RAPID SIMULTANEOUS HYPERSONIC AERODYNAMIC AND TRAJECTORY OPTIMIZATION FOR CONCEPTUAL DESIGN

A Thesis
Presented to
The Academic Faculty

by

Michael J. Grant

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
May 2012

RAPID SIMULTANEOUS HYPERSONIC AERODYNAMIC AND TRAJECTORY OPTIMIZATION FOR CONCEPTUAL DESIGN

Approved by:

Professor Robert D. Braun,
Committee Chair
School of Aerospace Engineering
Georgia Institute of Technology

Gregg H. Barton
Mission Design Group
Charles Stark Draper Laboratory

Visiting Professor Ian G. Clark
School of Aerospace Engineering
Georgia Institute of Technology

Professor Brian J. German
School of Aerospace Engineering
Georgia Institute of Technology

Professor Ryan P. Russell
Department of Aerospace Engineering
and Engineering Mechanics
University of Texas at Austin

Professor Panagiotis Tsiotras
School of Aerospace Engineering
Georgia Institute of Technology

Date Approved: 27 March 2012

To my wife, Jen.

To my parents, John and Susan.

To my sisters, Emily and Jessica.

It is the people that we love and hold close who shape our beliefs and give us the drive to change the world.

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the support of many people during my time at Georgia Tech. First, I would like to thank my advisor, Prof. Robert Braun. It has been a great privilege to work with you, to learn from you, to gain classroom experience from you, and to travel to meetings with you. Without your effort, I would not be the engineer I am today. I have always enjoyed our discussions, and I thank you for all the ideas you have given me. I would also like to thank Dr. Ian Clark for his insight during his time as a visiting professor at Georgia Tech.

I would also like to acknowledge the Charles Stark Draper Laboratory for funding my research throughout my years at Georgia Tech and for all the guidance I have received from many members of the organization. I would first like to thank Gregg Barton for his mentoring over the years. His guidance was not limited to my PhD work, and I always enjoyed our discussions during the summers I worked in Houston. I would also like to thank Ron Proulx, Sean George, and the many other team members.

I am very lucky to also have had such great lab mates in the Space Systems Design Lab. I really enjoyed working with everyone and having the opportunity to work in such a stimulating environment. I would especially like to thank Brad Steinfeldt who shared all but one class with me. I would not have learned so much without you. I would also like to thank Jarret Lafleur, Zarrin Chua, Ashley Korzun, Zach Putnam, Chris Tanner, and Richard Otero.

Finally, I would like to thank my family who has always supported my dreams and opportunities. I would also like to thank Jen, my wife, for being supportive of me during our time apart while I was at Georgia Tech. Without all of this support, the accomplishments contained in this document would not have been possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

Acronyms

ACT	Adjoint Control Transformation
AOA	Abort-Once-Around
APAS	Aerodynamic Preliminary Analysis System
ATO	Abort to Orbit
BLISS	Bi-Level Integrated System Synthesis
BVP	Boundary Value Problem
CBAERO	Configuration Based Aerodynamics
CFD	Computational Fluid Dynamics
CMT	Covector Mapping Theorem
CO	Collaborative Optimization
DP	Dynamic Programming
DSM	Design Structure Matrix
ECAL	East Coast Abort Landing
FPA	Flight Path Angle
GA	Genetic Algorithm
GPOPS	Gauss Pseudospectral Optimization Software
GPU	Graphics Processing Unit

GTS	GNU Triangulated Surface Library
HABP	Hypersonic Arbitrary Body Program
IAD	Inflatable Aerodynamic Decelerator
KKT	Karush-Kuhn-Tucker
LGL	Legendre-Gauss-Lobatto
MDO	Multidisciplinary Design Optimization
MECO	Main Engine Cutoff
MOPSO	Multi-Objective Particle Swarm Optimizer
NLP	Nonlinear Programming
OTIS	Optimal Trajectories by Implicit Simulation
PMP	Pontryagin's Minimum Principle
POST	Program to Optimize Simulated Trajectories
PSO	Particle Swarm Optimizer
PTM	Press to MECO
RK	Runge-Kutta
RTLS	Return to Launch Site
SORT	Simulation and Optimization of Rocket Trajectories
SQP	Sequential Quadratic Programming
STM	State Transition Matrix
STT	State Transition Tensor

TAL	Transoceanic Abort Landing
TPS	Thermal Protection System

Superscripts and Subscripts

$*$	reference solution
0	initial or surface
1	forward section
2	aft section
∞	freestream
A	axial force
A'	axial force in total angle of attack frame
b	Bezier curve
c	cone
D	drag
e	Earth
f	final
L	lift
l	moment about x-axis
m	moment about y-axis
m, α	moment stiffness about y-axis
N	normal force

n	moment about z-axis or nose
n,β	moment stiffness about z-axis
ref	reference
S	side force
sc	sphere-cone
X	force in x-direction
Y	force in y-direction
Z	force in z-direction

Symbols

α	angle of attack	rad
β	ballistic coefficient or sideslip	kg/m^2 or rad
ΔA	area of panel	m^2
δ	half-angle	rad
$\dot{q}$	heat rate	W/cm^2
ϵ	total effective angle of attack	rad
γ	relative flight path angle	rad
λ	costate	
$\hat{\mathbf{n}}$	inward unit normal	m
ν	undetermined constants	
$\mathbf{\Phi}$	state transition tensor	

$\boldsymbol{\Psi}$	terminal constraint	
$\mathbf{B}$	control node location of Bezier curve	m
$\mathbf{dA}$	differential area element	m^2
$\mathbf{df}$	differential force	N
$\mathbf{P}$	Bezier curve function	m
$\mathbf{r}$	position of differential surface element	m
$\mathbf{S}$	path constraint or unshadowed surface domain	
$\mathbf{u}$	control vector	
$\mathbf{V}$	velocity vector	m/s
$\mathbf{x}$	state vector	
$\mathbf{x}'$	augmented state vector	
μ	gravitational parameter	m^3/s^2
Φ	terminal cost	
ϕ	bank angle	rad
ϕ'	angle between the body frame and total angle of attack frame	rad
ρ	density or potential function radius	kg/m^3 or m
θ	downrange subtended angle	rad
θ	local body inclination relative to freestream direction	rad
A	vehicle area	m^2
C	aerodynamic coefficient	

C_p	pressure coefficient	
D	drag force magnitude	N
d	diameter	m
dA	differential area element	m^2
dA_{yz}	differential area projected on y-z plane	m^2
H	Hamiltonian or scale height	nd or m
h	height	m
I	path cost	
J	cost functional	
J'	augmented objective	
L	length or lift force magnitude	m or N
l	vehicle length	m
L/D	lift to drag ratio	
m	vehicle mass	kg
n	number of objectives or order of Bezier curve	
p	order of Taylor series solution or pressure	nd or Pa
r	radius	m
s	arclength	
t	time	s
u	surface parametrization variable	

v	velocity magnitude or surface parametrization variable	m/s or nd

CHAPTER I

MOTIVATION AND BACKGROUND

1.1 Motivation

Fundamentally, design optimization is a root-solving process in which a system with optimal characteristics is constructed within the constraints of the problem. To perform this root-solving process, disciplinary models, or contributing analyses, are generally connected into a large iterative design process often illustrated using a design structure matrix (DSM) as shown in Figure 1 for entry system design.[2] The DSM maps the interaction of various contributing analyses and provides a general framework for the multidisciplinary design optimization (MDO) of aerospace systems.[8] Due to the complexity of developing each individual contributing analysis, designers often select tools developed by disciplinary experts to perform the necessary calculations. As a result, these contributing analyses are often developed independently from the overall design process in which little or no modification can be made (*e.g.*, legacy and proprietary codes). To support this design environment, efficient MDO algorithms such as Collaborative Optimization (CO) and Bi-Level Integrated System Synthesis (BLISS) have been developed that analyze and reconstruct the various connections among the contributing analyses to improve the overall design process.[9, 10, 11, 12, 13, 14, 15] Consequently, many tools have been developed from various methods to address complex MDO problems, including iSIGHT[16], Integrated Hypersonic Aeromechanics Tool (IHAT)[17], and ModelCenter[18].

The independent advancements in MDO processes (by designers) and in contributing analyses (by disciplinary experts) are important for high fidelity design. While

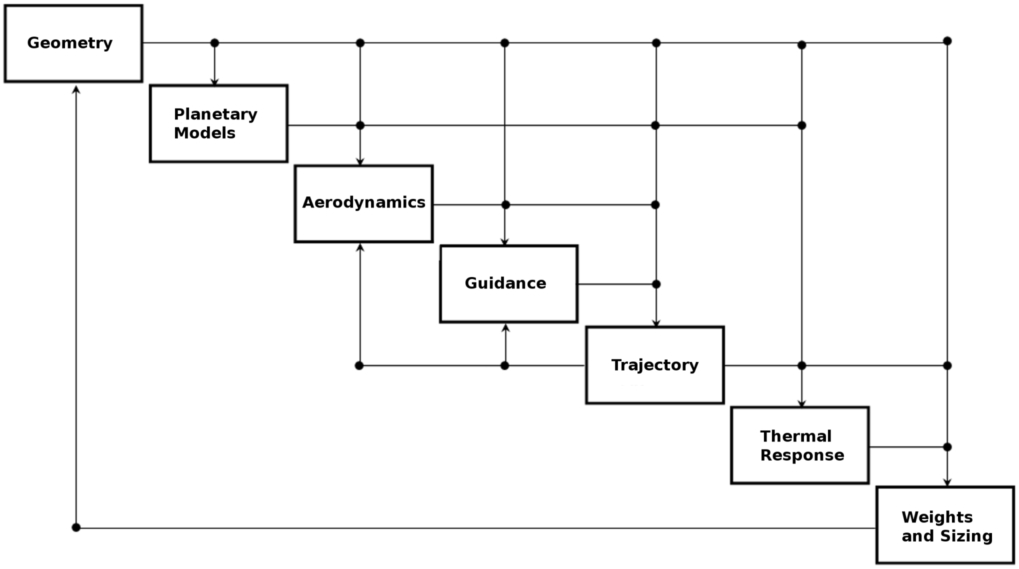


Figure 1: Example design structure matrix for planetary entry systems.[2]

current research in MDO attempts to improve the efficiency of the overall design process, disciplinary advancements often focus on achieving higher fidelity or capturing a wider range of solutions, both of which increase the computational requirements of the design process. Additionally, design variables must be separately chosen within each contributing analysis. As a result, the interaction among cross-discipline design variables is segregated, limiting the use of fast, specialized optimization methods in favor of slower, more generalized methods. For example, designers of high performance, hypersonic missions are initially interested in the extent of vehicle performance that can be achieved. With this consideration, conceptual vehicle capability studies are performed to identify the type and configuration of the vehicle that should be flown as well as the corresponding envelope of trajectory capability. If this envelope includes missions of interest, then further detailed trades can follow. With this consideration, hypersonic vehicle capability is generally obtained through iteration among vehicle shape, aerodynamic performance, and trajectory optimization routines. This sequential, iterative process is the result of a fundamental segregation between vehicle shape and trajectory optimization routines that results from the characterization of aerodynamic performance for each vehicle shape using large aerodynamic tables that are a function of angle of attack, sideslip, and flight conditions. As a result, this numerical relationship has also segregated advancements in vehicle shape design from advancements in trajectory optimization.

Analytic aerodynamic relations, like those derived in this investigation, are possible in any flow regime in which the flowfield model can be accurately described analytically. These relations eliminate the large aerodynamic tables that contribute to the segregation of disciplinary advancements, and as a result, greatly improves the speed of conceptual vehicle capability studies. Within the limits of Newtonian flow theory, many of the analytic expressions derived in this dissertation provide exact solutions that eliminate computational error of approximate methods widely used today while simultaneously improving computational performance. As a result, these relations can be used to instantaneously model configuration changes during flight and shape change due to ablation. These relations also provide an analytic mapping of vehicle shape to trajectory performance. This analytic mapping collapses the DSM into a single, unified, mathematical framework which enables fast, specialized trajectory optimization methods that are not sufficiently flexible to be supported in the traditional, segregated design environment to be extended and simultaneously applied across all disciplines. As such, a rapid trajectory optimization methodology suitable for this new, mathematically integrated design environment is also developed. This trajectory optimization methodology is capable of easily satisfying in-flight constraints such as sensed acceleration and stagnation point heat rate. The extension of this trajectory optimization methodology to include vehicle shape through the development of analytic hypersonic aerodynamic relations enables the construction of a new MDO methodology to perform rapid simultaneous hypersonic aerodynamic and trajectory optimization.

1.2 Hypersonic Aerodynamics Overview

1.2.1 Introduction

Simultaneously accounting for all major disciplines during the design of hypersonic vehicles is challenging due to the computational requirements of high fidelity codes. As a result, a time consuming iterative process is typically employed using a large team in which members separately analyze each discipline. For example, computational fluid dynamics (CFD) is often used to obtain the aerodynamics of various hypersonic vehicles. Due to the substantial computational requirements of CFD, vehicles are generally chosen that provide the necessary aerodynamic performance, such as L/D and ballistic coefficient, to accomplish a given mission. Subsequently, the vehicle is designed to meet these performance constraints.

During conceptual design, methods are typically employed to improve computational speed at the expense of fidelity. For example, panel methods can be used in conjunction with Newtonian flow theory to obtain the hypersonic aerodynamic characteristics of a vehicle with orders of magnitude reduction in computational requirements compared to CFD. Panel methods are widely used during conceptual design due to the ability of these methods to rapidly evaluate arbitrary shapes. Although panel methods are much faster than CFD, the designer is generally still required to limit the number of vehicle shapes analyzed due to the computational requirements of generating aerodynamic coefficient tables. Panel methods are necessary for complicated vehicle geometries including the Space Shuttle Orbiter and X-38. However, the geometry of many common hypersonic vehicles of interest such as sphere-cones, blunted biconics, and spherical forebody segments can be expressed analytically. As a result, the corresponding aerodynamic coefficients can also be developed analytically using Newtonian flow theory.

1.2.2 Newtonian Aerodynamic Theory

In 1687, Isaac Newton postulated that fluid flow can be viewed as a system of particles traveling in rectilinear motion as described in Propositions 34 and 35 of the *Principia*[19]. Newton assumed that when a particle strikes a surface, all of the momentum normal to the surface would be lost and all momentum tangential to the surface would be conserved as shown in Figure 2. Consequently, the pressure exerted by the fluid on the surface of a body is assumed to be solely originating from this loss of momentum normal to the surface. Under these assumptions, the nondimensional pressure coefficient, C_p, at any point on the surface of a body can be obtained from the Newtonian sine-squared relation shown in Eq. (1). As a result, the solution to the hypersonic aerodynamics problem is mapped to a geometry problem. Furthermore, the pressure exerted by the fluid on any portion of the surface not directly exposed to the flow, denoted as the shadowed region of the body, is assumed to be equivalent to the freestream pressure in which the motion of the fluid does not influence the pressure in this region. Consequently, $C_p = 0$ throughout the shadowed region as shown in Figure 3.[3]

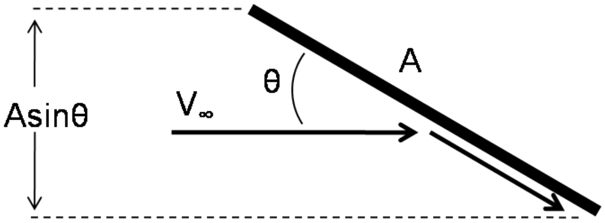


Figure 2: Momentum transfer of particle on inclined surface.[3]

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho_\infty V_\infty^2} = 2\sin^2\theta \qquad (1)$$

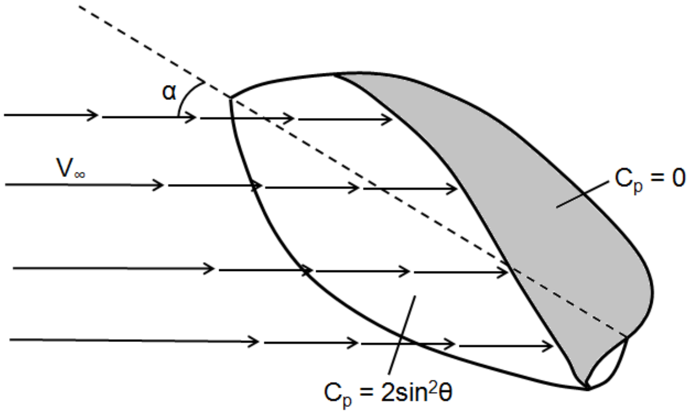


Figure 3: Example of shadowed body.[3]

Newton originally used his theory to model the pressure on the walls of a water channel. Experimental tests performed by d'Alembert later concluded that this model is inaccurate for subsonic flow conditions.[20] However, as the Mach number increases to hypersonic speeds in continuum flow, a shock wave is formed near the surface of the vehicle. As the fluid particles change direction after crossing the shock at hypersonic speeds, the flow appears to be deflected by the vehicle in a manner similar to Newtonian flow theory as shown in Figure 2. Since the shock approaches the surface of the vehicle as Mach number increases, the pressure exerted on the vehicle also approaches the Newtonian result. For conceptual design applications, the Mach-independent Newtonian pressure coefficient shown in Eq. (1) enables fast approximation of aerodynamic coefficients at hypersonic speeds.[3]

In order to calculate the aerodynamic coefficients of a hypersonic vehicle, a surface integral of C_p is traditionally evaluated. As previously mentioned, the pressure exerted on the vehicle is due to the total loss of momentum normal to the surface. Thus, the pressure coefficient exerted over a differential element on the surface of a vehicle, dA, results in a differential force, $\mathbf{df}$, imparted on the vehicle in the inward unit normal direction of the surface, $\hat{\mathbf{n}}$, as shown in Eq. (2). Using conventional aircraft body axes shown in Figure 4 and the corresponding freestream velocity vector, $\mathbf{V}_\infty$, shown in Eq. (3) as function of angle of attack, α, and sideslip, β, the aerodynamic force coefficients along the body axes, C_X, C_Y, and C_Z, can be calculated through surface integration of the differential force in the x, y, and z directions as shown in Eq. (4). The axial force coefficient, C_A, side force coefficient, C_S, and normal force coefficient, C_N, commonly used in hypersonic vehicle applications is related to the aerodynamic force coefficients along the body axes as shown in Eq. (4). The corresponding moment coefficients about the body axes, C_l, C_m, and C_n, are computed relative to the origin of the body axes as shown in Eq. (5), where $\mathbf{r}$ is the position vector of the differential area element. Furthermore, pitch and yaw stability derivatives can be calculated using Eq. (6) and Eq. (7), respectively.

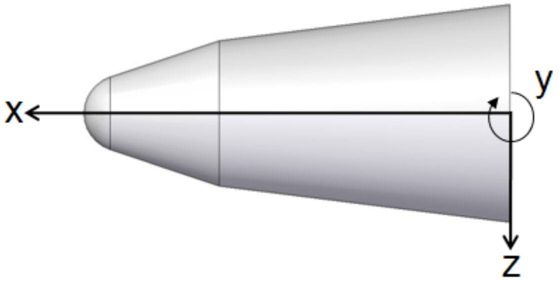


Figure 4: Body axes definition.

$$\mathbf{df} = C_p \; \hat{\mathbf{n}} \; dA \tag{2}$$

$$\mathbf{V}_\infty = [-\cos(\alpha)\cos(\beta) \qquad -\sin(\beta) \qquad -\sin(\alpha)\cos(\beta)]^T \tag{3}$$

$$\begin{bmatrix} C_X \\ C_Y \\ C_Z \end{bmatrix} = \begin{bmatrix} -C_A \\ C_S \\ -C_N \end{bmatrix} = \frac{1}{A_{ref}} \iint_S \begin{bmatrix} \mathbf{df}^T\hat{\mathbf{x}} \\ \mathbf{df}^T\hat{\mathbf{y}} \\ \mathbf{df}^T\hat{\mathbf{z}} \end{bmatrix} = \frac{1}{A_{ref}} \iint_S C_p \begin{bmatrix} \hat{\mathbf{n}}^T\hat{\mathbf{x}} \\ \hat{\mathbf{n}}^T\hat{\mathbf{y}} \\ \hat{\mathbf{n}}^T\hat{\mathbf{z}} \end{bmatrix} dA \tag{4}$$

$$\begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix} = \frac{1}{A_{ref} \cdot l_{ref}} \iint\limits_S \begin{bmatrix} (\mathbf{r} \times \mathbf{df})^T \hat{\mathbf{x}} \\ (\mathbf{r} \times \mathbf{df})^T \hat{\mathbf{y}} \\ (\mathbf{r} \times \mathbf{df})^T \hat{\mathbf{z}} \end{bmatrix} = \frac{1}{A_{ref} \cdot l_{ref}} \iint\limits_S C_p \begin{bmatrix} (\mathbf{r} \times \mathbf{n})^T \hat{\mathbf{x}} \\ (\mathbf{r} \times \mathbf{n})^T \hat{\mathbf{y}} \\ (\mathbf{r} \times \mathbf{n})^T \hat{\mathbf{z}} \end{bmatrix} dA \qquad (5)$$

$$C_{m,\alpha} = \frac{\partial C_m}{\partial \alpha} \qquad (6)$$

$$C_{n,\beta} = \frac{\partial C_n}{\partial \beta} \qquad (7)$$

Newtonian flow theory provides a first-order, computationally efficient means in which the hypersonic aerodynamics of various vehicle geometries can be obtained. The traditional Newtonian approximation is most relevant for slender bodies in which the bow shock follows the contour of the body. However, the traditional theory is typically modified for blunt bodies in which a strong normal shock resides upstream of the vehicle.[21] This modification has led to the development of Modified Newtonian theory in which the leading two in Eq. (1) is reduced to account for the pressure loss across the normal shock. The results detailed in this investigation assume traditional Newtonian flow in which the leading two is maintained. However, should the use of Modified Newtonian flow theory be required, the analytic results in this dissertation can be scaled by the appropriate multiplier. While these analytic aerodynamic relations enable rapid simultaneous hypersonic aerodynamic and trajectory optimization as described in Section 1.1, the Newtonian result is often approximated numerically in current design studies using panel methods.

1.2.3 Panel Methods and CBAERO

As previously mentioned, the Newtonian aerodynamics of hypersonic vehicles is traditionally calculated by integrating the pressure coefficient over the unshadowed surface of the vehicle as described by Eqs. (4) and (5).

In the current state of the art of Newtonian analysis, this surface integration is performed numerically using panel methods that approximate the shape of a vehicle using small flat plates. Thus, the integration is approximated as a finite summation of the integrand over the flat plates approximating the shape of the vehicle. For example, the approximation of C_X is shown in Eq. (8), where ΔA is the area of each panel. This numerical approximation is a source of error in the Newtonian estimate.

$$C_X \approx \frac{1}{A_{ref}} \sum_{i=0}^{N} C_{p,i} \; \mathbf{n_i}^T \hat{\mathbf{x}} \; \Delta A_i \tag{8}$$

Many paneling codes have been developed, including the Hypersonic Arbitrary Body Program (HABP) in conjunction with the Aerodynamic Preliminary Analysis System (APAS) developed in the late 1970s and early 1980s and the Configuration Based Aerodynamics (CBAERO) tool developed in the past decade.[22, 23, 24, 25] CBAERO serves as a means to verify the analytic relations developed in this work. The results of the CBAERO Modified Newtonian calculations are properly scaled to account for the difference in Modified Newtonian and traditional Newtonian theory. CBAERO provides a straightforward methodology to compute the aerodynamic coefficients of relatively complicated geometries. The process required for CBAERO is the following:

1.) Construct a mesh of the vehicle describing the nodes and corresponding flat plates. The construction of a mesh for complicated shapes would require modeling in a CAD package. Bodies of revolution can be meshed fairly easily using CBAERO built-in routines, and bodies derived from potential functions can be meshed using the GNU Triangulated Surface Library (GTS)[26].

2.) Calculate the unit inward normal and C_p for each panel. Any panel with a unit inward normal in the opposite direction from the flow is ignored since it is shadowed.

3.) Calculate the aerodynamic forces through numerical integration of C_p over the surface of the vehicle and generate tables of aerodynamic coefficients for various α and β.

Although CBAERO allows for rapid aerodynamic calculations compared to CFD, this process must be repeated for any change in shape of the vehicle. Additionally, the resolution of the mesh must be addressed when using panel methods. The number of required panels to achieve a desired accuracy is generally unknown in the beginning of the meshing process. Consequently, multiple meshes of various resolutions must be evaluated until the convergence of aerodynamic coefficients is observed. Furthermore, the construction of meshes in CAD packages increases the difficulty of automating hypersonic vehicle shape change necessary for parametric analysis and optimization. These time-consuming issues associated with panel methods limit the number of vehicle shapes analyzed during conceptual design.

1.2.4 Motivation for Analytic Hypersonic Aerodynamics

While panel methods would likely be required for the conceptual design of complicated geometries such as the Space Shuttle Orbiter, X-38, and HL-20, many hypersonic vehicle designs used in previous and current mission studies are not complex. For example, all previous and currently planned Mars missions have used blunt sphere-cones. Various human Mars mission studies have used blunt sphere-cones and blunted biconics.[27, 28, 29] The Stardust and Genesis Earth entries also used blunt sphere-cones.[30, 31] The Apollo and Orion command modules both utilized a spherical forebody segment. Many high performance military hypersonic vehicles are slender sphere-cones and slender biconics with minor nose blunting to account for extreme heating environments. Some high performance entry vehicles include blended wedge designs, such as the SHARP L1, that consist of flat plates, conical frustums, and nose blunting through a cylindrical segment.[32] These examples represent a subset of missions that implement relatively simple vehicle geometries, and Figure 5 illustrates the wide range of applications for these geometries.

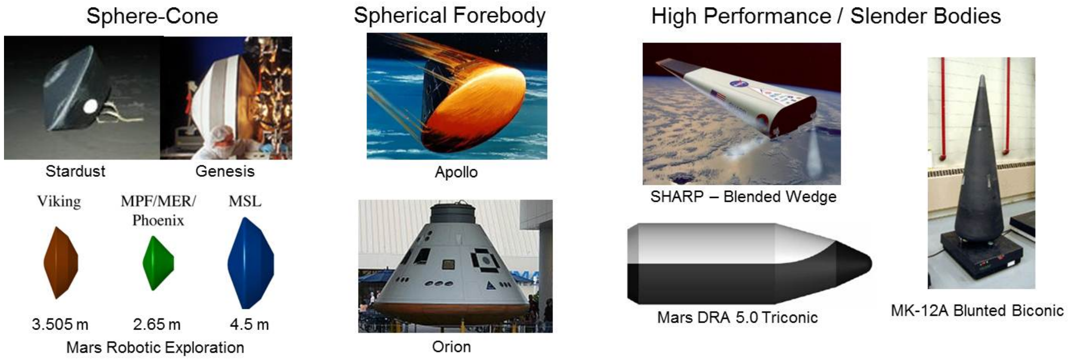


Figure 5: Example vehicles with analytic geometries.

The surface geometry of these basic shapes, along with additional complex shapes, can be expressed analytically. Consequently, the Newtonian surface integration that is traditionally performed numerically using panel methods can also be performed analytically. Many of the resulting analytic relations provide exact Newtonian aerodynamic coefficients currently approximated by panel methods. Additionally, the evaluation of most of the analytic relations is nearly instantaneous. As such, these relations could substitute panel methods widely used in traditional, segregated conceptual design environments to improve the computational performance of Newtonian aerodynamics calculations. More importantly, however, these relations eliminate the large aerodynamic tables as described in Section 1.1 to enable rapid simultaneous hypersonic aerodynamic and trajectory optimization for conceptual design.

1.2.5 Historical Analytic Hypersonic Aerodynamics Overview

Due to the absence of digital computers, analytic hypersonic aerodynamic relations based on Newtonian flow theory were developed in the late 1950s and early 1960s. However, this early work focused on developing analytic relations that could be processed through a computer to construct large aerodynamic tables of various vehicle configurations that consisted of basic shapes, and comparisons with experimental data verified that major trends necessary for conceptual design could be captured with these relations.[33] These large tables were designed for use by hand in the traditional, segregated design environment to quickly estimate aerodynamic characteristics of various sharp and blunted circular conics and elliptical conics at various angles of attack and sideslip.

Newtonian flow theory has also been applied in the past to generic bodies of revolution. Variational methods have been used to determine optimal profiles that minimize the pressure drag of Newtonian flow theory at zero angle of attack for various geometric constraints.[4, 34, 35] The resulting minimum drag bodies of revolution are shown in Figure 6. In order to accommodate nonzero angles of attack, generic bodies of revolution were approximated as a series of open rings at various stations along the axis of revolution.[36, 37, 38, 39] In this work, focus was given to the development of charts and tables that could be used to develop aerodynamic coefficients by numerically evaluating the local inclination of each ring relative to the axis of revolution.

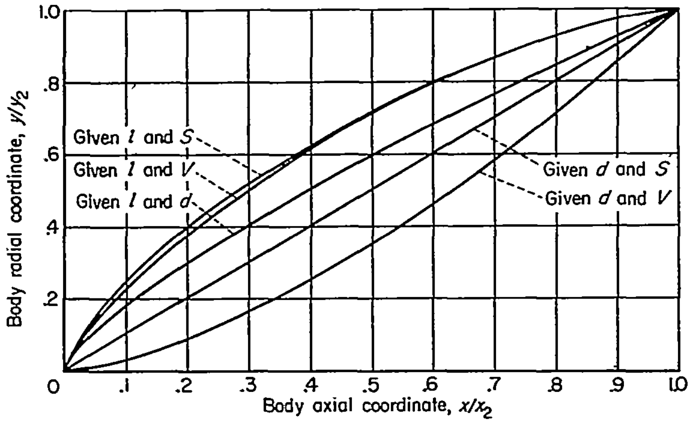


Figure 6: Minimum drag bodies of revolution for various geometric constraints.[4]

1.2.6 Absence of Analytic Aerodynamics in Present-Day Analyses

The analytic relations developed in the 1950s and 1960s have largely been forgotten by the aerospace community in the recent decades. The manual development of analytic relations is time-intensive and requires complex integrations to be performed. As a result, the integration process is largely dependent on integral tables and appropriate substitutions, requiring substantial trial-and-error effort that is prohibitive during conceptual design. Rather, the advent of the digital computer resulted in the widespread adoption of panel and CFD methods over analytic relations. This can be observed by the many recent shape design studies that employ panel methods.[40, 41] While the ability of these methods to model general and complex shapes is desirable, these methods are orders of magnitude slower than evaluating many of the analytic aerodynamic relations. Consequently, the computational requirements of present-day panel methods may limit the number of shapes evaluated during conceptual design.

The rapid evaluation of the analytic relations enables many shapes to be evaluated during conceptual design. Additionally, each basic shape is parametrized to construct a family of similar shapes. As a result of this parametrization, each analytic relation is only developed once and is valid across all parametrized values. This is a major advantage over panel methods that must be executed each time the shape of the vehicle changes. Furthermore, advances in symbolic manipulation tools, such as Mathematica[42] and Maple[43], enable the development of an automated process to construct analytic relations for various shapes. These symbolic tools are capable of querying large databases of integral tables and substitution techniques, addressing some of the limitations that have prevented the adoption of analytic methods.

The analytic relations also eliminate the large aerodynamic tables commonly found in design by allowing vehicle shape parameters to be directly incorporated into the equations of motion. The resulting unified mathematical framework in both trajectory and vehicle shape allows trajectory optimization methods to be extended to also include vehicle shape. By analytically coupling these two traditionally segregated disciplines, a trajectory optimization methodology based on fast, specialized optimization methods is constructed and extended to include vehicle shape. This trajectory optimization methodology substitutes the slower, more general optimization methodologies often used in traditional, segregated design environments to account for the wide range of possible design problems and interactions among contributing analyses.

1.3 Trajectory Optimization Overview

1.3.1 Introduction

In general, hypersonic trajectory optimization is viewed as a difficult task that requires a flyable path to be constructed while simultaneously accounting for possible constraints in initial conditions, terminal conditions, and trajectory constraints such as heating and g-loading. Similar challenges also exist for many other trajectory applications, and as a result, recent advances in trajectory optimization have largely focused on improving the ability of identifying a single solution for these challenging problems. However, conceptual design applications demand numerous optimal trajectories to support various trade studies. For example, the envelope of vehicle performance is often sought during hypersonic conceptual design. This family of optimal trajectories that defines a flight corridor is generally identified through many successive trajectory optimizations. As such, a rapid trajectory optimization methodology is constructed that deviates from the traditional, single-solution approach to quickly construct families of optimal trajectories to support conceptual design studies.

The trajectory optimization problem is often expressed in the form of Eq. (9), where J is the cost functional that is usually minimized, Φ is the terminal cost, and $\int_{t_0}^{t_f} I dt$ is the path cost. Terminal constraints that are typically present for hypersonic missions are expressed in the form of Eq. (10). Finally, the equations of motion are given in the form of Eq. (11).

$$J = \Phi[\mathbf{x}(t_f), t_f] + \int_{t_0}^{t_f} I(\mathbf{x}(t), \mathbf{u}(t), t)dt \tag{9}$$

$$\mathbf{\Psi}[\mathbf{x}(t_f), t_f] = 0 \tag{10}$$

$$\dot{\mathbf{x}} = f[\mathbf{x}(t), \mathbf{u}(t), t], \quad t_0 \text{ given} \tag{11}$$

Designers are generally forced to choose between direct and indirect methods when solving this problem. These methods will be described in greater detail, but the advantages and disadvantages of both approaches are highlighted in Table 1. In summary, direct methods are desired when performing trajectory design for a wide range of hypersonic problems. However, the rapid convergence of indirect methods is desired during conceptual design when constructing families of optimal trajectories.

Table 1: Comparison between direct and indirect methods.

	Advantages	*Disadvantages*
Direct Methods	Large region of attraction Widespread NLP solvers exist	Computationally intensive Optimality not guaranteed
Indirect Methods	Rapid convergence Necessary conditions satisfied	Small region of attraction Costates introduced

1.3.2 Indirect Methods

Prior to modern computing, indirect methods were used to obtain analytic solutions to simple optimal control problems such as the Brachistochrone problem.[34] For certain problems, these solutions can be generalized as functions of trajectory parameters. Indirect methods are used to identify an extremum of the functional J[34, 44] and require solution to a multi-point boundary value problem (BVP) that is formulated from the first order necessary conditions of optimality.[1, 45] Indirect methods augment the original system of equations of motion with costates that increase both the dimensionality and complexity of the problem.

The Hamiltonian, H, is defined as shown in Eq. (12), where $\boldsymbol{\lambda}$ is the vector of costates with trajectories defined by Eq. (13). The optimal control law can be obtained by stationarizing the Hamiltonian with respect to control as shown in Eq. (14). The boundary conditions at the beginning of the trajectory require the costates associated with free states to vanish as shown in Eq. (15), and the terminal boundary conditions are shown in Eq. (16), where $\boldsymbol{\nu}$ is a vector of undetermined constants. Additionally, the time of the hypersonic trajectories will vary, requiring the free final time condition shown in Eq. (17).

$$H = I(\mathbf{x}, \mathbf{u}, t) + \boldsymbol{\lambda}^T(t)\mathbf{f}(\mathbf{x}, \mathbf{u}, t) \tag{12}$$

$$\dot{\boldsymbol{\lambda}} = -\frac{\partial H}{\partial \mathbf{x}} \tag{13}$$

$$\left(\frac{\partial H}{\partial \mathbf{u}}\right)^T = 0 \tag{14}$$

$$\mathbf{x}(t_0) \text{ given} \qquad \text{or} \qquad \boldsymbol{\lambda}(t_0) = 0 \tag{15}$$

$$\boldsymbol{\lambda}(t_f) = \left(\frac{\partial \phi}{\partial \mathbf{x}} + \boldsymbol{\nu}^T \frac{\partial \boldsymbol{\psi}}{\partial \mathbf{x}}\right)^T_{t=t_f} \tag{16}$$

$$\left[\frac{\partial \phi}{\partial t} + \boldsymbol{\nu}^T \frac{\partial \boldsymbol{\psi}}{\partial t} + \left(\frac{\partial \phi}{\partial \mathbf{x}} + \boldsymbol{\nu}^T \frac{\partial \boldsymbol{\psi}}{\partial \mathbf{x}}\right)\mathbf{f} + I\right]_{t=t_f} = 0 \tag{17}$$

The introduction of path constraints further increases the complexity of the optimization problem through the addition of corner conditions and switching structure. Sensed deceleration and stagnation point heat rate constraints shown in Eq. (18) and Eq. (19), respectively, are common path constraints for hypersonic missions, where L is the lift of the vehicle, D is the drag of the vehicle, m is the mass of the vehicle, g_{max} is the maximum g-loading constraint, ρ is the atmospheric density, r_n is the nose radius of the vehicle, v is the velocity of the vehicle, $\dot{q}_{max}$ is the maximum stagnation heat rate constraint, and k is an empirically derived multiplier. These path constraints are generally expressed as inequality constraints of the form shown in Eq. (20).

$$\frac{\sqrt{L^2 + D^2}}{9.81m} - g_{max} \leq 0 \tag{18}$$

$$k\sqrt{\frac{\rho}{r_n}}v^3 - \dot{q}_{max} \leq 0 \tag{19}$$

$$\mathbf{S}(\mathbf{x}, t) \leq 0 \tag{20}$$

To obtain the control history along the constraint, time derivatives of the constraint are performed until the control explicitly appears. Assuming this occurs with the q^{th} total time derivative, the Hamiltonian is augmented as shown in Eq. (21), and the control law is obtained from $\mathbf{S}^{(q)} = 0$ on the constraint boundary.

$$H = I + \boldsymbol{\lambda}^T \mathbf{f} + \boldsymbol{\mu}^T \mathbf{S}^{(q)} \tag{21}$$

The introduction of path constraints also modifies the trajectories of the costates as shown in Eq. (22), where the multiplier $\boldsymbol{\mu}$ is calculated using Eq. (23).

$$\dot{\boldsymbol{\lambda}}^T = -H_{\mathbf{x}} = -I_{\mathbf{x}} - \boldsymbol{\lambda}^T \mathbf{f}_{\mathbf{x}} - \boldsymbol{\mu}^T \mathbf{S}_{\mathbf{x}} \tag{22}$$

$$H_{\mathbf{u}} = I_{\mathbf{u}} + \boldsymbol{\lambda}^T \mathbf{f}_{\mathbf{u}} + \boldsymbol{\mu} \mathbf{S}_{\mathbf{u}} = 0 \tag{23}$$

At the entrance of the constraint, a series of tangency conditions shown in Eq. (24) is enforced. Additionally, corner conditions at the entrance and exit of each path constraint must also be satisfied. The corner conditions were chosen to be continuous at the exit as shown in Eq. (25) and Eq. (26). As a result, the corresponding corner conditions at the entrance, shown in Eq. (27) and Eq. (28), must also be satisfied.

$$\mathbf{N}(\mathbf{x}, t) = \begin{bmatrix} \mathbf{S}(\mathbf{x}, t) \\ \mathbf{S}^{(1)}(\mathbf{x}, t) \\ . \\ . \\ . \\ \mathbf{S}^{(q-1)}(\mathbf{x}, t) \end{bmatrix} \tag{24}$$

$$\boldsymbol{\lambda}(t^-) = \boldsymbol{\lambda}(t^+) \tag{25}$$

$$H(t^-) = H(t^+) \tag{26}$$

$$\boldsymbol{\lambda}(t^-) = \boldsymbol{\lambda}(t^+) + \boldsymbol{\pi}^T \mathbf{N}_{\mathbf{x}} \tag{27}$$

$$H(t^-) = H(t^+) - \boldsymbol{\pi}^T \mathbf{N}_t \tag{28}$$

When the vehicle is not flying along the path constraint, the control can take on any feasible value, and the optimal solution is not apparent. The optimal control along these unconstrained arcs is determined through the use of a switching structure developed from Pontryagin's Minimum Principle (PMP). PMP states that the Hamiltonian must be minimized with respect to **u** for all time. For bounded controls, a switching structure can be constructed that usually resembles a bang-bang control policy that is generally a function of states, costates, and parameters of the problem.

The implementation of indirect methods can be challenging. The derivation of the necessary conditions of optimality is an intensive procedure that must be repeated for each optimization problem. However, this process is fairly straightforward and mostly requires the computation of predefined partial derivatives. To converge to a solution of the multipoint boundary value problem, a good initial guess in states, costates, and corner conditions must be constructed. In many applications, costates do not have physical meaning, prohibiting the development of a good initial guess. However, physical meaning has been successfully assigned to costates in certain situations. For example, in low-thrust astrodynamic applications, primer-vector theory relates the direction of the thrust to the direction of the costate vector.[46] Through adjoint control transformation (ACT), a mapping between the costates and both the direction and rate of change in direction of the thrust vector has also been constructed.[47, 48] Furthermore, a straightforward method can be used to determine the costates for unconstrained trajectory optimization problems. For this class of problems, a parameter optimization in discrete control values can be performed to identify the states of the optimal trajectory. After propagating the trajectory forward in time, the costates can be calculated through reverse integration from the terminal point along the optimal trajectory.[49, 50]

During the conceptual design of various constrained hypersonic missions, the designer may not be capable of providing a sufficient initial guess required for convergence of indirect methods. Additionally, the complex optimization of hypersonic trajectories requires the use of computer-based BVP methods, including shooting methods and collocation. The solution to the multipoint BVP can be difficult to obtain since one solution, many solutions, infinitely many solutions, or no solutions may exist. As a result, indirect methods are difficult to automate over a wide range of optimal control problems. This is especially problematic in the traditional, segregated design environment that requires a flexible trajectory optimization algorithm capable of providing solutions for a wide range of potential results from other contributing analyses. Consequently, direct methods have gained popularity for solving modern optimal control problems, particularly those defined in the conceptual design phase.

1.3.3 Direct Methods

Modern trajectory optimization problems of increasing complexity are often solved using direct methods. Instead of deriving the necessary conditions to stationarize the functional J , direct methods approximate the continuous control history and/or states with a finite set of discretized values. This allows various optimization methods to be directly applied to the optimal control problem. The most straightforward approach discretizes the control history, and an external optimizer manipulates the discrete control history to identify an optimal trajectory. During this search, a simulation is performed for each candidate control history to construct a full trajectory, and corrections are made based on violations of constraints and performance of the objective function.

More elaborate direct methods have been developed that intelligently discretize the state and/or control using an appropriate function approximation.[51, 52] Various forms of direct methods have been developed, including collocation[53, 54, 55] and differential inclusion[56, 57, 58]. The application of these carefully selected quadrature rules results in a discrete nonlinear optimization problem. The structure of these nonlinear optimization problems is often very sparse, and nonlinear programming (NLP) solvers, such as SNOPT[59], have been greatly advanced over time to efficiently solve this problem. As a result, indirect methods have largely been replaced by direct methods for conceptual design.

1.3.3.1 Integration Methods and Evolutionary Algorithms

Even when only the control history is discretized, a full factorial search of all possible control history combinations is generally impractical in hypersonic vehicle design. For example, assuming a hypersonic bank profile optimization in which the bank profile is discretized at ten bank points with five degree separation, a full factorial search would require the evaluation of $72^{10} \approx 4.81 \times 10^{15}$ candidate solutions. Furthermore, many hypersonic trajectory problems are multimodal, adding complexity to the optimization process. Hence, global search methods such as genetic algorithms (GA) and particle swarm optimization (PSO) algorithms have been developed to automate this optimization process and explore the design space for global optimal solutions.[60, 61] These methods eliminate the need for a good initial guess which is of benefit for the conceptual design of various hypersonic missions. These methods have also been used to simultaneously solve for Pareto frontiers, or optimal families, of hypersonic trajectories.[60] This departure from solving a single optimal trajectory to solving a family of optimal trajectories illustrates the need of designers to construct many optimal solutions as described in Section 1.3.1.

Although these algorithms have many advantages that generalize the trajectory optimization problem, several disadvantages limit their use for conceptual design. First, each algorithm has a unique set of tuning parameters that strongly influence the efficiency of the searching process. If these tuning parameters are not set at appropriate values, a sufficient global search may not be performed. Second, the generic optimization formulation of these algorithms ignores the physics of the problem, requiring many iterations that correspond to many propagated trajectories during the global search of these population-based methods. Third, no guarantee can be made about the global or local optimality of the final solution. Finally, these algorithms inefficiently account for path constraints indirectly through manipulation of the control history. In many applications, penalty functions are used to augment the objective function with path constraints and boundary conditions. Penalty functions convert the constrained optimization problem into an unconstrained optimization problem, eliminating the added complexity associated with finding constrained optima. In many applications, the designer is responsible for developing appropriate penalty functions that sufficiently balance the original objectives with the active constraints. Therefore, the construction of appropriate penalty functions is largely problem dependent and may require multiple iterations to achieve the desired balance. This manual and iterative process is not conducive for the conceptual design of families of hypersonic trajectories.

For unconstrained targeting problems, single and multiple shooting methods have been widely adopted. These methods generally discretize the control history and propagate the equations of motion from various initial states. In order to properly target the desired terminal conditions of the trajectory, a correction scheme is constructed from the state transition matrix (STM).[62]

1.3.3.2 Collocation Methods

Collocation methods have been developed as an alternative to integration methods.[53] Unlike explicit integration methods, collocation methods discretize both controls and states along the trajectory. The expansion of the optimization problem through the additional design variables from the discretization of state would initially appear as a disadvantage to the trajectory optimization process. However, the equations of motion are enforced at the discretized locations, resulting in a continuous trajectory without the need to explicitly integrate the equations of motion. Additional interior points are also selected to enforce a quadrature scheme with a desired order of accuracy. A common collocation method interpolates the state using a cubic polynomial and interpolates control using a linear function, and the equations of motion are enforced at the endpoints and midpoint of each segment. When converged, it can be shown that this solution is identical to a 4th order explicit Runge-Kutta integration scheme often used in integration methods.[58] As a result, collocation methods are often viewed as implicit integration methods.

If constraints are present, then the cost function is augmented according to the Karush-Kuhn-Tucker (KKT) conditions[63], and the trajectory optimization problem is converted to a constrained nonlinear programming (NLP) problem. This enables the use of NLP techniques to perform trajectory optimization. Many of these techniques make use of the sparse matrix structure that often results from this method. For example, SNOPT implements a sequential quadratic programming (SQP) algorithm to find locally optimal solutions. The search is performed by minimizing a quadratic model of the Lagrangian function subject to locally linearized constraints.

Alternative to integration methods, the direct manipulation of the states provided by collocation methods allows in-flight constraints to be easily satisfied. Collocation methods have been successfully applied to constrained entry trajectory applications[54] and prior work has enabled real-time construction of constrained X-34 autolanding trajectories initiated from various states[64, 65]. However, the constraint advantages provided by collocation methods are achieved at the expense of increasing the dimensionality of the optimization problem through the added discretization of states. As such, differential inclusion was designed to reduce the dimensionality of the optimization problem when compared to collocation methods while simultaneously retaining the benefits these methods provide.

1.3.3.3 Differential Inclusion

The computational requirements to solve each NLP problem is largely dependent on the dimensionality and choice in discretization of the trajectory.[66] Differential inclusion reduces the dimensionality of the discretized problem through elimination of control variables.[56, 57] As control variables are eliminated, they are replaced with constraints in state rates. Therefore, the dimensionality of the NLP problem is reduced at the penalty of additional constraints. This process can only be performed if the control variables can be solved as a function of state variables and time derivatives of state variables. Therefore, differential inclusion can only be applied to a subset of trajectory optimization problems[67, 68, 69], including bank-only hypersonic trajectory problems.

The reduction in dimensionality achieved by differential inclusion is appealing and highlights an important aspect of hypersonic trajectory design. Designers often do not care about the actual control history associated with optimal trajectories as long as sufficient margin is maintained. Instead, designers are interested in the limitations in trajectory options that result from limitations in control. However, differential inclusion is limited to explicit quadrature rules, and original work approximated state rates using a forward Euler approximation.[56, 57] When implicit quadrature rules are used, it is not possible to perform the necessary isolation in state rates required by differential inclusion. Alternatively, collocation methods are not limited to explicit quadrature rules and, hence, can achieve high orders of accuracy using numerical schemes including implicit Runge-Kutta [51] and Gauss-Lobatto[55]. Furthermore, higher-dimension collocation methods have been shown to require sufficiently fewer discretized trajectory nodes than differential inclusion methods to achieve the same level of accuracy.[58] As a result, collocation methods have gained popularity over differential inclusion.

