langle \tilde{Q}_{11} \rangle \frac{\partial \tilde{C}_{B_x}}{\partial \vec{q}_2} + \langle \tilde{Q}_{21} \rangle \frac{\partial \tilde{C}_{B_y}}{\partial \vec{q}_2} + \langle \tilde{Q}_{31} \rangle \frac{\partial \tilde{C}_{B_z}}{\partial \vec{q}_2} \\
 &\quad + \langle \tilde{C}_{B_x} \rangle \frac{\partial \tilde{Q}_{11}}{\partial \vec{q}_2} + \langle \tilde{C}_{B_y} \rangle \frac{\partial \tilde{Q}_{21}}{\partial \vec{q}_2} \langle \tilde{C}_{B_z} \rangle \frac{\partial \tilde{Q}_{31}}{\partial \vec{q}_2} \\
 \frac{\partial \tilde{C}_{I_x}}{\partial \vec{q}_3} &= \langle \tilde{Q}_{11} \rangle \frac{\partial \tilde{C}_{B_x}}{\partial \vec{q}_3} + \langle \tilde{Q}_{21} \rangle \frac{\partial \tilde{C}_{B_y}}{\partial \vec{q}_3} + \langle \tilde{Q}_{31} \rangle \frac{\partial \tilde{C}_{B_z}}{\partial \vec{q}_3} \\
 &\quad + \langle \tilde{C}_{B_x} \rangle \frac{\partial \tilde{Q}_{11}}{\partial \vec{q}_3} + \langle \tilde{C}_{B_y} \rangle \frac{\partial \tilde{Q}_{21}}{\partial \vec{q}_3} \langle \tilde{C}_{B_z} \rangle \frac{\partial \tilde{Q}_{31}}{\partial \vec{q}_3}
 \end{aligned} \tag{G.57}$$

$$\begin{aligned}\frac{\partial \vec{C}_{I_x}}{\partial \vec{q}_4} = & \langle \vec{Q}_{11} \rangle \frac{\partial \vec{C}_{B_x}}{\partial \vec{q}_4} + \langle \vec{Q}_{21} \rangle \frac{\partial \vec{C}_{B_y}}{\partial \vec{q}_4} + \langle \vec{Q}_{31} \rangle \frac{\partial \vec{C}_{B_z}}{\partial \vec{q}_4} \\ & + \langle \vec{C}_{B_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_4} + \langle \vec{C}_{B_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_4} \langle \vec{C}_{B_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_4}\end{aligned}$$

- **D^2 -Method Partial Derivatives**

The partial derivatives for C_{I_x} are given. The partial derivatives of C_{I_y} and C_{I_z} have a similar form, except that the quaternion matrix elements are different. To save space, the partial derivatives of C_{I_y} and C_{I_z} are not written.

