AIAA Atmospheric Flight Mechanics Conference
2 - 5 August 2010, Toronto, Ontario Canada

AIAA 2010-7810

Rapid Entry Corridor Trajectory Optimization for Conceptual Design

Michael J. Grant*, Ian G. Clark†, and Robert D. Braun‡
Georgia Institute of Technology, Atlanta, GA, 30332

During conceptual design, an entry configuration is chosen to provide an envelope of vehicle performance throughout the entry corridor that satisfies mission requirements. In many applications, this process is performed using computationally intensive direct methods. In this investigation, an automated process has been developed to perform rapid trajectory optimization using indirect methods. 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. The results from pseudospectral methods identify challenges in the formulation of corner conditions and switching structure associated with indirect methods. Examples demonstrate that families of optimal trajectories can be quickly constructed for varying trajectory parameters, vehicle shape, atmospheric properties, and gravity. These results validate the hypothesis that many entry trajectory solutions are linked through indirect methods. This framework enables rapid trajectory optimization and design space exploration, rapid sensitivity and robustness analysis, and rapid vehicle requirements definition.

Nomenclature

BVP	Boundary Value Problem
CBAERO	Configuration Based Aerodynamics
CMT	Covector Mapping Theorem
FPA	Flight Path Angle
GPOPS	Gauss Pseudospectral Optimization Software
KKT	Karush-Kuhn-Tucker
NLP	Nonlinear Programming
TPS	Thermal Protection System
A	vehicle reference area, m^2
C_D	drag coefficient
C_L	lift coefficient
D	drag force magnitude, N
H	scale height, m
I	path cost
J	cost function
L	lift force magnitude, N
L/D	lift to drag ratio
m	vehicle mass, kg
$\dot{q}$	heat rate, W/cm^2

*Graduate Research Assistant, Guggenheim School of Aerospace Engineering, AIAA Student Member.
†Visiting Assistant Professor, Guggenheim School of Aerospace Engineering, AIAA Member.
‡David and Andrew Lewis Associate Professor of Space Technology, Guggenheim School of Aerospace Engineering, AIAA Fellow.

r	radial magnitude, m
r_e	Earth radius, m
S	path constraint
t	time, s
t_0	initial time, s
t_f	final time, s
$\mathbf{u}$	control vector
v	relative velocity magnitude, m/s
$\mathbf{x}$	state vector
α	angle of attack, deg
β	ballistic coefficient, kg/m^2
μ	gravitational parameter, m^3/s^2
γ	relative flight path angle, rad
θ	downrange subtended angle, rad
ρ	atmospheric density, kg/m^3
ρ_0	atmospheric density at the surface, kg/m^3
Φ	terminal cost
ϕ	bank angle, deg

I. Introduction

Current and historical trajectory optimization research has largely focused on the single design of independent optimal trajectories. However, the solution to many optimal trajectories is desired during conceptual design to support design space exploration and trade studies. 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.[1–3] 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. Additionally, local optimality is not guaranteed for a converged direct solution. Indirect methods improve upon the computational requirements of direct methods through the use of necessary conditions of the optimal control problem.[4,5] Satisfaction of the necessary conditions implies 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.[6]

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.[7] This enables designers to capitalize on the advantages of both methods in which a direct method can be used to obtain the costates of an indirect method. For this reason, pseudospectral methods have been widely used for modern trajectory optimization problems.[8,9] 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 entry 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 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 entry problem, design studies can be rapidly performed using continuation.

II. Trajectory Optimization Overview

In general, the trajectory optimization problem is expressed in the form of Eq. (1), 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 are present for many entry missions and are expressed in the form of Eq. (2). Finally, the equations of motion are given in the form of Eq. (3).

$$J = \Phi[\mathbf{x}(t_f), t_f] + \int_{t_0}^{t_f} I(\mathbf{x}(t), \mathbf{u}(t), t) dt \tag{1}$$

$$\Psi[\mathbf{x}(t_f), t_f] = 0 \tag{2}$$

$$\dot{\mathbf{x}} = f[\mathbf{x}(t), \mathbf{u}(t), t], \quad t_0 \text{ given} \tag{3}$$

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

Table 1. Comparison between direct and indirect methods.

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

II.A. Indirect Methods

Prior to modern computing, indirect methods were used to obtain analytic solutions to simple optimal control problems. For certain problems, these solutions can be generalized as function of trajectory parameters. Indirect methods are derived to identify an extremum of the functional J.[4,5] This stationarization of the functional requires solution to a multi-point boundary value problem that is formulated from the first order necessary conditions of optimality.[6] Indirect methods require the use of costates that increase the complexity of the problem by effectively doubling the number of states. Additionally, if path constraints are introduced, then corner conditions must be satisfied at the entrance and exit of the constraint. For entry problems, common constraints such as g-loading and heat rate can be expressed as inequality constraints as shown in Eq. (4).

$$S(\mathbf{x}, t) \leq 0 \tag{4}$$

Indirect methods require the use of costates and, potentially, corner conditions that greatly increase the complexity of optimal solutions. As a result, a good initial guess in states, costates, control, and corner conditions is required to converge to an indirect solution. During conceptual design of various entry missions, the designer may not be capable of providing a sufficient initial guess. Additionally, the complex optimization of entry trajectories requires the use of computer-based boundary value problem methods, including shooting methods and collocation. The solution to the boundary value problem (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 problematic during conceptual design in which the optimization of many trajectories throughout the entry corridor is required. Consequently, direct methods have gained popularity for solving modern optimal control problems.