1.3.4 Pseudospectral Methods

During conceptual design, the trajectory is often discretized without regard to the numerical method chosen. Furthermore, the assumption is often made that as the number of discretized points increases, the discrete solution will approach the continuous solution. While this is not always the case[70], solutions are often sufficient for practical engineering applications. As a result, many of the aforementioned direct methods have been constructed without consideration of this assumption. Prior to the development of pseudospectral methods, trajectory designers were often forced to use either a direct or indirect method. Direct methods have been widely adopted due to ease of implementation for a wide range of trajectory optimization problems. However, the rapid convergence of indirect methods is desired for design studies. As such, pseudospectral methods provide an opportunity to capitalize on the strengths of both approaches during conceptual design.

If the designer is presented with a general trajectory optimization problem represented as Problem B shown in Figure 7, the designer could choose to apply an indirect method, arriving to problem $B^{\lambda N}$, or a direct method, arriving to problem $B^{N\lambda}$, where N represents the discretization process and λ represents the application of optimality conditions. As illustrated, the procedures of the direct and indirect methods are inverted. Direct methods first discretize the problem and apply optimality conditions (KKT conditions) to the discrete problem. Alternatively, indirect methods apply optimality conditions to the continuous problem through the adjoint of costates, and the resulting augmented problem is discretized to solve for a computer-based solution. Thus, the discrete costates of indirect methods are analogous to the KKT multipliers of direct methods. However, previous studies have shown that the KKT multipliers of direct methods and the discrete costates of indirect methods may not be consistent even if the trajectories and control histories are in agreement.[71] Such results would seem to imply that the discretization and application of optimality conditions (dualization) is not commutable.

It can be shown that the cause for this discrepancy is rooted in the nonuniqueness of the KKT multipliers for certain problems.[71] However, through proper choice in discretization, a mapping between the KKT multipliers and discrete costates can be constructed. The recent development of pseudospectral methods that make use of this proper choice in discretization has formalized this mapping through the Covector Mapping Theorem (CMT).[72] The CMT provides a set of closure conditions that enforce a unique set of KKT multipliers that are consistent with the discrete costates

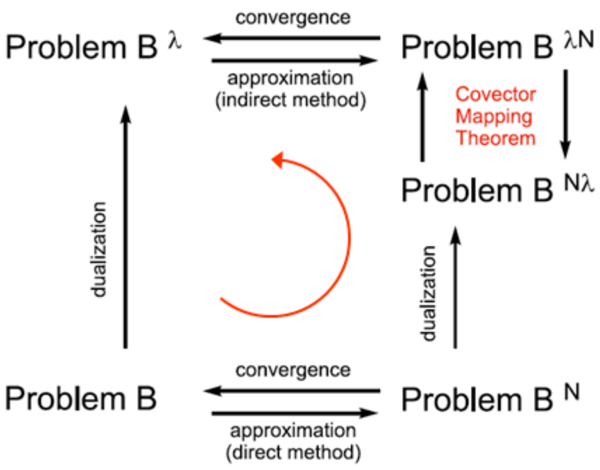


Figure 7: Covector Mapping Theorem diagram.[5]

of indirect methods after proper scaling. In many pseudospectral applications, the costates computed from the CMT are used to validate the necessary conditions of optimality provided by indirect methods[73] as suggested by prior work in collocation methods[55]. Furthermore, pseudospectral methods provide the opportunity to use intuitive direct methods to arrive to a converged indirect solution that is otherwise difficult to obtain. Starting with this initial indirect solution, designers may have the opportunity to perform design studies in a shorter time frame through the continuation of fast indirect methods. However, this is only practical if the optimal solutions to hypersonic trajectories are well connected, allowing sufficiently large changes in trajectory parameters necessary for trade studies.

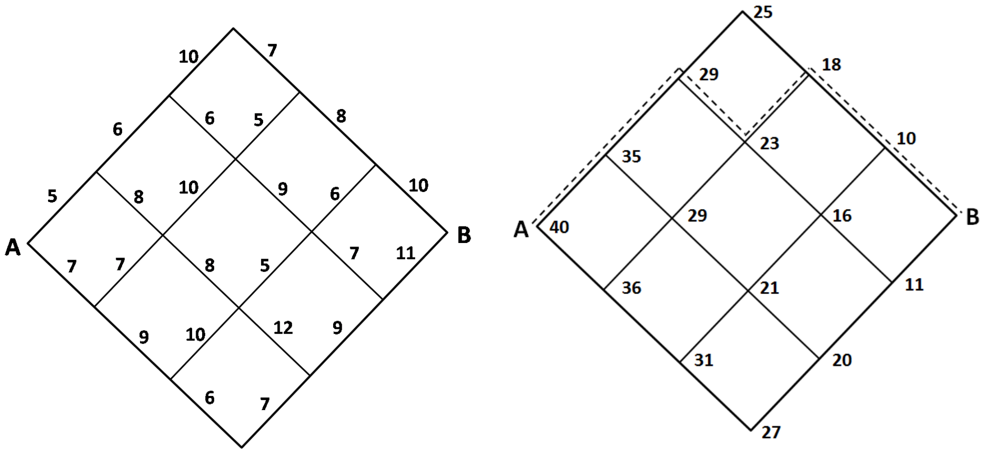
The common Legendre pseudospectral method uses global polynomial approximations for the state functions with discretized nodes at the Legendre-Gauss-Lobatto (LGL) points.[74] Unlike traditional collocation methods, the global polynomials enable high-order LGL quadrature rules. The LGL points are arranged in a non-uniform distribution along the trajectory, and the dense distribution of nodes near the endpoints also avoids large interpolation errors that may result from the Runge phenomenon.[73] For this pseudospectral implementation, the costates can be obtained by dividing the KKT multipliers by the LGL weights.[72]

While pseudospectral methods can serve as the foundation to perform rapid trajectory optimization through the continuation of indirect methods, several implementation concerns remain. These include the need to identify LGL node locations using computationally efficient, numerical linear algebra techniques[75] and noisy costate estimation[72]. As with any local searching method, pseudospectral methods also require a good initial guess in both states and control to converge to an optimal solution that is in the region of interest to the designer. However, the required control history to fly optimal trajectories is often of little interest to the designer, as long as it remains reasonably within control authority limitations. From this viewpoint, the trajectory optimization problem can be converted into a pathfinding problem, allowing efficient construction of the initial guess using dynamic programming.

1.3.5 Dynamic Programming

Dynamic programming (DP) is an efficient methodology to solve certain multistage optimization problems[1, 76, 77, 78, 79] and is based on Bellman's Principle of Optimality[80, 81, 82]. Using this approach, large optimization problems are reduced to smaller subproblems in which each trajectory segment is only evaluated once. An example of discrete dynamic programming is shown in Figure 8.[1] Figure 8(a) depicts the cost associated with traveling between adjacent nodes, and Figure 8(b) provides the optimal cost associated with traveling along the optimal path from each node to the terminal point, B.

The tree of optimal paths is efficiently constructed in reverse from the terminal point by leveraging optimal trajectory information from prior solutions as the trajectories are extended to the initial point, A. In this example, there is complete freedom to travel to any adjacent node. As such, the global optimal path from A to B is



(a) Discrete step cost.

(b) Discrete path cost.

Figure 8: Dynamic programming example.[1]

efficiently identified without evaluating all possible paths, as shown in Table 2 where 15 computations are required with 20 possible paths in this example.[1] As the mesh of the discrete paths from A to B increases in size, the advantages of reducing the number of computations through dynamic programming is evident. If the trajectory space is discretized in a similar manner, then dynamic programming could be used to efficiently construct an initial guess required by pseudospectral methods. While this initial guess would ignore control authority limitations when traveling from node to node, this globally optimal path would guide the search of the pseudospectral method to the general region of most interest to the designer.

Table 2: Dynamic programming computations compared to total possible routes.[1]

Segments on a Side	3	4	5	6	7	n
Possible Routes	20	70	252	724	2632	$(2n)!/n!n!$
Computations	15	24	35	48	63	$(n+1)^2-1$

1.3.6 Trajectory Optimization Tools

Various methods have been implemented in trajectory optimization software. At NASA, a variety of trajectory design tools have gained popularity, including the Program to Optimize Simulated Trajectories (POST)[83], Optimal Trajectories by Implicit Simulation (OTIS)[52], and Simulation and Optimization of Rocket Trajectories (SORT)[84]. The current version of POST presents the designer with the option to use a projected gradient algorithm or a sequential quadratic programming algorithm to perform trajectory optimization. POST has been widely used for trajectory design and validation of Mars entries[85], aerocapture missions[86, 87, 88], and hypersonic flight vehicles[89]. The current iteration of OTIS, version 4, has the ability to solve trajectory optimization problems using collocation or pseudospectral methods using SLSQP[90] and SNOPT. OTIS has been successfully used to optimize trajectories over a wide range of applications, including Orion ascent abort[91], low-thrust mission planning[92], single-stage-to-orbit missions[93], and multistage exo-atmospheric ascent trajectories[94].

Various tools have also been developed to perform trajectory optimization using pseudospectral methods, including DIDO[95] and the Gauss Pseudospectral Optimization Software (GPOPS)[96]. DIDO discretizes the trajectory at the Legendre-Gauss-Lobatto points whereas GPOPS selects the Legendre-Gauss points. DIDO has been successfully implemented in low thrust applications[97], spacecraft formations[98], rigid body reorientation[99], launch vehicles[100], and the zero propellant maneuver on the International Space Station[101]. GPOPS has been successfully implemented for spacecraft formations[102, 103, 104] and aeroassisted orbital transfers[105, 106].

All of the tools listed in this section represent candidate trajectory analyses in the traditional, segregated design environment described in Section 1.1. Alternatively, a rapid trajectory optimization methodology based on the continuation of successive indirect solutions is developed in this dissertation. The hypothesis that hypersonic trajectories can be efficiently connected in this manner is initially tested using the indirect solution seeded by a pseudospectral method. After validating the computational advantages of this approach, the trajectory optimization problem is extended to also include vehicle shape. The integration of vehicle shape parameters into the equations of motion, made possible by the development of the analytic hypersonic aerodynamic relations, enables the construction of a unified, mathematical framework to perform rapid simultaneous hypersonic aerodynamic and trajectory optimization.

1.4 Outline

Chapter 2 details the development of the analytic hypersonic aerodynamic relations based on Newtonian flow theory. The initial relations are constructed using the original model of Newtonian flow theory and are validated using CBAERO. To address the mathematical limitations of this approach, additional relations are developed by converting the traditional pressure calculation over the surface of the vehicle to a divergence calculation throughout the volume of the vehicle. In Chapter 3, a trajectory optimization methodology suitable for conceptual design is developed by relying on the continuation of solutions found via indirect methods. Performance comparisons are made relative to a pseudospectral method to highlight the computational advantages of this new approach. A unified mathematical framework is constructed in Chapter 4 to perform rapid, simultaneous hypersonic aerodynamic and trajectory optimization by extending the trajectory design methodology from Chapter 3 to also include vehicle shape using the analytic relations developed in Chapter 2. To illustrate the computational advantages of this unified framework, comparisons are made to a representative traditional design optimization algorithm. Finally, Chapter 5 summarizes the contributions of this dissertation and details the suggested future work in analytic hypersonic aerodynamics, rapid trajectory optimization, unified design techniques, and onboard applications.

1.5 Contributions of Thesis

The body of work presented herein advances the state of the art in the conceptual design of hypersonic vehicles with additional application to real-time trajectory planning. This is accomplished by providing a capability to rapidly identify optimal trajectories and vehicle configurations necessary to accomplish various mission objectives and satisfy a variety of mission requirements. The following summary lists the contributions of this research.

The departure from numerical hypersonic aerodynamic modeling to analytic methods: Hypersonic aerodynamic modeling is typically performed numerically during conceptual design using panel methods based on Newtonian flow theory. While these methods allow for straightforward analysis, the construction of large aerodynamic tables is computationally intensive and is a source of numerical error and discontinuity requiring iteration in the design process. Additionally, new tables must be constructed for each change in vehicle shape. In this investigation, a process has been developed to overcome the limitations of numerical methods by analytically deriving hypersonic aerodynamic coefficients of user-defined geometries. As such, a database is created that expands upon the few shapes developed within historical analytic work. Many of the relations contained within the database provide exact solutions that are currently approximated by panel methods. Additionally, the majority of the resulting relations can be evaluated nearly instantaneously, and these relations provide an analytic mapping of vehicle shape to trajectory performance. This analytic mapping enables vehicle shape to be directly incorporated into the rapid trajectory optimization framework to include system-level metrics such as payload volume. Although analytic relations are applicable to a wide variety of shapes, the limit in generality using analytic relations is addressed.

The extension of traditional optimal control theory to perform rapid

trajectory planning: Traditionally during mission design, the flight corridor is identified through the independent optimization of several trajectories with significant designer interaction. A methodology based on the continuation of fast indirect methods is presented to automate the construction of the flight corridor while simultaneously reducing computational requirements compared to current state of the art techniques. To validate that hypersonic trajectories can be efficiently connected in this manner, an initial rapid trajectory optimization methodology is constructed that is the first to execute a sequence of techniques including discrete dynamic programming, nonlinear inversion, pseudospectral methods, indirect methods, and continuation. While this process is likely required to perform rapid trajectory optimization in regions of the design space with high sensitivities, a simplified methodology is also constructed that is based on the continuation of indirect methods alone. Examples illustrate the computational advantages of this simplified approach for trajectory solutions that vary continuously with changes in trajectory parameters. This approach also has the potential to enable onboard real-time trajectory planning in which optimal trajectories must be rapidly and efficiently computed using the limited computational resources provided by flight computers. By combining these two approaches, a robust rapid trajectory optimization methodology suitable for conceptual design is constructed. This methodology enables rapid trades in trajectory parameters to be performed when compared to trajectory optimization tools traditionally used in segregated design environments. By extending this rapid trajectory methodology to also include vehicle shape through the development of the analytic aerodynamic relations, a unified design framework is constructed to perform rapid simultaneous hypersonic aerodynamic and trajectory optimization.

The development of a rapid hypersonic aerodynamic and trajectory design methodology: The connectedness of solutions illustrated by the rapid trajectory optimization methodology can be extended to other disciplines as a general design philosophy. For this dissertation, the rapid trajectory methodology is extended to also include vehicle shape, a process made possible by the development of analytic hypersonic aerodynamic relations. The resulting unified, mathematical framework enables interactions among trajectory and vehicle shape to be efficiently and simultaneously calculated. This is in direct contrast to the current state of the art design methodologies that rely on segregated disciplinary analyses. Rather than capitalizing on the efficiency of a unified, mathematical framework, these analyses are often designed to be sufficiently flexible to account for any potential interaction with other disciplinary analyses. With the unified design framework, solutions to hypersonic missions that were traditionally solved independently can now be rapidly identified by exploiting the connectivity among solutions. Examples demonstrate that this connectivity includes missions at different celestial bodies with varying gravities and atmospheres as well as missions with various vehicle configurations and requirements.

CHAPTER II

ANALYTIC HYPERSONIC AERODYNAMICS

2.1 A Note for the Reader on the Analytic Expressions

The analytic aerodynamics expressions discussed in this chapter can be downloaded from the following file when viewed in Adobe Acrobat (aerodynamicsDatabase.zipREMOVEME)[1]. The enclosed Matlab routines should be referenced for the contents of the analytic expressions. Due to the prohibitive length of many of the equations, only a select number of relations are provided as examples in Appendix A.

2.2 Introduction

During conceptual design, the hypersonic aerodynamics of vehicles is often calculated using Newtonian flow theory. This theory assumes fluid particles travel in rectilinear motion until they impact the vehicle. As such, the pressure coefficient is assumed to solely result from the momentum transfer between these fluid particles and the surface of the vehicle and, as a result, is only a function of the relative orientation of the vehicle's surface to the flow. These assumptions are reasonable for hypersonic flight, and the aerodynamic coefficients are traditionally computed by integrating the pressure coefficient over the surface of the vehicle. Shadowed regions of the vehicle that are not directly exposed to the flow are eliminated from this integration. Panel methods have been widely adopted to perform this surface integration numerically by approximating the surface of general vehicle configurations using small flat plates. As a result, the hypersonic aerodynamic coefficients of vehicles calculated in this manner

[1] Prior to unzipping the file, the 'REMOVEME' portion of the file extension must first be removed.

are approximated using large tables that are a function of angle of attack and sideslip. These methods are necessary for complicated shapes including the Space Shuttle Orbiter and X-38. However, the geometry of many common hypersonic vehicles of interest such as sphere-cones, blunted biconics, and spherical forebody segments can be expressed analytically. As a result, the aerodynamic relations can also be developed analytically. Note that analytic aerodynamic relations, like those derived in this investigation, are possible in any flow regime in which the flowfield model can be accurately described analytically.

Many of the analytic aerodynamic relations described in this dissertation provide exact solutions to Newtonian flow theory, eliminating the computational error associated with panel methods. Additionally, the evaluation of many of these relations is nearly instantaneous and improves upon the computational performance of panel methods. As a result, these relations can be used to instantaneously model configuration changes during flight and shape change due to ablation. While fielded systems would use large tables derived from CFD and testing for onboard aerodynamic prediction, the analytic relations could also serve as a substitute during the conceptual design of onboard, real-time trajectory planning algorithms. Finally, these relations provide an analytic mapping of vehicle shape to trajectory performance, enabling the construction of a unified mathematical framework that improves upon traditional vehicle and trajectory design methodologies.

2.3 Process for Development of Analytic Relations

An integrated Matlab and Mathematica environment has been constructed to automate the development of analytic relations for user-supplied shapes. Matlab is used to drive the process and employs Mathematica's symbolic engine to perform the integrations. Mathematica was chosen due to its ability to add constraints on symbolic variables (for example, the radius of a sphere is always greater than zero). Supplying this information to Mathematica as an assumption influences how the integration is performed. After the designer describes the surface of the shape, routines containing the analytic aerodynamic relations are generated in a Matlab-based aerodynamics module provided in Section 2.1. This module can be easily integrated into trajectory simulations, be used for parametric analyses, or be used in shape optimization. The six steps performed in this automated process are further detailed.

2.3.1 Step 1: Surface Parametrization

The analytic aerodynamic expressions are obtained by integrating C_p over the unshadowed surface of the vehicle as described in Eq. (4) and Eq. (5). In order to evaluate these integrals, the surface of each shape is parametrized by two independent variables via a position vector, $\mathbf{r}$, as shown in Eq. (29), where $f(u,v)$, $g(u,v)$, and $h(u,v)$ describe the x, y, and z location of a point on the surface of the vehicle as a function of the surface parametrization (u,v). The choice in parametrization variables, u and v, is largely at the discretion of the designer. However, the choice in parametrization can dramatically influence the ability of Mathematica to obtain closed-form solutions when performing the integrations. Additionally, due to the convention used when computing the surface normal, u and v must be chosen such that $\mathbf{r}_u \times \mathbf{r}_v$ is pointed inward, where $\mathbf{r}_u = \frac{\partial \mathbf{r}}{\partial u}$ and $\mathbf{r}_v = \frac{\partial \mathbf{r}}{\partial v}$. The choice in parametrization also influences the expression for the differential area, dA, of the integrations. The differential area is computed using the magnitude of the inward normal vector as shown in Eq. (30).

$$\mathbf{r} = [f(u,v) \quad g(u,v) \quad h(u,v)]^T \tag{29}$$

$$dA = \parallel \mathbf{n} \parallel = \parallel \mathbf{r}_u \times \mathbf{r}_v \parallel \tag{30}$$

2.3.2 Step 2: Compute Pressure Coefficient

With the surface parametrized by u and v, the pressure coefficient can be computed. Recall from Figure 2 that $\sin(\theta)$ is defined as shown in Eq. (31), where $\hat{\mathbf{n}}$ is calculated from Eq. (32) and $\mathbf{V}_\infty$ is defined in Eq. (3) of Section 1.2.2. With $\sin(\theta)$ known, C_p can be computed using Eq. (1).

$$\sin(\theta) = \hat{\mathbf{V}}_\infty^T \hat{\mathbf{n}} \tag{31}$$

$$\hat{\mathbf{n}} = \frac{\mathbf{r_u} \times \mathbf{r_v}}{\| \mathbf{r_u} \times \mathbf{r_v} \|} \tag{32}$$

2.3.3 Step 3: Compute Shadow Boundary

The major challenge in deriving analytic aerodynamic expressions is ensuring that the integration is not performed over shadowed regions of the vehicle where $C_p = 0$. With an analytic expression for $\sin(\theta)$, the shadow boundary can be computed by solving $\sin(\theta) = 0$ for v as a function of u since the surface integration is first performed with respect to v. Note that the solution to this equation may have multiple results, especially if the surface is parametrized with trigonometric functions. A numerical test is performed to determine which solutions should be incorporated as the lower and upper bounds. Note that the limits of integration are a function of vehicle shape and flow direction. Only convex shapes are currently supported. This allows the limits of integration in v to be defined by the solution of $\sin(\theta) = 0$. If the shape was not convex, then shadowed regions may have boundaries where $\sin(\theta) \neq 0$.

2.3.4 Step 4: Compute Reference Values

The reference area and reference length for each shape is computed based on the parametrization used. The reference area is computed as the projected area of the shape on the y-z plane using Eq. (33) assuming $\alpha = \beta = 0$, where dA_{yz} is the differential area projected on the y-z plane. Since $\mathbf{r}_u \times \mathbf{r}_v$ represents the inward normal of a convex shape and dA_{yz} must be positive to obtain a positive reference area, dA_{yz} is computed using Eq. (34). Due to the convexity requirement of the shape, the entire forward surface is unshadowed at $\alpha = \beta = 0$, and, consequently, the limits of integration must be chosen to span the entire surface of the shape. The reference length is computed as the maximum span of the vehicle in the x-direction.

$$A_{ref} = \int_{u_{min}}^{u_{max}} \int_{v_{min}}^{v_{max}} dA_{yz} \tag{33}$$

$$dA_{yz} = (\mathbf{r}_u \times \mathbf{r}_v)^T (-\hat{\mathbf{x}}) \tag{34}$$

2.3.5 Step 5: Evaluate Surface Integral

The aerodynamic coefficients are computed by evaluating the surface integral in Eq. (4) and Eq. (5). The first integration is performed with respect to v since the shadow boundary was solved as $v = v(u)$. After the integration is performed with respect to v, the shadow boundaries are substituted as the limits of integration for v. As a result, the remaining integrand is only a function of u, and the second integration is performed with respect to u. The limits of integration substituted for u are constants, u_1 and u_2. During evaluation of the analytic expressions, the designer specifies the portion of the shape used by supplying the values for u_1 and u_2.

2.3.6 Step 6: Output Aerodynamics Code

After the analytic relations have been developed, they are output to a Matlab-based aerodynamics module that can be downloaded in Section 2.1. This module contains the analytic relations, reference areas, and reference lengths. Hence, as the analytic relations of various shapes are developed, the aerodynamics module becomes a library containing analytic relations for a wide variety of shapes.

2.4 Derivation and Validation of Analytic Expressions for Basic Shapes

Common hypersonic vehicles can be constructed via the superposition of basic shapes as illustrated in Figure 5 of Section 1.2.4. The construction and parametrization of the various basic shapes will be shown. It is important to note that all basic shapes are parametrized by variables describing the family of the basic shape, such as the half-angle of a sharp cone, and by variables describing the surface, u and v, as previously described. This generalization of each basic shape allows the development of only one set of analytic relations for each basic shape family. All shapes are chosen to have symmetry along the body axes where applicable and to be centered at the origin. A range of angles of attack and sideslip was chosen to validate both shadowed and unshadowed orientations.

2.4.1 Sharp Cone Family

Although sharp cones are not used alone as hypersonic vehicles due to the significant heating that would occur on the sharp nose, conical frustums are commonly used as portions of hypersonic vehicles such as sphere-cones and biconics. The sharp cone family is parametrized by the cone half-angle, δ_c, and length along the axis of revolution, L. The surface of the sharp cone is parametrized using the local radius from the axis of revolution, $u = r$, and revolution angle, $v = \omega$, as shown in Figure 9. The resulting parametrized position vector, $\mathbf{r}$, is shown in Eq. (35).

$$\mathbf{r} = [L - \frac{u}{\tan(\delta)} \quad u\cos(v) \quad -u\sin(v)]^T \tag{35}$$

Comparisons between the analytic relations and CBAERO in both force and moment coefficients for a 20^o sideslip and various cone half angles are shown in Figure 10 and Figure 11, respectively. Initially, comparisons were also planned between the

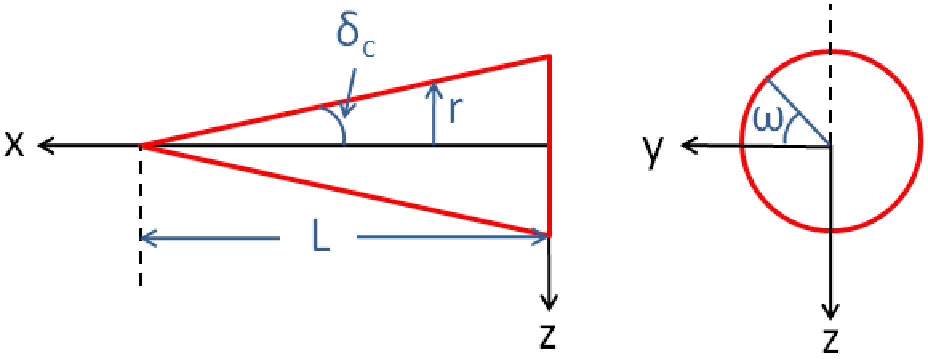


Figure 9: Side and front view of sharp cone parametrization.

analytic static stability derivatives and CBAERO. However, the CBAERO mesh resolution required to obtain accurate stability derivatives was prohibitive due to the presence of numerical noise. However, as shown in Figure 11, excellent agreement was observed between the analytic moment equations and CBAERO. Therefore, a comparison was made between the analytic stability derivatives and the finite difference of analytic moment coefficients (shown in Figure 12). Excellent agreement is observed in all cases, and the roll moment, C_l, is identically zero as expected for axisymmetric bodies. Note that shadowing occurs for $\alpha > \delta_c$, and these effects are included in the results.

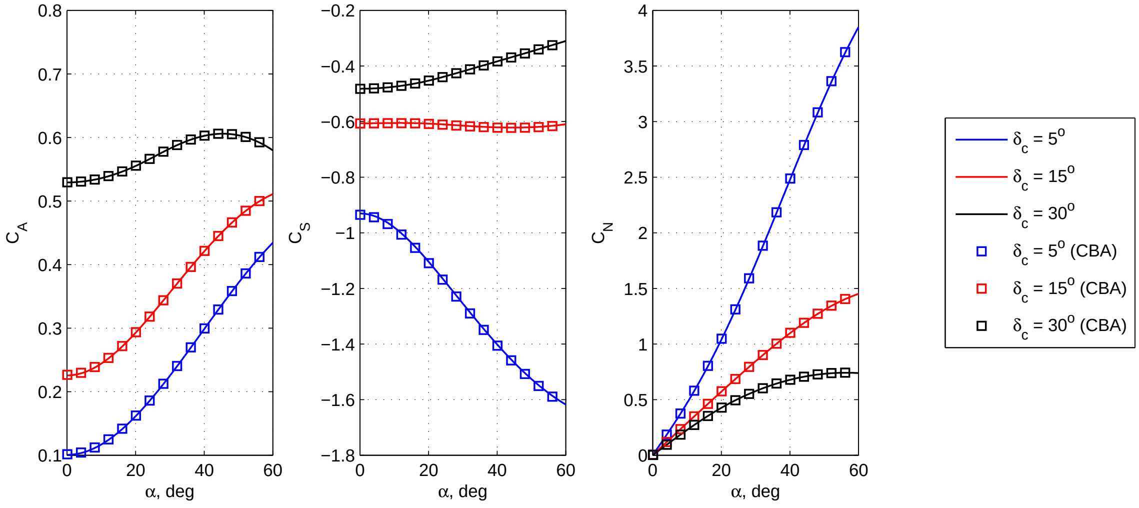


Figure 10: Sharp cone force coefficient validation, $\beta = 20^o$.

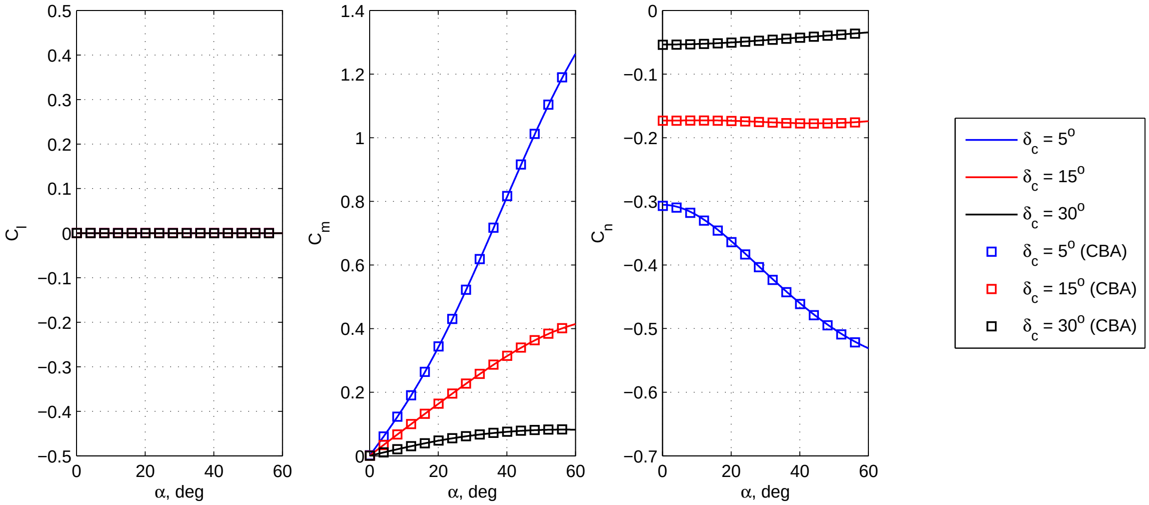


Figure 11: Sharp cone moment coefficient validation, $\beta = 20^o$.

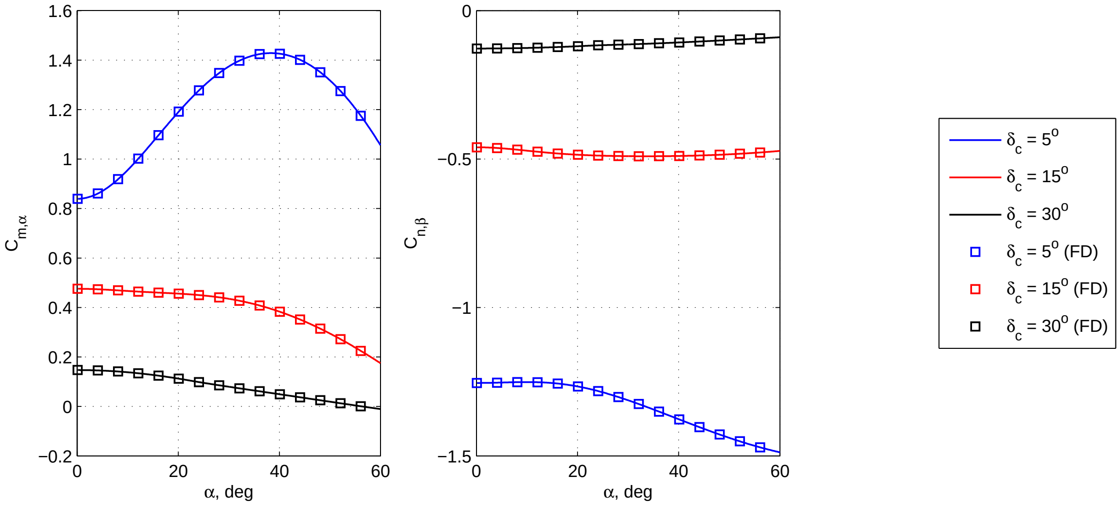


Figure 12: Sharp cone stability derivative validation, $\beta = 20^o$.

2.4.2 Spherical Segment Family

The blunting of hypersonic vehicles to reduce aeroheating is often achieved through the addition of a spherical segment as the nose of the vehicle. For common hypersonic vehicles such as sphere-cones and blunted biconics, the spherical segment family is parametrized by the nose radius, r_n, and cone half-angle, δ_c, as shown in Figure 13.

The nose radius and cone half-angle determine the portion of the spherical segment used to blunt the vehicle due to tangency conditions enforced between the spherical segment and conical frustum. The surface of the spherical segment is parametrized by the distance from the origin along the x-axis, $u = x$, and revolution angle, $v = \omega$, as shown in Figure 13. The resulting position vector, $\mathbf{r}$, is shown in Eq. (36). A more natural parameterization based on a spherical coordinate system could also be chosen. However, this system results in expressions that cannot currently be integrated analytically using Mathematica. This illustrates the challenge of properly selecting a surface parametrization that leads to analytically integrable expressions.

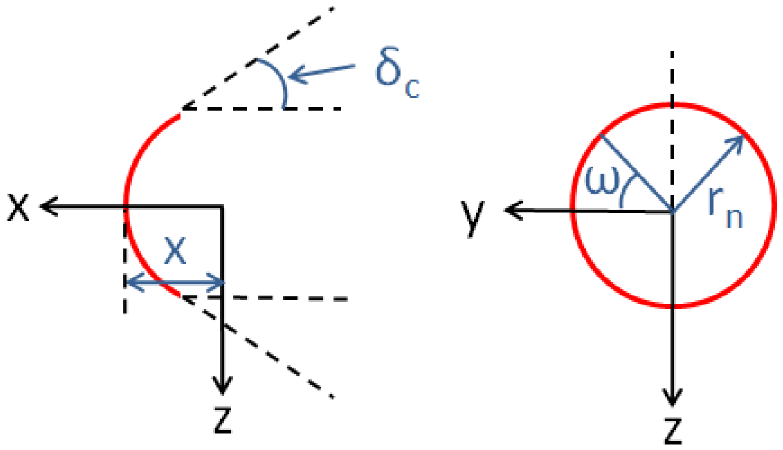


Figure 13: Side and front view of spherical segment parametrization.

$$\mathbf{r} = [u \quad \sqrt{r_n^2 - u^2}\cos(v) \quad -\sqrt{r_n^2 - u^2}\sin(v)]^T \tag{36}$$

In order to reduce the integration complexity associated with a spherical segment, analytic relations were derived using a total angle of attack, ϵ, that is a function of both angle of attack and sideslip as shown in Eq. (37). The resulting normal and axial force coefficients in the total angle of attack frame, C'_N and C'_A, are converted back to the body frame using Eq. (38), where ϕ' is the angle between the body frame and total angle of attack frame.[33] Comparisons between the analytic force coefficients and CBAERO for a 20^o sideslip and various half angles are shown in Figure 14. As shown, excellent agreement exists when using the total angle of attack formulation. Note that the distribution of Newtonian pressure forces always points to the center of the spherical segment. Thus, a spherical segment centered at the origin will exhibit no moments. This is confirmed by the solution of zero for all moments from the integration process. Consequently, the force coefficients are the sole basis of comparison with CBAERO (Figure 14).

$$\epsilon = \arccos(\cos(\alpha)\cos(\beta)) \tag{37}$$

$$\begin{bmatrix} C_N \\ C_S \\ C_A \end{bmatrix} = \begin{bmatrix} C'_N \cos(\phi') \\ -C'_N \sin(\phi') \\ C'_A \end{bmatrix} \tag{38}$$

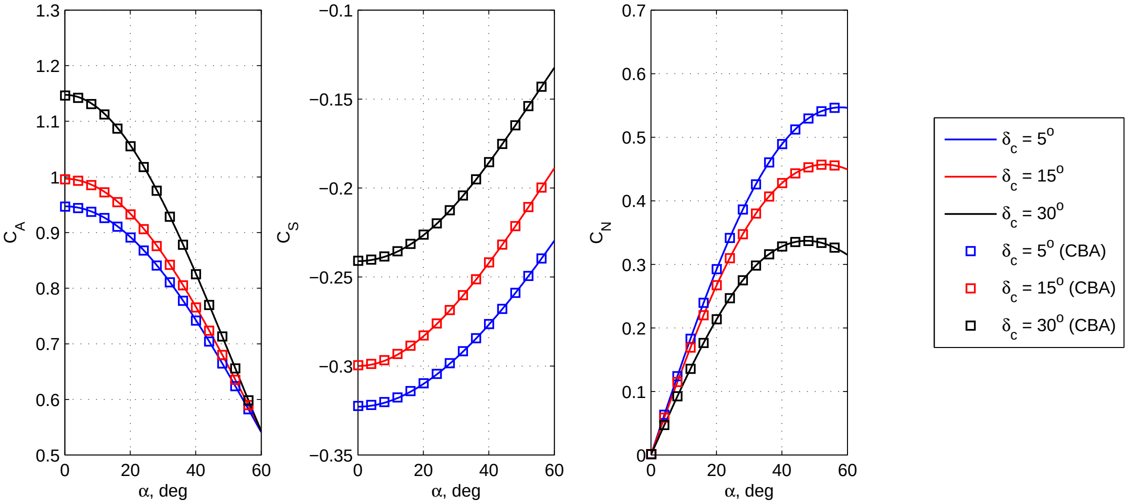


Figure 14: Spherical segment force coefficient validation, $\beta = 20^o$.

2.4.3 Flat Plate Family

Various portions of hypersonic vehicles, such as stationary fins and moving control surfaces, can be approximated as flat plates. Additionally, the flat bodies of advanced hypersonic vehicle concepts such as the blended wedge[32] can be modeled using flat plates. The flat plate family is parametrized by the half-angle of the blended wedge, δ, as shown in Figure 15.

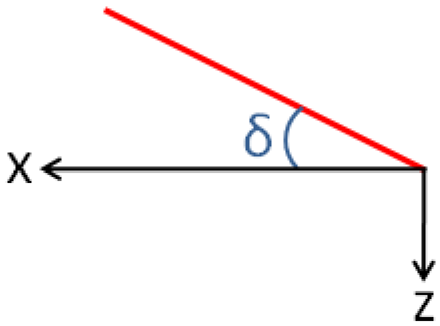


Figure 15: Side view of flat plate parametrization.

According to Newtonian flow theory, the C_p distribution of a flat plate is constant. Thus, the force coefficients are obtained using Eq. (1), resulting in Eqs. (39)-(41). Additionally, the moment coefficients about the centroid of the flat plate are identically zero. This is the simplest example of the form the analytic aerodynamics will appear. In general, a panel method approximates a shape as a collection of flat plates and does not provide a useful validation.

$$C_A = 2[\sin(\delta)\cos(\alpha)\cos(\beta) + \cos(\delta)\sin(\alpha)\cos(\beta)]^2 \sin(\delta) \tag{39}$$

$$C_S = 0 \tag{40}$$

$$C_N = 2[\sin(\delta)\cos(\alpha)\cos(\beta) + \cos(\delta)\sin(\alpha)\cos(\beta)]^2 \cos(\delta) \tag{41}$$

2.4.4 Cylindrical Segment Family

Although the cylindrical segment is not a common shape observed in hypersonic vehicles, it is used to blunt advanced vehicle concepts such as the blended wedge.[32] The cylindrical segment family is parametrized by the nose radius, r_n, and blended wedge half-angle, δ. The nose radius and half-angle determine the portion of the cylindrical segment used to blunt the vehicle due to tangency conditions enforced between the spherical segment and flat plates of a blended wedge design. The surface of the cylindrical segment is parametrized by the distance from the origin along the y-axis, $u = y$, and revolution angle about the y-axis, $v = \theta$, as shown in Figure 16. The resulting position vector, **r**, is shown in Eq. (42).

$$\mathbf{r} = [r_n \cos(v) \quad u \quad -r_n \sin(v)]^T \tag{42}$$

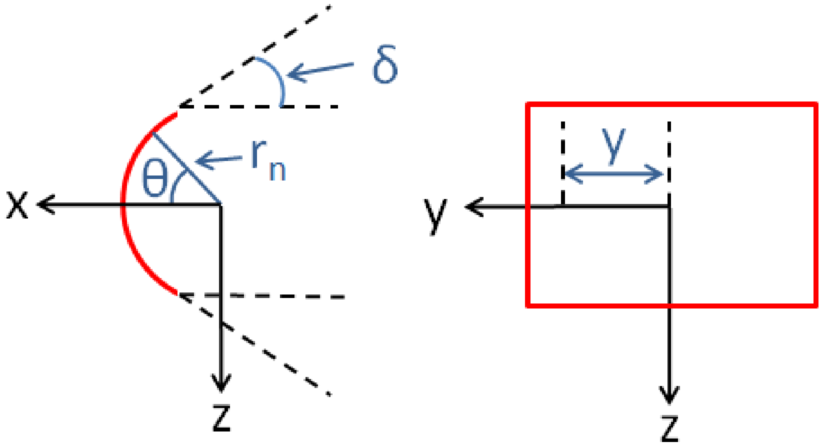


Figure 16: Side and front view of cylindrical segment parametrization.

In order to validate the cylindrical segment, a panel method based on the equations of a flat plate in Section 2.4.3 is developed. This allowed for easy verification of the aerodynamic coefficients since the construction of a vehicle in CBAERO is time consuming when the axis of revolution is not parallel to the flow at zero angle of attack and zero sideslip. Comparisons were made between the analytic force coefficients and the developed panel method for a 20^o sideslip and various half angles. As shown in Figure 17, excellent agreement exists between the cylindrical segment and the panel method. Note that the sideforce and moment coefficients of a cylinder centered at the origin are identically zero.

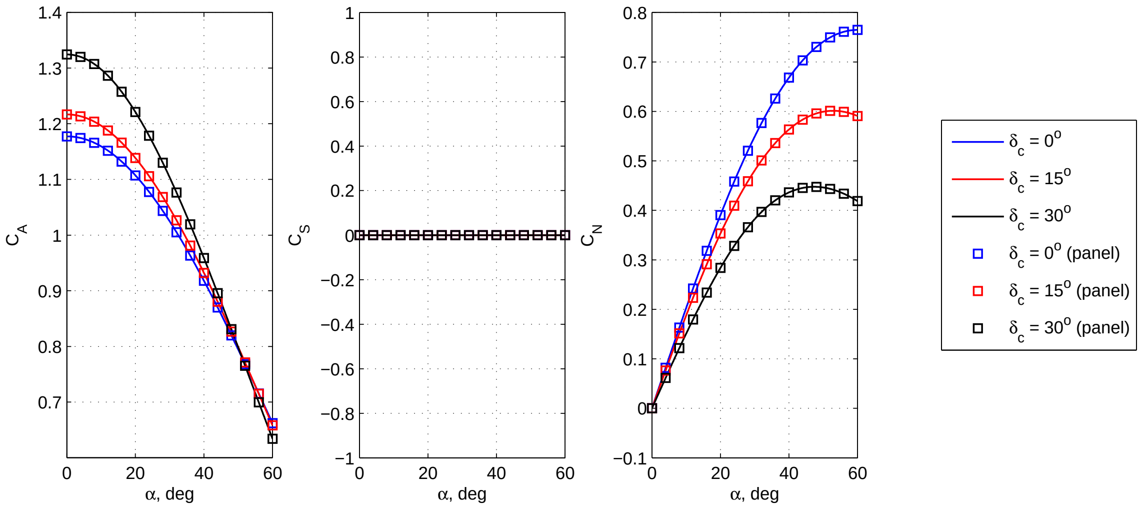


Figure 17: Cylindrical segment force coefficient validation, $\beta = 20^o$.

2.5 Vehicle Construction Through Superposition

2.5.1 Methodology

The aerodynamics of common hypersonic vehicles can be determined through superposition of basic shapes. Sphere-cones can be constructed using a spherical segment and a single conical frustum, and biconics can be constructed using a spherical segment and two conical frustums. The geometry of a sphere-cone is specified by the nose radius, r_n, cone half angle, δ_1, and base diameter, d, as shown in Figure 18. The geometry of a blunted biconic is specified by the nose radius, r_n, forward cone half angle, δ_1, aft cone half angle, δ_2, base diameter, d, and height, h, as shown in Figure 19. Each basic shape used will likely have different reference areas and lengths. Therefore, the superpositioning of basic shapes cannot be performed by simply adding the aerodynamic coefficients from each shape. Rather, the aerodynamic coefficients of each basic shape must be scaled to a common reference area and length. For example, the axial force coefficient of a sphere-cone, $C_{A,sc}$, would be calculated using Eq. (43), where the overall vehicle reference area, A_{sc}, is chosen to be the base area of the sphere-cone.

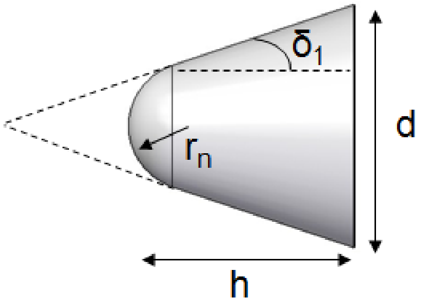


Figure 18: Sphere-cone parametrization.

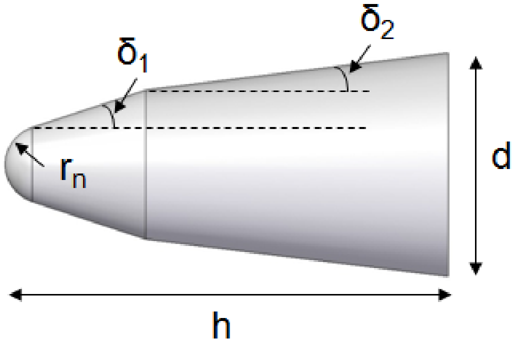


Figure 19: Biconic parametrization.

$$C_{A,sc} = C_{A,cone}\frac{A_{ref,cone}}{A_{sc}} + C_{A,sphere}\frac{A_{ref,sphere}}{A_{sc}} \tag{43}$$

2.5.2 Sphere-Cone and Blunted Slender Biconic Validation

The force and moment coefficients as well as static stability derivatives for both a blunt sphere-cone and blunted slender biconic at a 20^o sideslip are validated with CBAERO using parameters listed in Table 3. As shown in Figures 20-22, excellent agreement in force, moment, and stability derivative coefficients is observed for both vehicles. Note that the analytic relations properly account for both the shadowed and unshadowed orientations included in these results.

Table 3: Sphere-cone and blunted biconic parameters.

Parameter	*Sphere-Cone*	*Biconic*
r_n	0.3 ft.	1.0 in.
δ_1	70.0^o	17^o
δ_2	-	8^o
d	2.5 ft.	19.6 in.
h	-	48.0 in.

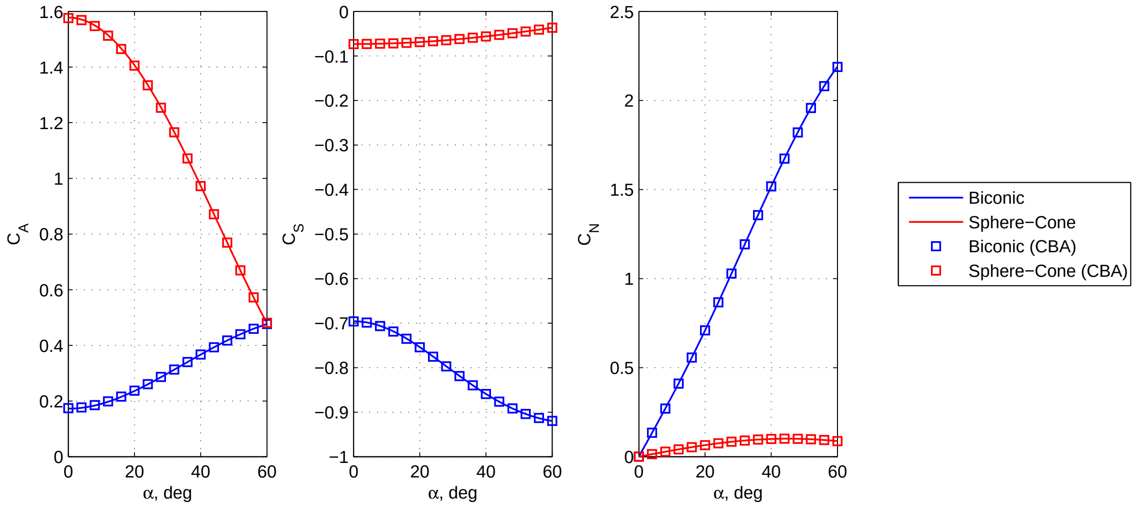


Figure 20: Sphere-cone and blunted biconic force coefficient validation, $\beta = 20^o$.

