INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.
The effect of lift on aeroassisted orbital transfer trajectories

Boyle, Philip P., M.S.

Rice University, 1992
RICE UNIVERSITY

The Effect of Lift on Aeroassisted Orbital Transfer Trajectories

by

Philip P. Boyle

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE Master of Science

APPROVED, THESIS COMMITTEE:

Angelo Miele, Chairman
A.J. Foyt Family Professor in Engineering
and Professor of Aerospace Sciences
and Mathematical Sciences

Yves Angel
Associate Professor of Mechanical Engineering

Dan C. Sorensen
Professor of Mathematical Sciences

Houston, Texas
May, 1992
Abstract

The Effect of Lift on Aeroassisted Orbital Transfer Trajectories

by

Philip P. Boyle

Miele, Wang, and Deaton have demonstrated in Ref. 1 how optimal trajectories are produced by eliminating positive lift and fixing the lift coefficient at its lower, negative (downward) bound all throughout the atmospheric pass. Because of this fixed control, the problem is posed here as a highly nonlinear two-point boundary-value problem (TPBVP) which is solved by enlisting a multipoint version of the modified quasilinearization algorithm. This thesis supplements Ref. 1 by performing trade studies showing how all the performance indices improve with decreasing lift coefficient lower bound, that is, more negative lift. We also formulate a two-subarc TPBVP for the purpose of testing the effect of a short period of positive lift applied during the descent phase followed by a switch to negative lift. The result is that all performance indices worsen when any positive lift is applied compared to the constant negative lift coefficient case.
Acknowledgments

The author expresses sincere gratitude to his advisor, Dr. A. Miele, for suggesting the topic of this thesis and for providing guidance throughout the ensuing research. Thanks are also due Dr. T. Wang for many helpful discussions and consultations.

The contributions of the other thesis committee members, Dr. Y. Angel and Dr. D. Sorensen, are also gratefully acknowledged.

This research was sponsored by NASA Marshall Space Flight Center Grant No. NAG-8-820 and by Texas Advanced Technology Program Grant No. TATP-003604020.

Dedication

This thesis is dedicated to my wife, Nancy, for her loving support and devoted inspiration both at the beginning and all throughout the pursuance of this degree. Her steadfast and selfless backing helped me weather the hard times and push through to the end.

I also dedicate this thesis to my parents, Mr. Phillip N. Boyle and Mrs. Mary Ann Boyle, for starting me on the right path and for their continued support and encouragement.
Table of Contents

Abstract ........................................................................................................... ii
Acknowledgments ............................................................................................ iii
List of Tables ........................................................................................................ vi
List of Figures ........................................................................................................ vi
Notations ................................................................................................................ vii
1. Introduction ..................................................................................................... 1
2. System Description ......................................................................................... 5
   2.1 Atmospheric Pass ....................................................................................... 6
   2.2 Differential System .................................................................................... 6
   2.3 Control Constraint ...................................................................................... 8
   2.4 Boundary Conditions ................................................................................ 8
   2.5 Summary ..................................................................................................... 9
3. Performance Criteria ....................................................................................... 10
   3.1 Performance Indices .................................................................................. 10
   3.2 Grazing Trajectory .................................................................................... 13
4. TPBVP Formulation ...................................................................................... 15
   4.1 Single Arc .................................................................................................. 15
   4.2 Two Subarcs .............................................................................................. 16
5. Experimental Data ........................................................................................ 20
6. MQA for Nonlinear Multipoint Boundary-Value Problems ............................. 22
   6.1 Multipoint Boundary-Value Problem ....................................................... 22
   6.2 Performance Index .................................................................................... 24
   6.3 Modified Quasilinearization Algorithm .................................................... 24
   6.4 System of Variations ................................................................................ 26
   6.5 Coordinate Transformation ......................................................................... 27
6.6 Method of Particular Solutions (MPS) ........................................... 28
6.7 Determination of the Stepsize ...................................................... 32
6.8 Summary of the Algorithm .......................................................... 33

7. Algorithm Performance .............................................................. 35
  7.1 Nominal Functions ................................................................. 35
  7.2 Programming Details .............................................................. 37
  7.3 Numerical Examples ............................................................... 39
  7.4 Condition Bound .................................................................. 44
  7.5 Grazing Trajectory ................................................................ 55

8. Numerical Results .................................................................... 57
  8.1 Single Arc ....................................................................... 57
  8.2 Two Subarcs ................................................................... 62

9. Conclusions .............................................................................. 66

References ..................................................................................... 68
List of Tables

Table 1. Trajectory parameters, Example 1 ................................................................. 40
Table 2. Grazing trajectory parameters ....................................................................... 56
Table 3. Single-arc trajectory summary ....................................................................... 57
Table 4. Comparison of critical points for two single-arc trajectories ......................... 61
Table 5. Two-subarc trajectory summary .................................................................... 63

List of Figures

Fig. 1. Coplanar aeroassisted orbital transfer .............................................................. 70
Fig. 2. Air density profile, U.S. Standard Atmosphere, 1976 (Ref. 6) ......................... 70
Fig. 3. Two-subarc trajectory definition ...................................................................... 71
Fig. 4. Linear system for method of particular solutions ........................................... 71
Fig. 5. MQA performance, Example 1 ....................................................................... 72
Fig. 6. MPS linear system condition number record ................................................. 72
Fig. 7. Perturbation profiles, Example 1 .................................................................... 73
Fig. 8. Perturbation profiles, Example 3 .................................................................... 74
Fig. 9. Grazing trajectory history ............................................................................... 75
Fig. 10. Single-arc trajectory parameters vs. lift coefficient ...................................... 76
Fig. 11. Single-arc trajectory variable histories, $C_L = -0.9, -0.5, +0.06$ ................ 77
Fig. 12. Two-subarc trajectory parameters vs. entrance flight path angle ................. 78
Notations

\[ a_D = \text{drag acceleration, m/s}^2; \]
\[ a_{IL} = \text{inertial acceleration along lift direction, m/s}^2; \]
\[ a_L = \text{lift acceleration, m/s}^2; \]
\[ C_D = \text{drag coefficient}; \]
\[ C_{D0} = \text{zero-lift drag coefficient}; \]
\[ C_L = \text{lift coefficient}; \]
\[ C_{LL} = \text{lift coefficient lower bound}; \]
\[ C_{LU} = \text{lift coefficient upper bound}; \]
\[ D = \text{drag, N}; \]
\[ g = \text{local acceleration of gravity, m/s}^2; \]
\[ g_D = \text{local acceleration of gravity along drag direction, m/s}^2; \]
\[ g_L = \text{local acceleration of gravity along lift direction, m/s}^2; \]
\[ g_0 = \text{acceleration of gravity at surface of Earth, m/s}^2; \]
\[ h = \text{altitude, m}; \]
\[ HR = \text{heating rate, W/m}^2; \]
\[ L = \text{lift, N}; \]
\[ m = \text{mass, kg}; \]
\[ n = \text{load factor}; \]
\[ q = \text{dynamic pressure, Pa}; \]
\[ r = \text{radial distance from center of Earth, m}; \]
\[ r_e = \text{radius of Earth, m}; \]
\[ S = \text{reference area, m}^2; \]
\[ t = \frac{\theta}{\tau} = \text{dimensionless time}; \]
\[ V = \text{velocity, m/s}; \]
\[ V_a = \text{circular velocity at top of atmosphere, m/s}; \]
\[ W = \text{weight, N}; \]
\[ \beta = \text{negative inverse of atmospheric scale height, } 1/m; \]
\[ \gamma = \text{flight path angle, rad}; \]
\[ \mu = \text{Earth's gravitational constant, } m^3/s^2; \]
\[ \theta = \text{dimensional time, s}; \]
\[ \rho = \text{air density, } kg/m^3; \]
\[ \tau = \text{total atmospheric flight time, s}; \]
\[ \Delta V = \text{characteristic velocity, m/s}. \]

**Subscripts**

\[ a = \text{top of atmosphere}; \]
\[ 00 = \text{GEO exit}; \]
\[ 0 = \text{entry into the atmosphere}; \]
\[ 1 = \text{exit from the atmosphere}; \]
\[ 11 = \text{LEO entry}; \]
\[ * = \text{reference condition}. \]

**Superscript**

\[ \cdot = \text{derivative with respect to dimensionless time}. \]
Acronyms

AOT = aeroassisted orbital transfer;
GEO = geosynchronous Earth orbit;
HEO = high Earth orbit;
IVP = initial-value problem;
LEO = low Earth orbit;
MPS = method of particular solutions;
MQA = modified quasilinearization algorithm;
MQAMP = modified quasilinearization algorithm, multipoint version;
NLTP-BVP = nonlinear two-point boundary-value problem;
ODE = ordinary differential equation;
SGRA = sequential gradient-restoration algorithm;
SI = Système Internationale;
TPBVP = two-point boundary-value problem.
1. **Introduction**

The technology of space travel, both on a systems level and on a component level, has advanced at a tremendous pace from the crude low-Earth-orbiting satellites of the 1950's to the sophisticated interplanetary probes and space shuttles of today. Whereas chemical rocket propulsion has traditionally and almost exclusively supplied all of the force necessary to send the various space vehicles along their way, designers have constantly sought new ways to change a spacecraft’s energy state. Some of these methods involve advanced propulsion systems; for example, nuclear rockets, electric rockets, mass drivers, etc. Some of the more exotic harness “natural forces” present in the surrounding environment (Ref. 2); solar sails, tethered satellites, and planetary gravity assists are a few examples. The method of aeroassisted orbital transfer (AOT) belongs to this latter class.

While some of these methods have been extensively studied and are finding increasing application in space travel (e.g. gravity assist), others have not advanced far beyond the feasibility stage. The field of AOT is somewhere in the middle, with the majority of study just having been done in the last decade, a complete theory still under development, and a flight demonstration yet to be made.

Specifically, aeroassisted orbital transfer involves a spacecraft flying through a planetary atmosphere and using aerodynamic forces to modify its orbit. In this way, reliance on onboard propulsion can be greatly reduced and propellant weight thereby saved. The upshot is that withstanding these aerodynamic forces creates offsetting design requirements which may negate the propellant savings. For example, high levels of dynamic pressure and aerodynamic heating may have to be endured. The goal of the design analyst is to tailor the trajectory in such a way that the most fuel can be saved for the least amount of structural protection.
At this level of optimization study, it is most useful to pare the problem to its most basic elements. In this vein, optimal AOT trajectories have been studied for coplanar, circular orbits about the Earth, since orbital transfer vehicles operating around our home planet are among the first likely candidates to use this technique. Of primary interest is the transfer from geosynchronous Earth orbit (GEO), where important communications and surveillance satellites operate, to low Earth orbit (LEO) — the domain of the Space Shuttle. Changing the plane of the orbit using AOT is also important, but we do not consider it here in order to keep the system equations as simple as possible.

Previous optimal AOT researches have generally found that utilizing aerodynamic lift is preferable to exclusively relying on drag for planar transfers. The reason is that a spacecraft on a descending leg of a transfer orbit from GEO about to enter the atmosphere is falling at a tremendous rate, even though the flight path angle is generally less than $5^\circ$ at entry altitude (120 km). Consequently, to preclude dipping too deeply into the atmosphere, previous studies (Refs. 3–5) have mandated the application of positive, upward lift on the descending branch. The vehicle passes through perigee in a relatively short time after atmospheric entry; therefore, drag has not had much chance to significantly reduce the velocity. Because centrifugal force is still at a high level, the craft would like to skip right back out of the atmosphere. To forestall this and to allow time for drag to act, negative, downward lift is applied during the climbing part of the atmospheric pass.

Miele, Wang, and Deaton in Ref. 1 were the first to challenge this conventional view. They started by considering the grazing trajectory — an ideal path which circumferentially follows the boundary of the atmosphere at 120 km for as long as is needed to deplete the required velocity. It is ideal because the traditional measures of merit for the AOT problem, such as characteristic velocity, peak altitude drop, peak dynamic pressure, peak heating rate, and peak load factor are minimized or nearly minimized for the grazing trajectory. The atmosphere is so thin at 120 km, however, that unrealistically large
lift coefficients are needed to maintain the circular path. Miele, Wang, and Deaton proceeded to note the key point that these lift coefficients are always negative, that is, the lift is always directed radially downward toward the center of the Earth. They reasoned that although no actual craft could achieve these lift coefficients and fly the grazing trajectory, the best approximation might be realized by fixing the lift coefficient at its lowest achievable value.

When the lift coefficient was fixed at its lower bound in Ref. 1, the resulting nearly-grazing trajectory was in fact found to be an improvement over previous optimal trajectories utilizing some positive lift. There, the system equations were solved using the sequential gradient-restoration algorithm (SGRA) for optimal control problems. However, this was done only for computational purposes; the problem was no longer one of optimal control because of the fixed lift coefficient history. It was pointed out in Ref. 1 that the problem reduces to a two-point boundary value problem (TPBVP) — a highly nonlinear one at that. It is the numerical solution of this TPBVP which constitutes a main thrust of this thesis.

Against this backdrop, the thesis is organized as follows. Section 2 describes the AOT system and simplifying assumptions; Section 3 defines several common measures of merit or performance indices and fully describes the grazing trajectory. Section 4 formulates two TPBVPs: the first is the single-arc case of Ref. 1 with fixed lower bound lift coefficient; the second is a two-subarc problem in which maximum positive lift is applied during the initial subarc followed by a discontinuous switch to negative lift for the subsequent subarc. The intent here is to test whether the presence of positive lift can improve any of the performance indices as compared to the single-arc case. Section 5 presents the numerical data used in the solution of these TPBVPs.

In Section 6, a multipoint version of the modified quasilinearization algorithm (MQAMP) is completely developed. Since the standard MQA suffers from divergence when applied to these highly nonlinear TPBVPs, this modification effectively prevents
such occurrence and gives accurate results. Section 7 highlights MQAMP performance by presenting computational details and some preliminary examples which illustrate the unstable nature of the TPBVPs and the utility of MQAMP in solving them.

Finally, in Section 8 we present the result of varying the lift coefficient in the single-arc formulation. We also conduct a trade study using the two-subarc approach which has the effect of changing the duration of positive lift, that is, the duration of the first subarc. In these ways, we obtain numerical evidence which bolsters the claim that fixing the lift coefficient at its lowermost value is optimal. These conclusions are lastly summarized in Section 9.
2. System Description

The AOT vehicle, like the Space Shuttle, is a hybrid aircraft/spacecraft conducting purely propulsive maneuvers in space and utilizing aerodynamic forces in the atmosphere. This study will consider a single specific type of orbital transfer: GEO to LEO. The generic trajectory includes a preatmospheric branch, an atmospheric branch, and a postatmospheric branch, and the terminal orbits are assumed to be circular. All motion is confined to a single plane — there are only two degrees of freedom.

We choose to study the GEO-to-LEO transfer for the following reasons: (1) It is one of the most likely applications of AOT for Earth operations; and (2) the terminal orbits represent a large energy difference; consequently a larger energy dissipation burden is placed on the atmospheric penetration than on that of lower energy HEO-to-LEO or LEO-to-LEO transfers\(^1\). As we will see, a large requirement for velocity depletion implies a deep plunge into the atmosphere where the large air density gradient increases the difficulty of solving the system equations.

We lay out the key points of the maneuver as follows: point 00, GEO exit; point 0, atmospheric entry; point 1, atmospheric exit; point 11, LEO entry. An impulsive retroburn with characteristic velocity \(\Delta V_{00}\) is tangentially applied at point 00, making this the apogee of the preatmospheric transfer orbit \(0 \rightarrow 00\); similarly, a tangential forward impulse with characteristic velocity \(\Delta V_{11}\) is applied to circularize into LEO at point 11, the apogee of the postatmospheric elliptical transfer orbit \(1 \rightarrow 11\). We designate these terminal points to be the only two along the entire trajectory where rocket propulsion is required. Refer to Fig. 1 for a schematic.