$$\begin{aligned}\frac{\partial \vec{C}_{I_x}}{\partial \vec{R}_x} = & \langle \vec{Q}_{11} \rangle \frac{\partial \vec{C}_{B_x}}{\partial \vec{R}_x} + \langle \vec{Q}_{21} \rangle \frac{\partial \vec{C}_{B_y}}{\partial \vec{R}_x} + \langle \vec{Q}_{31} \rangle \frac{\partial \vec{C}_{B_z}}{\partial \vec{R}_x} \\ \frac{\partial \vec{C}_{I_x}}{\partial \vec{R}_y} = & \langle \vec{Q}_{11} \rangle \frac{\partial \vec{C}_{B_x}}{\partial \vec{R}_y} + \langle \vec{Q}_{21} \rangle \frac{\partial \vec{C}_{B_y}}{\partial \vec{R}_y} + \langle \vec{Q}_{31} \rangle \frac{\partial \vec{C}_{B_z}}{\partial \vec{R}_y} \\ \frac{\partial \vec{C}_{I_x}}{\partial \vec{R}_z} = & \langle \vec{Q}_{11} \rangle \frac{\partial \vec{C}_{B_x}}{\partial \vec{R}_z} + \langle \vec{Q}_{21} \rangle \frac{\partial \vec{C}_{B_y}}{\partial \vec{R}_z} + \langle \vec{Q}_{31} \rangle \frac{\partial \vec{C}_{B_z}}{\partial \vec{R}_z} \\ \frac{\partial \vec{C}_{I_x}}{\partial \vec{q}_1} = & \langle \vec{Q}_{11} \rangle \frac{\partial \vec{C}_{B_x}}{\partial \vec{q}_1} + \langle \vec{Q}_{21} \rangle \frac{\partial \vec{C}_{B_y}}{\partial \vec{q}_1} + \langle \vec{Q}_{31} \rangle \frac{\partial \vec{C}_{B_z}}{\partial \vec{q}_1} \\ & + \langle \vec{C}_{B_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_1} + \langle \vec{C}_{B_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_1} \langle \vec{C}_{B_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_1} \\ \frac{\partial \vec{C}_{I_x}}{\partial \vec{q}_2} = & \langle \vec{Q}_{11} \rangle \frac{\partial \vec{C}_{B_x}}{\partial \vec{q}_2} + \langle \vec{Q}_{21} \rangle \frac{\partial \vec{C}_{B_y}}{\partial \vec{q}_2} + \langle \vec{Q}_{31} \rangle \frac{\partial \vec{C}_{B_z}}{\partial \vec{q}_2} \\ & + \langle \vec{C}_{B_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_2} + \langle \vec{C}_{B_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_2} \langle \vec{C}_{B_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_2} \quad (G.58) \\ \frac{\partial \vec{C}_{I_x}}{\partial \vec{q}_3} = & \langle \vec{Q}_{11} \rangle \frac{\partial \vec{C}_{B_x}}{\partial \vec{q}_3} + \langle \vec{Q}_{21} \rangle \frac{\partial \vec{C}_{B_y}}{\partial \vec{q}_3} + \langle \vec{Q}_{31} \rangle \frac{\partial \vec{C}_{B_z}}{\partial \vec{q}_3} \\ & + \langle \vec{C}_{B_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_3} + \langle \vec{C}_{B_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_3} \langle \vec{C}_{B_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_3} \\ \frac{\partial \vec{C}_{I_x}}{\partial \vec{q}_4} = & \langle \vec{Q}_{11} \rangle \frac{\partial \vec{C}_{B_x}}{\partial \vec{q}_4} + \langle \vec{Q}_{21} \rangle \frac{\partial \vec{C}_{B_y}}{\partial \vec{q}_4} + \langle \vec{Q}_{31} \rangle \frac{\partial \vec{C}_{B_z}}{\partial \vec{q}_4} \\ & + \langle \vec{C}_{B_x} \rangle \frac{\partial \vec{Q}_{11}}{\partial \vec{q}_4} + \langle \vec{C}_{B_y} \rangle \frac{\partial \vec{Q}_{21}}{\partial \vec{q}_4} \langle \vec{C}_{B_z} \rangle \frac{\partial \vec{Q}_{31}}{\partial \vec{q}_4} \\ \frac{\partial \vec{C}_{I_x}}{\partial \tau_f} = & (\vec{Q}_{11}) \frac{\partial \vec{C}_{B_x}}{\partial \tau_f} + (\vec{Q}_{21}) \frac{\partial \vec{C}_{B_y}}{\partial \tau_f} + (\vec{Q}_{31}) \frac{\partial \vec{C}_{B_z}}{\partial \tau_f}\end{aligned}$$

All other derivatives are zero.

G.5.10 Partial Derivatives of Mach Number

The partial derivatives of the Mach number defined in section 6.6.10 are derived in this Appendix. The partial derivatives for the D - and D^2 -methods are different.