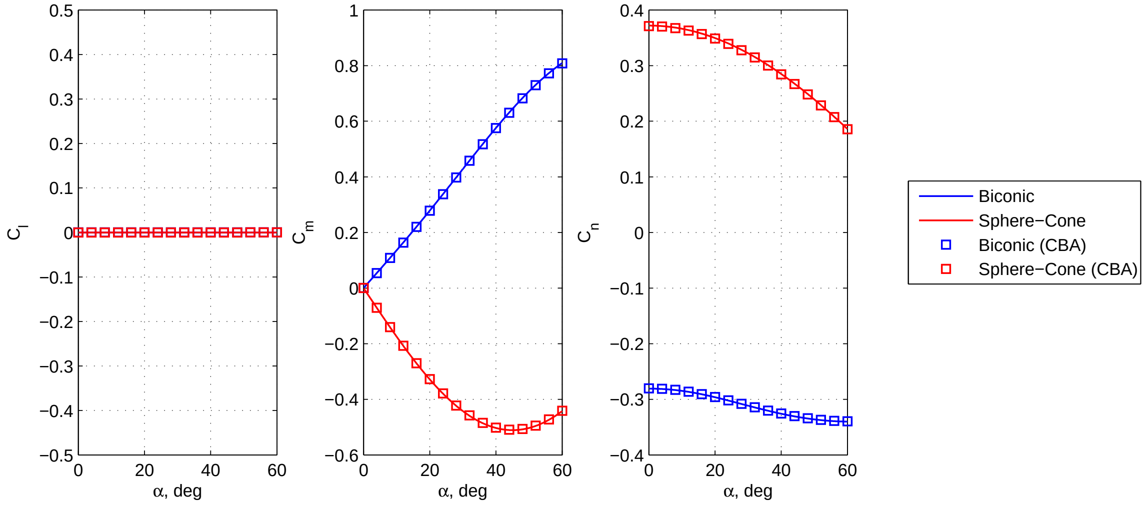


Figure 21: Sphere-cone and blunted biconic moment coefficient validation, $\beta = 20^o$.

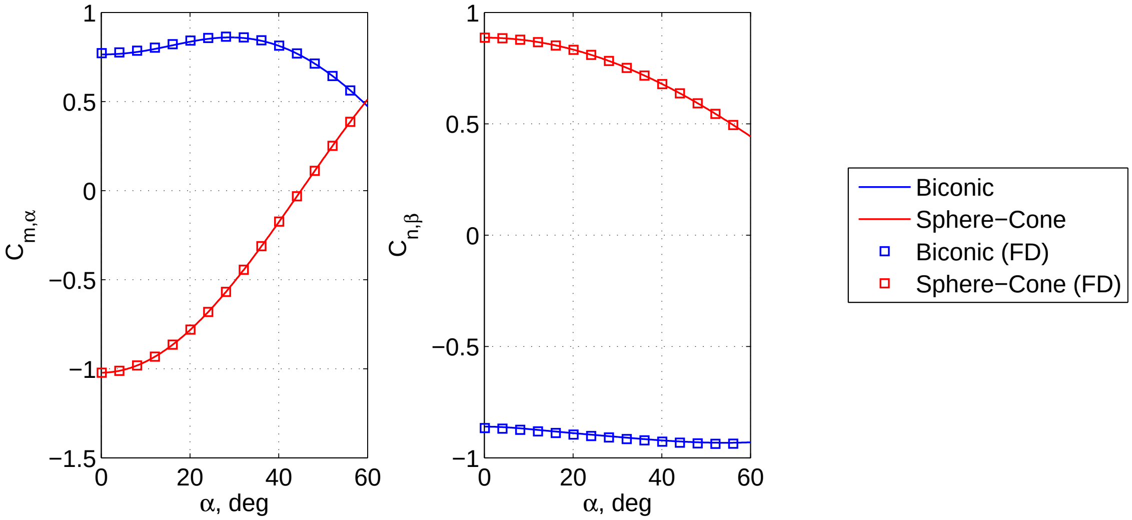


Figure 22: Sphere-cone and blunted biconic stability derivative validation, $\beta = 20^o$.

2.5.3 Example Application Using the Superposition of Basic Shapes

As shown, basic shapes can be superpositioned to construct full hypersonic vehicles commonly used for various missions. The analytic relations allow for rapid parametric sweeps and shape optimization. As an example, vehicle designers may be interested in a biconic entry vehicle with a peak L/D of 2 subject to certain geometric constraints.

For this problem, a maximum vehicle height of 48 in. and a maximum base diameter of 21 in. was assumed. For a given diameter, an increase in the height would result in a more slender vehicle with higher peak L/D. Therefore, in order to maximize the base diameter, the height of the vehicle must also be maximized. Parametric sweeps over a wide range of angles of attack and zero sideslip were performed for various cone half angles, δ_1 and δ_2, in 1^o increments as well as various base diameters. Contours in peak L/D for the maximum allowable base diameter of 21 in. are shown in Figure 23. As shown, no sharp biconic meeting the given geometric constraints is capable of achieving an L/D of 2. A theoretical best L/D of only 1.86 could be achieved with $\delta_1 = 18^o$ and $\delta_2 = 11^o$. Consequently, the base diameter was reduced until a sharp biconic with an L/D of 2 was identified. As shown in Figure 24, an L/D of 2 is achieved with $d = 19.6$, $\delta_1 = 17^o$, and $\delta_2 = 10^o$. Note that each point in the contour plots corresponds to a full sweep of angles of attack, a process that greatly benefits from the orders of magnitude reduction in computational requirements provided by the analytic relations. While this comparison is based on user experience, a more detailed computational comparison will be provided in Section 2.11 after all of the current analytic relations are discussed.

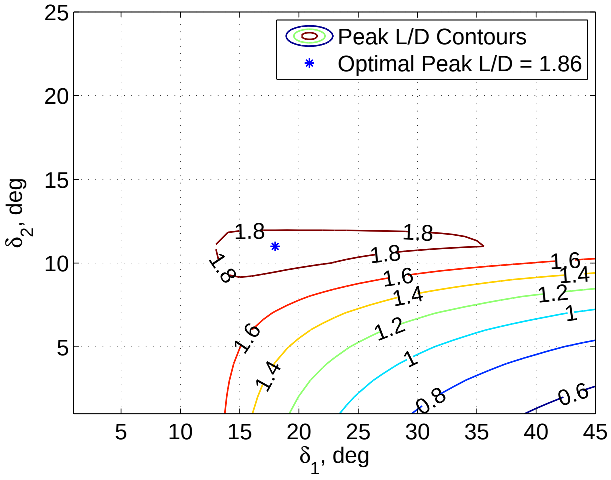


Figure 23: Contour of peak L/D for d = 21 in.

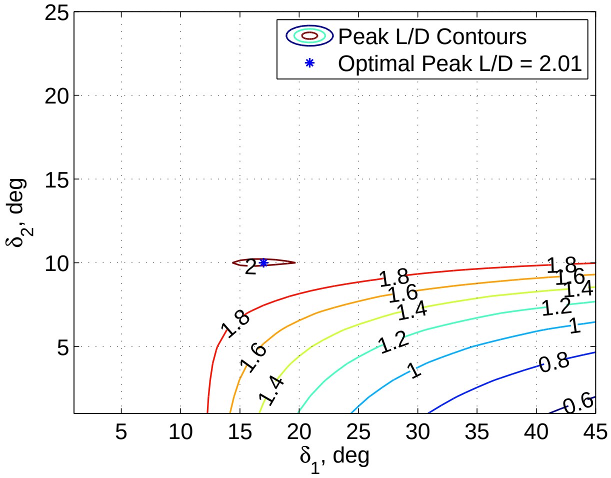


Figure 24: Contour of peak L/D for d = 19.6 in.

2.6 General Shapes

The superposition of basic shapes provides the capability to determine the analytic aerodynamics of common hypersonic vehicles. However, shapes with improved aerodynamic performance outside of the range of basic shapes may be required to accomplish future missions. Therefore, it is desirable to be able to obtain the analytic aerodynamics of more general shapes. Many methods from computer-aided design exist to describe general shapes. As an initial step in this direction, analytic relations of low-order Bezier curves of revolution have been developed for unshadowed total angles of attack in which the full body is exposed to the flow.

2.6.1 Bezier Curves of Revolution

The development of analytic relations for Bezier curves of revolution allow for rapid analysis of general bodies of revolution. The geometry of a Bezier curve is parametrized by a nondimensional arclength, s_b, as shown in Eq. (44)-(46) where $0 \leq s_b \leq 1$.[107] The location of the i^{th} control node is specified in the vector $\mathbf{B_i}$ and the order of the Bezier curve is specified by n. The control nodes specify a control polygon inside which the Bezier curve must reside. Example second-order Bezier curves, along with their corresponding control node locations and control polygons, are shown in Figure 25. As expected, each Bezier curve resides inside the control polygon and is connected to the initial and final control nodes.

$$\mathbf{P}(s_b) = \sum_{i=0}^{n} \mathbf{B_i} \; J_{n,i}(s_b) \tag{44}$$

$$J_{n,i}(s_b) = \binom{n}{i} s_b^i (1 - s_b)^{n-i} \tag{45}$$

$$\binom{n}{i} = \frac{n!}{i!(n-i)!} \tag{46}$$

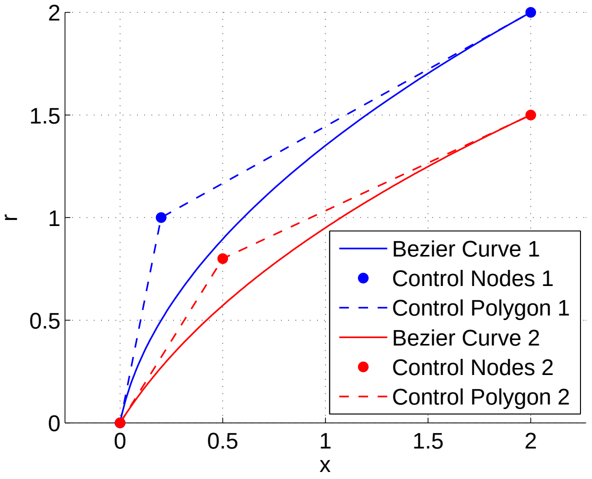


Figure 25: Bezier curves of revolution with control nodes and control polygons.

Analytic relations have been developed for a second-order Bezier curve of revolution in which the control node locations are generalized. This allows the development of one set of analytic relations to fully describe all second-order Bezier curves of revolution. These relations assume that the entire vehicle is exposed to the flow. This ensures the limits of integration are separated by eliminating the shadow boundary that couples the surface parameters. Shadowed orientations were also considered, but the coupling provided by the shadow boundary resulted in expressions that could not be analytically integrated using Mathematica. The resulting force and moment coefficients of the Bezier curves of revolution shown in Figure 25 have been validated with CBAERO as shown in Figures 26 and 27, respectively.

Even at unshadowed orientations, the resulting analytic relations are quite lengthy. In this situation, Mathematica is unable to analytically integrate the large intermediate expressions that appear during the integration process. As a result, these lengthy expressions are segmented into individual terms that Mathematica is able to manipulate. This process results in hundreds of terms with an estimated total character length of 500,000 for each analytic expression. To visualize the massive length of these expressions, the example analytic relation for a sharp cone shown in Appendix

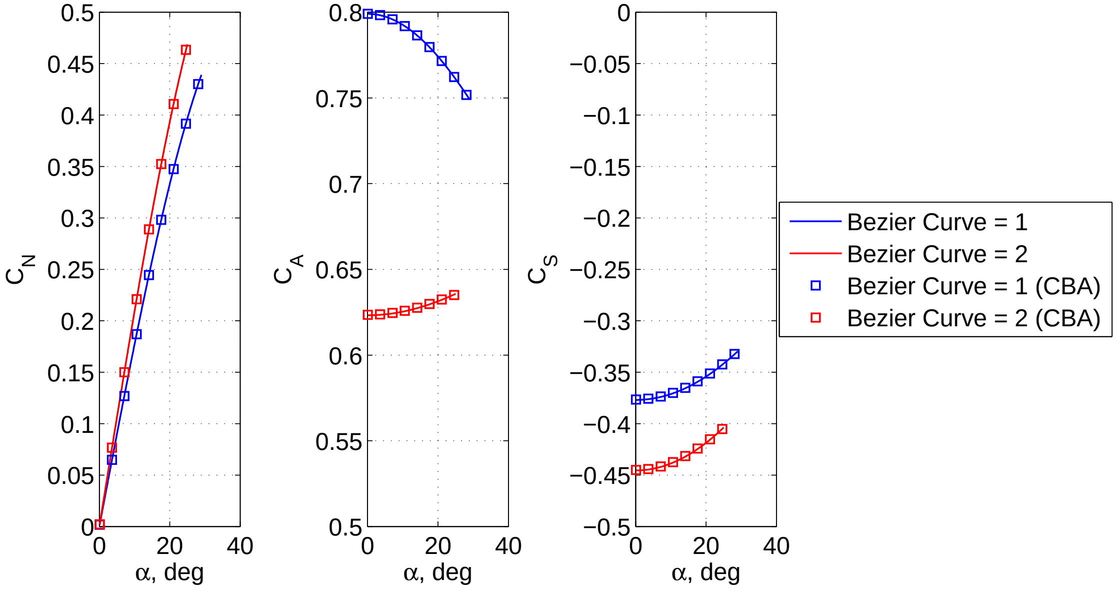


Figure 26: Unshadowed Bezier curves of revolution force coefficient validation, $\beta = 20^o$.

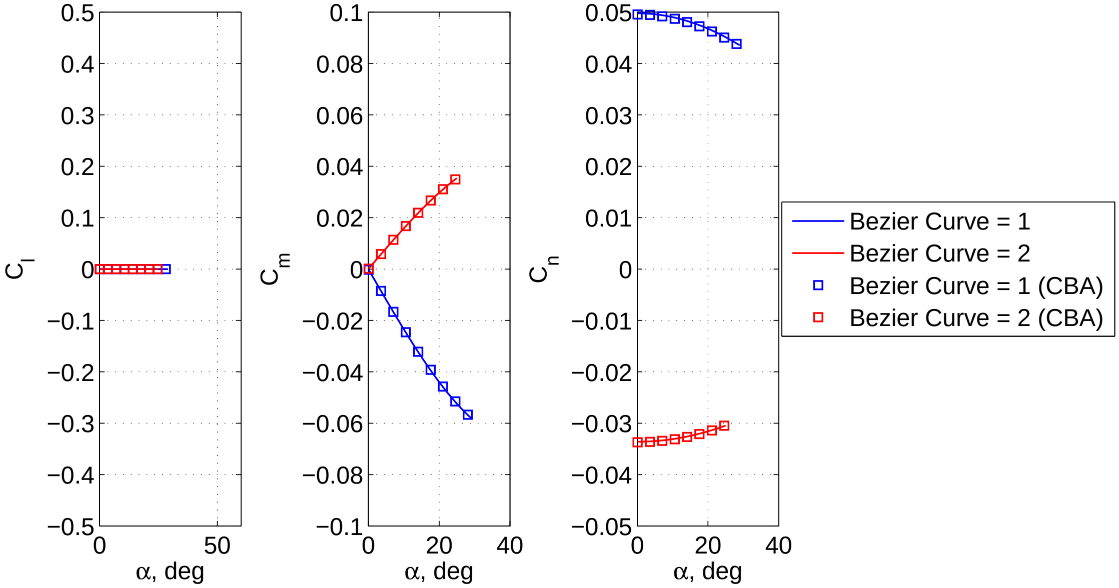


Figure 27: Unshadowed Bezier curves of revolution moment coefficient validation, $\beta = 20^o$.

A.3 contains only about 12,100 characters. With this consideration, the normal force coefficient of a Bezier curve of revolution would require approximately 248 pages if it was included in the appendix. For certain control node locations, a singularity in the analytic relations has also been observed at a single location along the Bezier curve. While removal of this point results in analytic values that agree with CBAERO, the root cause of this singularity is difficult to determine across the hundreds of analytic terms. The complexity provided by the lengthy expressions and the inability to solve for shadowed orientations illustrates the current mathematical limit in the analytic relations. While solutions can generally be obtained for unshadowed orientations, shadowed solutions for other simple geometries such as parabolas of revolution and stretched spherical segments also cannot be obtained using current methods. However, by ignoring the physical model of Newtonian flow theory and analyzing the resulting mathematical model, a new analytical approach is developed that reduces the length of certain existing expressions, enables analytical solutions for new configurations, and provides a foundation to construct compact, hybrid exact-approximate analytic solutions by addressing the fundamental challenges of analytic Newtonian aerodynamics.

2.7 Fundamental Challenges of Analytic Newtonian Aerodynamics

One of the major challenges during the integration process is that many functions cannot be analytically integrated. For example, the simple function $\sin(x)/x$ has no known analytic integral. As vehicles of greater complexity are analyzed, there is no guarantee that a closed form, analytic solution exists. In the prior section, simple bodies of revolution are analyzed as an attempt to expand the analytic database. The surface of these bodies are naturally parametrized using trigonometric functions. As a result, the corresponding shadow boundaries are functions of inverse trigonometric functions. These functions, when combined with other polynomial expressions that appear during the integration process, generally result in expressions that cannot be analytically integrated. For example, the term $(2x\text{ArcSec}[2/\sqrt{4+\cot a^2/x}])/(1-4x)^2$ that appears during the integration of the parabola of revolution cannot be analytically integrated. The remaining terms that can be analytically integrated have very long integrals that are similar in nature to the lengthy expressions obtained from the unshadowed Bezier curves of revolution.

To avoid the complexity of inverse trigonometric functions, bodies of revolution could be parametrized using Cartesian coordinates. This parametrization eliminates the inverse trigonometric functions that appear in the shadow boundary from the revolution angle. However, when this parametrization is used, inverse trigonometric functions reappear from the mapping of Cartesian distances (used to provide integration limits that describe the surface of the shape) to inverse trigonometric functions during integration. For example, $\int \mathrm{Sqrt}[1 - x\hat{\ }2]\,\mathrm{d}x = \frac{1}{2}\left(x\sqrt{1-x^2} + \mathrm{ArcSin}[x]\right)$. Similar expressions also occur due to the normal vector normalizations required to calculate the pressure coefficient and direction of force as shown in Eq. (2) in Section 1.2.2. For example $\int 1/\mathrm{Sqrt}[1 - x\hat{\ }2]\,\mathrm{d}x = \mathrm{ArcSin}[x]$. Many vehicle shapes suffer from this inverse trigonometric complexity due to both of these complications.

It is important to note that the sphere does not suffer from the complexity that results from vector normalizations since the magnitude of the normal vector is constant over the entire surface. Instead, this shape only suffers from the complexity that arises from the integration limits. If the expressions that result from these integration limits can be appropriately modified, then it is possible to avoid this complexity entirely for spherical segments. While solutions to spherical segments were obtained in Section 2.4.2, this new approach enables compact aerodynamic expressions that are otherwise not possible from the traditional Newtonian calculation and serves as a means to expand the analytic aerodynamic database. The necessary modification can be achieved by departing from the traditional surface integration model of Newtonian flow theory to a volume integration using the Divergence Theorem.

2.8 Calculation of Newtonian Aerodynamics Using The Divergence Theorem

In traditional Newtonian flow theory, the aerodynamic coefficients are calculated by integrating the pressure coefficient over the unshadowed surface of the vehicle. In this model, the pressure coefficient is derived from the momentum transfer of fluid particles as they are deflected by the surface of the vehicle as shown in Figure 2 of Section 1.2.2. While this model is representative of the physics of the flow, it is mathematically equivalent to a flux that penetrates the surface of the vehicle with an assumed permeable outer mold line. As an example, the calculation of the normal force coefficient can be converted in this manner as shown in Eq. (47) where the flux does not originate from the physical fluid flow. Rather, a mathematical flow field is constructed, and the flux is calculated from this mathematical vector field. For the normal force coefficient calculation, the mathematical vector field is $\mathbf{F} = C_p\ \hat{\mathbf{z}}$ as shown in Eq. (47). Note that this mathematical vector field acts along $\hat{\mathbf{z}}$ only and is independent of the direction of freestream flow shown in Figure 28. However, the magnitude of this mathematical vector field is equivalent to the pressure coefficient and, consequently, is a function of both vehicle shape and the relative orientation of the vehicle to the freestream flow. Although this calculation has never been previously performed for Newtonian flow, the mapping between interior and boundary calculations provided by the Divergence Theorem is similar to other aerodynamic applications that can be expressed using potential flow theory.[20]

$$\begin{aligned} C_N &= \frac{1}{A_{ref}} \iint\limits_S C_p\ \hat{\mathbf{n}}_{\mathbf{in}}^T (-\hat{\mathbf{z}})\ \mathrm{d}A \\ &= \frac{1}{A_{ref}} \iint\limits_S (C_p\ (-\hat{\mathbf{z}}))^T \hat{\mathbf{n}}_{\mathbf{in}}\ \mathrm{d}A \\ &= \frac{1}{A_{ref}} \iint\limits_S (C_p\ \hat{\mathbf{z}})^T \hat{\mathbf{n}}_{\mathbf{out}}\ \mathrm{d}A \end{aligned} \tag{47}$$

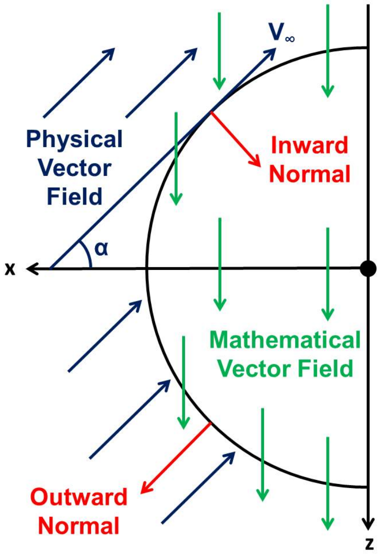


Figure 28: Comparison between physical and mathematical vector fields.

While this mathematical conversion does not change any calculations that would be made on the surface of the vehicle, it enables an alternative calculation to be performed using the Divergence Theorem. The Divergence Theorem is used to map the flux of a vector field, $\mathbf{F} = M\mathbf{i} + N\mathbf{j} + P\mathbf{k}$, across a closed oriented surface S in the direction of the surface's outward unit normal field, $\hat{\mathbf{n}}_{\mathbf{out}}$, to the divergence of the vector field $\mathbf{F}$ over the region D enclosed by the surface as shown in Eq. (48). Note that while the divergence form requires an additional integration to be performed, this is offset to a degree by the extra derivative of the integrand. The divergence form also eliminates the unit normal from the integrand that generally increases the complexity of the integration process. As an example, the fundamental theorem of calculus shown in Eq. (49) is a one-dimensional implementation of the Divergence Theorem. If $\mathbf{F} = f(x)\mathbf{i}$, then $\frac{\mathrm{d}f}{\mathrm{d}x} = \nabla^T\mathbf{F}$. By defining the outward unit normal to be $\mathbf{i}$ at b and $-\mathbf{i}$ at a as shown in Figure 29, then the total outward flux of $\mathbf{F}$ across the boundary of $[a, b]$ is equivalent to $f(b) - f(a)$ as shown in Eq. (50).[6] To compute the analytic force coefficients of various vehicles using the Divergence Theorem, this one-dimensional example is extended to three dimensions.

$$\iint\limits_S \mathbf{F}^T\mathbf{n}\,\mathrm{d}\sigma = \iiint\limits_D \nabla^T\mathbf{F}\,\mathrm{d}V \tag{48}$$

$$f(b) - f(a) = \int_a^b \frac{\mathrm{d}f}{\mathrm{d}x} \tag{49}$$

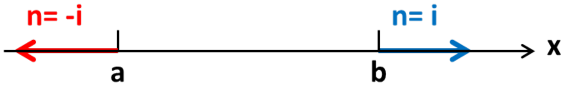


Figure 29: Outward unit normals at the boundary of $[a, b]$ in one-dimensional space.[6]

$$\begin{aligned} f(b) - f(a) &= f(b)\mathbf{i}^T\mathbf{i} + f(a)\mathbf{i}^T(-\mathbf{i}) \\ &= \mathbf{F}(b)^T\mathbf{n} + \mathbf{F}(a)^T\mathbf{n} \end{aligned} \tag{50}$$

The transformation provided by the Divergence Theorem is of little use when numerically approximating the hypersonic aerodynamic coefficients. The addition of a third integral would likely increase the computational requirements of numerical methods. However, the Divergence Theorem enables the analytic flux calculation (that is mathematically equivalent to the analytic surface integration described by Section 2.3) to be mapped to an analytic divergence calculation. Since the surface integration is only performed over the unshadowed surface of the vehicle, the analytic flux calculation must also only be performed over the unshadowed surface of the vehicle. As a result, the analytic divergence calculation must be performed throughout the unshadowed volume of the vehicle as shown in Figure 30 for an example hemisphere. Note that the shadow boundary occurs where $C_p = 0$, forming a plane for the hemisphere example.

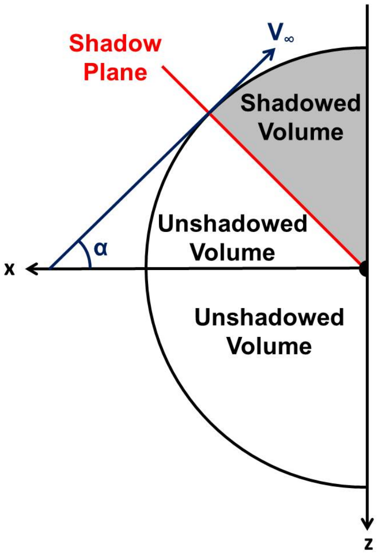


Figure 30: Shadowed and unshadowed volumes of hemisphere.

As previously mentioned, only the flux across the unshadowed surface of the vehicle is used to calculate the aerodynamic coefficients. However, the integration of the divergence of the mathematical vector field throughout the unshadowed volume is equivalent to the flux across all enclosing boundaries. As such, the flux across any shadowed enclosing boundaries must be removed from the divergence solution. Since the magnitude of the mathematical vector field is equivalent to C_p as described in Eq. (47), the velocity of the mathematical vector field vanishes along the shadowed boundary where $C_p = 0$. As a result, the corresponding flux is identically zero along this boundary. Additionally, the shadowed boundary along the lower portion of the example hemisphere is parallel to the direction of the mathematical vector field as shown in Figure 31. As a result, the flux along this boundary is also identically zero. In summary, the flux corresponding to the normal force coefficient as described by Eq. (47) across the unshadowed surface of the example hemisphere is equivalent to the integral of the divergence throughout the unshadowed volume.

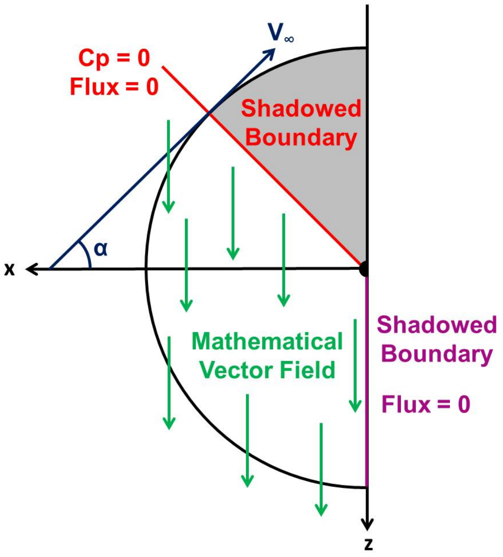


Figure 31: Flux of mathematical vector field for normal force calculation of hemisphere.

When computing additional aerodynamic coefficients, such as the axial force coefficient, the direction of the mathematical vector field changes as shown in Figure 32. To compute the axial force coefficient, the divergence of the new mathematical vector field is calculated throughout the unshadowed volume. While the flux is zero across the shadowed boundary where $C_p = 0$, the flux across the lower shadowed boundary is no longer zero. To accurately compute the axial force coefficient, the flux across this boundary must be removed from the divergence solution. While this method is mathematically correct, the analytic expression for the flux across the lower shadowed boundary is generally quite lengthy. This flux calculation resembles the surface integrations performed by traditional Newtonian methods and, as a result, reduces the compactness of the analytic aerodynamics made possible by the Divergence Theorem.

This issue can be eliminated by extending the shadow plane to the maximum z-coordinate of the example hemisphere. An add-on volume can then be created by extending the lower portion of the hemisphere along the negative x-direction until the extended shadow plane is reached as shown in Figure 33. The surface created by this extension is parallel to the mathematical vector field, and as a result, the flux across this surface is zero. Since the flux along the shadowed plane where $Cp = 0$ is also zero, the integral of the divergence of the mathematical vector field throughout this add-on volume is equivalent to the flux across the adjacent shadowed boundary shown in Figure 32. As such, the compactness of the analytic expressions made possible by the Divergence Theorem is fully realized using this approach. Note that the addition of these volumes would alter the aerodynamics when performing traditional Newtonian calculations since these volumes would alter the unshadowed surface of the vehicle. However, the direction of the mathematical vector field remains constant regardless of vehicle orientation and physical flow direction, allowing the construction of these additional volumes. While an example hemisphere is used, the general approach described in this section is applied to all divergence calculations where applicable.

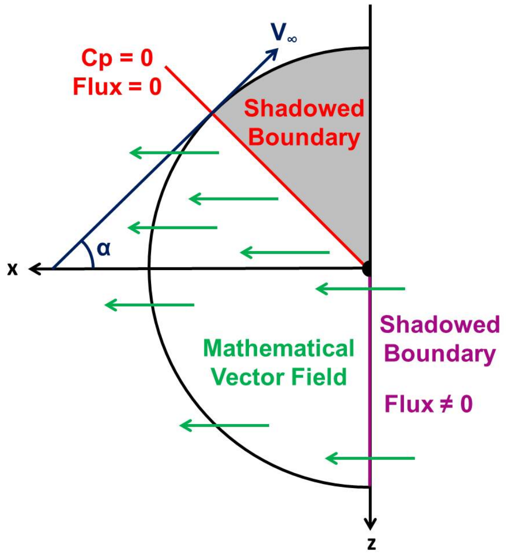


Figure 32: Flux of mathematical vector field for axial force calculation of hemisphere.

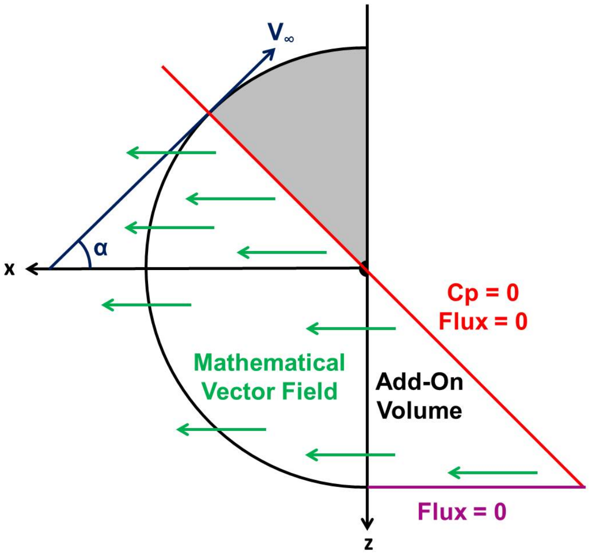


Figure 33: Flux of mathematical vector field with add-on volume for axial force calculation of hemisphere.

2.9 Application of the Divergence Theorem to Various Shapes

The spherical shape that motivates the use of the divergence theorem serves as a starting point for the exploration of shapes that benefit from this approach. In this investigation, Cartesian potential functions are used to describe each shape that is analyzed, and Mathematica is used to perform the necessary integrations.

2.9.1 Quadratic and Semi-Quadratic Potential Functions

The spherical potential shown in Eq. (51) has several simplifying properties that make it a useful starting point to implement the divergence theorem. As previously mentioned, the magnitude of the normal vector is constant across the entire surface of the shape. This potential recovery of the gradient shown in Eq. (52) eliminates the inverse trigonometric functions that result from normal vector normalizations as described in Section 2.7. As a result, the divergence of the flux corresponding to the normal force coefficient can be expressed as shown in Eq. (53) for an assumed zero sideslip. This assumption is applied to all divergence calculations in this investigation and also eliminates y from the numerator for reasons discussed later. When using the traditional Newtonian calculation, terms of the form $x^n\sqrt{1-x^2}$ appear, where n is even. These terms integrate to inverse trigonometric functions that add complexity to the integration process, resulting in lengthy analytic aerodynamic expressions. Alternatively, the extra derivative provided by the divergence theorem results in terms with a similar form where n is odd. These expressions eliminate the introduction of inverse trigonometric functions during the integration process, resulting in compact aerodynamic expressions. While the specific expressions for both the traditional and divergence calculations can be obtained from the aerodynamic database in Section 2.1, the divergence calculation reduces the number of characters in the normal force coefficient by a factor of 24 and in the axial force coefficient by a factor of 15 when the add-on volume described in Section 2.8 is used. If the flux across the shadowed boundary is used instead of the add-on volume, then the length of the axial force coefficient expression is only reduced by a factor of 10.

$$\phi = x^2 + y^2 + z^2 = \rho^2 \tag{51}$$

$$|\nabla\phi| = \sqrt{4(x^2+y^2+z^2)} = 2\rho = 2\sqrt{\phi} \tag{52}$$

$$\nabla^T\mathbf{F} = \frac{-8\sin\alpha(-2x\cos\alpha - 2z\sin\alpha)}{4\phi} \tag{53}$$

Prior to integration, the unshadowed volume is divided into various regions. If these regions are not chosen carefully, then the limits of integration that define these regions may result in expressions that cannot be analytically integrated. After evaluating various options, a set of regions has been identified that enables the analytic integrations to be performed. This set has proven to be successful across all shapes included in this investigation. As an example, the quadratic potential of a hemisphere is divided into four regions. The first region is located in the forward section that is fully exposed to the flow as shown in Figure 34, and the fourth region consists of the entire lower section that is also fully exposed to the flow. Regions 2 and 3 fill the remaining unshadowed volume and are defined as shown in Figure 35 for an example cross-section of this remaining volume. Region 2 is bounded from above by the shadow plane whereas region 3 is bounded from above by the spherical shape. Note that these regions vary in size for each x location.

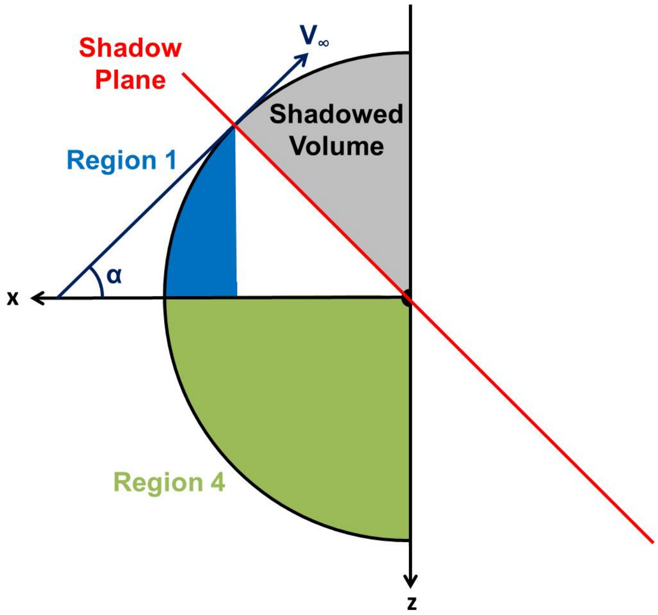


Figure 34: Regions 1 and 4 for divergence integrations.

These regions were also chosen to provide integration limits that naturally result from the choice in integration order. Many integration orders result in expressions that cannot be analytically integrated. The integration order of z, x, then y is chosen for this work. This order avoids the introduction of inverse trigonometric functions during the z and x integrations that generally prevent analytic solutions. Since y is the last integration to be performed, there is no concern if inverse trigonometric functions are introduced in this step. This integration order is also successfully used across all shapes in this investigation. The divergence solutions for the spherical segments described in Section 2.4.2 are validated using CBAERO as shown in Figure

36. Note that the analytic solution for the side force coefficient is identically zero as expected for a zero sideslip whereas numerical noise is present in the approximate solutions from CBAERO.

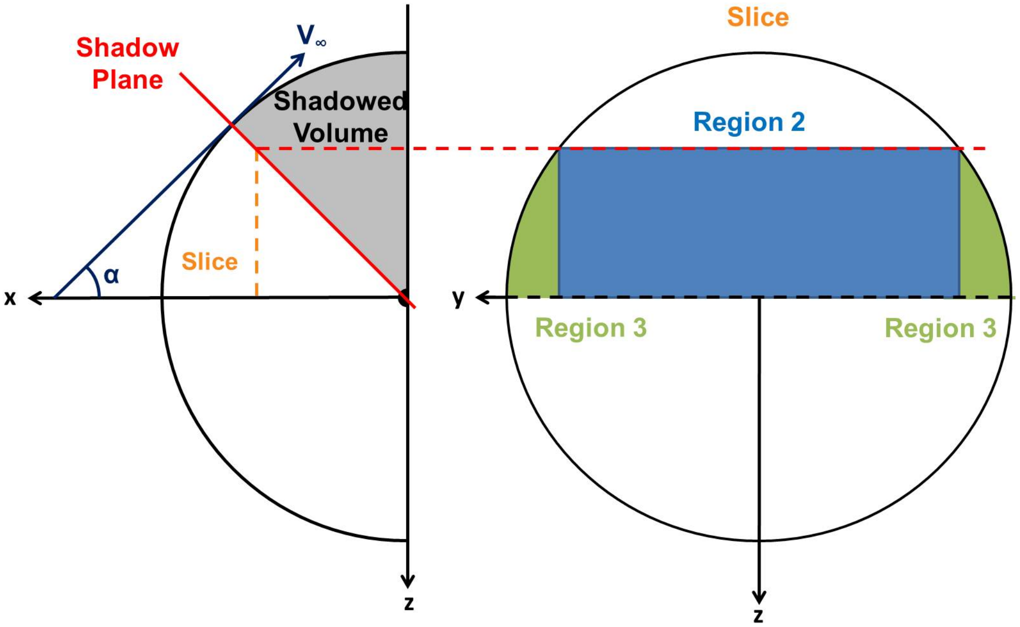


Figure 35: Regions 2 and 3 for divergence integrations.

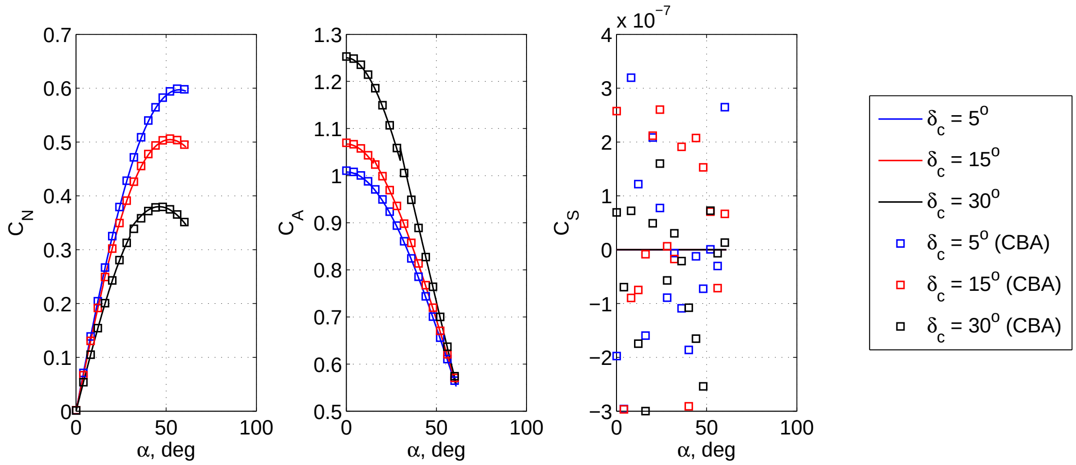


Figure 36: Comparison of spherical segment relations to CBAERO for various δ_c.

The simplification of analytic solutions provided by the divergence theorem also enables the construction of analytic solutions for configurations that could not be solved analytically using the traditional Newtonian calculations outlined in Section

2.3. To obtain analytic solutions for more general shapes, the complexity of the spherical potential is increased as shown in Eq. (54), where c_{xz} is a constant. This form enables a partial recovery of the potential function in the gradient magnitude as shown in Eq. (55), where the magnitude is now a function of y and is denoted as $g(y)$. As a result, the expressions for the mathematical fluxes also increase in complexity as shown in Eq. (56) for the flux corresponding to the normal force calculation. The form of $g(y)$ was chosen to ensure that both x and z are absent from the gradient magnitude expression. As a result, the added complexity of the gradient is delayed until the final integration is performed with respect to y.

$$\phi = c_{xz}x^2 + f(y) + c_{xz}z^2 = \rho^2 \tag{54}$$

$$|\nabla\phi| = \sqrt{4c_{xz}(\rho^2 - f(y)) + \left(\frac{\partial f(y)}{\partial y}\right)^2} = g(y) \tag{55}$$

$$\nabla^T\mathbf{F} = \frac{-8\sin\alpha(-2x\cos\alpha - 2z\sin\alpha)}{g(y)^2} \tag{56}$$

Initially, $f(y)$ is chosen as shown in Eq. (57) to maintain a quadratic potential, and the analytic relations can be obtained from the aerodynamic database. The results for the shapes shown in Figure 37 where validated using CBAERO as shown in Figure 38. For non-axisymmetric bodies such as these, the GNU Triangulated Surface Library is used to construct the necessary meshes for evaluation in CBAERO.[26] Note that a singularity exists in these analytic relations for the spherical potential where $c_y = c_{xz}$. In this case, the prior analytic relations should be used instead.

$$f(y) = c_y y^2 \tag{57}$$

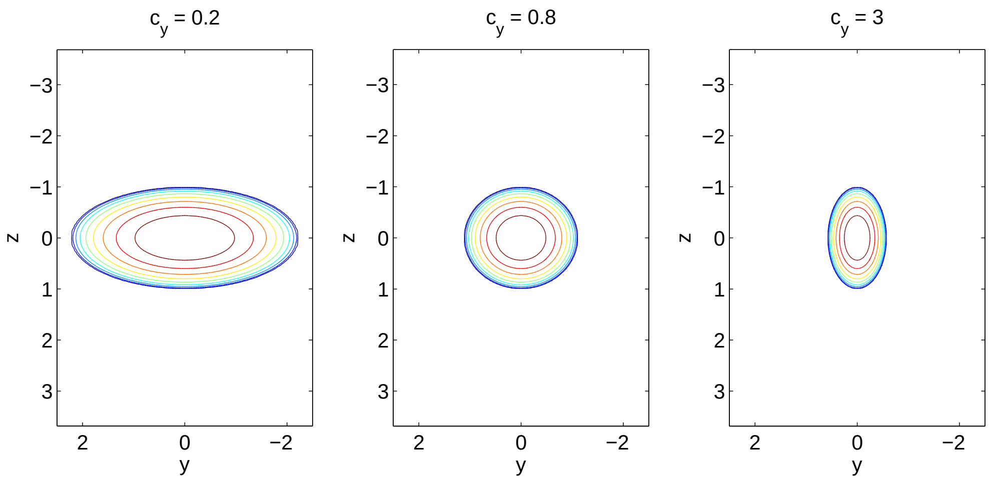


Figure 37: Front view contours of shape for $f(y) = c_y y^2$ at various c_y values.

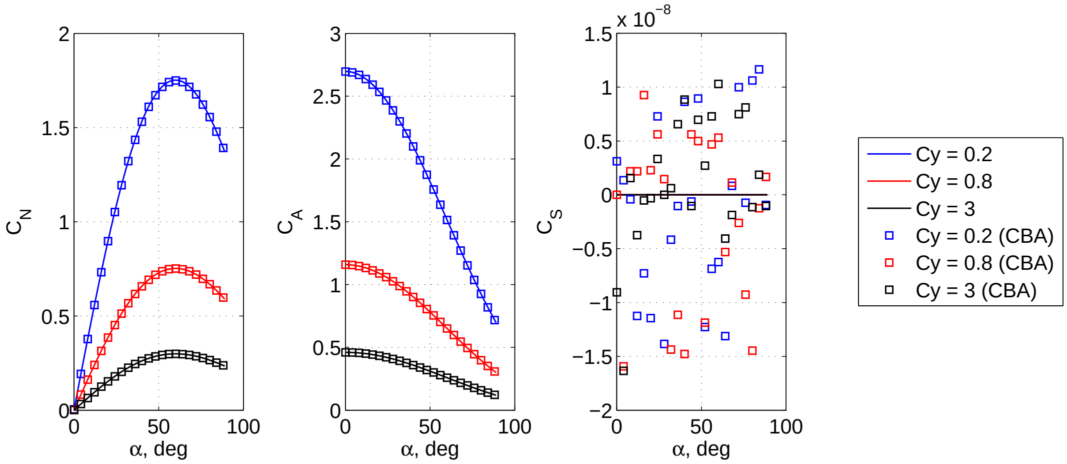


Figure 38: Comparison of quadratic relations to CBAERO for various values of c_y.

Other candidate functions for $f(y)$ were also evaluated including various powers of y, trigonometric functions of y, etc. However, an analytic solution could not be found for these functions. This mathematical limitation is a fundamental result of the normalization of vectors. As already shown in Eq. (55), $f(y)$ appears in two forms ($f(y)$ and $\frac{\partial f(y)}{\partial y}$) due to the normalization. When $f(y)$ is a quadratic function, both of these terms can be combined as a function of y^2. When any other power is used or when other functions are used, this simplification no longer applies, and the increased complexity of the normalization term prohibits an analytic solution. As a result, a mathematical limit in the analytic relations appears to have been reached. However, the function $f(y) = e^{Ay+B} - e^B$ is an exception to this observation. The derivative property of this function results in a unique vector normalization in which terms in $f(y)$ can be combined using a complete-the-square technique. While the resulting analytic expressions in the database contain many individual imaginary terms, the combined result is purely real (to machine precision). Various configurations (shown in Figure 39 where $B = 0$) are validated using CBAERO as shown in Figure 40.

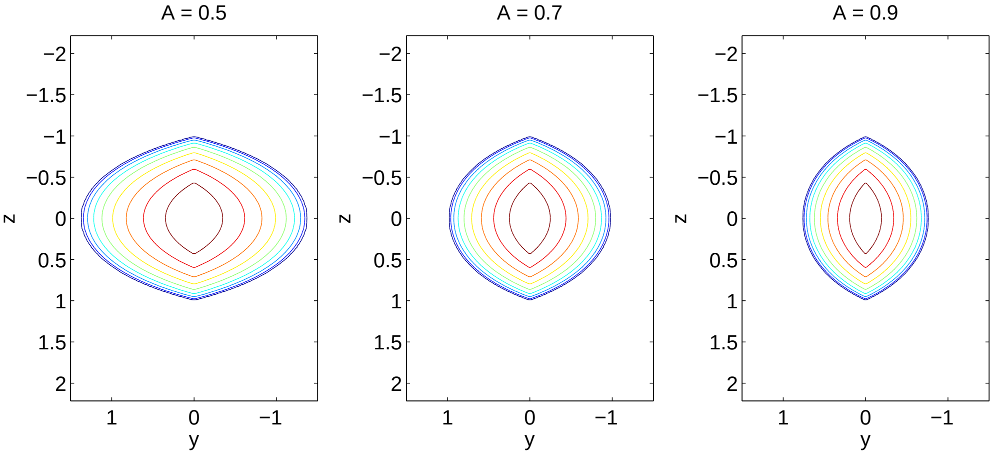


Figure 39: Front view contours of shape for $f(y) = e^{Ay+B} - e^B$ at various A values.

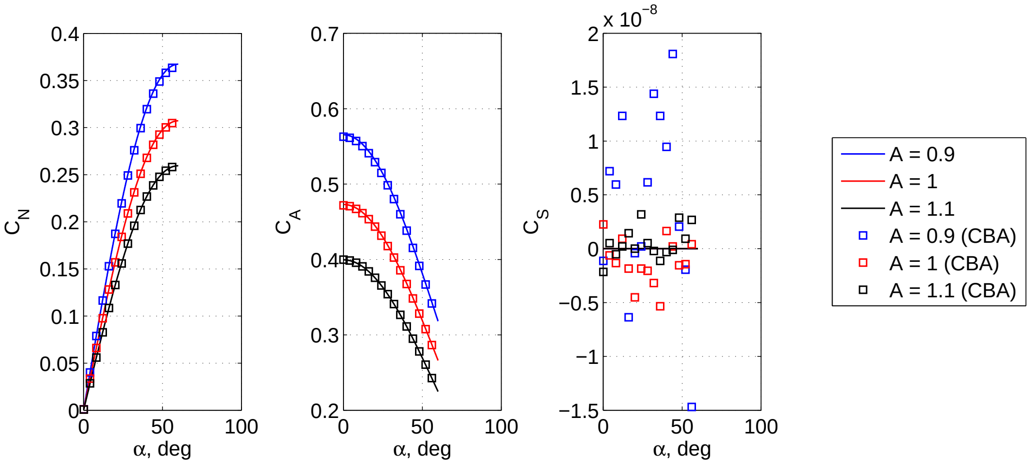


Figure 40: Comparison of quadratic exponential relations to CBAERO for various values of A.