\(^1\)HEO = High Earth Orbit.
For simplicity, we idealize the Earth as a uniform, nonrotating sphere surrounded by an atmospheric shell of constant thickness, \(h_a = 120\) km. The air density in the region of applicability (\(h = 50\) km to \(h = 120\) km) essentially varies exponentially with altitude as given by the U.S. Standard Atmosphere, 1976 (Ref. 6), which we utilize for this study. Above 120 km altitude, we arbitrarily set the density to zero. In accordance with our uniform Earth assumption, we use the simple inverse square law for the gravitational force.

2.1 Atmospheric Pass. For this, the most important phase of the transfer, we make the following additional assumptions: (a) we consider translational motion only; (b) the AOT vehicle produces zero net thrust and expends a negligibly small amount of fuel for attitude control in the atmosphere; thus the craft is modelled here as a particle of constant mass; (c) a very simple aerodynamic model is adopted whereby the force coefficients are functions only of the angle of attack—the dependence on Mach and Reynolds numbers and other dimensionless groups is neglected; (d) the two degree of freedom restriction obviates consideration of any side forces due to sideslip; and (e) AOT vehicle control is effected via the angle of attack. Since the lift coefficient \(C_L\) is assumed to be a function only of angle of attack, \(C_L\) is equivalently regarded as the control.

2.2 Differential System. If we define \(\theta\) to be the dimensional time elapsed from atmospheric entry, then a minimum of three variables chosen to comprise the translational state are expressed as dependent functions of this time: altitude \(h(\theta)\), velocity magnitude \(V(\theta)\), and flight path angle \(\gamma(\theta)\). If \(\tau\) is taken to be the total atmospheric flight time from entry to exit, then we have \(0 \leq \theta \leq \tau\). Since \(\tau\) may vary, we normalize the exit time to unity by defining a nondimensional time \(t\) as follows:

\[
t = \frac{\theta}{\tau}.
\] (1)
The state may now be functionally expressed as \( h(t), V(t), \) and \( \gamma(t) \), where \( 0 \leq t \leq 1 \). With this understanding and the assumptions of Section 2.1, the equations of motion appear as (from Ref. 1)

\[
\begin{align*}
\dot{h} &= \tau [ V \sin \gamma ], \\
\dot{V} &= \tau [ -D/m - g \sin \gamma ], \\
\dot{\gamma} &= \tau [ L/m + (V^2/lr - g \cos \gamma) ] / V,
\end{align*}
\]

where the dot superscript refers to differentiation with respect to the nondimensional time \( t \).

To elaborate,

\[ r = r_e + h, \quad g = \mu / r^2, \]

where \( \mu \) is the Earth's gravitational constant. The relevant aerodynamic forces are standardly expressed as

\[ D = C_D q S, \quad L = C_L q S, \quad q = (1/2) \rho V^2, \]

where \( \rho = \rho(h) \) and \( q \) is the dynamic pressure. We also make the simple assumption that the drag polar is parabolic,

\[ C_D = C_{D0} + K C_L^2. \]

Note that \( C_L \) and \( L \) in Eqs.(4) may be plus or minus. By convention, positive lift is rotated 90° upward from the velocity vector, and negative lift is rotated 90° downward. The
upward direction is defined as that direction for which the inner product of the lift vector and the radius vector originating from the Earth's center is positive.

2.3 **Control Constraint.** Hypersonic aircraft are typically limited in the lift coefficient range within which they may operate due to trim and aerodynamic heating design considerations. To this end, we must consider the two-sided inequality constraint

\[ C_{LL} \leq C_L \leq C_{LU}, \quad 0 \leq t \leq 1, \]  

(6)

which must be satisfied within the atmosphere.

2.4 **Boundary Conditions.** These, to be applied at times \( t = 0 \) and \( t = 1 \), are supplied by the atmospheric threshold definition along with the dynamic requirement which constrains the pairs of points \((00, 0)\) and \((11, 1)\) to lie along the same preatmospheric and postatmospheric transfer orbits, respectively.

For any given orbit in a central force field, the angular momentum is constant; so, considering points 00 and 0,

\[ r_{00}V_{00} = r_0V_0\cos \gamma_0. \]  

(7a)

The total energy is also constant,

\[ V_{00}^2/2 - \mu r_{00} = V_0^2/2 - \mu r_a. \]  

(7b)

Solve Eq.(7a) for \( V_{00} \), substitute into Eq.(7b) and rearrange,
\[ r_{00}^2(2V_*^2 - V_0^2) - 2r_{00}r_aV_*^2 + r_a^2V_0^2\cos^2\gamma_0 = 0. \] (8a)

Here, \( r_a \) measures the radius at the top of the atmosphere, and \( V_* = V_a = \sqrt{\mu/r_a} = 7.832 \) km/s is the circular velocity there, which we employ as a reference velocity. Now, by definition,

\[ h_0 = h_a. \] (8b)

In an entirely analogous fashion, we may immediately write the exit boundary conditions,

\[ r_{11}^2(2V_*^2 - V_1^2) - 2r_{11}r_aV_*^2 + r_a^2V_1^2\cos^2\gamma_1 = 0, \] (9a)

\[ h_1 = h_a. \] (9b)

2.5 Summary. Because the preatmospheric and postatmospheric transfer orbits are completely defined once we know the atmospheric entry and exit state, we need only concern ourselves further with the atmospheric portion of the transfer. This part is entirely described by the differential system (2)-(5), the control constraint (6), and the boundary conditions (8)-(9). Here we have set up the dimensionless time \( t \) to be the independent variable along the interval \( 0 \leq t \leq 1 \), with the dependent variables including the state \( h(t) \), \( V(t) \), \( \gamma(t) \), the control \( C_L(t) \), and the parameter \( \tau \).
3. Performance Criteria

3.1 Performance Indices. Different quantities of merit (performance indices) are usually considered in the study of AOT. These parameters are centrally important both from optimal design and feasibility standpoints.

Characteristic Velocity. One of the most basic performance indices in AOT design is the total characteristic velocity associated with the maneuver. This quantity is directly related to the amount of propellant which must be carried onboard, and ultimately the overall vehicle weight. The characteristic velocity is commonly defined as the sum of the impulsive velocity changes at GEO exit and at LEO entry. Symbolically,

$$\Delta V = \Delta V_{00} + \Delta V_{11},$$  \hspace{1cm} (10)

where

$$\Delta V_{00} = \sqrt{(r_{a}/r_{00})V_\bullet} - (r_{a}/r_{00})V_0 \cos \gamma_0,$$  \hspace{1cm} (11a)

$$\Delta V_{11} = \sqrt{(r_{a}/r_{11})V_\bullet} - (r_{a}/r_{11})V_1 \cos \gamma_1.$$  \hspace{1cm} (11b)

If the boundary condition Eq.(8a) is enforced, then $V_0$ can be thought of as a function of $\gamma_0$. Consequently, $\Delta V_{00}$ in Eq.(11a) is entirely a function of $\gamma_0$. Similarly, $\Delta V_{11}$ is a function only of $\gamma_1$.

Minimum Altitude. Figure 2 is a plot of air density vs. altitude as given by the U.S. Standard Atmosphere, 1976 (Ref. 6). The major point to be made here is that the density is roughly exponential with altitude and it changes by 8 orders of magnitude from
the designated 120 km threshold down to the surface. Because of this large gradient and because aerodynamic effects are largely dependent upon air density, a small change in minimum altitude during the atmospheric pass portends significant change in the vehicle structural design requirements.

**Peak Dynamic Pressure.** One of the most important considerations in terms of aerodynamic force is the greatest dynamic pressure to be endured, that is,

\[
q_{\text{max}} = \max_{0 \leq t \leq 1} q(t), \quad \text{(12a)}
\]

\[
q = (1/2) \rho V^2. \quad \text{(12b)}
\]

Because \( V \) generally varies by less than one order of magnitude, \( q_{\text{max}} \) is most affected by the magnitude of the density \( \rho \). See Section 8 for a further discussion of this point.

**Peak Heating Rate.** Another extremely important structural design factor is the maximum aerodynamic heating. The weight savings realized from a low characteristic velocity can be quickly negated by a heavy heat shield if the heating is excessive. We compute the peak heating rate as follows:

\[
HR_{\text{max}} = \max_{0 \leq t \leq 1} HR(t), \quad \text{(13a)}
\]

\[
HR = HR_\star (\rho/\rho_\star)(V/V_\star)^{3.07}. \quad \text{(13b)}
\]

Equation (13b), from Ref. 1, uses reference heating conditions as denoted by the \( \star \) subscript, to compute the heating rate at any other off-reference state. These reference conditions are: the density \( \rho_\star = 3.097 \times 10^{-4} \) kg/m\(^3\) at an altitude \( h_\star = h_d/2 = 60 \) km; the
velocity \( V_\ast = 7.832 \text{ km/s} \) as before; and the resultant heating rate \( HR_\ast = 282.3 \text{ W/cm}^2 \). Based on a nose radius of one foot, Eq. (13b) provides a conservative preliminary design estimate of the heating rate at the stagnation point — typically a location of severe aerodynamic heating on the vehicle. Other AOT researchers have used very similar forms for estimating the stagnation point heating rate (Ref. 2).

Like the maximum dynamic pressure, \( HR_{\text{max}} \) is affected most by changes in altitude. However, the sensitivity is half that of \( q_{\text{max}} \) because the heating rate is proportional to the square root of the density. More on this in Section 8.

**Peak Load Factor.** The greatest amount of aerodynamic force transmitted to the vehicle structure is best summarized by this quantity. It is given by

\[
q_{\text{max}} = \max_{0 \leq t \leq 1} \sqrt{\left( \frac{L}{W} \right)^2 + \left( \frac{D}{W} \right)^2},
\]

where \( L \) and \( D \) are the lift and drag forces, and \( W = m g_0 \) is the sea-level weight. Recalling Eqs. (4)-(5), and assuming for the moment that \( C_L \) stays constant, we find

\[
q_{\text{max}} = C \max_{0 \leq t \leq 1} q(t),
\]

\[
C = \sqrt{(C_L^2 + C_D^2) / (W/S)}.
\]

Hence, the peak load factor occurs at the same point as and is directly proportional to the peak dynamic pressure. This would not be the case if the lift coefficient (angle of attack) were modulated.
3.2 Grazing Trajectory. Optimization studies seek to minimize $\Delta V$ in the most efficient way possible. Previous work (see for example Refs. 1, 5) has recognized that the grazing trajectory, defined by

\begin{align*}
  h(t) &= h_0, \\
  V(t) &\text{satisfies dynamic Eqs.}(2)-(5) \text{ and boundary Eqs.}(8)-(9), \\
  \gamma(t) &= 0,
\end{align*}

is the one which produces the lowest possible $\Delta V$. A spacecraft flying the grazing trajectory skims along the edge of the atmosphere at constant altitude, depleting its velocity until such time as the exit boundary conditions are met. Reference 1 goes on to show that for realistic AOT vehicle designs, the lift coefficient history needed to satisfy Eqs.(16) is unrealistically large in the negative sense. In other words, control inequality constraint (6) is undershot everywhere along the grazing trajectory (see Section 7.5).

As a benchmark, we compute the characteristic velocity components for the grazing trajectory by setting $\gamma_0 = 0$, solving for $V_0$ in Eq.(8a), and substituting into Eq.(11a). A similar approach is employed with $\gamma_1 = 0$, Eqs.(9a) and (11b) to produce

\begin{align*}
  \Delta V_{00} &= V_0 \sqrt{(r_d/r_{00})} - V_0(r_d/r_{00}) \sqrt{[2r_{00}(r_{00} + r_d)]}, \\
  \Delta V_{11} &= V_0 \sqrt{(r_d/r_{11})} - V_0(r_d/r_{11}) \sqrt{[2r_{11}/(r_{11} + r_d)]}.
\end{align*}

Reference 1 proves that the smallest possible $\Delta V$ is that of the grazing trajectory. For all others, the increases of $\Delta V_{00}$ and $\Delta V_{11}$ above that given by Eqs.(17) are respectively proportional to the squares of the entrance and exit flight path angles, for $\gamma_0$ and $\gamma_1$ small (generally less than $5^\circ$).
Reference 1 goes on to observe that separate optimal control problems formulated using each of the performance indices detailed above appear to have similar solutions. That is, a trajectory which optimizes the characteristic velocity also optimizes or nearly optimizes the peak dynamic pressure, the peak heating rate, and the peak altitude drop. One need only consider the grazing trajectory to bolster this observation.

Keeping the above in mind, therefore, the objective of real, flyable trajectories is to produce the smallest possible entrance and exit flight path angles. Consequently, we shall find the grazing trajectory useful as an ideal benchmark.
4. TPBVP Formulation

4.1 Single Arc. Recall Section 3 where the grazing trajectory was asserted to be optimal both from characteristic velocity and minimum altitude standpoints, and probably from peak dynamic pressure and heating rate standpoints as well. The only drawback, and a severe one at that, is the necessary lift coefficient is much too large in the negative sense for a practical design. Realizing this situation, Miele, Wang, and Deaton surmised in Ref. 1 that a next-best alternative to the grazing trajectory satisfying all the dynamic constraints would be produced by freezing the lift coefficient at the lower bound $C_{LL}$. In symbols,

$$C_L(\theta) = C_{LL}, \quad 0 \leq \theta \leq \tau, \quad (18a)$$

or

$$C_L(t) = C_{LL}, \quad 0 \leq t \leq 1. \quad (18b)$$

Enlisting Eqs.(2), (8) and (9), the result is the following differential system:

$$\dot{h} = \tau [ V \sin \gamma ], \quad (19a)$$

$$\dot{V} = \tau [ -D(h,V,C_L)/m - g \sin \gamma ], \quad (19b)$$

$$\dot{\gamma} = \tau [ L(h,V,C_L)/m + (V^2/r - g \cos \gamma )/V, \quad (19c)$$

$$\dot{\tau} = 0, \quad (19d)$$

with initial conditions

$$r_{00}^2(2V_0^* - V_0^2) - 2r_{00}r_{aV_0^*}^2 + r_{a}^2V_0^2\cos^2\gamma_0 = 0, \quad (20a)$$

$$h_0 = h_a, \quad (20b)$$
and final conditions

\begin{align}
  r_{11}^2(2V_*^2 - V_1^2) - 2r_{11}r_aV_*^2 + r_a^2V_1^2\cos^2\gamma_1 &= 0, \quad (21a) \\
  h_1 &= h_a. \quad (21b)
\end{align}

The system (19)-(21) is a two-point boundary-value problem in which the unknowns are \( h(t), V(t), \gamma(t), \) and \( \tau(t) \). Notice the extra differential equation (19d) was added to complete the system and is valid because \( \tau \) is a parameter. This TPBVP is quite obviously nonlinear; a central objective of this thesis is to confirm the existence of a solution via numerical techniques. Assuming this solution can be found, the characteristic velocity is evaluated from Eqs.(10)-(11) and may be compared with that of the grazing trajectory, Eqs.(10) and (17). We shall also test the effect on the various performance indices of increasing the lift coefficient; if in fact the trajectory produced with \( C_L = -0.9 \) is the best approximation to the grazing trajectory, then those produced with greater (less negative) lift coefficients should feature less favorable performance indices.