- D -Method Partial Derivatives

$$\begin{aligned}\frac{\partial \tilde{M}}{\partial R_x} &= \left\langle \frac{1}{\tilde{a}_\infty} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial R_x} - \left\langle \frac{\vec{V}_{rel}}{\tilde{a}_\infty^2} \right\rangle \frac{\partial \tilde{a}_\infty}{\partial R_x} & \frac{\partial \tilde{M}}{\partial V_x} &= \left\langle \frac{1}{\tilde{a}_\infty} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial V_x} \\ \frac{\partial \tilde{M}}{\partial R_y} &= \left\langle \frac{1}{\tilde{a}_\infty} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial R_y} - \left\langle \frac{\vec{V}_{rel}}{\tilde{a}_\infty^2} \right\rangle \frac{\partial \tilde{a}_\infty}{\partial R_y} & \frac{\partial \tilde{M}}{\partial V_y} &= \left\langle \frac{1}{\tilde{a}_\infty} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial V_y} \\ \frac{\partial \tilde{M}}{\partial R_z} &= \left\langle \frac{1}{\tilde{a}_\infty} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial R_z} - \left\langle \frac{\vec{V}_{rel}}{\tilde{a}_\infty^2} \right\rangle \frac{\partial \tilde{a}_\infty}{\partial R_z} & \frac{\partial \tilde{M}}{\partial V_z} &= \left\langle \frac{1}{\tilde{a}_\infty} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial V_z}\end{aligned}\quad (G.59)$$

- D^2 -Method Partial Derivatives

$$\begin{aligned}\frac{\partial \tilde{M}}{\partial R_x} &= \left\langle \frac{1}{\tilde{a}_\infty} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial R_x} - \left\langle \frac{\vec{V}_{rel}}{\tilde{a}_\infty^2} \right\rangle \frac{\partial \tilde{a}_\infty}{\partial R_x} \\ \frac{\partial \tilde{M}}{\partial R_y} &= \left\langle \frac{1}{\tilde{a}_\infty} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial R_y} - \left\langle \frac{\vec{V}_{rel}}{\tilde{a}_\infty^2} \right\rangle \frac{\partial \tilde{a}_\infty}{\partial R_y} \\ \frac{\partial \tilde{M}}{\partial R_z} &= \left\langle \frac{1}{\tilde{a}_\infty} \right\rangle \frac{\partial \vec{V}_{rel}}{\partial R_z} - \left\langle \frac{\vec{V}_{rel}}{\tilde{a}_\infty^2} \right\rangle \frac{\partial \tilde{a}_\infty}{\partial R_z} \\ \frac{\partial \tilde{M}}{\partial \tau_f} &= \left[\frac{1}{\tilde{a}_\infty} \frac{\partial \vec{V}_{rel}}{\partial \tau_f} \right]\end{aligned}\quad (G.60)$$

All other derivatives are zero.

G.5.11 Partial Derivatives of α and β

The partial derivatives of the aerodynamic angles α and β defined in section 6.6.11 are derived in this Appendix. The tangent function is chosen to define α and β . From [25, p. 334], the derivative of an inverse tangent is:

$$\frac{\partial}{\partial x} (\tan^{-1} u) = \frac{1}{1 + u^2} \frac{\partial u}{\partial x} \quad (G.61)$$

- Partial Derivatives of α

The general form of the derivative is:

$$\frac{\partial \tilde{\alpha}}{\partial \tilde{x}} = \left\langle \frac{1}{\vec{V}_{B_x}^2 + \vec{V}_{B_z}^2} \right\rangle \left[\left\langle \vec{V}_{B_x} \right\rangle \frac{\partial \vec{V}_{B_x}}{\partial \tilde{x}} - \left\langle \vec{V}_{B_z} \right\rangle \frac{\partial \vec{V}_{B_z}}{\partial \tilde{x}} \right] \quad (G.62)$$