These results, in addition to the analytic solutions derived from traditional Newtonian calculations, represent the current mathematical limit of exact solutions within the aerodynamic database. However, shapes outside of this exact database have solutions that can largely be represented with exact, analytic solutions and only require a numerical approximation for a small portion of the solution. As such, the divergence theorem serves as the cornerstone to develop hybrid exact-approximate solutions.

2.10 Hybrid Exact-Approximate Solutions

As each shape is divided into various regions as described in Section 2.9.1, an exact solution is found only if all integrations of each region can be performed analytically. For many shapes outside of the current aerodynamic database, the majority of these regions can be integrated analytically and require no numerical approximations. For the remaining regions, the first set of integrations can generally be performed analytically while the final integration must be approximated numerically. If the expressions that must be numerically integrated remain small compared to the exact, analytic solution, then these hybrid exact-approximate solutions would likely serve as a computationally efficient approximation to the corresponding hypersonic aerodynamic coefficients. As an example, the aerodynamic solution to a parabola of revolution is nearly exact and requires only a small set of expressions to be approximated numerically.

2.10.1 Parabola of Revolution

The parabola of revolution is described by the potential function shown in Eq. (58) with a gradient magnitude shown in Eq. (59). As shown, the potential recovery from the normal vector normalization of this shape is a function of x alone. As a result, the integration order for this shape is altered to z, y, then x. For this shape, only one term must be approximated for both axial and normal force coefficients as shown in Eqs. (60) and (61), respectively, where the exact solutions can also be found in the aerodynamic database. Both of the terms that must be approximated originate from Region 3 in which all other terms from this region can be analytically integrated. Additionally, these approximated terms result from analytic integrations in both z and y such that only a single integral with respect to x must be approximated. The limits of integration in x are determined by the boundaries of Region 3, where l is the length of the vehicle. Note that for this shape, the shadow boundary forms a plane that is parallel to the x-y plane. As a result, the add-on volume technique described in Section 2.8 cannot be used to simplify the flux through the base of the vehicle.

$$\phi = x + y^2 + z^2 = 0 \tag{58}$$

$$|\nabla\phi| = \sqrt{1 - 4x} \tag{59}$$

$$C_A = C_{A,exact} + \frac{2\left(\int_{-l}^{-1/4*\cot a^2} - \frac{2x\mathrm{ArcSec}\left[\frac{2}{\sqrt{4+\frac{\mathrm{Cot}[a]^2}{x}}}\right](1-x+(1+x)\mathrm{Cos}[2a])}{(1-4x)^2}\right)}{A_{ref}} \tag{60}$$

$$C_N = C_{N,exact} + \frac{2\left(\int_{-l}^{-1/4*\cot a^2} \frac{2x\mathrm{ArcSec}\left[\frac{2}{\sqrt{4+\frac{\mathrm{Cot}[a]^2}{x}}}\right]\mathrm{Sin}[2a]}{-1+4x}\right)}{A_{ref}} \tag{61}$$

Each integral is approximated as a finite, analytic sum. As a result, these exact-approximate solutions also provide an analytic mapping of vehicle shape to vehicle performance in the same manner as the prior exact solutions. As such, shapes that can be approximated in this manner can also be included into the unified mathematical design framework developed in this dissertation. Unlike panel methods, these expressions directly approximate the final aerodynamic result, and the error of this result can be directly controlled to obtain a desired accuracy. Alternatively, panel methods approximate vehicle shape to a certain level of accuracy, and the designer usually does not know *a priori* how this approximation translates to the accuracy of the aerodynamic coefficients. Consequently, multiple meshes of varying resolutions must be evaluated until convergence of the aerodynamic coefficients is observed. Using a trapezoidal approximation, the hybrid exact-approximate results are validated for various lengths with geometries shown in Figure 41 using CBAERO as shown in Figure 42.

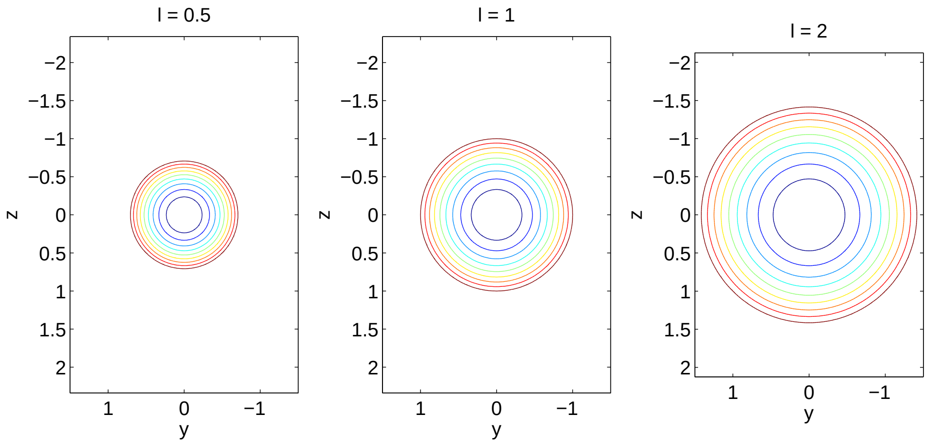


Figure 41: Front view contours of shape for parabola of revolution at various l values.

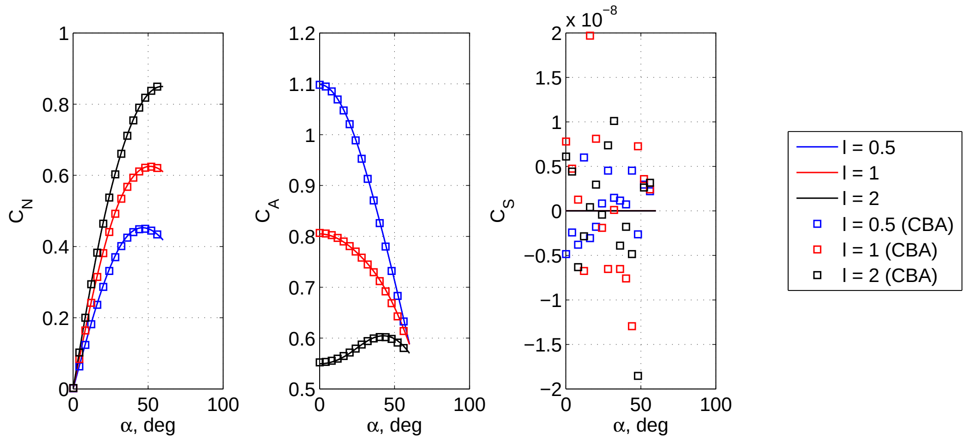


Figure 42: Comparison of parabola of revolution to CBAERO for various values of l.

There are likely many shapes with hybrid exact-approximate solutions. As more complex potential functions are used, the potential recovery technique of the gradient will likely no longer result in an expression that is a function of one variable, such as y as shown in Eq. (55). For example, the general spherical potential shown in Eq. (62) has a gradient magnitude shown in Eq. (63), where the potential recovery technique is only capable of eliminating a single variable (chosen to be x). However, integrations within certain regions can make use of simplifying properties to offset this added complexity. For example, Region 2 shown in Figure 35 is bounded above by the shadow plane where $C_p = 0$ and bounded below where $z = 0$. Since the divergence of the flux for the normal force coefficient is $\frac{dC_p}{dz}$, a first integration with respect to z would recover the pressure coefficient. As such, this expression vanishes along the upper boundary of Region 2 where $C_p = 0$. Additionally, all instances of z vanish along the lower boundary where $z = 0$. These calculations eliminate z from the potential recovery expression in Eq. (63). As a result, this new expression takes the form of a single variable, y, allowing the remaining integrations to be carried forward in the same manner as before. Techniques such as these may be beneficial as the aerodynamic database is expanded to include hybrid exact-approximate solutions of more complex shapes.

$$\phi = c_x x^2 + c_y y^2 + c_z z^2 = \rho^2 \tag{62}$$

$$|\phi| = \sqrt{4(c_x^2 x^2 + c_y^2 y^2 + c_z^2 z^2)} = \sqrt{4c_x(\rho^2 - c_y y^2 - c_z z^2) + 4c_y^2 y^2 + 4c_z^2 z^2} \tag{63}$$

2.11 Computational Advantages of the Analytic Aerodynamics

While all of the previous shapes enable the construction of a unified mathematical framework to perform rapid simultaneous hypersonic aerodynamic and trajectory optimization, their solutions could also serve as a computationally efficient substitute for panel methods within traditional design environments. A computational comparison between the analytic relations and CBAERO is shown in Figure 43. For these data, the maximum discrepancy in a relevant design parameter, L/D, across varying angles of attack is shown for various CBAERO mesh resolutions. In this comparison, only L/D values greater than 0.01 are compared to avoid the L/D error singularity as α approaches zero. Note that the discrepancy shown in Figure 43 does not directly correspond to the numerical approximation error of CBAERO. CBAERO is performing many other calculations and is not directly approximating the analytic solutions obtained in this dissertation. However, as the mesh resolution is made more fine, the CBAERO solutions increase in accuracy, and as a result, the discrepancy between the CBAERO solutions and the analytic solutions decrease. This accuracy benefit is attained at the increased computational cost of more panel calculations. Alternatively, the analytic relations provide an exact solution to Newtonian flow theory, and the computational cost of evaluating each relation remains fixed.

For this comparison, a speedup factor is also calculated and is defined as the ratio of clock time between the CBAERO calculations and the evaluation of the analytic relations over a range of angles of attack for a single geometry. Since CBAERO is written in a compiled language, the Matlab-based analytic relations were autocoded into C to make a reasonable comparison of clock times. Note that the speedup factor also includes the clock time necessary to generate the mesh of panels for CBAERO since this process would be required for any change in vehicle shape. Alternatively, the speedup factor does not include the time required to develop the analytic relations since this has already been performed for the shapes evaluated in Figure 43 and does not need to be repeated. While the CBAERO solutions for each shape were individually scaled in this chapter to account for the Modified Newtonian calculations that differ from the traditional Newtonian calculations performed analytically, this scaling was not performed to allow a consistent comparison across shapes.

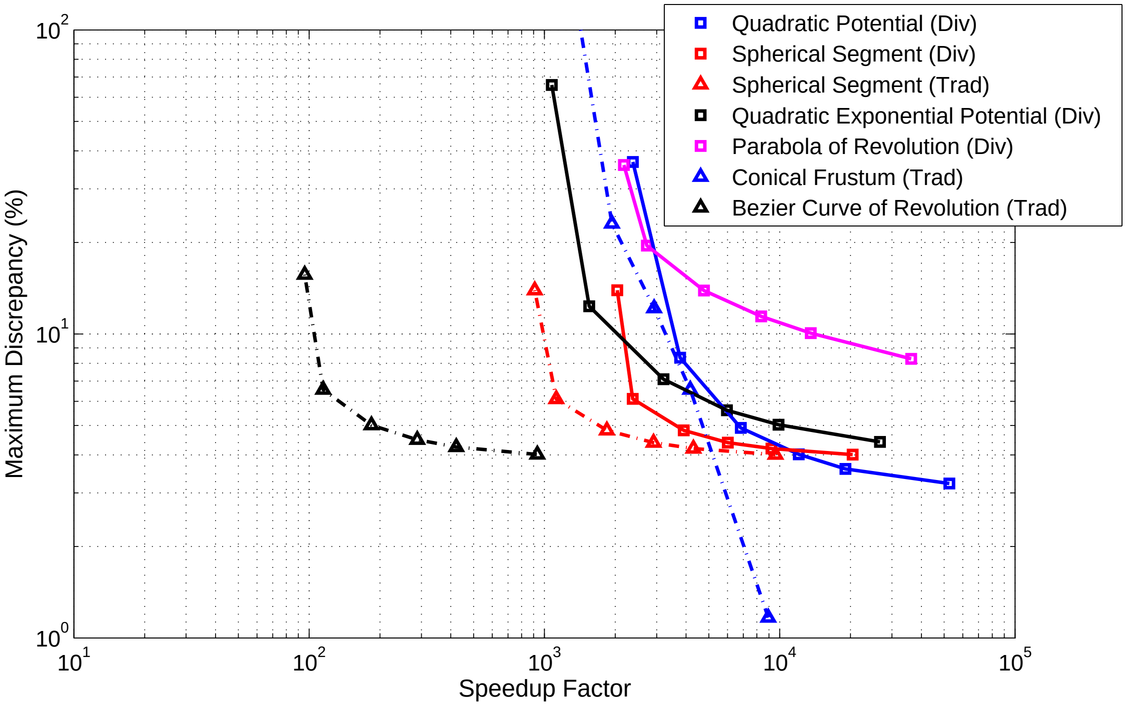


Figure 43: Maximum percent error in L/D vs. speedup of analytic aerodynamics.

As shown in Figure 43, convergence in numerical accuracy is only observed for those shapes that show a maximum discrepancy that is largely insensitive to speedup factor (spherical segments and Bezier curve of revolution). While the other shapes do not display full convergence of this numerical accuracy, the large speedups provided by the analytic relations would eliminate the usefulness of CBAERO calculations with higher mesh resolutions. As such, these higher resolutions were not included in the computational comparison. Note that even for large, unconverged discrepancies, many of the exact, analytic relations still provide a speedup of approximately three orders of magnitude. This includes the example hybrid exact-approximate analytic solution (parabola of revolution), illustrating that further opportunities exist in this area. Bezier curves of revolution are the only exception to this observation due to the very lengthy analytic solutions obtained for these shapes. However, the analytic relations for these shapes still provide a speedup of two orders of magnitude. Additionally, greater speedup factors are observed for the divergence solution to the spherical segment as expected from the compact relations provided by this approach. Note that for small speedup factors, large changes in discrepancy is observed for small changes in speedup factor. This is due to CBAERO calculations that are dominated by meshing algorithms that must be executed to construct the surface panels. For low mesh resolutions, this upfront cost is relatively constant and serves as a minimum time in which CBAERO can be executed.

The conical frustum illustrated in Figure 9 of Section 2.4.1 is in general more slender than the other shapes considered. As a result, the Modified Newtonian scaling is unnecessary, and the numerical accuracy in L/D of these slender shapes greatly improves for higher mesh resolutions. Note that while the discrepancy in L/D is shown in Figure 43, many other quantities such as moment coefficients are also calculated by CBAERO. As such, the same calculations were also performed using the analytic relations to provide a reasonable speedup comparison. Finally, when multiple geometries of the analytic relations are combined such as in Section 2.5, the speedup factor will be reduced accordingly. However, the combination of many geometries will still provide a speedup of multiple orders of magnitude.

2.12 Potential Applications of Analytic Aerodynamics

Many of the analytic aerodynamic relations provide an exact solution to Newtonian flow theory, eliminating the numerical error associated with panel methods. These relations, in addition to the hybrid exact-approximate solutions, also reduce the computational requirements, compared to panel methods and CFD, for the aerodynamic analysis of hypersonic vehicles. This allows for rapid aerodynamic calculations when performing trade studies or shape optimization. Additionally, the evaluation of many of these aerodynamic relations is nearly instantaneous in which the aerodynamics module within traditional design methodologies can transition from being among the slowest to the fastest module during hypersonic vehicle conceptual design.

Typically during conceptual design, optimal trajectories are identified to determine the nominal aerodynamic performance requirements of hypersonic vehicles, such as peak L/D and ballistic coefficient. A vehicle can then be constructed to meet these aerodynamic constraints as well as other geometric constraints. However, in some cases, high performance hypersonic trajectories are not flown at a constant angle of attack, resulting in L/D and ballistic coefficient varying throughout the trajectory. Consequently, an iterative process must be performed until convergence between vehicle shape and time-varying aerodynamic performance required by the trajectory is observed. Furthermore, within the trajectory optimization process today, the mapping of vehicle shape to aerodynamic performance is largely performed using tables of aerodynamic coefficients as a function of angle of attack and sideslip.

If direct methods are used for hypersonic trajectory optimization, then the addition of shape design parameters would simply augment the overall optimization problem.[60, 61] In this case, the analytic aerodynamic relations would be directly integrated into the trajectory simulation, dramatically reducing the computational requirements of evaluating the aerodynamic coefficients along each candidate trajectory. Additionally, any constraints on the shape parameters would constrain the search domain of the direct optimization method.

If indirect methods are used to perform trajectory optimization, then the analytic mapping of vehicle shape to aerodynamic performance would allow simultaneous trajectory and vehicle shape optimization using indirect methods. In this approach, the analytic sensitivities (or variations) of the trajectory utilized by indirect methods can be extended to also make use of the analytic sensitivity information provided by the analytic aerodynamic relations. This is accomplished by directly incorporating the analytic relations into the equation of motion, modeling the varying L/D and ballistic coefficient throughout the trajectory. Additionally, constraints on vehicle shape such as diameter and volume can be incorporated into the overall optimization process.

Predictor-corrector guidance algorithms require aerodynamic estimates for all possible combinations of control. As such, large aerodynamic tables have been constructed for various combinations of angle of attack and sideslip. While high-fidelity tables derived from CFD analysis would be required for fielded systems, the analytic aerodynamic relations could serve as a substitute during the conceptual design and evaluation of candidate onboard predictor-corrector guidance algorithms. This benefit is strengthened if configuration changes are present during the hypersonic trajectory in which the analytic relations could also substitute for the additional aerodynamic tables required for each configuration change. Additionally, the propagation of trajectories generally ignores the shape change due to ablation experienced by the vehicle throughout the hypersonic trajectory. Instead, ablation studies are usually performed after the design of both the trajectory and vehicle in order to estimate the thermal protection system thickness and mass. However, the analytic aerodynamic relations would also enable the modeling of shape change due to ablation during trajectory propagation.

2.13 Summary

In this chapter, analytic hypersonic aerodynamic relations are developed based on Newtonian flow theory. Many of these relations provide exact solutions that are traditionally approximated by panel methods for conceptual design. While panel methods enable straightforward analysis of complicated vehicle configurations, numerical comparisons with a state-of-the-art paneling tool, CBAERO, illustrate the three to four orders of magnitude reduction in computational resources provided by many of the analytic relations. More importantly, however, these relations provide an analytical mapping of vehicle shape to trajectory performance. These relations eliminate the large aerodynamic tables calculated using panel methods, enabling vehicle shape parameters to be directly inserted into the equations of motion to perform rapid simultaneous hypersonic aerodynamic and trajectory optimization. Additionally, these relations could be used to assist the design of onboard guidance algorithms as well as enable aerodynamic modeling of shape change due to configuration changes or ablation.

Analytic force coefficients, moment coefficients, and static stability derivatives were computed for basic shapes including conical frustums, spherical segments, cylindrical segments, and flat plates. Many common hypersonic vehicles of interest such as sphere-cones and blunted biconics can be constructed through the superposition of these basic shapes. As such, the corresponding aerodynamics of each basic shape can also be combined to form analytic relations for vehicles of interest. The analytic relations for these basic shapes were derived by performing the Newtonian surface integration of pressure coefficient analytically. An automated process was developed to perform this integration, and each basic shape was parametrized according to the characteristics of the shape. As a result, only one set of analytic relations must be developed for each basic shape. The analytic relations account for both shadowed and unshadowed angles of attack and sideslip and were validated using CBAERO or another paneling code. An example biconic shape design was performed to maximize payload packaging capability while maintaining required aerodynamic performance. This process required the evaluation of many shapes at full sweeps of angles of attack. While this process would have been time consuming when using panel methods, the analytic relations reduced the computational cost of this analysis by approximately three orders of magnitude. In order to obtain analytic relations for more general bodies of revolution, analytic force and moment coefficients were developed for second-order Bezier curves of revolution at unshadowed angles of attack and sideslip. These relations were parametrized by the location of control nodes associated with the Bezier curve. The resulting expressions are quite lengthy and illustrate the complexity of the integrations that are performed for more general shapes. This observation is further strengthened by the inability to arrive to closed-form solutions at shadowed angles of attack and sideslip.

To reduce the complexity of certain existing relations and to obtain analytic solutions for new configurations, the traditional Newtonian model is converted into a mathematically equivalent model of a flux that passes through an assumed permeable outer-mold line of the vehicle. This technique enables the traditional Newtonian calculation to be converted into a volume integration of the divergence of the mathematical flux. The resulting compact relations for a spherical segment were shown to provide computational advantages over the traditional approach. Furthermore, analytic solutions of semi-quadratic potential functions were also possible using the Divergence Theorem. These solutions represent the current limit in exact, analytic relations. For more complex configurations, a portion of the aerodynamics can be calculated exactly while the remaining terms require numerical approximation. As an example, the parabola of revolution provides an efficient hybrid exact-approximate analytic solution in which only one integrated term in each coefficient must be approximated. Finally, comparisons with CBAERO illustrate the three to four orders of magnitude reduction in computational requirements of the analytic relations. The only exception to this observation is the Bezier curve of revolution in which the lengthy analytic expressions provide only a two to three orders of magnitude reduction.

CHAPTER III

RAPID HYPERSONIC TRAJECTORY OPTIMIZATION

3.1 Introduction

Current and historical trajectory optimization research has largely focused on design of an optimal trajectory for a given set of vehicle and mission assumptions. However, the solution to many optimal trajectories is desired during conceptual design to support design space exploration and trade studies. In these cases, designers are generally forced to choose a single method that is used repeatedly to construct these families of optimal trajectories. In many current design studies, direct methods are chosen due to their ease of implementation for a wide range of design problems.[53, 54, 55] Although these methods are robust to choice in initial guess, they are computationally intensive relative to indirect methods and, as a result, only a limited number of trajectories are evaluated. Indirect methods improve upon the computational requirements of direct methods through the use of necessary conditions of the optimal control problem.[34, 44] Satisfaction of the necessary conditions generally results in local optimality, but a good initial guess is often required to converge to a solution. Additionally, indirect methods increase the complexity of the optimal control problem through the introduction of costates, corner conditions, and switching structure.[1]

Historically, trajectory designers were required to choose between a direct and indirect method. Prior to the development of pseudospectral methods, designers were unable to map the results of one method to the other. However, with the development of the Covector Mapping Theorem (CMT), the results of specific direct methods can be mapped to the discrete results of indirect methods.[5] This enables designers to capitalize on the advantages of both methods. For this reason, pseudospectral methods have been widely used for modern trajectory optimization problems.[108, 109] Pseudospectral methods require an initial guess in states and control. In many studies, this guess is provided using designer intuition. However, using discrete dynamic programming, the construction of an initial guess can be automated for a wide range of hypersonic design problems. In current design studies, many initial guesses are provided to obtain a series of pseudospectral solutions used during analysis. However, as a direct method, pseudospectral methods are computationally intensive and limit the number of solutions analyzed.

In this study, a rapid trajectory optimization framework is presented that combines and advances the disparate trajectory optimization techniques developed over the previous century into a unified framework that is capable of solving a wide range of design problems. In this framework, discrete dynamic programming, nonlinear inversion, and pseudospectral methods are used to converge to an indirect solution. Once an indirect solution is found for a particular hypersonic problem, design space exploration can be rapidly performed using continuation. For trajectory optimization problems that are highly sensitive, have many switches in control, or have solutions that are not continuous with trajectory parameters, this robust optimization process is likely required. However, for well-behaved trajectory problems, only a subset of this process is necessary in which optimal trajectories can be identified through the continuation of indirect methods alone. This trajectory optimization framework serves as the foundation to also perform rapid simultaneous hypersonic aerodynamic and trajectory optimization through the inclusion of analytic hypersonic relations developed in Chapter 2.

3.2 An Alternative Perspective on Hypersonic Trajectory Design

Both direct and indirect methods were developed to identify optimal trajectories and their corresponding control histories without evaluating all possible solutions. However, the required control history to fly optimal trajectories is often of little interest to the designer, as long as it remains reasonably within control authority limitations. Therefore, in this investigation, a shift in focus from manipulating control histories to manipulating trajectory profiles is pursued to provide useful insight into the trajectory design space.

For this study, an unpowered, planar entry trajectory is assumed with equations of motion shown in Eq. (64)-(67), where t is the time, r is the radial magnitude, θ is the downrange subtended angle, v is the relative velocity magnitude, γ is the relative flight path angle, D is the drag force magnitude, m is the mass of the vehicle, μ is the gravitational parameter, L is the lift force magnitude, and ϕ is the bank angle. Trajectories are optimized to minimize total heat load. This objective was chosen to illustrate the optimality of results. To minimize heat load, the heat rate must be maximized along every portion of the trajectory, and this result is evident from the optimal solutions obtained. A high performance blunted biconic was chosen with geometric and aerodynamic parameters shown in Table 4. An entry mass of 136 kg was assumed, resulting in a ballistic coefficient, β, of 4400 kg/m^2.

$$\frac{dr}{dt} = v \sin\gamma \tag{64}$$

$$\frac{d\theta}{dt} = \frac{v\cos\gamma}{r} \tag{65}$$

$$\frac{dv}{dt} = -\frac{D}{m} - \frac{\mu \sin\gamma}{r^2} \tag{66}$$

$$\frac{d\gamma}{dt} = \frac{L\cos(\phi)}{mv} + \left(\frac{v}{r} - \frac{\mu}{vr^2}\right)\cos\gamma \tag{67}$$

Table 4: Biconic parameters.

Geometric Parameter	*Value*	*Aerodynamic Parameter*	*Value*
Length	1.22 m	α	10 deg
Nose Radius	0.025 m	C_D	0.157
Base Radius	0.25 m	C_L	0.307
Forward Cone Half-Angle	17 deg	L/D	1.96
Aft Cone Half-Angle	10 deg	β	4400 kg/m^2

This vehicle is assumed to have bank-only control with g-loading and heat rate constraints. The g-loading constraint could be the result of payload considerations or structural limitations, and the heat rate constraint is determined by the choice in thermal protection system (TPS) material. An exponential atmosphere and a spherical mass distribution of Earth is assumed. The initial parameters used in this study are shown in Table 5.

Table 5: Constraint and environment parameters.

Parameter	*Value*
Scale Height, H	7200 m
Surface Density, ρ_o	1.217 kg/m^3
Gravitational Parameter, μ	3.986e14 m^3/s^2
Earth Radius, r_e	6378000 m
Maximum G-Loading	30
Maximum Heating	3000 W/cm^2

This study presents a unified framework to perform rapid trajectory optimization by advancing and combining disparate optimal control techniques. An outline of this framework is shown in Figure 44 building on advances made in discrete dynamic programming, nonlinear inversion, pseudospectral methods, indirect methods, and continuation. In this methodology, the entry corridor is constructed in altitude-velocity space by eliminating regions that violate path constraints. Discrete dynamic programming is used to construct an initial guess in the entry corridor that is used by a pseudospectral method to converge to a solution. The costates provided by the pseudospectral method are used to converge to an indirect solution. With this indirect solution, rapid trajectory optimization can be performed for a wide range of entry problems using successive indirect solutions from continuation. The convergence of indirect solutions is rapid and forms the foundation to perform rapid trajectory optimization. The necessary conditions of optimality associated with indirect methods can be mathematically complex. In this framework, an automated process has been developed that uses symbolic manipulation tools to derive these necessary conditions.

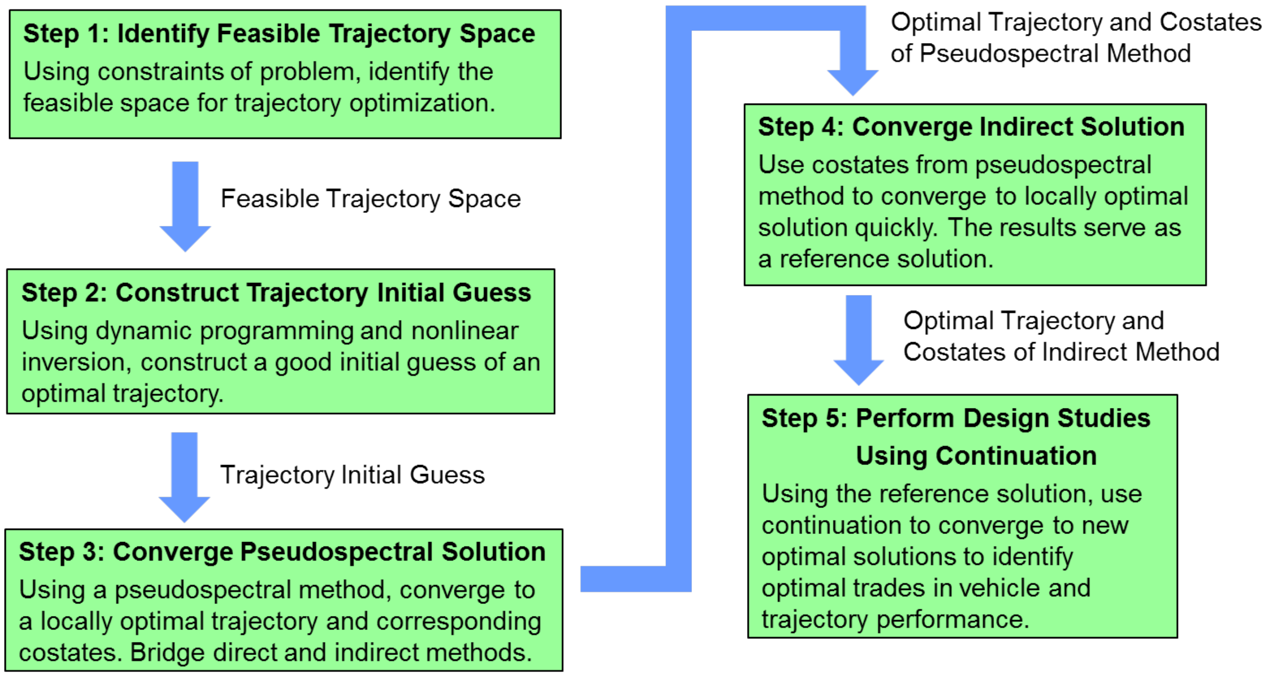

Step 1: Identify Feasible Trajectory Space
Using constraints of problem, identify the feasible space for trajectory optimization.
Feasible Trajectory Space
Step 2: Construct Trajectory Initial Guess
Using dynamic programming and nonlinear inversion, construct a good initial guess of an optimal trajectory.
Trajectory Initial Guess
Step 3: Converge Pseudospectral Solution
Using a pseudospectral method, converge to a locally optimal trajectory and corresponding costates. Bridge direct and indirect methods.
Optimal Trajectory and Costates of Pseudospectral Method
Step 4: Converge Indirect Solution
Use costates from pseudospectral method to converge to locally optimal solution quickly. The results serve as a reference solution.
Optimal Trajectory and Costates of Indirect Method
Step 5: Perform Design Studies Using Continuation
Using the reference solution, use continuation to converge to new optimal solutions to identify optimal trades in vehicle and trajectory performance.

Figure 44: Flowchart of the trajectory optimization framework.

3.3 Rapid Trajectory Optimization Methodology

3.3.1 Step 1: Entry Corridor Identification

Ultimately, trajectory designers are interested in constructing optimal trajectories that the vehicle should fly. While many trajectory analyses are performed with respect to time, construction of trajectories is most naturally accomplished in altitude-velocity space. Common constraints, including maximum heat rate and g-loading, can be constructed to remove trajectory regions in the lower portion of this space as shown in Figure 45. The upper portion of this space is traditionally bounded by a maximum altitude of approximately 120 km. This altitude represented the limit in atmospheric data from high-altitude weather balloons and, consequently, is traditionally chosen as the entry interface where entry simulations begin. Hypersonic vehicle trajectory optimization near this entry interface is inefficient since gravitational forces dominate aerodynamic forces, and, as a result, the vehicle lacks sufficient aerodynamic control authority to meaningfully alter its trajectory. Instead, an entry interface should be chosen at altitudes where the vehicle has sufficient control authority. For this example, a new entry interface, or control authority boundary, is chosen where the magnitude of drag is equivalent to the magnitude of gravity as shown in Figure 45. Unlike the traditional entry interface of constant altitude, the altitude of the control authority boundary is a function of velocity, vehicle, and celestial body. After eliminating the region of low control authority, trajectory optimization can then be performed throughout the remaining feasible space defined as the entry corridor. If the entry conditions at the traditional entry interface of 120 km are desired, then the state obtained at the control authority boundary can be propagated backward in time as necessary.

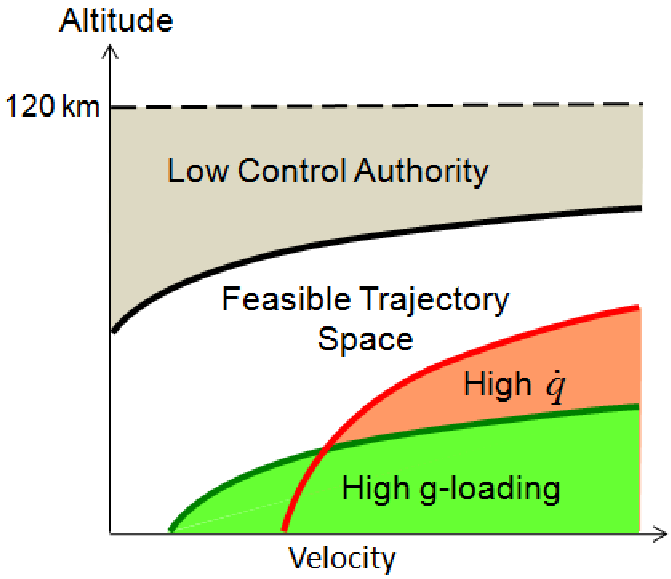


Figure 45: Entry corridor.

3.3.2 Step 2: Entry Corridor Discretization and Initial Guess Construction

In many optimal control algorithms, the addition of path constraints increases the complexity of the problem. However, as shown in Figure 45, constraints reduce the feasible space and, consequently, the number of trajectories that can be constructed. The construction of trajectories in the entry corridor is most naturally performed using discrete dynamic programming. In this approach, the entry corridor is discretized using a spherical, non-Cartesian mesh as shown in Figure 46. This mesh represents a set of potential waypoints that can be used to construct optimal trajectories. If vehicle control authority is neglected, then the trajectory optimization process resembles the pathfinding problem discussed in Section 1.3.5. With this assumption, a set of global optimal discrete paths can be efficiently constructed throughout the entry corridor using discrete dynamic programming. These paths correspond to trajectories the designer would like the vehicle to fly. Since control authority is neglected, the vehicle may or may not be capable of following this set of optimal waypoints.

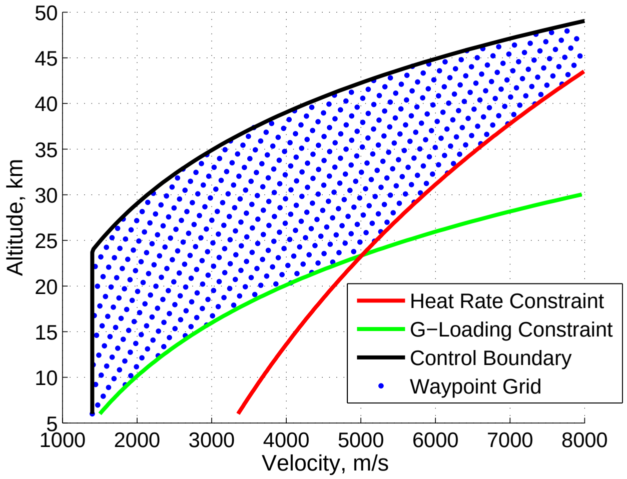


Figure 46: Example corridor waypoint grid.

Slender entry vehicles with substantial aerodynamic control authority will likely be able to follow the majority of the waypoints, whereas blunt entry vehicles with little control authority are not likely able to follow the discrete optimal paths. Therefore, the discrete dynamic programming solutions represent global optimal, unlimited control authority solutions that would generally serve as a better initial trajectory guess for slender entry vehicles than for blunt entry vehicles. Using a high-density mesh, several example discrete dynamic programming solutions are shown in Figure 47, differing in initial altitude and velocity. As expected for this problem, all of the obtained solutions travel along the constraint boundaries. For this study, the discrete dynamic programming solution shown in Figure 48 is chosen as an initial guess. This initial guess will enable a pseudospectral method to converge to a nearby optimal solution that can be flown. The discrete dynamic programming solution only provides a good initial guess in altitude and velocity. The trajectory is approximated using line segments between the nodes from the dynamic programming solution, and the remaining states and control required to construct an initial guess for pseudospectral methods are computed using nonlinear inversion.

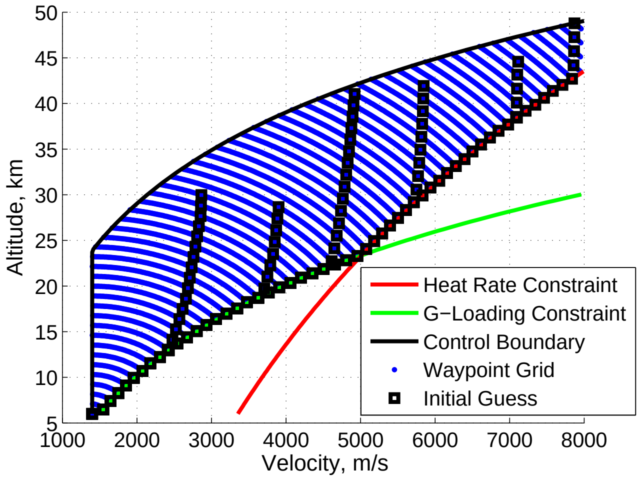


Figure 47: Potential initial guesses from various discrete dynamic programming solutions.

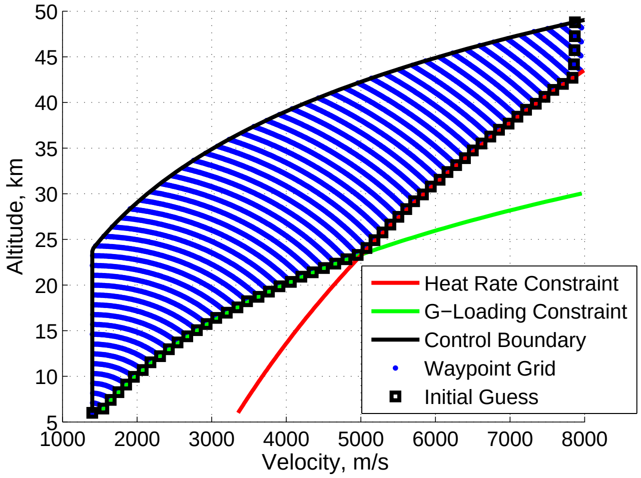


Figure 48: Dynamic programming solution selected for initial guess.

After the trajectory is constructed in altitude-velocity space from discrete dynamic programming, the flight-path angle (FPA) can be obtained from Eq. (68) through inversion of the nonlinear equations of motion. As shown, the FPA is a function of the slope and location of the trajectory in altitude-velocity space. Note that as $\frac{dh}{dv}$ becomes large, the right-hand side of Eq. (68) may be larger than 1, signifying an infeasible path in altitude-velocity space based on entry dynamics. Furthermore, as $\frac{dh}{dv} \to \infty$, $\sin\gamma \to -\frac{D/m}{\mu/r^2}$. This ratio between drag and gravitational acceleration presents a useful constraint that reduces the number of feasible path options that must be evaluated during the discrete dynamic programming process. If the vehicle is below the control authority boundary, then drag dominates gravity. Thus, all trajectories throughout the entry corridor in altitude-velocity space must have a finite, negative slope since the vehicle is not capable of reducing its altitude without reducing its velocity. Note that this constraint is due to entry dynamics and is independent of the objective function and vehicle control authority. The remaining quantities of interest (time, downrange angle, and bank angle) can also be computed using nonlinear inversion.

$$\sin\gamma = \frac{\frac{dh}{dv}\left(-\frac{1}{2m}\rho V C_D A_{ref}\right)}{1+\frac{dh}{dv}\frac{\mu}{r^2 V}} \tag{68}$$

3.3.3 Step 3: Pseudospectral Method Execution

The Gauss Pseudospectral Optimization Software (GPOPS) was selected to converge to a nearby solution using the initial guess from dynamic programming and nonlinear inversion. Like other direct methods, the pseudospectral method is used to converge to a minimum heat load solution that also satisfies the equations of motion. Additionally, in-flight constraints such as heat rate and g-loading limitations are satisfied and the corresponding KKT multipliers are computed. However, unlike other direct methods, the KKT multipliers from pseudospectral methods can be accurately mapped to discrete costates associated with indirect methods using the CMT.[5] Thus, the pseudospectral method serves as a bridge between the intuitive direct methods and the fast indirect methods. The pseudospectral method is executed to obtain a single solution that is consistent in states, costates, control, and corner conditions.

An example converged solution is shown in Figure 49 using the dynamic programming initial guess from Figure 48. There are two regions where the pseudospectral solution does not follow the dynamic programming initial guess. First, the vehicle (with limited control authority) is not capable of following the dynamic programming solution in the early part of the trajectory as shown in Figure 50. Hence, the vehicle does not have sufficient lift to enter as steep as the dynamic programming solution and execute the turn in trajectory necessary to satisfy the heat rate constraint. As a result, the pseudospectral solution converges to a more shallow entry flight path angle. Additionally, the vehicle appears to not be capable of following the dynamic programming solution along a portion of the g-loading constraint as shown in Figure 51. This particular vehicle has sufficient control authority to follow the g-loading constraint, and the departure from the constraint is an unfortunate result of the slow convergence of the pseudospectral method. In this trajectory optimization framework, the pseudospectral solution need not be fully converged. Instead, the solution is obtained for the sole purpose of providing a consistent set of states, costates, control, and corner conditions that serve as a good initial guess for convergence of indirect methods.

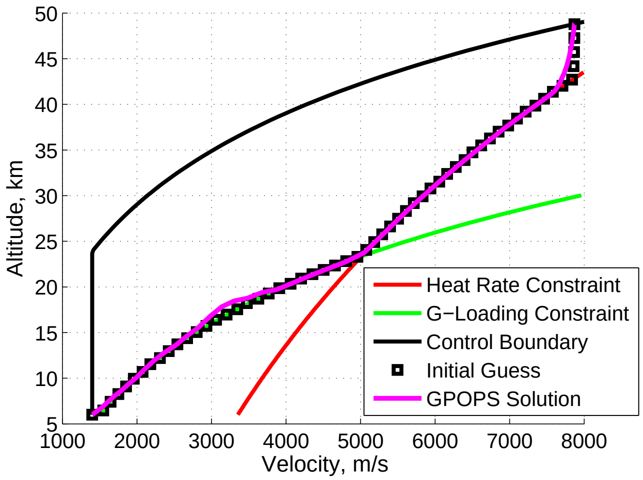


Figure 49: Converged pseudospectral solution.

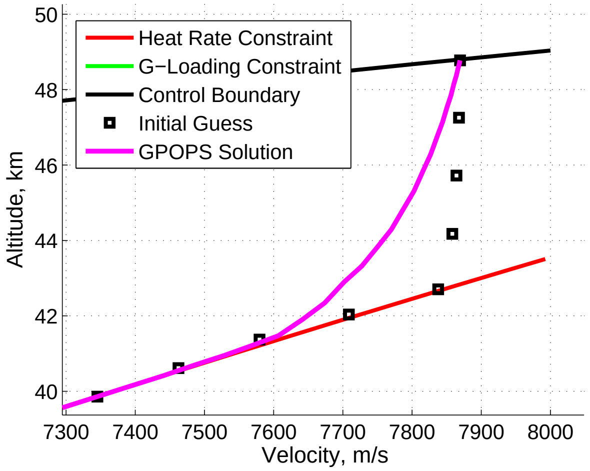


Figure 50: Beginning of converged pseudospectral solution.

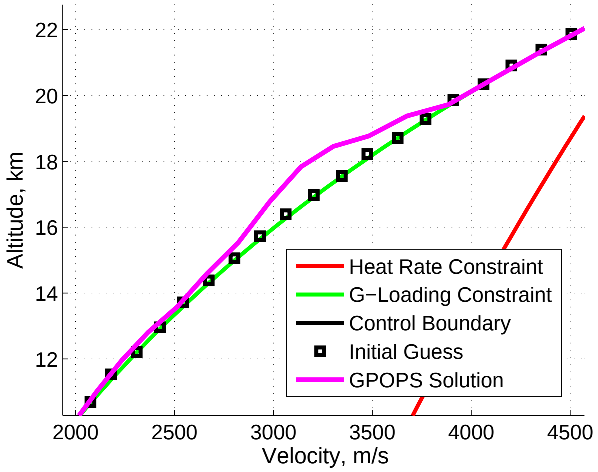


Figure 51: Converged pseudospectral solution along g-loading constraint.

As a direct method, execution of the pseudospectral method can be computationally intensive. Many current design studies use pseudospectral methods repeatedly to obtain the many optimal trajectories required for design space exploration and trade studies. If the pseudospectral solution is used as the final result for design studies, then improved convergence would be required along the g-loading constraint. This could be further improved through scaling techniques and use of analytic derivatives. Although GPOPS implements an automatic scaling feature, convergence performance can vary widely. Hence, the computational requirements to converge a single pseudospectral solution is highly dependent on designer interaction, and the time required for convergence can vary widely.

Initially, the solution obtained from GPOPS would not provide the necessary accuracy in costates to converge to an indirect solution when using Matlab's multi-point boundary value problem solver, BVP4C. This inability to converge to a solution can be attributed to the nonuniqueness of costates along constraints and varying forms of corner conditions. First, the costates do not have unique values along path constraints, such as g-loading or heat rate constraints, as illustrated in Figure 52.[7] The particular values in costates along the constraint are determined by the corner conditions. For this study, the costates were chosen to be continuous at the exit of the constraint. This assumption eliminates the nonuniqueness of costates along the constraint, and the discontinuity at the entrance of the constraint must provide continuous costates at the exit. However, the corner conditions provided by the pseudospectral method are not consistent with this assumption.

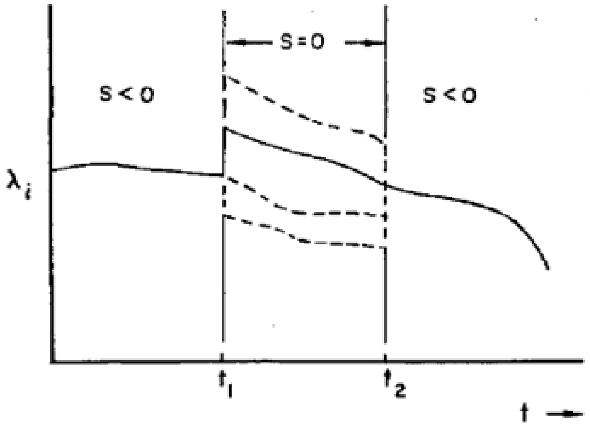


Figure 52: Nonuniqueness of costates.[7]

Corner conditions can be classified as a direct form (D-form) or Pontryagin form (P-form).[110] The KKT multipliers of GPOPS model the D-form corner conditions in which the multipliers are influenced directly by the path constraints. In this form, discontinuities can only occur in costates associated with states that explicitly appear in path constraints. In this study, the g-loading and heat rate path constraints are only a function of altitude and velocity. Consequently, no discontinuity will appear in the costate associated with flight path angle as shown in Figure 53. Alternatively, indirect methods implement P-form corner conditions in which path constraints and their derivatives influence the discontinuities in costates, and, consequently, discontinuities can occur in all costates. Both forms of corner conditions are equivalent, and the corner conditions obtained from GPOPS can converge to the corner conditions of the indirect method. However, the D-form corner conditions result in a singularity of the control switching structure used by indirect methods.

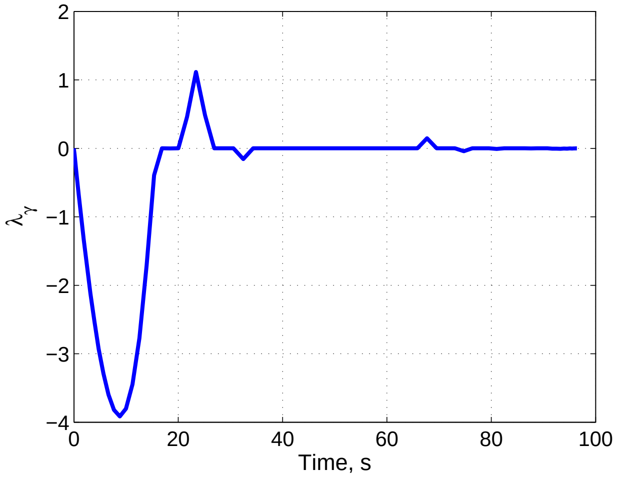


Figure 53: λ_γ of pseudospectral solution.