4.2 Two Subarcs. Previous optimal AOT studies have generally found that a short period of positive lift is required on the descending branch of the atmospheric pass followed by a relatively quick transition to negative lift during the much longer duration ascending branch (see, for example, Refs. 3, 4, and 5). Furthermore, if there are inequality constraints of the form (6), the positive lift coefficient begins at the upper bound \( C_{LU} \) and the negative lift coefficient is pegged at the lower bound \( C_{LL} \). The transition between the two extremes usually takes place in a short amount of time compared to the duration of the entire pass.
This result may be idealized by dividing the trajectory into two subarcs, as shown by Fig. 3. Here, the first subarc is defined by a constant lift coefficient at the upper bound of duration $\tau_1$. The second subarc spans the time interval $\tau_2$ and is characterized by a constant lift coefficient at the lower bound. This definition is summarized by

\begin{align}
C_L(\theta) &= C_{LU}, & 0 \leq \theta \leq \eta_1, & \text{(subarc 1), } (22a) \\
C_L(\theta) &= C_{LL}, & \eta_1 \leq \theta \leq \tau, & \text{(subarc 2), } (22b) \\
\tau &= \tau_1 + \tau_2. & & \text{(22c)}
\end{align}

Note that the control $C_L(\theta)$ is discontinuous at $\theta = \eta_1$, the junction between subarcs. We again may transform the independent variable time domain such that the duration of each subarc is normalized to unity. This is done by means of the following equations:

\begin{align}
\theta &= \tau_1 t, & 0 \leq t \leq 1, & \text{(subarc 1), } (23a) \\
\theta &= \tau_1 + \tau_2 t, & 0 \leq t \leq 1, & \text{(subarc 2), } (23b)
\end{align}

which imply

\begin{align}
\dot{\theta} &= \tau_1, & 0 \leq t \leq 1, & \text{(subarc 1), } (24a) \\
\dot{\theta} &= \tau_2, & 0 \leq t \leq 1, & \text{(subarc 2). } (24b)
\end{align}

Reconsider momentarily the single-arc system Eqs.(19)-(21). Define the state vector as

$$x(t) = [ h(t), \ V(t), \ \gamma(t), \ \pi(t) ]^T. \quad (25)$$

Then the TPBVP (19)-(21) may be compactly written as
\[ \dot{x}(t) = \phi(x(t)), \quad 0 \leq t \leq 1, \quad (26a) \]
\[ \Omega(x(0), x(1)) = 0, \quad (26b) \]

where \( \phi(x(t)) \) is a vector function of dimension four consisting of the right-hand sides of Eqs. (19), and \( \Omega(x(0), x(1)) \) is the vector derived from boundary Eqs. (20)-(21), namely\(^2\)

\[
\Omega(x(0), x(1)) = \begin{bmatrix}
  r_{00}^2[2V_0^2 - V(0)^2] - 2r_{00}r_aV_0^2 + r_a^2V(0)^2\cos^2\gamma(0) \\
  h(0) - h_a \\
  r_{11}^2[2V_1^2 - V(1)^2] - 2r_{11}r_aV_1^2 + r_a^2V(1)^2\cos^2\gamma(1) \\
  h(1) - h_a
\end{bmatrix}.
\quad (27)\]

To handle the two-subarc case, we merely define the augmented state vector

\[ \dot{\tilde{x}}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}, \quad 0 \leq t \leq 1, \quad (28) \]

where the 1 subscript denotes the state vector in subarc 1 \((0 \leq \theta \leq \tau_1)\), and the 2 subscript refers to the state vector in subarc 2 \((\tau_1 \leq \theta \leq \tau)\). Both \( x_1(t) \) and \( x_2(t) \) must of course satisfy the differential constraint Eq. (26a). The augmented state vector is now of dimension eight; we therefore require eight boundary conditions for a potential solution. Four are provided by Eq. (27); three more arise from enforcing continuity of the altitude, velocity, and flight path angle at the subarc interface. We are thus free to choose the remaining boundary

---
\(^2\)Note we have changed notation slightly: \( h_0, V_0, \gamma_0, h_1, V_1, \gamma_1 \), from Eqs. (20)-(21) have become \( h(0), V(0), \gamma(0), h(1), V(1), \gamma(1) \). The reason is to avoid conflicting subscript notation with the two-subarc equations to follow.
condition. Here we simply fix the entrance flight path angle at an arbitrary value \( \gamma_0 \) to round out the vector of boundary conditions as follows:

\[
\Omega(\hat{x}(0), \hat{x}(1)) = 
\begin{bmatrix}
 r_{00}^2[2V_*^2 - V_1(0)^2] - 2r_{00}r_aV_*^2 + r_a^2V_1(0)^2 \cos^2 \gamma_1(0) \\
 h_1(0) - h_a \\
 r_{11}^2[2V_*^2 - V_2(1)^2] - 2r_{11}r_aV_*^2 + r_a^2V_2(1)^2 \cos^2 \gamma_2(1) \\
 h_2(1) - h_a \\
 h_1(1) - h_2(0) \\
 V_1(1) - V_2(0) \\
 \gamma_1(1) - \gamma_2(0) \\
 \gamma_1(0) - \gamma_0
\end{bmatrix}
\]  

(29)

Note that the boundary conditions of the single-arc problem Eq.(27) are separable into initial conditions (the first two components) and final conditions (the last two). This is no longer possible in the two-subarc formulation due to the mixed continuity conditions, components \( \Omega_5, \Omega_6, \Omega_7 \) of the vector (29).

We may investigate different two-subarc trajectories by varying the initial flight path angle \( \gamma_0 \). This freedom will allow us to determine how long positive lift must be applied as a function of the entrance conditions. To assess their relative merit, we also may compare the performance indices of the single-arc and two-subarc solutions.
5. Experimental Data

In this section, numerical values are given for various parameters found in the system equations.

Physical Constants. For our purposes, the Earth is idealized as a nonrotating, uniform sphere surrounded by an atmospheric shell of constant thickness. The Earth radius is taken to be \( r_e = 6378 \) km; the radius of the outer boundary of the atmosphere is \( r_a = 6498 \) km, implying an atmospheric thickness of \( h_a = 120 \) km; the gravitational constant is \( \mu = 398600 \text{ km}^3/\text{s}^2 \).

Atmospheric Model. Air density \( \rho \) is found as a function of altitude by means of a lookup table provided by the U.S. Standard Atmosphere, 1976 (Ref. 6). For altitudes between those of the table, the density is interpolated assuming an exponential variation with altitude. Above \( h_a = 120 \) km, \( \rho \) is taken to be zero. For the altitude region of interest (50 km to 120 km), the overall functional dependence of \( \rho \) on \( h \) is roughly exponential (see Fig. 2).

Transfer Boundaries. We restrict our study to one of the most likely Earth applications of AOT: the GEO-to-LEO transfer. This is also one of the more difficult problems to numerically solve due to the high speed at entry and the resultant deep plunge into the atmosphere required to satisfy the boundary conditions. The GEO radius is \( r_{00} = 42164 \) km, which expressed in units of atmospheric radii is \( r_{00}/r_a = 6.4888 \). LEO radius is \( r_{11} = 6558 \) km — only 60 km beyond the assumed outer edge of the atmosphere — this corresponds to \( r_{11}/r_a = 1.0092 \). Note again the initial and final orbits are circular.

Spacecraft. The physical and aerodynamic properties were taken from Ref. 5. The mass per unit reference area is \( m/S = 300 \) kg/m\(^2\). A parabolic drag polar is assumed as given in Eq.(5) with the zero-lift drag \( C_{D0} = 0.1 \), and the induced drag factor \( K = 1.11 \). Inequality (6) defines the allowable range for the lift coefficient, whereby the lower bound
is $C_{LL} = -0.9$ and the upper bound is $C_{LU} = +0.9$. Maximum $L/D$ is 1.5 and occurs at a lift coefficient of $C_L = 0.3$. 
6. MQA for Nonlinear Multipoint Boundary-Value Problems

This algorithm was originally outlined in Ref. 7. In that work, a multipoint approach was devised for solving highly nonlinear two-point boundary-value problems (see Refs. 8-11 for related documentation). In this section, we generalize by first deriving the algorithm for a multipoint boundary-value problem. Because TPBVPs are a subclass of multipoint boundary-value problems, the solution of the former using an algorithm developed for the latter is straightforward.

6.1 Multipoint Boundary-Value Problem. Consider the partition

\[ \{ t_0 = t_a, t_1, t_2, \ldots, t_m = t_b \}, \quad t_0 < t_1 < \ldots < t_m, \]  

(30)

and the nonlinear differential equation

\[ \dot{x}(t) = \phi(x(t)), \quad t_a \leq t \leq t_b, \]  

(31)

where \( t \) is the scalar independent variable, \( x \) is an \( n \)-vector of dependent variables, and the \( n \)-vector \( \phi \) is a function of \( x \). Add to Eq.(31) the boundary conditions\(^3\)

\[ \Omega( x(t_0), x(t_1), x(t_2), \ldots, x(t_m) ) = 0, \]  

(32)

\(^3\)Note the label "boundary conditions" is somewhat of a misnomer. Since the differential equation is defined over the whole interval \([t_a, t_b]\), the functional dependence of (32) on the interior nodes \( t_1, t_2, \ldots, t_{m-1} \) as well as the true boundary points \( t_a \) and \( t_b \) might plausibly support a change of terminology.
where $\Omega$ is an $n$-vector. Our problem thus is posed as the search for the function $x(t)$ (assuming it exists) on the interval $[t_a,t_b]$ which satisfies Eq. (31), subject to the boundary conditions (32). Defining the $i$th subinterval as $[t_{i-1},t_i]$, the function $x(t)$ on $[t_a,t_b]$ is correspondingly partitioned by the following convention:

$$x(t) = \bigcup_{i=1}^{m} x_i(t), \quad (33)$$

where it is understood the domain of each $x_i(t)$ is $[t_{i-1},t_i]$. To represent a solution, these component arcs must satisfy the $m$ differential equations

$$\dot{x}_i(t) = \phi(x_i(t)), \quad t_{i-1} \leq t \leq t_i, \quad 1 \leq i \leq m, \quad (34)$$

along with the boundary conditions

$$\Omega( x_1(t_0), x_2(t_1), x_3(t_2), ..., x_m(t_{m-1}), x_m(t_m) ) = 0. \quad (35)$$

To enforce continuity of the state at the interior nodes of the partition (30), we also require $m-1$ interface conditions,

$$x_i(t_i) = x_{i+1}(t_i), \quad 1 \leq i \leq m-1. \quad (36)$$

To recap, arriving at the solution $x(t)$ on $[t_a,t_b]$ to the differential system Eqs. (31)-(32) is equivalent to uncovering the functions $x_i(t)$ which satisfy Eqs. (34)-(36).
6.2 Performance Index. Owing to the nonlinear nature of the differential system, numerical methods must be employed to solve the general problem posed by Eqs.(31)-(32). Regardless of the procedure used, a scalar performance index or norm \( P \) can be defined which provides a measure of the accuracy of the solution. This index is defined with the properties that \( P = 0 \) at the solution and \( P > 0 \) away from the solution.

Using the differential equations, boundary conditions, and interface conditions, the performance index \( P \) is defined as

\[
P = \sum_{i=1}^{m} \int_{t_{i-1}}^{t_i} [\dot{x}_i(t) - \phi(x_i(t))]^T [\dot{x}_i(t) - \phi(x_i(t))] dt + \Omega^T \Omega
+ \sum_{i=1}^{m-1} [x_{i+1}(t_i) - x_i(t_i)]^T [x_{i+1}(t_i) - x_i(t_i)]. \tag{37}
\]

The relative size of \( P \) can be utilized as a gauge during the course of an iterative algorithm as well as at the end as a convergence test. Since computers are limited to finite precision, a practical stopping condition representing convergence is

\[
P \leq \varepsilon, \tag{38}
\]

where \( \varepsilon \) is a small number appropriate to the machine precision.

6.3 Modified Quasilinearization Algorithm. An iterative algorithm featuring a descent property in \( P \) is attractive from the standpoint of stability. We can achieve this by modifying the standard quasilinearization algorithm (Newton's method) in a particular way (see Refs. 10-11 for the original developments). Start by considering the nominal functions \( x_i(t) \) and varied functions \( \tilde{x}_i(t) \) such that
\( \tilde{x}_i(t) = x_i(t) + \Delta x_i(t), \quad t_{i-1} \leq t \leq t_i, \quad 1 \leq i \leq m. \) (39)

Here, the terms \( \Delta x_i(t) \) represent changes in the nominal functions \( x_i(t) \) over their domains which correspondingly cause a change in the performance index from its nominal value \( P \) to its perturbed value \( \tilde{P} \). This resultant change may be expressed in a Taylor series expansion as follows:

\[ \tilde{P} = P + \delta P + \text{higher-order terms}. \] (40)

The first variation \( \delta P \) is the linear part of the perturbation and may be given as

\[
\delta P = 2 \sum_{i=1}^{m} \int_{t_{i-1}}^{t_i} [\dot{x}_i(t) - \phi(x_i(t))] \dot{\delta}[\dot{x}_i(t) - \phi(x_i(t))] dt + 2 \Omega^T \delta \Omega \\
+ 2 \sum_{i=1}^{m-1} [x_{i+1}(t_i) - x_i(t_i)] \dot{\delta}[x_{i+1}(t_i) - x_i(t_i)].
\] (41)

If we make the first variations of the vectors on the right-hand side of Eq. (41) antiparallel to those vectors, we obtain the following:

\[
\delta[ \dot{x}_i(t) - \phi(x_i(t))] = -\alpha[ \dot{x}_i(t) - \phi(x_i(t))], \quad t_{i-1} \leq t \leq t_i, \quad 1 \leq i \leq m, \quad (42a)
\]

\[
\delta \Omega = -\alpha \Omega, \quad (42b)
\]

\[
\delta[ x_{i+1}(t_i) - x_i(t_i) ] = -\alpha[ x_{i+1}(t_i) - x_i(t_i) ], \quad 1 \leq i \leq m, \quad (42c)
\]

where \( \alpha \) is a nonnegative scalar. Substituting Eqs. (42) into (41) and recalling the definition (37) produces
\[ \delta P = -2\alpha P. \] (43)

Now, since \( P \) is positive for all nominal \( x_i(t) \) not satisfying Eqs.(34)-(36), and restricting \( \alpha \) to be positive, Eq.(43) insures that

\[ \delta P < 0. \] (44)

Thus, by virtue of Eqs.(42), the requisite descent property in \( P \) has been established. Provided \( \alpha \) is sufficiently small, the first variation \( \delta P \) is the dominant component in the total variation of \( P \), and we can guarantee

\[ \bar{P} < P \] (45)

whenever \( x(t) \) changes to \( \bar{x}(t) \) over \([t_a,t_b] \).

### 6.4 System of Variations.

The first variations referred to in Eqs. (41)-(42) are linear in the perturbations \( \Delta x_i(t) \). Proceeding, we obtain

\[ \delta[ \dot{x}_i(t) - \phi(x_i(t)) ] = \Delta \dot{x}_i(t) - \phi_i^T(x_i(t))\Delta x_i(t), \quad t_{i-1} \leq t \leq t_i, \quad 1 \leq i \leq m, \] (46a)

\[ \delta \Omega = \Omega_0^T \Delta x_1(t_0) + \Omega_1^T \Delta x_2(t_1) + \cdots + \Omega_{m-1}^T \Delta x_m(t_{m-1}) + \Omega_m^T \Delta x_m(t_m), \] (46b)

\[ \delta[ x_{i+1}(t_i) - x_i(t_i) ] = \Delta x_{i+1}(t_i) - \Delta x_i(t_i), \quad 1 \leq i \leq m-1, \] (46c)

where the matrices \( \phi_i \) and \( \Omega_{x_i} \), \( 0 \leq i \leq m \), are of dimension \( n \times n \). Substituting these equations into Eqs.(42) then gives
\[ \Delta \dot{x}_i(t) - \varphi_i^T(x_i(t)) \Delta x_i(t) = -\alpha [ \dot{x}_i(t) - \phi(x_i(t)) ], \quad t_{i-1} \leq t \leq t_i, \quad 1 \leq i \leq m, \quad (47a) \]

\[ \Omega_{x_0}^T \Delta x_1(t_0) + \Omega_{x_1}^T \Delta x_2(t_1) + \cdots + \Omega_{x_{m-1}}^T \Delta x_{m-1}(t_{m-1}) + \Omega_{x_m}^T \Delta x_m(t_m) = -\alpha \Omega, \quad (47b) \]

\[ \Delta x_{i+1}(t_i) - \Delta x_i(t_i) = -\alpha [ x_{i+1}(t_i) - x_i(t_i) ], \quad 1 \leq i \leq m-1. \quad (47c) \]

If we consider the stepsize \( \alpha \) fixed for the moment, Eq. (47a) represents \( mn \) scalar differential equations while Eqs. (47b)-(47c) total \( mn \) scalar boundary conditions. Thus, the original nonlinear multipoint boundary-value problem in component arc form transcribed in Eqs. (34)-(36) has been replaced by a multipoint boundary-value problem (47) linear in the perturbations \( \Delta x_i(t) \). Solving for these perturbations allows us to adjust the nominal functions \( x_i(t) \) via Eqs. (39), thereby lowering the performance index \( P \) and approaching the solution to our original problem. This is the essence of the modified quasilinearization algorithm (MQA).

