NASA TECHNICAL
MEMORANDUM

NASA TM X-58047
June 1970

A MODIFIED SWEEP METHOD FOR CONTROL OPTIMIZATION

A Dissertation Presented to
the Faculty of the Graduate School of
The University of Texas at Austin
In Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
MANNED SPACECRAFT CENTER
HOUSTON, TEXAS

NASA TM X-58047

A MODIFIED SWEEP METHOD FOR CONTROL OPTIMIZATION

Daniel Colunga
Manned Spacecraft Center
Houston, Texas

PREFACE

It is well known that a calculus-of-variations approach to solving the Bolza form of trajectory optimization problems usually yields a nonlinear two-point boundary value problem in terms of the state and Lagrange multiplier variables. Closed-form solutions for problems of this type are difficult to obtain except for a few simple problems. As a result, recent work in trajectory optimization has focused on numerical procedures for obtaining solutions using high-speed digital computers. Particular interest has centered on a group known as the second-order methods.

One such method is the Successive Sweep Method (SSM). It uses the generalized Riccati transformation technique to bypass a direct numerical integration of the perturbation equations. The reasons such an approach has much potential appeal are presented in this study; however, because the SSM iterates on the control values over the interval of interest, considerable computer storage is necessary even for problems of small dimension. This storage is required to compute corrections to the assumed control. Furthermore, the Eulerian control is not obtained upon convergence.

This research develops a new second-order numerical optimization method, the Modified Sweep Method (MSM). It requires very little computer storage and provides the Eulerian control. In addition, the properties and information contained in the Riccati transformation variables are preserved.

Furthermore, this research also presents a new scheme for defining classes of numerical optimization methods. The Successive Sweep Method and the Modified Sweep Method are then discussed in terms of differences arising because each falls into a different class. The Modified Sweep Method is subsequently compared numerically to the Method of Perturbation Functions (MPF), both of which belong to the same class.

The author extends thanks to Mr. Walt Williamson of The University of Texas at Austin for many helpful discussions concerning the Apollo reentry problem, to Mr. I. J. Kim, of Lockheed Electronics Company, Houston, for programming the plots, and both to Mr. Kim and Mr. Mike Frederick of The University of Texas at Austin for helping with the data. He also expresses gratitude to Dr. J. M. Lewallen of NASA/MSC and Professors A.M. Bedford and E.J. Prouse of The University of Texas at Austin for serving on the dissertation committee and helping with the manuscript. Special thanks is due to Mr. E. L. Davis of NASA/MSC for making this research possible, and for his personal interest and friendship which have provided constant inspiration.

The author expresses his indebtedness to Professor B. D. Tapley of The University of Texas at Austin, who suggested this research and provided many stimulating discussions while serving as committee supervisor.

The author expresses deep appreciation to his family for their understanding and cooperation during the course of this research.

Daniel Colunga

A MODIFIED SWEEP METHOD FOR CONTROL OPTIMIZATION

Daniel Colunga, Ph.D.
The University of Texas at Austin, 1970

Supervising Professor: B. D. Tapley

A Modified Successive Sweep Method was devised which yields Eulerian solutions to two-point boundary value problems of control optimization. This was accomplished by requiring control satisfaction of local optimality over the entire time interval of interest while simultaneously relaxing terminal transversality requirements on the Lagrange multipliers.

The new method was tested successfully on several classes of problems including optimizing the roll program for an Apollo-type three-dimensional reentry trajectory so as to minimize a time integral of spacecraft heating and acceleration.

This new method was shown to require significantly less computer storage than the original Successive Sweep Method while requiring numerical integration of fewer variables. In addition, the method was shown to possess rapid terminal convergence and a conjugate-point test capability.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOTATION LEGEND

Derivatives:

Ordinary

For any variable W

$$\dot{W} = \frac{DW}{Dt} = \frac{dW}{dt}$$

Partial

For the variables x,y the scalar S and the vector V

$$V_x = \frac{\partial V}{\partial x}$$

$$S_y = \frac{\partial S}{\partial y} \qquad S_{yx} = \frac{\partial}{\partial x}\left(\frac{\partial S}{\partial y}\right)^T$$

Differentials:

dW	total differential of the variable W

Variations:

δW	total variation of W
$\delta^1 W$	first variation of W
$\delta^2 W$	second variation of W

Transpose:

W^T	transpose of W

Subscripts:

For any variable W

$W_o = (W)_{t_o} = W(t_o)$ value of W at the initial time

$W_f = (W)_{t_f} = W(t_f)$ value of W at the final time

Norm:

The Euclidean norm of the error in satisfying the term terminal constraints is used as follows:

$$||\text{Error}|| = \Big|\Big| \; ||\Sigma_f|| + ||M_f|| + \Omega_f^2 \; \Big|\Big|$$

SYMBOLS

Theoretical Developments

Indices:

n	number of state variables
m	number of control variables
p	number of initial constraints
q	number of terminal constraints

Scalars:

t	independent variable time
J	real functional which is to be minimized
I	augmented functional
G	terminal payoff quantity (the classical Mayer term)
Q	classical Lagrange term
P	augmented terminal payoff quantity
Γ	initial valued function
H	variational Hamiltonian
$\tilde{H}$	variational Hamiltonian for the Modified Sweep Method
Ω	classical transversality condition associated with free final-time problems
s, ϕ	auxiliary Riccati variables
β, τ_4, τ_5	terminal boundary values
P_t	terminal boundary value

Vectors

x	n-vector of state variables
$\bar{x}$	n-vector of state histories for stationary trajectory
$\bar{x}_o$	p-vector of initial state values
$\bar{x}_f$	q-vector of terminal state values
f	n-vector of functions
N	p-vector of initial conditions
M	q-vector of terminal conditions
μ	p-vector of constant Lagrange multipliers
ν	q-vector of constant Lagrange multipliers
λ	n-vector of time-dependent Lagrange multipliers
$\bar{\lambda}$	n-vector of Lagrange multiplier histories for stationary trajectory
u	m-vector of control variables
$\bar{u}$	m-vector of control histories for stationary trajectory
$\overset{*}{u}$	m-vector of optimal control histories
Σ	n-vector of dissatisfaction associated with transversality conditions for the Lagrange multipliers
H_u	m-vector of control optimality conditions
H_x, H_λ	n-vectors of Hamiltonian partial derivatives with respect to state and Lagrange multipliers
v, ω	inhomogeneous terms in perturbation equations
ℓ, y, η	n-vectors of auxiliary Riccati variables
g, z, ζ	q-vectors of auxiliary Riccati variables
ρ	n+q+1 vector of Riccati variables

$\alpha, \tau_1 \ \tau_2$	n-vectors of terminal boundary values
τ_3	q-vector of terminal boundary values
P_x	n-vector of terminal boundary values

Matrices:

$H_{ux}, H_{uu}, H_{u\lambda}$	partial derivatives of H_u with respect to state, control, and Lagrange multiplier vectors
$H_{xx}, H_{x\lambda}$	partial derivatives of H_x with respect to state and Lagrange multiplier vectors
$\tilde{H}_{xx}$	partial derivatives of $\tilde{H}_x$ with respect to state vector
$\tilde{H}_{\lambda x}, \tilde{H}_{\lambda\lambda}$	partial derivatives of $\tilde{H}_\lambda$ with respect to state and Lagrange multiplier vectors
A, B, C	n x n square perturbation equations matrices
K	n x n square matrix of Riccati variables
D	n x q matrix of Riccati variables
E	q x n matrix of Riccati variables
F	q x q square matrix of Riccati variables
I_n	n x n identity matrix
P_{xx}	n x n matrix of terminal boundary values
M_x	q x n matrix of terminal boundary values
R	n+q+1 square matrix of Riccati variables

Problem Applications

Brachistochrone:

x, y	position coordinates
v	particle speed
u	control angle
g	acceleration due to gravity
a	problem parameter

Reentry:

m	spacecraft mass
ρ_o	reference density for atmosphere
R	earth radius
$\overset{*}{\beta}$	atmospheric model parameter
S	cross sectional area of the spacecraft
V	speed
C_L, C_D	lift and drag coefficients
$\tilde{\lambda}_o$	weight number for heating term
h	altitude
θ	longitude
Δ	latitude
γ	angle of attack
A	heading angle
G	force due to gravity
$\bar{D}$	drag force per unit mass
$\bar{L}$	lift force per unit mass

β	roll angle for the spacecraft
μ	gravitational constant

CHAPTER 1

INTRODUCTION

Optimal control of spacecraft trajectories requires obtaining an optimal (maximum or minimum) value for an appropriate scalar quantity. This scalar quantity measures spacecraft performance and is called the *performance index*. In addition, terminal conditions such as might be specified for intercept or rendezvous problems must often be satisfied simultaneously.

After the problem has been formulated mathematically, several conceptual approaches are available to obtain the conditions required to solve the optimization problem. Among the more usual approaches are the calculus-of-variations, dynamic programming, and Pontryagin's Principle.

The calculus-of-variations is considered here because specific optimal control problems can be considered as particular cases of the more generalized Bolza problem of the classical calculus-of-variations. The powerful results associated with this classical theory thus are available for attacking optimal control problems.

The calculus-of-variations approach yields a nonlinear two-point boundary value problem for which closed-form solutions are usually not possible. Sophisticated numerical procedures have been developed because of the need to solve these problems. These procedures have become feasible in the last decade because of the development of large-scale digital computers.

This chapter introduces the definitions and terminology used throughout the dissertation. A brief history of the development of the numerical optimization methods is given also with interest centering on the second-order variational methods. The class of trajectory optimization problems to be solved is stated, as well as the associated nonlinear two-point boundary value problem obtained from using a calculus-of-variations approach.

Chapter II discusses (1) the second-order variation approach used to solve the nonlinear two-point boundary value problem, (2) the general set of perturbation equations for second-order methods, and (3) techniques used to achieve an integration of these perturbation equations.

Chapter III presents the Modified Sweep Method based on the generalized Riccati transformation. Chapter IV develops the linear feedback control law for the Modified Sweep Method. The numerical results obtained using this new method are discussed in Chapter V, with conclusions and recommendations presented in Chapter VI.

1.1 Definition of Terms Used

The MSM (Modified Sweep Method) is obtained from the SSM (Successive Sweep Method) by requiring that the control satisfy both local optimality and strengthened Legendre-Clebsch condition over the entire time interval of interest. This optimal control is then eliminated from the Hamiltonian for the problem and the restructured Hamiltonian used to obtain the nonlinear differential equations for both the state and Lagrange multiplier variables. As in the case of the SSM, the MSM then uses the generalized Riccati transformation to solve the linearized two-point boundary value problem in terms of the state and Lagrange multiplier perturbations. This is done while simultaneously relaxing the terminal transversality requirements on the time-dependent Lagrange multipliers. It is desirable to compare the proposed MSM to other existing numerical optimization methods. For this reason, a study was made of several well-known methods which appear in the literature. This author felt that these methods were representative of the properties contained in the set of variational methods for the numerical solution of optimization problems. It is emphasized that only a representative portion of the total number of existing numerical methods has been used for this study. In addition, the second-order methods intentionally have been selected more extensively than the first-order methods. The generalizations made, therefore, pertain only to those methods contained in Section 1.2 on the historical development of numerical optimization methods.

This study of the selected group of existing methods revealed a set of properties which can be employed to describe the characteristic features for each method. These properties have been used to specify, arbitrarily, twelve classes of numerical optimization methods. Each method can then be identified as belonging to a particular class according to the following properties:

(1) the *ORDER* (first or second) of the theory upon which the method is based.

(2) the *APPROACH* (direct or indirect) used by the method to compute the required corrections.

(3) the *ITERATION PROCESS* (interval, boundary or hybrid) used by the method.

The following definitions are used:

Definition 1: Order of the Method

A method is described as *first order* if it is based only upon the theory of the first variation for a real functional. If a method is based upon the theory of the second variation for a real functional, it is described as a *second order* method.

Definition 2: Approach of the Method

A method is said to take a *direct approach* if the required corrections (state, control or Lagrange multipliers) are computed such that the performance index for the augmented variational problem is itself directly affected in some manner to expedite convergence. If a method chooses to compute the required corrections based on the set of first-order necessary conditions required for optimality with respect to the control, then the method is said to take an *indirect approach*.

Definition 3: Iteration Process for the Method

A method is said to use an *interval value iteration process* if the end result of a particular iteration is the computation of corrections to the variables (state, control, or Lagrange multipliers) over the entire time interval of interest. Methods which compute corrections to these same variables at a boundary only are said to use a *boundary value iteration process*. If a method combines both an interval and boundary value iteration process, it is described as using a *hybrid iteration process*.

Definition 4: Convergence for a Method

(a) Control Function Iteration Method. Given an arbitrary numerical tolerance ε, a control function iteration method is said to have achieved convergence if

$$\left|\left| \; ||\overset{*}{H}{}_u^T|| + ||M_f|| + \Omega_f^2 \; \right|\right| \leq \varepsilon$$

where

$$||\overset{*}{H}{}_u^T|| \triangleq \max_{u(t)} \{\text{Abs}[H_u^T(x,u,\lambda,t)]\} \qquad t_o \leq t \leq t_f$$

(b) Boundary Value Iteration Method. Given a numerical tolerance ε, a boundary value iteration method is said to have achieved convergence if

$$\left|\left| \; ||\Sigma_f|| + ||M_f|| + \Omega_f^2 \; \right|\right| \leq \varepsilon \; .$$

The symbols H, Σ_f, M_f and Ω_f are defined in Section 1.4.

Using these definitions, Table I summarizes the twelve classes of numerical optimization methods extracted from the methods chosen as representative for the study. Reference numbers specify major studies in each class while the acronyms (SSM, etc.) identify the particular class for the three methods to be compared in detail.

TABLE I

CLASSES OF NUMERICAL OPTIMIZATION METHODS

CLASS	ORDER	APPROACH	ITERATION PROCESS	REFERENCES
1	1	Direct	Hybrid	
2	1	Direct	Interval value	4,12
3	1	Direct	Boundary value	
4	1	Indirect	Hybrid	
5	1	Indirect	Interval value	
6	1	Indirect	Boundary value	
7	2	Direct	Hybrid	
8	2	Direct	Interval value	19,26
9	2	Direct	Boundary value	35
10	2	Indirect	Hybrid	3,13
11	2	Indirect	Interval value	23,27 (SSM)
12	2	Indirect	Boundary value	10,11,16 (MSM,MPF)

As shown in Table I, the class of second-order methods which take a direct approach in computing the desired corrections and implement a hybrid iteration process was not represented in the methods chosen for the study. Furthermore, Classes 1,3,4,5 and 6, all first-order methods, were also not represented. This can be attributed to the fact that second-order methods were of primary interest in this study.

The similarities and differences between the three numerical optimization methods (SSM, MSM and MPF) to be discussed are now obvious. All three are second-order methods which use an indirect approach in computing the required corrections for the state, control or Lagrange multiplier variables. The SSM falls into Class 11 because it uses an interval iteration process. Both the MSM and MPF fall into Class 12 because each uses a boundary iteration process.

A historical development of the methods chosen for the study is detailed now for reference purposes.

1.2 Historical Information

First-Order Methods. The first numerical procedure for solving control optimization problems which generated active interest was developed independently by Kelley[12] and Bryson and Denham[4]. Their research extended the concept of steepest descent developed earlier by Courant[5]. The Class 2 method was based on the first-order variation of a scalar functional, with a control function assumed for the time interval of interest. Corrections to this control were then computed iteratively using an ordinary gradient technique. Applications showed that the method was easy to implement and tended to convergence with even gross initial control estimates.

The method, however, possesses two undesirable features. First, the convergence rate decreased asymptotically during the terminal stages of iteration. Second, once convergence was achieved, the control obtained was only within a numerical tolerance of the Eulerian control.

Due to the first undesirable feature, numerical procedures to increase the convergence rate flourished. These were all first-order methods and are not discussed in this research. Both features led to the development of the second-order methods which sought to increase convergence as well as provide the true Eulerian control.

Second-Order Methods. Jurovics and McIntyre[11] solved the two-point boundary value problem of trjaectory optimization by using the equations which are adjoint to the linearized Euler-Lagrange equations. Their method was called the Adjoint Method (Method of Adjoint Functions), and extended the work of Goodman and Lance[7] to allow for variable terminal time. The Adjoint Method is a Class 12 method wherein an indirect approach is used to compute the desired corrections while iterating on the initial boundary values of the Lagrange multipliers.

Breakwell, Speyer, and Bryson[3] developed a "second variation" method (Class 10) to solve control optimization problems. Kelley, Kopp, and Moyer[13] also developed a "second variation" method similar to the previous one. Jazwinski[10] developed a modified adjoint method equivalent to the second-variation method of Breakwell, Speyer, and Bryson[3] by extending the method of Jurovics and McIntyre[11]. Jazwinski's method had the specific advantage, however, of requiring considerably less storage. Furthermore, it required less computer time in that fewer integrations of an equivalent set of equations were necessary.

McGill and Kenneth[20] developed the Generalized Newton-Raphson Operator Method for solving two-point boundary value problems. This method falls into Class 11, which uses the indirect approach linked with an interval value iteration process. A proof of quadratic convergence for the method had previously been given by these same authors[21]. This method's major drawback was the laborious manner in which corrections to the final time value were computed.

An alternate approach, the Modified Generalized Newton-Raphson Method, using the same method was developed by Long[17]. To eliminate the awkward handling of free final time, a change of independent variable was used for the free-final-time corrections. Another method based on the Newton-Raphson approach, but incorporating a better technique for computing the free-final-time corrections, is the Modified Quasilinearization Method developed by Lewallen[16]. Sylvester and Meyer[36] have also used the Newton-Raphson approach, calling it quasilinearization.

A method based on the theory of both the first and second variations was devised by Merriam[26]. This is a Class 8 method in which a direct approach is taken for computing corrections to the control functions assumed throughout the time interval of interest. This particular method was instrumental in the development of the successive sweep method discussed in the next paragraph.

McReynolds and Bryson[25] introduced the successive sweep method for solving optimal control problems. Although the method is a second-order method, the Eulerian control requirement is relaxed. The method is based on the generalized Riccati transformation and falls into Class 11 (Table I). A similar method called successive approximation was developed by Mitter[27]. He also showed the formal equivalence of this method to Newton's Method.

Lewallen[16] also introduced the Method of Perturbation Functions (MPF), based on previous work by Breakwell, et al.,[3] and Jazwinski[10]. The method falls into Class 12. Lastman[14] has shown the equivalence of all these methods to Newton's Method.

Sutherland and Bohn[35] have recently developed a method which falls into Class 9, which uses a direct approach to compute boundary corrections for the initial values of the multipliers.

Mayne[19] developed a second-order method which falls into Class 8 (Table I). However, a dynamic programming technique is used to attack the optimization problem instead of a calculus-of-variations technique.