When deriving the necessary conditions of optimality, Pontryagin's Minimum Principle requires the Hamiltonian be minimized with respect to control for all time.[111] This results in a switching structure shown in Table 6 that governs the bank angle when the vehicle is not following a path constraint. Pseudospectral methods do not require use of a switching structure, and the D-form corner conditions are not influenced by the structure. However, as shown in Figure 53, the bank angle is indeterminant along the majority of the latter part of the trajectory. Consequently, perturbations used by boundary value problem solvers about this indeterminant solution results in numerical instability and prevents convergence of the indirect method. Although GPOPS lacks switching structure information, the pseudospectral method does provide the optimal control history along the converged solution. Information from the bank profile can be used to replace the switching structure to eliminate numerical difficulties. The bank profile history shown in Figure 54 is used to identify a bank angle of 0 deg at times along unconstrained trajectory arcs ($t < 15s$ and $t \approx 67s$). With this choice in bank angle along unconstrained arcs, the indirect method is able to converge to a solution. The oscillatory bank profile illustrates the challenge of direct methods to follow path constraints.

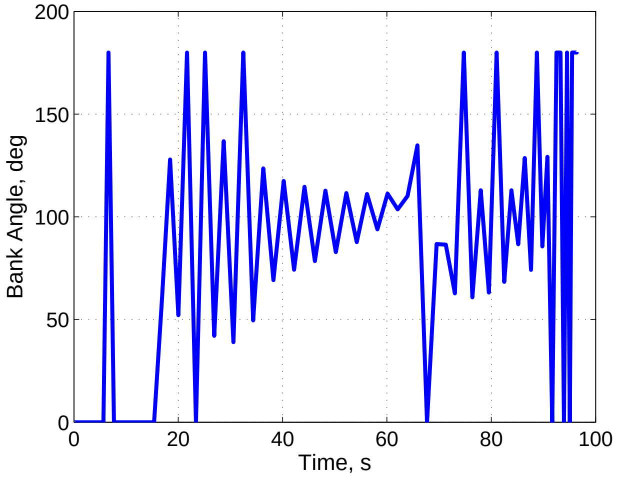


Figure 54: Bank angle of pseudospectral solution.

Table 6: Control switching structure.

$\lambda_\gamma < 0$	Bank = 0 deg
$\lambda_\gamma > 0$	Bank = 180 deg
$\lambda_\gamma = 0$	Bank is indeterminate

3.3.4 Step 4: Indirect Method Convergence

The pseudospectral solution obtained in the previous step is used as an initial guess for the indirect method. As shown in Figure 55, the pseudospectral solution is within the region of attraction, allowing rapid convergence of the indirect method. Convergence of the indirect method is performed using Matlab's multi-point boundary value problem solver, BVP4C. As expected for slender entry vehicles, the indirect solution eliminates the temporary departure from the g-loading constraint. Additionally, the indirect solution further reduces heat load by traveling closer to the heat rate constraint than the pseudospectral solution in the beginning part of the trajectory as shown in Figure 56.

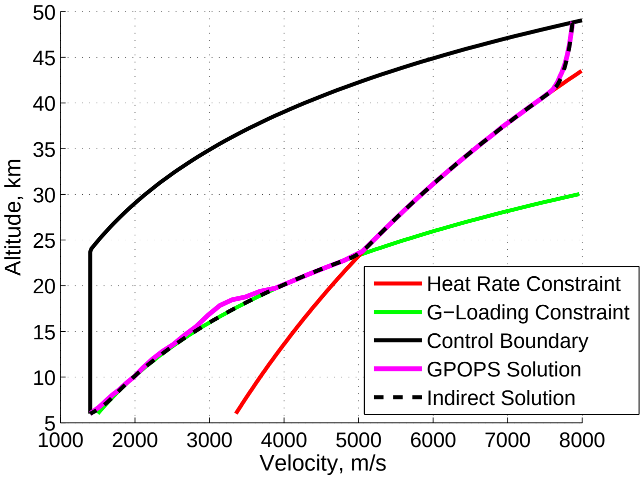


Figure 55: Fully converged indirect solution.

These steps provide an automated means to obtain a fully converged indirect solution in state, costate, control, initial conditions, terminal conditions, and corner conditions. This solution serves as a reference when performing trajectory design studies. The convergence of indirect methods is approximately two orders of magnitude faster than direct methods based on observations for this hypersonic problem. Note that this performance gap is both problem and algorithm dependent. Nearby optimal trajectories can be obtained by substituting the pseudospectral solution with

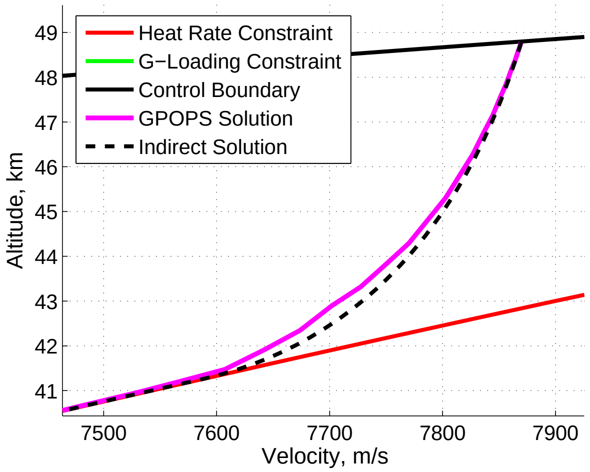


Figure 56: Beginning of converged indirect solution.

the reference solution as an initial guess. As such, nearby indirect solutions can be obtained in succession to rapidly obtain a family of indirect solutions using continuation.

3.3.5 Step 5: Design Using Continuation

Starting with the reference optimal solution obtained from the previous step, trajectory parameters can be incrementally varied to rapidly obtain a family of optimal trajectories using indirect methods and continuation. As an example, the family of optimal trajectories in Figure 57 is rapidly obtained over an approximate 30% change in initial velocity. Each trajectory obtained is a fully converged indirect optimal solution in states, costates, control, initial conditions, terminal conditions, and corner conditions. Using continuation, the optimal trade in initial velocity and heat load can now be constructed as shown in Figure 58. When performing continuation, the change in trajectory parameters must be small enough such that the initial guess from the prior indirect solution resides in the region of attraction of the indirect method. The convergence of the indirect method with relatively large changes in initial velocity illustrates that large changes can be made to trajectory parameters during the continuation process.

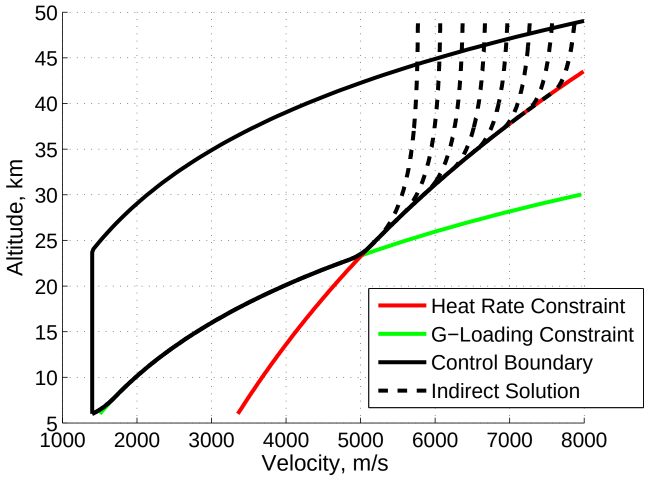


Figure 57: Optimal trajectories for varying initial velocity.

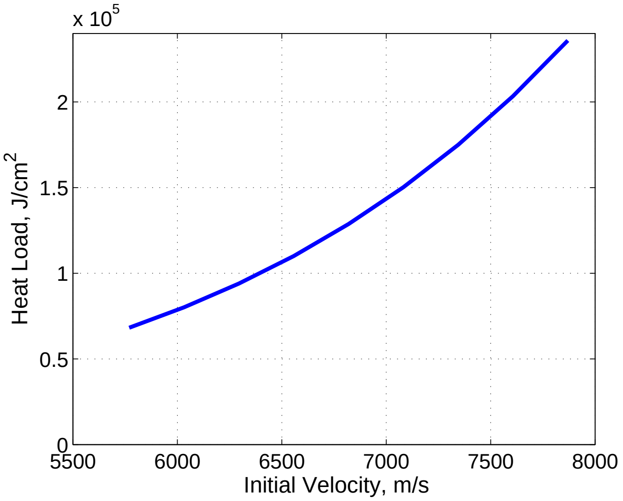


Figure 58: Optimal trade in initial velocity and minimum heat load.

3.4 Extensibility of Continuation

3.4.1 Additional Trajectory Parameters

The previous example demonstrates that initial state can be varied in the continuation process. However, additional trajectory parameters can also be varied, including terminal conditions and vehicle requirements such as heat rate and g-loading constraints. The inclusion of these additional parameters allows the sensitivity of system-level requirements to be rapidly mapped to vehicle performance to address margin requirements. For example, the optimal trade in maximum heat rate, which governs the choice in TPS material, and minimum heat load, which governs TPS mass, can be rapidly constructed using continuation. The minimum heat load trajectories corresponding to various heat rate constraints are shown in Figure 59, and the optimal trade in heat rate and heat load is shown in Figure 60.

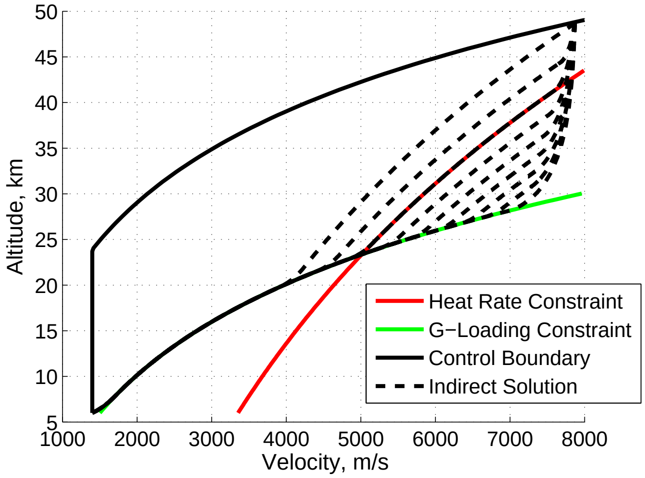


Figure 59: Optimal trajectories for varying heat rate constraint.

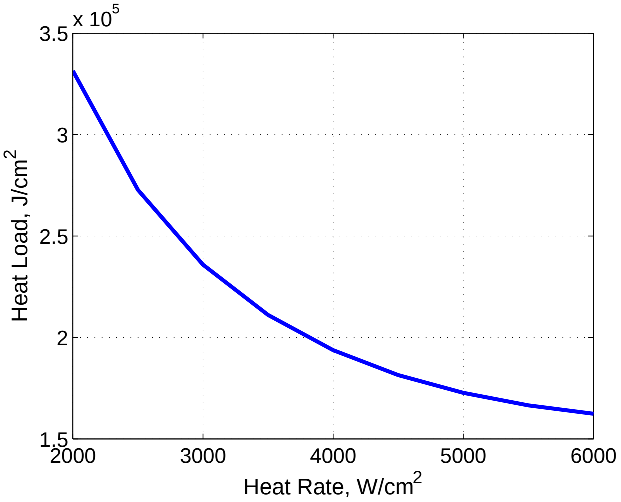


Figure 60: Optimal trade in heat rate constraint and minimum heat load.

For this example, the minimum heat load appears to asymptote near 1.5e5 J/cm^2 as the maximum allowable heat rate increases. The asymptotic behavior is due to the particular choice in g-loading constraint. As the maximum heat rate constraint increases, the vehicle travels along greater portions of the g-loading constraint. At a maximum heat rate near 6000 W/cm^2, the heat rate constraint lies entirely within the constrained g-loading region. Thus, the vehicle travels directly to the g-loading constraint at these high heat rate values, and the trajectory is no longer influenced by changes in the heat rate constraint. Optimal trades in heat rate and heat load can be obtained for various g-loading constraints, enabling structural or payload requirements (g-loading) to be rapidly linked to thermal protection system performance (heat rate and heat load) such that the integrated system performance can be rapidly evaluated.

3.4.2 Vehicle Shape

Although indirect methods have largely been developed to solve optimal control problems, the continuation process is not limited to changes in trajectory parameters. Chapter 2 presents the development of analytic hypersonic relations as a function of vehicle shape, angle of attack, and sideslip.[112] These relations were validated with a current state-of-the-art hypersonic aerodynamics design tool, CBAERO. Unlike panel methods (that include CBAERO), the analytic mapping of vehicle shape to aerodynamic performance allows vehicle shape to be easily included in the continuation process. As an example, the slender biconic used during the continuation of trajectory parameters is evolved to a more blunt biconic as shown in Figure 61. As the vehicle shape is altered, minimum heat load trajectories converge rapidly using continuation. The resulting trajectories are shown in Figure 62. Since the vehicle shape is incrementally modified, an enlarged view of the trajectories is shown in Figure 63 where the constraints shown correspond to the reference indirect solution.

In this example, the bluntness of the vehicle was increased without modifying the nose radius. Thus, the stagnation point heat rate used for this analysis was unaffected

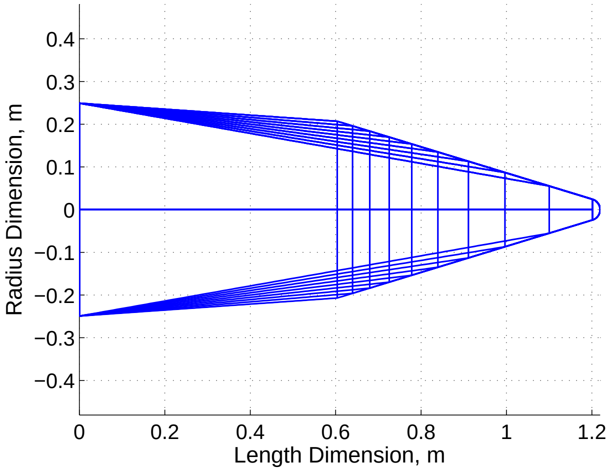


Figure 61: Vehicle shape change.

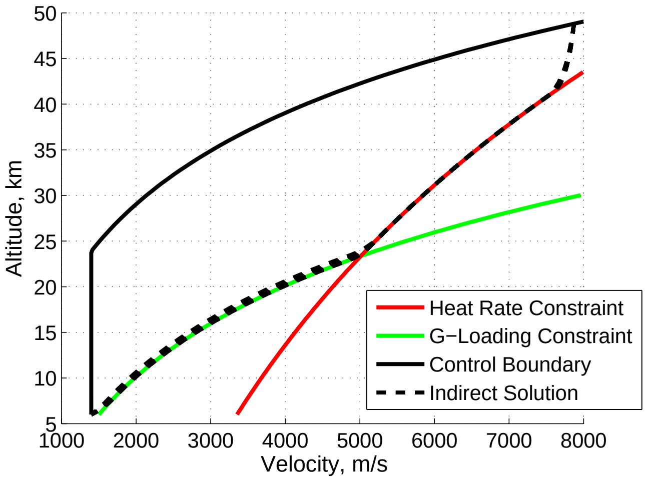


Figure 62: Optimal trajectories for varying vehicle shape.

by changes in vehicle shape, and all minimum heat load solutions follow the original heat rate constraint. However, the lift and drag coefficients and, consequently, g-loading are dependent on vehicle shape change. Thus, as the bluntness of the vehicle increases, trajectories must be flown at higher altitudes to satisfy the same g-loading constraint. Evidence of this result is shown in Figure 63 for the relatively small change in vehicle bluntness used for this example. Additionally, the equations of motion are satisfied along each new trajectory, a result from the convergence of the indirect method. Using continuation, the optimal trade in heat rate and ballistic coefficient

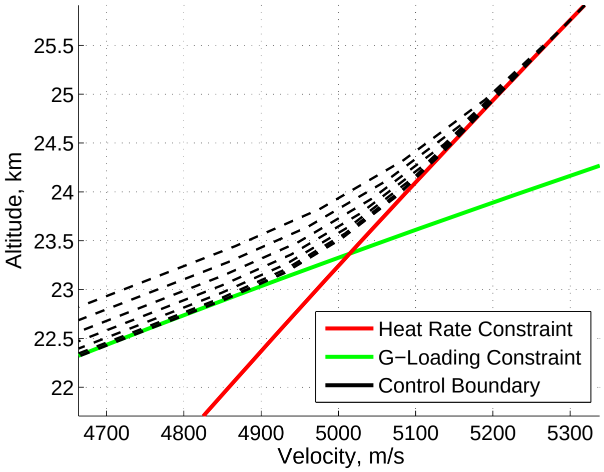


Figure 63: Optimal trajectories for varying vehicle shape (enlarged at original constraint intersection).

can be rapidly constructed for this range of shapes as shown in Figure 64. Although the altitude-velocity trajectory of each vehicle shape is similar, the downrange and time of each trajectory differ as shown in Figure 65. As the ballistic coefficient is decreased, the time of the trajectory also decreases, resulting in reduced optimal heat loads.

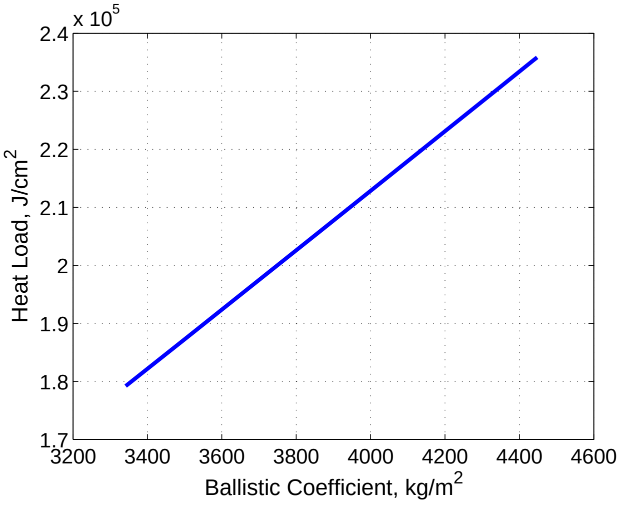


Figure 64: Optimal trade in ballistic coefficient and minimum heat load.

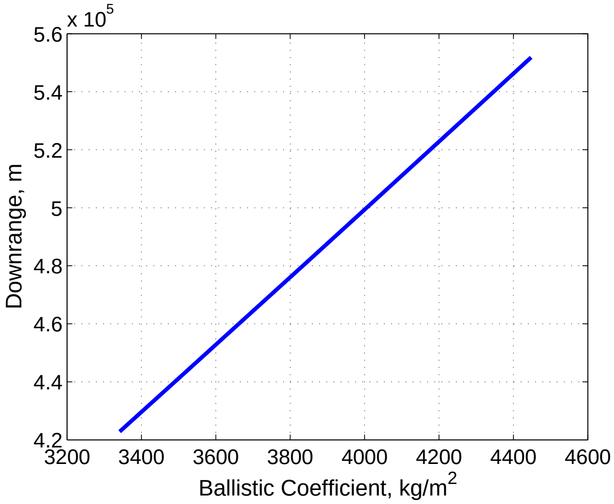


Figure 65: Downrange vs. ballistic coefficient for minimum heat load trajectories.

3.4.3 Additional Parameters

The prior examples that modify trajectory parameters and vehicle shape demonstrate common trades of interest for a particular entry problem. However, the continuation process can be extended to include any parameter associated with the analysis. For example, the scale height and surface density of the exponential atmosphere can be modified using continuation to converge to optimal trajectories associated with different atmospheres. Minimum heat load trajectories for varying scale heights and surface densities are shown in Figure 66 and Figure 67, respectively. As the atmosphere is modified, the sensed acceleration and stagnation point heat rate constraints move in altitude-velocity space. Although only the reference constraints are shown, all indirect solutions satisfy the new constraint locations. In addition to the atmosphere, gravity can be modified to rapidly evolve Earth-based optimal trajectories to optimal trajectories associated with other celestial bodies such as Mars and Titan using continuation.

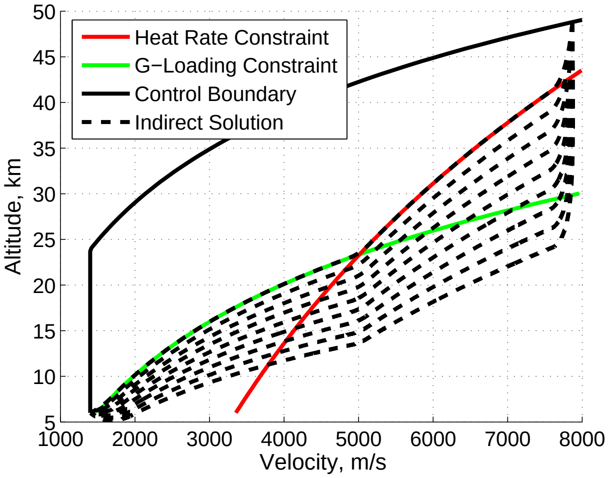


Figure 66: Optimal trajectories for varying atmospheric scale height.

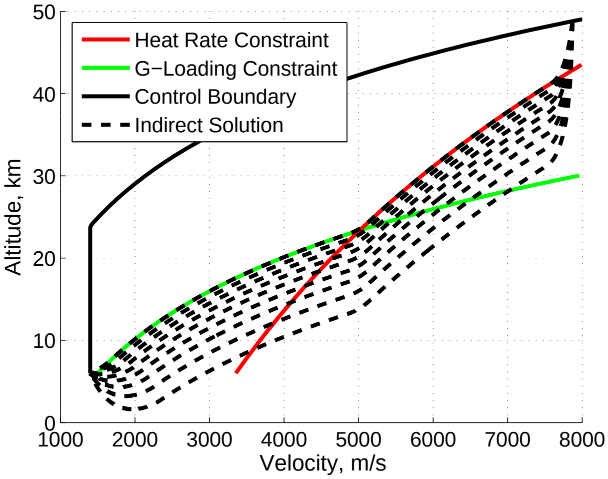


Figure 67: Optimal trajectories for varying atmospheric surface density.

The continuation process is not limited to varying a single set of parameters as shown in the prior examples. All parameters associated with the problem can be modified simultaneously to obtain solutions to new problems of interest as well as to identify optimal trades in vehicle and trajectory performance. As an example, the initial velocity, terminal altitude, vehicle geometry, g-loading constraint, heat rate constraint, surface density, scale height, and gravity are all varied simultaneously according to Table 7, and the resulting minimum heat load trajectories are shown in Figure 68. Because many entry problems are physically related, they can be rapidly solved using continuation and indirect methods. Finally, this methodology is not limited to entry problems and could also be applied to other hypersonic flight applications such as aerocapture or access to space.[113]

Table 7: Ranges of parameter changes from reference solution.

Parameter	*Value*
Initial Velocity	[-1000,0] m/s
Terminal Altitude	[2000,15000] m
Aft Cone Half Angle	[-9,0] deg
G-Loading Constraint	[-5,20] Earth-g
Heat Rate Constraint	[-500,1800] W/cm^2
Surface Density	[-0.3,0] kg/m^3
Scale Height	[-1500,0] m
Gravitational Parameter	[-5/8,0]*3.986e14 m^3/s^2

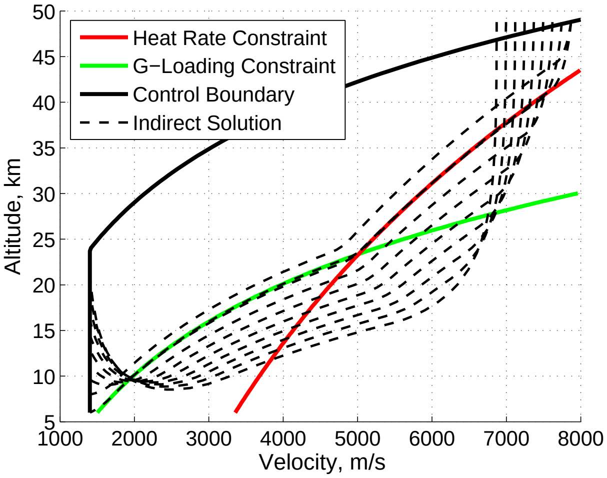


Figure 68: Optimal trajectories from variation of all parameters.

3.5 *Rapid Design Space Exploration for Conceptual Design of Hypersonic Missions*

The rapid trajectory methodology described in Section 3.3 is formed through the combination and advancement of disparate trajectory optimization techniques that have been developed over the previous century into a unified framework that is capable of solving a wide range of design problems. Specifically, this framework implements discrete dynamic programming, nonlinear inversion, pseudospectral methods, indirect methods, and continuation. The use of pseudospectral methods provides a robust capability to converge to an initial indirect solution (reference solution) from which design studies can be rapidly performed through continuation of indirect methods. This initial solution in states, costates, control, corner conditions, and switching structure can be constructed for complex trajectories with path constraints, an otherwise difficult task to perform. By appropriately balancing the advantages and disadvantages of both direct and indirect methods, this methodology is capable of performing rapid design studies for a range of vehicle shapes, constraints, environment parameters, initial conditions, and terminal conditions. As such, the hypothesis that hypersonic trajectories can be efficiently connected in this manner has been confirmed.

For trajectory optimization problems that are highly sensitive, have many switches in control, or have solutions that are not continuous with trajectory parameters, this robust optimization process is likely required. However, for well-behaved trajectory problems, only a subset of this process is necessary in which optimal trajectories can be identified through the continuation of indirect methods alone. For these trajectory problems, this alternative approach eliminates several limitations of the full trajectory optimization methodology, enabling rapid design space exploration for the conceptual design of hypersonic missions.

3.6 Prior Methodology Limitations

3.6.1 Framework Complexity

In order to successfully execute the full design methodology, several optimization techniques are required. These techniques include discrete dynamic programming, nonlinear inversion, pseudospectral methods, indirect methods, and continuation. The integration of these algorithms greatly increases the complexity of the process and would likely limit its adoption by the design community. As such, a simplified methodology is needed that reduces the number of required optimization techniques.

3.6.2 Path Cost Limitation and Sensitivity of Discrete Dynamic Programming

In the prior methodology, path constraints, such as maximum heat rate and g-loading, as well as design constraints, such as minimum control authority, are shown to reduce the number of trajectory options that define an entry corridor. As such, discrete dynamic programming is used to construct a good initial guess within this corridor. While this approach is very efficient to minimize path cost throughout the corridor, incorporating terminal cost into the objective would be problematic. Furthermore, the dynamic programming solution provides a good initial guess within altitude-velocity space. As such, the remaining states must be estimated using nonlinear inversion of the altitude-velocity solution. Along certain portions of the trajectory, the estimation of flight path angle is highly sensitive to the slope of the altitude-velocity solution. For certain problems, these numerical challenges may reduce the quality of the initial guess provided by discrete dynamic programming. These challenges could be addressed through various grid densities or adaptive mesh techniques, but these solutions would continue to add complexity into the process. As such, an alternative means of constructing a quality initial trajectory is needed that is capable of incorporating both path and terminal cost without the use of discrete waypoints.

3.6.3 Pseudospectral Complexities

In Section 3.3, the solution from a pseudospectral method is used to provide a good initial guess required for the continuation of indirect methods. Pseudospectral methods approximate the trajectory as a polynomial by collocating nodes at efficient locations such as the Legendre-Gauss-Lobatto (LGL) points shown in Figure 69. Note that for a given number of nodes, the location of these nodes is fixed and problem independent. Additionally, these nodes are highly clustered near the endpoints. While this reduces the onset of the Runge phenomenon when interpolating high order polynomials near the endpoints, high dynamic regions of many hypersonic missions, such as entry, occur in the middle of the trajectory where the dynamic pressure peaks. This is observed when comparing the LGL node locations with the node locations of an error-controlled, adaptive Runge-Kutta (RK) method for a representative Earth hypersonic problem shown in Figure 69. Without assuming a form of the solution *a priori*, the adaptive RK method clusters nodes during high dynamic periods which occur near the center of the trajectory.

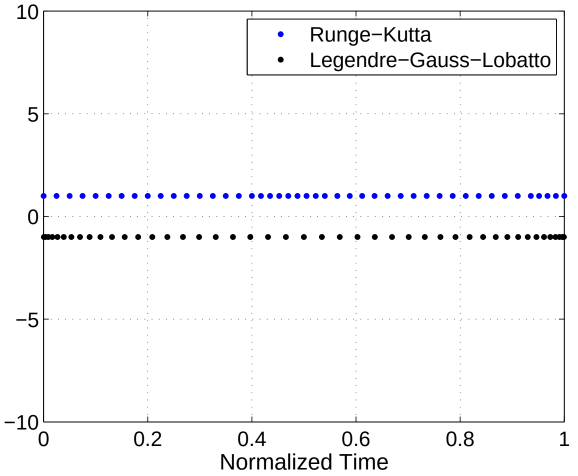


Figure 69: Comparison of node locations for a representative Earth entry problem (45 nodes for both methods).

Collocating nodes at the LGL points also requires use of an NLP solver in which the equations of motion are enforced as constraints at the nodes. Due to the clustering of nodes at the endpoints, a sufficient number of nodes must be chosen to ensure accuracy of the interpolated dynamics during the high dynamic pressure region near the center of the trajectory. While adaptive mesh refinement methods have been developed to address this issue, successive trajectory optimizations with varying node counts and arrangements are required.[114, 115] Furthermore, NLP solvers such as SNOPT employ penalty functions that may result in additional complexity during convergence to the optimal solution. Finally, the converged pseudospectral solution is mapped to an indirect solution using the Covector Mapping Theorem.[71, 5] The mapping requires identification of the discontinuous changes in costates that occur at the entrance of path constraints. This identification can be challenging to accomplish in an automated manner when this discontinuity resides between the collocated nodes.[1, 116, 44] As such, a methodology that automatically adapts to the dynamics of the problem and enables direct computation of the costates is needed to construct a good initial guess for the reference indirect solution.

3.6.4 The Disparity Between Conceptual Design and Onboard Operations

In general, the open-loop optimal trajectories simulated during conceptual design are not consistent with the closed-loop trajectories flown by fielded systems. When developing a flight vehicle, substantial effort is placed on the selection and development of a suitable guidance algorithm. Once a guidance algorithm has been selected, modifications are often made to tailor the algorithm to a specific mission.[50] Efforts such as these often complicate the design process and are generally omitted from conceptual design.

During conceptual design, optimal solutions are used to analyze various vehicle and trajectory combinations over a range of environment and mission requirements uncertainties that form the design space as notionally shown in Figure 70. Analogously, real-time trajectory planning would enable optimal trajectory solutions to be constructed during flight to identify the best path within a range of expected flight conditions that form a subset of this design space. As a result, traditional guidance algorithms would be replaced, and the conceptual design process would better reflect onboard operations. If the prior methodology is used, then an appropriate balance between the use of pseudospectral methods and continuation of indirect methods is required to perform efficient exploration of this design space. For example, if the pseudospectral solution creates an anchor point in this design space to seed the continuation of indirect solutions, then the designer must decide where to create these anchor points. Furthermore, many options likely exist when continuing from the anchor solution to another solution of interest as shown in Figure 71. The optimal continuation policy may influence the choice of anchor solution locations, but this policy is problem dependent and would likely require substantial computational resources. If the continuation of indirect methods is used to perform real-time trajectory planning, then this analysis would likely improve the onboard performance of the continuation process. However, this analysis would likely not be worthwhile during conceptual design due to the required increase in computational resources. Finally, while the prior methodology could provide a consistent rapid optimization framework for both conceptual design and real-time trajectory planning, the complexity resulting from the use of many disparate optimization techniques would likely lead to additional challenges during certification for onboard operations.

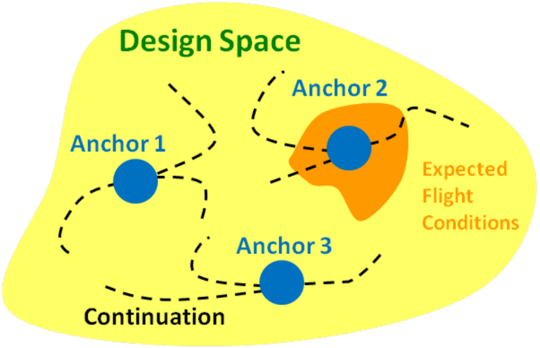


Figure 70: Example of design space exploration using continuation from anchor solution.

This prior methodology is analogous to many MDO algorithms in which relevant design solutions are immediately sought. As such, pseudospectral methods are required to seed the continuation process due to the difficulty of providing a sufficient initial guess to converge to an indirect solution, especially in the presence of path

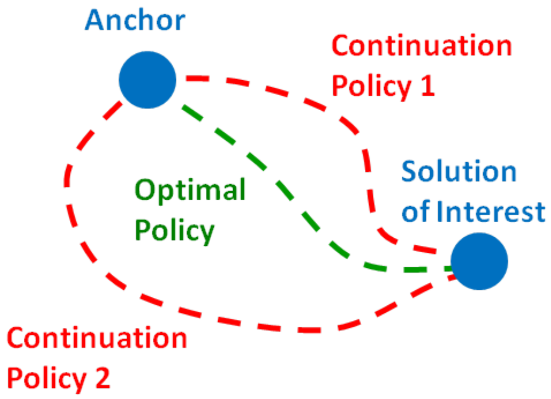


Figure 71: Example of optimal continuation policy.

constraints such as maximum heat rate and g-loading. If solutions of minimal immediate value to the designer are also considered, then optimal solutions throughout the design space can be efficiently constructed using continuation of indirect methods alone. This is accomplished by seeding the indirect method with a simple, short, unconstrained trajectory solution outside of the design space as opposed to a constrained pseudospectral solution within the design space. This greatly simplifies the process by eliminating the dynamic programming, nonlinear inversion, and pseudospectral steps of the prior methodology. As a result, a simplified methodology that relies only on the continuation of indirect methods is developed to address the aforementioned limitations and perform rapid design space exploration.

3.7 Rapid Design Space Exploration Using Continuation

For this study, a planar entry trajectory is again assumed with equations of motion shown in Eq. (64)-(67) and nominal environment parameters shown in Table 5 of Section 3.2. For comparison with the prior methodology, trajectories are again optimized to minimize total heat load.[113] To minimize heat load, the heat rate must be maximized along every portion of the trajectory, and this result is evident from the optimal solutions presented in Section 3.4. A high performance blunted conic is chosen with geometric and aerodynamic parameters shown in Table 8. An entry mass of 4080 kg is assumed, resulting in a large ballistic coefficient of 6,070 kg/m^2 for illustrative purposes in the following examples.

Table 8: Blunted conic parameters.

Geometric Parameter	*Value*	*Aerodynamic Parameter*	*Value*
Length	6.92 m	α	10 deg
Nose Radius	0.025 m	C_D	0.144
Base Radius	1.22 m	C_L	0.311
Cone Half-Angle	10 deg	L/D	2.16

3.7.1 Construction of the Initial Indirect Solution Outside of the Design Space

As previously mentioned, it is generally impractical for the designer to provide a sufficient initial guess within the constrained design space without the use of pseudospectral methods. Alternatively, the designer can easily provide a sufficient initial guess to converge to a short, unconstrained, optimal trajectory that is outside of the design space. For the following example, a terminal altitude of 6 km and velocity of 1.4 km/s is enforced. After choosing a reasonable terminal flight path angle of -49^o and terminal costate values of zero, an initial guess is constructed through reverse propagation of the states and costates for a short period of time which was chosen to be one second for this example. Before the trajectory is propagated, the lift of the vehicle is artificially reduced by a factor of 100. This prevents switches in control that could result along the unconstrained arc as described by the switching structure in Section 3.3.3. Finally, the initial altitude and velocity resulting from the reverse propagation are chosen as fixed initial conditions.

With this initial guess as well as fixed initial and terminal conditions, an indirect method is able to quickly converge to the short, unconstrained, minimum heat load solution. The resulting trajectory and costates are shown in Figures 72 and 73, respectively. As expected, the costates are smooth throughout the entire unconstrained trajectory. As with many common indirect method solvers, only the necessary conditions of optimality are guaranteed to be satisfied. The optimality of this short, unconstrained solution can be easily verified for this relatively simple trajectory optimization problem. As expected to maximize heat rate throughout the trajectory, the optimal solution consists of a full lift-up trajectory as described by the switching structure.

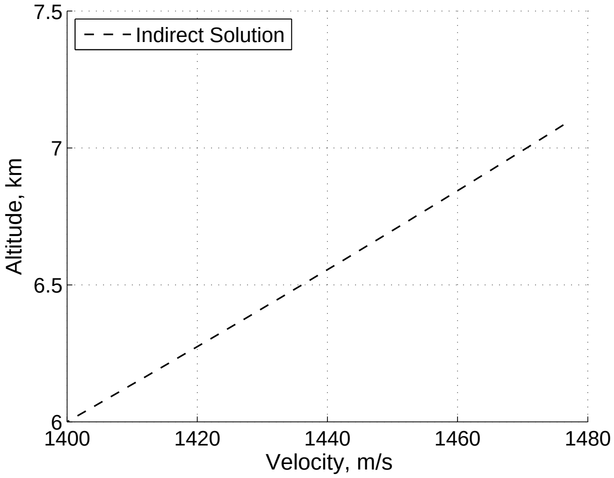


Figure 72: Initial indirect solution.

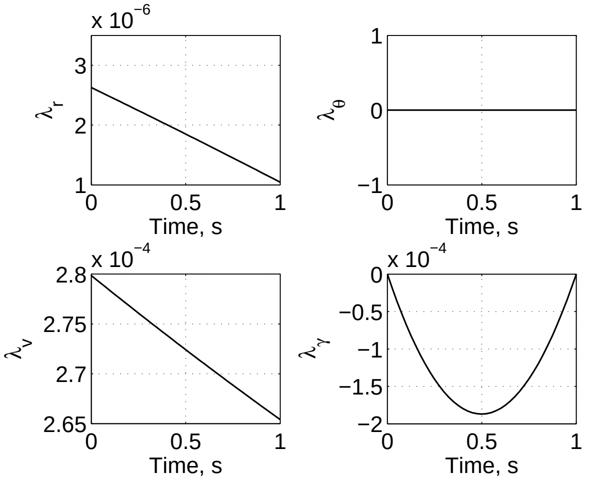


Figure 73: Costates from initial indirect solution.

3.7.2 Unconstrained Continuation to the Desired Initial State

The short, unconstrained trajectory is outside of the design space and is not of immediate interest to the designer. However, this solution serves as the starting point from which the trajectory is evolved to a solution of interest within the design space. With this converged indirect solution, the unconstrained trajectory must be extended through continuation to the set of desired entry conditions. The evolution of the optimal trajectories and corresponding costates is shown in Figures 74 and 75, respectively. During the continuation process, the prior solution serves as an initial guess to converge to the subsequent nearby solution. As the trajectory is lengthened, the switching structure is monitored to ensure no switches occur in the bank angle. If switches in control began to appear, then the lift would be decreased further. During this process, λ_γ is always negative, implying that the artificial decrease in lift is sufficient to prevent switches in control. Additionally, each successive solution is monitored during the continuation process to determine if large changes occur between solutions. For example, the initial flight path angle is monitored to determine if the vehicle is initially traveling towards the target as desired. If the distance between solutions is too large, then the indirect method solver may converge to a solution in which the vehicle is initially traveling away from the target. If this occurs, then smaller distances are chosen between solutions during the continuation process, ensuring convergence to a proper initial flight path angle.

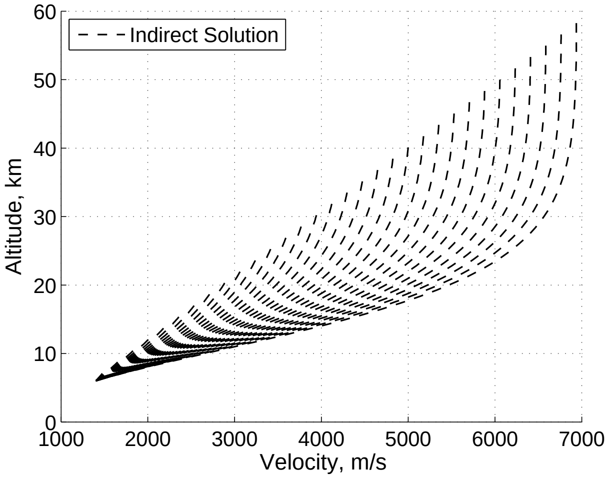


Figure 74: Trajectories from unconstrained continuation.

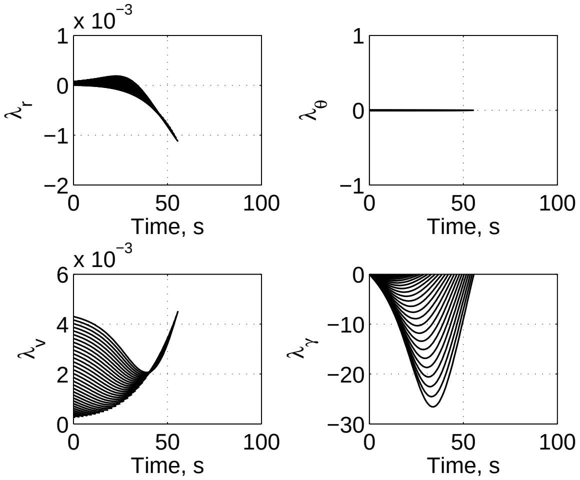


Figure 75: Costates from unconstrained continuation.

3.7.3 Introduction of Path Constraints

With the trajectory fully extended to include the desired initial conditions, path constraints such as maximum heat rate and g-loading are incrementally introduced. In this process, path constraints that define a flight corridor are introduced at a location of tangency to the unconstrained trajectory as depicted in Figure 76. This approach serves two purposes. First, indirect methods require the number and order of unconstrained and constrained trajectory arcs to be specified prior to optimization.

This is difficult to estimate *a priori*, and, in the prior methodology, this sequence is provided from the pseudospectral solution. After identifying the magnitude of the constraint that provides this tangency condition, a determination can be made as to whether or not the constraint is initially active and should be included in the optimization. Second, by introducing a constraint at the point of tangency, the states and time of the optimal trajectory remain unchanged. Only the costates change with the introduction of a discontinuity as a result of the corner conditions.[1, 44, 116] This provides a gradual change in solution necessary for convergence of the indirect method.

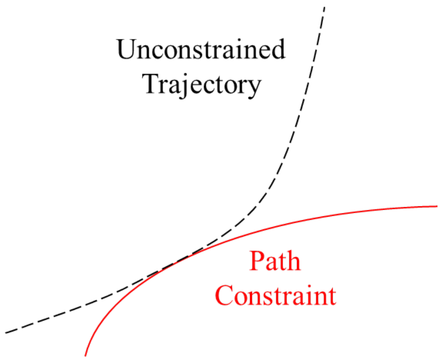


Figure 76: Introduction of path constraint.

In the prior unconstrained continuation process, lift is artificially decreased to prevent switches in bank angle. As path constraints are introduced, the lift of the vehicle must be incrementally restored to the physical value to enable the vehicle to maneuver along these constraints. However, only a sufficient amount of lift should be incrementally restored in this manner. If the lift restoration is performed too aggressively, then switches in control would likely appear even if the final optimal solution has no switches in control. If the lift restoration is performed gradually, then this process would likely also accommodate final solutions with a relatively small number of switches in control. However, sensitive problems with a large number of switches in control would likely require the robust, full trajectory optimization process in which the results from a pseudospectral method (that does not require a switching structure) is used to converge to the reference indirect solution.

As an example of the simplified process, the maximum heat rate constraint is introduced at a location of tangency along the final trajectory from the previous unconstrained continuation. Once the heat rate constraint is introduced in this manner, it is incrementally decreased to the desired value. As the heat rate constraint is decreased, the lift of the vehicle is restored to prevent saturation in bank angle while traveling along the constraint. To prevent switches in control for this problem, the lift is chosen to be restored to 50% of the actual value by the end of this process. The resulting trajectories and costates from this process are shown in Figures 77 and 78, respectively. As shown, the initial flight path angle must be shallowed as the heat rate constraint is decreased to prevent the vehicle from violating the constraint. As expected, discontinuities occur in the costates at the time the vehicle enters the heat rate constraint. Note that as the magnitude of the heat rate constraint is reduced and the lift of the vehicle is increased, both the length of the trajectory and time of the entrance to the heat rate constraint increase. It is also important to note that the switching structure only applies to the unconstrained portions of the trajectory. While λ_γ is shown to be positive during the middle portion of each trajectory, this occurs while the vehicle is traveling along the heat rate constraint and, as a result, does not conflict with the switching structure. As expected, λ_γ is verified to be negative along the unconstrained arcs.

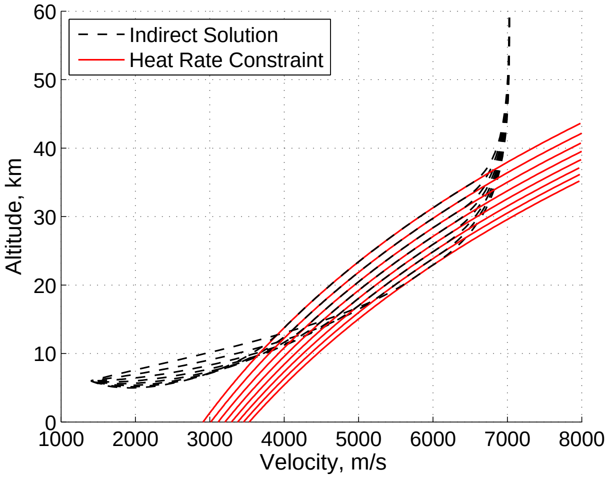


Figure 77: Trajectories from continuation of heat rate constraint and L/D.

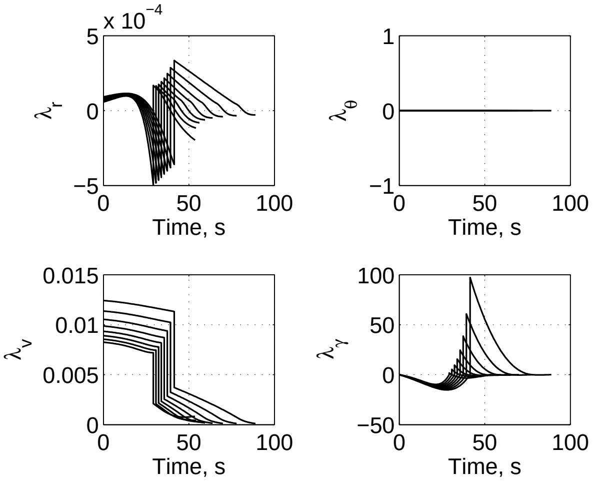


Figure 78: Costates from continuation of heat rate constraint and L/D.

After the continuation of the heat rate constraint is completed, additional path constraints such as maximum g-loading can also be introduced and modified in a similar manner. The trajectories and costates that result from the continuation of the g-loading constraint are shown in Figures 79 and 80, respectively. During this process, the lift of the vehicle is fully restored. As expected, a second discontinuity occurs where the vehicle enters the g-loading constraint at around 80 s. At the end of this process, a constrained indirect solution is obtained within the design space of interest. It is important to note that this process is executed quickly through continuation of fast indirect methods, eliminating the need for the slower processes of discrete dynamic programming, nonlinear inversion, and pseudospectral methods associated with the full trajectory optimization methodology. Consequently, this eliminates the need to construct anchor solutions at strategic locations throughout the design space. Instead, any point within the design space can be quickly accessed through continuation from a simple solution outside of the design space. As a result, rapid design space exploration can be performed using continuation of fast indirect methods alone.

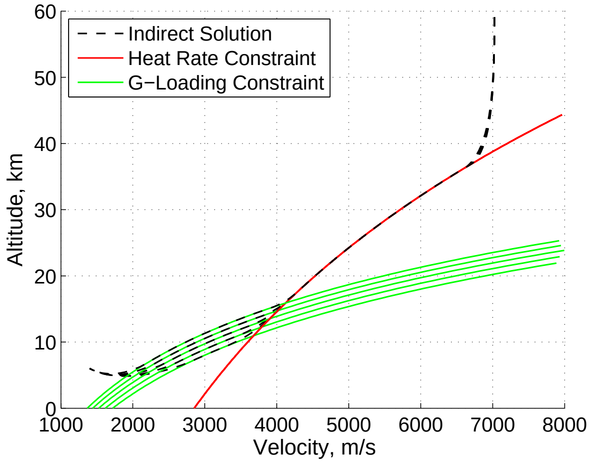


Figure 79: Trajectories from continuation of g-loading constraint and L/D.

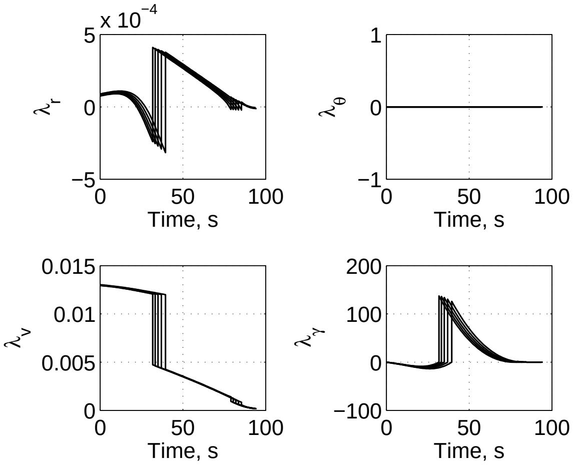


Figure 80: Costates from continuation of g-loading constraint and L/D.