Taking \( \alpha = 1 \), Eqs. (47) are the same as those obtained from ordinary quasilinearization (Newton's method). This algorithm suffers from the potentially unstable possibility that \( P \) may increase over an iteration, particularly when starting from a nominal condition far from the solution (i.e. nominal \( P \) large). In contrast, the freedom to choose \( \alpha \) such that condition (45) is always satisfied is the key stabilizing feature of MQA. Nevertheless, the use of any arbitrary starting function does not guarantee MQA will converge; the start-up function must be close to the solution to enable convergence.

**6.5 Coordinate Transformation.** Scale the perturbations \( \Delta x_i(t) \) by the stepsize \( \alpha \) as follows:

\[ \Delta x_i(t) = \alpha A_i(t), \quad t_i-1 \leq t \leq t_i, \quad 1 \leq i \leq m, \quad (48) \]

substitute into Eqs. (47) and cancel the scalar \( \alpha \) to obtain
\[ \dot{A}_{i}(t) - \phi^{T}_{i}(x_{i}(t))A_{i}(t) = -[ \ddot{x}_{i}(t) - \phi(x_{i}(t)) ], \quad t_{i-1} \leq t \leq t_{i}, \quad 1 \leq i \leq m, \quad (49a) \]

\[ \Omega_{x_{1}}^{T}A_{1}(t_{0}) + \Omega_{x_{2}}^{T}A_{2}(t_{1}) + \cdots + \Omega_{x_{m-1}}^{T}A_{m}(t_{m-1}) + \Omega_{x_{m}}^{T}A_{m}(t_{m}) = -\Omega, \quad (49b) \]

\[ A_{i+1}(t_{i}) - A_{i}(t_{i}) = -[ x_{i+1}(t_{i}) - x_{i}(t_{i}) ], \quad 1 \leq i \leq m-1. \quad (49c) \]

The above multipoint boundary-value problem is linear in the perturbations per unit stepsize \( A_{i}(t) \). Solving for these scaled vectors allows us to subsequently choose an acceptable value of \( \alpha \) and calculate the true variations \( \Delta x_{i}(t) \) via (48) and the varied functions via (39) such that condition (45) is satisfied.

6.6 Method of Particular Solutions (MPS). Consider first the linear differential equations (49a) devoid of their boundary conditions and assume the general solutions \( A_{i}(t) \) may be expressed in the following form:

\[ A_{i}(t) = \tilde{A}_{i}(t)k_{i}, \quad t_{i-1} \leq t \leq t_{i}, \quad 1 \leq i \leq m, \quad (50) \]

where \( k_{i} \) are \((n+1)\)-vectors and

\[ \tilde{A}_{i}(t) = [ A_{i1}(t), A_{i2}(t), \ldots, A_{in+1}(t) ], \quad t_{i-1} \leq t \leq t_{i}, \quad 1 \leq i \leq m, \quad (51) \]

is the \( n \times (n+1) \) particular solution matrix function over each component interval \([t_{i-1}, t_{i}]\). Each column of (51) represents a particular solution to a differential equation in Eq.(49a); see Refs. 9-11.

One way to generate the particular solution matrix is to choose the following initial conditions:
\[ \tilde{A}_i(t_{i-1}) = [I_n, \ 0], \quad 1 \leq i \leq m, \quad (52) \]

where \( I_n \) is the \( n \times n \) identity matrix. Eq.(52) in concert with Eq.(49a) represents \( n+1 \) initial-value problems which are easily solved using standard numerical techniques.

To isolate the unique solution (for each interval) which satisfies the boundary conditions (49b)-(49c), we need merely substitute the general solution (50) into Eqs.(49b)-(49c) and solve for \( k_i \),

\[
\Omega_{x_0}^T \tilde{A}_1(t_0)k_1 + \Omega_{x_1}^T \tilde{A}_2(t_1)k_2 + \cdots + \Omega_{x_{m-1}}^T \tilde{A}_m(t_{m-1})k_m + \Omega_{x_m}^T \tilde{A}_m(t_m)k_m = -\Omega, \quad (53)
\]

\[
\tilde{A}_{i+1}(t_i)k_{i+1} - \tilde{A}_i(t_i)k_i = -[x_{i+1}(t_i) - x_i(t_i)], \quad 1 \leq i \leq m-1. \quad (54)
\]

Equation (53) represents \( n \) scalar equations and Eq.(54) contains \( n(m-1) \) for a total of \( nm \) overall. However, the \( k_i \) scalar components number \( m(n+1) \); therefore, linear system (53)-(54) is underdetermined. To make the \( k_i \) unique, we generate the missing \( m \) equations by substituting Eq.(50) into (49a) as follows:

\[
\dot{A}_i(t)k_i - \phi^T_i(x_i(t))\tilde{A}_i(t)k_i = -[\dot{x}_i(t) - \phi(x_i(t))], \quad t_{i-1} \leq t \leq t_i, \quad 1 \leq i \leq m. \quad (55)
\]

Because each of the columns of (51) satisfies the differential equation (49a), Eq.(55) reduces to

\[
\begin{bmatrix}
-[\dot{x}_i(t) - \phi(x_i(t))] & \cdots & -[\dot{x}_i(t) - \phi(x_i(t))]
\end{bmatrix}k_i = -[\dot{x}_i(t) - \phi(x_i(t))],
\]

\[
t_{i-1} \leq t \leq t_i, \quad 1 \leq i \leq m,
\]

or

\[
U_{n+1}^T k_i = 1, \quad 1 \leq i \leq m, \quad (56)
\]
where

\[ U_{n+1}^T = [1, 1, 1, \ldots, 1] \]  \hspace{1cm} (57)  

is an \((n+1)\)-vector of ones. We have thus derived the full square linear system of equations, embodied in Eqs. (53), (54) and (56), which when solved for the \(k_i\) and substituted into Eq. (50) produces the solution to the linear two-point boundary-value problem (49).

A further simplification of the linear system is achieved when we substitute the particular solution initial conditions (52) into Eqs. (53)-(54). The system then reduces to

\[
\begin{align*}
U_{n+1}^T k_i &= 1, \\
[\Omega_{x_0}^T, 0] k_1 + [\Omega_{x_1}^T, 0] k_2 + \cdots + ([\Omega_{x_{m-1}}^T, 0] + \Omega_{x_m}^T \tilde{A}_m(t_m)) k_m &= -\Omega, \\
\tilde{A}_f(t_i) k_i - [I_n, 0] k_{i+1} &= x_{i+1}(t_i) - x_i(t_i), \hspace{1cm} 1 \leq i \leq m - 1.
\end{align*} \hspace{1cm} (58a, 58b, 58c)
\]

The above may be recast as follows

\[ \Lambda k = b, \]  \hspace{1cm} (58d)  

where

\[ k = \begin{bmatrix} k_1 \\ k_2 \\ \vdots \\ k_m \end{bmatrix} \]  \hspace{1cm} (58e)
is the unknown vector, \( \Lambda \) is the coefficient matrix, and \( b \) is the right-hand side. \( \Lambda \) is an \( m(n+1) \times m(n+1) \) matrix, while \( k \) and \( b \) are \( m(n+1) \)-vectors. Linear system (58) is depicted in Fig. 4 for \( m = 4 \). Note the large areas of zeroes: as \( m \) grows large, block algorithms which take advantage of this structure to reduce the storage and processing requirements are highly desirable\(^4\).

As a final note to this section, we observe that actually performing the matrix multiplication suggested by Eq.(50) to generate the solutions \( A_i(t) \) to system (49) entails a very large amount of storage, since all \( \tilde{A}_i(t) \) must be memorized over each of the component domains. An attractive alternative is realized by evaluating Eq.(50) at the fixed time station \( t = t_{i-1} \) and substituting Eq.(52),

\(^4\)Linear system (58) is not the only one derivable. By virtue of the interface conditions (36), the boundary conditions (35) might equivalently have been written

\[ \Omega(x_1(t_0), x_1(t_1), \ldots, x_m(t_m)) = 0. \]

The linearization and scaling processes would then have produced [compare with Eq.(49b)]

\[ \Omega^T_{A_1} A_1(t_0) + \Omega^T_{A_1} A_1(t_1) + \ldots + \Omega^T_{A_m} A_m(t_m) = -\Omega. \]

Now with the initial condition choice (52), the simplification (58b) is no longer possible (except for the first term). However, if we shift the initial conditions to the right as follows,

\[ \tilde{A}_i(t_i) = [I_m, 0], \quad 1 \leq i \leq m, \]

and solve the differential equations (49a) backward from \( i \) to \( i-1 \), forming a different set of solutions \( \tilde{A}_i(t) \), then the linear system appears as

\[ U_{n+1}^T k_i = 1, \quad 1 \leq i \leq m, \]

\[ (\Omega^T_{x_1} A_1(t_0) + [\Omega^T_{x_1}, 0]) k_1 + [\Omega^T_{x_2}, 0] k_2 + \ldots + [\Omega^T_{x_m}, 0] k_m = -\Omega, \]

\[ \tilde{A}_{i+1}(t_i) k_{i+1} - [I_m, 0] k_i = -[x_{i+1}(t_i) - x_i(t_i)], \quad 1 \leq i \leq m-1. \]

Uniqueness demands that the solution \( A_{i}(t) \) generated from the above linear system and Eq.(50) be identical to that obtained from system (58) and Eq.(50).
where \( \hat{k}_i \) denotes the first \( n \) components of each \( k_i \). Numerically solving one more initial-value problem described by Eqs.(49a) and (59) must produce the same solution (50), since it is unique. Consequently, storage is traded for processing; the nature of the problem and the computer attributes are usually the deciding factors as to the preference of this approach.

6.7 Determination of the Stepsize. Putting Eqs.(39) and (48) together produces the following:

\[
\tilde{x}_i(t) = x_i(t) + \alpha A_i(t), \quad t_{i-1} \leq t \leq t_i, \quad 1 \leq i \leq m. \tag{60}
\]

Since the nominal functions \( x_i(t) \) are given and the variations per unit stepsize \( A_i(t) \) are in hand thanks to MPS, the varied functions \( \tilde{x}_i(t) \) are strictly dependent on the stepsize \( \alpha \). As a result, the performance index may be written as a scalar function of \( \alpha \), \( \tilde{P}(\alpha) \). In accordance with the demonstrated descent property, the slope of this function is negative at \( \alpha = 0 \) as follows:

\[
\tilde{P}_\alpha(\alpha) = -2 \tilde{P}(0). \tag{61}
\]

Because \( \tilde{P}(\alpha) \) is a nonnegative function [recall definition (37)], \( \tilde{P}(\alpha) \to +\infty \) as \( \alpha \to +\infty \); therefore, this fact combined with Eq.(61) and the continuity assumption mandates the existence of an absolute minimum for \( \tilde{P}(\alpha) \) at some \( \alpha \geq 0 \). We could search for this minimum, but since we are already within an iterative loop owing to the linearization of the
original boundary-value problem, there is little to be gained at the expense of additional computer time. Thus, the preferred approach is to first restrict the stepsize to the range

\[ 0 \leq \alpha \leq 1. \]  \hspace{1cm} (62)

Then we first assign the value

\[ \alpha = 1, \]  \hspace{1cm} (63)

which as previously pointed out corresponds to Newton's method and provides one-step convergence should the original system (34)-(36) happen to be linear. As the goal is the reduction of the performance index, stepsize (63) is acceptable only if

\[ \bar{P}(\alpha) < \bar{P}(0). \]  \hspace{1cm} (64)

If this is not the case, \( \alpha \) must be reduced via a bisection process until Ineq.(64) is satisfied. We are sure to find a suitable \( \alpha \) as a result of the descent property (45).

6.8 Summary of the Algorithm. The modified quasilinearization algorithm for multipoint boundary-value problems is outlined as follows:

(a) Partition the interval of integration into \( m \) subintervals, as in Eq.(30).
(b) Cast nominal functions \( x_i(t), \ 0 \leq t \leq t_i, \ 1 \leq i \leq m. \)
(c) For each subdomain, compute the vectors \( \hat{x}_i(t) - \phi(x_i(t)) \) and the matrices \( \phi_x(x_i(t)) \).
(d) Compute the vector boundary condition \( \Omega(x_1(t_0), x_2(t_1), x_3(t_2), \ldots, x_m(t_{m-1}), x_m(t_m)) \) and its matrix derivatives \( \Omega_{x_0}, \Omega_{x_1}, \ldots, \Omega_{x_m} \).

(e) Compute the vectors \( x_{i+1}(t_i) - x_i(t_i) \) for adjoining subintervals.

(f) Solve the linear multipoint boundary-value problem (58) for the functions \( A_i(t_i), t_{i-1} \leq t \leq t_i, \ 1 \leq i \leq m, \) by means of the method of particular solutions as outlined in Section 6.6.

(g) Consider the one-parameter family of varied functions (60) and perform a bisectional search on the scalar function \( \tilde{P}(\alpha) \) using the sequence \( \alpha = \{1, 1/2, 1/4, 1/8, \ldots\} \) until Ineq.(64) is satisfied.

(h) The varied functions \( \tilde{x}_i(t) \) will be in hand after step (g), completing the iteration. The varied functions \( \tilde{x}_i(t) \) (and their dependent quantities, e.g., performance index \( \tilde{P} \)) in turn become the nominal functions \( x_i(t) \) for the next iteration.

(i) Terminate the algorithm if stopping condition (38) is satisfied; otherwise loop to step (c) and continue iterating.
7. Algorithm Performance

The algorithm described in the previous section is not only useful for nonlinear differential equations with multipoint boundary conditions of functional form (32). It can also be utilized for — in fact its motivation for development was to handle — nonlinear two-point boundary-value problems (NLTP-BVP) whose Jacobian matrix features large norms along the interval of integration $[t_{n}, t_{b}]$. This inherent instability causes difficulty for numerical methods; in our case, $A(t)$ diverges over $[t_{n}, t_{b}]$. The multipoint algorithm of Section 6 circumvents this problem by providing the freedom to divide this interval into an arbitrary number of subintervals. In this way, the component functions

$$A_i(t), \quad t_{i-1} \leq t \leq t_i, \quad 1 \leq i \leq m, \quad (65)$$

while still subject to the influence of large Jacobian norms, may be integrated over arbitrarily small subintervals and excessive divergence is forestalled.

Note that for NLTP-BVPs, the boundary conditions may be stated as

$$\Omega(x(t_0), x(t_m)) = 0, \quad (66)$$

and in lieu of Eq.(49b), the linearized boundary conditions become

$$\Omega_{x_0} A_1(t_0) + \Omega_{x_m} A_m(t_m) = -\Omega. \quad (67)$$

7.1 Nominal Functions. To demonstrate multipoint MQA performance and to lay the groundwork for subsequent study, we consider the single-arc formulation, Eqs.(19)-(21), of the AOT system equations. This system is highly nonlinear due to the
exponential variation of air density. We employ MQAMP using two sets of nominal functions.