Jacobson[9] extended the Class 11 features into a new second-order algorithm through use of a differential dynamic programming technique. His method generalizes the successive sweep methods of McReynolds[23] and Mitter[27].

The development of the MSM completes the historical development for the numerical optimization methods chosen.

As was mentioned previously, it is now desirable to compare the MSM to both the SSM and the MPF. Toward this end, a general class of control optimization problems is first chosen. This class of problems is presented in the next section.

1.3 Class of Control Optimization Problems to be Solved

Posed as a special form of the Bolza problem from the calculus-of-variations (see Bliss[1]), the general class of control optimization problems to be solved is stated as follows:

In the time interval $t_o \leq t \leq t_f$, find an m-vector of control variables $u(t)$ to minimize the real functional,

$$J(u) = G(x_f,t_f) + \int_{t_o}^{t_f} Q(x,u,t)\,dt \quad . \tag{1}$$

subject to the n-vector of differential constraints,

$$\dot{x} = f(x,u,t) \tag{2}$$

while satisfying the p-vector of known initial conditions

$$N(x_o,t_o) = 0 \tag{3}$$

and the q-vector of desired terminal conditions

$$M(x_f,t_f) = 0 \tag{4}$$

The control and state variables in the following discussion are assumed to be defined on completely open regions and thus are not subject to inequality constraints.

1.4 Associated Nonlinear Two-Point Boundary Value Problem

Proceeding in the usual calculus-of-variations manner for solving the Bolza problem of control optimization stated in the previous section, an augmented functional denoted as I is first formed. This augmented functional has the property of being formally equivalent to the original functional; it incorporates the desired auxiliary conditions through the use of Lagrange multipliers. To form this augmented functional, the n-vector of Lagrange multipliers $\lambda(t)$ and the p and q vectors of constant Lagrange multipliers μ and ν adjoin the desired auxiliary conditions to the original functional as follows:

$$I = J + \mu^T N + \nu^T M + \int_{t_o}^{t_f} \lambda^T (f - \dot{x})\, dt \tag{5}$$

For convenience of notation, this functional is rewritten as

$$I = P + \Gamma + \int_{t_o}^{t_f} (H - \lambda^T\dot{x})\, dt \tag{6}$$

where

$$P = P(x_f,\nu,t_f) \triangleq G(x_f,t_f) + \nu^T M(x_f,t_f)$$

$$\Gamma = \Gamma(x_o,\mu,t_o) \triangleq \mu^T N(x_o,t_o)$$

$$H = H(x,u,\lambda,t) \triangleq Q(x,u,t) + \lambda^T f(x,u,t)$$

and $x = x(t)$, $u = u(t)$, $\lambda = \lambda(t)$

The scalar H is the variational Hamiltonian for this class of problems.

Necessary Conditions. The set of first-order necessary conditions which must be satisfied by the extremal control for the augmented functional of the type above is obtained by requiring the first variation of this functional to vanish. These conditions are well documented in the literature (for example Bliss[1], Hestenes[8], Pontryagin[30], et al., Tapley and Lewallen[38]). In summary, these conditions are

$$t_o \le t \le t_f \left\{ \begin{array}{ll} \dot{x} - H_\lambda^T(x,u,\lambda,t) = 0 & (7) \\ \dot{\lambda} + H_x^T(x,u,\lambda,t) = 0 & (8) \\ H_u^T(x,u,\lambda,t) = 0 & (9) \end{array} \right.$$

$$t = t_o \begin{cases} \Sigma_o^T \triangleq (\Gamma_x + \lambda^T)_{t_o} = 0 & (10) \\ N(x_o, t_o) = 0 & (11) \\ \Omega_o \triangleq (\Gamma_t - H)_{t_o} = 0 & (12) \end{cases}$$

$$t = t_f \begin{cases} \Sigma_f^T \triangleq (P_x - \lambda^T)_{t_f} = 0 & (13) \\ M(x_f, t_f) = 0 & (14) \\ \Omega_f \triangleq (P_t + H)_{t_f} = 0 & (15) \end{cases}$$

Equations (7) through (9) constitute $2n+m$ Euler-Lagrange equations for this class of problems. Equations (11) and (14) are the $p+q$ specified initial and final values of the problem state variables. The remaining equations form the $2n+2$ set of classical transversality conditions from the calculus-of-variations.

The control optimization problem thus is posed as a nonlinear two-point boundary value problem for the $2n+m$ variables $x(t)$, $u(t)$, and $\lambda(t)$ and the $p+q+2$ parameters μ, ν, t_o, and t_f in terms of the $2n$ differential equations (7) and (8), the m algebraic equations (9), and the $2n+p+q+2$ conditions in equations (10) through (15).

It is assumed that the initial time t_o and n values of the initial state $x(t_o) = \bar{x}_o$ are specified. Equations (10) and (12) are then identically satisfied and therefore disregarded in subsequent discussions.

Sufficiency Conditions. To ensure that the control satisfying these first-order necessary conditions does indeed generate a weak minimizing solution, the second-order variation of the augmented functional I(u) must be positive everywhere in the interval of interest when it is evaluated along an extremal trajectory (Gelfand and Fomin[6]). This requirement leads to the following additional set of second-order conditions:

1. Strengthened Legendre-Clebsch Condition

   The strengthened Legendre-Clebsch condition must be satisfied everywhere in the interval of interest. Specifically, for any arbitrary change in control $\delta u(t)$,

   $$\delta u^T H_{uu} \delta u > 0 \tag{16}$$

   is required.

2. Jacobi (Mayer) Conjugate Point Condition

   The Jacobi (Mayer) conjugate point condition must be satisfied everywhere in the interval of interest. This requires that no two points exist in the interval $t_o \leq t \leq t_f$ which are conjugate to one another.

The following restrictions on the definitions presented in Section I are subsequently assumed in this research: interval iteration corresponds to control function iteration and boundary iteration corresponds to Lagrange multiplier iteration.

CHAPTER 2

THE PERTURBATION EQUATIONS

The second-order variational methods seek to solve the nonlinear two-point boundary value problem associated with trajectory optimization by solving an equivalent linearized problem in terms of perturbations in the problem variables. The six classes of second-order methods are distinguished by *approach* and *iteration process*. Furthermore, each method for a given class is distinguished by the technique used to perform the numerical integration of the perturbation equations which are obtained from a first-order perturbation of the Euler-Lagrange equations, viz., Equations (7) through (9). These perturbation equations can assume one of two forms with each form patterned by the iteration process selected for a given method.

As in the case of the SSM (Successive Sweep Method), if a control function iteration process is used, a "PE" scheme is used to obtain the perturbation equations. The acronym "PE" designates the following procedure: Perturb the Euler-Lagrange equations and then Eliminate the control perturbations. If a Lagrange multiplier boundary value iteration process is used, an "EP" scheme is used to obtain the perturbation equations. The acronym "EP" similarly designates the following procedure: Eliminate the optimal control and then Perturb the resulting Euler-Lagrange equations. These two schemes for generating the perturbation equations for second-order methods are detailed below.

2.1 The PE Scheme

The PE Scheme is associated with control function iteration schemes such as the SSM. It proceeds by first perturbing the first-order necessary conditions for stationary control. Then the perturbations in the control are eliminated from the perturbed Euler-Lagrange equations for the state and Lagrange multipliers and the appropriate transversality conditions. This scheme tacitly assumes that the matrix H_{uu} is nonsingular everywhere in the time interval of interest. The set of perturbation equations associated with this scheme is outlined in Appendix A and are summarized here as

$$\begin{pmatrix} \delta\dot{x} \\ \\ \delta\dot{\lambda} \end{pmatrix} = \begin{pmatrix} A & B \\ \\ -C & -A^T \end{pmatrix} \begin{pmatrix} \delta x \\ \\ \delta\lambda \end{pmatrix} + \begin{pmatrix} v \\ \\ -w \end{pmatrix} \tag{17}$$

where

$$A = -H_{\lambda u} H_{uu}^{-1} H_{ux} + H_{\lambda x}$$

$$B = -H_{\lambda u} H_{uu}^{-1} H_{u\lambda}$$

$$C = -H_{xu} H_{uu}^{-1} H_{ux} + H_{xx}$$

$$v = H_{\lambda u} H_{uu}^{-1} \delta H_u^T$$

$$w = H_{xu} H_{uu}^{-1} \delta H_u^T$$

and $H = H(x,\lambda,u,t)$ while the Hamiltonian partials $H_{\lambda u}$, H_{xu}, etc., are evaluated on the known trajectory. The known trajectory is called a nominal (or reference) trajectory.

Note that for this scheme, the system of perturbation equations assumes the form of a set of inhomogeneous first-order linear differential equations with time-dependent coefficient matrices.

2.2 The EP Scheme

The EP Scheme is used by the boundary value iteration methods. It proceeds by first using the control optimality condition $H_u^T = 0$ and the strengthened Legendre-Clebsch condition $\delta u^T H_{uu} \delta u > 0$ to eliminate the control from the Euler-Lagrange equations for the state and Lagrange multipliers, as well as the appropriate transversality conditions. The revised set of equations are perturbed then to obtain the following homogeneous set of first-order linear differential equations.

$$\begin{pmatrix} \delta\dot{x} \\ \\ \delta\dot{\lambda} \end{pmatrix} = \begin{pmatrix} \tilde{A} & \tilde{B} \\ \\ -\tilde{C} & -\tilde{A}^T \end{pmatrix} \begin{pmatrix} \delta x \\ \\ \delta\lambda \end{pmatrix} \qquad (18)$$

where

$$\tilde{A} = \tilde{H}_{\lambda x}$$

$$\tilde{B} = \tilde{H}_{\lambda\lambda}$$

$$\tilde{C} = \tilde{H}_{xx}$$

$$\tilde{H} = H\left\{x(t),\ \lambda(t),\ \overset{*}{u}(x,\lambda,t),t\right\}$$

and $\overset{*}{u}(x,\lambda,t)$ is the control obtained using the conditions that $H_u^T = 0$ and $H_{uu} > 0$. It is important to note that this scheme involves the assumption that the Hamiltonian H for the particular problem is structured such that $H_u^T = 0$ and $H_{uu} > 0$ can be used to obtain the explicit relation $\overset{*}{u}(x,\lambda,t)$. Implicitly, such a relation is assured if $H_u^T = 0$ and $H_{uu} > 0$; however, such an explicit relation may be impossible to obtain for some problems.

2.3 Integrating the Set of Perturbation Equations

It has been pointed out previously that second-order methods within a given class differ only in the technique that is used to integrate the set of perturbation equations. The two techniques presently available for accomplishing this integrating are detailed below.

Explicit Integration. Methods using this technique choose to integrate directly the perturbation equations to obtain the perturbed values over the interval of interest. Some investigators have found (see for example the work of Merriam[26]) that these methods suffer from numerical instabilities. Instability here is used in the sense that small errors in numerical precision will become exponentially very large over a long interval of numerical integration. The nature of these instabilities is associated with the numerical integration for coupled systems of linear differential equations which have split boundary conditions and admit both increasing and decreasing exponential solutions.

Implicit Integration. Methods which presently use this technique are based upon a transformation process such as the generalized Riccati transformation. The technique consists of bypassing direct integration of the perturbation equations, and integrating a set of auxiliary variables. These in turn can be used to compute the perturbed values for the variables in the perturbation equations. Several advantages are claimed for this technique. First, the differential equations for the new auxiliary variables have been reported to be more stable numerically than the original perturbation equations. Second, these auxiliary variables contain additional intrinsic information about the optimal trajectory for the problem being solved.

A strong case concerning increased numerical stability in first-order linear two-point boundary value problems has been made by Rybicki and Usher[31]. However, work by Williamson[39] and this author has revealed that problems which have large differences in sign and magnitude for the eigenvalues of the coefficient matrix in the linear system of equations do not behave well numerically with the generalized Riccati transformation technique. This author's opinion is that a valid generalized statement is yet to be made concerning the numerical stability properties of the Riccati transformation technique.

That the auxiliary variables could contain additional intrinsic information certainly proves to be true. The earlier methods lacked in one respect: after convergence had been achieved, they required that post-convergence procedures be used to test the Legendre-Clebsch condition and/or the Jacobi-Mayer conjugate point condition. Those methods which used the "EP" Scheme to generate the set of perturbation equations automatically took the Legendre-Clebsch condition into account when eliminating the control from the set of Euler-Lagrange equations. However, the Jacobi-Mayer conjugate point condition still must be tested. This condition was often ignored and the converged solution was assumed to be a local optimum.

However, using the generalized Riccati transformation technique on the perturbation equations provides the advantage of additional information for the current reference trajectory. Information is contained among these auxiliary Riccati variables for testing the Jacobi-Mayer conjugate point and abnormality conditions from the calculus-of-variations (McReynolds[23]). It is well-known that the existence of a conjugate point precludes a trajectory from being optimal. The existence of such a conjugate point can be automatically detected continuously during the backward sweep process. This is accomplished by use of the fact that the matrix solution to the Riccati differential equation becomes unbounded at a conjugate point.

On the other hand, the abnormality condition is equivalent to a certain matrix of these auxiliary Riccati transformation variables becoming singular at the initial time. This information is important because such a condition is tantamount to the inability in making corrections for values of the terminal constraints. This abnormality condition occurs for the Bolza problem if the boundary conditions at the final time are not linearly independent (McReynolds[23]).

This leads to speculation concerning additional information about the reference trajectory which might be contained in the other Riccati transformation variables, either individually or in some combined form. To this author's knowledge, little work has been done in attempting to extract such additional information.

2.4 The Generalized Riccati Transformation Technique

The generalized Riccati transformation is a transformation which changes the original two-point boundary value problem in terms of the coupled linear system of differential equations to an initial-value problem having uncoupled variables and boundary conditions. This initial value problem is now stated in terms of n original problem variables, and in the general case, a total of $\left[n(n+1) + q(q+1)\right] / 2 + \left[(nxq) + 3(n+q) + 2\right]$ auxiliary Riccati variables where n is the number of state variables and q is the number of terminal constraint. Since the coupled system of perturbation equations is integrated implicitly by integrating these auxiliary variables, it was expected that the differential equations for the auxiliary variables would be numerically more stable than the original equations. Furthermore, Breakwell and Ho[2] have shown agreement with McReynolds[23] in that the conjugate point condition is related directly to the boundedness of an (nxn) matrix of Riccati variables which must satisfy a matrix version of the scalar Riccati equation over the time interval of interest.

This transformation approach proceeds to solve the original nonlinear two-point boundary value control optimization problem in the following manner. A solution to the original nonlinear problem is assumed, and the corresponding terminal conditions are obtained. In general, these conditions are not satisfied to within the specified error tolerances. Desired changes, in these terminal conditions are specified, and the generalized Riccati transformation is used then to generate a linearized field of solutions about this assumed solution. The transformation allows the specified changes in the set of terminal conditions to be mapped back to the initial time, when the particular member of the field that also satisfies the initial conditions is selected. A new solution to the original nonlinear two-point boundary value problem is then computed using the linearized corrections, which are obtained through use of the auxiliary Riccati variables. As before, the new solution does not satisfy the desired terminal conditions exactly due to the linearity assumptions. However, the process can be applied iteratively until the desired terminal conditions are satisfied to within a suitable error tolerance.

Historical Background. A Riccati transformation technique was first used by Gelfand and Fomin[6] in their successive sweep procedure of solving two-point boundary value problems for linear inhomogeneous systems of second-order differential equations. The same transformation was generalized and discussed for systems of first-order equations by Rybicki and Usher.[31] McReynolds[23,24] and Mitter[27] used the successive sweep method with the generalized Riccati transformation technique to solve the nonlinear two-point boundary value problem of control optimization. Schley and Lee[32] have developed a Newton-Raphson method which uses the Riccati transformation technique. Speyer and Byrson[34] have extended the concept of the Riccati variables for the case when some of these variables become unbounded. Narha and Berry[28] and Omicioli[29] have applied the successive sweep method of McReynolds to the shaping of optimal finite-thrust orbit transfer trajectories for which the control function is characterized by discontinuities. McGregor[22] has used the same method but has introduced modifications to handle problems with inequality constraints which contain the control explicitly. Most recently, Longmuir and Bohn[18] have shown how this technique can be used with any second-order method.

Analytical Development. The generalized Riccati transformation for the linearized control optimization problem can be written in matrix form as

$$\begin{bmatrix} \delta\lambda(t) \\ dM_f \\ d\Omega_f \end{bmatrix} = R(t) \begin{bmatrix} \delta x(t) \\ d\nu \\ dt_f \end{bmatrix} + \rho(t) \tag{19}$$

where dM_f, $d\Omega_f$, $d\nu$, and dt_f are constants for a particular iteration and

$$R \triangleq \begin{bmatrix} K & D & \ell \\ E & F & g \\ y^T & z^T & s \end{bmatrix} \qquad \rho \triangleq \begin{bmatrix} \eta \\ \zeta \\ \phi \end{bmatrix}$$

where K, E, y^T respectively map given state perturbations $\delta x(t)$ into changes in the multipliers $\delta\lambda(t)$, terminal state dissatisfactions dM_f, and terminal Hamiltonian transversality dissatisfaction $d\Omega_f$ and

K(t) is an nxn symmetric matrix

E(t) is a qxn matrix

y(t) is an n-vector

Also, D, F, z^T, respectively map changes in the multipliers $d\nu$ into changes in the multipliers $\delta\lambda(t)$, terminal state dissatisfactions dM_f, and Hamiltonian terminal dissatisfaction $d\Omega_f$, and

D(t) is an nxq matrix

F(t) is a qxq symmetric matrix

z(t) is a q-vector

The scalars ℓ, g and s respectively map changes in the final time dt_f into changes in the multipliers $\delta\lambda(t)$, terminal state dissatisfactions dM_f and the dissatisfaction in the terminal Hamiltonian, $d\Omega_f$.

The quantities η, ζ and ϕ respectively map changes in terminal local optimality dissatisfaction or terminal transversality dissatisfactions by the multipliers into changes $\delta\lambda(t)$, dM_f, and $d\Omega_f$.