In the prior example, algorithm monitors are used to ensure the initial flight path angle evolution and lift restoration provide feasible solutions during the continuation process. However, parameters are changed manually when evolving trajectory solutions. Without prior knowledge of optimal solutions during the continuation process, small steps must be conservatively chosen after trial and error to ensure that each prior solution serves as a good initial guess to converge to the next indirect solution. If indirect solutions could be predicted during the continuation process, then improved initial guesses could be constructed. Additionally, if the error of this prediction could also be estimated, then the choice in step size can be automated to perform efficient design space exploration.

3.8 Unconstrained Solution Prediction Using State Transition Tensors

In the prior example, the construction of optimal solutions within the design space begins with the continuation of unconstrained solutions to the desired initial condition. As such, the prediction of unconstrained solutions must be addressed. While traditional MDO methods generally rely on direct optimization techniques that perform a constrained search for the optimum, indirect methods convert the optimization problem into a root-solving problem of boundary conditions. If a relationship between these boundary conditions can be constructed, then optimal solutions can be predicted during the continuation process. This relationship is constructed using State Transition Tensors (STTs).

3.8.1 State Transition Tensors

In this work, STTs are used to relate changes in initial states and costates from a reference solution (denoted by *) to changes in states and costates at a future time as shown in Eq. (69), where $\mathbf{x}'_i$ is the i$^{\text{th}}$ component of an augmented state vector that also includes costates and $\mathbf{\Phi}_{i,k_1\cdots k_p}$ is the corresponding p$^{\text{th}}$ order STT. As a result, deviations in the augmented state with n elements can be expressed as a Taylor series of STTs to order m as shown in Eq. (70), and deviations in the rate of the augmented state can expressed in the form of Eq. (71). The equations of motion can be combined with the costate rate equations obtained from the necessary conditions of optimality to construct a relationship for the rate of the augmented state as shown in Eq. (72). The corresponding deviations in the rate of the augmented state can also be calculated by expressing Eq. (72) as a Taylor series as shown in Eq. (73), where the Jacobian of the augmented state is shown in Eq. (74). After substitution of Eq. (70) into Eq. (73), the STTs can be calculated by matching coefficients with Eq. (71).[117] An automated process has been developed to perform this coefficient matching for any order m chosen by the designer as well as to construct the appropriate subroutines for computing the STTs associated with the trajectory optimization of interest. After computing the STTs, predictions can be made during the unconstrained continuation process in both the states and costates for nearby optimal solutions.

$$\mathbf{\Phi}_{i,k_1\cdots k_p} = \frac{\partial^p \mathbf{x}'_i}{\partial \mathbf{x}'^o_{k_1} \cdots \partial \mathbf{x}'^o_{k_p}} \tag{69}$$

$$\delta \mathbf{x}'_i(t) = \sum_{p=1}^{m} \frac{1}{p!} \left(\sum_{k_1=1}^{n} \sum_{k_2=1}^{n} \cdots \sum_{k_p=1}^{n} \mathbf{\Phi}_{i,k_1\cdots k_p} \delta \mathbf{x}'^o_{k_1} \cdots \delta \mathbf{x}'^o_{k_p} \right) \tag{70}$$

$$\delta \dot{\mathbf{x}}'_i(t) = \sum_{p=1}^{m} \frac{1}{p!} \left(\sum_{k_1=1}^{n} \sum_{k_2=1}^{n} \cdots \sum_{k_p=1}^{n} \dot{\mathbf{\Phi}}_{i,k_1\cdots k_p} \delta \mathbf{x}'^o_{k_1} \cdots \delta \mathbf{x}'^o_{k_p} \right) \tag{71}$$

$$\dot{\mathbf{x}}'_i(t) = \mathbf{f}'_i[t, \mathbf{x}'(t)] \tag{72}$$

$$\delta \dot{\mathbf{x}}'_i(t) = \sum_{p=1}^{m} \frac{1}{p!} \left(\sum_{k_1=1}^{n} \sum_{k_2=1}^{n} \cdots \sum_{k_p=1}^{n} \mathbf{f}'^*_{i,k_1\cdots k_p} \delta \mathbf{x}'_{k_1} \cdots \delta \mathbf{x}'_{k_p} \right) \tag{73}$$

$$\mathbf{f}'^{*}_{i,k_1\cdots k_p} = \left.\frac{\partial^p \mathbf{f}'_i}{\partial \mathbf{x}'_{k_1}\cdots\partial \mathbf{x}'_{k_p}}\right|_{\mathbf{x}'=\mathbf{x}'^{*}} \tag{74}$$

Higher order STTs are often used in astrodynamics applications to maintain accuracy for the long propagation times required.[118, 119, 120, 121] For relatively short hypersonic missions, higher order STTs could potentially be used to accurately predict optimal solutions at greater distances from the reference solution. If the optimal solutions throughout the design space can be expressed as a convergent STT series, then the accuracy of the predictions will improve as the order of the STT approximation is increased. However, not all functions can be approximated by a convergent Taylor series. A scalar example of each scenario will illustrate the considerations made when predicting optimal solutions using STTs.

3.8.2 Taylor Series Examples

A scalar Taylor series approximation about the reference point a can be expressed in the form of Eq. (75), where $a = 0$ in the following examples. As shown in Figure 81, the Taylor series approximation of the function e^x appears to improve for all x as the order of the approximation increases. Alternatively, Figure 82 illustrates that the error of the Taylor series increases as the order of the approximation increases for the function $\log(1 + x)$ where $x > 1$. During the continuation process, the exact solution of the predicted optimal trajectories is not known *a priori*, and as a result, this comparison cannot be made to directly determine if the STT prediction improves as the order of the approximation increases. Instead, convergence of the STT series can be indirectly observed from convergence of the approximate STT solutions as the order of the approximation increases.

For a convergent series, the discrepancy between the approximate solution of order p, denoted as f_p, and of order $p+1$, denoted as f_{p+1}, should decay as p increases. For the convergent Taylor series example, this decay is observed at various values of x as shown in Figure 83. For the divergent Taylor series example, this decay is observed for $x < 1$ as shown in Figure 84. However, for $x > 1$, this decay only occurs for small orders of p. As p increases, the discrepancy also eventually increases. As shown in both examples, the discrepancy is likely to decrease as the order p increases for x near the reference point ($a = 0$). The discrepancy between an approximation of order p and an approximation of order $p+1$ is shown in Eq. (76). For $|x-a| << 1$, $(x-a)^{p+1}$ will likely dominate this expression, resulting in the observed decay in discrepancy for both examples near the reference point. However, as x is chosen farther from the reference point, the discrepancy will only decrease with order p if $\frac{f^{(p+1)}(a)}{(p+1)!}$ decreases in magnitude faster than $(x-a)^{p+1}$ increases in magnitude.

$$f(x+a) = f(a) + \frac{f^{'}(a)}{1!}(x-a) + \frac{f^{''}(a)}{2!}(x-a)^2 + \cdots + \frac{f^{(p)}(a)}{p!}(x-a)^p \qquad (75)$$

$$|f_{p+1} - f_p| = \left|\frac{f^{(p+1)}(a)}{(p+1)!}(x-a)^{p+1}\right| \qquad (76)$$

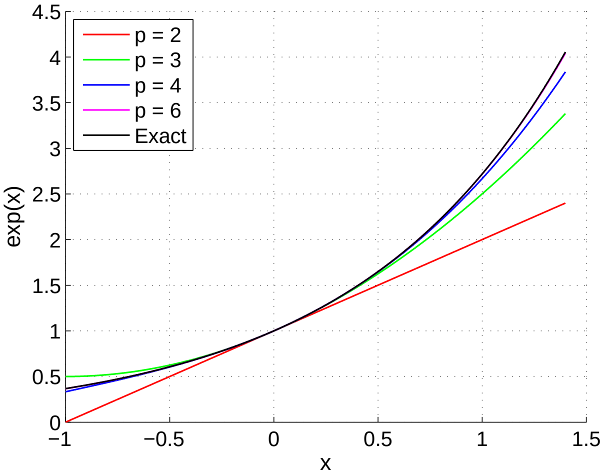


Figure 81: Example convergent Taylor series approximations for $\mathbf{f(x) = e^x}$.

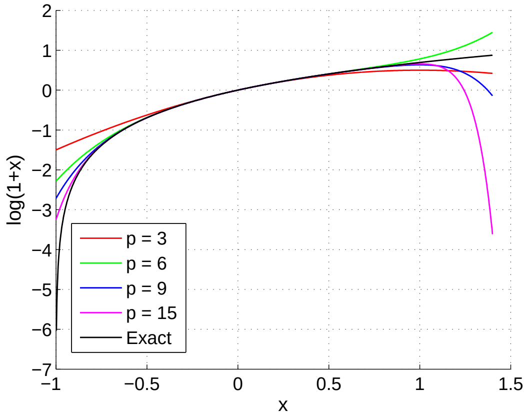


Figure 82: Example divergent Taylor series approximations for $\mathbf{f(x) = log(1 + x)}$.

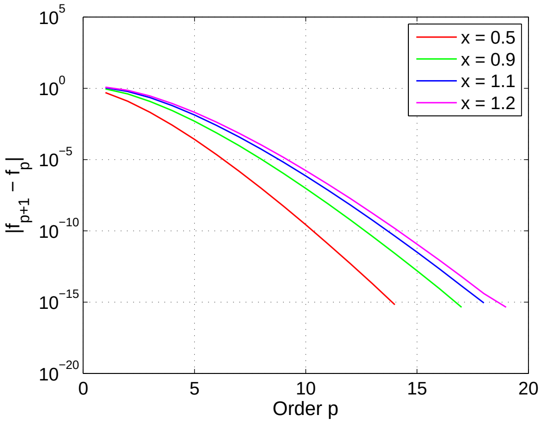


Figure 83: Discrepancy in convergent Taylor series estimates at various $\mathbf{x}$ locations for $\mathbf{f(x) = e^x}$.

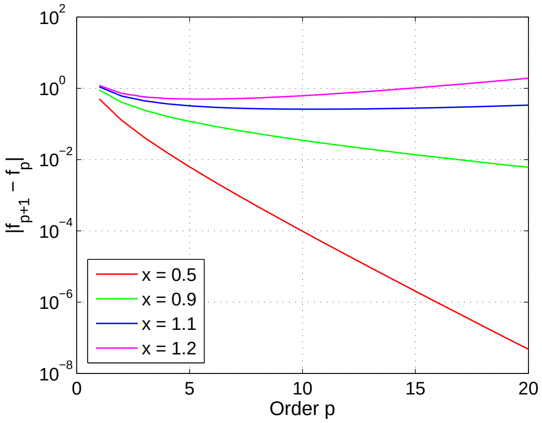


Figure 84: Discrepancy in divergent Taylor series estimates at various $\mathbf{x}$ locations for $\mathbf{f(x) = log(1 + x)}$.

3.8.3 Optimal Solution Prediction Using State Transition Tensors

The prior Taylor series examples illustrate two important considerations when predicting optimal solutions during the continuation process. First, the accuracy of predicted solutions is not guaranteed to increase as the order of the STT approximation increases. As such, higher order STT approximations should only be used to improve accuracy if the observed discrepancy decreases. Second, for well-posed hypersonic problems, optimal solutions can likely be predicted to any degree of accuracy using STTs of sufficient order as long as these solutions are sufficiently close to the reference solution. As predictions are made farther away from the reference solution, the accuracy of predictions with increasing order p is governed by the optimal solution space. To highlight the advantages of using higher order STTs, optimal trajectories from a continuation process are also predicted using both first and second order STTs as shown in Figure 85. During this process, the prior converged indirect solution serves as the reference trajectory when calculating the STTs. The errors in predicted altitude and velocity with the corresponding optimal solutions are shown in Figure 86. As desired, the predicted errors from the second order approximation are smaller throughout each trajectory than the errors from the first order approximation. Additionally, the accuracy of these predictions suggests that larger steps can be taken during the continuation process and that the error in predicted solutions can be controlled using higher order STT approximations.

As an initial step to automate the continuation process, a methodology has been developed to predict optimal solutions throughout the unconstrained trajectory space using STTs. By controlling the error of this prediction, the methodology dynamically determines the appropriate change in initial conditions originally chosen through trial and error as shown in Figure 74. Furthermore, these predicted solutions are used to provide better initial guesses that were originally formulated using the prior indirect solution. To predict optimal, unconstrained trajectory solutions, boundary conditions

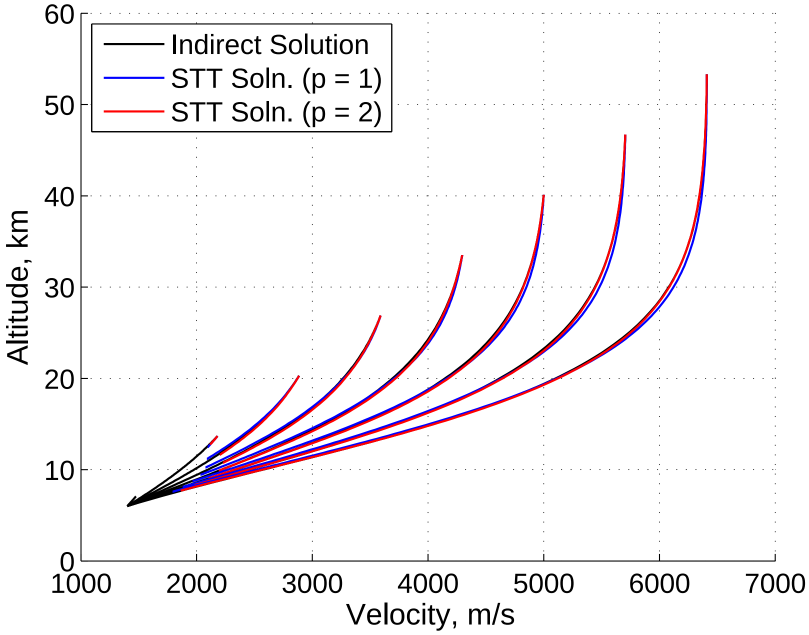


Figure 85: Improved STT-predicted trajectories with order of approximation.

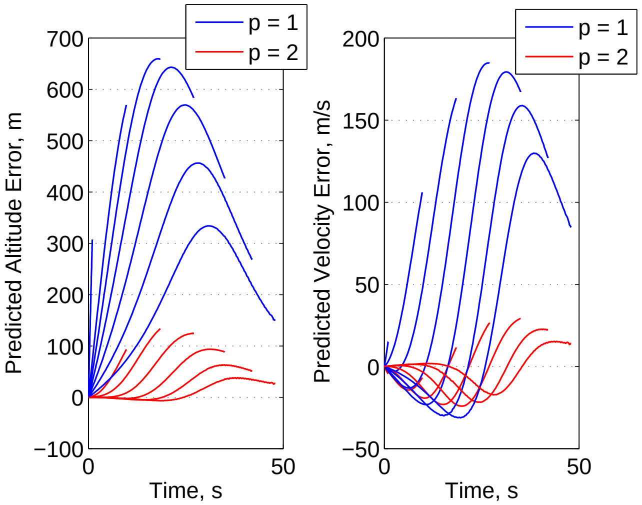


Figure 86: Error in predicted altitude and velocity for various orders of approximation.

must be satisfied at both the initial and terminal points of the trajectory. In general, the STTs provide an analytic mapping between these boundary conditions, enabling an analytic root-solving process to be used to quickly construct predicted optimal solutions. During the unconstrained continuation process, the initial altitude and velocity are incrementally increased to eventually match the desired initial conditions while the terminal conditions remain fixed. As a result, the time of each optimal trajectory increases during this process. However, the STTs are only able to relate changes in initial conditions to changes in terminal conditions with respect to the indirect solution that serves as the reference trajectory during computation of the STTs. This is evident in Figure 85 in which the predicted STT solutions terminate before reaching the desired terminal point. To predict optimal trajectories with longer flight times, the indirect solution used to construct the STTs must be propagated beyond the original terminal conditions by a time dt_f as shown in Figure 87. As such, STTs can then be used to predict the change in initial conditions, dx_i, that yields an optimal trajectory with this increased flight time to the desired terminal state.

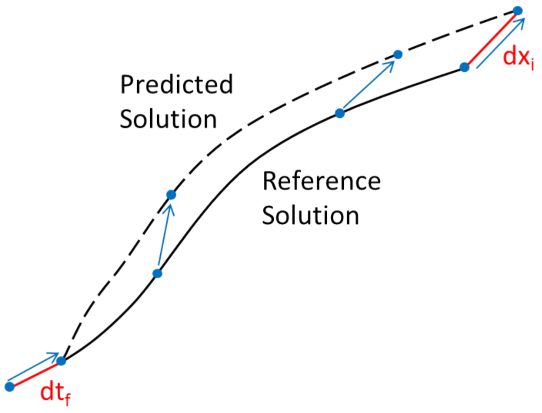


Figure 87: Optimal solution prediction of trajectories with varying initial conditions and fixed terminal conditions.

While this process enables the prediction of optimal trajectories with greater flight times, many solutions in dx_i likely exist for a given choice in dt_f. To enforce a unique solution for dx_i, all changes in the initial state are restricted to the analytic shooting line that connects the initial state of the indirect solution to the desired initial condition as shown in Figure 88. Additionally, as the magnitude of dx_i increases as dt_f increases, the STT-predicted solutions will reside at greater distances from the initial indirect solution. Consequently, the error associated with these predictions will also likely increase. Without knowledge of the exact optimal solutions during this process, the error at the boundary conditions is approximated as the discrepancy between STT-predicted solutions of order p and of order $p+1$. To limit the number of steps during the continuation process, the magnitude of dx_i should be chosen to be as large as possible without violating the maximum approximate error chosen by the designer. However, the proper choice in dt_f that provides this desired magnitude of dx_i is not known *a priori*. As such, an intermediate rapid continuation process is employed to predict optimal solutions along the analytic shooting line using STTs.

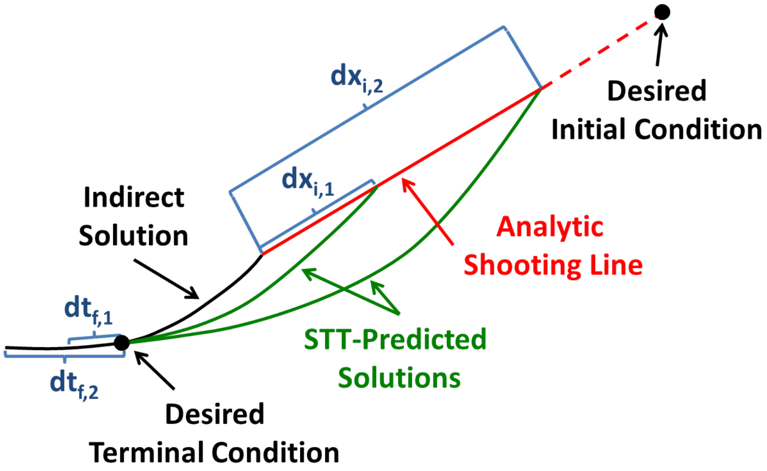


Figure 88: Example of unconstrained analytic shooting using STT predictions.

During this intermediate process, dt_f is incrementally increased, and the analytic mapping between the initial and final conditions of the STT-predicted solutions is used to converge to the appropriate dx_i along the analytic shooting line. During this process, the STTs provide the opportunity to perform an analytic shooting in which dx_i is iterated to converge to the appropriate terminal conditions. Once the proper change in initial conditions is found, the final time is incrementally increased again from $dt_{f,1}$ to $dt_{f,2}$, and a new initial condition, $dx_{i,2}$, is identified through analytic shooting of the STTs using $dx_{i,1}$ as an initial guess. This process is repeated until dt_f can no longer be increased without exceeding the limit in approximate error chosen by the designer. Note that the intermediate continuation process is naturally regulating. As the indirect solution is propagated beyond the terminal conditions using an error-controlled, adaptive RK scheme, the sequence of dt_f values used by the intermediate continuation process adapts to the dynamics of the optimization problem. If the dynamics of the problem are relatively strong, then smaller steps are taken in dt_f, resulting in more steps during the intermediate continuation process.

To maximize computational efficiency, the analytic shooting process is initially performed using an STT of order one, and the approximate error of each solution is calculated using an STT of order two. As the limit in approximate error is reached, the order of the STT used for each calculation can be increased as long as the discrepancy is decreased. If no higher order STT solution is available due to the increase in computational requirements, then the STT-predicted solution is used as an initial guess to converge to a new indirect solution, and the process is repeated. As an example, this process is executed using a maximum STT order of two shown in Figure 89 and a maximum order of three shown in Figure 90. As expected, as the maximum order of the STT increases, larger changes in dx_i can be predicted with the same limit in approximate error. Note that this error includes predicted flight times, and as a result, the STT-predicted solutions do not reach the terminal point. However, these predictions do provide greatly improved initial guesses that enable rapid convergence to the nearby indirect solutions also shown in Figures 89 and 90.

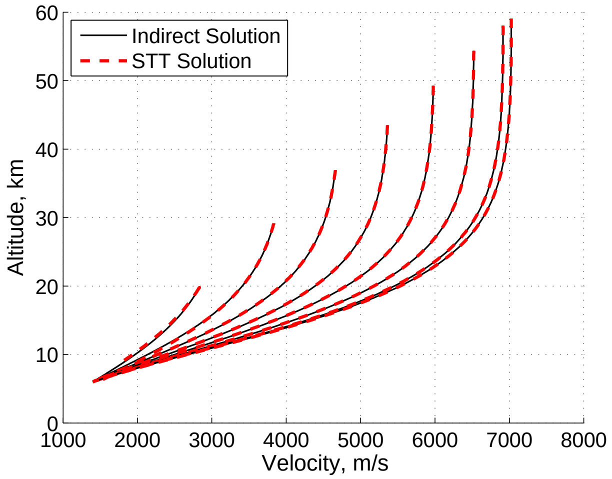


Figure 89: Unconstrained continuation using 2$^{\text{nd}}$ order STT predictions.

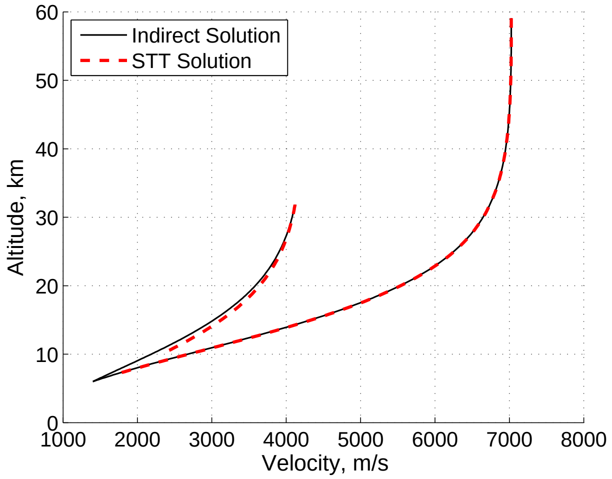


Figure 90: Unconstrained continuation using 3$^{\text{rd}}$ order STT predictions.

3.9 Addressing Prior Methodology Limitations

As shown, the complexities of the prior methodology that relies on the combination of many trajectory optimization techniques including discrete dynamic programming, nonlinear inversion, and pseudospectral methods are eliminated. For well-behaved trajectory optimization problems, this framework can be collapsed into a simplified methodology that only relies on the continuation of indirect methods. As a result, the path cost limitation and sensitivity of the discrete dynamic programming process are eliminated. Additionally, the complexities of pseudospectral methods associated with predetermined node locations, mesh refinement techniques, use of penalty functions in NLP solvers, and identification of discontinuous changes in costates are also eliminated. Alternatively, indirect methods enable the use of single and multiple shooting methods that employ an error-controlled, adaptive RK scheme. As a result, path constraints are directly enforced, eliminating the need for penalty functions, and discontinuous changes in costates that result from these path constraints can be directly observed during the continuation process.

The dynamic node arrangement provided by adaptive RK schemes ensures that the dynamics of the trajectory are satisfied to an accuracy specified by the designer. This eliminates the need for mesh refinement techniques that may be required when node locations are predetermined as shown in Figure 69. Additionally, the dynamic node arrangement efficiently clusters nodes in regions of most interest that have strong dynamics, whereas pseudospectral methods cluster nodes near the endpoints with generally weak dynamics for hypersonic missions. The efficient clustering of nodes by adaptive RK schemes could assist onboard real-time trajectory planning operations in which optimal trajectory solutions are only accepted if the dynamics at many locations throughout the entire solution are validated to a certain level of accuracy. Furthermore, the use of penalty functions, internal subproblems, and slack variables by NLP solvers greatly increase the complexity of identifying optimal solutions. As a result, the certification of these algorithms for flight operations would also increase in complexity. Alternatively, indirect methods only require a root-solver and a numerical integration scheme. Onboard calculations such as these have already been planned to support the entry monitor system of the Orion command module.[122] As such, the continuation of indirect methods could potentially bridge the gap between conceptual design and onboard trajectory operations.

3.10 Summary

In this chapter, an automated, robust trajectory optimization process has been developed to compute the necessary conditions of optimality and execute the required steps to obtain a converged indirect solution. This process combines and advances disparate trajectory optimization techniques developed over the previous century into a unified framework that is capable of solving a wide range of design problems. Specifically, this framework implements discrete dynamic programming, nonlinear inversion, pseudospectral methods, indirect methods, and continuation. This process is based on the perspective that trajectory designers are usually interested in directly constructing optimal trajectories. Additionally, a new entry interface based on the physics of the problem that is a function of velocity, vehicle shape, and celestial body was constructed to improve the efficiency of trajectory optimization when compared to previous and current studies. This framework enables rapid trajectory optimization using indirect methods and continuation, enabling (i) rapid trajectory optimization and design space exploration, (ii) rapid sensitivity and robustness analysis, and (iii) rapid vehicle requirements definition.

Examples demonstrate that families of optimal trajectories can be rapidly constructed for a wide range of vehicle, planet, and trajectory parameters. Each trajectory obtained is a fully converged indirect optimal solution in states, costates, control, initial conditions, terminal conditions, and corner conditions. In this manner, optimal trades in trajectory performance and vehicle requirements can be performed. Advancements in analytic hypersonic aerodynamics from Chapter 2 enabled the continuation process to expand beyond trajectory parameters to include vehicle shape. Convergence was also demonstrated when atmospheric properties and gravity were varied, validating the hypothesis that many optimal hypersonic trajectories are linked through indirect methods. For trajectory optimization problems that are highly sensitive, have many switches in control, or have solutions that are not continuous with trajectory parameters, this robust optimization process is likely required.

However, for well-behaved trajectory problems, only a subset of this process is necessary in which optimal trajectories can be identified through the continuation of indirect methods alone. This approach eliminates the complexities associated with the integration of discrete dynamic programming, nonlinear inversion, and pseudospectral methods required by the full trajectory optimization methodology before the continuation of indirect methods. These optimization techniques are required in the Section 3.3 methodology due to the challenges associated with constructing a good initial guess to converge to an indirect solution within a constrained design space. Alternatively, a good initial guess is shown to be easily constructed for a short, unconstrained, optimal trajectory that resides outside of the design space of interest. As such, solutions within the design space of interest are shown to be efficiently constructed through continuation of indirect methods from this unconstrained solution. During this simplified trajectory optimization process, path constraints are introduced at points of tangency with the trajectory, and examples with maximum heat rate and g-loading constraints illustrate that the indirect method is able to converge to optimal solutions as the magnitude of these constraints is modified. For these examples, the continuation process is performed manually. As an initial step toward the automation of the continuation process, state transition tensors are shown to provide accurate predictions of optimal solutions throughout the unconstrained trajectory space. A methodology is developed to perform accurate predictions of trajectories with varying flight times, and the error of these predictions is controlled to identify the required steps during the continuation process. Finally, it is noted that the use of state transition tensors and continuation of indirect methods could be used for both conceptual design studies and onboard real-time trajectory planning, and as a result, could serve as a means to bridge the gap between conceptual design and onboard operations.

CHAPTER IV

RAPID SIMULTANEOUS HYPERSONIC AERODYNAMIC AND TRAJECTORY OPTIMIZATION USING INDIRECT METHODS

4.1 Introduction

The advancements made in analytic hypersonic aerodynamics (Chapter 2) and rapid trajectory optimization (Chapter 3) could serve as a means to improve the overall segregated, traditional design environment described in Section 1.1 by individually reducing the computational requirements of each corresponding discipline. However, the analytic mapping between vehicle shape and trajectory performance made possible by the development of analytic hypersonic aerodynamic relations enables the DSM in Figure 1 of Section 1.1 to be collapsed into a single, unified, mathematical framework. In this new framework, the analytic hypersonic aerodynamic relations enable vehicle shape parameters to be analytically incorporated into the trajectory equations of motion. As a result, the capability of performing rapid trajectory optimization through the continuation of indirect optimization methods described in Chapter 3 is extended to also include vehicle shape. This unified framework is capable of simultaneously optimizing to objectives in both vehicle shape and trajectory while simultaneously accounting for constraints in both disciplines. As such, this capability provides the necessary means to perform rapid simultaneous hypersonic aerodynamic and trajectory optimization for conceptual design.

4.2 Multidisciplinary Optimization Methodologies for Comparison

4.2.1 Indirect Methods

4.2.1.1 Augmentation of the Rapid Trajectory Optimization Methodology

To optimize the vehicle simultaneously with the trajectory, the equations of motion shown in Eq. (64)-(67) in Section 3.2 must be augmented with the appropriate analytic aerodynamic relations.[112] For example, the analytic aerodynamics of a conical frustum can be parametrized by the cone half angle, d, as shown in Section 2.4.1. As a result, the aerodynamic forces in Eq. (66) and (67) can be expressed as a function of vehicle shape as shown in Eq. (77) and (78). As new shape parameters such as the cone half angle are introduced into the equations of motion, a new state can be added as shown in Eq. (79). For this study, shape parameters are assumed to be constant throughout the trajectory. If the shape of the vehicle can be altered during flight, for example to control inflatable aerodynamic decelerators during entry, then Eq. (79) can be modified accordingly. When applying indirect methods to this augmented system, a costate is introduced for each corresponding shape parameter. If the objective is only a function of the trajectory in the form of Eq. (9), then the indirect method can be extended to include the additional states and costates associated with each shape parameter. As a result, optimal solutions in both vehicle and trajectory are simultaneously constructed. Note that indirect methods can only perform a single objective optimization expressed through the cost functional, J. If vehicle objectives are also included, then the optimization must be modified according to the convexity of the Pareto frontier.

$$\dot{v} = -\frac{D(d)}{m} - \frac{\mu \sin\gamma}{r^2} \tag{77}$$

$$\dot{\gamma} = \frac{L(d)\cos\phi}{mv} + \left(\frac{v}{r} - \frac{\mu}{vr^2}\right)\cos\gamma \tag{78}$$

$$\dot{d} = 0 \tag{79}$$

4.2.1.2 Multi-Objective Formulation

If the Pareto frontier is convex, then an augmented objective, J', can be constructed through the combination of multiple objectives as shown in Eq. (80) for n objectives. The influence of each individual objective, J_i, can be controlled through the corresponding relative weighting, w_i. The Pareto frontier can be constructed by performing successive optimizations through continuation of these weightings. This approach minimizes the added complexity required to perform a multiobjective optimization but has been shown to be ineffective for non-convex frontiers.[123, 124, 125] Various approaches have been developed to address non-convex frontiers, including transformations to convex shapes and the addition of constraints. For this work, non-convex frontiers are addressed by performing a single objective optimization in J_i while constraining the remaining objectives to specific values. The Pareto frontier is then constructed through continuation of the constrained objective values. It is important to note that this approach will also work for convex Pareto frontiers. However, the added complexity that results from the addition of constraints can reduce the efficiency of the continuation process for convex frontiers. In general, the convexity of the Pareto frontier will not be known *a priori*. As such, an augmented objective can be used to identify the convex portions of the frontier. If any gaps are observed in the Pareto frontier, then the continuation of constrained objectives can be used to identify these non-convex regions.

$$J' = \frac{1}{\displaystyle\sum_{i=0}^{n} w_i} \cdot \sum_{i=0}^{n} w_i J_i \tag{80}$$

4.2.2 Baseline Multidisciplinary Optimization Methodology

To illustrate the advantages of indirect methods for design, comparisons are made to a baseline multidisciplinary optimization methodology. While numerous MDO algorithms and techniques exist, many recent entry design studies have used population-based global searching algorithms such as genetic algorithms and particle swarm optimizers.[29, 60, 61, 126] As such, a particle swarm optimizer (PSO) was chosen to perform these comparisons. Parameters associated with the PSO were chosen based on prior experience.[60] Note that the choice in these parameters can dramatically influence the efficiency of the optimization process, but identifying an optimal set of parameters *a priori* is not practical.[127] Due to the stochastic nature of the algorithm, a sufficient number of iterations is chosen based on available computational resources or when marginal improvement of the optimal solution is observed. In general, the optimizer required a population size of 100 particles that searched the design space for approximately 300 iterations.

The PSO solves for optimal solutions through direct shooting of the equations of motion in which free initial states, a discretized bank profile, and vehicle parameters form the design space.[60, 61] In this study, ten discrete bank angles are chosen at equidistant velocity increments throughout the entry, and the bank angle at any point along the trajectory is calculated by linear interpolation of these angles. Note that the location of the ten bank angles with respect to velocity could be improved by clustering more values in high dynamic pressure regions. While the use of error-controlled, adaptive RK schemes by indirect methods automatically performs this clustering as described in Section 3.6.3, this process would further increase the complexity of the traditional PSO analysis, and as a result, was omitted from the following comparisons. Each PSO-derived trajectory is propagated from the initial state until the vehicle reaches either the desired terminal velocity or the ground at 0 km altitude. Although many traditional MDO methods employ panel methods to characterize the hypersonic aerodynamics of a vehicle, the analytic aerodynamics are used as a substitute to make reasonable comparisons with indirect methods. To highlight some of the challenges associated with using traditional MDO methods to perform simultaneous hypersonic aerodynamic and trajectory optimization, comparisons are made to solutions obtained from indirect methods.

4.3 Comparison of Indirect Methods and Baseline Multidisciplinary Optimization Methodology

For the following comparisons, an Earth-based entry is again chosen with environment parameters shown in Table 5 of Section 3.2. To improve execution speed by approximately a factor of 30, the required functions to solve for the indirect solutions and perform trajectory propagations for the PSO are autocoded into C from Matlab. In all comparisons, the initial and terminal conditions in altitude and velocity are assumed to be fixed, whereas the initial and terminal conditions in flight path angle and downrange are assumed to be free. To highlight a range of shapes that can be included in the optimization, the following examples consist of a blunted cone, blended wedge, and blunted biconic. Note that for the following comparisons, only the analytic aerodynamics derived from traditional Newtonian calculations are used. If compact solutions obtained using the Divergence Theorem are also included, then the performance gap between PSO methods and indirect methods would widen. The mass for each vehicle was chosen to provide a reasonable ballistic coefficient to provide a wide range of solutions. As such, entry masses of 4100 kg for the blunted cone, 16,300 kg for the blended wedge, and 410 kg for the blunted biconic are chosen. For each vehicle, the mass is assumed to be constant throughout the flight. As a result, the ballistic coefficient of each vehicle is varied as the shape is modified during the optimization process. Finally, the angle of attack of the blunted cone and blended wedge was chosen to be 20^o, and the angle of attack of the blunted biconic was chosen to be 10^o.

4.3.1 Minimum Heat Load for a Blunted Cone Subject to Terminal Constraints

As an initial comparison of the two methodologies, the trajectory and geometry of a blunted cone are simultaneously optimized to minimize stagnation point heat load resulting from minor blunting of the nose. This trajectory objective was chosen to enable comparisons with examples from Chapter 3.[113, 128] To minimize heat load, the heat rate must be maximized along every portion of the trajectory, and this result is evident from the optimal solutions presented. Additionally in Chapter 3, it has been shown that unconstrained optimal trajectories can be quickly constructed through continuation of indirect solutions from a short, unconstrained trajectory that is relatively easy to optimize.[113] This approach is repeated on the augmented system in which the vehicle is also optimized during the continuation process. During the optimization, the analytic aerodynamics of the blunted cone is altered through modification of the cone half angle, d. In this work, the short, unconstrained trajectory is chosen to have the desired terminal conditions in altitude and velocity. As such, a continuation is chosen to modify the initial conditions in altitude and velocity to match the desired entry conditions. During this process, the initial velocity is originally increased while maintaining a fixed initial altitude. After this continuation in initial velocity, the initial altitude is increased to match the desired initial conditions as shown in Figure 91. Additionally, Figure 92 shows the evolution of the corresponding costates. During this series of optimizations, the initial and terminal conditions in flight path angle and downrange are allowed to vary. As expected, λ_γ and λ_θ are zero at the initial and terminal points of the trajectory. Furthermore, λ_θ is identically zero throughout the trajectory as expected by the absence of θ in the equations of motion.

During the continuation process, the cone half angle, d, is also optimized to provide minimum heat load trajectories. As expected, each solution consists of an optimal

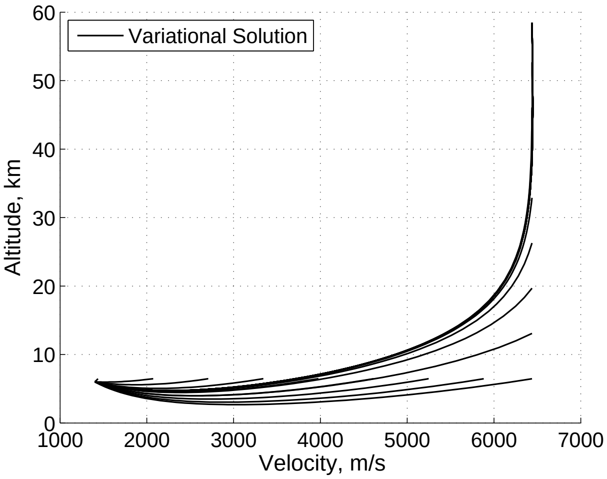


Figure 91: Continuation of minimum heat load solutions.

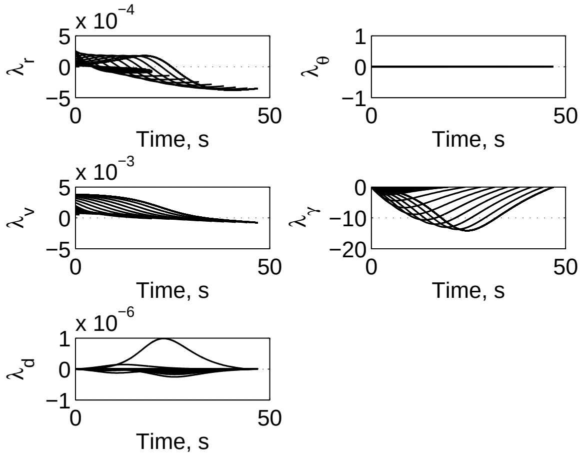


Figure 92: Corresponding minimum heat load costates.

cone angle of 90^o. This forms the most blunt conic vehicle in the shape of a circular disk. This shape minimizes heat load by minimizing the ballistic coefficient of the vehicle, and, consequently, the flight time of the trajectory. Additionally, the flight path angle costate is negative throughout the trajectory, resulting in a constant commanded bank angle of 180^o as expected to fly full lift-up for a blunt body as described by the switching structure shown in Table 9. Note that this bank is opposite to convention for blunt bodies and is a consequence of the presence both slender and blunt bodies during optimization. In this work, the bank angle convention is chosen to be consistent with slender bodies. The resulting bank angle of 180^o allows the blunt vehicle to fly a steep entry that further reduces flight time. These expected minimum heat load solutions that are consistent with prior studies validate the simultaneous hypersonic aerodynamic and trajectory optimization framework.[113, 128]

Table 9: Control switching structure.

$C_L \cdot \lambda_\gamma < 0$	Bank = 0 deg
$C_L \cdot \lambda_\gamma > 0$	Bank = 180 deg
$C_L \cdot \lambda_\gamma = 0$	Bank is indeterminate

For comparison, a single objective PSO is also used to perform the same hypersonic aerodynamic and trajectory optimization from the desired initial conditions. Indirect methods convert the optimization problem into a root-solving problem, and, as a result, trajectory constraints are easily enforced along the optimal solution. Alternatively, trajectory constraints are indirectly enforced by the PSO through modification of the bank angle profile. As a result, penalty functions are generally constructed to enforce trajectory constraints. The optimization is then performed on the augmented objective function that includes these penalty functions rather than the original objective function of interest. After selection of the penalty function, the designer must choose an appropriate balance between the original objective and this penalty when constructing the augmented objective function.

If the penalty associated with the terminal constraint is too strong, then the trajectory will be optimized to the constraint. As a result, the trajectory objective will be largely ignored as shown in Figure 93 in which the heat load is multiplied by the magnitude of the final altitude error. While the optimal PSO solution consists of a cone half angle of 89.3^o, the heat load is 2.26 times greater than that of the indirect solution. To construct minimum heat load solutions, the penalty must be relaxed. This can be done by constructing a sphere around the terminal constraints in which any terminal points inside this sphere are not penalized, and any points outside of this sphere are strongly penalized in the same manner as the prior example. Since the PSO trajectories are terminated at the desired final velocity, a 3000 m altitude penalty buffer was constructed such that only trajectories that have a terminal altitude error greater than 3000 m are penalized. The resulting optimal PSO solution is shown in Figure 94. As expected, the PSO optimizes to the augmented objective function, and the resulting minimum heat load trajectory terminates at the lowest altitude within the penalty buffer.

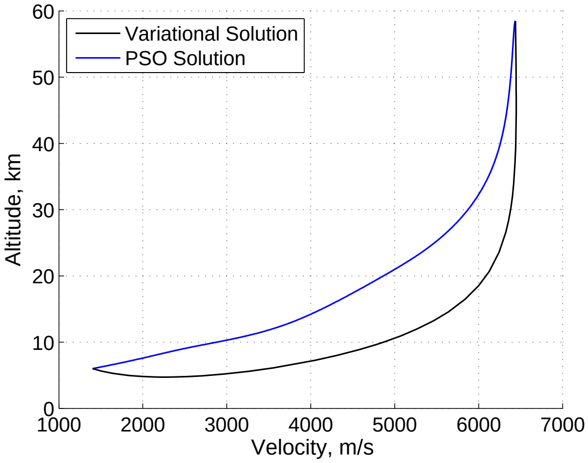


Figure 93: Comparison between indirect method and PSO with strong terminal conditions penalty.

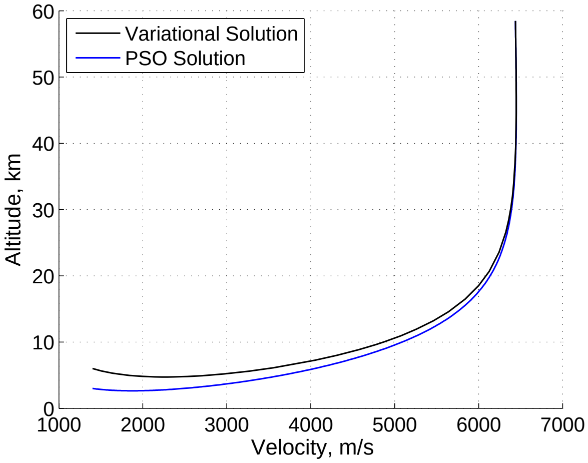


Figure 94: Comparison between indirect method and PSO with 3000 m penalty buffer.

The heat load of this trajectory is only 93% of the heat load associated with the indirect solution, and this is accomplished by eliminating the additional heat load incurred when the final altitude is increased. In order to guide the terminal altitude to the desired value, the altitude penalty buffer can be reduced. Figure 95 shows the optimal PSO trajectory when the altitude penalty buffer is reduced to 200 m. As expected, the PSO optimizes to the augmented objective function, and the terminal altitude resides at the lowest altitude within the penalty buffer as shown in Figure 96. By placing a greater emphasis on the terminal conditions, the heat load for this trajectory is 1.09 times greater than that of the indirect solution. Furthermore, the time required to construct the PSO solutions is greater than the time required to construct the indirect solutions by approximately a factor of eight to ten. This performance gap would also likely be widened if multiple shooting techniques are used to identify the indirect solutions.[129] These prior examples illustrate the inability of direct optimization methods to perform the precise optimization desired when used in this manner. As a result, the traditional, segregated design environments that depend on this approach also suffer from these inefficiencies.

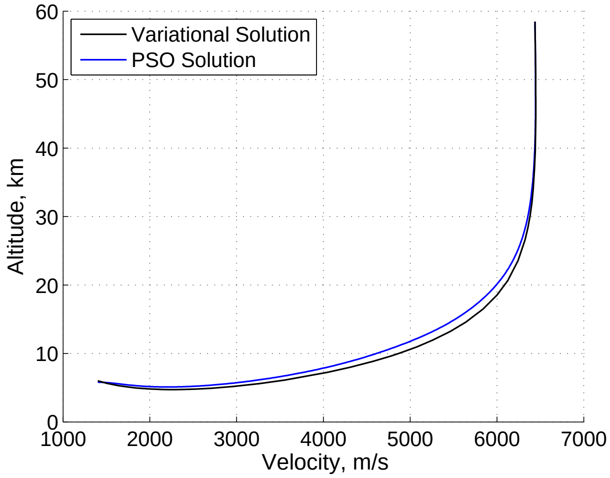


Figure 95: Comparison between indirect method and PSO with 200 m penalty buffer.

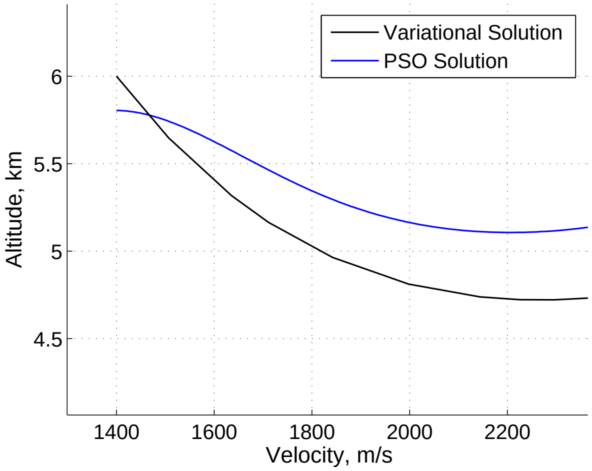


Figure 96: Comparison of terminal trajectory segment with 200 m penalty buffer.

To verify the optimization process, the design space is dramatically reduced to determine if the PSO could arrive to the same solution as that of the indirect method. As such, the cone half angle was restricted to $80^o \leq d \leq 90^o$, the initial flight path angle was restricted to $-30^o \leq \gamma_o \leq -10^o$, and the bank angle was restricted to $170^o \leq \phi \leq 180^o$. These ranges were constructed to encompass the indirect solution, and the resulting optimal PSO trajectory is shown in Figure 97. As expected, the PSO solution matches the indirect solution with a heat load that is only 1% greater than that of the indirect solution. This illustrates that the PSO is capable of achieving nearly the same results as the indirect method in the prior examples if additional computational resources are provided. Due to the relatively substantial computational resources already required by the PSO, this increase would only further contribute to the computational inefficiency of the approach. As shown, the balance between penalty functions and the original objective adds complexity to the optimization process. While the PSO allows a global search of the design space, the resulting optimal solutions will likely capitalize on the designer's choice of penalty functions. To address this issue, a multi-objective optimization process is constructed that is more robust to the choice in penalty functions.

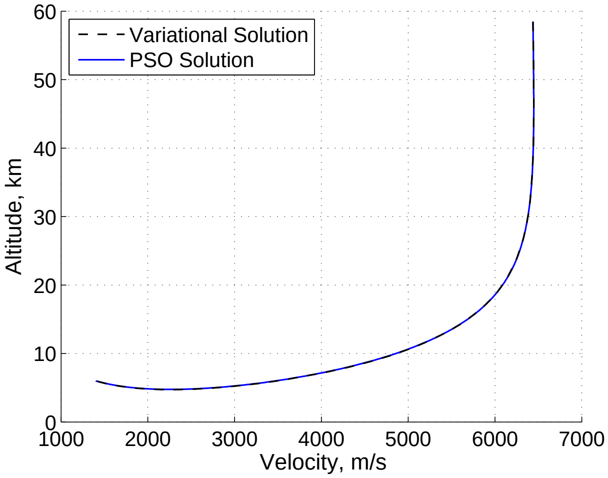


Figure 97: Comparison between indirect method and PSO with reduced design space.