II.B. 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 is used to identify optimal solutions based on simulation results.

A full factorial search of all possible discrete control history combinations is impractical. For example, assuming a 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 $37^{10} \approx 4.81 \times 10^{15}$ candidate solutions. Furthermore, the entry trajectory problem is highly multimodal, adding complexity to the optimization process. Hence, intelligent global search methods such as genetic algorithms and particle swarm optimization algorithms have been developed to automate the optimization process and intelligently explore the design space for global optimal solutions.[10,11] These methods eliminate the need for a good initial guess and are beneficial for conceptual design of various entry trajectories. These methods have also been used to simultaneously solve for Pareto frontiers, or optimal families, of entry trajectories.[10] This departure from solving a single optimal trajectory to solving a family of optimal trajectories is desired in conceptual design. However, no guarantee can be made about the global optimality of the final solution. Additionally, these algorithms inefficiently attempt to account for path constraints indirectly through manipulation of the control history. While these algorithms automate the trajectory design process, computational requirements are massive since many iterations, corresponding to many propagated trajectories, are required during the global search of these population-based methods.

More elaborate direct methods have been developed that intelligently discretize the state and/or control using an appropriate function approximation.[12,13] Various forms of direct methods have been developed, including collocation[1–3] and differential inclusion.[14–16] The application of these carefully selected quadrature rules results in a discrete nonlinear optimization problem. The structure of these nonlinear optimization problems is very sparse, and nonlinear programming (NLP) solvers, such as SNOPT[17] and NPSOL,[18] have been greatly advanced over the decades to efficiently solve this problem. Additionally, these methods do not require the use of costates. As a result, indirect methods have largely been replaced with direct methods.

II.C. Pseudospectral Methods

Prior to the development of pseudospectral methods, trajectory designers were 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. The recent development of pseudospectral methods has enabled the mapping of results between direct and indirect methods. This allows designers to combine the ease of implementation of direct methods and speed of indirect methods into a single unified rapid trajectory optimization framework.

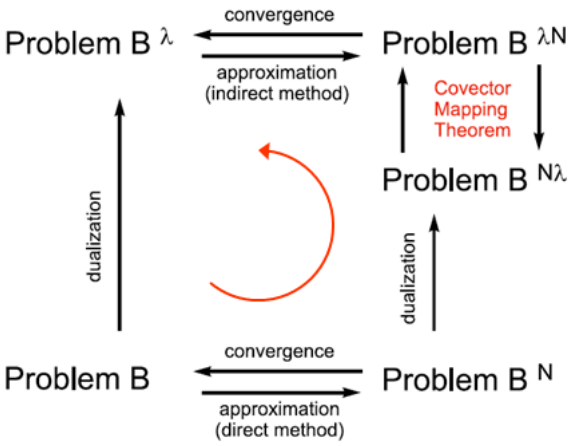


Figure 1. Covector Mapping Theorem diagram.[7]

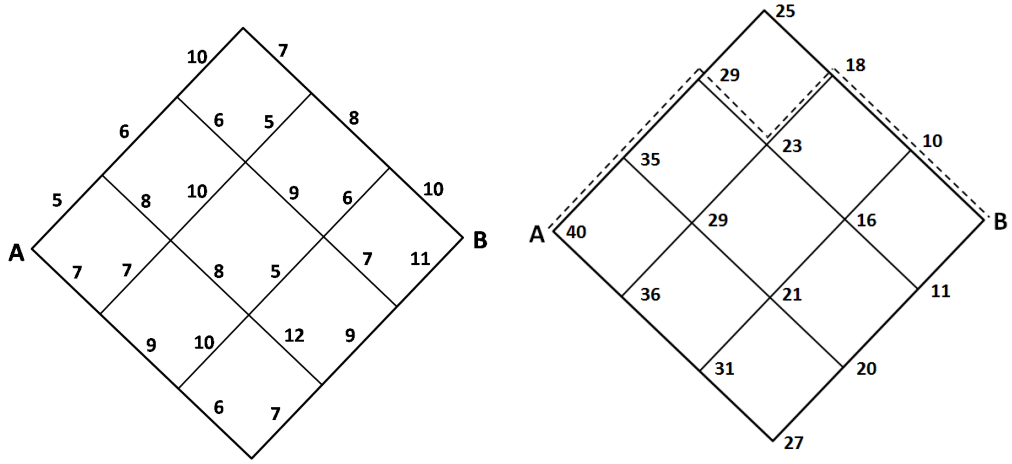
If the designer is presented with a general trajectory optimization problem represented as Problem B shown in Figure 1,[7] the designer could choose to apply an indirect method, arriving to problem $B^{\lambda N}$, or a direct method, arriving to problem $B^{N\lambda}$. Previous studies have shown that if an inappropriate choice in discretization is made, then the Lagrange multipliers of direct methods are not consistent with the discrete costates of indirect methods. This inability to commute the discretization and dualization was originally viewed as an unfortunate reality of trajectory optimization. Example problems have been developed to demonstrate this inconsistency between the Karush-Kuhn-Tucker (KKT) multipliers and discrete costates

even though convergence between the states and controls is observed.[19] However, the recent development of the Covector Mapping Theorem has enabled a mapping between the results of direct and indirect methods. Through proper choice in discretization, the CMT provides a mapping between the discrete KKT multipliers of direct methods and the discrete costates of indirect methods.[20] In many pseudospectral applications, the costates are used to validate the necessary conditions of optimality provided by indirect methods.[21] Moreover, the designer is presented with the opportunity to use intuitive direct methods to arrive to a converged indirect solution.