Differentiating the generalized Riccati transformation (Eq. 19) with respect to time gives

$$\begin{pmatrix} \delta\dot{\lambda} \\ 0 \\ 0 \end{pmatrix} = \dot{R} \begin{pmatrix} \delta x \\ d\nu \\ dt_f \end{pmatrix} + R \begin{pmatrix} \delta\dot{x} \\ 0 \\ 0 \end{pmatrix} + \dot{\rho} \qquad (20)$$

Expanding the middle term on the right and transposing to the left yields

$$\begin{pmatrix} I_n & -K \\ 0 & -E \\ 0 & -y^T \end{pmatrix} \begin{pmatrix} \delta\dot{\lambda} \\ \delta\dot{x} \end{pmatrix} = \dot{R} \begin{pmatrix} \delta x \\ d\nu \\ dt_f \end{pmatrix} + \dot{\rho} \qquad (21)$$

Using the perturbation equations, Equations (17), to eliminate $\delta\dot{\lambda}$ and $\delta\dot{x}$ leads to the following expression

$$\begin{pmatrix} I_n & -K \\ 0 & -E \\ 0 & -y^T \end{pmatrix} \left\{ \begin{pmatrix} -A^T & -C \\ B & A \end{pmatrix} \begin{pmatrix} \delta\lambda \\ \delta x \end{pmatrix} + \begin{pmatrix} -w \\ v \end{pmatrix} \right\} = \dot{R} \begin{pmatrix} \delta x \\ d\nu \\ dt_f \end{pmatrix} + \dot{\rho} \qquad (22)$$

Now

$$\begin{bmatrix} -A^T & -C \\ B & A \end{bmatrix} \begin{bmatrix} \delta\lambda \\ \delta x \end{bmatrix} = \begin{bmatrix} -C & -A^T \\ A & B \end{bmatrix} \begin{bmatrix} \delta x \\ \delta\lambda \end{bmatrix} \tag{23}$$

Using the first row of the Riccati transformation, Equation (19), the following relation can be readily obtained.

$$\begin{bmatrix} \delta x \\ \delta\lambda \end{bmatrix} = \begin{bmatrix} I_n & 0 & 0 \\ K & D & \ell \end{bmatrix} \begin{bmatrix} \delta x \\ d\nu \\ dt_f \end{bmatrix} + \begin{bmatrix} 0 \\ \eta \end{bmatrix} \tag{24}$$

Substituting Equations (23) and (24) back into Equation (22) gives

$$\begin{bmatrix} I_n & -K \\ 0 & -E \\ 0 & -y^T \end{bmatrix} \left\{ \begin{bmatrix} -C & -A^T \\ A & B \end{bmatrix} \left(\begin{bmatrix} I_n & 0 & 0 \\ K & D & \ell \end{bmatrix} \begin{bmatrix} \delta x \\ d\nu \\ dt_f \end{bmatrix} + \begin{bmatrix} 0 \\ \eta \end{bmatrix} \right) + \begin{bmatrix} -w \\ v \end{bmatrix} \right\}$$

$$= \dot{R} \begin{bmatrix} \delta x \\ d\nu \\ dt_f \end{bmatrix} + \dot{\rho} \tag{25}$$

Multiplying and collecting terms for arbitrary δx, $d\nu$ and dt_f, the following equations must hold

$$\dot{R} = -S^T W T \tag{26}$$

$$\dot{\rho} = -S^T r \tag{27}$$

where

$$S \triangleq \begin{pmatrix} I_n & 0 & 0 \\ K & E^T & y \end{pmatrix}$$

$$T \triangleq \begin{pmatrix} I_n & 0 & 0 \\ K & D & \ell \end{pmatrix}$$

$$W \triangleq \begin{pmatrix} C & A^T \\ A & B \end{pmatrix}$$

and

$$r \triangleq \begin{pmatrix} A^T\eta + w \\ B\eta + v \end{pmatrix}$$

Performing the matrix multiplications yields the familiar set of equations for the Riccati variables.

$$\frac{d}{dt}\begin{pmatrix} K & D & \ell \\ E & F & g \\ y^T & z^T & s \end{pmatrix} =$$

$$- \left(\begin{array}{c|c|c} (A^T + KB)\ K + KA + C & (A^T + KB)\ D & (A^T + KB)\ell \\ E(A + BK) & EBD & EB\ell \\ y^T(A + BK) & y^T BD & y^T B\ell \end{array}\right) \tag{28}$$

$$\frac{d}{dt}\begin{pmatrix}\eta \\ \zeta \\ \phi\end{pmatrix} = -\begin{pmatrix}(A^T + KB)\eta + (Kv + w) \\ E(B\eta + v) \\ y^T(B\eta + v)\end{pmatrix} \tag{29}$$

From Equation (28), the following rates of change for the Riccati variables are found to be equal: $\dot{E}^T(t) = \dot{D}(t)$, $\dot{y}(t) = \dot{\ell}(t)$ and $\dot{z}(t) = \dot{g}(t)$. If, then, at the terminal boundary $E^T(t_f) = D(t_f)$, $y(t_f) = \ell(t_f)$ and $z(t_f) = g(t_f)$, the following will be true: $E^T(t) = D(t)$, $y(t) = \ell(t)$ and $z(t) = g(t)$. This means not only that the matrix of Riccati variables $R(t)$ given in Equation (19) is symmetric but also that Equation (26) itself is also symmetric. In this case,

$$\dot{R} = -S^TWS \tag{30}$$

where S and W are defined on page 26.

Terminal Boundary Conditions. The derivations of the generalized set of terminal boundary conditions for the Riccati variables are presented in Appendix B. In summary, these boundary conditions are obtained from

$$\begin{pmatrix}\delta\lambda_f \\ dM_f \\ d\Omega_f\end{pmatrix} = \begin{pmatrix}(P_{xx})_f & M_{x_f}^T & \alpha_f \\ M_{x_f} & 0 & \dot{M}_f \\ \alpha_f^T + \tau_2 & \dot{M}_f^T + \tau_3 & \beta_f + \tau_4\end{pmatrix}\begin{pmatrix}\delta x_f \\ d\nu \\ dt_f\end{pmatrix} + \begin{pmatrix}\tau_1 \\ 0 \\ \tau_5\end{pmatrix} \tag{31}$$

The boundary conditions thus are

$$R(t_f) = \begin{bmatrix} K(t_f) & D(t_f) & \ell(t_f) \\ E(t_f) & F(t_f) & g(t_f) \\ y^T(t_f) & z^T(t_f) & s(t_f) \end{bmatrix}$$

$$= \begin{bmatrix} (P_{xx})_f & M^T_{x_f} & \alpha_f \\ M_{x_f} & 0 & \dot{M}_f \\ \alpha^T_f + \tau_2 & \dot{M}^T_f + \tau_3 & \beta_f + \tau_4 \end{bmatrix} \tag{32}$$

and

$$\rho(t_f) = \begin{bmatrix} \eta(t_f) \\ \zeta(t_f) \\ \phi(t_f) \end{bmatrix} = \begin{bmatrix} \tau_1 \\ 0 \\ \tau_5 \end{bmatrix} \tag{33}$$

where

$$\alpha_f \triangleq \left(\frac{DP^T_x}{Dt} + H^T_x\right)_{t_f}$$

$$\beta_f \triangleq \frac{D}{Dt}\left(\frac{DP}{Dt} + Q\right)_{t_f}$$

$$\tau_1 \triangleq -d\Sigma_f$$

$$\tau_2 \triangleq \left(-H_u H^{-1}_{uu}(H_{ux} + H_{u\lambda}P_{xx})\right)_{t_f}$$

$$\tau_3 \triangleq \left(-H_u H^{-1}_{uu} H_{u\lambda} M^T_x\right)_{t_f}$$

$$\tau_4 \quad \triangleq \quad -\left\{\Sigma_f^T \frac{Df}{Dt} + H_u H_{uu}^{-1} H_{u\lambda} \alpha_f\right\}_{t_f}$$

and
$$\tau_5 \quad \triangleq \quad \left\{H_u H_{uu}^{-1} \delta H_u^T - (\dot{x}^T - H_u H_{uu}^{-1} H_{u\lambda})\, d\Sigma_f\right\}_{t_f}$$

Interval Value Process. Methods using this process start by assuming a control function over the time interval of interest. In general then, $H_u^T(t) \neq 0$. For such methods, the Lagrange multipliers can be made to satisfy the transversality conditions identically; i.e., $\Sigma_f = 0$. The Successive Sweep Method is an example of a control function iteration process. The terminal boundary conditions for the Riccati variables are then obtained from

$$\begin{pmatrix} \delta\lambda_f \\ dM_f \\ d\Omega_f \end{pmatrix} = \begin{pmatrix} (P_{xx})_f & M_{x_f}^T & \alpha_f \\ M_{x_f} & 0 & \dot{M}_f \\ \alpha_f^T + \tau_2 & \dot{M}_f^T + \tau_3 & \beta_f + \tau_4 \end{pmatrix} \begin{pmatrix} \delta x_f \\ d\nu \\ dt_f \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ \tau_5 \end{pmatrix} \qquad (34)$$

where

$$\tau_2 = \left\{-H_u H_{uu}^{-1}(H_{ux} + H_{u\lambda} P_{xx})\right\}_{t_f}$$

$$\tau_3 = \left\{-H_u H_{uu}^{-1} H_{u\lambda} M_x^T\right\}_{t_f}$$

$$\tau_4 = \left\{-H_u H_{uu}^{-1} H_{u\lambda} \alpha_f\right\}_{t_f}$$

and
$$\tau_5 = \left\{H_u H_{uu}^{-1} \delta H_u^T\right\}_{t_f}$$

Note that the terminal matrix $R(t_f)$ of Riccati variables is symmetric only when the control function satisfies the local optimality condition that $H_u^T(t_f) = 0$. This same conclusion has been drawn by McGregor[22]. However, this contradicts the results presented by McReynolds and Bryson[25] and Mitter[27]. This argument needs to be resolved.

Boundary Value Process. These methods assumed that $H_u^T(t) = 0$, where $t_o \leq t \leq t_f$. In this case, initial values are guessed for the Lagrange multipliers, and subsequent corrections are computed iteratively. In general then, $\Sigma_f \neq 0$: likewise, $d\Sigma_f \neq 0$. The terminal boundary conditions in this case are obtained as

$$\begin{pmatrix} \delta\lambda_f \\ dM_f \\ d\Omega_f \end{pmatrix} = \begin{pmatrix} (P_{xx})_f & M_{x_f}^T & \alpha_f \\ M_{x_f} & 0 & \dot{M}_f \\ \alpha_f^T & \dot{M}_f^T & \beta_f - \Sigma_f^T \frac{Df}{Dt} \end{pmatrix} \begin{pmatrix} \delta x_f \\ d\nu \\ dt_f \end{pmatrix} + \begin{pmatrix} -d\Sigma_f \\ 0 \\ -\dot{x}_f^T d\Sigma_f \end{pmatrix} \qquad (35)$$

Note then that there is a basic difference in philosophy between a control function and a boundary value iteration process. In the first case, the optimality condition that $H_u^T = 0$ is relaxed at each point in the interval to ensure satisfaction of the transversality condition $\Sigma_f = 0$ by the Lagrange multipliers. In the second case, the optimality condition $H_u^T = 0$ is satisfied at each point in the interval while the terminal transversality $\Sigma_f = 0$ is relaxed on the multipliers.

CHAPTER 3

THE MODIFIED SWEEP METHOD

Merriam[26] and Mitter[27] have pointed out that boundary-condition iteration methods have certain programming advantages; viz., computer logic is relatively simple, and programming storage requirements are small. Furthermore, accurate trajectories are obtained in problems where these methods are successful. Experience has shown that such methods have one main disadvantage, viz., the numerical instability mentioned previously. The nature of this instability has been discussed by several researchers, among them Rybicki and Usher.[31] Since the Riccati transformation technique attempts to circumvent this problem by dealing with new uncoupled variables, this approach enhances the desirable features already known about boundary iteration schemes.

3.1 Differential Equations

For the modified sweep method, it is assumed that the Euler-Lagrange equations are satisfied over the time interval of interest. Furthermore, the optimality condition that $H_u^T = 0$ is assumed to yield an explicit expression for the control in terms of the other variables. The Legendre-Clebsch condition is then used to yield the extremal control

$$\overset{*}{u} = u(x,\lambda,t) \tag{36}$$

This expression can now be used to eliminate the control from the original nonlinear Euler-Lagrange equations for x and λ as well as from the appropriate transversality conditions. The set of first-order necessary conditions can now be rewritten as

$$t_o \le t \le t_f \begin{cases} \dot{x} - \tilde{H}_\lambda^T(x,\lambda,t) = 0 & (37) \\ \dot{\lambda} + \tilde{H}_x^T(x,\lambda,t) = 0 & (38) \end{cases}$$

$$t = t_o \begin{cases} N(x_o,t_o) = 0 & (39) \end{cases}$$

$$t = t_f \begin{cases} \Sigma_f^T = (P_x - \lambda^T) = 0 & (40) \\ M(x_f,t_f) = 0 & (41) \\ \bar{\Omega}_f \overset{\Delta}{=} (P_t + \tilde{H}) = 0 & (42) \end{cases}$$

Equations (37) and (38) are $2n$ equations for the $2n$ unknowns $x(t)$ and $\lambda(t)$. Equations (39) through (42) are $2n+q+1$ conditions for the $2n$ unknowns $x(t)$, $\lambda(t)$, and the $q+1$ unknown parameters ν and t_f. These equations constitute the familiar nonlinear two-point boundary value problem. A first-order perturbation of the nonlinear Euler-Lagrange equations is now considered. This yields the following homogeneous linear system of equations (see Section 2.2).

$$\begin{pmatrix} \delta\dot{x} \\ \delta\dot{\lambda} \end{pmatrix} = \begin{pmatrix} \tilde{H}_{\lambda x} & \tilde{H}_{\lambda\lambda} \\ -\tilde{H}_{xx} & -\tilde{H}_{x\lambda} \end{pmatrix} \begin{pmatrix} \delta x \\ \delta\lambda \end{pmatrix} \tag{43}$$

As was mentioned on page 27, the differential equation for the $n+q+1$ matrix of Riccati variables, $R(t)$, will be symmetric if the terminal boundary values are such that $R(t_f) = R^T(t_f)$. In the next section it is shown that $R(t_f)$ will always be symmetric for the MSM. This reason, along with the fact that an EP scheme is used by the MSM to obtain the perturbation equations given in Equation (18), give the following differential equations for the MSM Riccati variables:

$$\dot{R} = -S^T W S \tag{44}$$

$$\dot{\rho} = -S^T r \tag{45}$$

where

$$S = \begin{pmatrix} I_n & 0 & 0 \\ K & D & \ell \end{pmatrix}$$

$$W \triangleq \begin{pmatrix} \tilde{H}_{xx} & \vdots & \tilde{H}_{x\lambda} \\ \tilde{H}_{\lambda x} & \vdots & \tilde{H}_{\lambda\lambda} \end{pmatrix}$$

$$r \triangleq \begin{pmatrix} \tilde{H}_{x\lambda} \\ \tilde{H}_{\lambda\lambda} \end{pmatrix} \eta$$

3.2 Boundary Conditions

Boundary conditions for these equations are obtained by a first-order perturbation of equations (39) and (40)

$$\delta x(t_o) = \delta \bar{x}_o \tag{46}$$

$$\delta\lambda_f = (P_{xx})_f \delta x_f + M_{x_f}^T d\nu + \alpha_f dt_f - d\Sigma_f \tag{47}$$

Equations (43) represent 2n equations for the 2n unknowns $\delta x(t)$ and $\delta\lambda(t)$. Furthermore, equations (46) and (47) give 2n split-boundary conditions for these variables in terms of the 2n known parameters δx_o and $\delta\lambda_f$ plus the q+1 additional unknown parameters $d\nu$ and dt_f. The required additional q+1 conditions are obtained by also performing a first-order perturbation of equations (41) and (42). It is shown in Appendix B that this procedure yields

$$dM_f = M_{x_f}\delta x + \dot{M}_f dt_f \tag{48}$$

$$d\bar{\Omega}_f = \alpha_f^T \delta x + \dot{M}_f^T d\nu + (\beta + \tau_4)\, dt_f + \tau_5 \tag{49}$$

where

$$\tau_4 = -\Sigma_f^T \frac{Df}{Dt}$$

and $$\tau_5 = -\dot{x}^T d\Sigma_f$$

In matrix form, these boundary conditions can be summarized as

$$\begin{pmatrix} \delta\lambda_f \\ dM_f \\ d\bar{\Omega}_f \end{pmatrix} = \begin{pmatrix} (P_{xx})_f & M_{x_f}^T & \alpha_f \\ M_{x_f} & 0 & \dot{M}_f \\ \alpha_f^T & \dot{M}_f^T & (\beta_f + \tau_4) \end{pmatrix} \begin{pmatrix} \delta x_f \\ d\nu \\ dt_f \end{pmatrix} + \begin{pmatrix} -d\Sigma_f \\ 0 \\ \tau_5 \end{pmatrix} \tag{50}$$

Note that this coefficient matrix is symmetric. Furthermore, if n values of the state are specified at t_f, $\Sigma_f = d\Sigma_f = 0$. For this case, the terminal boundary conditions reduce to the homogeneous form

$$\begin{pmatrix} \delta\lambda_f \\ dM_f \\ d\bar{\Omega}_f \end{pmatrix} = \begin{pmatrix} (P_{xx})_f & M_{x_f}^T & \alpha_f \\ M_{x_f} & 0 & \dot{M}_f \\ \alpha_f^T & \dot{M}_f^T & \beta_f \end{pmatrix} \begin{pmatrix} \delta x_f \\ d\nu \\ dt_f \end{pmatrix} \tag{51}$$

To summarize, Equations (43), (46), and (50) constitute the linear first-order, two-point boundary value problem for the 2n functions $\delta x(t)$, $\delta\lambda(t)$, and the q+1 parameters $d\nu$ and dt_f in terms of the 2n+q+1 specifiable parameters $\delta\bar{x}_o$, $\delta\lambda_f$, dM_f, and $d\bar{\Omega}_f$. The computational procedure followed by the MSM will now be outlined.

3.3 Computational Algorithm

The modified sweep method can be implemented as follows:

Step 1 - Assume n+q+1 values for $\lambda(t_o)$, ν and t_f.

Step 2 - Integrate the nonlinear Euler-Lagrange Equations (37) and (38) forward from t_o to t_f, viz.,

$$\dot{x} = \tilde{H}_{\lambda}^{T}(x,\lambda,t)$$

$$\dot{\lambda} = -\tilde{H}_{x}^{T}(x,\lambda,t)$$

Step 3 - Test the error norm

$$||\text{Error}|| \triangleq \left|\left| \, ||\Sigma_f|| + ||M_f|| + \bar{\Omega}_f^2 \, \right|\right| \leq \varepsilon \, .$$

If this criterion is satisfied, exit; otherwise, continue to the next step.