To capture optimal solutions that have terminal altitudes closer to the desired value, a second objective in final altitude error is added. As such, a multi-objective particle swarm optimizer (MOPSO) is used to construct the optimal trade, or Pareto frontier, in these two objectives as shown in Figure 98. In this example, the terminal altitude buffer is expanded to 6000 m. During the optimization process, MOPSO attempts to construct the global Pareto frontier with well-spaced solutions that expand to both ends of the frontier. As expected, the Pareto frontier consists of altitude errors between 0 m and 6000 m. For comparison, a Pareto frontier is also constructed from continuation of indirect methods in which the terminal altitude is varied. This process only requires about one-tenth of the computational time when compared to MOPSO. As shown, a well-defined Pareto frontier is constructed using indirect methods, and this frontier dominates the MOPSO Pareto frontier. Additionally, the increase in heat load associated with the increase in terminal altitude further validates the single objective PSO trajectories that terminate at the lowest altitude within the penalty buffer. The corresponding MOPSO trajectories are shown in Figure 99 along with a few indirect solutions that span the 6000 m altitude penalty buffer. In general, the MOPSO trajectories are consistent with the indirect solutions except for a few high-altitude MOPSO trajectories.

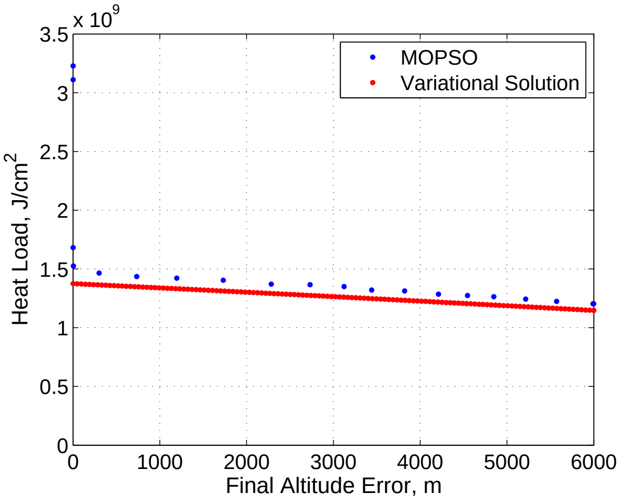


Figure 98: Comparison between indirect method and MOPSO frontiers with 6000 m penalty buffer.

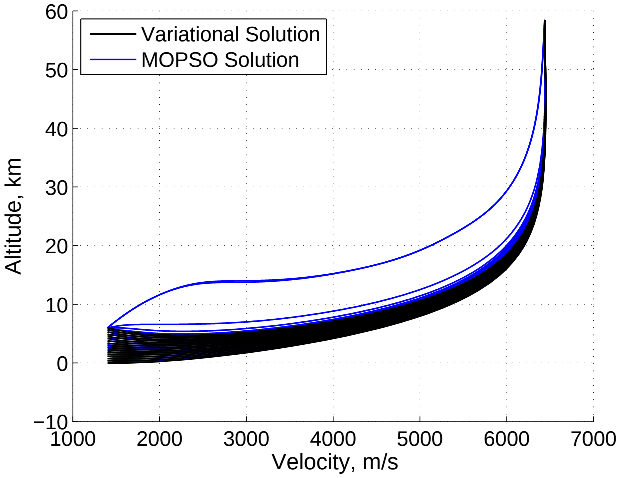


Figure 99: Comparison between indirect method and MOPSO trajectories with 6000 m penalty buffer.

These high-altitude trajectories illustrate the difficulty of precisely targeting the desired final altitude when using MOPSO, and these trajectories correspond to the high heat load solutions that have a near zero terminal altitude error shown in Figure 98. While the MOPSO Pareto frontier provides a comprehensive set of optimal solutions, this trade is of little interest to the designer if a terminal altitude error near zero is required. As such, the designer is likely only interested in the design point with a small terminal altitude error near the bend in the frontier. Furthermore, this single design point is obtained by increasing the complexity of the design problem through the addition of a second objective. These examples illustrate the challenges associated with using direct, population-based optimization algorithms commonly used in hypersonic design studies as well as validate the simultaneous vehicle and trajectory solutions obtained through continuation of indirect methods. While the optimal blunted cones minimize heat load, the disk-like shape resulting from a cone half-angle of 90^o is not practical for payload packaging. As such, the trajectory objective must be expanded to include vehicle objectives, such as usable payload volume.

4.3.2 Minimum Heat Load and Maximum Usable Volume for a Blended Wedge Subject to Terminal Constraints

To illustrate the diversity of entry vehicle shapes that can be rapidly optimized using the analytic aerodynamic relations, a blended wedge is constructed by combining the aerodynamics of a cylindrical nose, two flat plates, and two half-cones.[112, 32] An example cross-section of a blended wedge with a 20^o half-angle is shown in Figure 100. Tangency is enforced among all components of the blended wedge, and the bluntness of the vehicle can be varied through the wedge half angle, d. In this example, the minimum heat load trajectory objective is expanded to also include usable volume in the form shown in Eq. (80) of Section 4.2.1.2, and the Pareto frontier is constructed through continuation of w as shown in Eq. (81). The continuation process begins with $w = 0$ such that the vehicle and trajectory are simultaneously designed to minimize heat load. During the continuation process, w is incrementally increased to place greater emphasis on usable volume at the penalty of increased heat load. The usable volume of the blended wedge is assumed to reside between the two square-shaped plates that form the upper and lower surfaces of the vehicle. This process required about one-tenth of the computational time when compared to MOPSO which was also used to construct Pareto frontiers of minimum heat load and maximum usable volume.

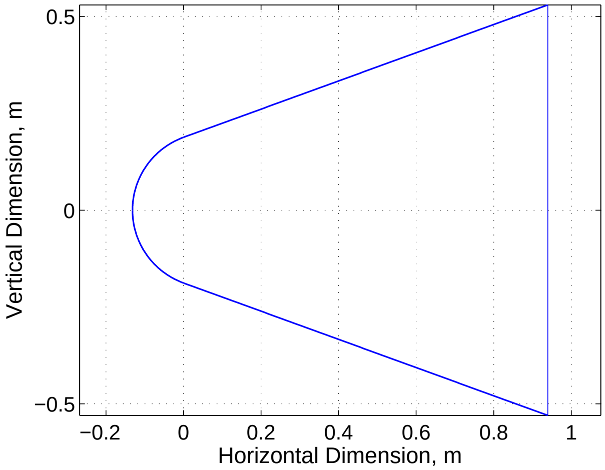


Figure 100: Example cross-section of blended wedge with a 20^o half-angle.

$$J' = w(-V_{\text{usable}}) + (1-w)\int_{t_0}^{t_f} \dot{q}dt \tag{81}$$

Initially, the design space is chosen to be as large as possible with the wedge half angle restricted to $12^o \leq d \leq 90^o$, the initial flight path angle between $-90^o \leq \gamma_o \leq 0^o$, and the bank angle between $0^o \leq \phi \leq 180^o$. A wedge half angle of 12^o is chosen as a lower bound to provide a reasonable amount of vehicle bluntness based on an assumed set of dimensions. In general, this design space allows MOPSO to explore a wide range of design options when constructing the frontier. Additionally, a third objective in terminal altitude error was included in MOPSO to allow a range of solutions to be constructed within a 200 m terminal altitude buffer. Figure 101 shows the Pareto frontiers constructed using both indirect methods and MOPSO, where three times as many iterations were used by MOPSO than in the prior example. This equates to approximately 25-30 times the computational requirements of the indirect method. As shown in Figure 101, many solutions on the MOPSO frontier are heavily dominated by solutions from the indirect method. This illustrates a lack of convergence of the MOPSO frontier, even when substantial resources are used. The corresponding trajectories are shown in Figure 102 in which several extreme MOPSO solutions are observed. For example, the MOPSO Pareto frontier points with a usable volume of approximately 0.35-0.45 m^3 that are near the indirect solutions correspond to relatively slender vehicles with steep entry flight path angles that terminate at the ground at a velocity of approximately 6500 m/s. These solutions illustrate that the penalty in heat rate for trajectories that terminate outside of the 200 m terminal altitude buffer was not sufficiently strong. Alternatively, the large skipping trajectories illustrate that solutions with accurate terminal conditions remain on the Pareto frontier at a large expense in heat load. These solutions are the result of penalty conditions that are too strong. This example also illustrates the challenges of using traditional design environments in which the manner that segregated disciplines are coupled may not be appropriate for all solutions of interest.

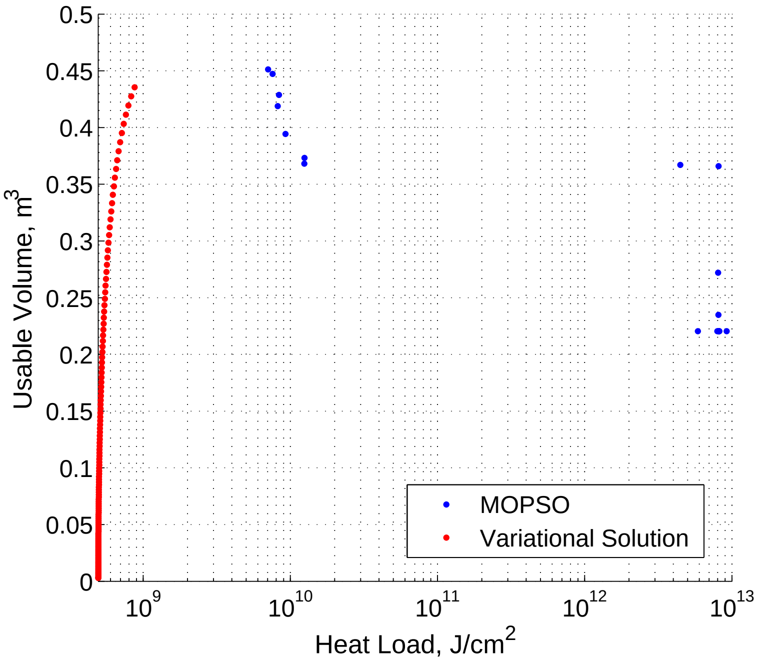


Figure 101: Poor Pareto frontier resulting from large MOPSO design space.

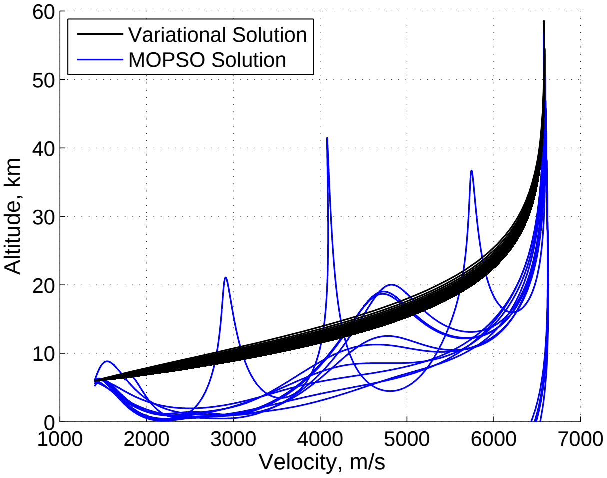


Figure 102: Corresponding trajectory solutions resulting from large MOPSO design space.

To assist MOPSO, the design space was reduced to encompass the Pareto optimal solutions obtained from the indirect methods. The smaller design space consists of a wedge half angle restricted to $40^o \leq d \leq 90^o$, an initial flight path angle between $-60^o \leq \gamma_o \leq 0^o$, and bank angle between $0^o \leq \phi \leq 180^o$. Using the same number of iterations as before, the new MOPSO Pareto frontier is constructed as shown in Figure 103. While the Pareto frontier is dramatically improved, the MOPSO solutions are still dominated by the indirect solutions. Additionally, the indirect Pareto frontier is more expansive in usable volume than the MOPSO frontier. Note that in both MOPSO frontiers shown in Figures 101 and 103, certain MOPSO solutions appear to be dominated by other MOPSO solutions. This is the result of the third objective in terminal error used to span solutions throughout the 200 m terminal altitude buffer. While this objective is not shown, all MOPSO solutions were verified to reside in this buffer. The corresponding trajectories are shown in Figure 104. While these trajectories have a greater resemblance to the indirect solutions, it is clear that certain lofted trajectories with accurate terminal conditions reside on the Pareto frontier at the expense of increased heat loads.

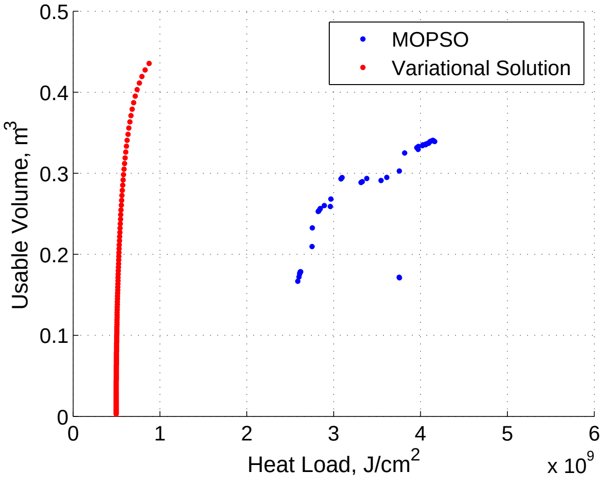


Figure 103: 2-D view of Pareto frontier in volume vs. heat load for blended wedge.

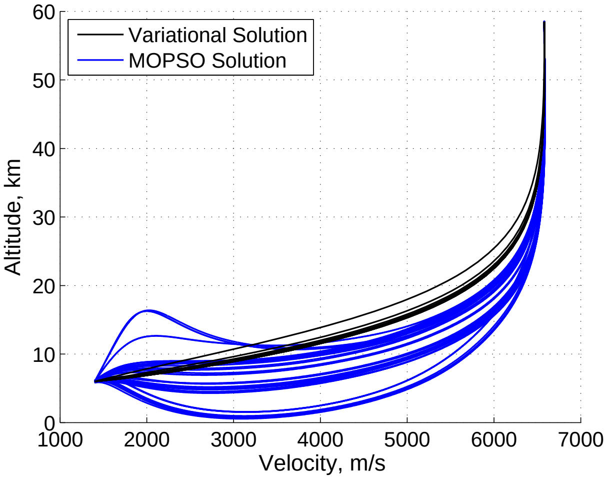


Figure 104: Corresponding trajectories from Pareto frontier.

The evolution of the cross-section of the blended wedge upper surface (note the full cross-section example shown in Figure 100) across the indirect Pareto frontier is shown in Figure 105. Analogous to the disk shape of the minimum heat load blunted cone, the minimum heat load blended wedge has the lowest ballistic coefficient. At an angle of attack, this corresponds to a wedge half angle near 90^o, resulting in a geometry with no usable volume. Through continuation of the relative weighting between heat load and usable volume, greater usable volumes are achieved at the expense of higher heat loads that result from increased ballistic coefficients. The cross-section of the nose region shown in Figure 106 illustrates the tangency that is enforced between the flat upper surface and cylindrical nose. As the wedge half angle is decreased, greater portions of the cylindrical nose are exposed, and the altered aerodynamics that result are automatically captured in the analytic aerodynamic relations. As optimal trajectories and vehicle shapes are constructed during the continuation process, these solutions can be monitored to identify limits in vehicle capability. In this example, if the vehicle is made more slender than what is shown, then the vehicle is unable to reach the terminal constraint without flying lofted and eventually flying a skipping trajectory that was chosen to be avoided in this work. This monitoring allows the designer to quickly guide the continuation process to solutions that are of most interest. Note that by using the weighted sum approach from Section 4.2.1.2, the Pareto frontier will only be fully constructed through continuation if the frontier is convex. In both the blunted cone and blended wedge examples, the Pareto frontiers are convex. However, if a more challenging, constrained trajectory is required, then convexity may not be guaranteed. In the following biconic example, the Pareto frontier is non-convex, requiring continuation of a constrained objective.

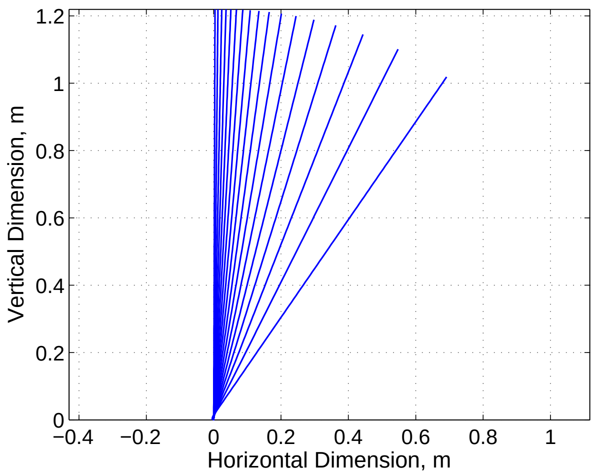


Figure 105: Evolution of blended wedge upper surface cross-section from Pareto frontier.

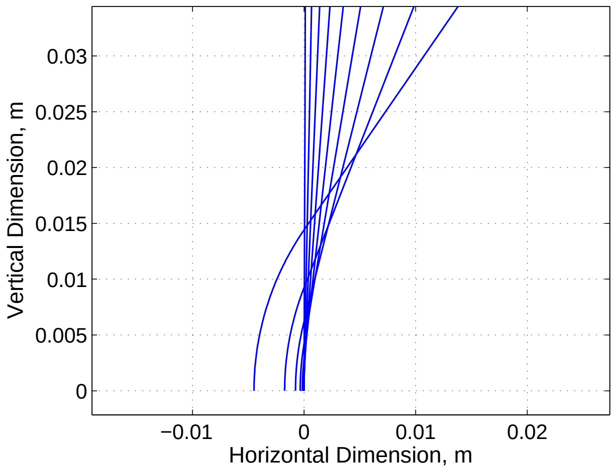


Figure 106: Evolution of blended wedge nose region cross-section from Pareto frontier.

4.3.3 Minimum Heat Load and Minimum Required Volume for a Blunted Biconic Subject to Path and Terminal Constraints

The prior example illustrates the ability to construct Pareto frontiers using indirect methods when vehicle objectives are included in the optimization. While the prior example includes terminal constraints, it does not include common path constraints such as maximum heat rate and g-loading. In the following example, a maximum heat rate constraint of approximately 2.0e7 W/cm^2 and a maximum g-loading constraint of approximately 24 Earth-g are chosen. These constraints limit low altitude and high velocity combinations, and Section 3.7.3 has shown that optimal trajectories can be constructed for a fixed vehicle by incrementally introducing these path constraints.[128] This initial continuation process was repeated for a fixed biconic with minor nose blunting, and this solution serves as a starting point when constructing the Pareto frontier using indirect methods. For this example, the vehicle objective is chosen to minimize the required volume of the two conical frustums. This objective may result from packaging considerations for storage in a carrier vehicle or launch shroud. Due to concerns of the convexity of the frontier, the required volume is enforced as a constraint, allowing indirect methods to be used to optimize the remaining trajectory objective in heat load. The Pareto frontier is then constructed through continuation of the constrained required volume. In order to eliminate the possibility of multiple biconic configurations with the same volume, only the forward conic half angle is allowed to vary to satisfy the volume constraint. Note that this process could also be repeated for the aft conic half angle should these solutions be of interest to the designer.

For comparison, MOPSO was also used to construct the Pareto frontier in which a third objective in terminal altitude error is again introduced. Initially, MOPSO is executed for 300 iterations. Trajectories that violated the heat rate and g-loading constraints were severely penalized to prevent these solutions from appearing in the Pareto frontier. The resulting Pareto frontiers from both MOPSO and the indirect method are shown in Figure 107, and the corresponding trajectories are shown in Figure 108. Consistent with the prior examples, the solutions from MOPSO are dominated by solutions constructed from the indirect method. While the indirect trajectories are tightly grouped together early in the entry, these trajectories form a large band in the second half of the entry. This band is due to the constant g-loading constraint enforced for all optimal solutions. As the forward cone angle is increased to satisfy an increasing required volume constraint during the continuation process, the ballistic coefficient of the vehicle is reduced. As a result, the vehicle must travel at higher altitudes to satisfy the g-loading constraint. While a reduction in ballistic coefficient generally results in shorter flight times that reduce heat load, the higher altitudes that must be flown in the presence of a g-loading constraint increase flight times that also increase heat load. This results in a slightly non-convex Pareto frontier as shown by the indirect solutions in Figure 107. Note that the MOPSO solutions that appear to be dominated by other MOPSO solutions are the result of the third objective in terminal error used to span solutions throughout the 200 m terminal altitude buffer. As such, solutions with much higher heat loads have much smaller terminal errors, and some of the MOPSO solutions close to the indirect frontier capitalize on the penalty-free 200 m altitude buffer.

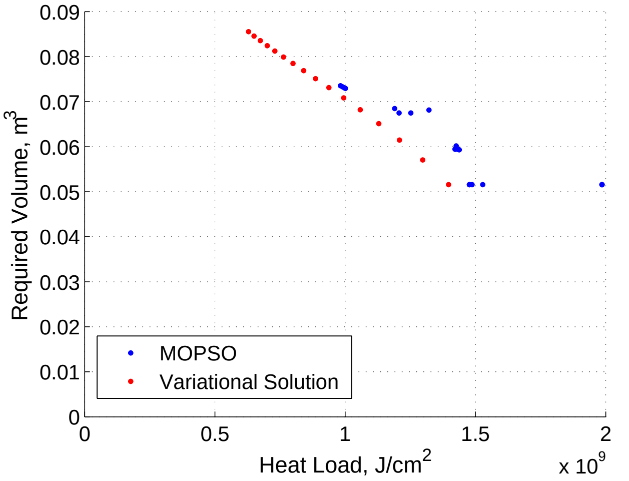


Figure 107: 2-D view of Pareto frontier in required volume vs. heat load for blunted biconic (300 iterations).

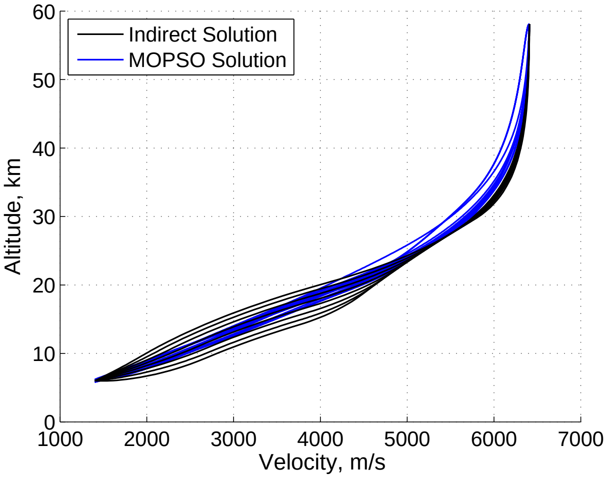


Figure 108: Corresponding trajectories from Pareto frontier (300 iterations).

The MOPSO Pareto frontier shown in Figure 107 has solutions that are grouped together, resulting in a poorly defined frontier. As such, MOPSO is executed for 900 iterations to improve the structure of the frontier, requiring approximately 25-30 times the computational requirements of indirect methods. As shown in Figure

109, the new MOPSO Pareto frontier is better distributed with many solutions near the indirect frontier. Additionally, the corresponding MOPSO trajectories shown in Figure 110 also have an increased distribution where the g-loading constraint is active along the second half of the trajectory. However, a select number of high-altitude trajectories are introduced. These trajectories correspond to slender biconic solutions that have small terminal altitude errors. This result is largely due to the difficulty of constructing path-constrained optimal trajectories using direct shooting methods often employed in traditional hypersonic design environments. To follow the g-loading constraint, the vehicle must dive further into the atmosphere near a velocity of 4500 m/s. Complex maneuvers such as these are difficult to construct when using direct shooting methods. The evolution of the blunted biconic shape is shown for three optimal solutions in Figure 111. Note that as the forward cone angle is increased, greater portions of the aft conic are exposed, and the resulting change in aerodynamics is automatically captured through the use of analytic hypersonic aerodynamic relations.

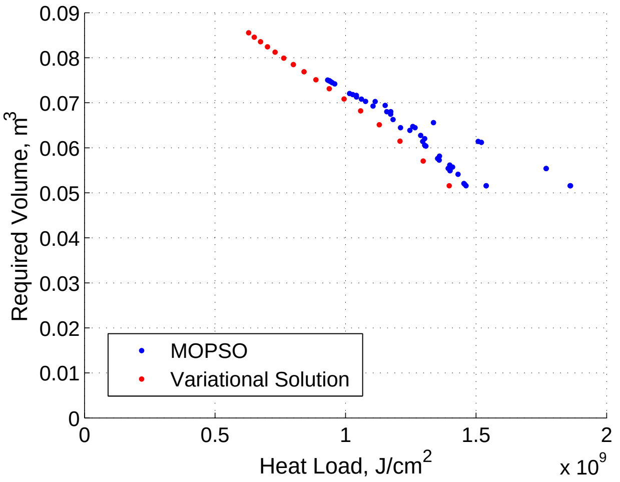


Figure 109: 2-D view of Pareto frontier in required volume vs. heat load for blunted biconic (900 iterations).

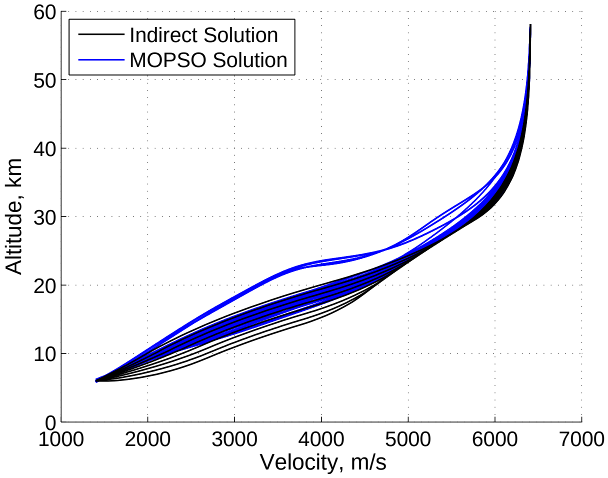


Figure 110: Corresponding trajectories from Pareto frontier (900 iterations).

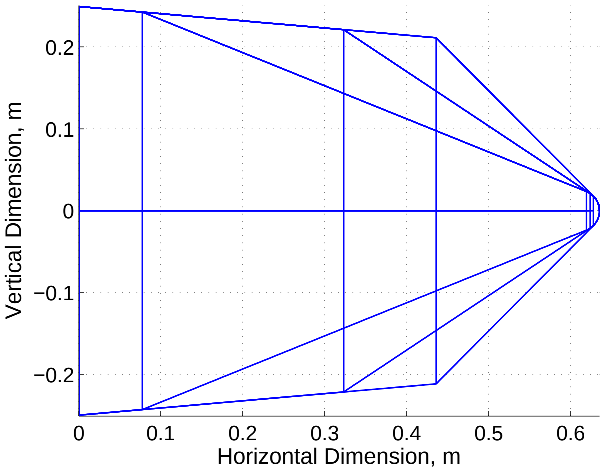


Figure 111: Samples from evolution of blunted biconic geometry.

4.4 Summary

In this chapter, a rapid simultaneous hypersonic aerodynamic and trajectory optimization methodology is constructed based on indirect methods. This design framework is made possible from enabling advancements in analytic hypersonic aerodynamics and rapid trajectory optimization that relies on the continuation of indirect methods. By extending this rapid trajectory optimization methodology to include the analytic aerodynamic relations, an augmented system is constructed that includes both trajectory and vehicle shape parameters. As a basis for comparison, a single and multi-objective particle swarm optimizer is used as a representative state-of-the-art multidisciplinary optimization methodology that relies on direct shooting for trajectory optimization. Comparisons illustrate that improved solutions can be obtained through continuation of indirect methods. Furthermore, indirect methods are shown to be effective across a range of vehicle shapes that include blunted cones, blended wedges, and blunted biconics for single and multiple design objectives in both trajectory and vehicle shape. For the examples shown, the computational requirements for the particle swarm optimizer are approximately 10-30 times greater than that of the indirect methods.

Examples using penalty functions also demonstrate the challenges associated with traditional, segregated design environments in which the manner that segregated disciplines are coupled may not be appropriate for all solutions of interest. These examples also illustrate the inability of common direct optimization methods to perform the precise optimization desired. Alternatively, indirect methods are shown to be able to perform the constrained optimization across a wide range of designs by precisely satisfying constraints. For convex Pareto frontiers, this optimization is most efficiently performed through continuation of a weighted sum of the individual objectives. Finally, a non-convex frontier is shown to be easily constructed through continuation of a constrained objective.

CHAPTER V

SUMMARY AND FUTURE WORK

5.1 *Summary of Contributions*

The body of work presented in Chapters 2-4 advances the state of the art in the conceptual design of hypersonic vehicles with additional application to real-time trajectory planning. This is accomplished by providing a capability to rapidly identify optimal trajectories and vehicle configurations necessary to accomplish various mission objectives and satisfy a variety of mission requirements. These advancements signal a departure from traditional, segregated design environments that are composed of individually-designed disciplinary analyses. While this traditional MDO approach is likely necessary to capture the most recent disciplinary advancements for use in high-fidelity design, it restricts the integration of the fundamental principles used during conceptual design. Since each analysis is developed individually, they are generally designed to be robust to a wide range of possible inputs. As a result, traditional design methodologies are generally unable to capitalize on the fundamental connectedness of solutions. To address these limitations, the state-of-the-art in conceptual hypersonic aerodynamic analysis and trajectory optimization was advanced in this dissertation. Specifically, the development of analytic hypersonic aerodynamic relations (Chapter 2) and a rapid trajectory optimization methodology (Chapter 3) enabled the construction of a unified mathematical design framework to perform rapid simultaneous hypersonic aerodynamic and trajectory optimization for conceptual design (Chapter 4). The following summarizes each of these three contributions.

5.1.1 The Departure from Numerical Hypersonic Aerodynamic Modeling to Analytic Methods

Prior to this dissertation, hypersonic aerodynamic modeling has been typically performed numerically during conceptual design using panel methods based on Newtonian flow theory. While implementation of these methods is straightforward, the construction of large aerodynamic tables is computationally intensive and is a source of numerical error and discontinuity requiring iteration in the design process. Additionally, new tables must be constructed for each change in vehicle shape. In this research, the limitations of this numerical approach was overcome through the development of analytic hypersonic aerodynamic relations for force coefficients, moment coefficients, and stability derivatives at shadowed and unshadowed angles of attack and sideslip. These relations were parametrized for a number of basic shapes (conical frustums, spherical segments, cylindrical segments, and flat plates) as well as for Bezier curves of revolution such that only one set of analytic relations is needed for each family of shapes. The aerodynamics of common hypersonic vehicles of interest such as sphere-cones and blunted biconics can also be expressed analytically using these relations.

To compact these analytic relations where possible and to identify analytic solutions for new configurations, the traditional Newtonian calculation was converted into a flux calculation of a mathematical vector field through an assumed permeable outer mold-line of the vehicle. While the latter deviates from the physical model used to derive Newtonian flow theory, this mathematical model enables the construction of compact, analytic solutions using the Divergence Theorem. Using this technique, the current limit in exact, analytic relations has been established at specific quadratic and semi-quadratic potential functions. As an example of more complex shapes, a parabola of revolution is shown to have nearly an exact solution in which only one integration must be approximated. Since this integration can be analytically approximated, the Divergence Theorem also enables the construction of hybrid exact-approximate analytic solutions. As a result, the aerodynamic coefficients are directly approximated by these hybrid solutions, allowing the error of these coefficients to be directly controlled. This is in direct contrast with current panel methods that approximate vehicle aerodynamics indirectly by approximating vehicle shape.

Comparisons with CBAERO illustrate that these relations provide accurate solutions and can be evaluated nearly instantaneously. As such, the majority of these relations were shown to reduce the computational requirements of panel methods by approximately three orders of magnitude. As a result, this advancement could improve the aerodynamic analysis within traditional, segregated design environments. More importantly, however, these relations provide an analytic mapping of vehicle shape to trajectory performance. This fundamental connection enables vehicle shape to be directly incorporated into the equations of motion, allowing advances in rapid trajectory optimization to be extended to also include vehicle shape.

5.1.2 The Construction of a Rapid Hypersonic Trajectory Optimization Methodology

In traditional, segregated design environments, direct optimization techniques are commonly used to perform a parameter optimization of design variables that span the independently-constructed disciplinary analyses. As a result, trajectory optimization calculations are often limited to these direct optimization techniques to allow the interactions among disciplines to be calculated in a consistent manner. In this dissertation, a rapid trajectory optimization methodology based on fast indirect methods was constructed that leverages the fundamental connectedness of optimal hypersonic trajectory solutions and overcomes the limitations of traditional, segregated design environments. This approach enables the rapid construction of families of optimal trajectory solutions to support design space exploration, trade studies, and vehicle requirements definition. As such, a rapid trajectory optimization framework was presented that combines and advances the disparate trajectory optimization techniques developed over the previous century into a unified framework that is capable of solving a wide range of design problems. This automated framework is the first to execute a sequence of techniques including discrete dynamic programming, nonlinear inversion, pseudospectral methods, indirect methods, and continuation. This process is based on the perspective that trajectory designers are usually interested in directly constructing optimal trajectories. Additionally, a new entry interface based on the physics of the problem that is a function of velocity, vehicle shape, and celestial body was constructed to improve the efficiency of entry trajectory optimization when compared to previous and current studies.

For trajectory optimization problems that are highly sensitive, have many switches in control, or have solutions that are not continuous with trajectory parameters, this robust optimization process is likely required. However, for well-behaved trajectory problems, only a subset of this process was shown to be necessary in which optimal trajectories can be identified through the continuation of fast indirect methods alone. This approach eliminates the complexities associated with the integration of discrete dynamic programming, nonlinear inversion, and pseudospectral methods required by the full trajectory optimization methodology before the continuation of indirect methods. These optimization techniques are required in the full trajectory optimization methodology due to the challenges associated with constructing a good initial guess to converge to an indirect solution within a constrained design space. This approach is largely consistent with traditional, segregated design environments in which all constraints are active from the beginning to guide the calculations within each individual disciplinary analysis. Alternatively, a good initial guess was shown to be easily constructed for a short, unconstrained, optimal trajectory that resides outside of the design space of interest. As such, this dissertation illustrates that solutions within the design space of interest can be efficiently constructed through continuation of indirect methods from this unconstrained solution. In this approach, path constraints are individually introduced and incrementally modified, providing a sequence of rapidly converged optimal trajectory solutions.

To automate this continuation process, state transition tensors were shown to provide accurate predictions of optimal solutions throughout the unconstrained trajectory space. A methodology was developed to predict optimal trajectories with varying flight times, and the error of these predictions was controlled to identify the required steps during the continuation process. It is noted that the use of state transition tensors and continuation of indirect methods could be used for both conceptual design studies and onboard real-time trajectory planning. This approach is easily adaptable to future ground and onboard computing considerations and could serve as a means to bridge the gap between conceptual design and onboard operations. While the performance gap between this simplified approach and traditional hypersonic trajectory design methods is problem dependent, the continuation of indirect methods alone was observed to be approximately two orders of magnitude faster than the current state-of-the-art for the examples presented in this report. These examples also demonstrate that families of optimal trajectories can be rapidly constructed for a wide range of vehicle, planet, and trajectory parameters. These results validate the hypothesis that many hypersonic trajectory solutions are linked through indirect methods. As such, this rapid trajectory methodology was extended to also include vehicle shape through the development of the analytic aerodynamic relations. This enabled the construction of a unified design framework to perform rapid simultaneous hypersonic aerodynamic and trajectory optimization.

5.1.3 The Development of a Unified, Mathematical Framework to Perform Rapid Simultaneous Hypersonic Aerodynamic and Trajectory Optimization

While the prior contributions in analytic hypersonic aerodynamics and rapid trajectory optimization could advance the current state-of-the-art disciplinary analyses within traditional, segregated conceptual design environments, these contributions were carefully selected to create the first mathematically unified design environment that is capable of performing rapid simultaneous hypersonic aerodynamic and trajectory optimization. In this research, the connectedness of solutions illustrated by the rapid trajectory optimization methodology serves as a general design philosophy. For this dissertation, the rapid trajectory methodology was extended to also include vehicle shape, a process made possible by the development of analytic hypersonic aerodynamic relations. The resulting unified mathematical framework enables interactions among trajectory and vehicle shape to be efficiently and simultaneously calculated and is in direct contrast to the current state of the art design methodologies that rely on segregated disciplinary analyses. With the unified design framework, solutions to hypersonic missions that were traditionally solved independently can now be rapidly identified by exploiting the connectivity among indirect optimal solutions.

As a basis for comparison, a single and multi-objective particle swarm optimizer is used as a representative state-of-the-art multidisciplinary optimization methodology that relies on direct shooting for trajectory optimization. Comparisons illustrated that improved solutions can be obtained through continuation of indirect methods. Furthermore, indirect methods are shown to be effective across a range of vehicle shapes that include blunted cones, blended wedges, and blunted biconics for single and multiple design objectives in both trajectory and vehicle shape. For the examples shown, the computational requirements for the particle swarm optimizer were approximately 10-30 times greater than that of the indirect method. Examples using penalty functions also demonstrate the challenges associated with traditional, segregated design environments in which the manner that segregated disciplines are coupled may not be appropriate for all solutions of interest. These examples also illustrate the inability of common direct optimization methods to perform the precise optimization desired. Alternatively, indirect methods were shown to be able to perform the constrained optimization across a wide range of designs by precisely satisfying constraints. For convex Pareto frontiers, this optimization was most efficiently performed through continuation of a weighted sum of the individual objectives. Finally, a non-convex frontier was shown to be easily constructed through continuation of a constrained objective. This unified methodology enables rapid trajectory optimization and design space exploration, rapid sensitivity and robustness analysis, and rapid vehicle requirements definition.

5.2 Future Work

5.2.1 Analytic Hypersonic Aerodynamics

The current analytic aerodynamics database contains all of the solutions obtained to date. Ongoing work should include attempts to expand this database as much as possible. This expansion may be accomplished through alteration of the parametrization of the vehicle and consideration of new classes of shapes. While the parabola of revolution provided an example of a hybrid exact-approximate analytic solution, the extent of these hybrid solutions should also be investigated. Comparisons should also be made with CBAERO to determine the limit in computational usefulness for complex shapes. These solutions could be obtained using the Divergence Theorem or by performing the traditional Newtonian calculation, and comparisons should be made to determine the most efficient approach. While the Divergence Theorem provides an alternate analytic form of the Newtonian calculation, other mathematical techniques such as differential Galois theory should also be explored. Additionally, Taylor series approximations should also be considered to construct hybrid exact-approximate analytic solutions. Note that the limited radius of convergence of common functions such as inverse trigonometric functions and natural logarithms that commonly appear in the analytic relations may limit the extent of approximations using this approach.

While the Divergence Theorem is used to compute analytic force coefficients, consideration should be given to alternate moment calculation techniques. For example, the moment calculation could potentially be converted into a circulation problem, allowing Stokes' Theorem to be used to construct new, compact moment coefficients. While evaluating various mathematical techniques, focus should be given to the compactness of solutions. When these solutions are incorporated into the simultaneous optimization environment, derivatives are taken with respect to the analytic expressions, and compact expressions greatly mitigate the size of the analytic derivatives that result from the chain rule. As such, the computational efficiency of alternate derivative calculations such as complex-step methods should also be compared to the analytic derivatives. Due to the challenges of integration, consideration should also be given to the inverse aerodynamic problem in which a form of the aerodynamic solution is assumed from which the shape could then be obtained through differentiation.

When considering additional classes of shapes, special consideration should be given to ongoing technology development efforts that include inflatable aerodynamic decelerators (IADs). For flexible technologies such as IADs, the analytic relations may provide an opportunity to simplify steady-state fluid structure interaction analysis for conceptual design. An investigation of the coupling between the analytic aerodynamics and other shape-changing systems level analyses should also be performed. For example, shape change due to in-flight configuration changes of morphing vehicles or from TPS ablation should also be considered. Hybrid exact-approximate analytic solutions should also be investigated for non-Newtonian regimes such as free molecular flow. If these solutions exist, they could improve conceptual analyses of satellite and rocket body trajectories. To further improve upon the computational advantages of the simultaneous optimization design environment, the computational efficiency of the analytic relations could be further improved through symbolic code optimization and automatic differentiation techniques.[130, 131] Finally, the database of analytic relations should also be open-sourced to allow widespread adoption of these lengthy expressions while simultaneously incorporating contributions from the hypersonic design community.

5.2.2 Rapid Trajectory Optimization

For well-behaved trajectory problems, optimal trajectories were constructed through the continuation of indirect methods alone. While there are likely many regions within design spaces that qualify as well-behaved, there are likely boundaries between these regions in which large changes in solution are observed. To cross these boundaries, the full, robust trajectory optimization methodology is likely required. As an example, vehicles with decreased ballistic coefficients typically fly higher trajectories in order to arrive to the same terminal point. This is evident in Figure 104 of Section 4.3.2. At a certain ballistic coefficient, the vehicle will likely be required to fly a skipping trajectory to reach the desired terminal point. This sudden change in solution would likely be problematic for the simplified, continuation-only approach, requiring the full, robust trajectory optimization process to bridge this gap. As such, an automated process should be developed to detect the onset of these sudden changes, allowing efficient switching between both trajectory approaches. Alternatively, this new region could possibly be accessed through continuation that begins at a different point outside of the design space. A rigorous methodology that balances these considerations should be developed to allow efficient, automated design space exploration.

In the constrained trajectory examples provided in this dissertation, only two path constraints in heat rate and g-loading were applied. As such, the order of the constraint introduction described in Section 3.7.3 does not influence the final result. Problems with many path constraints should also be evaluated to determine if an intelligent ordering is required to arrive to the desired optimal solution. The prediction of optimal solutions using STTs should also be expanded to include various vehicle parameters, environment parameters, and path constraints to assist the continuation process as these parameters are varied. Additionally, the monitoring of changes in solution, such as vehicle lift to follow path constraints, should also be automated. While this dissertation focused on the optimization of deterministic trajectories for conceptual design, this approach should be expanded to also consider robust design in the presence of uncertainties. For example, uncertainties could be propagated along nominal trajectories using a linear or higher order covariance technique. The family of trajectories provided by these techniques may improve the time required to design reference trajectories of guided entry vehicles. Additionally, a streamtube that contains each trajectory family could be used to visualize the robustness of each design with respect to path constraints, enabling rapid calculation of the corresponding family of feasible solutions. To improve the efficiency of the continuation process for more complex problems, multiple shooting tools, such as BNDSCO, should be used in place of Matlab's built-in BVP4C function used in this dissertation.[129] With improved solvers, a rigorous study on the computational advantages of the continuation process should be performed across various design problems of interest. Finally, alternate problem formulations using quantities such as drag and energy should also be considered.

5.2.3 Rapid Simultaneous Design

If the analytic aerodynamics are expanded to include morphing or flexible structures such as IADs, then the simultaneous vehicle and trajectory optimization could be expanded to include vehicle shape change during flight by modifying Eq. (79) in Section 4.2.1.1. This would allow rapid analysis of new missions of interest, such as downrange targeting capability provided by drag modulation. Additional vehicle objectives such as aerodynamic stability and control surface design should also be included in the design framework. While this dissertation focused on the fundamental integration of hypersonic aerodynamics and trajectory design, other disciplines in the DSM shown in Figure 1 of Section 1.1 such as aeroheating and structures should also be evaluated for incorporation into this rapid design methodology. Physics-based models should be used where possible, but analytic metamodels could also be used as a substitute. This work would serve as a means to study and catalog the fundamental manner in which each discipline can be mathematically integrated into the rapid conceptual design framework.

This general design philosophy of mathematically integrating the fundamental analyses across disciplines should also be extended beyond the unpowered hypersonic flight examples used in this dissertation. Other phases of flight such as launch and orbit should also be included to construct a unified, end-to-end conceptual mission design capability. This capability would enable interactions among various mission phases to be rapidly assessed. For example, the interactions among the fluctuations in Mars' atmosphere during the Martian year, synodic period phasing, fluctuations in Earth's atmosphere, and targeting requirements at the end of entry, descent, and landing could be simultaneously used to drive entry vehicle design and launch vehicle requirements. Techniques used for optimal solution prediction such as multi-order state transition tensors should also be updated to include vehicle parameters, environment parameters, constraints, and other parameters associated with the fundamental analyses of additional disciplines. Finally, incorporating this design methodology into a computing cluster would provide a unified computational framework for further development of this design methodology. By enabling cloud-based access to the design framework through this cluster, the techniques developed in this research could be easily integrated into other research efforts.

5.2.4 Onboard Applications and Hardware Testing

While the work in this dissertation has largely focused on ground-based conceptual design, limited computational resources onboard flight vehicles also demand rapid calculations. Unlike ground-based conceptual design, onboard processes are only concerned with any remaining real-time design calculations. These calculations include real-time trajectory planning as well as configuration or shape change during flight. The continuation-only optimization methodology discussed in this dissertation could be used to perform real-time trajectory planning for small perturbations during flight, and this approach could potentially be expanded to greater flight envelopes. To this end, techniques should be investigated that provide evidence of convergence from an initial guess over a range of expected flight conditions. This may be accomplished through the use of regions of attraction, higher-order state transition tensors, or other techniques. While fielded systems would implement aerodynamic tables derived from high-fidelity CFD analysis, the analytic aerodynamic relations could serve as a substitute during the conceptual design of various onboard algorithms. As an example, indirect methods only require a root-solver and a numerical integration scheme. Onboard calculations such as these have already been planned to support the entry monitor system of the Orion command module.[122] This approach could replace traditional guidance algorithms and eliminate the current disparity between conceptual design and onboard operations. Further advancements in this approach should be evaluated during conceptual design as well as in a flight-relevant hardware environment to address additional gaps that may remain. For example, additional real-time considerations including estimation of vehicle and atmospheric properties could also be incorporated into the conceptual design process.

As methods are evaluated for both conceptual design and onboard applications, consideration must be given to future advances in ground and onboard computing. In both environments, limited computational resources demand rapid calculation of optimal solutions. For example, the emergence of the massive parallelization of scientific computing provided by graphics processing units (GPU) will likely alter the manner in which conceptual design is performed. While NLP solvers can be parallelized, STTs can be computed with relatively small amounts of memory.[132] This enables the relatively limited but fast memory on a GPU to be utilized for rapid parallel computation of STTs.[133] As the capability of GPUs increases, higher order STTs can be computed in parallel to reduce the number of steps required during the continuation process. Additionally, the analytic shooting line should be evaluated to determine if a more efficient path for the continuation policy can be constructed. Finally, as the balance between serial speed and parallelization matures in future computing, indirect methods can also be adapted through proper division of the problem and use of multiple shooting techniques.

The real-time trajectory optimization capability would not be limited to the entry examples provided in this dissertation. The consolidation of trajectory planning into a single mathematical framework could be used to consolidate the many guidance modes commonly found on launch vehicles. For example, the Space Shuttle implemented many guidance modes (RTLS, ECAL, TAL, AOA, ATO, PTM) to account for various vehicle energies, locations, and failure modes at abort initiation. This methodology could be applied across a wide range of launch vehicles to assist commercial space access and would enable abort scenarios to be easily included within the design framework. By expanding the trajectory optimization research in this dissertation to include thrust, closed-loop targeting would be possible during the high dynamic pressure regime that is typically flown open-loop by launch vehicles. Unlike the unpowered hypersonic solutions in this dissertation that are uniquely defined by each altitude-velocity trajectory, the nonunique timeline of powered solutions along these trajectories must also be addressed.

Real-time trajectory planning enabled by this research could also be used for other time-critical applications such as missile avoidance or adversary intercept. Focus should be given to these military applications in which real-time trajectory planning would enable damage mitigation from incoming missiles, effective countermeasure maneuvers, and engagement of air and ground adversaries from advantageous directions. Additionally, these solutions could potentially be updated real-time during maneuver execution to account for dynamic ground and air threats. This work could be used to provide recommend trajectories to pilots via a heads-up display or to command fully autonomous unmanned air vehicles. As such, future work in this research could provide a means for collaboration with current autonomous flight planning research as well as cognitive engineering efforts.

This research may also have applications outside of aerospace. For example, real-time trajectory optimization could be used by the automotive industry for future autonomous vehicle collision mitigation. The rapid construction of trajectories could optimize collision locations (if avoidance is not possible) based on car frame design while accounting for varying traction as a result of current road conditions (rain, ice, etc.) as well as the relative motion between vehicles. Applications such as these may allow development and testing for lower cost applications before implementation into costly aerospace systems. Initial testing of real-time trajectory optimization could be performed in a test bed using inexpensive LEGO Mindstorms robots. While the dynamics of these machines are slow in comparison to autonomous vehicles, the onboard processors could be artificially slowed to a consistent time-constant, providing a relevant test of these algorithms in which obstacles are dynamically placed along optimal paths. In all of these possible areas of future work, the continuation of indirect methods both onboard and during conceptual design may eliminate the current disparity between conceptual design and onboard operations.