II.D. Dynamic Programming

Dynamic programming is an efficient methodology when solving multistage optimization problems.[6,22–25] Dynamic programming is based on Bellman's Principle of Optimality that states an optimal policy has the property that no matter what the previous decisions have been, the remaining decisions must constitute an optimal policy with regard to the state resulting from those previous decisions.[26,27] Using this approach, large optimization problems are reduced to smaller subproblems. An example of discrete dynamic programming is shown in Figure 2.[6] Figure 2(a) depicts the cost associated with traveling between adjacent nodes, and Figure 2(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 constructed in reverse from the terminal node, B. Optimal paths are efficiently constructed by only storing the optimal trajectory from each node to the terminal point. Thus, the optimal paths to the terminal point from left-side nodes are comprised of optimal sub-paths from right-side nodes to the terminal point. The tree of optimal paths is constructed until the initial point, A, is reached. In this example, there is complete freedom to travel to any adjacent node. As such, the global optimal path from A to B is efficiently identified without evaluating all possible paths, as shown in Table 2 where 15 computations were required with 20 possible paths in this example.[6] 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.



(a) Discrete Step Cost.

(b) Discrete Path Cost.

Figure 2. Dynamic programming example.[6]

Table 2. Dynamic programming computations compared to total possible routes.[6]

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	$(n+1)^2-1$

III. A New Perspective on Trajectory Design

Both direct and indirect methods were developed to identify optimal trajectories and the 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, a shift in focus from manipulating control histories to manipulating trajectory profiles may provide useful insight into the trajectory design space.

For this study, a planar entry trajectory is assumed with equations of motion shown in Eq. (5)-(8), 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 presented in this report. A high performance blunted biconic was chosen with geometric and aerodynamic parameters shown in Table 3. An entry mass of 136 kg was assumed, resulting in a ballistic coefficient, β, of 4400 kg/m^2.

$$\frac{dr}{dt} = v\sin\gamma \tag{5}$$

$$\frac{d\theta}{dt} = \frac{v\cos\gamma}{r} \tag{6}$$

$$\frac{dv}{dt} = -\frac{D}{m} - \frac{\mu\sin\gamma}{r^2} \tag{7}$$

$$\frac{d\gamma}{dt} = \frac{L\cos(\phi)}{mv} + \left(\frac{v}{r} - \frac{\mu}{vr^2}\right)\cos\gamma \tag{8}$$

Table 3. 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 4.

Table 4. 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 3 and includes 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.

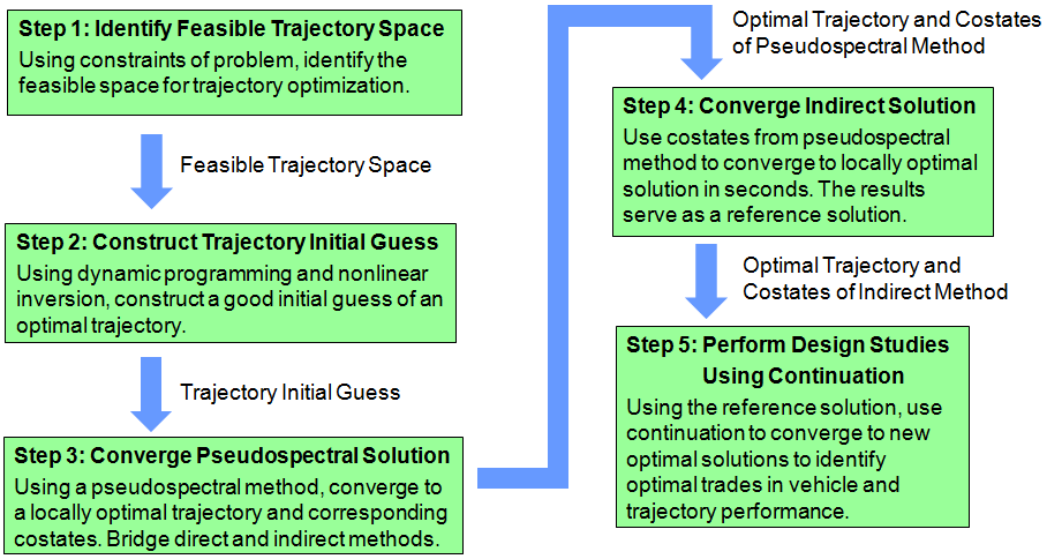


Figure 3. Flowchart of the trajectory optimization framework.