**Nominal Functions A.** The first set was inspired by the grazing trajectory [compare with Eqs.(16)] and is given as follows:

\[
\begin{align*}
    h(t) &= h_a, \quad \text{(68a)} \\
    V(t) &= V_0 + (V_1 - V_0)t, \quad \text{(68b)} \\
    \gamma(t) &= 0, \quad \text{(68c)} \\
    \tau(t) &= 2\pi a / (V_0 + V_1), \quad 0 \leq t \leq 1, \quad \text{(68d)}
\end{align*}
\]

where \( V_0 \) and \( V_1 \) were computed such that the boundary conditions (20)-(21) are satisfied for these nominal functions. The term \( V_0 \) in Eqs.(68) represents the velocity at point 0 resulting from a Hohmann transfer from point 00 to point 0 and \( V_1 \) is the velocity at point 1 producing a Hohmann transfer from point 1 to point 11 (see Fig. 1). The transit estimate (68d) is simply the time required to traverse half the Earth's circumference at the top of the atmosphere for the average speed \((V_0 + V_1) / 2\).

**Nominal Functions B.** The second set of nominal functions is identical to the first with the exception of the altitude history. In this case, we enlist a parabolic variation of altitude with time, guessing at a minimum altitude \( h_{\text{min}} \) to occur at \( t = 0.5 \),

\[
\begin{align*}
    h(t) &= h_{\text{min}} + [h_a - h_{\text{min}}][(t - 0.5)/0.5]^2, \quad \text{(69a)} \\
    V(t) &= V_0 + (V_1 - V_0)t, \quad \text{(69b)} \\
    \gamma(t) &= 0, \quad \text{(69c)} \\
    \tau(t) &= 2\pi a / (V_0 + V_1), \quad 0 \leq t \leq 1. \quad \text{(69d)}
\end{align*}
\]
For the subsequent test cases, we choose \( h_{\text{min}} = 70 \) km. As will be seen, dipping the nominal altitude into the atmosphere portends severe consequences for MQAMP convergence.

7.2 Programming Details. MQAMP was programmed in VS FORTRAN (IBM extension of FORTRAN 77) in double precision on an IBM ES/9000 mainframe. The linear, initial-value equations (49a), (52), and (59) were solved using Hamming's modified predictor-corrector method (Ref. 12). Since this integration method utilizes functional values at several time points, another routine is required to start it: Newton’s interpolation formula using forward differences (Ref. 13) was enlisted for this purpose. The linear algebraic system (58) was solved using Gaussian elimination with scaled partial pivoting and iterative refinement for high accuracy (IMSL routine DLFCRG—Ref. 14). Finally, the definite integrals appearing in Eq.(37) were computed using 4th-order Newton-Cotes integration (Ref. 13).

Convergence Criteria. MQAMP was terminated when the performance index satisfied the following inequality:

\[
P \leq 10^{-20}.
\]

(70)

Nonconvergence was declared when any one of the following conditions arose:

(a) \( N > 50 \), \hspace{1cm} (71a)
(b) \( N_s > 20 \), \hspace{1cm} (71b)
(c) \( \omega_i = 0, \quad 1 \leq i \leq m(n+1) \). \hspace{1cm} (71c)
Here, \( N \) is the iteration number, \( N_s \) is the number of stepsize bisections during any given iteration needed to satisfy Ineq.(64), and the \( \omega_i \) are the diagonal elements of the upper triangular matrix \( U \) returned from the LU factorization of \( A \) performed by DLFCRG. If criterion (71c) is satisfied, then the matrix \( A \) is most probably singular.

**Canonical Units.** The numerical values of the parameters and nominal functions input to MQAMP were internally converted from Système Internationale (SI) units to canonical units by the following identities:

\[
1 \text{ du} = 6.498\times10^6 \, \text{m}, \quad (72)
\]
\[
1 \text{ tu} = 829.7 \, \text{s}, \quad (73)
\]

where du denotes the canonical distance unit and tu is the canonical time unit. Using these conversions, the atmospheric radius \( r_a \), the gravitational constant \( \mu \), the local acceleration due to gravity \( g_a \) at \( r_a \), and the circular orbital velocity at the top of the atmosphere \( V_a \) assume the following values:

\[
r_a = 6.498\times10^6 \, \text{m} = 1 \text{ du}, \quad (74a)
\]
\[
\mu = 3.986\times10^{14} \, \text{m}^3/\text{s}^2 = 1 \text{ du}^3/\text{tu}^2, \quad (74b)
\]
\[
g_a = \mu / r_a^2 = 9.440 \, \text{m/s}^2 = 1 \text{ du}/\text{tu}^2, \quad (74c)
\]
\[
V_a = \sqrt{\mu/r_a} = 7832 \, \text{m/s} = 1 \text{ du}/\text{tu}. \quad (74d)
\]

The use of canonical units in MQAMP was found to produce the best balance between retaining precision in the mantissa and keeping down the size of the exponent in the computer floating-point numerical representation. If the exponent is too large, then roundoff error can adversely affect the calculations. Consider for example, entrance boundary condition (20a): to compute our nominal velocity, we set \( \gamma_0 = 0 \) and solve for
$V_0$ using all 16 digits of machine precision. Now if $V_0 = 1.0310... \times 10^4$ m/s is stored in memory in SI units and is substituted back into Eq.(20a), the result should be zero. However, due to the large initial radius ($r_{00} = 4.2164 \times 10^7$ m), the first term on the left-hand side of that equation when using SI units is $O(10^{23})$ m$^4$/s$^2$. Because we only have 16 digits of precision, roundoff produces a result of $O(10^7)$ (the actual number was $1.68 \times 10^7$ m/s). Canonical units alleviate this situation by computing $V_0 = 1.316...$ du/tu — when this value is put back into Eq.(20a), along with $r_{00} = 6.488...$ du, the left-hand side now equals $7.11... \times 10^{-15}$ du$^4$/tu$^2$.

There is an even stronger motivation, however, for using canonical units which directly affects the conditioning of the matrix $A$ of linear system (58) and therefore the successful convergence of MQAMP. We expound on this critical issue in Section 7.4.

7.3 Numerical Examples

Example 1. As a first step in demonstrating the key issues involved in solving the AOT problem, we key in nominal functions $A$, Eqs.(68), and do not partition the dimensionless time interval $[0,1]$; that is, we essentially use the conventional MQA,

\begin{align}
\text{m} = 1 & \Rightarrow \text{time partition } [0, 1], \quad (75a) \\
\text{nominal functions } A. & \quad (75b)
\end{align}

A total of $2^7 = 128$ integration steps were used to numerically solve the differential equations over this interval. Utilizing the physical data enumerated in Section 5, we recall the lift coefficient lower bound,

\begin{equation}
C_{LL} = -0.9. \quad (76)
\end{equation}
The result is convergence in \( N = 9 \) iterations. A summary of the solution is provided in Table 1, and the behavior of \( P \) and \( \alpha \) over the course of iterations is depicted in Fig. 5. Note in this graph how rapidly \( P \) decreases once it is driven below 1; also, bisection of the stepsize below \( \alpha = 1 \) is only necessary in the early iterations.

<table>
<thead>
<tr>
<th>Table 1. Trajectory parameters, Example 1.</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \Delta V_{00} )</td>
</tr>
<tr>
<td>( \Delta V_{11} )</td>
</tr>
<tr>
<td>( \Delta V )</td>
</tr>
<tr>
<td>( \tau )</td>
</tr>
<tr>
<td>( h_{\text{min}} )</td>
</tr>
<tr>
<td>( \theta(h_{\text{min}}) )</td>
</tr>
<tr>
<td>( q_{\text{max}} )</td>
</tr>
<tr>
<td>( \theta(q_{\text{max}}) )</td>
</tr>
<tr>
<td>( H R_{\text{max}} )</td>
</tr>
<tr>
<td>( \theta(H R_{\text{max}}) )</td>
</tr>
<tr>
<td>( n_{\text{max}} )</td>
</tr>
<tr>
<td>( \theta(n_{\text{max}}) )</td>
</tr>
</tbody>
</table>

**Example 2.** Since we were able to solve Example 1 essentially not using the multipoint version of MQA, just what is its utility? The next example points the way toward the answer by showing what happens when nominal functions \( B \), Eqs.(69), are utilized,

\[ m = 1 \Rightarrow \text{time partition } [0, 1], \]  
\[ \text{nominal functions } B. \]
On first glance, a parabolic altitude history with $h_{\text{min}} = 70$ km seems much closer to the solution found in Example 1, and we might therefore expect faster convergence. However, not only do we not get faster convergence, but nonconvergence criterion (71b) is encountered on the 4th iteration. What happened?

The answer lies in examining the 1-norm condition number of the coefficient matrix $A$ of the linear system (58), which we denote by $\kappa_1(A)$. Recall that $A$ contains information from $n+1$ particular solutions of the linearized system equations, and components of the solution vector $k$ represent initial conditions for the true variations per unit stepsize which we seek [Eq.(59)]. From linear algebra (Ref. 15), we know that the relative accuracy in the solution $k$ is bounded by $\kappa_1(A)$ times the relative accuracy in $A$ (or the relative accuracy in $b$),

$$\frac{\|\delta k\|_1}{\|k+\delta k\|_1} \leq \kappa_1(A) \frac{\|\delta A\|_1}{\|A\|_1}, \quad (78a)$$

or

$$\frac{\|\delta k\|_1}{\|k\|_1} \leq \kappa_1(A) \frac{\|\delta b\|_1}{\|b\|_1}, \quad (78b)$$

where $\delta A$ denotes the error in $A$, $\delta b$ is the error in $b$, and $\delta k$ is the error in $k$ (due to roundoff, for example).

Figure 6 plots an estimate for the 1-norm condition number $\hat{\kappa}_1(A)$ obtained from the IMSL routine DLFCSR5. Note that for Example 1, $A$ is relatively well-conditioned on the first iteration [$\hat{\kappa}_1(A) = 9.0 \times 10^4$]. Contrast this with Example 2, where $A$ starts out

\footnote{The actual condition number $\kappa_1(A)$ is expensive to compute directly, so DLFCSR returns the estimate $\hat{\kappa}_1(A)$, which is actually an upper bound, i.e. $\kappa_1(A) \leq \hat{\kappa}_1(A)$. DLFCSR is based on the LINPACK routine DGESSL, and Ref. 16 states that $\hat{\kappa}_1(A)$ is usually within an order of magnitude of $\kappa_1(A)$, based on numerical tests.}
very ill-conditioned $[\hat{\kappa}_1(\Lambda)= 1.5 \times 10^{19}]$. Since the condition number is well above the reciprocal of machine precision (about $10^{17}$), there are essentially no correct digits in the solution vector $k$ for the first iteration of Example 2; hence, the lack of convergence is not surprising.

**Example 3.** Before we show why $\hat{\kappa}_1(\Lambda)$ has blown up when using nominal functions $B$, we enlist the power of MQAMP to divide the interval of integration into two subintervals and proceed as follows:

\[ m = 2 \Rightarrow \text{time partition } [0, 0.5, 1] , \quad (79a) \]

nominal functions $B$. \quad (79b)

Each subinterval in turn was further divided into 64 integration steps for a total of 128 over the entire time interval, as in Example 1. Now we are able to achieve convergence in $N=10$ iterations, and the results are exactly the same as shown in Table 1, to the precision given there. The reason for this success is found in Fig. 6: MQAMP with $m = 2$ has lowered the condition number estimate to $\hat{\kappa}_1(\Lambda)= 7.1 \times 10^{13}$ on the first iteration for Example 3. While this is still high, it is three orders of magnitude less than the level at which we expect trouble. Even though $\hat{\kappa}_1(\Lambda)$ approaches this critical level on iteration 3, it subsequently drops to near the starting value. This improved situation is not without cost: $\Lambda$ is now a $10 \times 10$ matrix versus a $5 \times 5$ for Examples 1-2. In terms of overall computational time, however, this increase is not significant, because most of the numerical effort goes into solving the system ODEs (6 times per iteration). Because we have taken care to spread the same number of steps across the entire time interval $[0,1]$ in all the example problems, computational time spent on solving the ODEs does not significantly vary.
Example 4. To further illustrate the reduction in $\kappa_1(\Lambda)$ possible with MQAMP and nominal functions $B$, we double the partition,

$$m = 4 \Rightarrow \text{time partition } [0, 0.25, 0.5, 0.75, 1],$$

nominal functions $B$. \hfill (80a)

$$m = 8 \Rightarrow \text{time partition } [0, 0.125, 0.25, ..., 0.875, 1].$$

nominal functions $B$. \hfill (81a)

Example 5. This final example with nominal functions $B$ doubles the partition one more time,

$$m = 8 \Rightarrow \text{time partition } [0, 0.125, 0.25, ..., 0.875, 1].$$

nominal functions $B$. \hfill (81b)

Again, both these examples spread a total of 128 integration steps over $[0,1]$, and the solution attributes agree with Table 1 to the number of digits given there. Figure 6 confirms the reduction in condition number which we expect.

To analyze why $\kappa_1(\Lambda)$ is reduced both when we increase the nominal altitude (i.e., the altitude at iteration 0) and the number of subintervals $m$ along the time domain, it is instructive to examine the iteration history of some key variables. Figure 7 presents the nominal profiles plus 9 subsequent iterations performed in Example 1. Note in Figs. 7A,E,F how the iteration 0 histories of $h$ vs. $t$, $V$ vs. $t$, and $\gamma$ vs. $t$ are exactly those stipulated in Eq. (68). Figure 7B plots the air density profile; Fig. 7C graphs the time history of the Jacobian matrix 1-norm, i.e., $\|\phi_1^T(t)\|_1$ vs. $t$; while Fig. 7D plots the real part of the dominant Jacobian eigenvalue, denoted by $\lambda(\phi_1^T(t))$. By dominant, we mean that eigenvalue having the largest real part. The most notable feature of Figs. 7C,D is how their behavior closely mimics the altitude and density histories of Figs. 7A,B. For iteration 0, $h(t)$ is constant at the high value of 120 km; correspondingly, both $\|\phi_1^T(t)\|_1$ and $\lambda(\phi_1^T(t))$ are
nearly constant versus time and are at their lowest levels. Referring back to Fig. 6 for Example 1, we see how $\dot{k}_1(\Lambda) = 9.0 \times 10^4$ for iteration 1, representing its lowest value over the course of the 9 iterations. As the trajectory is pushed deeper into the atmosphere on subsequent iterations, we note the time of minimum altitude corresponds exactly to the time of maximum Jacobian 1-norm and eigenvalue. Moreover, the levels of these maxima are increasing over two orders of magnitude as the minimum altitude dips lower. Figure 6 reflects this trend with increasing $\dot{k}_1(\Lambda)$ over several orders of magnitude to a maximum of $3.6 \times 10^{14}$ at the beginning of iteration 5, remaining close to this level thereafter. Figure 7 shows that iterations 4–9 are virtually the same — the differences are almost indistinguishable at the scale of these plots. The suspicion at this point is that $k_1(\Lambda)$ on any iteration is closely tied to the minimum altitude, and since air density varies exponentially with altitude, it must be a driving factor on $k_1(\Lambda)$.

Figure 8 plots the same variables for Example 3 [see Eqs.(69)]. Focusing on the iteration 0 curves, we note that the large variation in density produces large variations in the Jacobian 1-norm and dominant Jacobian eigenvalue. Since the minimum altitude only varies about 6 km over the 10 iterations, the maximum values of $\|\phi^T_2(t)\|_1$ and $\lambda(\phi^T_2(t))$ also do not significantly change, and we expect little variability in $\dot{k}_1(\Lambda)$. A glance at Fig. 6 reveals that this is true for all the examples using nominal functions B.