- **Partial Derivatives of β**

The general form of the derivative is:

$$\frac{\partial \vec{\beta}}{\partial \vec{x}} = \left\langle \frac{\sqrt{\vec{V}_{B_x}^2 + \vec{V}_{B_z}^2}}{\vec{V}_{B_x}^2 + \vec{V}_{B_y}^2 + \vec{V}_{B_z}^2} \right\rangle \dots$$

$$\left[\frac{\partial \vec{V}_{B_y}}{\partial \vec{x}} - \left\langle \frac{\vec{V}_{B_y}}{\vec{V}_{B_x}^2 + \vec{V}_{B_z}^2} \right\rangle \left(\left\langle \vec{V}_{B_x} \right\rangle \frac{\partial \vec{V}_{B_x}}{\partial \vec{x}} + \left\langle \vec{V}_{B_z} \right\rangle \frac{\partial \vec{V}_{B_z}}{\partial \vec{x}} \right) \right] \quad (G.63)$$

G.5.12 Partial Derivatives of Body Aerodynamic Coefficients

The partial derivatives of the body aerodynamic coefficients defined in section 6.6.12 are derived in this Appendix. The body aerodynamic force coefficients are numerical functions of α , β , and Mach number. In this thesis, a finite difference scheme is used to compute the partial derivatives of C_{B_x} , C_{B_y} and C_{B_z} with respect to Mach number, α , and β . These partial derivatives can be written mathematically as diagonal matrices. For example, C_{B_x} at the first LGL point varies only with the Mach number, α , and β at the first LGL point. It does not vary with values at the other LGL points.

The partial derivatives of the body aerodynamic coefficients with respect to the optimization vector are required. The general form of the partial derivatives of the body force coefficients is given below:

$$\frac{\partial \vec{C}_{B_x}}{\partial \vec{x}} = \frac{\partial \vec{C}_{B_x}}{\partial \vec{M}} \frac{\partial \vec{M}}{\partial \vec{x}} + \frac{\partial \vec{C}_{B_x}}{\partial \vec{\alpha}} \frac{\partial \vec{\alpha}}{\partial \vec{x}} + \frac{\partial \vec{C}_{B_x}}{\partial \vec{\beta}} \frac{\partial \vec{\beta}}{\partial \vec{x}}$$

$$\frac{\partial \vec{C}_{B_y}}{\partial \vec{x}} = \frac{\partial \vec{C}_{B_y}}{\partial \vec{M}} \frac{\partial \vec{M}}{\partial \vec{x}} + \frac{\partial \vec{C}_{B_y}}{\partial \vec{\alpha}} \frac{\partial \vec{\alpha}}{\partial \vec{x}} + \frac{\partial \vec{C}_{B_y}}{\partial \vec{\beta}} \frac{\partial \vec{\beta}}{\partial \vec{x}} \quad (G.64)$$

$$\frac{\partial \vec{C}_{B_z}}{\partial \vec{x}} = \frac{\partial \vec{C}_{B_z}}{\partial \vec{M}} \frac{\partial \vec{M}}{\partial \vec{x}} + \frac{\partial \vec{C}_{B_z}}{\partial \vec{\alpha}} \frac{\partial \vec{\alpha}}{\partial \vec{x}} + \frac{\partial \vec{C}_{B_z}}{\partial \vec{\beta}} \frac{\partial \vec{\beta}}{\partial \vec{x}}$$

Note that the form of the partial derivatives is the same for both the D - and D^2 -methods. However, also note that $\frac{\partial \vec{M}}{\partial \vec{x}}$, $\frac{\partial \vec{\alpha}}{\partial \vec{x}}$, and $\frac{\partial \vec{\beta}}{\partial \vec{x}}$ may be different for each method, depending on what \vec{x} represents. For example, $\frac{\partial \vec{M}}{\partial \vec{T}}$ is the same for both methods, but $\frac{\partial \vec{M}}{\partial \vec{R}_x}$ is not the same.