Step 4 - Set the terminal boundary conditions for the Riccati variables

$$R(t_f) = \begin{bmatrix} K(t_f) & D(t_f) & \ell(t_f) \\ D^T(t_f) & F(t_f) & g(t_f) \\ \ell^T(t_f) & g^T(t_f) & s(t_f) \end{bmatrix}$$

$$= \begin{bmatrix} P_{xx} & M_x^T & \alpha \\ M_x & 0 & \dot{M} \\ \alpha^T & \dot{M}^T & \beta - \Sigma_f^T \frac{Df}{Dt} \end{bmatrix}_{t_f} \tag{52}$$

$$\rho(t_f) = \begin{bmatrix} \eta(t_f) \\ \zeta(t_f) \\ \phi(t_f) \end{bmatrix} = \begin{bmatrix} -d\Sigma_f \\ 0 \\ -\dot{x}^T \; d\Sigma_f \end{bmatrix}_{t_f} \tag{53}$$

Step 5 - Integrate the Riccati variables backward from the final to the initial time using the differential equations

$$\dot{R} = -S^T WS$$

$$\dot{\rho} = -S^T r$$

Step 6 - Compute the n+q+1 corrections $\delta\lambda_o$, $d\nu$ and dt_f using the generalized Riccati transformation (Equation 19), evaluated at the initial time. These corrections are

$$d\nu = (F - gs^{-1}g^{T})^{-1}_{t_o} \Big((dM_f - \zeta) - gs^{-1}(d\Omega_f - \phi)$$

$$- (D^{T} - gs^{-1}\ell^{T})\ \delta x \Big)_{t_o}$$

$$dt_f = s^{-1}_{t_o}\Big((d\Omega_f - \phi) - g^{T}d\nu - \ell^{T}\delta x \Big)_{t_o}$$

and

$$\delta\lambda_o = \Big(K\delta x + Dd\nu + \ell\ dt_f + \eta \Big)_{t_o}$$

where it has been assumed that

$$dM_f = -\overset{*}{\varepsilon}\ M_f$$

$$d\Omega_f = -\overset{*}{\varepsilon}\ \Omega_f$$

$$0 < \overset{*}{\varepsilon} \leq 1$$

Step 7 - Repeat from Step 2 using the new values

$$\lambda^{i+1}(t_o) = \lambda^{i}(t_o) + \delta\lambda_o$$

$$\nu^{i+1} = \nu^{i} + d\nu$$

$$t_f^{i+1} = t_f^{i} + dt_f$$

3.4 Computational Advantages

The computational advantages of the MSM over the SSM are that it has to integrate n+q less variables and requires considerably less storage than the SSM. The exact comparisons are shown in Tables II and III. Table II shows the number of variables which must be integrated by each method in forward and backward directions. Table III shows the storage requirements for each method. Only $3n + 2(q+1)$ quantities need to be stored in the MSM case. For the SSM, however, $M\left(n(q+3)+m+n(n+1)/2\right)$ quantities have to be stored where M is the total number of points in the integration interval. A quick check for a typical reentry problem with $M = 1{,}000$, $n = 6$, $q = 3$, and $m = 1$ shows the MSM requires storage of 26 quantities while the SSM must store 58,000 quantities.

It is speculated that use of the SSM for large complex problems such as the Apollo 3-D reentry will require fixed step-size integration routines with a large enough step-size to remain within the computer storage limitations. Furthermore, the large step size may lead to unsatisfactory numerical accuracy.

TABLE II

MSM AND SSM VARIABLES TO BE NUMERICALLY INTEGRATED

STANDARD SWEEP METHOD		MODIFIED SWEEP METHOD	
Forward:	x – n	Forward:	x – n
Backward:	λ – n		λ – n
	K – $\frac{n(n+1)}{2}$	Backward:	K – $\frac{n(n+1)}{2}$
	D – nxq		D – nxq
	ℓ – n		ℓ – n
	η – n		η – n
	F – $\frac{q(q+1)}{2}$		F – $\frac{q(q+1)}{2}$
	g – q		g – q
	ζ – q		ζ – q
	s – 1		s – 1
	ϕ – 1		ϕ – 1
	y – n		
	z – q		
Totals:	$n(q+5) + (3q+2)$ $+ \frac{1}{2}\left(n(n+1) + q(q+1)\right)$	Totals:	$n(q+4) + (2q+2)$ $+ \frac{1}{2}\left(n(n+1) + q(q+1)\right)$
		Difference:	$(n+q)$ less variables

Note: If all values of the terminal state are constrained, i.e., if $q = n$, then $\eta = 0$, $\zeta = 0$, $\phi = 0$ and the difference increases to $2(n+q)+1$ less variables.

TABLE III
COMPUTER STORAGE REQUIREMENTS

STANDARD SWEEP METHOD	MODIFIED SWEEP METHOD
At every point in the integration interval, the following values must be stored: $x(t)$: n $u(t)$: m $\lambda(t)$: n $K(t)$: $\frac{n(n+1)}{2}$ $D(t)$: $n\text{x}q$ $\ell(t)$: n	At the initial and final points only, the following values must be stored: x_o - n λ_o - n ν - q t_f - 1 Σ_f - n M_f - q $\bar{\Omega}_f$ - 1
Let M = total number of points in the integration, then $M\left[n(q+3) + m + \frac{n(n+1)}{2}\right]$ quantities have to be stored. A typical reentry problem has M = Order(1,000), $n = 6$, $q = 3$, and $m = 1$. Thus, 58,000 quantities must be stored over the integration interval.	Only $3n+2(q+1)$ quantities need to be stored from iteration to iteration. Compare 26 quantities with 58,000 for the Apollo reentry problem.

CHAPTER 4

THE MODIFIED SWEEP METHOD GUIDANCE SCHEME

Initial conditions for dynamical processes are difficult to control in actual problems. Errors often occur which may be due to internal mechanical causes such as premature cutoff by a thrusting rocket motor. Regardless of where these errors occur, they have the cumulative effect of causing a deviation from the intended optimal path. It is then desirable to use the known information about the path to recompute a new control program to accomplish the mission objectives. This is done by determining the control function corrections δu as a function of the state perturbation; i.e., $\delta u = \delta u(\delta x, t)$. This is a guidance problem in optimization theory. The guidance relations are now derived using the MSM.

The MSM assumes that from the local optimality condition $H_u^T = 0$ and the strengthened Legendre-Clebsch condition $\delta u^T H_{uu} \delta u > 0$, the minimizing control $u(t)$ can be expressed as an explicit function of the state and Lagrange multiplier variables; i.e.,

$$u = U(x, \lambda, t) \tag{54}$$

Perturbing this expression to first-order gives

$$\delta u = U_x \delta x + U_\lambda \delta\lambda \tag{55}$$

The generalized Riccati transformation is

$$\delta\lambda = K\delta x + D\,d\nu + \ell\,dt_f + \eta \tag{56}$$

$$dM_f = D^T\delta x + F\,d\nu + g\,dt_f + \zeta \tag{57}$$

$$d\Omega_f = \ell^T\delta x + g^T\,d\nu + s\,dt_f + \phi \tag{58}$$

Solving the last two equations simultaneously yields

$$d\nu = \pi_{11}\delta x + \pi_{12}y + \pi_{13}z \tag{59}$$

$$dt_f = \pi_{21}\delta x + \pi_{22}y + \pi_{23}z \tag{60}$$

where

$$y \triangleq (dM_f - \zeta)$$

$$z \triangleq (d\Omega_f - \phi)$$

$$\Lambda \triangleq (F - gs^{-1}g^T)^{-1}$$

and

$$\pi_{11} \triangleq \Lambda(gs^{-1}\ell^T - D^T)$$

$$\pi_{12} \triangleq \Lambda$$

$$\pi_{13} \triangleq -\Lambda gs^{-1}$$

$$\pi_{21} \triangleq -s^{-1}(g^T\pi_{11} + \ell^T)$$

$$\pi_{22} \triangleq -s^{-1}g^T\pi_{12}$$

$$\pi_{23} \triangleq -s^{-1}(g^T\pi_{13}-1)$$

Using Equations (59) and (60) to eliminate $d\nu$ and dt_f from Equation (56) gives

$$\delta\lambda = (K + D\pi_{11} + \ell\pi_{21})\delta x + (D\pi_{12} + \ell\pi_{22})y + (D\pi_{13} + \ell\pi_{23})z \tag{61}$$

Substituting this expression into Equation (55) then gives

$$\delta u = \left(U_x + U_\lambda(K + D\pi_{11} + \ell\pi_{21})\right)\delta x$$

$$+ U_\lambda\left((D\pi_{12} + \ell\pi_{22})y + (D\pi_{13} + \ell\pi_{23})z\right) \tag{62}$$

This equation represents the linear-feedback control law for continuously correcting the optimal control program for a given perturbation $\delta x(t)$ in the vehicle state. The new control program is then obtained by adding these corrections to the converged control program. Note that for a given perturbation $\delta x(t)$, the coefficients are obtained by integrating the Riccati variables forward using the initial values corresponding to the converged solution.

In guidance work it is desirable to know the control corrections in terms of a known time-dependent matrix and the initial vehicle state perturbations, i.e.,

$$\delta u(t) = L(t)\ \delta x(t_o) \tag{63}$$

where $L(t)$ is an explicit relationship between the Riccati variables from the converged optimal trajectory. Attempts to yield a relation such as Equation (63) were unsuccessful. Numerical studies using the MSM guidance scheme therefore were conducted with the assumption of a continuously correcting procedure.

An immediate disadvantage is obvious in the guidance scheme represented by Equation (62). Since values for the converged Riccati variables are not stored by the MSM except at the initial point,these variables must again be integrated forward from this initial time regardless of when the perturbation $\delta x(t)$ occurs. A more detailed discussion of this problem is contained in the next chapter.

CHAPTER 5

DISCUSSION OF NUMERICAL RESULTS

The modified sweep method algorithm was programmed for the UNIVAC 1108 digital computer at the Manned Spacecraft Center in Houston, Texas. The integration schemes used follow.

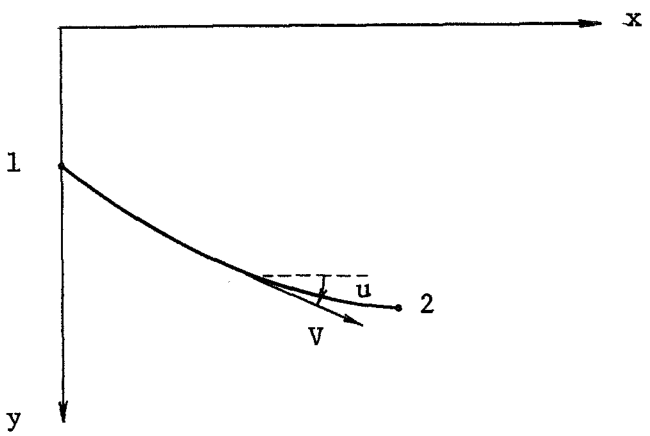
5.1 Numerical Integration Routines

Fixed Step-Size Integration. Fixed step-size integrations were carried out using an Adams-Bashforth predictor-corrector procedure with a Runge-Kutta starter (Lastman and Fowler[15]). The Adams predictor had a discretization error of $o(h^5)$, and the Bashforth corrector discretization error was of $o(h^6)$; h is the step-size. The Runge-Kutta starter had a discretization error of $o(h^5)$. Partial double-precision arithmetic was used as follows: the values of the dependent variables were carried in full double precision, but the derivatives were evaluated and stored as single-precision numbers. This technique minimized the effect of round-off error.

Variable Step-Size Integration. Variable step-size integrations were carried out using a predictor-corrector-starter procedure as mentioned above. However, the discretization error in all cases was of $o(h^5)$. These integrations were carried out in full double precision (Schwausch[33]).

5.2 A Brachistochrone Problem

To compare Modified Sweep Method converged results with known analytical solutions for a problem of sufficient complexity, a class of free-final-time Brachistochrone problems was posed as follows.



Minimize the value of the final time t_f for a particle to fall along a frictionless path in a constant gravitational field from point 1 to point 2 subject to the constraints

$$\dot{x} = V \cos u$$

$$\dot{y} = V \sin u$$

$$x(t_o) = 0.0$$

$$y(t_o) = 1.0$$

where

$$V = \sqrt{2g(y - a)}$$

and $$a = y_o - \frac{V_o^2}{2g}$$

The variational Hamiltonian is $H = V(\lambda_x \cos u + \lambda_y \sin u)$. Two different cases were solved for the terminal constraint vector M_f :

$$M_f^1 = x(t_f) - 5.0 = 0$$

and
$$M_f^2 = \begin{pmatrix} x(t_f) - 5.0 \\ y(t_f) - 8.0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

The Modified Sweep Hamiltonian and its Partial Derivatives

Using the optimality condition $H_u^T = 0$ and the strengthened Legendre-Clebsch condition that $\delta u^T H_{uu} \delta u > 0$, the control vector can be eliminated to yield the following Euler-Lagrange equations

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = -\frac{V}{\Lambda} \begin{pmatrix} \lambda_x \\ \lambda_y \end{pmatrix} = \tilde{H}_{\bar{\lambda}}^T$$

$$\begin{pmatrix} \dot{\lambda}_x \\ \dot{\lambda}_y \end{pmatrix} = \frac{\Lambda}{V} \begin{pmatrix} 0 \\ g \end{pmatrix} = -\tilde{H}_{\bar{x}}^T$$

where

$$\Lambda \triangleq \sqrt{\lambda_x^2 + \lambda_y^2}$$

Furthermore, the second partial derivatives required by the perturbation equations are

$$\tilde{H}_{\bar{x}\bar{x}} = \begin{pmatrix} 0 & 0 \\ 0 & \frac{\Lambda g^2}{V^3} \end{pmatrix}$$

$$\tilde{H}_{\underline{\lambda x}} = \begin{bmatrix} 0 & -\frac{g}{V}\left(\frac{\lambda_x}{\Lambda}\right) \\ 0 & -\frac{g}{V}\left(\frac{\lambda_y}{\Lambda}\right) \end{bmatrix}$$

$$\tilde{H}_{\underline{\lambda\lambda}} = \left(\frac{V}{\Lambda^3}\right) \begin{bmatrix} -\lambda_y^2 & \lambda_x\lambda_y \\ \lambda_y\lambda_x & -\lambda_x^2 \end{bmatrix}$$

Results

The M_f^1 and M_f^2 cases were computed using the fixed step-size integrator mentioned on page 44. For comparison purposes, these two cases were solved using the Method of Perturbation Functions program, MPF (see Lewallen[16]). A step-size $h = 0.01$ second was used in all cases, with the initially assumed values of the unknown Lagrange multipliers and final time as follows:

$$\lambda_x^o = -0.2368 \text{ sec/ft}$$

$$\lambda_y^o = -0.6095 \text{ sec/ft}$$

$$t_f = 0.5410 \text{ sec}$$

The convergence criterion ε was specified as 0.1×10^{-8}. The correction procedure used in all cases was 25%, 50%, 75%, and 100% from the fourth iteration onward.

Rate of Convergence. Figures 1 and 2 show plots of the terminal constraint norms versus time for the Brachistochrone problem. Both the MPF and MSM results are plotted. Figure 1 shows the error