### 7.4 Condition Bound

The linear phase of the MQAMP method [see Eqs.(49)-(58)] is related to the multiple superposition method for linear ODEs described in Ref. 17. The latter algorithm differs from the former in that, for each subinterval, $n$ solutions of the homogeneous equation, i.e., Eq.(49a) with right-hand side set to zero, plus one solution of the full inhomogeneous system is used instead of $n+1$ particular

---

6 Note the linear system (58) for iteration $j$ uses nominal functions supplied by iteration $j-1$.

7 This is also known as the method of complementary functions.
solutions of the inhomogeneous system. If we assume that the linear systems produced by both methods behave similarly\(^8\), then we may enlist the approximation developed in Ref. 17,

\[ \kappa_1(A) = m v K_1, \]  
\[ K_1 = \max_{1 \leq i \leq m} \| \tilde{A}_i(t_i) \|_1, \]  

where \( v \) is a conditioning (or stability) constant for the linear multipoint boundary-value problem (49). This constant is the same no matter what numerical scheme we attempt to use — it is strictly a property of the multipoint BVP. We can bound \( K_1 \) by returning to the basic initial-value problem (IVP) over each subinterval,

\[ \dot{A}_i(t) - \phi(x_i(t)) A_i(t) = -[ \dot{x}_i(t) - \phi(x_i(t)) ], \quad t_{i-1} \leq t \leq t_i, \]  
\[ A_i(t_{i-1}) = c, \quad 1 \leq i \leq m. \]  

The general solution \( A_i(t), 1 \leq i \leq m, \) of each of these linear IVPs is well-known and is given by (Ref.17)

\[ A_i(t) = Y_i(t) \left[ c + \int_{t_{i-1}}^{t} Y_i^{-1}(s)[\phi(x_i(s)) - \dot{x}_i(s)] ds \right], \quad t_{i-1} \leq t \leq t_i. \]  

---

\(^8\)This assumption is reasonable since the linear matrix produced by each method can be derived from the other.
Here, the $n \times n$ matrix function $Y_i(t)$ is a fundamental solution of the following homogeneous IVP:

\begin{align}
\dot{Y}_i(t) - \phi^T_2(x_i(t))Y_i(t) &= 0, & t_{i-1} \leq t \leq t_i, \quad (85a) \\
Y_i(t_{i-1}) &= I_n, & 1 \leq i \leq m. \quad (85b)
\end{align}

Note that $Y_i(t)$ is guaranteed to be nonsingular over the entire subinterval $[t_{i-1}, t_i]$. By virtue of Theorem 7.1.4 of Ref.18, we can show that

\begin{align}
\|Y_i(t_i)\|_1 &\leq e^{L_i(t - t_{i-1})}, \quad (86a)
\end{align}

with

\[ L_i = \max_{t_{i-1} \leq t \leq t_i} \|\phi_2^T(x_i(t))\|_1. \quad (86b) \]

Setting $t = t_i$ and taking norms in Eq.(84), we obtain

\[ \|A_i(t_i)\|_1 \leq \|Y_i(t_i)\|_1 [\|c\|_1 + K_{2i}K_{3i}(t_i - t_{i-1})], \quad (87a) \]

where

\[ K_{2i} = \max_{t_{i-1} \leq t \leq t_i} \|Y_i^{-1}(t)\|_1, \quad (87b) \]

and

\[ K_{3i} = \max_{t_{i-1} \leq t \leq t_i} \|\phi(x_i(t)) - \dot{x}_i(t)\|_1, & 1 \leq i \leq m. \quad (87c) \]

Because $Y_i(t)$ is nonsingular, a finite $K_{2i}$ is certain to be found; also $K_{3i} \to 0$ as MQAMP converges and $P \to 0$. Recall the method of particular solutions in which the initial vector $c$ is a column of the identity matrix or the zero vector [Eq.(52)]; therefore,
\[ \|e\|_1 \leq 1. \]  \hspace{1cm} (88)

Using Ineqs.(86)-(88), we can bound the MPS matrix function at the end of each subinterval as follows:

\[ \|\tilde{A}_i(t_i)\|_1 \leq [1 + K_{2i}K_{3i}(t_i-t_{i-1})]e^{L_i(t_i-t_{i-1})}, \quad 1 \leq i \leq m. \]  \hspace{1cm} (89)

If we find the largest \( K_{2i}, K_{3i} \) and \( L_i \) together with the longest subinterval, we can obtain an upper bound for \( K_1 \) in Eq.(82),

\[ K_1 \leq [1 + K_2K_3H]e^{LH}, \]  \hspace{1cm} (90a)

where\footnote{Note Ineqs.(90c)-(90d) are valid by virtue of the definition Eq.(33). Also \( K_3 \to 0 \) as MQAMP converges.}

\[ K_2 = \max_{1 \leq i \leq m} K_{2i}, \]  \hspace{1cm} (90b)

\[ K_3 = \max_{1 \leq i \leq m} \max_{0 \leq t \leq 1} \|\phi(x(t)) - \dot{x}(t)\|_1, \]  \hspace{1cm} (90c)

\[ L = \max_{1 \leq i \leq m} L_i = \max_{0 \leq t \leq 1} \|\phi_2(x(t))\|_1, \]  \hspace{1cm} (90d)

\[ H = \max_{1 \leq i \leq m} (t_i-t_{i-1}). \]  \hspace{1cm} (90e)

Finally, we can produce an upper bound for the condition number of \( \Lambda \),

\[ \kappa_1(\Lambda) \leq m\sqrt{1 + K_2K_3H}e^{LH}. \]  \hspace{1cm} (91)
If we evenly divide \([0,1]\), then we have \(H = 1/m\), and (91) becomes

\[
\kappa_1(A) \leq m \sqrt{1 + K_2 K_3/m} e^{L/m}. \tag{92}
\]

Equation (92) shows how \(A\) is well-conditioned if the maximum norm \(L\) of the Jacobian is small, and conversely, how breaking the integration interval into many subintervals, i.e., increasing \(m\), can mitigate the ill effect of a large \(L\).

**Jacobian Norm.** The analysis of the numerical behavior chronicled in Section 7.3 is not complete without a detailed look at the Jacobian \(\phi_x\). In this way, not only will we justify our use of canonical units, but we will directly expose the effect of the density and hence the altitude on the Jacobian norm and therefore the conditioning of \(A\). Start by recalling the definition (25) of the state vector,

\[
x(t) = [h(t), V(t), \gamma(t), \tau(t)]^T. \tag{93}
\]

Then our differential equation is [see Eqs.(19) & (26)]

\[
\dot{x}(t) = \phi(x(t)), \tag{94a}
\]

with

\[
\phi(x(t)) = \begin{bmatrix}
\tau [V \sin \gamma] \\
\tau [-C_{D1} \rho(h) V^2 S/m - g \sin \gamma] \\
\tau [C_{L1} \rho(h) V^2 S/m + (V^2/r - g \cos \gamma)] / V \\
0
\end{bmatrix}, \tag{94b}
\]

where \(r = r_e + h\) and \(g = \mu/r^2\). To simplify notation, define the following accelerations:
\[ a_D = -C_{D_2} \rho V^2 S/m, \quad a_L = C_{L_2} \rho V^2 S/m, \quad a_{IL} = (V^2/r) \cos \gamma \]  
(95a)

\[ g_D = -(\mu r^2) \sin \gamma, \quad g_L = -(\mu r^2) \cos \gamma. \]  
(95b)

Here, \( a_D, a_L \) are the drag and lift accelerations, \( a_{IL} \) is the inertial acceleration projected along the lift direction, and \( g_D, g_L \) is the local acceleration due to gravity projected along the drag and lift directions, respectively. Using (95), Eqs.(94b) appear more compactly as

\[
\phi(x(t)) = \begin{bmatrix} \tau V \sin \gamma \\ \tau (a_D + g_D) \\ \tau (a_L + a_{IL} + g_L) / V \\ 0 \end{bmatrix}. \]  
(96)

The partial derivative matrix \( \phi_x \) thus appears as

\[
\phi_x(x(t)) = \begin{bmatrix} 0 & \tau \left( \frac{a_D}{\rho} \frac{d \rho}{dh} - \frac{2}{r} g_D \right) & \tau \left( \frac{a_L}{V} \frac{d \rho}{dh} - \frac{2}{r} g_L - \frac{a_{IL}}{r} \right) & 0 \\ \tau \sin \gamma & \tau \left( \frac{2a_D}{V} \right) & \frac{\tau}{V^2} (a_L + a_{IL} - g_L) & 0 \\ \tau V \cos \gamma & \tau g_L & -\frac{\tau}{V} \tan \gamma (a_{IL} + g_L) & 0 \\ V \sin \gamma & a_D + g_D & (a_L + a_{IL} + g_L) / V & 0 \end{bmatrix}. \]  
(97)

To evaluate the 1-norm of the Jacobian \( \phi_x^T \), we note that

\[
\| \phi_x^T \|_1 = \| \phi_x \|_\infty = \max_{1 \leq i \leq 4} \sum_{j=1}^{4} |\phi_{xij}|. \]  
(98)

where \( \phi_{xij} \) denotes the entry in the \( i \)th row and \( j \)th column of \( \phi_x \).
The first item to note when using this norm is the critical effect the choice of units has on the value obtained. Eq.(98) directs one to sum the absolute values across each row\(^{10}\): obviously each of the entries do not have the same units. Consider for example the second row and assume the use of SI units: \(\phi_{x11}\) has units of s, \(\phi_{x\tau2}\) is nondimensional, and \(\phi_{x23}\) is in s/m. Observe that the first three rows will have a greater 1-norm than the last row because they are multiplied by \(\tau\) — typically a large number when expressed in seconds. However, regardless of our choice of units, the first finding we can cite is the following:

**Fact 1.** \(\|\phi_{x1}\|_1\) and thus \(\kappa_1(A)\) are less for \(\tau\) small.

As we shall see in Section 8, greater values of \(C_L\) (i.e., lesser values of \(|C_L|\)) result in lesser times of flight \(\tau\); in some cases, the result is that once again \(m=1\) allows convergence (e.g. \(C_L=0\)). The accelerations, on the other hand, are numerically much less than \(\tau\)—for all of the examples of Section 7, they are approximately \(g_0 = 9.81\, \text{m/s}^2\) vs. \(\tau = 2257\) s. The Earth radius \(r\) and velocity \(V\) are bigger still, \(r = 6.5\times10^6\) m and \(V = 10^4\) m/s. Also, \(|\gamma|<5^\circ\) is typical, so \(\cos\gamma=1\) and \(\sin\gamma=\gamma\) are excellent approximations. The only other factor not accounted for is

\[
\frac{1}{\rho} \frac{dp}{dh}. \tag{99}
\]

Due to the nearly exponential behavior of the air density for \(h \leq 120\) km (see Section 3), this term turns out to be nearly constant. The density may be functionally related to the altitude as follows:

\(^{10}\)In other words, take the vector 1-norm of each row.
\[ \rho = \rho_0 e^{\beta h}, \]  
(100)

where \( \rho_0 \) is the density at \( h=0 \) and \( \beta \) is the negative inverse of the scale height. From the Standard Atmosphere, we have \( \rho_0 = 1.2250 \text{ kg/m}^3 \) and \( \rho_{120 \text{ km}} = 2.222 \times 10^{-8} \text{ kg/m}^3 \); fitting Eq.(100) to these points yields

\[ \beta = \frac{1}{h} \log \left( \frac{\rho}{\rho_0} \right) = -1.5 \times 10^{-4} \text{ 1/m.} \]  
(101)

Now taking the derivative of Eq.(100),

\[ \frac{d\rho}{dh} = \beta \rho_0 e^{\beta h} = \beta \rho, \]

or

\[ \frac{1}{\rho} \frac{d\rho}{dh} = \beta. \]  
(102)

Therefore, in SI units, all the terms in the first row of (97) are small, due to the large \( r \) and small \( \beta \). A very good estimate for the 1-norm of the Jacobian comes from the \( \phi_{x31} \) entry, which is the only term with two large SI magnitudes in the numerator. Thus we have for our numerical examples,

\[ ||\phi_{x}^{1}(t)||_1 \approx \tau V(t) = 2.3 \times 10^7 \text{ m.} \]  
(103)

This very large magnitude, which is essentially invariant with \( t \) [because \( V(t) \) is of \( O(10^4) \) m/s for \( 0 \leq t \leq 1 \)], dooms any attempt to enlist SI units in MQAMP. The linear matrix \( A \) is always ill-conditioned and accurate steps cannot be computed. Canonical units, however,
knock down the dominance of the $\phi_{x31}$ term, because now $V(t) = \mathcal{O}(1)$ du/tu and $\tau = \mathcal{O}(1)$ tu. Thus,

$$|\phi_{x31}| = 2.3 \times 10^7 \text{ m} = 3.5 \text{ du}. \quad (104)$$

The accelerations are then of order

$$g_0 = 9.81 \text{ m/s}^2 \approx 1 \text{ du/tu}^2. \quad (105)$$

Unfortunately, the reciprocal scale height grows,

$$\beta = -1.5 \times 10^{-4} \text{ 1/m} = -975 \text{ 1/du}. \quad (106)$$

The effect on the Jacobian norm is to make the first row dominant, and we obtain the close approximation

$$\|\phi_x^T(t)\|_1 = |\tau \beta a_D| + |\tau \beta a_L / V|. \quad (107)$$

This result directly explains the variation of Jacobian 1-norm with density, and hence altitude, we observed in Figs. 7C and 8C. The lift and drag accelerations $a_L, a_D$ are directly proportional to the density [Eqs.(95)]. Because $\tau$ and $\beta$ are constant with $t$, and $V$ nearly so, $\|\phi_x^T(t)\|_1$ will change over about the same orders of magnitude as the density, and due to the latter's exponential variation with altitude, trajectories dipping further into the atmosphere inevitably produce increasingly ill-conditioned linear systems (58). As a result, nominal trajectories (iteration 0) that start high (such as in Example 1), while they are farther from the solution, produce better steps than those with lower minimum altitudes,
such as in Example 2. MQAMP gives us a way to circumvent the deleterious effect of these lower nominal trajectories by increasing \( m \); however, the relentless exponential rise in \( \rho \) with decreasing \( h_{\text{min}} \) makes this ineffective after a certain point. If we take \( h_{\text{min}} = 40 \) km, then even \( m = 64 \) and 8 integration steps per subinterval is insufficient to produce convergence: \( \kappa_1(A) = 10^{24} \) results on the first iteration. We thus summarize with the following:

**Fact 2.** \( \kappa_1(A) \) is less for \( h_{\text{min}} \) high.

As we will see in Section 8, increasing \( C_L \) results in monotonically lower \( h_{\text{min}} \) for the converged trajectory. Therefore, convergence becomes more and more difficult with increasing \( C_L \); despite the fact \( \tau \) is decreasing as we have already alluded, \( \rho_{\text{max}} \) and thus \( a_{D_{\text{max}}} \) and \( a_{L_{\text{max}}} \) are increasing faster.