Partial Derivatives of Simplified Body Force Coefficients

The nonzero partial derivatives of the simplified body force coefficients are:

$$\begin{aligned}\frac{\partial \vec{C}_{B_y}}{\partial \vec{\beta}} &= \left\langle \frac{dC_{B_y}}{d\beta} \right\rangle \\ \frac{\partial \vec{C}_{B_z}}{\partial \vec{\alpha}} &= \left\langle \frac{dC_{B_z}}{d\alpha} \right\rangle\end{aligned}\quad (\text{G.65})$$

G.5.13 Partial Derivatives of Atmosphere Model

The atmosphere model used for the three dimensional model is given in Appendix C.2. This atmosphere model gives the derivatives of atmospheric density, pressure, temperature, and speed of sound with respect to altitude (h). Note that:

$$\vec{R}_{mag} = \sqrt{\vec{R}_x^2 + \vec{R}_y^2 + \vec{R}_z^2} \quad (\text{G.66})$$

$$h = \vec{R}_{mag} - R_P \quad (\text{G.67})$$

Equation G.66 leads to:

$$\begin{aligned}\frac{\partial \vec{R}_{mag}}{\partial \vec{R}_x} &= \left\langle \frac{\vec{R}_x}{\vec{R}_{mag}} \right\rangle \\ \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_y} &= \left\langle \frac{\vec{R}_y}{\vec{R}_{mag}} \right\rangle \\ \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_z} &= \left\langle \frac{\vec{R}_z}{\vec{R}_{mag}} \right\rangle\end{aligned}\quad (\text{G.68})$$

Equation G.67 leads to:

$$dh = dR_{mag} \quad (\text{G.69})$$

Therefore,

$$\begin{aligned}\frac{\partial \tilde{p}}{\partial \vec{R}_x} &= \frac{\partial \tilde{p}}{\partial h} \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_x} & \frac{\partial \tilde{p}}{\partial \vec{R}_y} &= \frac{\partial \tilde{p}}{\partial h} \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_y} & \frac{\partial \tilde{p}}{\partial \vec{R}_z} &= \frac{\partial \tilde{p}}{\partial h} \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_z} \\ \frac{\partial \tilde{p}}{\partial \vec{R}_x} &= \frac{\partial \tilde{p}}{\partial h} \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_x} & \frac{\partial \tilde{p}}{\partial \vec{R}_y} &= \frac{\partial \tilde{p}}{\partial h} \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_y} & \frac{\partial \tilde{p}}{\partial \vec{R}_z} &= \frac{\partial \tilde{p}}{\partial h} \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_z} \\ \frac{\partial \tilde{T}_{emp}}{\partial \vec{R}_x} &= \frac{\partial \tilde{T}_{emp}}{\partial h} \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_x} & \frac{\partial \tilde{T}_{emp}}{\partial \vec{R}_y} &= \frac{\partial \tilde{T}_{emp}}{\partial h} \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_y} & \frac{\partial \tilde{T}_{emp}}{\partial \vec{R}_z} &= \frac{\partial \tilde{T}_{emp}}{\partial h} \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_z} \\ \frac{\partial \tilde{a}_{\infty}}{\partial \vec{R}_x} &= \frac{\partial \tilde{a}_{\infty}}{\partial h} \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_x} & \frac{\partial \tilde{a}_{\infty}}{\partial \vec{R}_y} &= \frac{\partial \tilde{a}_{\infty}}{\partial h} \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_y} & \frac{\partial \tilde{a}_{\infty}}{\partial \vec{R}_z} &= \frac{\partial \tilde{a}_{\infty}}{\partial h} \frac{\partial \vec{R}_{mag}}{\partial \vec{R}_z}\end{aligned}\quad (\text{G.70})$$

where

- $\vec{\rho}$ = atmospheric densities at the LGL points
- \vec{p} = atmospheric pressures at the LGL points
- \vec{T}_{temp} = atmospheric temperatures at the LGL points
- \vec{a}_∞ = atmospheric speeds of sound at the LGL points

Appendix H

Guidelines for Applying the Legendre Pseudospectral Method

While completing this research, much experience was gained in applying the Legendre Pseudospectral Method of this thesis. Following are some (hopefully) useful guidelines for programming an optimization problem using this method.