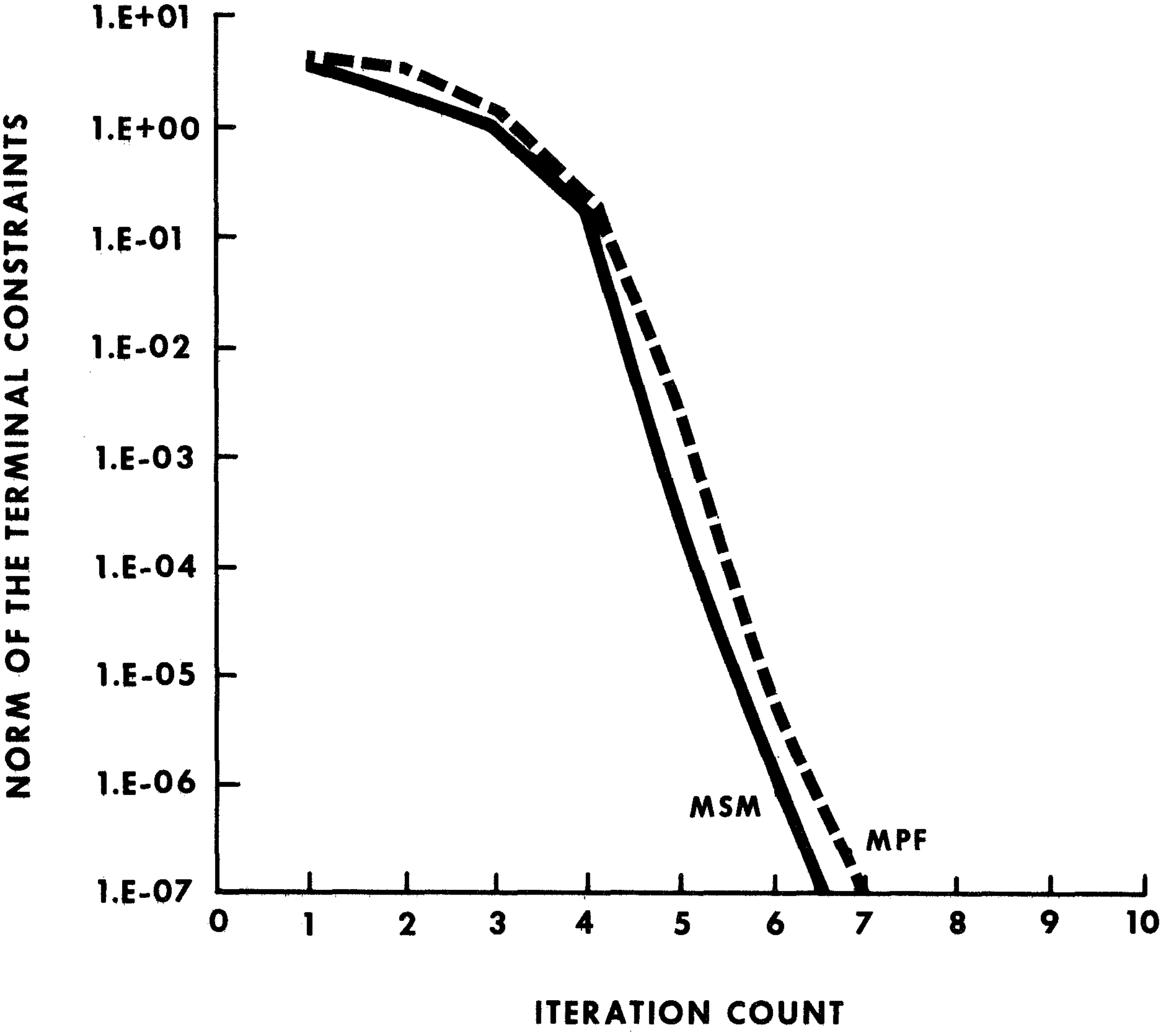


FIGURE 1

M_f^1 CASE FOR BRACHISTOCHRONE PROBLEM

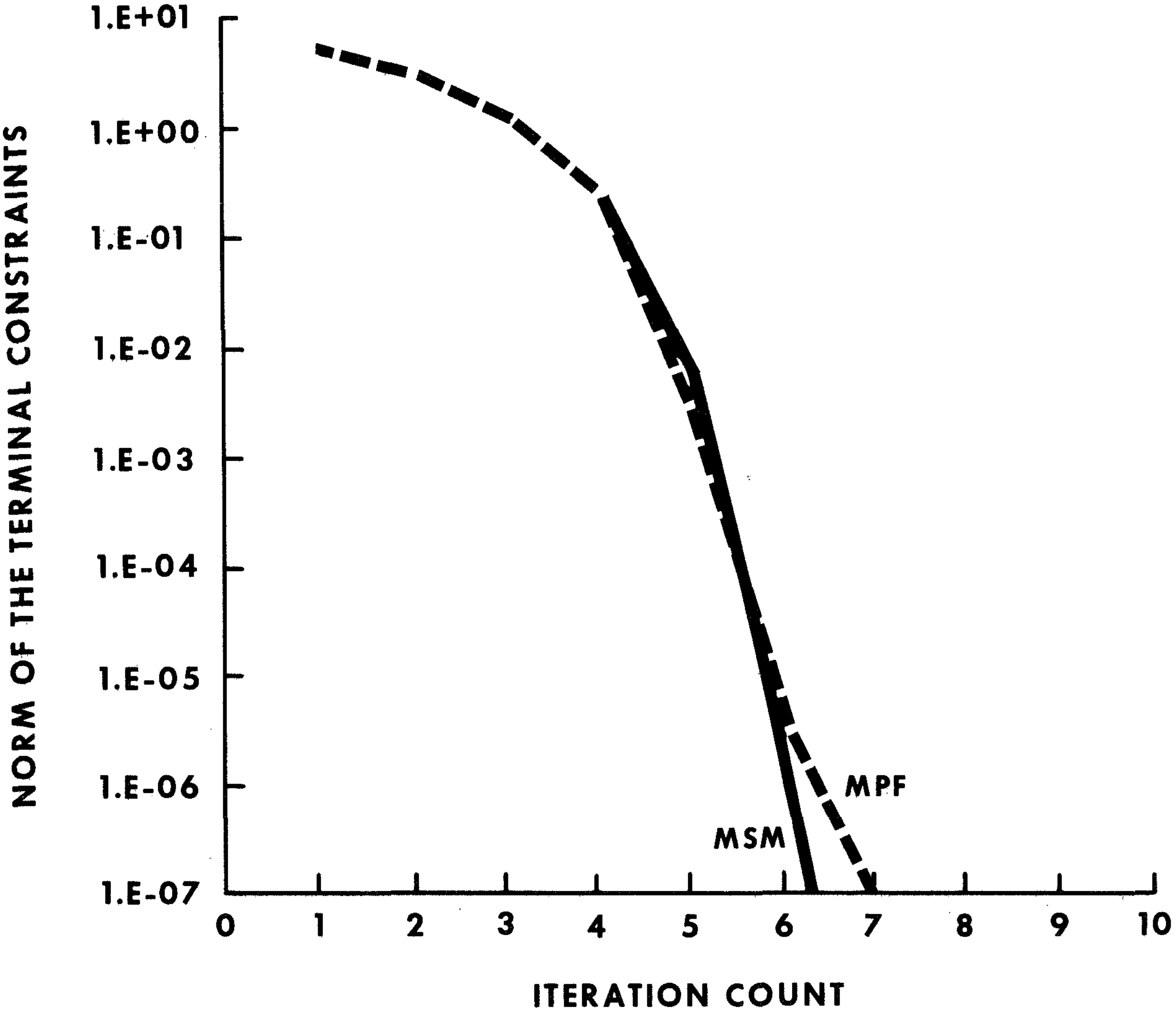


FIGURE 2

M_f^2 CASE FOR BRACHISTOCHRONE PROBLEM norm for the case $M_f^1 = x(t_f) - 5.0$. The two methods converge in seven iterations; however, the MSM consistently shows a smaller error than the MPF for each iteration. Note that the decrease in the error norm for the MSM is significantly less than the MPF for the last iteration.

Figure 2 shows the error norms for the case $M_f^2 = \left(x(t_f) - 5.0 \;\vdots\; y(t_f) - 8.0\right)^T$; the MSM error norm is not always less than the MPF error norm. However, the error difference is never large. The terminal stages of iteration reveal the same high rate of convergence as in the M_f^1 case.

The MSM for this problem at its worst took 20% less computational time. However, this figure is not considered significant because two unrelated computer programs were used.

Accuracy of Converged Results. The MSM converged solutions gave six decimal place agreement for both the M_f^1 case and the M_f^2 case when they were compared to the known analytical solutions. For the M_f^1 case, the initially guessed multipliers λ_x^o and λ_y^o were in error with the converged values 224% and 282%, respectively, the initial guess at the final time was 2% in error. For the M_f^2 case, the initial guesses on the same λ_x^o and λ_y^o multipliers were 500% and 260%, respectively. In this case, the initial-guess error for the final time was 14%. These results are tabulated in Table IV.

To summarize, the MSM has exhibited rapid terminal convergence and reasonable convergence envelopes for the free-final-time Brachistochrone problem.

TABLE IV

MSM INITIAL-GUESS ERROR RESULTS

	M_f^1			M_f^2	
	Converged Values	% Initial Error		Converged Values	% Initial Error
λ_x^o	-0.0689	244%		-0.0357	500%
λ_y^o	-0.1623	282%		-0.1726	260%
t_f	0.5271	2%		0.6277	14%

The excellent results warranted further applications of the MSM to more complex problems whose analytical solutions were not known and which were of current interest to the space program. For these reasons, an Apollo three-dimensional reentry problem was chosen.

5.3 Apollo Three-Dimensional Reentry Problem

In the time interval $t_o \leq t \leq t_f$, find the roll angle program $\beta(t)$ which can be used to control an Apollo spacecraft so as to minimize the weighted sum of heating and acceleration effects

$$I = \int_{t_o}^{t_f} \left(\frac{\sqrt{L^2 + D^2}}{m} + \tilde{\lambda}_o \rho^{\frac{1}{2}} V^3 \right) dt$$

where $\tilde{\lambda}_o$ is a constant weighting parameter. Here, the first-term in the integrand serves to measure the acceleration effects due to aerodynamic forces while the second term measures the convective heating experienced by the spacecraft. This minimization is to be accomplished subject to the differential equations of motion given as follows

$$\begin{pmatrix} \dot{h} \\ \dot{\theta} \\ \dot{\Delta} \\ \dot{V} \\ \dot{\gamma} \\ \dot{A} \end{pmatrix} = \begin{pmatrix} V \sin \gamma \\ V \cos \gamma \cos A/(R + h) \cos \Delta \\ V \cos \gamma \sin A/(R + h) \\ G \sin \gamma - \bar{D} \\ (G \cos \gamma/V) + \{V \cos \gamma/(R + h)\} + (\bar{L} \cos \beta/V) \\ \{-V \cos \gamma \cos A \tan \Delta/(R + h)\} - \{(\bar{L} \sin \beta/(V \cos \gamma))\} \end{pmatrix}$$

The following initial conditions represent the reentry conditions for a space vehicle on an Apollo-type lunar return mission.

$$\begin{pmatrix} \bar{h}(t_o) \\ \bar{\theta}(t_o) \\ \bar{\Delta}(t_o) \\ \bar{V}(t_o) \\ \bar{\gamma}(t_o) \\ \bar{A}(t_o) \end{pmatrix} = \begin{pmatrix} 400{,}000 \text{ ft} \\ 0.0^\circ \\ 0.0^\circ \\ 35{,}000 \text{ fps} \\ -6.5^\circ \\ 0.0^\circ \end{pmatrix} = \begin{pmatrix} 75.757576 \text{ mi} \\ 0.0 \text{ rad} \\ 0.0 \text{ rad} \\ 6.8181818 \text{ mi/sec} \\ -0.11344640 \text{ rad} \\ 0.0 \text{ rad} \end{pmatrix}$$

where

$$G = -\mu/(R + h)^2$$

$$\bar{D} = \rho S V^2 C_D/2m$$

$$\bar{L} = \rho S V^2 C_L / 2m$$

$$\rho = \rho_o e^{-\overset{*}{\beta} h}$$

and m = spacecraft mass (assumed constant)

Optimal reentry trajectories were determined for two cases of terminal conditions

$$M_f^1 = \begin{bmatrix} h(t_f) - \bar{h}_f \\ \theta(t_f) - \bar{\theta}_f \\ \Delta(t_f) - \bar{\Delta}_f \\ V(t_f) - \bar{V}_f \\ \gamma(t_f) - \bar{\gamma}_f \\ A(t_f) - \bar{A}_f \end{bmatrix} \quad \text{and} \quad M_f^2 = \begin{bmatrix} \theta(t_f) - \bar{\theta}_f \\ \Delta(t_f) - \bar{\Delta}_f \\ V(t_f) - \bar{V}_f \\ A(t_f) - \bar{A}_f \end{bmatrix}$$

Values for the terminal state represent a typical set of conditions at drogue parachute deployment for the Apollo space vehicle

$$\begin{bmatrix} \bar{h}_f \\ \bar{\theta}_f \\ \bar{\Delta}_f \\ \bar{V}_f \\ \bar{\gamma}_f \\ \bar{A}_f \end{bmatrix} = \begin{bmatrix} 75{,}504 \text{ ft} \\ 24.1^{\circ} \\ -0.6^{\circ} \\ 856 \text{ fps} \\ -44.3^{\circ} \\ -29.4^{\circ} \end{bmatrix}$$

Numerical values for the Apollo parameters were assumed as

$$S = 129.3 \text{ ft}^2 = 0.46379993\text{E-05 mi}^2$$

$$m = 204.0 \text{ slugs}$$

$$\rho_o = 0.27\text{E-02 slug/ft}^3 = 0.39743447\text{E} + 09 \text{ slug/mi}^3$$

$$\overset{*}{\beta} = 0.42\text{E-04 ft}^{-1} = 0.2217600\text{E} + 00 \text{ mi}^{-1}$$

$$\tilde{\lambda}_o = 0.24509804\text{E-07 sec/(slug-ft)}^{1/2}$$

$$= 0.17809708\text{E-05 sec/(slug-mi)}^{1/2}$$

$$\mu = 0.14076519\text{E} + 17 \text{ ft}^3/\text{sec}^2 = 0.95629856\text{E} + 05 \text{ mi}^3/\text{sec}^2$$

$$R = 0.20908800\text{E} + 08 \text{ ft} = 0.39600000\text{E} + 04 \text{ mi}$$

Figure 3 shows the essential geometrical relationships between the state variables for the three-dimensional Apollo reentry problem. The variables chosen to specify the state of the point mass spacecraft were h = altitude, θ = longitude, Δ = latitude, V = speed, γ = angle of attack, and A = heading angle. The following assumptions have been made: the earth is a nonrotating homogeneous sphere with its center fixed in interial space. Furthermore, its gravitational potential is characterized by an inverse-square law and it possesses an exponential atmosphere.

The Modified Sweep Method Reentry Hamiltonian. In Appendix C, the Apollo reentry optimization problem is restated. The mechanics of restructuring the problem Hamiltonian by use of local control optimality and the strengthened Legendre-Clebsch condition are shown. The Hamiltonian which is optimal with respect to the choice of roll angle β is given as follows:

$$\tilde{H} = (\rho S V^2 \sqrt{C_L^2 + C_D^2} /2m) + \tilde{\lambda}_0 \rho^{1/2} V^3 + \lambda_1 V \sin\gamma + \frac{V \cos\gamma}{(R + h)}$$

$$\cdot \left\{\frac{\cos A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) + (\lambda_3 \sin A + \lambda_5)\right\} - \frac{\mu}{(R + h)^2}$$

$$\cdot \left\{ \lambda_4 \sin\gamma + \frac{\lambda_5 \cos\gamma}{V} \right\}$$

$$- \frac{1}{2m} \rho S V \left[\lambda_4 C_D V + \frac{C_L}{\cos\gamma} \sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma}\right]$$

The Euler-Lagrange equations for this problem are then generated by taking first partial derivatives of $\tilde{H}$ as follows:

$$\dot{x}_i = \frac{\partial \tilde{H}}{\partial \lambda_i} \quad \text{and} \quad \dot{\lambda}_j = - \frac{\partial \tilde{H}}{\partial x_j} \quad \text{where} \quad i,j = 1, \ldots, 6.$$

These results, along with the second partial derivatives $\tilde{H}_{xx}$, $\tilde{H}_{x\lambda}$ and $\tilde{H}_{\lambda\lambda}$ which serve as coefficients for the matrix Riccati equation are also presented in Appendix C.

<u>Results</u>. Initial attempts to solve the Apollo three-dimensional reentry problem encountered some difficulties when the modified sweep method algorithm was used. Using the system of units ft/lb/sec, certain elements of the Riccati matrix K grew very large at the initial time. Because of these large values, the matrix F also became very large at the initial time. Consequently, when its inverse was used to compute initial-time corrections, they were so small that the initial values altered only in the seventh decimal place. As a result, the initial trajectory was essentially duplicated by subsequent sweeps.

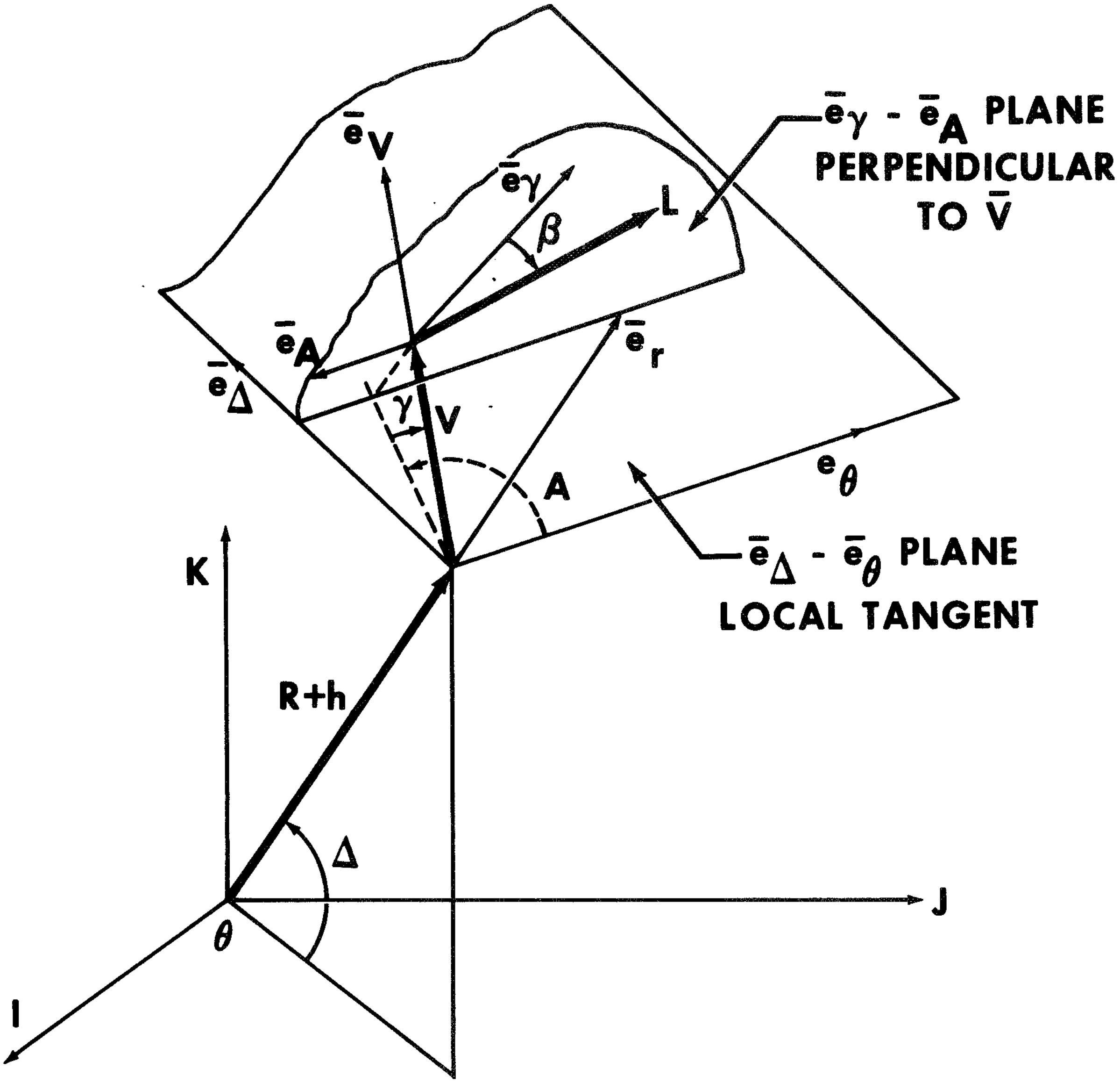


FIGURE 3

THREE-DIMENSIONAL APOLLO REENTRY GEOMETRY

An attempt to impose arbitrary bounds on these variables was tried for several bounding orders of magnitude ranging from 0.5 to 1×10^6. In all cases, every element in the Riccati matrix achieved the bounding value by the third sweep. An attempt was then made to generate more accuracy by cycling through the evaluation-correction procedure $(EC)^N$ of the fixed-step-size integrator several times. Values were tried for N ranging from 2 through 9. This effort to prevent the Riccati matrix from going onto the limiting boundary was unsuccessful.

A scheme which used a scaled fractional part of the corrections $\delta\lambda_o$ was then attempted, and this did not eliminate the numerical difficulties with the Riccati matrix. The vector M_f was then altered with respect to size and to choice of terminal state variables, neither of which was successful. The system of units then was altered to slug/mi/sec, for which the range of Lagrange multiplier magnitudes became smaller. Altering the unit system was tried after discussion with Williamson[39], whose studies on the same problem with the MPF revealed a correlation between the numerical sensitivities of the Lagrange multiplier equations and the unit system chosen. The choice of slug/mi/sec achieved a more suitable scaling for the magnitudes of the multipliers; however, this did not succeed in eliminating the difficulties with the Riccati matrix.