III.A. Rapid Trajectory Optimization

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 options in the lower portion of this space as shown in Figure 4. The upper portion of this space is traditionally bounded by a maximum altitude of approximately 120 km. This altitude represents the limit in atmospheric data from high-altitude weather balloons and, consequently, is chosen as the entry interface where entry simulations begin. Trajectory optimization near this entry interface is inefficient since 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 4. 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 be performed throughout the remaining feasible space defined as the entry corridor.

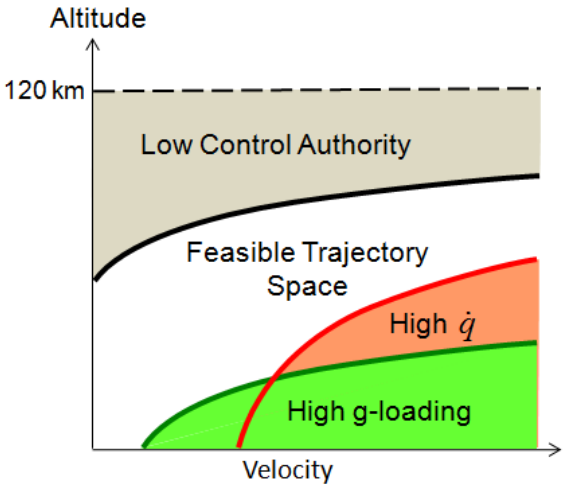


Figure 4. Entry corridor.

Step 2: Entry Corridor Discretization and Initial Guess Construction

In many optimal control algorithms, the addition of path constraints increases complexity of the problem. However, as shown in Figure 4, 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 mesh as shown in Figure 5. This mesh represents a set of potential waypoints that can be used to construct optimal trajectories. If vehicle control authority is neglected, then the entry trajectory optimization process resembles the pathfinding problem discussed in Section II.D. 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.

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 will likely not be able to follow the discrete optimal paths. Therefore, the discrete dynamic programming solutions that represent global optimal, unlimited control authority solutions would likely serve as a good initial trajectory guess for slender entry vehicles. Several example discrete dynamic programming solutions are shown in Figure 5. For this study, the discrete dynamic programming solution shown in Figure 6 was 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 remaining states and control required to construct an initial guess for pseudospectral methods are computed using nonlinear inversion.

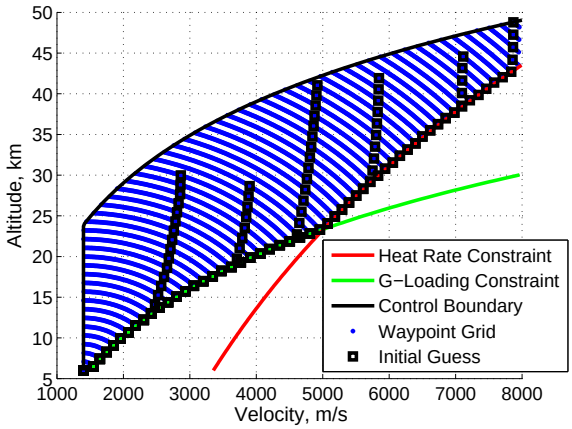


Figure 5. Potential initial guesses from various discrete dynamic programming solutions.

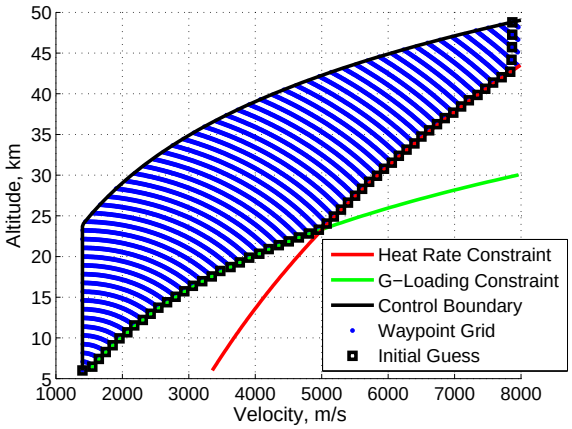


Figure 6. Dynamic programming solution selected for initial guess.

After the trajectory is constructed in altitude-velocity space from discrete dynamic programming, the flight-path angle (FPA) can be obtained from Eq. (9) 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. (9) 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 vehicle control authority. The remaining quantities of interest (time, downrange angle, and bank angle) can also be computed using nonlinear inversion.

$$\sin\gamma = \frac{\frac{dh}{dv}\left(-\frac{1}{2m}\rho V C_D A\right)}{1 + \frac{dh}{dv}\frac{\mu}{r^2 V}} \tag{9}$$

Step 3: Pseudospectral Method Execution

The Gauss Pseudospectral Optimization Software (GPOPS)[28] 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.[7] 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 7 using the dynamic programming initial guess from Figure 6. There are two regions where the pseudospectral solution does not follow the dynamic programming initial guess. First, the vehicle is not capable of following the dynamic programming solution in the early part of the trajectory as shown in Figure 8. This is an example where the vehicle with limited control authority is not capable of following the dynamic programming solution. 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 9. This particular vehicle has sufficient control authority to follow the g-loading constraint, and the departure from the constraint is an artifact of a low convergence tolerance used by the pseudospectral method.