As a final point, let us compare the nominal trajectories for Examples 1 and 2. For Example 1, we have

\[
|a_{D_{\text{max}}}| = 3.94 \times 10^{-3} \text{ m/s}^2 = 4.17 \times 10^{-4} \text{ du/tu}^2,
\]

\[
|a_{L_{\text{max}}}| = 3.54 \times 10^{-3} \text{ m/s}^2 = 3.76 \times 10^{-4} \text{ du/tu}^2.
\]

Our nominal time of flight is

\[
\tau = 2248 \text{ s} = 2.71 \text{ tu}.
\]

Also,

\[
V_{a_{D_{\text{max}}}} = 1.03 \times 10^4 \text{ m/s} = 1.32 \text{ du/tu}.
\]
Equations (90d) and (107) predict\textsuperscript{11}

\[
L = \max_{0 \leq t \leq 1} \| \phi_t^T(x(t)) \|_1 \\
= (2.71)(975)(4.17 \times 10^{-4} + 3.76 \times 10^{-4}/1.32) = 1.85. \quad (108)
\]

The first row of $\phi_x$ is no longer dominant because $a_D$ and $a_L$ are so low; however (108) gives the correct order of magnitude. The actual maximum Jacobian 1-norm along the nominal trajectory was, for

Example 1: $L = 6.28$. \quad (109)

Now, for Example 2, $h_{\min} = 70$ km, so

\[
|a_{D_{\max}}| = 11.4 \text{ m/s}^2 = 1.20 \text{ du/tu}^2, \\
|a_{L_{\max}}| = 10.2 \text{ m/s}^2 = 1.08 \text{ du/tu}^2, \\
V_{a_{D_{\max}}} = 9.08 \times 10^3 \text{ m/s} = 1.16 \text{ du/tu},
\]

\[
L = (2.71)(975)(1.20 + 1.08/1.16) = 5640.
\]

This closely compares to the actual value for

Example 2: $L = 5276$. \quad (110)

\textsuperscript{11}Note we do not list units because $\phi_{x12}$ and $\phi_{x13}$ have disparate units.
It is the three order-of-magnitude reduction in $L$ in Example 1 versus Example 2 that allows $\kappa_1(\lambda)$ to be small, as Eq.(92) predicts and Fig. 6 confirms.

7.5 Grazing Trajectory. As a final task for this section, we numerically compute the grazing trajectory. Doing so is particularly easy: we merely use the system equations (19)-(21) in which the control $C_L$ is no longer constant. If we make the lift plus weight precisely equal to the value needed to offset centrifugal force, then the vehicle should hug the top of the atmosphere and produce the grazing trajectory we seek. Since centrifugal force is proportional to the square of the velocity, this means the required lift coefficient will be a function of the velocity. Substituting the second of Eqs.(4) in Eq.(2c), setting $\gamma = 0$ and $\dot{\gamma} = 0$, and solving for $C_L$ yields the following:

$$C_L = C[1 - (V_*/V)^2], \quad (111a)$$
$$C = -2(m/S) / (\rho_ar_a), \quad (111b)$$

where $V_* = 7.832 \text{ km/s}$ is circular orbital velocity at the top of the atmosphere. Utilizing the data from Section 5, we get

$$C = -4156. \quad (112)$$

Using control history (111), MQAMP quite readily converges to the answer: in fact we can use $m=1$ as we did in Example 1 because we know the Jacobian is well-behaved. Table 2 and Fig. 9 summarize the results. The values of $\Delta V_{00}$ and $\Delta V_{11}$ in Table 2 are precisely those we compute using Eqs.(17).

The most important result to note is the large lift coefficient range $-1758 \leq C_L \leq -19$. Because the denominator term $\rho_ar_a = 0.144 \text{ kg/m}^2$ of Eq.(111b) is so small, the
wing loading term $m/S$ would have to be unrealistically small (i.e., three orders of magnitude smaller than it is at 300 kg/m$^2$) to reduce the absolute value of $C$ and produce lift coefficients within an achievable range. The parabolic drag polar [see Eq.(5)] squares an already large lift coefficient to produce a huge drag force which decelerates the vehicle quite quickly — hence the short time of flight $\tau$.

These results make quite evident the impracticality of the grazing trajectory. Nonetheless, its theoretical value will become apparent as we compare its trajectory parameters to the results of the trade studies using realistic lift coefficients in the next section.

### Table 2. Grazing trajectory parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta V_{00}$</td>
<td>1.486 km/s</td>
</tr>
<tr>
<td>$\Delta V_{11}$</td>
<td>0.018 km/s</td>
</tr>
<tr>
<td>$\Delta V$</td>
<td>1.504 km/s</td>
</tr>
<tr>
<td>$\tau$</td>
<td>19.68 s</td>
</tr>
<tr>
<td>$q_{\text{max}}$</td>
<td>1.18 Pa</td>
</tr>
<tr>
<td>$HR_{\text{max}}$</td>
<td>5.57 W/cm$^2$</td>
</tr>
<tr>
<td>$n_{\text{max}}$</td>
<td>1376</td>
</tr>
<tr>
<td>$C_{L0}$</td>
<td>$-1758$</td>
</tr>
<tr>
<td>$C_{D0}$</td>
<td>$3.43 \times 10^6$</td>
</tr>
<tr>
<td>$C_{L1}$</td>
<td>$-19.0$</td>
</tr>
<tr>
<td>$C_{D1}$</td>
<td>401</td>
</tr>
</tbody>
</table>
8. Numerical Results

8.1 Single Arc. In the previous section, we have summarized critical kinematic and dynamic points for the single-arc case \( C_L(\theta) = C_{LL} = -0.9, 0 \leq \theta \leq \tau \) solved via MQAMP with various numbers \( m \) of subintervals. We now investigate the effect of changing the lift coefficient in these single-arc trajectories.

Table 3 summarizes the results obtained by varying \( C_L \) from \(-0.9\) to \(+0.06\). The parameters of the grazing trajectory are also included for comparison.

<table>
<thead>
<tr>
<th>( C_L )</th>
<th>( \gamma_0 ) (deg)</th>
<th>( \gamma_1 ) (deg)</th>
<th>( h_{\min} ) (km)</th>
<th>( q_{\max} ) (kPa)</th>
<th>( HR_{\max} ) (W/cm²)</th>
<th>( n_{\max} ) (m/s)</th>
<th>( \Delta V ) (m/s)</th>
<th>( \Delta V_{00} ) (m/s)</th>
<th>( \Delta V_{11} ) (m/s)</th>
<th>( \tau ) (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grazing†</td>
<td>0</td>
<td>0</td>
<td>120</td>
<td>0.0012</td>
<td>5.6</td>
<td>1376</td>
<td>1503.7</td>
<td>1485.7</td>
<td>19.7</td>
<td>19.7</td>
</tr>
<tr>
<td>-0.9</td>
<td>-4.081</td>
<td>0.319</td>
<td>76.0</td>
<td>1.7</td>
<td>196</td>
<td>0.77</td>
<td>1514.4</td>
<td>1489.9</td>
<td>24.6</td>
<td>2257</td>
</tr>
<tr>
<td>-0.8</td>
<td>-4.121</td>
<td>0.319</td>
<td>75.1</td>
<td>1.9</td>
<td>211</td>
<td>0.74</td>
<td>1514.5</td>
<td>1489.9</td>
<td>24.6</td>
<td>2434</td>
</tr>
<tr>
<td>-0.7</td>
<td>-4.167</td>
<td>0.320</td>
<td>74.1</td>
<td>2.3</td>
<td>228</td>
<td>0.73</td>
<td>1514.6</td>
<td>1490.0</td>
<td>24.6</td>
<td>2636</td>
</tr>
<tr>
<td>-0.6</td>
<td>-4.220</td>
<td>0.323</td>
<td>72.9</td>
<td>2.7</td>
<td>250</td>
<td>0.71</td>
<td>1514.8</td>
<td>1490.1</td>
<td>24.6</td>
<td>2864</td>
</tr>
<tr>
<td>-0.5</td>
<td>-4.281</td>
<td>0.327</td>
<td>71.6</td>
<td>3.3</td>
<td>277</td>
<td>0.70</td>
<td>1515.1</td>
<td>1490.3</td>
<td>24.9</td>
<td>3108</td>
</tr>
<tr>
<td>-0.4</td>
<td>-4.355</td>
<td>0.333</td>
<td>70.0</td>
<td>4.2</td>
<td>313</td>
<td>0.69</td>
<td>1515.6</td>
<td>1490.4</td>
<td>25.1</td>
<td>3339</td>
</tr>
<tr>
<td>-0.3</td>
<td>-4.450</td>
<td>0.343</td>
<td>67.8</td>
<td>5.6</td>
<td>363</td>
<td>0.68</td>
<td>1516.2</td>
<td>1490.6</td>
<td>25.6</td>
<td>3466</td>
</tr>
<tr>
<td>-0.2</td>
<td>-4.583</td>
<td>0.363</td>
<td>64.9</td>
<td>8.2</td>
<td>440</td>
<td>0.69</td>
<td>1517.4</td>
<td>1490.9</td>
<td>26.4</td>
<td>3278</td>
</tr>
<tr>
<td>-0.1</td>
<td>-4.805</td>
<td>0.405</td>
<td>60.1</td>
<td>14.9</td>
<td>583</td>
<td>0.76</td>
<td>1520.0</td>
<td>1491.4</td>
<td>28.5</td>
<td>2414</td>
</tr>
<tr>
<td>-0.05</td>
<td>-5.014</td>
<td>0.474</td>
<td>55.8</td>
<td>24.4</td>
<td>728</td>
<td>0.95</td>
<td>1524.4</td>
<td>1492.0</td>
<td>32.5</td>
<td>1625</td>
</tr>
<tr>
<td>-0.02</td>
<td>-5.248</td>
<td>0.651</td>
<td>51.4</td>
<td>38.9</td>
<td>891</td>
<td>1.35</td>
<td>1537.9</td>
<td>1492.6</td>
<td>45.4</td>
<td>1018</td>
</tr>
<tr>
<td>0</td>
<td>-5.532</td>
<td>1.025</td>
<td>47.5</td>
<td>61.1</td>
<td>1080</td>
<td>2.08</td>
<td>1578.7</td>
<td>1493.3</td>
<td>85.3</td>
<td>656</td>
</tr>
<tr>
<td>+0.02</td>
<td>-6.043</td>
<td>1.710</td>
<td>43.2</td>
<td>105.6</td>
<td>1377</td>
<td>3.68</td>
<td>1696.4</td>
<td>1494.8</td>
<td>201.6</td>
<td>430</td>
</tr>
<tr>
<td>+0.04</td>
<td>-7.006</td>
<td>2.766</td>
<td>38.9</td>
<td>189.8</td>
<td>1793</td>
<td>7.05</td>
<td>1971.0</td>
<td>1498.0</td>
<td>473.1</td>
<td>296</td>
</tr>
<tr>
<td>+0.06</td>
<td>-9.120</td>
<td>4.604</td>
<td>33.7</td>
<td>390.2</td>
<td>2460</td>
<td>15.92</td>
<td>2623.0</td>
<td>1506.3</td>
<td>1116.7</td>
<td>204</td>
</tr>
</tbody>
</table>

†\(-1758 \leq C_L \leq -19.0\) for the grazing trajectory.
Figure 10 includes plots of the quantities in Table 3 versus lift coefficient: presented are minimum altitude \( h_{\text{min}} \), peak dynamic pressure \( q_{\text{max}} \), peak heating rate \( HR_{\text{max}} \), peak load factor \( n_{\text{max}} \), entrance flight path angle \( \gamma_0 \), time of flight \( \tau \), and characteristic velocity \( \Delta V \) versus \( C_L \).

Inspection of Table 3 and Fig. 10 reveals that the trajectory having the most negative lift coefficient, \( C_L = -0.9 \), is the one which comes closest to the attributes of the grazing trajectory. Its entrance flight path angle is the shallowest, its minimum altitude the highest, and its peak dynamic pressure, peak heating rate, peak load factor, and characteristic velocity are the lowest of all the single-arc cases run.\(^{13}\)

Note that since the entrance flight path angle only changes a small amount over the lift coefficient range \(-0.9 \leq C_L \leq -0.02\), and since the exit angle changes even less, there is very little variation in the characteristic velocity \( \Delta V \). For \( C_L > -0.02 \), however, the exit flight path angle \( \gamma_1 \) begins to climb sharply: what this means is the exit velocity must decrease in order to satisfy the dynamic boundary condition (21a) at point 1. Consequently, this velocity deficiency translates to point 11, and an ever greater circularization impulse must be applied there, as the \( \Delta V_{11} \) column in Table 3 attests (recall again Ref. 1, which demonstrated the growth in \( \Delta V_{11} \) to be proportional to the square of \( \gamma_1 \)). Now contrast the behavior of \( \Delta V, \gamma_0, \gamma_1 \) throughout the lift coefficient range with the relatively large diminishing of minimum altitude and the absolutely massive order-of-magnitude variations in peak dynamic pressure and peak heating rate. We expand on the reason for the latter situation below.

Table 3 and Fig. 10F show the time of flight to almost linearly increase with lift coefficient to a maximum \( \tau = 3466 \) s at \( C_L = -0.3 \), rapidly dropping thereafter. The existence of this maximum may be heuristically explained by two conflicting trends: (1) the

\(^{13}\)We reiterate that optimizing one performance index appears to optimize or nearly optimize the rest.
trajectories with lower $|C_L|$ feature increasingly deeper plunges into the atmosphere which
require longer times to climb out; and (2) lower lift is less able to counteract centrifugal
force, and the craft is more rapidly pulled back out of the atmosphere.

The $C_L = +0.06$ case was the last for which convergence satisfying Ineqs.(71) was
achieved; setting $C_L = +0.08$ resulted in nonconvergence. It has not been determined
whether this is a true limit to the existence of a solution to the TPBVP Eqs.(19)-(21) or
whether more extensive experimentation with the nominal trajectory will break through this
barrier. We should note at this point that nominal functions A [Eqs.(68)] were used for
cases spanning $-0.9 \leq C_L \leq -0.02$. For $C_L \geq 0$, convergence could no longer be achieved
using transit time estimate (68d), $\tau(t) = 2248$ s; however, setting nominal $\tau(t) = 1000$ s for
$C_L = 0$, $+0.02$, and $\tau(t) = 500$ s for $C_L = +0.04$, $+0.06$ did produce convergence. The
analysis of Section 7 explains why these smaller nominal flight times were beneficial — the
maximum Jacobian norms were lower for the initial iterations, the MPS linear coefficient
matrix $A$ was better conditioned, and higher accuracy in the state variations was obtained.
No value of $\tau(t)$ along with nominal functions (68a)-(68c) would enable convergence when
we set $C_L = +0.08$. As Fig. 10F shows, the time of flight is rapidly dropping with positive
lift coefficients: if we extrapolate the curve past $C_L = +0.06$, it appears $\tau$ would go to zero
somewhere near $C_L = +0.1$. Therefore, we are probably not too far from the nonexistence
of a solution at $C_L = +0.06$.

We choose three trajectories ($C_L = -0.9, -0.5, +0.06$) spanning the single-arc cases
run and plot selected kinematic and dynamic variables versus the dimensional time $\theta$ in Fig.
11. Figures 11A and 11C illustrate that for all three trajectories, the AOT vehicle dives to
its minimum altitude quickly, followed by a slow climb back out. Most of the velocity
depletion, however, occurs on this longer post-perigee leg, as Fig. 11B attests. The fact
that the air density history is plotted on a log scale in Fig. 11E emphasizes the large
variability in this quantity which the craft must endure. Because dynamic pressure, heating
rate, and load factor are all functions of the density, they also vary widely over the course of the trajectory, as Figs. 11F–H indicate.

**Sensitivity of \( q_{\text{max}} \) and \( HR_{\text{max}} \) to Altitude.** In Section 3, we pointed out that the peak values of dynamic pressure and heating rate are sensitive to the altitudes at which they occur due to the large density gradient. We illustrate by considering two single-arc atmospheric passes and comparing the dynamic states at three critical points: (a) minimum altitude, (b) peak dynamic pressure, and (c) peak heating rate. Trajectory 1 was flown with \( C_L = -0.9 \), and trajectory 2 had \( C_L = -0.5 \). Table 4 shows that, for each trajectory, these three critical points occur very close to each other (within 36 s). Minimum altitudes for the two trajectories are only 4.4 km apart.