1. Choose values to nondimensionalize the problem. This is a very important step. Try to choose values that make sense for the given problem. Try to pick nondimensionalization values that will make all the values in the optimization vector be roughly the same order of magnitude. This will usually improve the numerical properties of the problem.
2. Write the dynamic equations. These usually consist of any differential equations that describe the system and that must be met at all points of the simulation. Some examples include the equations of motion and the relation between thrust and mass flow rate.
3. Write the trajectory constraint conditions. These conditions apply at every point of the trajectory and are usually inequality constraints. Note that these constraints may include derivatives that require the use of the pseudospectral differentiation matrix.

- /
4. Write the point constraint conditions. These conditions apply only at certain points in the simulations. Usually, these are used to apply initial and final value constraints. These may also include derivatives.
 5. Transform the domain of the dynamic equations to the LGL domain. It is also necessary to transform the time domain of any trajectory or point condition that contains a derivative. If a condition does not contain a derivative, then no time transformation is necessary. For the launch problem, this means transforming from the real time domain to the LGL time domain. However, it may be possible to have a problem that has derivatives with respect to variables other than time.
 6. Use the D - or D^2 -matrix to transform the derivatives into algebraic expressions.
 7. Rearrange the dynamic equations into dynamic constraints. Note that the best way to write a constraint is problem dependent. Note also that it may be possible to combine multiple dynamic equations into a single dynamic constraint.
 8. Choose the optimization vector based on the variables in the dynamic constraints. Note that this step may be helpful in identifying dynamic constraints that can be combined, thus leading to a smaller optimization vector.
 9. Rearrange the trajectory and point constraint conditions into trajectory and point constraints. The best way to write a constraint is problem dependent.
 10. Decide the proper bounds for the optimization vector. Identify trajectory and point constraints that can be written in terms of bounding the optimization variables. In general, it is better to write constraints in terms of variable bounds instead of direct constraints.
 11. If desired, find the analytic Jacobian matrix. This will make the numerical optimizer run much faster. Use the matrix derivative rules reviewed in Appendix B. It is usually easier to divide the Jacobian into simpler pieces using the chain rule.

References

- [1] Anderson, J. D., *Introduction to Flight*, McGraw-Hill Series in Aeronautical and Aerospace Engineering, McGraw-Hill, New York, New York, 3rd ed., 1989.
- [2] Bate, R., Mueller, D., and White, J., *Fundamentals of Astrodynamics*, Dover Publications, New York, 1971.
- [3] Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control and Dynamics*, Vol. 21, No. 2, Mar.-Apr. 1998, pp. 193–207.
- [4] Brauer, G. L., Cornick, D. E., and Stevenson, R., "Capabilities of the Program to Optimize Simulated Trajectories (POST)," Tech. Rep. NASA CR-2770, NASA, Feb. 1977.
- [5] Bryson, A. E., *Dynamic Optimization*, Addison-Wesley, Menlo Park, California, 1999.
- [6] Cooper, W. A.. "1992 Advanced Study Program Summer Colloquium Notes," <http://www.asp.ucar.edu/colloquium/1992/notes/part1/node75.html>, National Center for Atmospheric Research, Boulder, Colorado, Mar. 2001.
- [7] Fahroo, F. and Ross, I. M., "A Spectral Patching Method for Direct Trajectory Optimization," *The Journal of the Astronautical Sciences*, Vol. 48, No. 2/3, 2000.
- [8] Fahroo, F. and Ross, I. M., "Costate Estimation by a Legendre Pseudospectral Method," *Journal of Guidance, Control and Dynamics*, Vol. 24, No. 2, Mar.-Apr. 2001, pp. 270–277.