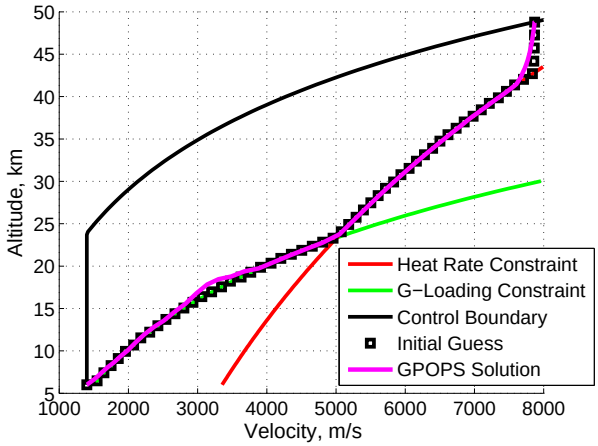


Figure 7. Converged pseudospectral solution.

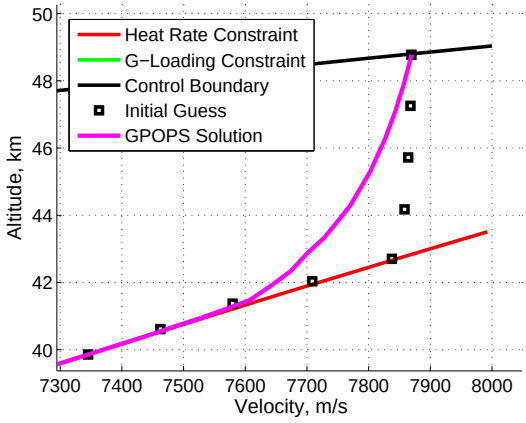


Figure 8. Beginning of converged pseudospectral solution.

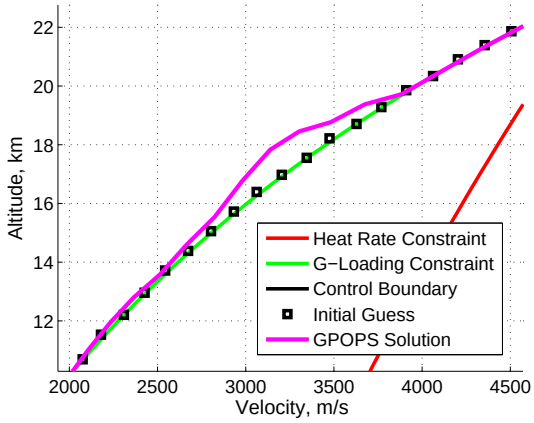


Figure 9. Converged pseudospectral solution along g-loading constraint.

As a direct method, execution of the pseudospectral method can be computationally intensive. Many current design studies, including those for the conventional prompt global strike mission, 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 pseudospectral solution was obtained in 45 minutes, however the rate of convergence and accuracy of the solution can 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 convergence to a single pseudospectral solution is highly dependent on designer interaction, and the time required for convergence can range from several minutes to hours. 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.

Initially, the solution obtained from GPOPS would not converge 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 10.[29] 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.

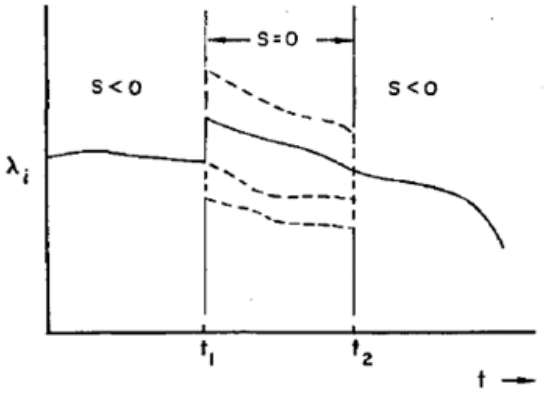


Figure 10. Nonuniqueness of costates.[29]

Corner conditions can be classified as a direct form (D-form) or Pontryagin form (P-form).[30] 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 11. 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.

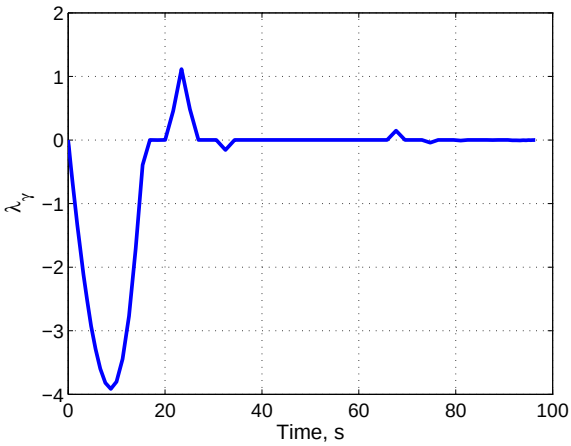


Figure 11. λ_γ of pseudospectral solution.

When deriving the necessary conditions of optimality, Pontryagin's Minimum Principle requires the Hamiltonian be minimized with respect to control for all time.[31] This results in a switching structure shown in Table 5 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 11, 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 indirect methods. 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 12 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.