APPENDIX A

SAMPLE OF ANALYTIC AERODYNAMICS RELATIONS

A.1 Flat Plate

The following analytic aerodynamic relation for the normal force coefficient provides an example of short expressions that result from simple configurations such as the flat plate described in Section 2.4.3.

$$C_N = 2[\sin(\delta)\cos(\alpha)\cos(\beta) + \cos(\delta)\sin(\alpha)\cos(\beta)]^2\cos(\delta) \tag{82}$$

A.2 Cylindrical Segment

The following analytic aerodynamic relation for the normal force coefficient provides an example of medium-length expressions that result from more complex configurations such as the spherical segment described in Section 2.4.2. Note that this compact expression was constructed using the Divergence Theorem.

$$\begin{aligned}C_N =& 1/192 * \rho^2 * (192 * \cos(\alpha)^2 * (\arccos(\csc(a) * \sin(\delta)) * \sin(\alpha)^3 + \\ & \sin(\delta) * (\sin(\alpha)^2 - 2 * \sin(\delta)^2) * \sqrt{}(1 - \csc(\alpha)^2 * \sin(\delta)^2)) + \\ & (-16 + 16 * \cos(2 * \alpha) + 30 * \sin(\alpha) - \sin(3 * \alpha) + \sin(5 * \alpha)) * \\ & (6 * (-2 * \delta + \pi) + 8 * \sin(2 * \delta) - \sin(4 * \delta)) + \\ & 32 * \sin(\alpha) * (6 * \pi * \cos(\alpha) * \cos(\delta)^4 - \\ & \sin(\alpha) * (12 * \delta - 6 * \pi + 8 * \sin(2 * \delta) + \sin(4 * \delta))))\end{aligned} \tag{83}$$

A.3 Sharp Cone

The following analytic aerodynamic relation for the normal force coefficient provides an example of the lengthy expressions that result from complex configurations such as the sharp cone described in Section 2.4.1. Note that for this expression, the traditional Newtonian calculation was performed.

$$Cn = (-3 \ast u1^2 \ast cos(b)^2 \ast cos(d)^2 \ast cos((4 \ast kapi \ast pi + k1a \ast acos((k2a \ast cos(b) \ast (sqrt(1-2 \ast cos(2 \ast a) \ast cos(b)^2-cos(2 \ast b)+2 \ast cos(2 \ast d)) \ast sec(d) \ast sin(a)+2 \ast k3a \ast cos(a) \ast sin(b) \ast tan(d)))/(2 \ast (cos(b)^2 \ast sin(a)^2+sin(b)^2)))+conj(k1a) \ast conj(acos((k2a \ast cos(b) \ast (sqrt(1-2 \ast cos(2 \ast a) \ast cos(b)^2-cos(2 \ast b)+2 \ast cos(2 \ast d)) \ast sec(d) \ast sin(a)+2 \ast k3a \ast cos(a) \ast sin(b) \ast tan(d)))/(2 \ast (cos(b)^2 \ast sin(a)^2+sin(b)^2)))))/2) \ast cot(d) \ast sin(a)^2)/(4 \ast Aref)+(3 \ast u2^2 \ast cos(b)^2 \ast cos(d)^2 \ast cos((4 \ast kapi \ast pi+k1a \ast acos((k2a \ast cos(b) \ast (sqrt(1-2 \ast cos(2 \ast a) \ast cos(b)^2-cos(2 \ast b)+2 \ast cos(2 \ast d)) \ast sec(d) \ast sin(a)+2 \ast k3a \ast cos(a) \ast sin(b) \ast tan(d)))/(2 \ast (cos(b)^2 \ast sin(a)^2+sin(b)^2)))+conj(k1a) \ast conj(acos((k2a \ast cos(b) \ast (sqrt(1-2 \ast cos(2 \ast a) \ast cos(b)^2-cos(2 \ast b)+2 \ast cos(2 \ast d)) \ast sec(d) \ast sin(a)+2 \ast k3a \ast cos(a) \ast sin(b) \ast tan(d)))/(2 \ast (cos(b)^2 \ast sin(a)^2+sin(b)^2)))))/2) \ast cot(d) \ast sin(a)^2)/(4 \ast Aref)+(u1^2 \ast cos(b)^2 \ast cos(d)^2 \ast cos((3 \ast (4 \ast kapi \ast pi+k1a \ast acos((k2a \ast cos(b) \ast (sqrt(1-2 \ast cos(2 \ast a) \ast cos(b)^2-cos(2 \ast b)+2 \ast cos(2 \ast d)) \ast sec(d) \ast sin(a)+2 \ast k3a \ast cos(a) \ast sin(b) \ast tan(d)))/(2 \ast (cos(b)^2 \ast sin(a)^2+sin(b)^2)))+conj(k1a) \ast conj(acos((k2a \ast cos(b) \ast (sqrt(1-2 \ast cos(2 \ast a) \ast cos(b)^2-cos(2 \ast b)+2 \ast cos(2 \ast d)) \ast sec(d) \ast sin(a)+2 \ast k3a \ast cos(a) \ast sin(b) \ast tan(d)))/(2 \ast (cos(b)^2 \ast sin(a)^2+sin(b)^2))))))/2) \ast cot(d) \ast sin(a)^2)/(12 \ast Aref)-(u2^2 \ast cos(b)^2 \ast cos(d)^2 \ast cos((3 \ast (4 \ast kapi \ast pi+k1a \ast acos((k2a \ast cos(b) \ast (sqrt(1-2 \ast cos(2 \ast a) \ast cos(b)^2-cos(2 \ast b)+2 \ast cos(2 \ast d)) \ast sec(d) \ast sin(a)+2 \ast k3a \ast cos(a) \ast sin(b) \ast tan(d)))/(2 \ast (cos(b)^2 \ast sin(a)^2+sin(b)^2)))+conj(k1a) \ast conj(acos((k2a \ast cos(b) \ast (sqrt(1-2 \ast cos(2 \ast a) \ast cos(b)^2-cos(2 \ast b)+2 \ast cos(2 \ast d)) \ast sec(d) \ast sin(a)+2 \ast k3a \ast cos(a) \ast sin(b) \ast tan(d)))/(2 \ast (cos(b)^2 \ast sin(a)^2+sin(b)^2))))))/2) \ast cot(d) \ast sin(a)^2)/(12 \ast Aref)+(3 \ast u1^2 \ast cos(b)^2 \ast cos(d)^2 \ast cos((4 \ast kbpi \ast pi+k1b \ast acos((k2b \ast cos(b) \ast (sqrt(1-2 \ast cos(2 \ast a) \ast cos(b)^2-cos(2 \ast b)+2 \ast cos(2 \ast d)) \ast sec(d) \ast sin(a)+2 \ast k3b \ast cos(a) \ast sin(b) \ast tan(d)))/(2 \ast (cos(b)^2 \ast sin(a)^2+sin(b)^2)))+$$

$conj(k1b)*conj(acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*$
$cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+$
$sin(b)^2)))))/2)*cot(d)*sin(a)^2)/(4*Aref)-(3*u2^2*cos(b)^2*cos(d)^2*cos((4*kbpi*pi+$
$k1b*acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*$
$sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1b)*$
$conj(acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*$
$sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))))/2)*cot(d)*$
$sin(a)^2)/(4*Aref)-(u1^2*cos(b)^2*cos(d)^2*cos((3*(4*kbpi*pi+k1b*acos((k2b*cos(b)*$
$(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*$
$sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1b)*conj(acos((k2b*cos(b)*$
$(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*$
$sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2))))))/2)*cot(d)*sin(a)^2)/(12*Aref)+$
$(u2^2*cos(b)^2*cos(d)^2*cos((3*(4*kbpi*pi+k1b*acos((k2b*cos(b)*(sqrt(1-2*cos(2*$
$a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*$
$(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1b)*conj(acos((k2b*cos(b)*(sqrt(1-2*cos(2*$
$a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*$
$(cos(b)^2*sin(a)^2+sin(b)^2))))))/2)*cot(d)*sin(a)^2)/(12*Aref)-(kapi*pi*u1^2*$
$cos(b)^2*cos(d)^2*sin(2*a))/Aref+(kbpi*pi*u1^2*cos(b)^2*cos(d)^2*sin(2*a))/Aref+$
$(kapi*pi*u2^2*cos(b)^2*cos(d)^2*sin(2*a))/Aref-(kbpi*pi*u2^2*cos(b)^2*cos(d)^2*$
$sin(2*a))/Aref-(k1a*u1^2*acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*$
$b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+$
$sin(b)^2)))*cos(b)^2*cos(d)^2*sin(2*a))/(4*Aref)+(k1a*u2^2*acos((k2a*cos(b)*$
$(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*$
$sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))*cos(b)^2*cos(d)^2*sin(2*a))/(4*$
$Aref)+(k1b*u1^2*acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*$
$d))*sec(d)*sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))*$
$cos(b)^2*cos(d)^2*sin(2*a))/(4*Aref)-(k1b*u2^2*acos((k2b*cos(b)*(sqrt(1-2*cos(2*$

$$a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2* (cos(b)^2*sin(a)^2+sin(b)^2)))*cos(b)^2*cos(d)^2*sin(2*a))/(4*Aref)-(u1^2*conj(k1a)* conj(acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)* sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2))))*cos(b)^2* cos(d)^2*sin(2*a))/(4*Aref)+(u2^2*conj(k1a)*conj(acos((k2a*cos(b)*(sqrt(1-2* cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)* tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2))))*cos(b)^2*cos(d)^2*sin(2*a))/(4*Aref)+ (u1^2*conj(k1b)*conj(acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+ 2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+ sin(b)^2))))*cos(b)^2*cos(d)^2*sin(2*a))/(4*Aref)-(u2^2*conj(k1b)*conj(acos((k2b* cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b* cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2))))*cos(b)^2*cos(d)^2*sin(2* a))/(4*Aref)-(u1^2*cos(d)^2*cos((4*kapi*pi+k1a*acos((k2a*cos(b)*(sqrt(1-2* cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)* tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1a)*conj(acos((k2a*cos(b)* (sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)* sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))))/2)^3*cot(d)*sin(b)^2)/(3*Aref)+ (u2^2*cos(d)^2*cos((4*kapi*pi+k1a*acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2- cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2* sin(a)^2+sin(b)^2)))+conj(k1a)*conj(acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2- cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2* sin(a)^2+sin(b)^2)))))/2)^3*cot(d)*sin(b)^2)/(3*Aref)+(u1^2*cos(d)^2*cos((4*kbpi*pi+ k1b*acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)* sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1b)* conj(acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)* sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))))/2)^3* cot(d)*sin(b)^2)/(3*Aref)-(u2^2*cos(d)^2*cos((4*kbpi*pi+k1b*acos((k2b*cos(b)*$$

$$(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)* sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1b)*conj(acos((k2b*cos(b)* (sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)* sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2))))))/2)^3*cot(d)*sin(b)^2)/(3*Aref)- (u1^2*cos(a)*cos(d)^2*cos((4*kapi*pi+k1a*acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)* cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2* (cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1a)*conj(acos((k2a*cos(b)*(sqrt(1-2*cos(2* a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2* (cos(b)^2*sin(a)^2+sin(b)^2))))))/2)^2*sin(2*b))/(2*Aref)+(u2^2*cos(a)*cos(d)^2* cos((4*kapi*pi+k1a*acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+ 2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+ sin(b)^2)))+conj(k1a)*conj(acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2* b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+ sin(b)^2))))))/2)^2*sin(2*b))/(2*Aref)+(u1^2*cos(a)*cos(d)^2*cos((4*kbpi*pi+k1b* acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)* sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1b)* conj(acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)* sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2))))))/2)^2* sin(2*b))/(2*Aref)-(u2^2*cos(a)*cos(d)^2*cos((4*kbpi*pi+k1b*acos((k2b*cos(b)* (sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)* sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1b)*conj(acos((k2b*cos(b)* (sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)* sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2))))))/2)^2*sin(2*b))/(2*Aref)-(u1^2* cos(a)^2*cos(b)^2*cos((4*kapi*pi+k1a*acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)* cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2* (cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1a)*conj(acos((k2a*cos(b)*(sqrt(1-2*cos(2* a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*$$

$(cos(b)^2 * sin(a)^2 + sin(b)^2)))))/2) * sin(2 * d))/(2 * Aref) + (u2^2 * cos(a)^2 * cos(b)^2 *$

$cos((4*kapi*pi+k1a*acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+$

$2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+$

$sin(b)^2)))+conj(k1a)*conj(acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*$

$b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+$

$sin(b)^2)))))/2)*sin(2*d))/(2*Aref)+(u1^2*cos(a)^2*cos(b)^2*cos((4*kbpi*pi+k1b*$

$acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*$

$sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1b)*$

$conj(acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*$

$sec(d)*sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))))/2)*$

$sin(2*d))/(2*Aref)-(u2^2*cos(a)^2*cos(b)^2*cos((4*kbpi*pi+k1b*acos((k2b*cos(b)*$

$(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*$

$sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1b)*conj(acos((k2b*cos(b)*$

$(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*$

$sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))))/2)*sin(2*d))/(2*Aref)-(u1^2*$

$cos(d)^2*cot(d)*sin(a)*sin(2*b)*sin((4*kapi*pi+k1a*acos((k2a*cos(b)*(sqrt(1-$

$2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*$

$tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1a)*conj(acos((k2a*cos(b)*$

$(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*$

$sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))))/2)^3)/(3*Aref)+(u2^2*cos(d)^2*$

$cot(d)*sin(a)*sin(2*b)*sin((4*kapi*pi+k1a*acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*$

$cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*$

$(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1a)*conj(acos((k2a*cos(b)*(sqrt(1-2*cos(2*$

$a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*$

$(cos(b)^2*sin(a)^2+sin(b)^2)))))/2)^3)/(3*Aref)+(u1^2*cos(b)^2*cos(d)^2*sin(2*a)*$

$sin(4*kapi*pi+k1a*acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+$

$2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+$

$$
\begin{gathered}
sin(b)^2)))+conj(k1a)*conj(acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2* \\
b)+2*cos(2*d))*sec(d)*sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2* \\
sin(a)^2+sin(b)^2))))))/(4*Aref)-(u2^2*cos(b)^2*cos(d)^2*sin(2*a)*sin(4*kapi*pi+ \\
k1a*acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)* \\
sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1a)* \\
conj(acos((k2a*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)* \\
sin(a)+2*k3a*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2))))))/(4* \\
Aref)+(u1^2*cos(d)^2*cot(d)*sin(a)*sin(2*b)*sin((4*kbpi*pi+k1b*acos((k2b*cos(b)* \\
(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)* \\
sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1b)*conj(acos((k2b*cos(b)* \\
(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)* \\
sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))))/2)^3)/(3*Aref)-(u2^2*cos(d)^2* \\
cot(d)*sin(a)*sin(2*b)*sin((4*kbpi*pi+k1b*acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)* \\
cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2* \\
(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1b)*conj(acos((k2b*cos(b)*(sqrt(1-2*cos(2* \\
a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2* \\
(cos(b)^2*sin(a)^2+sin(b)^2)))))/2)^3)/(3*Aref)-(u1^2*cos(b)^2*cos(d)^2*sin(2*a)* \\
sin(4*kbpi*pi+k1b*acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+ \\
2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+ \\
sin(b)^2)))+conj(k1b)*conj(acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2* \\
b)+2*cos(2*d))*sec(d)*sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2* \\
sin(a)^2+sin(b)^2))))))/(4*Aref)+(u2^2*cos(b)^2*cos(d)^2*sin(2*a)*sin(4*kbpi*pi+ \\
k1b*acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)* \\
sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2)))+conj(k1b)* \\
conj(acos((k2b*cos(b)*(sqrt(1-2*cos(2*a)*cos(b)^2-cos(2*b)+2*cos(2*d))*sec(d)* \\
sin(a)+2*k3b*cos(a)*sin(b)*tan(d)))/(2*(cos(b)^2*sin(a)^2+sin(b)^2))))))/(4*Aref)
\end{gathered}
$$

APPENDIX B

PAPERS STATUS

B.1 Forthcoming Publications

1. Journal submission to JSR regarding analytic aerodynamics including divergence theorem results and computational comparisons to CBAERO.

2. Journal submission to JSR regarding the rapid trajectory optimization framework based on continuation methods.

3. Journal submission to JSR regarding the rapid, simultaneous design framework made possible by the advances in analytic hypersonic aerodynamics and rapid trajectory optimization.

B.2 Relevant Publications

1. Grant, M. and Braun, R., "Analytic Hypersonic Aerodynamics for Conceptual Design of Entry Vehicles," AIAA 2010-1212, 48th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition, Orlando, FL, 4-7 Jan. 2010.

2. Grant, M., Clark. I., and Braun, R., "Rapid Entry Corridor Trajectory Optimization for Conceptual Design," AIAA 2010-7810, AIAA Atmospheric Flight Mechanics Conference and Exhibit, Toronto, Ontario, Canada, 2-5 Aug. 2010.

3. Grant, M., Clark. I., and Braun, R., "Rapid Simultaneous Hypersonic Aerodynamic and Trajectory Optimization Using Variational Methods," AIAA 2011-6640, AIAA Atmospheric Flight Mechanics Conference and Exhibit, Portland, OR, 8-11 Aug. 2011.

4. Grant, M., Clark. I., and Braun, R., “Rapid Design Space Exploration for Conceptual Design of Hypersonic Missions,” AIAA 2011-6576, AIAA Atmospheric Flight Mechanics Conference and Exhibit, Portland, OR, 8-11 Aug. 2011.

B.3 Additional Publications

1. Grant, M., Steinfeldt, B., Braun, R., and Barton, G., “Smart Divert: A New Mars Robotic Entry, Descent, and Landing Architecture,” Journal of Spacecraft and Rockets, Vol. 47, No. 3, 2010, pp. 385-393.

   doi: 10.2514/1.47030

2. Steinfeldt, B., Grant, M., Matz, D., and Braun, R., “Guidance, Navigation, and Control System Performance Trades for Mars Pinpoint Landing,” Journal of Spacecraft and Rockets, Vol. 47, No. 1, 2010, pp. 188-198.

   doi: 10.2514/1.45779

3. Baird, D., Grant, M., Kadwa, B., et al., “Orion Entry Display Feeder and Interactions with the Entry Monitor System,” AIAA 2010-8062, AIAA Guidance, Navigation, and Control Conference, Toronto, Ontario, Canada, 2-5 Aug. 2010.

4. Steinfeldt, B., Theisinger, J., Korzun, A., Clark, I., Grant, M., and Braun, R., “High Mass Mars Entry, Descent, and Landing Architecture Assessment,” AIAA 2009-6684, AIAA Space 2009 Conference and Exposition, Pasadena, CA, 14-17 Sept. 2009.

5. Grant, M., Steinfeldt, B., Braun, R., and Barton, G., “Smart Divert: A New Entry, Descent, and Landing Architecture,” AIAA 2009-522, 47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition, Orlando, FL, 5-8 Jan. 2009.

6. Steinfeldt, B., Grant, M., Matz, D., and Braun, R., "Guidance, Navigation, and Control Technology System Trades for Mars Pinpoint Landing," AIAA 2008-6216, AIAA Atmospheric Flight Mechanics Conference and Exhibit, Honolulu, HI, 18-21 Aug. 2008.

7. Grant, M. and Mendeck, G., "Mars Science Laboratory Entry Optimization Using Particle Swarm Methodology," AIAA 2007-6393, AIAA Atmospheric Flight Mechanics Conference and Exhibit, Hilton Head, SC, 20-23 Aug. 2007.

8. Lafleur, J., Restrepo, C., and Grant, M., "Development, Structure, and Application of MAST: A Generic Mission Architecture Sizing Tool," AIAA 2006-1717, 2nd AIAA Multidisciplinary Design Optimization Specialist Conference, Newport, RI, 1-4 May 2006.

REFERENCES

[1] Bryson, A. E. and Ho, Y.-C., *Applied Optimal Control*, Taylor and Francis, 1975.

[2] Otero, R. E. and Braun, R. D., “The Planetary Entry Systems Synthesis Tool: A Conceptual Design and Analysis Tool for EDL Systems,” IEEEAC 1331, *2010 IEEE Aerospace Conference*, Big Sky, MT, Mar. 2010.

[3] Anderson, J. D., *Hypersonic and High Temperature Gas Dynamics*, AIAA, 1989.

[4] Eggers, A. J., Resnikoff, M. M., and Dennis, D. H., “Bodies of Revolution Having Minimum Drag at High Supersonic Airspeeds,” *NACA TR-1306*, 1957.

[5] Gong, Q., Ross, I. M., Kang, W., and Fahroo, F., “On the Pseudospectral Covector Mapping Theorem for Nonlinear Optimal Control,” *45th IEEE Conference on Decision and Control*, San Diego, CA, 13-15 Dec. 2006.

[6] George B. Thomas, J. and Finney, R. L., *Calculus*, Addison-Wesley Publishing Company, 1996.

[7] Bryson, A. E., Denham, W. F., and Dreyfus, S. E., “Optimal Programming Problems with Inequality Constraints I: Necessary Conditions for Extremal Solutions,” *AIAA Journal*, Vol. 1, No. 11, 1963.

[8] Acton, D. E. and Olds, J. R., “Computational Frameworks for Collaborative Multidisciplinary Design of Complex Systems,” AIAA 98-4942, *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, St. Louis, MO, 2-4 Sep. 1998.

[9] Olds, J. R., “The Suitability of Selected Multidisciplinary Design and Optimization Techniques to Conceptual Aerospace Vehicle Design,” AIAA 92-4791, *4th AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization*, Cleveland, OH, 21-23 Sep. 1992.

[10] Braun, R. D., Powell, R. W., Lepsch, R. A., Stanley, D. O., and Kroo, I. M., “Comparison of Two Multidisciplinary Optimization Strategies for Launch-Vehicle Design,” *Journal of Spacecraft and Rockets*, Vol. 32, No. 3, 1995.

[11] Brown, N. F. and Olds, J. R., “Evaluation of Multidisciplinary Optimization Techniques Applied to a Reusable Launch Vehicle,” *Journal of Spacecraft and Rockets*, Vol. 43, No. 6, 2006.

[12] Perez, R. E., Liu, H. H. T., and Behdinan, K., "Evaluation of Multidisciplinary Optimization Approaches for Aircraft Conceptual Design," AIAA 2004-4537, *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, NY, 30 Aug. - 1 Sep. 2004.

[13] Braun, R. D., *Collaborative Optimization: An architecture for large-scale distributed design*, PhD Thesis, Stanford University, Department of Aeronautics and Astronautics, May 1996.

[14] Sobieszczanski-Sobieski, J., Agte, J. S., and Sandusky, R. R., "Bi-Level Integrated System Synthesis (BLISS)," *NASA-TM-1998-208715*, 1998.

[15] Rowell, L. F., Braun, R. D., Olds, J. R., and Unal, R., "Multidisciplinary Conceptual Design Optimization of Space Transportation Systems," *Journal of Aircraft*, Vol. 36, No. 1, 1999.

[16] Padula, S. L., Korte, J. J., Dunn, H. J., and Salas, A. O., "Multidisciplinary Optimization Branch Experience Using iSIGHT Software," *NASA-TM-1999-209714*, 1999.

[17] Baker, M. L., Munson, M. J., Hoppus, G. W., and Alston, K. Y., "The Integrated Hypersonic Aeromechanics Tool (IHAT), Build 4," AIAA 2004-4565, *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, NY, 30 Aug. - 1 Sep. 2004.

[18] Leary, S. J., Birtwell, P. M., Holden, H., and Johnson, G. A., "The Efficient Multi-Objective Design of Air-Vehicle Configurations using ModelCenter," AIAA 2006-6946, *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Portsmouth, VA, 6-8 Sep. 2006.

[19] Newton, I., *Principia - Motte's Translation Revised*, University of California Press, 1946.

[20] Anderson, Jr., J. D., *Fundamentals of Aerodynamics*, McGraw-Hill, 2001.

[21] Cummings, R. M. and Yang, H.-T., "Lester Lees and Hypersonic Aerodynamics," *Journal of Spacecraft and Rockets*, Vol. 40, No. 4, 2003.

[22] Bonner, E., Clever, W., and Dunn, K., "Aerodynamic Preliminary Analysis System II: Part I Theory," *NASA-CR-165627*, Apr. 1981.

[23] Smyth, D. N. and Loo, H. C., "Analysis of Static Pressure Data from 1/12-scale Model of the YF-12A. Volume 3: The MARK IVS Supersonic-Hypersonic Arbitrary Body Program, User's Manual," *NASA-CR-151940*, Oct. 1981.

[24] Cunningham, M., "Hypersonic Aerodynamics for an Entry Research Vehicle," *Journal of Spacecraft and Rockets*, Vol. 24, No. 2, 1987.

[25] Kinney, D. J., "Aero-Thermodynamics for Conceptual Design," AIAA-2004-31-962, *42nd AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, 5-8 Jan. 2004.

[26] GTS Library Reference Manual, http://gts.sourceforge.net/reference/book1.html.

[27] Hoffman, S. J. and Kaplan, D. I., "Human Exploration of Mars: The Reference Mission of the NASA Mars Exploration Study Team," *NASA Special Publication 6107*, July 1997.

[28] Drake, B. G., "Reference Mission Version 3.0 Addendum to the Human Exploration of Mars: The Reference Mission of the NASA Mars Exploration Study Team," *NASA Special Publication 6107-ADD*, June 1998.

[29] Steinfeldt, B., Theisinger, J., Korzun, A., Clark, I., Grant, M., and Braun, R., "High Mass Mars Entry Descent and Landing Architecture Assessment," AIAA 2009-6684, *AIAA Space 2009*, Pasadena, CA, 14 - 17 Sept. 2009.

[30] Desai, P., Lyons, D., Tooley, J., and Kangas, J., "Entry, Descent, and Landing Operations Analysis for the Stardust Entry Capsule," *Journal of Spacecraft and Rockets*, Vol. 45, No. 6, 2008.

[31] Desai, P. and Lyons, D., "Entry, Descent, and Landing Operations Analysis for the Genesis Entry Capsule," *Journal of Spacecraft and Rockets*, Vol. 45, No. 1, 2008.

[32] Kinney, D. J. and Bowles, J. V., "Conceptual Design of a 'SHARP'-CTV," AIAA 2001-2887, *35th AIAA Thermophysics Conference*, Anaheim, CA, 11-14 Jun. 2001.

[33] Wells, W. and Armstrong, W., "Tables of Aerodynamic Coefficients Obtained From Developed Newtonian Expressions for Complete and Partial Conic and Spheric Bodies at Combined Angles of Attack and Sideslip with Some Comparisons with Hypersonic Experimental Data," *NASA TR R-127*, 1962.

[34] Elsgolc, L. D., *Calculus of Variations*, Dover Publications, Inc., 2007.

[35] Auman, L. M. and Wilks, B., "Supersonic and Hypersonic Minimum Drag for Bodies of Revolution," *AIAA 21st Applied Aerodynamics Conference*, Orlando, FL, 23-26 Jun. 2003.

[36] Rainey, R. W., "Working Charts for Rapid Prediction of Force and Pressure Coefficients on Arbitrary Bodies of Revolution by Use of Newtonian Concepts," *NASA TN D-176*, 1959.

[37] Regan, F. J. and Anandakrishnan, S. M., *Dynamics of Atmospheric Re-Entry*, AIAA, 1993.

[38] Grimminger, G., Williams, E. P., and Young, G. B. W., "Lift on Inclined Bodies of Revolution in Hypersonic Flow," *Journal of the Aeronautical Sciences*, Vol. 17, No. 11, 1950.

[39] Margolis, K., "Theoretical Evaluation of the Pressures, Forces, and Moments at Hypersonic Speeds Acting on Arbitrary Bodies of Revolution Undergoing Separate and Combined Angle-of-Attack and Pitching Motions," *NASA TN D-652*, 1961.

[40] Theisinger, J. E. and Braun, R. D., "Multi-Objective Hypersonic Entry Aeroshell Shape Optimization," *Journal of Spacecraft and Rockets*, Vol. 46, No. 5, 2009.

[41] Kinney, D. J., "Aerodynamic Shape Optimization of Hypersonic Vehicles," AIAA 2006-239, *44th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, 9-12 Jan. 2006.

[42] Mathematica, Ver. 7, Wolfram Research, Champaign, IL.

[43] Maple, Ver. 13, Waterloo Maple Inc., Waterloo, Ontario, Canada.

[44] Petrov, I. P., *Variational Methods in Optimum Control Theory*, Academic Press Inc., 1968.

[45] Hull, D. G., *Optimal Control Theory for Applications*, Mechanical Engineering Series, Springer, 2003.

[46] Jezewski, D. J., "Primer Vector Theory and Applications," *NASA TR R-454*, 1975.

[47] Ranieri, C. L. and Ocampo, C. A., "Optimization of Roundtrip, Time-Constrained, Finite Burn Trajectories via an Indirect Method," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 2, 2005.

[48] Russell, R., "Primer Vector Theory Applied to Global Low-Thrust Trade Studies," AAS 06-156, *16th AAS/AIAA Space Flight Mechanics Conference*, Tampa, FL, 22-26 Jan. 2006.

[49] Sakai, T., Old, J. R., and Alemany, K., "Development of *SAMURAI* - Simulation and Animation Model for Rockets with Adjustable Isp," AAS 05-181, *15th AAS/AIAA Space Flight Mechanics Conference*, Copper Mountain, CO, Jan. 2005.

[50] Mendeck, G. F. and Carman, G. L., "Guidance Design for Mars Smart Landers Using The Entry Terminal Point Controller," AIAA-2002-4502, *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Monterey, CA, 5-8 Aug. 2002.

[51] Enright, P. and Conway, B., "Discrete Approximations to Optimal Trajectories Using Direct Transcription and Nonlinear Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 4, 1992.

[52] Riehl, J. P., Paris, S. W., and Sjauw, W. K., "Comparison of Implicit Integration Methods for Solving Aerospace Trajectory Optimization Problems," AIAA-2006-6033, *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, Keystone, CO, 21-24 Aug. 2006.

[53] Hargraves, C. R. and Paris, S. W., "Direct Trajectory Optimization Using Nonlinear Programming and Collocation," *Journal of Guidance*, Vol. 10, No. 4, 1987.

[54] Bibeau, R. and Rubenstein, D., "Trajectory Optimization for a Fixed-Term Reentry Vehicle Using Direct Collocation and Nonlinear Programming," AIAA-2000-4262, *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Denver, CO, 14-17 Aug. 2000.

[55] Herman, A. and Conway, B., "Direct Optimization Using Collocation Based on High-Order Gauss-Lobatto Quadrature Rules," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 3, 1996.

[56] Seywald, H., "Trajectory Optimization Based on Differential Inclusion," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 3, 1994.

[57] Kumar, R. and Seywald, H., "Should Controls Be Eliminated While Solving Optimal Control Problems via Direct Methods?" *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 2, 1996.

[58] Conway, B. A. and Larson, K. M., "Collocation Versus Differential Inclusion in Direct Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 5, 1998.

[59] User's Guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming, University of California, San Diego, CA, Stanford University, Stanford, CA, 16 Jun. 2008.

[60] Grant, M. J. and Mendeck, G. F., "Mars Science Laboratory Entry Optimization Using Particle Swarm Methodology," AIAA 2007-6393, *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Hilton Head, SC, 20-23 Aug. 2007.

[61] Lafleur, J. and Cerimele, C., "Mars Entry Bank Profile Design for Terminal State Optimization," AIAA 2008-6213, *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Honolulu, HI, 18-21 Aug. 2008.

[62] Carter, T. E., "State Transition Matrices for Terminal Rendezvous Studies: Brief Survey and New Example," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 1, 1998.

[63] Vanderplaats, G. N., *Numerical Optimization Techniques for Engineering Design*, Vanderplaats Research and Development, Inc., 2005.

[64] Barton, G. H., “New Methodologies for Assessing the Robustness of the X-34 Autolanding Trajectories,” *American Astronomical Society*, Breckenridge, CO, 31 Jan. - 4 Feb. 2001.

[65] Barton, G. H., USN, L. A. C. G., and USN, L. T. R. D., “New Methodologies for Onboard Generation of TAEM Trajectories for Autonomous RLVs,” *Core Technologies for Space Systems Conference*, 2002.

[66] Jain, S. and Tsiotras, P., “Multiresolution-Based Direct Trajectory Optimization,” *46th IEEE Conference on Decision and Control*, New Orleans, LA, 12-14 Dec. 2007.

[67] Seywald, H., “Optimal and Suboptimal Minimum Time-To-Climb Trajectories,” AIAA-94-3554, *AIAA Guidance, Navigation, and Control Conference and Exhibit*, New Orleans, LA, Aug. 1994.

[68] Seywald, H., Cliff, E. M., and Well, K. H., “Range Optimization for a Supersonic Aircraft,” AIAA-91-2712, *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Scottsdale, AZ, Aug. 1991.

[69] Seywald, H., “Long Flight-Time Range-Optimal Aircraft Trajectories,” *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 1, 1996.

[70] Cullum, J., “Finite-Dimensional Approximations of State-Constrained Continuous Optimal Control Problems,” *SIAM Journal of Control*, Vol. 10, No. 4, 1972.

[71] Gong, Q., Ross, I. M., Kang, W., and Fahroo, F., “Connections Between the Covector Mapping Theorem and Convergence of Pseudospectral Methods for Optimal Control,” *Journal of Computational Optimization and Applications*, Vol. 41, No. 3, 2008.

[72] Fahroo, F. and Ross, I. M., “Costate Estimation by a Legendre Pseudospectral Method,” *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 2, 2001.

[73] Fornberg, B., *A Practical Guide to Pseudospectral Methods*, Cambridge University Press, 1996.

[74] Fahroo, F. and Ross, I. M., “Direct Trajectory Optimization by a Chebyshev Pseudospectral Method,” *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 1, 2002.

[75] Golub, G. H., “Some Modified Matrix Eigenvalue Problems,” *SIAM Review*, Vol. 15, No. 2, 1973.

[76] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C., *Introduction to Algorithms, Second Edition*, MIT Press and McGraw-Hill, 2001.

[77] Bellman, R. E., *Dynamic Programming*, Princeton University Press, 1957.

[78] Bellman, R. E. and Dreyfus, S. E., *Applied Dynamic Programming*, Princeton University Press, 1962.

[79] McCausland, I., *Introduction to Optimal Control Theory*, John Wiley and Sons, Inc., 1969.

[80] Lin, Y., *General Systems Theory: A Mathematical Approach*, Springer, 2002.

[81] Lewis, F. L., *Optimal Estimation with an Introduction to Stochastic Control Theory*, John Wiley and Sons, 1986.

[82] Kirk, D. E., *Optimal Control Theory: An Introduction*, Prentice-Hall, Inc., 1970.

[83] POST, Program to Optimize Simulated Trajectories, Ver. 5.2, NASA Langley Research Center, Hampton, Virginia, Martin Marietta Corporation, Denver, CO, 1997.

[84] SORT, Simulation and Optimization of Rocket Trajectories, Ver. 8.3, NASA Johnson Space Center, Houston, TX.

[85] Striepe, S. A., Way, D. W., Dwyer, A. M., and Balaram, J., "Mars Smart Lander Simulations for Entry, Descent, and Landing," AIAA-2002-4412, *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Monterey, CA, 5-8 Aug. 2002.

[86] Way, D. W., Powell, R. W., Edquist, K. T., Masciarelli, J. P., and Starr, B. R., "Aerocapture Simulation and Performance for the Titan Explorer Mission," AIAA-2003-4951, *39th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*, Huntsville, AL, 20-23 Jul. 2003.

[87] Starr, B. R., Westhelle, C. H., and Masciarelli, J. P., "Aerocapture Performance Analysis for a Neptune-Triton Exploration Mission," AIAA-2004-4955, *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Providence, RI, 16-19 Aug. 2004.

[88] Starr, B. R. and Westhelle, C. H., "Aerocapture Performance Analysis of a Venus Exploration Mission," AIAA-2005-5913, *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, San Francisco, CA, 15-18 Aug. 2005.

[89] Tartabini, P. V., Bose, D. M., McMinn, J. D., Martin, J. G., and Strovers, B. K., "Hyper-X Stage Separation Trajectory Validation Studies," AIAA-2003-5819, *AIAA Modeling and Simulation Technologies Conference and Exhibit*, Austin, TX, 11-14 Aug. 2003.

[90] Kraft, Dieter, A Software Package for Sequential Quadratic Programming, DFVLR-FB 88-28, Koln, Germany SNAP References, 1988.

[91] Falck, R. D. and Gefert, L. P., "Crew Exploration Vehicle Ascent Abort Trajectory Analysis and Optimization," AIAA-2007-6775, *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Hilton Head, SC, 20-23 Aug. 2007.

[92] Polsgrove, T., Kos, L., Hopkins, R., and Crane, T., "Comparison of Performance Predictions for New Low-Thrust Trajectory Tools," AIAA-2006-6742, *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, Keystone, CO, 21-24 Aug. 2006.

[93] Krivanek, T. M., Roche, J. M., Riehl, J. P., and Kosareo, D. N., "Affordable Flight Demonstration of the GTX Air-Breathing SSTO Vehicle Concept," *NASA-TM-2003-212315*, Apr. 2003.

[94] Lu, P., Griffin, B., Dukeman, G. A., and Chavez, F. R., "Rapid Optimal Multi-Burn Ascent Planning and Guidance," AIAA-2007-6773, *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Hilton Head, SC, 20-23 Aug. 2007.

[95] Ross, I. Michael, A Beginner's Guide to DIDO, Ver. 7.3, Document TR-711 Elissar, LLC, Monterey, CA, 1997.

[96] User's Manual for GPOPS Version 3.0: A MATLAB Software for Solving Multiple-Phase Optimal Control Problems Using Pseudospectral Methods, University of Florida, Gainesville, FL, The Charles Stark Draper Laboratory, Cambridge, MA, and Blue Origin, Seattle, WA., 2010.

[97] Ross, I. M., Gong, Q., and Sekhavat, P., "Low-Thrust, High-Accuracy Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 4, 2007.

[98] Infeld, S. I., Josselyn, S. B., Murray, W., and Ross, I. M., "Design and Control of Libration Point Spacecraft Formations," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 4, 2007.

[99] Fleming, A., Sekhavat, P., and Ross, I. M., "Minimum-Time Reorientation of a Rigid Body," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 1, 2010.

[100] Shaffer, P. J., Ross, I. M., Oppenheimer, M. W., Doman, D. B., and Bollino, K. P., "Fault-Tolerant Optimal Trajectory Generation for Reusable Launch Vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 6, 2007.

[101] Bedrossian, N., Bhatt, S., Lammers, M., Nguyen, L., and Zhang, Y., "First Ever Flight Demonstration of Zero Propellant Maneuver Attitude Control Concept," AIAA-2007-6734, *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Hilton Head, SC, 20-23 Aug. 2007.

[102] Huntington, G. T. and Rao, A. V., "Optimal Configuration of Spacecraft Formations Via a Gauss Pseudospectral Method," AAS 05-103, *AAS/AIAA Astrodynamics Specialists Conference*, Lake Tahoe, CA, 7-11 Aug. 2005.

[103] Huntington, G. T. and Rao, A. V., “Optimal Reconfiguration of a Tetrahedral Formation Via a Gauss Pseudospectral Method,” AAS 05-338, *AAS/AIAA Astrodynamics Specialists Conference*, Lake Tahoe, CA, 7-11 Aug. 2005.

[104] Huntington, G. T., Benson, D. A., and Rao, A. V., “Post-Optimality Evaluation and Analysis of a Formation Flying Problem Via a Gauss Pseudospectral Method,” AAS 05-339, *AAS/AIAA Astrodynamics Specialists Conference*, Lake Tahoe, CA, 7-11 Aug. 2005.

[105] Gogu, C., Matsumura, T., Haftka, R. T., and Rao, A. V., “Aeroassisted Orbital Transfer Trajectory Optimization Considering Thermal Protection System Mass,” *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 3, 2009.

[106] Rao, A. V., Scherich, A. W., Cox, S., and Mosher, T., “A Concept for Operationally Responsive Space Mission Planning Using Aeroassisted Orbital Transfer,” AIAA-RS6-2008-1001, *AIAA Responsive Space 6*, Los Angeles, CA, 27 Apr. - 1 May 2008.

[107] Rogers, D. F., *An Introduction to NURBS: With Historical Perspective*, Academic Press, London, 2001.

[108] Josselyn, S. and Ross, I. M., “Rapid Verification Method for the Trajectory Optimization of Reentry Vehicles,” *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 3, 2003.

[109] Jorris, T. R., Schulz, C. S., Friedl, F. R., and Rao, A. V., “Constrained Trajectory Optimization Using Pseudospectral Methods,” AIAA-2008-6218, *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Honolulu, HI, 18-21 Aug. 2008.

[110] Hartl, R. F., Sethi, S. P., and Vickson, R. G., “A Survey of the Maximum Principles for Optimal Control Problems with State Constraints,” *SIAM Review*, Vol. 37, No. 2, 1995.

[111] Pontryagin, L. S., Boltyanskii, V. G., Gamkrelidze, R. V., and Mishchenko, E. F., *The Mathematical Theory of Optimal Processes*, John Wiley and Sons, Inc., 1962.

[112] Grant, M. J. and Braun, R. D., “Analytic Hypersonic Aerodynamics for Conceptual Design of Entry Vehicles,” *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, Orlando, FL, AIAA 2010-1212, 4-7 Jan. 2010.

[113] Grant, M. J., Clark, I. G., and Braun, R. D., “Rapid Entry Corridor Trajectory Optimization for Conceptual Design,” AIAA 2010-7810, *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Toronto, Ontario, Canada, 2-5 Aug. 2010.

[114] Darby, C. L., Hager, W. W., and Rao, A. V., "An hp-Adaptive Pseudospectral Method for Solving Optimal Control Problems," *Optimal Control Applications and Methods*, Vol. 32, 2010.

[115] Zhao, Y. and Tsiotras, P., "Density Functions for Mesh Refinement in Numerical Optimal Control," *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 1, 2011.

[116] Elsgloc, L. D., *Calculus of Variations*, Dover Publications, Inc., 2007.

[117] Park, R. S. and Scheeres, D. J., "Nonlinear Mapping of Gaussian Statistics: Theory and Applications to Spacecraft Trajectory Design," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 6, 2006.

[118] Turner, J. D., Majji, M., and Junkins, J. L., "High-Order State and Parameter Transition Tensor Calculations," AIAA-2008-6453, *AIAA/AAS Astrodynamics Specialist Conference*, Honolulu, HI, 18-21 Aug. 2008.

[119] Sengupta, P., Vadali, S. R., and Alfriend, K. T., "Second-Order State Transition for Relative Motion Near Perturbed, Elliptic Orbits," *Celestial Mechanics and Dynamical Astronomy*, Vol. 97, No. 2, 2007.

[120] Russell, R. P. and Lantoine, G., "A Hybrid Differential Dynamic Programming Algorithm for Robust Low-Thrust Optimization," AIAA-2008-6615, *AIAA/AAS Astrodynamics Specialist Conference*, Honolulu, HI, 18-21 Aug. 2008.

[121] Russell, R. P. and Lantoine, G., "Optimal Control of Relative Motion in Arbitrary Fields: Application at Deimos," AAS 10-313, *Kyle T. Alfriend Astrodynamics Symposium*, Monterey, CA, 17-19 May 2010.

[122] Baird, D., Grant, M., Kadwa, B., Gillespie, E., Matthews, D., Penny, W., Zak, T., and Bihari, B., "Orion Entry Display Feeder and Interactions with the Entry Monitor System," AIAA 2010-8062, *AIAA Guidance, Navigation, and Control Conference*, Toronto, Ontario, Canada, 2-5 Aug. 2010.

[123] Messac, A. and Ismail-Yahaya, A., "Required Relationship Between Objective Function and Pareto Frontier Orders: Practical Implications," *AIAA Journal*, Vol. 39, No. 11, 2001.

[124] Messac, A., Sundararaj, G. J., Tappeta, R. V., and Renaud, J. E., "Ability of Objective Functions to Generate Points on Nonconvex Pareto Frontiers," *AIAA Journal*, Vol. 38, No. 6, 2000.

[125] de Weck, O. and Kim, I. Y., "Adaptive Weighted Sum Method for Bi-objective Optimization," AIAA 2004-1680, *45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, Palm Springs, CA, 19-22 Apr. 2004.

[126] Garcia, J. A., Brown, J. L., Kinney, D. J., Bowles, J. V., Huynh, L. C., Jiang, X. J., Lau, E., and Dupzyk, I. C., “Co-Optimization of Mid Lift to Drag Vehicle Concepts for Mars Atmospheric Entry,” AIAA 2010-5052, *10th AIAA/ASME Joint Thermophysics and Heat Transfer Conference*, Chicago, IL, 28 Jun. - 1 Jul. 2010.

[127] Wolpert, D. H. and Macready, W. G., “No Free Lunch Theorems for Optimization,” *Technical Report, IBM Almaden Research Center and Santa Fe Institute*, 1996.

[128] Grant, M. J., Clark, I. G., and Braun, R. D., “Rapid Design Space Exploration for Conceptual Design of Hypersonic Missions,” *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Portland, OR, 8-11 Aug. 2011.

[129] Oberle, H. J. and Grimm, W., BNDSCO: A Program for the Numerical Solution of Optimal Control Problems, University of Hamburg, Hamburg, Germany, Oct. 2001.

[130] Su, J. and Renaud, J. E., “Automatic Differentiation in Robust Optimization,” *AIAA Journal*, Vol. 35, No. 6, 1997.

[131] Turner, J. D., “Automated Generation of High-Order Partial Derivative Models,” *AIAA Journal*, Vol. 41, No. 8, 2003.

[132] Dayde, M., “Parallel Algorithms for Nonlinear Programming Problems,” *Journal of Optimization Theory and Applications*, Vol. 61, No. 1, 1989.

[133] Arora, N., Russell, R. P., and Vuduc, R. W., “Fast Sensitivity Computations for Trajectory Optimization,” AAS 09-337, *AIAA/AAS Astrodynamics Specialist Conference*, Pittsburgh, PA, 9-13 Aug. 2009.

[134] Grant, M. J., Steinfeldt, B. A., Braun, R. D., and Barton, G. H., “Smart Divert: A New Entry, Descent, and Landing Architecture for Mars Robotic Exploration,” *Journal of Spacecraft and Rockets*, Vol. 47, No. 3, 2010, pp. 385-393.

[135] Striepe, S., Way, D., Dwyer, A., and Balaram, J., “Mars Science Laboratory Simulations for Entry, Descent, and Landing,” Vol. 43, 2006.

[136] Putnam, Z. R., Braun, R. D., Bairstow, S. H., and Barton, G. H., “Improving Lunar Return Entry Footprints Using Enhanced Skip Trajectory Guidance,” AIAA-2006-7438, *Space 2006*, San Jose, CA, 19-21 Sep. 2006.

[137] Luus, R., *Iterative Dynamic Programming*, Chapman and Hall, 2000.

[138] GTS, GNU Triangulated Surface Library, http://gts.sourceforge.net/.

[139] User’s Guide for NPSOL 5.0: A Fortran Package for Nonlinear Programming, University of California, San Diego, CA, Stanford University, Stanford, CA, Lucent Technologies, Murray Hill, NJ, 30 Jul. 1998.

[140] Liao, L. Z. and Shoemaker, C. A., “Advantages of Differential Dynamic Programming Over Newton’s Method for Discrete-Time Optimal Control Problems,” *Technical Report, Cornell University*, 1993.

[141] Tsuchiya, T. and Mori, T., “Optimal Conceptual Design of Two-Stage Reusable Rocket Vehicles Including Trajectory Optimization,” *Journal of Spacecraft and Rockets*, Vol. 41, No. 5, 2004.

[142] Castellini, F., Lavagna, M. R., Riccardi, A., and Buskens, C., “Multidisciplinary Design Optimization Models and Algorithms for Space Launch Vehicles,” AIAA 2010-9086, *13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, Fort Worth, TX, 13-15 Sep. 2010.

[143] Grant, M. J., Clark, I. G., and Braun, R. D., “Rapid Simultaneous Hypersonic Aerodynamic and Trajectory Optimization Using Variational Methods,” *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Portland, OR, 8-11 Aug. 2011.

[144] Armellin, R., Lavagna, M., Starkey, R. P., and Lewis, M. J., “Aerogravity-Assist Maneuvers: Coupled Trajectory and Vehicle Shape Optimization,” *Journal of Spacecraft and Rockets*, Vol. 44, No. 5, 2007.