A variable step-size integrator routine was then introduced which revealed the numerical sensitivity of the Apollo three-dimensional reentry problem to the single-step error on the UNIVAC 1108. This sensitivity was measured by fixing the final time at $t_f = 437.263$ seconds; the initial values for the state and Lagrange multipliers were defined as

X_o and Λ_o as shown in Table V. The state and Lagrange multiplier differential equations were then integrated forward from $t_o = 0$ seconds to t_f. Using the terminal values for the state and Lagrange multipliers, the process was reinitialized at t_f and a backward integration carried out. The values obtained at t_o using this procedure were then compared for agreement with the defined values of X_o and Λ_o. Four cases were tested in which the single step error ε was bounded:

Case I $\quad 1.0 \times 10^{-10} \leq \varepsilon \leq 1.0 \times 10^{-07}$

Case II $\quad 1.0 \times 10^{-12} \leq \varepsilon \leq 1.0 \times 10^{-09}$

Case III $\quad 1.0 \times 10^{-13} \leq \varepsilon \leq 1.0 \times 10^{-10}$

Case IV $\quad 1.0 \times 10^{-14} \leq \varepsilon \leq 1.0 \times 10^{-11}$

To numerically integrate forward from the initial to the final time and reinitialize and integrate backward to reproduce the initial values to eight decimal places, the single-step error ε had to be bounded as

$$1 \times 10^{-14} \leq \varepsilon \leq 1 \times 10^{-11}$$

When the error became less than 1×10^{-14}, the integration step size was doubled for the next step. If the error exceeded 1×10^{-11}, the step size was halved; otherwise, the step size remained unchanged. These results are summarized in Table V where the bar under the digits denotes deviations from agreement with initially assumed values.

Figures 4 through 9 give a particular set of numerical results for this problem. This set of results was essentially identical for both set of terminal conditions M_f^1 and M_f^2 .

TABLE V

SINGLE-STEP ERROR TOLERANCE

INITIAL VECTOR

$$\Lambda_o = \begin{vmatrix} 0.71895168 - 02 \\ -0.72100154 + 01 \\ 0.81929784 + 01 \\ 0.24206058 + 01 \\ 0.16453211 + 02 \\ 0.35760665 + 01 \end{vmatrix} \qquad X_o = \begin{vmatrix} 0.75757576 + 02 \\ 0 \\ 0 \\ 0.68181818 + 01 \\ -0.11344640 + 00 \\ 0 \end{vmatrix}$$

NUMERICAL RESULTS

	Case I	Case II	Case III	Case IV
ERRMIN	1.OD-10	1.OD-12	1.OD-13	1.OD-14
ERRMAX	1.OD-07	1.OD-09	1.OD-10	1.OD-11
λ_h	$0.71\underline{9}34619-02$	$0.71895\underline{7}86-02$	$0.71895\underline{2}65-02$	$0.718951\underline{7}8-02$
λ_θ	$-0.72100154+01$	$-0.72100154+01$	$-0.72100154+01$	$-0.72100154+01$
λ_Δ	$0.819297\underline{6}9+01$	$0.8192978\underline{3}+01$	$0.81929784+01$	$0.81929784+01$
λ_V	$0.2420\underline{7}833+01$	$0.242060\underline{7}9+01$	$0.242060\underline{6}3+01$	$0.2420605\underline{9}+01$
λ_γ	$0.1645\underline{8}225+02$	$0.164532\underline{7}7+02$	$0.164532\underline{2}5+02$	$0.1645321\underline{2}+02$
λ_A	$0.357\underline{6}0746+01$	$0.3576066\underline{6}+01$	$0.35760665+01$	$0.35760665+01$
h_o	$0.757\underline{6}6004+02$	$0.75757\underline{6}26+02$	$0.757575\underline{9}6+02$	$0.7575757\underline{7}+02$
θ_o	$-0.40744522-\underline{06}$	$-0.13346897-\underline{07}$	$-0.15685918-\underline{08}$	$-0.14599980-\underline{09}$
Δ_o	$-0.65481932-\underline{06}$	$-0.13262769-\underline{08}$	$-0.18274255-\underline{08}$	$-0.10809805-\underline{09}$
V_o	$0.6818\underline{5}440+01$	$0.681818\underline{8}6+01$	$0.681818\underline{2}7+01$	$0.6818181\underline{9}+01$
γ_o	$-0.1134\underline{6}347+00$	$-0.1134464\underline{7}+00$	$-0.1134464\underline{4}+00$	$-0.11344640+00$
A_o	$0.41275475-\underline{05}$	$0.20899670-\underline{07}$	$0.10682438-\underline{07}$	$0.73654477-\underline{09}$