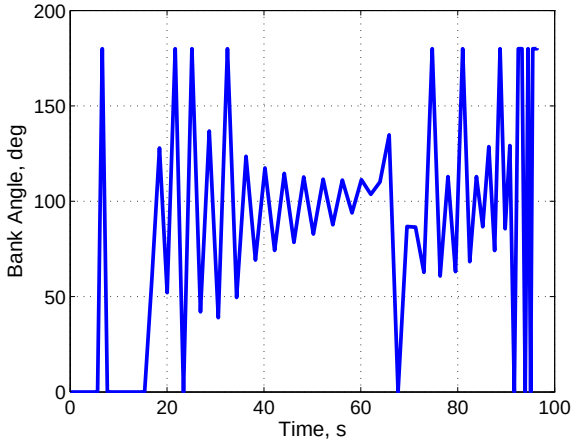


Figure 12. Bank angle of pseudospectral solution.

Table 5. Control switching structure.

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

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 13, the pseudospectral solution is within the region of attraction, allowing convergence of the indirect method within seconds. 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 in the beginning part of the trajectory as shown in Figure 14.

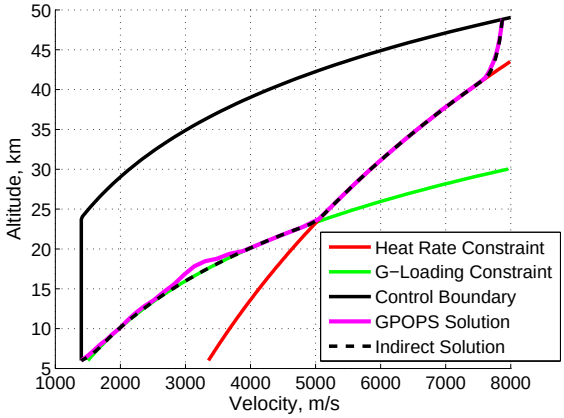


Figure 13. Fully converged indirect solution.

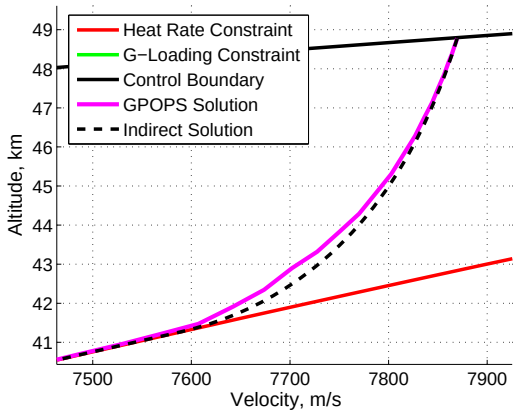


Figure 14. Beginning of converged indirect solution.

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. Nearby optimal trajectories can be obtained by substituting the pseudospectral solution with the reference solution as an initial guess. This process can be repeated in succession using nearby indirect solutions to rapidly obtain a family of indirect solutions using continuation.

Step 5: Design Using Continuation

Starting with the reference 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 initial velocity was reduced, and a family of optimal trajectories in Figure 15 was obtained in minutes. Using continuation, the optimal trade in initial velocity and heat load can be constructed as shown in Figure 16. When performing continuation, the change in trajectory parameters must be small enough 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 illustrate that large changes can be made to trajectory parameters during the continuation process.

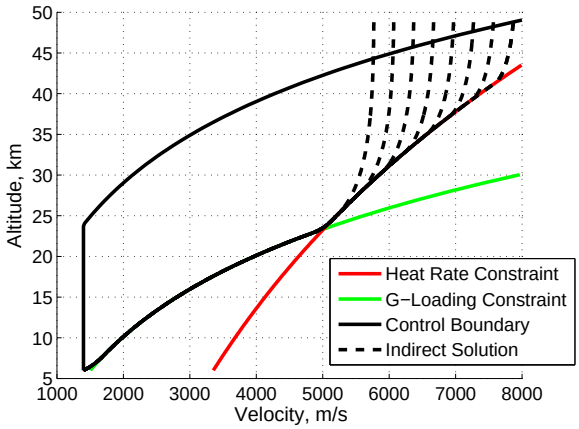


Figure 15. Optimal trajectories for varying initial velocity.

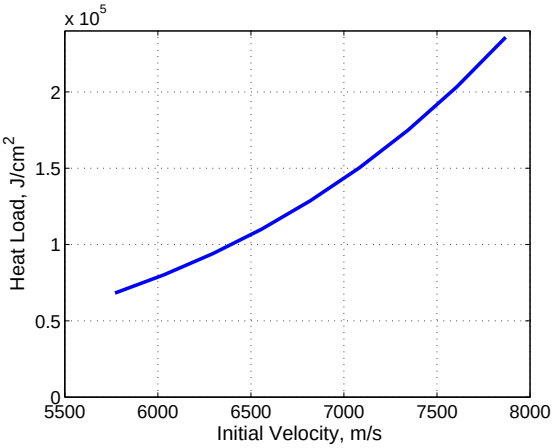


Figure 16. Optimal trade in initial velocity and minimum heat load.

III.B. Extensibility of Continuation

III.B.1. Additional Trajectory Parameters

The previous example demonstrated 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 constructed in minutes using continuation. The minimum heat load trajectories corresponding to various heat rate constraints are shown in Figure 17, and the optimal trade in heat rate and heat load is shown in Figure 18.