Let us first take the definition of dynamic pressure, Eq.(12b), to derive the following:

\[
\frac{q_2}{q_1} = \left(\frac{\rho_2}{\rho_1}\right)\left(\frac{V_2}{V_1}\right)^2.
\]  

(113)

Comparing the \( q_{\text{max}} \) points of trajectories 1 and 2, we find the velocities there are not very much different,

\[
\frac{V_2}{V_1} = 1.007.
\]  

(114)

However, the 4.4 km altitude difference between the two \( q_{\text{max}} \) points has almost doubled the air density,

\[
\frac{\rho_2}{\rho_1} = 1.921.
\]  

(115)

Hence, Eq.(113) predicts the change in dynamic pressure that results,
\[ q_2/q_1 = 1.95. \]  
\[ (116) \]

**Table 4.** Comparison of critical points for two single-arc trajectories.

<table>
<thead>
<tr>
<th>Critical Point</th>
<th>Trajectory 1: ( C_L = -0.9 )</th>
<th>Trajectory 2: ( C_L = -0.5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \theta ) (s)</td>
<td>168</td>
<td>150</td>
</tr>
<tr>
<td>( h ) (km)</td>
<td>76.0</td>
<td>76.3</td>
</tr>
<tr>
<td>( \rho ) (kg/m(^3))</td>
<td>3.42×10(^{-5})</td>
<td>3.29×10(^{-5})</td>
</tr>
<tr>
<td>( V ) (km/s)</td>
<td>9.92</td>
<td>10.0</td>
</tr>
<tr>
<td>( q ) (Pa)</td>
<td>1685</td>
<td>1653</td>
</tr>
<tr>
<td>( HR ) (W/cm(^2))</td>
<td>194</td>
<td>196</td>
</tr>
<tr>
<td>( n )</td>
<td>0.770</td>
<td>0.775</td>
</tr>
</tbody>
</table>

Thus we have shown how only a small 4.4 km change in altitude will practically double the maximum dynamic pressure imposed on the vehicle.

We carry out an identical procedure for the peak heating rate by using Eq.(13b) and forming the following ratio:

\[ HR_2/HR_1 = \sqrt{(\rho_2/\rho_1)(V_2/V_1)^{3.07}}. \]  
\[ (117) \]

Comparing the peak heating rate points in Table 4, we find

\[ HR_2/HR_1 = 1.42. \]  
\[ (118) \]
This aside has served to illustrate the point that controlling the minimum altitude has profound implications for the structural design of an AOT vehicle. The dynamic pressure is most affected because it is proportional to the first power of air density, while the heating rate [as approximated by Eq.(13b)] is slightly less sensitive due to its square root proportionality. Nonetheless, a 42% increase in heating rate is significant over an altitude difference of only 4.4 km. As Table 3 and Fig. 10 show, larger altitude differences imply ever-worsening effects on the critical design parameters $q_{\text{max}}, HR_{\text{max}}$, and $n_{\text{max}}$. High lift is clearly desirable for an AOTV.

**Remark.** In this subsection, we have presented numerical evidence which supports the conjecture of Miele, Wang, and Deaton in Ref. 1 that the actual AOT trajectory produced by fixing the lift coefficient at its lower bound — the so-called nearly-grazing trajectory — is optimal. There is of course an infinite variety of other candidate control distributions which may be optimal. The following subsection, however, favorably compares the single-arc trajectory against a significant class of control distributions which heretofore has been considered optimal.

### 8.2 Two Subarcs

In Section 4.2, we described a trajectory produced by fixing the lift coefficient at its upper bound for a certain time, transitioning discontinuously to its lower bound for the remainder of the pass. Each portion of constant lift constituted a separate subarc of this two-subarc problem. Adapting the system equations and boundary conditions to each subarc and enforcing continuity at the subarc interface produced an 8-dimensional TPBVP in which 6 variables represented the physical state $(h,V,\gamma)$ in both subarcs, and the other two represented the duration of each subarc.

For our numerical experiments, we utilized the genuine control bounds $C_{LL} = -0.9$, $C_{LU} = +0.9$ as delineated in Section 5. Recall in our formulation of Section 4.2 we were
free to choose the entrance flight path angle. Table 5 presents the range of $\gamma_0$ for which convergence to the criteria outlined in Section 6 was achieved.

### Table 5. Two-subarc trajectory summary.

<table>
<thead>
<tr>
<th>$\gamma_0$ (deg)</th>
<th>$\gamma_1$ (deg)</th>
<th>$q_{\text{max}}$ (Pa)</th>
<th>$\Delta V$ (m/s)</th>
<th>$\tau_1$ (s)</th>
<th>$\tau_2$ (s)</th>
<th>$\tau$ (s)</th>
<th>$\tau_1/\tau$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-4.54</td>
<td>0.319</td>
<td>1782.84</td>
<td>1515.4</td>
<td>92.9</td>
<td>2135.29</td>
<td>2228.2</td>
<td>0.0417</td>
</tr>
<tr>
<td>-4.53</td>
<td>0.319</td>
<td>1773.22</td>
<td>1515.37</td>
<td>92.6</td>
<td>2136.63</td>
<td>2229.21</td>
<td>0.0415</td>
</tr>
<tr>
<td>-4.52</td>
<td>0.319</td>
<td>1764.33</td>
<td>1515.35</td>
<td>92.2</td>
<td>2137.97</td>
<td>2230.2</td>
<td>0.0414</td>
</tr>
<tr>
<td>-4.5</td>
<td>0.319</td>
<td>1749.57</td>
<td>1515.31</td>
<td>91.5</td>
<td>2140.65</td>
<td>2232.15</td>
<td>0.0410</td>
</tr>
<tr>
<td>-4.4</td>
<td>0.319</td>
<td>1707.07</td>
<td>1515.09</td>
<td>86.9</td>
<td>2154.11</td>
<td>2240.97</td>
<td>0.0388</td>
</tr>
<tr>
<td>-4.35</td>
<td>0.319</td>
<td>1697.04</td>
<td>1514.98</td>
<td>83.8</td>
<td>2160.99</td>
<td>2244.78</td>
<td>0.0373</td>
</tr>
<tr>
<td>-4.3</td>
<td>0.319</td>
<td>1690.8</td>
<td>1514.87</td>
<td>80.0</td>
<td>2168.13</td>
<td>2248.14</td>
<td>0.0356</td>
</tr>
<tr>
<td>-4.2</td>
<td>0.319</td>
<td>1685.57</td>
<td>1514.66</td>
<td>69.2</td>
<td>2184.32</td>
<td>2253.49</td>
<td>0.0307</td>
</tr>
<tr>
<td>-4.1</td>
<td>0.319</td>
<td>1684.71</td>
<td>1514.45</td>
<td>42.9</td>
<td>2213.78</td>
<td>2256.68</td>
<td>0.0190</td>
</tr>
<tr>
<td>-4.09</td>
<td>0.319</td>
<td>1684.65</td>
<td>1514.43</td>
<td>34.5</td>
<td>2222.35</td>
<td>2256.83</td>
<td>0.0153</td>
</tr>
<tr>
<td>-4.085</td>
<td>0.319</td>
<td>1684.36</td>
<td>1514.42</td>
<td>26.5</td>
<td>2230.35</td>
<td>2256.88</td>
<td>0.0118</td>
</tr>
<tr>
<td>-4.08057†</td>
<td>0.319</td>
<td>1684.68</td>
<td>1514.41</td>
<td>0</td>
<td>2256.91</td>
<td>2256.91</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\gamma_0$ (deg)</th>
<th>$\Delta V_{00}$ (m/s)</th>
<th>$\Delta V_{11}$ (m/s)</th>
<th>$h_{\text{min}}$ (km)</th>
<th>$HR_{\text{max}}$ (W/cm²)</th>
<th>$n_{\text{max}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-4.54</td>
<td>1490.83</td>
<td>24.5647</td>
<td>75.7178</td>
<td>206.148</td>
<td>0.815</td>
</tr>
<tr>
<td>-4.53</td>
<td>1490.81</td>
<td>24.5647</td>
<td>75.7414</td>
<td>205.118</td>
<td>0.810</td>
</tr>
<tr>
<td>-4.52</td>
<td>1490.79</td>
<td>24.5647</td>
<td>75.762</td>
<td>204.205</td>
<td>0.806</td>
</tr>
<tr>
<td>-4.5</td>
<td>1490.74</td>
<td>24.5647</td>
<td>75.7982</td>
<td>202.658</td>
<td>0.801</td>
</tr>
<tr>
<td>-4.4</td>
<td>1490.52</td>
<td>24.5647</td>
<td>75.9012</td>
<td>198.307</td>
<td>0.780</td>
</tr>
<tr>
<td>-4.35</td>
<td>1490.41</td>
<td>24.5647</td>
<td>75.9261</td>
<td>197.28</td>
<td>0.776</td>
</tr>
<tr>
<td>-4.3</td>
<td>1490.31</td>
<td>24.5647</td>
<td>75.9414</td>
<td>196.664</td>
<td>0.773</td>
</tr>
<tr>
<td>-4.2</td>
<td>1490.09</td>
<td>24.5647</td>
<td>75.9551</td>
<td>196.116</td>
<td>0.770</td>
</tr>
<tr>
<td>-4.1</td>
<td>1489.89</td>
<td>24.5647</td>
<td>75.958</td>
<td>196</td>
<td>0.770</td>
</tr>
<tr>
<td>-4.09</td>
<td>1489.87</td>
<td>24.5647</td>
<td>75.9577</td>
<td>195.987</td>
<td>0.770</td>
</tr>
<tr>
<td>-4.085</td>
<td>1489.86</td>
<td>24.5647</td>
<td>75.9577</td>
<td>195.992</td>
<td>0.770</td>
</tr>
<tr>
<td>-4.08057†</td>
<td>1489.85</td>
<td>24.5647</td>
<td>75.9576</td>
<td>196.01</td>
<td>0.770</td>
</tr>
</tbody>
</table>

†Limiting case: the two-subarc trajectory becomes the single-arc trajectory.
Figure 12 includes plots of minimum altitude, peak dynamic pressure, peak heating rate, peak load factor, time ratio, time of flight, and characteristic velocity vs. entrance flight path angle. For both $\gamma_0 = -4.545^\circ$ and $\gamma_0 = -4.08^\circ$, the multipoint algorithm could not be made to converge. The suspicion that at least the $\gamma_0 = -4.085^\circ$ case represents a feasibility boundary for the two-subarc problem is buttressed by noticing that it is approaching the single-arc $\gamma_0 = -4.08057^\circ$. This case is included as the last row of Table 5 for comparison.

**Observations.** While the range of convergence for $\gamma_0$ was small ($\Delta \gamma_0 = 0.455^\circ$), the variation in $\gamma_1$ was practically nonexistent. For all eleven two-subarc cases and the single-arc case, the exit flight path angle $\gamma_1$ was identical to 6 decimal places. The exit velocity must also be constant to this many significant figures by virtue of boundary condition (9a). As Eq. (11b) then predicts and Table 5 shows, the postatmospheric component of the characteristic velocity $\Delta V_{11}$ is nonvarying to this level. The differences seen in $\Delta V_{00}$ and $\Delta V$ then arise due to the small variability in $\gamma_0$. Note, significantly, that $\Delta V$ decreases as $\gamma_0$ becomes shallower and as less time is spent under positive lift in subarc 1, i.e., as $\tau_1$ decreases. *The minimum $\Delta V$ is produced by the single-arc trajectory*, although the 0.01 m/s difference is admittedly small compared to the $O(10^3)$ m/s magnitude of $\Delta V$.

Minimum altitude and time of flight monotonically increase, while peak dynamic pressure, heating rate, and load factor monotonically decrease with the time $\tau_1$ spent under positive lift. Again comparing the corresponding performance indices of the single-arc trajectory, we find that they are not always better than the closest two-subarc counterparts. However, the differences in these parameters for the two-subarc $\gamma_0 = -4.085^\circ$ trajectory compared with the single-arc $\gamma_0 = -4.08057^\circ$ trajectory are typically found in the sixth significant figure. While the single-arc $\Delta V$ is 0.01 m/s less than the two-subarc $\Delta V$ and the time of flight is 0.03 s longer, the single-arc $h_{\text{min}}$ is 0.1 m lower, $q_{\text{max}}$ is 0.32 Pa higher, $HR_{\text{max}}$ is 0.018 W/cm² higher, and $n_{\text{max}}$ is $1.48 \times 10^{-4}$ higher than the corresponding two-
subarc values. These tiny differences could be accounted for by numerical granularity, so it is difficult to draw conclusions about optimality. Nonetheless, the time of flight and characteristic velocity differences are more believable because those quantities are not as directly tied to the size of the integration step as are the others. The number of integration steps used was $2^9 = 512$ for both single-arc and two-subarc cases; therefore, the time uncertainty of critical point occurrence was $0.5(2257 \text{ s/512}) = \pm 2.2 \text{ s}$. More important than this single-arc/two-subarc comparison are the trends with decreasing duration of positive lift mentioned above. While these trends imply that perhaps no positive lift is needed to optimize the various performance indices, the actual AOT vehicle may need to utilize positive lift coupled with a steeper entry flight path angle to satisfy stability requirements. The results of this section show that only very small penalties in all the performance indices need be paid for such a course.
9. Conclusions

This thesis has concentrated on solving the two-point boundary-value problem associated with planar aeroassisted orbital transfer from GEO to LEO first posed in Ref. 1. This TPBVP results from fixing the lift coefficient at its lower bound in the governing system equations and is highly nonlinear due to the exponential variation of density with altitude. Conventional quasilinearization algorithms are difficult or impossible to use due to the large Jacobian norms which lead to divergence in trial solutions during the course of iteration.

Our response has been to utilize a multipoint variant of the modified quasilinearization algorithm which is ideally suited to solving highly nonlinear TPBVPs. In the multipoint approach, the time domain is divided into an arbitrary number of subdomains, and the system equations are integrated in these smaller regions. Consequently, divergence is attenuated and accurate solutions can be computed. This algorithm is naturally developed for and is capable of solving multipoint boundary-value problems as well.

Armed with this computational tool, we conduct trade studies to test the hypothesis of Ref. 1 that fixing the lift coefficient at the lower bound will produce optimal AOT trajectories. Such optimality can be gauged by a number of performance indices, the most important including characteristic velocity, minimum altitude, peak dynamic pressure, peak heating rate, and peak load factor. By varying the level of the lift coefficient, we find that the most negative $C_L = -0.9$ (greatest $|C_L|$) produces a trajectory most favorable from the standpoint of all performance indices listed above. This case best approximates the grazing trajectory — an ideal trajectory flown at constant altitude along the edge of the atmosphere which represents the theoretical optimum, but which is also inherently unflyable due to the exorbitant lift coefficients required.
Previous studies have identified as requisite for optimality a short period of positive lift on the descending portion of the atmospheric pass followed by a switch to negative lift for the climb out. We idealize this control by formulating a two-subarc trajectory in which fixed maximum positive lift in one subarc is discontinuously followed by extreme negative lift in the other. A TPBVP is again formulated with the entrance flight path angle chosen to be a free parameter. We find only a small range of convergence for $\gamma_0$, and notice that all the performance indices improve with diminishing periods of positive lift. The optimal case is most likely produced by eliminating the positive lift altogether, that is, by flying the single-arc trajectory produced with strictly negative lift, as hypothesized by Miele, Wang, and Deaton in Ref. 1; however, actual AOT vehicle stability considerations might demand the use of initial positive lift combined with steeper entry flight path angle which results in only a small loss of performance.
References


Fig. 1. Coplanar aeroassisted orbital transfer.

Fig. 2. Air density profile, U.S. Standard Atmosphere, 1976 (Ref. 6).
Fig. 3. Two-subarc trajectory definition.

Fig. 4. Linear system for method of particular solutions.
Fig. 5. MQA performance, Example 1.

Fig. 6. MPS linear system condition number record.
Fig. 7. Perturbation profiles, Example 1.
Fig. 8. Perturbation profiles, Example 3.
Fig. 9. Grazing trajectory history.
Fig. 10. Single-arc trajectory parameters vs. lift coefficient.
Fig. 11. Single-arc trajectory variable histories, $C_L = -0.9, -0.5, +0.06$. 
Fig. 12. Two-subarc trajectory parameters vs. entrance flight path angle.