- [9] Fogiel, M., editor, *Handbook of Mathematical, Scientific, and Engineering Formulas, Tables, Functions, Graphs, Transforms*, Research and Education Association, Piscataway, New Jersey, 1997.
- [10] Fornberg, B., *A Practical Guide to Pseudospectral Methods*, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press, Cambridge, United Kingdom, 1998.
- [11] Gill, P. E., Murray, W., Saunders, M. A., and Wright, M. H., "User's Guide for NPSOL 5.0: A FORTRAN Package for Nonlinear Programming," Tech. rep., Stanford Business Software, Inc., Palo Alto, California, July 1998.
- [12] Isakowitz, S. J., Hopkins, J. P., and Hopkins, J. B., *International Reference Guide to Space Launch Systems*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 3rd ed., 1999.
- [13] Meriam, J. L. and Kraige, L. G., *Dynamics*, Engineering Mechanics, John Wiley and Sons, Inc., New York, New York, 3rd ed., 1992.
- [14] Moog, R. D., Smith, N. G., Corwin, R. A., and Livezey, W. F., "Parachute Simulation User's Guide Computer Program UD233A," Tech. rep., Martin Marietta Corporation, Denver Aerospace Division, Denver, Colorado, Feb. 1986.
- [15] Nakamura, S., *Numerical Analysis and Graphic Visualization with MATLAB*, Prentice-Hall, Upper Saddle River, New Jersey, 1996.
- [16] Riehl, J. P., "Optimal Trajectories by Implicit Simulation (OTIS)," <http://trajectory.lerc.nasa.gov/NewPage/Tools/otis.html>, Analysis and Integration Group, NASA Glenn Research Center, Cleveland, Ohio, May 2001.
- [17] Ross, I. M., Personal notes and conversations, 2001.
- [18] Ross, I. M., "3-DOF Model: Canonical Equations of Motions," Personal notes, 2001.

- [19] Seywald, H. and Cliff, E. M., "Goddard Problem in Presence of a Dynamic Pressure Limit," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 4, Jul.-Aug. 1993, pp. 776–781.
- [20] Shuster, M. D., "A Survey of Attitude Representations," *The Journal of the Astronautical Sciences*, Vol. 41, No. 4, 1993, pp. 439–517.
- [21] Smith, J. O., "Notes on Interpolation," http://www-ccrma.stanford.edu/~jos/Interpolation/Lagrange_Interpolation.html, Center for Computer Research in Music and Acoustics, Stanford University, Stanford, California, Mar. 2001.
- [22] Stott, J., "The Constants and Equations Pages: Astronomy: Planets: Earth," <http://tcaep.co.uk/astro/planets/earth/index.htm>, Mar. 2001.
- [23] Trefethen, L. N., *Spectral Methods in MATLAB*, Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania, 2000.
- [24] Yedavalli, R. K., "AAE-520 Flight Vehicle Dynamics Class Notes," Department of Aeronautical and Astronautical Engineering, Ohio State University, Columbus, Ohio, 1998.
- [25] Zwillinger, D., editor, *CRC Standard Mathematical Tables and Formulae*, CRC Press, Boca Raton, Florida, 30th ed., 1996.

[This page intentionally left blank.]

THESIS PROCESSING SLIP

FIXED FIELD: ill. _____ name _____

index _____ biblio _____

► COPIES: Archives Aero Dewey Barker Hum
Lindgren Music Rotch Science Sche-Plough

TITLE VARIES: ► _____

NAME VARIES: ► _____

IMPRINT: (COPYRIGHT) _____

► COLLATION: _____

► ADD: DEGREE: _____ ► DEPT.: _____

► ADD: DEGREE: _____ ► DEPT.: _____

SUPERVISORS: _____

NOTES:

cat'r:

date:

page:

► DEPT: Aero ► J1EL

► YEAR: 2001 ► DEGREE: B.M.

► NAME: BCH, Jeremy Ryan