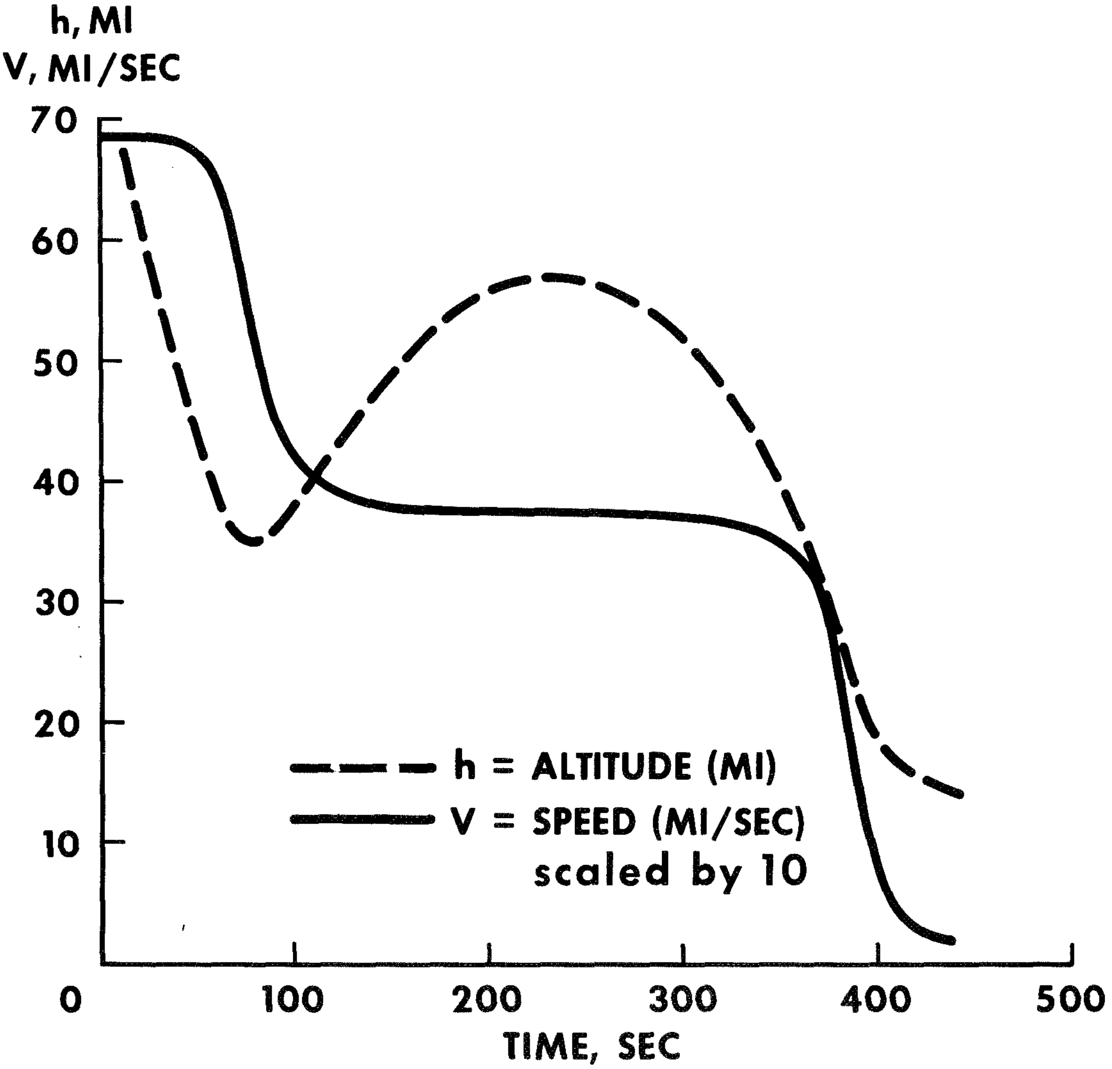


FIGURE 4

REENTRY ALTITUDE AND SPEED VERSUS TIME

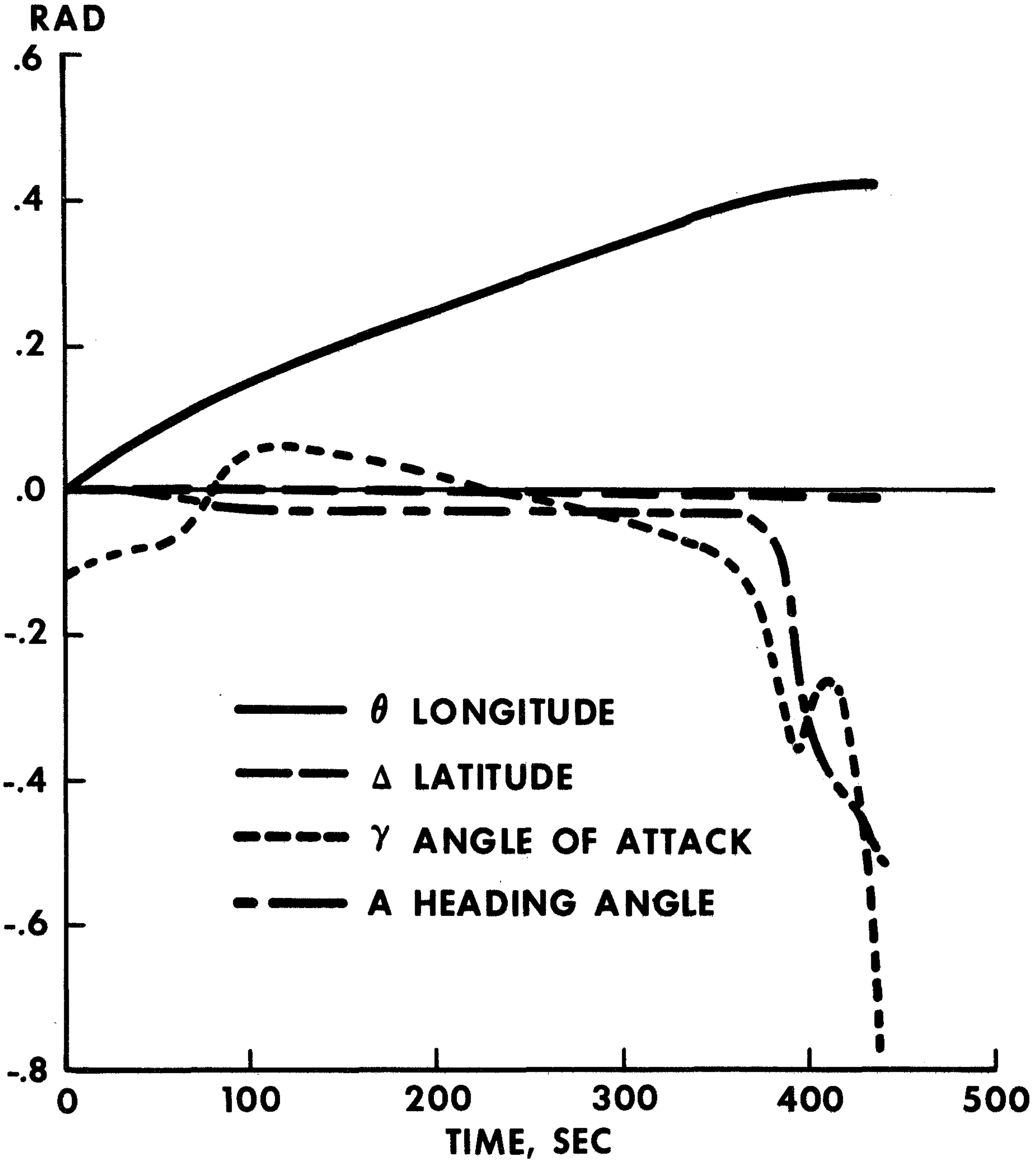


FIGURE 5

REENTRY LONGITUDE, LATITUDE, ANGLE OF ATTACK, AND HEADING ANGLE VERSUS TIME

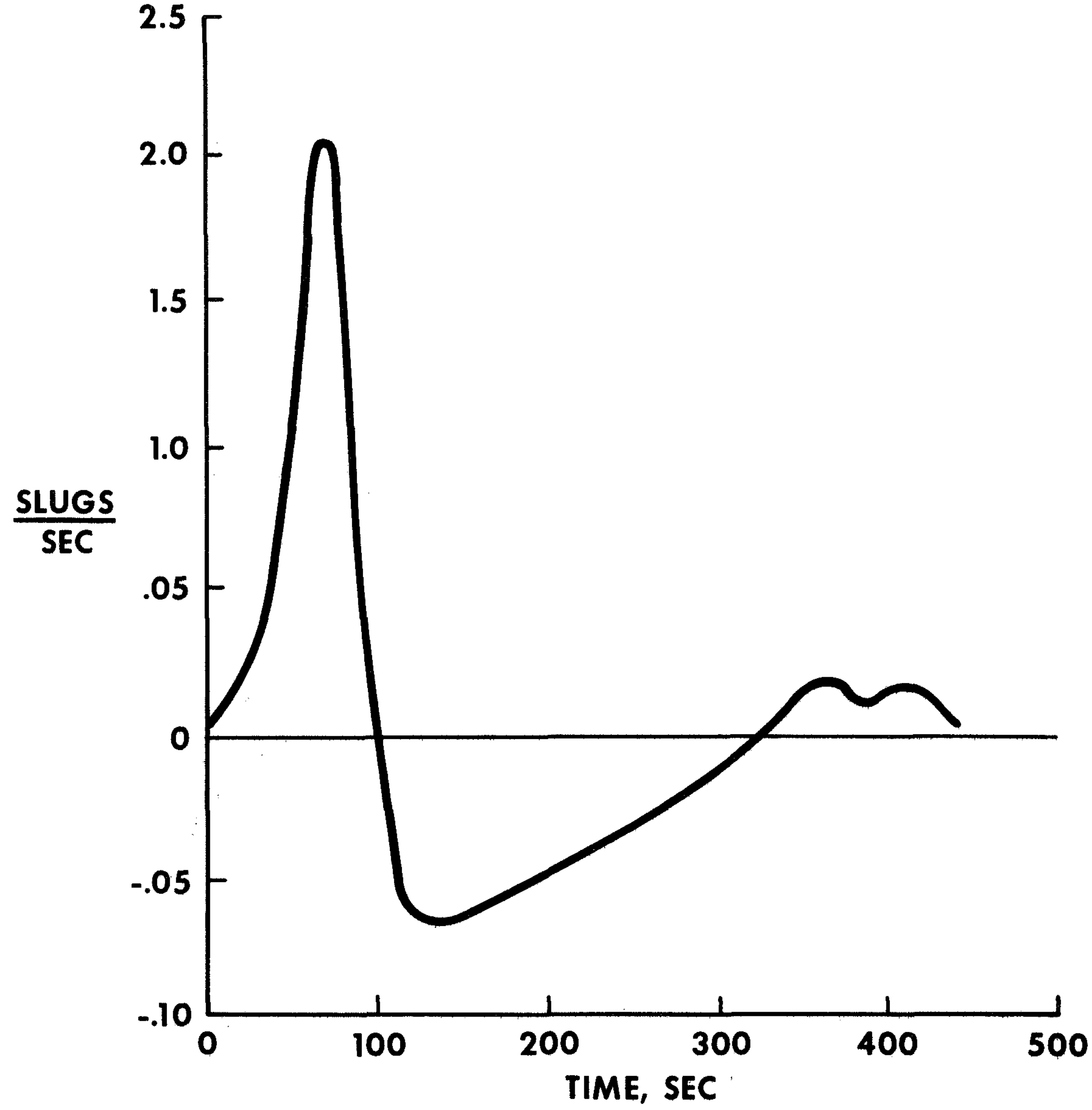


FIGURE 6

λ_h VERSUS TIME

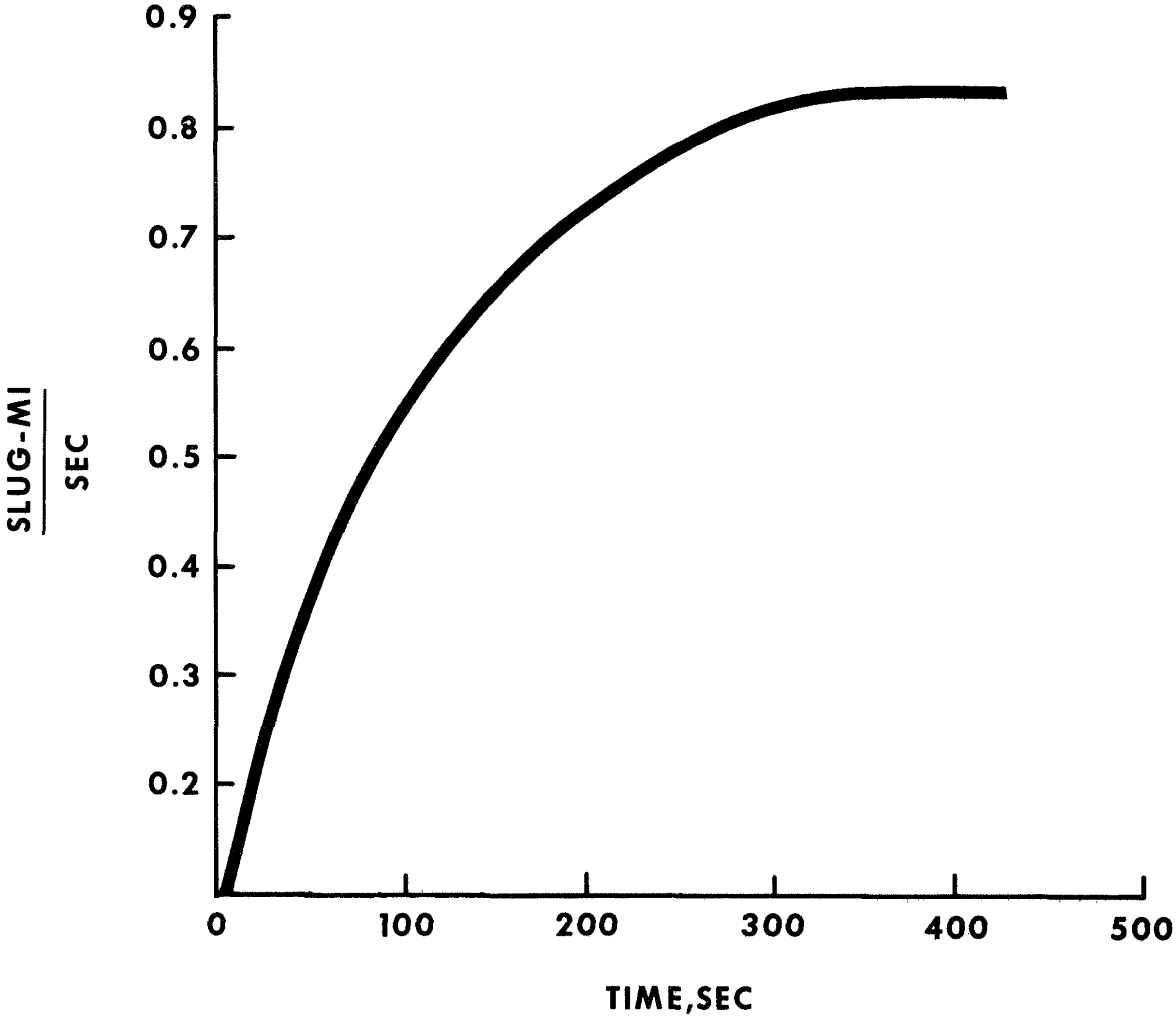


FIGURE 7

λ_{Δ} VERSUS TIME

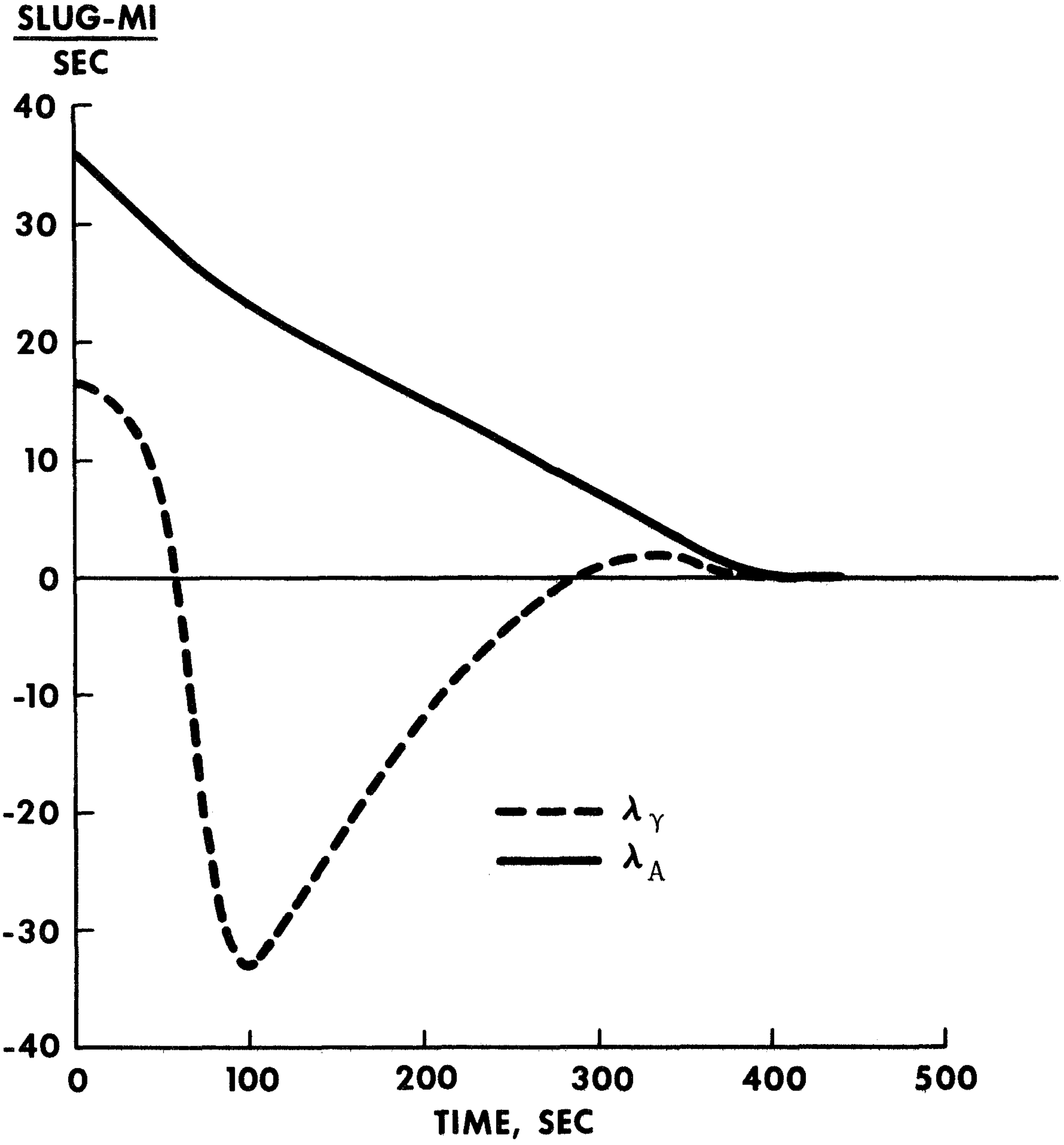


FIGURE 8

λ_γ AND $10\lambda_A$ VERSUS TIME

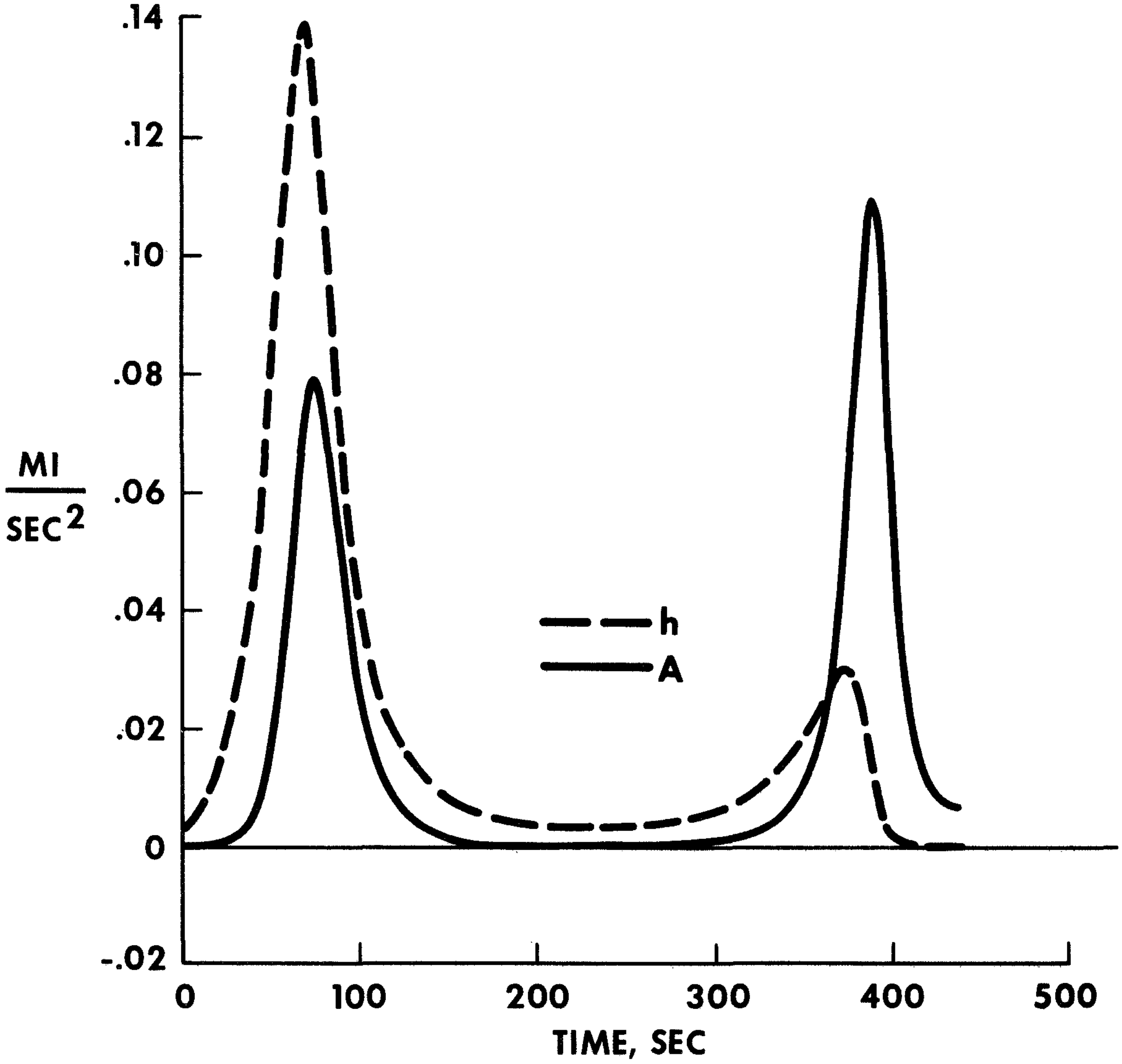


FIGURE 9

ACCELERATION (A) AND WEIGHTED HEATING TERM (H) VERSUS TIME

Figure 4 shows the altitude and scaled reentry speed histories that the Apollo spacecraft would follow during optimal reentry for both the M_f^1 and M_f^2 cases. Figure 5 shows the histories for the longitude, latitude, angle of attack and heading angle state variables during these optimal reentries. These two figures are of interest because they define the optimal state histories which the Apollo spacecraft should fly to achieve the specified terminal conditions while minimizing aerodynamic acceleration and convective heating. Figures 6 and 7 show the λ_h and λ_Δ multiplier histories, respectively. These are the Lagrange multipliers associated with the rates of change in altitude and latitude. They are included here to define the trends to be anticipated for the specified set of initial reentry conditions.

The two Lagrange multipliers which are required to define the optimal reentry roll profile are shown in Figure 8. This particular figure shows the λ_γ and λ_A histories where λ_γ and λ_A are associated with the rates of change in the reentry angle and heading angle, respectively. Figure 9 shows the reentry history for the payoff function. The aerodynamic acceleration and weighted convective heating experienced by the spacecraft have been plotted to reveal their individual characteristics. A study of this figure shows that two peaks occur in both spacecraft acceleration and heating during the optimal reentries. The optimal reentry roll procedures seem to call for a trade-off philosophy between acceleration and heating experienced by the Apollo spacecraft. The high peak in convective heating is initially balanced with a smaller acceleration peak in the vicinity of 100 seconds. This situation is subsequently reversed in the vicinity of 400 seconds where the high acceleration peak is balanced with the smaller heating peak.

Figure 10 and 11 show the λ_V histories for the M_f^1 and M_f^2 cases, respectively. This is the Lagrange multiplier which is associated with the rate of change of speed for the Apollo spacecraft. These figures were included to reveal, for the specified initial reentry conditions, the sensitivities of this variable to a change in terminal conditions. It is interesting to note that both histories are similar with the major difference arising beyond 400 seconds.

Figures 12 and 13 show the optimal reentry roll programs for the M_f^1 and $M_f{}^2$ cases, respectively. These are the roll profiles that an astronaut would have to use during reentry from a lunar mission to minimize aerodynamic acceleration and convective heating while satisfying the desired terminal constraints on the vehicle state.

In each case, optimal reentry calls for the spacecraft to commence the reentry maneuver with the lift vector pointed almost straight downward. The spacecraft is then quickly rolled such that the lift vector is pointed almost straight up after 90 seconds. A slower downward roll of the lift vector is then initiated so that this vector is at a value of 169 degrees by 350 seconds. The lift vector is subsequently rolled upward to approximately 15 degrees by 410 seconds at which time terminal downward roll procedures differ depending on the specified values for the final vehicle state. Specifying two less conditions on the final state of the Apollo spacecraft calls for less terminal roll of the lift vector.

An attempt was made then to obtain a precise evaluation of the MSM computational characteristics. For comparison purposes, the three-dimensional Apollo reentry M_f^2 case was chosen. The MSM program was then altered to assume computational characteristics similar to