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 in minutes, enabling system-level requirements such as g-loading to be rapidly mapped to vehicle performance such as heat rate and heat load.

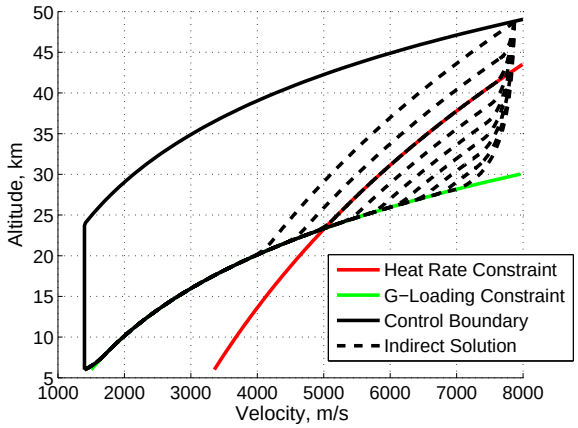


Figure 17. Optimal trajectories for varying heat rate constraint.

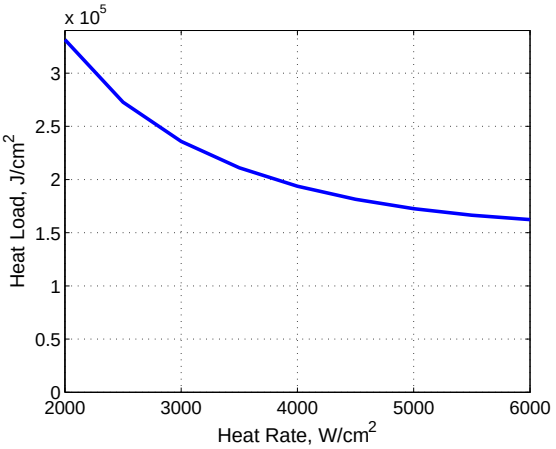


Figure 18. Optimal trade in heat rate constraint and minimum heat load.

III.B.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. Prior work focused on the development of analytic hypersonic relations as a function of vehicle shape, angle of attack, and sideslip.[32] These relations were validated with a current state-of-the-art hypersonic design tool, the Configuration Based Aerodynamics (CBAERO) tool. Unlike panel methods that include CBAERO, the analytic mapping of vehicle shape to aerodynamic performance allows vehicle shape to be included in the continuation process. As an example, the slender biconic used during the continuation of trajectory parameters was evolved to more blunt biconics as shown in Figure 19. As the vehicle shape is altered, minimum heat load trajectories converge in seconds using continuation. The resulting trajectories are shown in Figure 20.

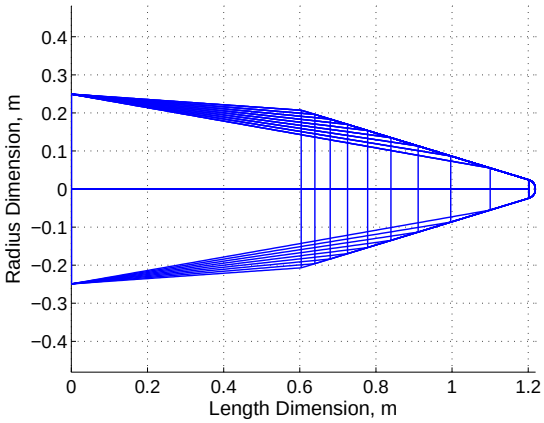


Figure 19. Vehicle shape change.

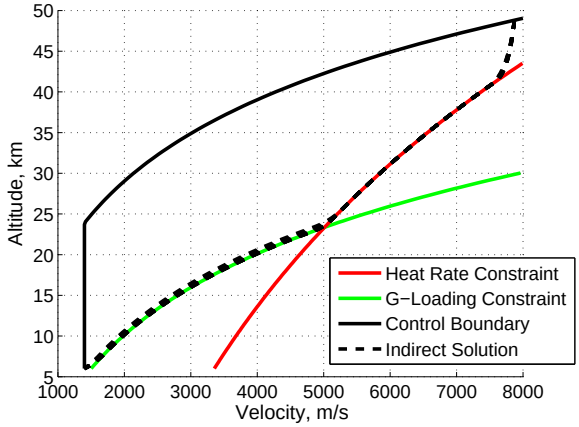


Figure 20. Optimal trajectories for varying vehicle shape.

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 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 20 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 can be rapidly constructed for this range of shapes as shown in Figure 21. Although the altitude-velocity trajectory of each vehicle shape is similar, the downrange and time of each trajectory differ substantially. As the ballistic coefficient is decreased, the time of the trajectory also decreases, resulting in reduced optimal heat loads.

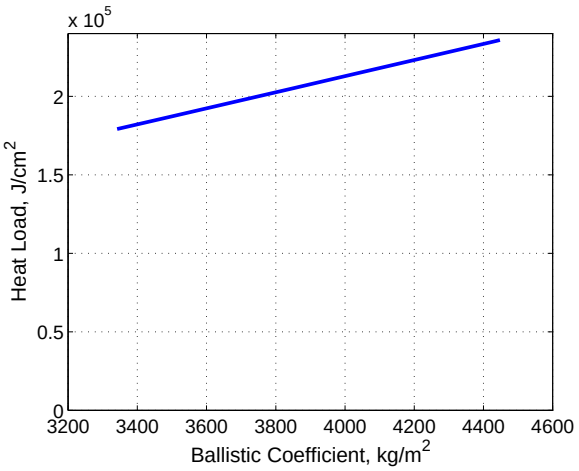


Figure 21. Optimal trade in ballistic coefficient and minimum heat load.

III.B.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 22 and Figure 23, respectively. 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.

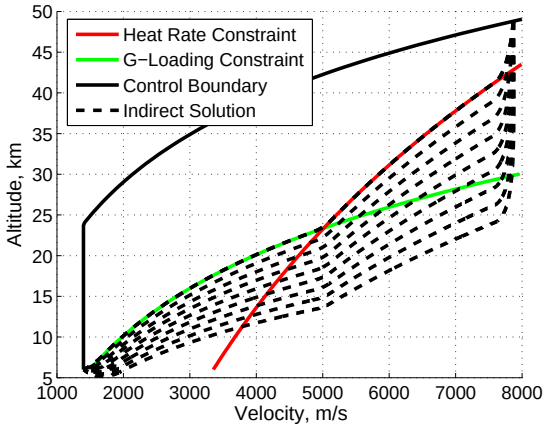


Figure 22. Optimal trajectories for varying atmospheric scale height.

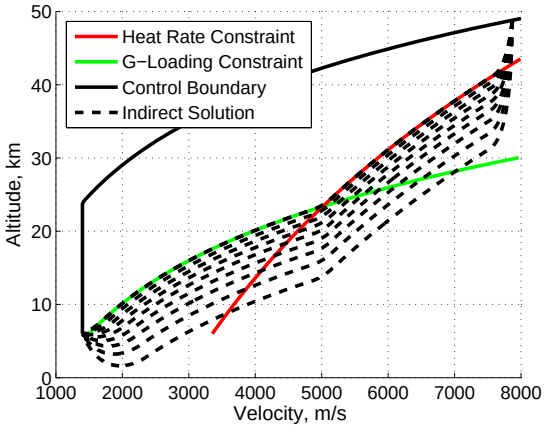


Figure 23. Optimal trajectories for varying atmospheric surface density.

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 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 were varied simultaneously, and the resulting minimum heat load trajectories are shown in Figure 24. Thus, many entry problems are linked and can be rapidly solved using continuation and indirect methods. The extent of entry problems that are linked will be investigated in future work. Specifically, the transformation from an Earth-based, slender entry body, conventional prompt global strike solution to a blunt body, Mars entry solution will be evaluated. Finally, this methodology is not limited to entry problems and could also be applied to other hypersonic flight applications such as aerocapture.

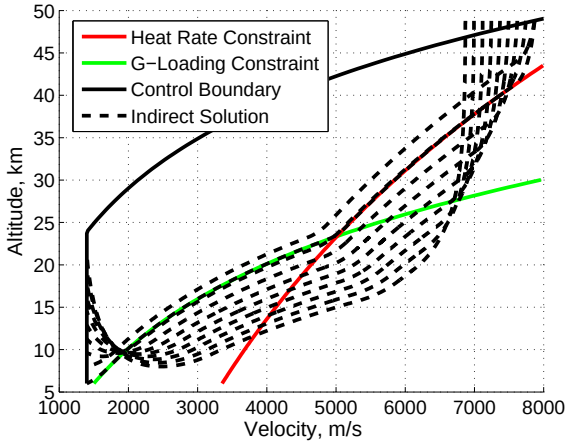


Figure 24. Optimal trajectories from variation of all parameters.

IV. Conclusion

In this investigation, an automated 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 that is a function of velocity, vehicle shape, and celestial body was constructed to improve 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 constructed in minutes for varying trajectory parameters such as initial velocity. Additionally, optimal trades in trajectory performance and vehicle requirements can be performed, and an example optimal trade was shown for heat rate which governs TPS material and heat load which governs TPS mass. Prior advancements in analytic hypersonic aerodynamics enabled the continuation process to expand beyond trajectory parameters to include vehicle shape. The resulting optimal solutions inherently satisfied the equations of motion for each vehicle shape as well as the change in g-loading location throughout the altitude-velocity space. Finally, convergence was demonstrated when atmospheric properties and gravity were varied, validating the hypothesis that many optimal entry trajectories are linked through indirect methods. For example, solutions of slender body Earth-based entry missions can be evolved to solutions of blunt body Mars-based entry missions. Finally, the results of this work can be easily extended to any hypersonic flight application including aerocapture.

V. Acknowledgments

The authors would like to thank Ronald Proulx, Sean George, Gregg Barton, and Linda Fuhrman of the Charles Stark Draper Laboratory for their guidance and support of this research.

References

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

[2]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.

[3]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.

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

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

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

[7]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.

[8]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.

[9]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.

[10]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.

[11]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.

[12]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.

[13]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.

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

[15]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.

[16]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.

[17] "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.

[18] "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.

[19]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.

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

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

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

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

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

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

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

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

[28] "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.

[29]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.

[30]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.

[31]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.

[32]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.