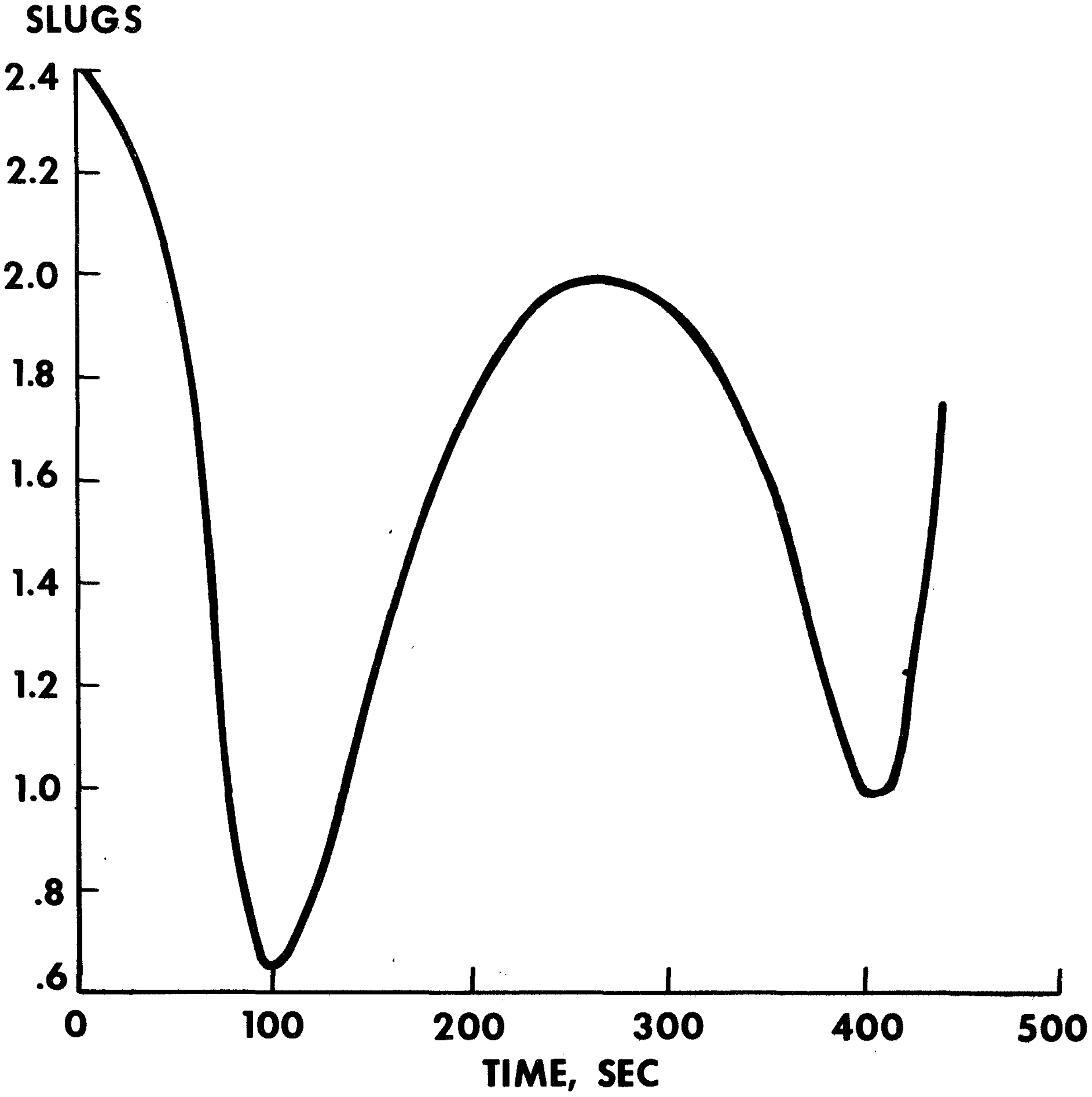


FIGURE 10

λ_V VERSUS TIME FOR M_f^1 CASE

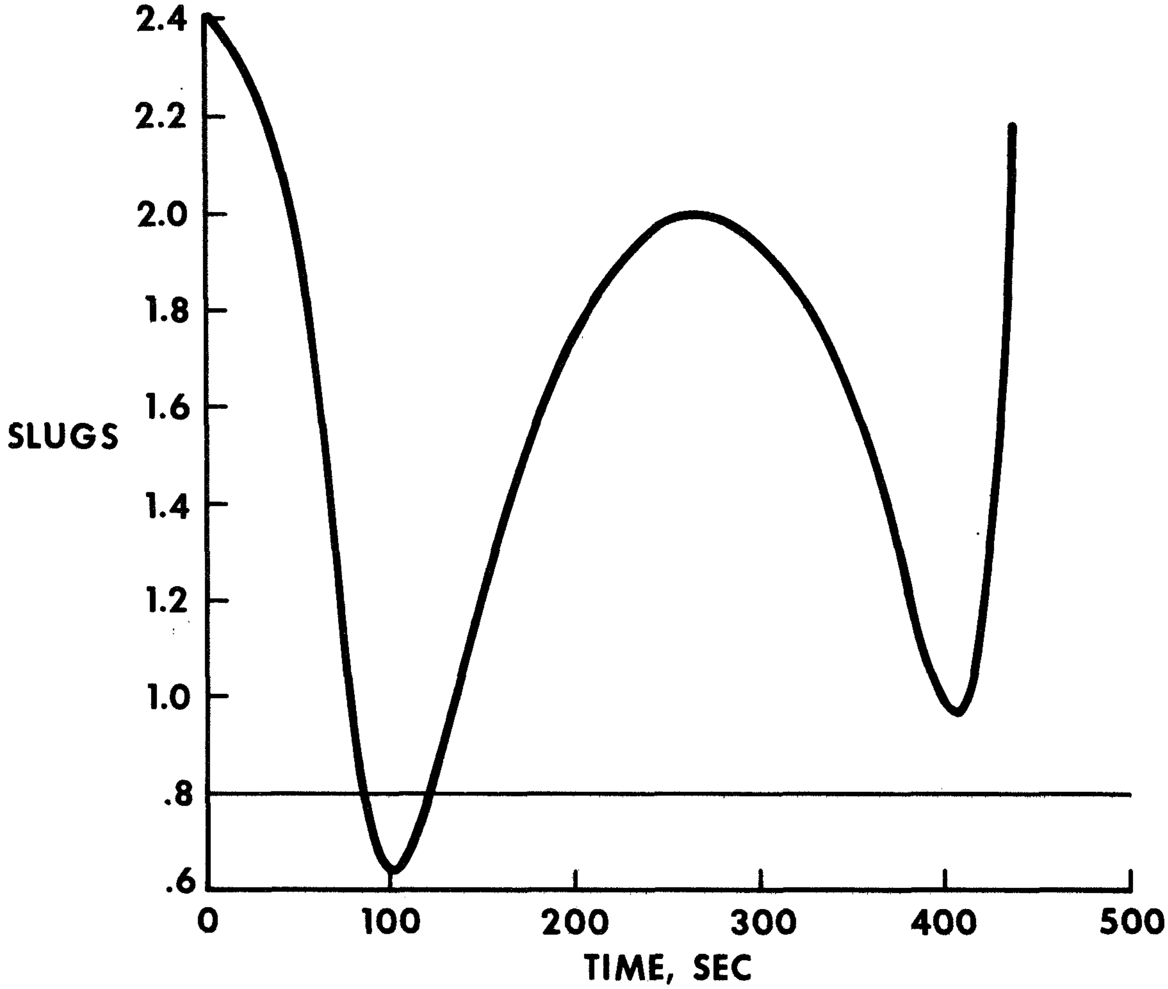


FIGURE 11

λ_V VERSUS TIME FOR M_f^2 CASE

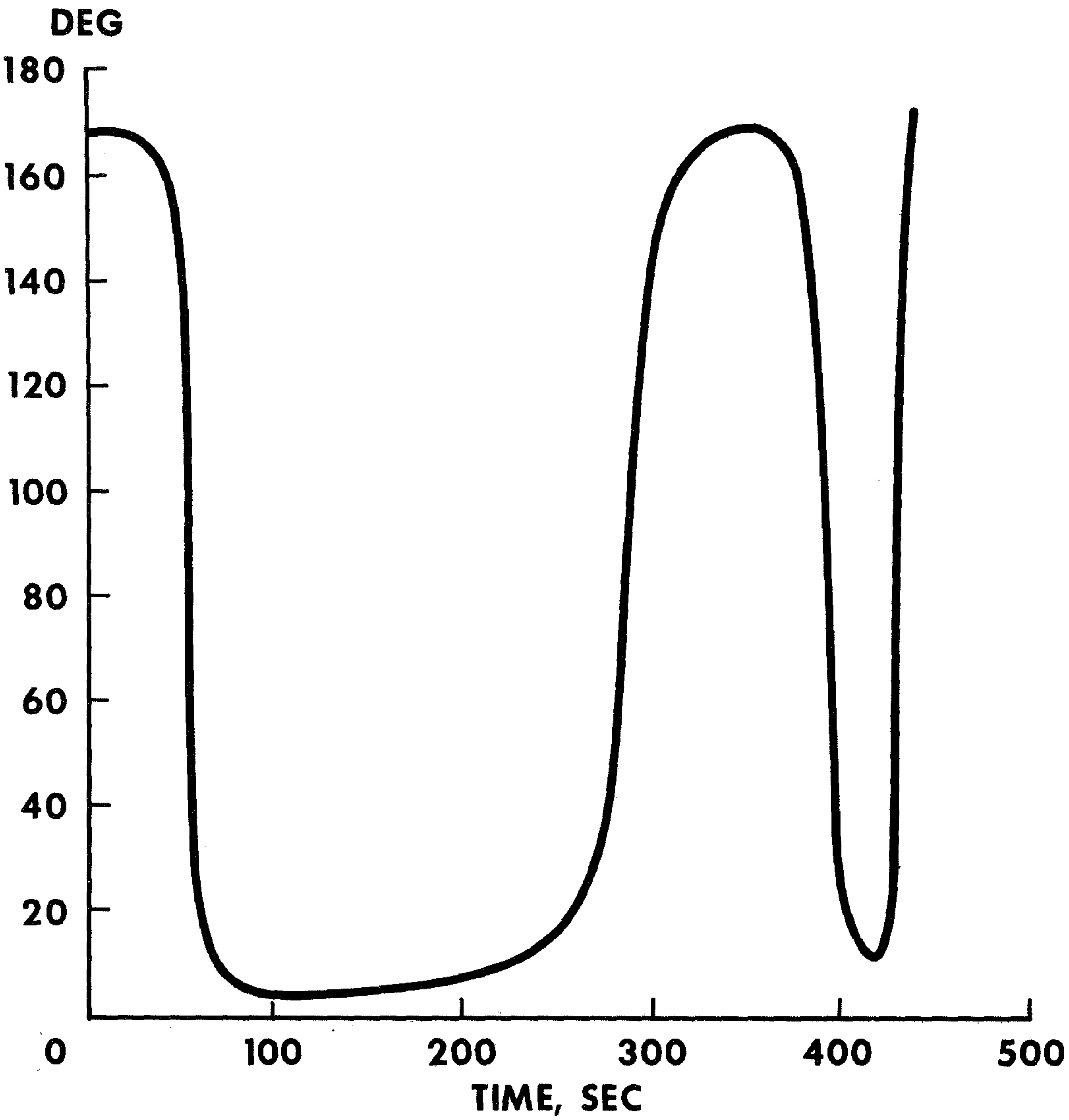


FIGURE 12

CONTROL U VERSUS TIME FOR THE M_f^1 CASE

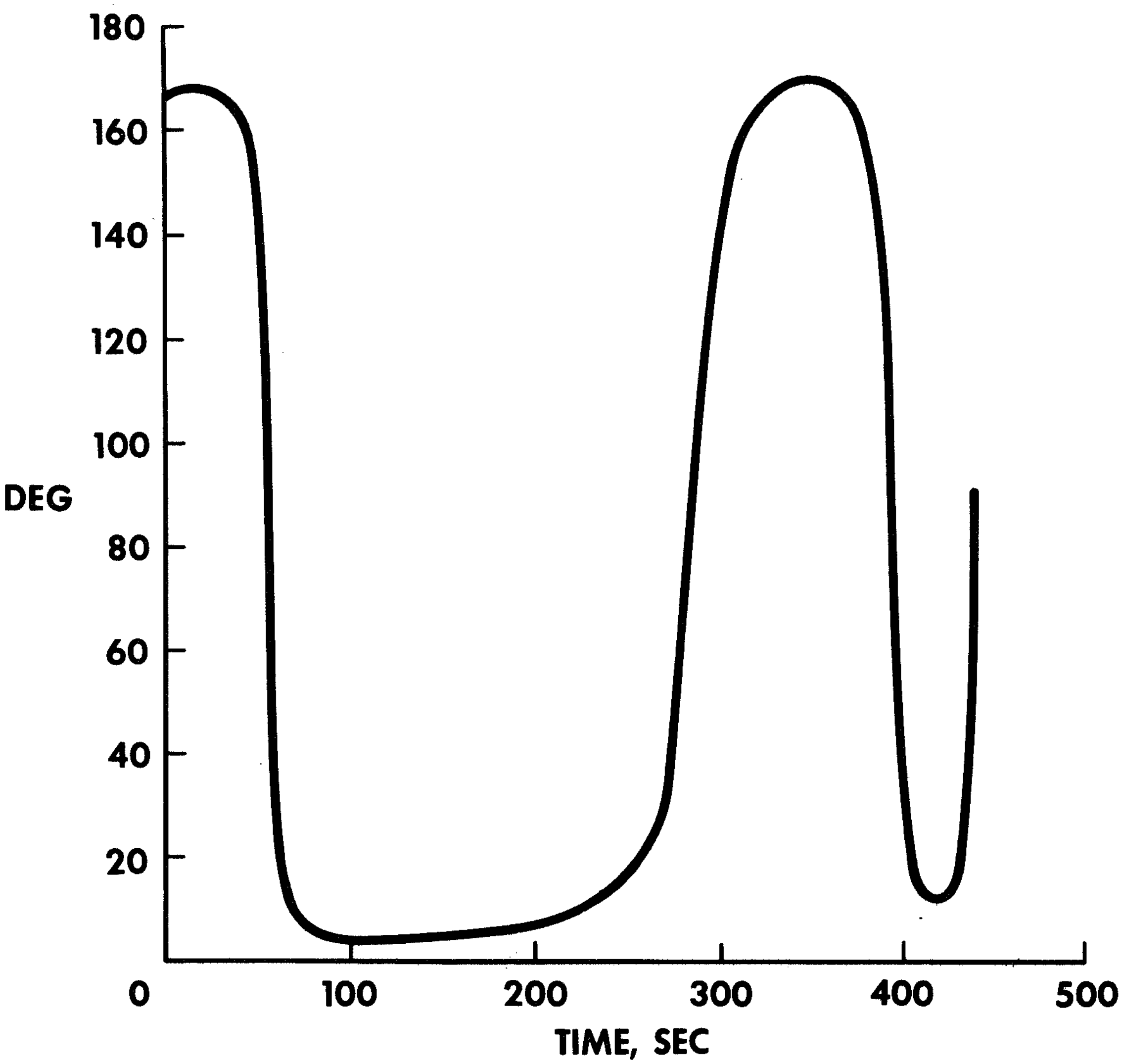


FIGURE 13

CONTROL U VERSUS TIME FOR THE M_f^2 CASE

Williamson's[39] MPF program. Values for the initial Lagrange multipliers and final time were specified in varying degrees of accuracy ranging from four to eight decimal places. Both programs were run on the University of Texas CDC 6600 digital computer using single-precision arithmetic everywhere except in the variable step-size numerical integrators where partial double-precision was used. Each integrator had a single-step error tolerance of $1.0E{-}10 \leq \varepsilon \leq 1.0E{-}08$. The correction procedure required correcting 100% of the terminal error after each iteration.

Results of this comparison study are summarized in Table VI. The

TABLE VI

MPF/MSM

Convergence Characteristics

for M_f^2 Case of the Apollo Reentry Problem

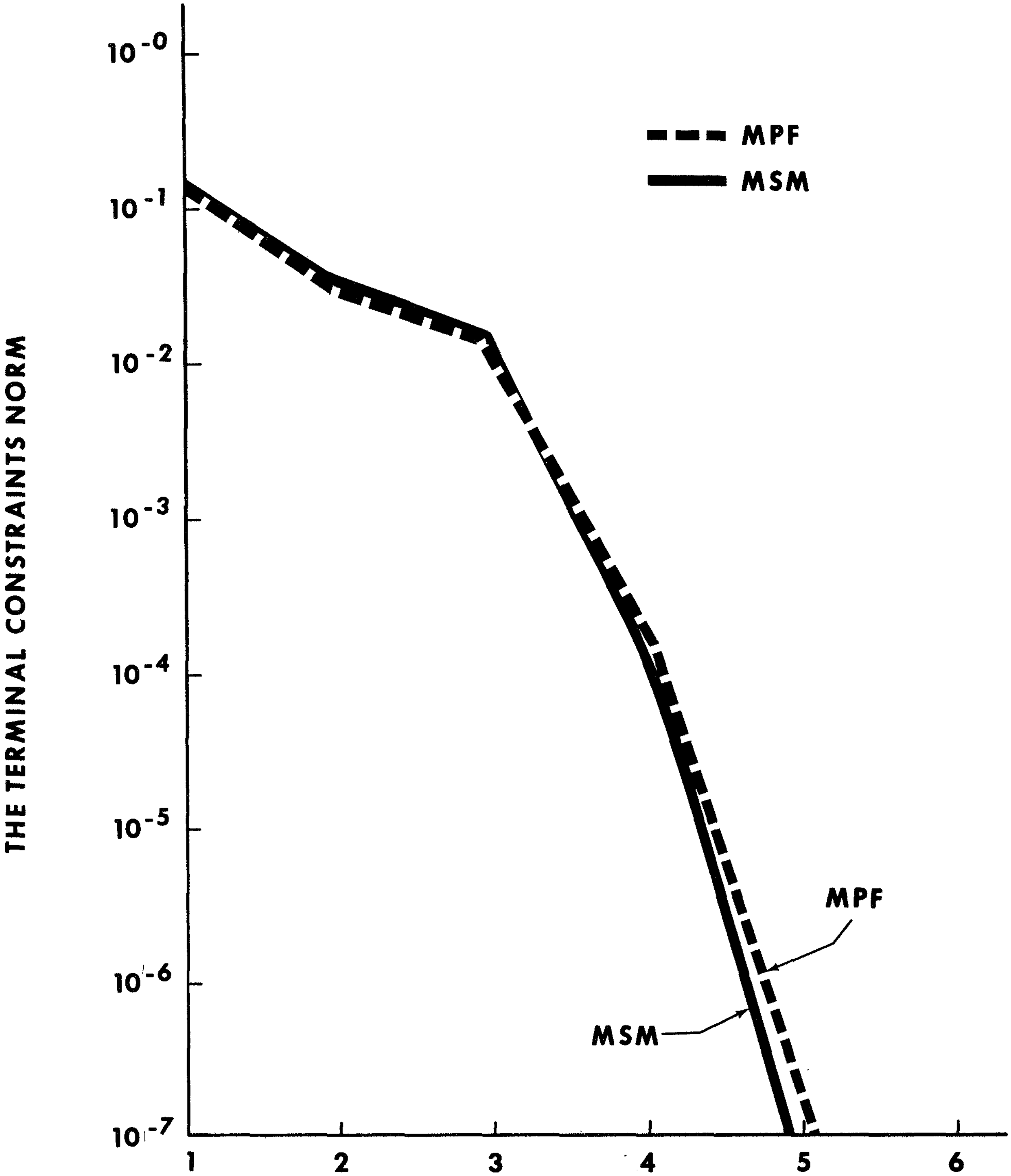
Significant Digits for Λ_o, t_f	Time to Converge CDC 6600 (Seconds)		% More Time Required by MSM	Number of Corrections Required	
	MPF	MSM		MPF	MSM
8	33	49	49%	1	1
6	66	106	60%	2	2
4	165	207	25%	5	4

MPF program required less time to converge in each case. However, with decreasing accuracy in the initial guesses for Λ_o and t_f, the MSM revealed a tendency toward fewer required corrections and more competitive time-to-converge. Terminal error norms of both the MPF and MSM programs for the case of four significant digits in Λ_o and t_f are shown by Figure 14.

No direct computational comparisons with the SSM were available. McGregor[22] used the SSM to converge the three-dimensional Apollo reentry problem for the case of terminally specified values for θ, Δ, and V. This particular case was converged using a fixed step-size integration routine. However, as was discussed previously, the MSM required a variable step-size numerical integration scheme to preserve the numerical integrity of the state and Lagrange multiplier equations. In addition, the MSM failed to converge this particular case of the three-dimensional Apollo reentry problem. This failure is currently under study by this author. Numerical comparison between the SSM and MPF for this particular case of the Apollo reentry problem can be found in the study by Tapley, et al.[37]

5.4 MSM Guidance Results

The MSM guidance scheme was implemented for the three-dimensional Apollo reentry M_f^2 case. A 5% perturbation in altitude, speed, and angle of attack was initiated at $t = 0$ seconds to study initial reentry condition perturbation effects. A similar perturbation was initiated at $t = 75$ seconds to correspond to initial peaks in spacecraft heating and acceleration. In either case, it was assumed desirable to



ITERATION COUNT

FIGURE 14

$M_f^{|2}$ CASE FOR APOLLO REENTRY PROBLEM correct so as to satisfy the originally specified terminal conditions. The guidance scheme then assumed that

$$y(t) = z(t) = 0 \qquad (t_o \leq t \leq t_f) \tag{64}$$

Control corrections reduced to

$$\delta u = \left[U_x + U_\lambda \; (K + D\pi_{11} + \ell\pi_{21})\right]\delta x \tag{65}$$

Numerical results revealed that the MSM guidance scheme failed to satisfy desired terminal conditions for specified vehicle state perturbations. Investigation revealed that numerical instabilities arising from attempts to forward-integrate the matrix Riccati equation were responsible for compromising effective terminal guidance. Further study to suppress these instabilities is needed to achieve an effective MSM guidance scheme.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

A new second-order variational method (the Modified Sweep Method) was developed for solving the two-point boundary value problem of trajectory optimization. If differs from the original Successive Sweep Method in that the iteration process is now associated with modifying the initial values of the Lagrange multipliers instead of the control function over the time interval of interest. This approach requires considerably less computer storage and yields the Eulerian control. The new method was tested successfully on several classes of problems. The following conclusions were reached about the Modified Sweep Method:

CONCLUSIONS

1. The method has appeal for problems in which knowledge of the Eulerian control is critical. The MSM yields the Eulerian control over the entire time interval of interest upon convergence.

2. Significantly less computer storage than the SSM was required. Only $3n + 2(q+1)$ quantities were required by the MSM algorithm to compute the desired corrections from one iteration to the next. This is a desirable characteristic for larger-dimensional problems and small-storage computers.

3. The numerical integration of a least $n+q$ less variables than the SSM is required. This feature is desirable because less computation time is required.

4. Rapid terminal convergence characteristics of second-order numerical optimization methods is retained by the MSM.

5. The conjugate-point test feature contained in the original SSM is also retained by the MSM.

6. A numerical comparison of the MSM with the MPF for the class of free final-time Brachistochrone problems revealed that the MSM possesses acceptable convergence envelopes and competitive time-to-converge features.

RECOMMENDATIONS

1. The basic nature of the generalized Riccati transformation technique for solving the linear two-point boundary value problem of control optimization should be studied. It is possible that other equivalent combinations might possess a better structure for solving the two-point boundary value problem than the combination presently being used.

2. Sensitivity of the MSM algorithm to classes of problems should be determined. This recommendation is made because of the method's failure to converge the three-dimensional Apollo reentry problem when terminal state values are specified for longitude, latitude and speed.

3. The correction procedure used with the MSM should be optimized such that the largest allowable correction is always attempted during a given iteration. This should be accomplished for the obvious reason of reducing computational costs by requiring fewer iterations.

4. Properties of the other Riccati transformation variables and their relations to the reference trajectory should be studied. Currently, only information about the Jacobi-Mayer conjugate point condition

and the abnormality condition is being extracted. This information is contained in only two matrices of the many used by the Riccati transformation technique.

5. The MSM algorithm should be extended to treat state as well as control inequality constraints. The need for this extension is obvious since most practical problems are subject to such constraints.

APPENDICES

APPENDIX A

THE INHOMOGENEOUS SET OF PERTURBATION EQUATIONS

Let $\bar{x}(t)$, $\bar{u}(t)$, and $\bar{\lambda}(t)$ be functions associated with an extreme trajectory for the functional to be optimized. With the assumptions made in Necessary Conditions, page 12, the following Euler-Lagrange equations are necessarily satisfied:

$$\dot{\bar{x}} = H_{\lambda}^{T}(\bar{x},\bar{u},\bar{\lambda},t) = f(x,u,t) \tag{A.1}$$

$$\dot{\bar{\lambda}} = -H_{x}^{T}(\bar{x},\bar{u},\bar{\lambda},t) \tag{A.2}$$

$$0 = H_{u}^{T}(\bar{x},\bar{u},\bar{\lambda},t) \tag{A.3}$$

where $(\bar{\ })$ indicates that the variables are to be evaluated on the extreme trajectory.

Now assume a nearby trajectory characterized by the $2n+m$ functions $x = \bar{x} + \delta x$, $u = \bar{u} + \delta u$, and $\lambda = \bar{\lambda} + \delta\lambda$. Substituting x, u, and λ into Equations (A.1) through (A.3) and expanding into a Taylor Series to first order about this nearby trajectory, the following equations are obtained

$$\delta\dot{x} = H_{\lambda x}\delta x + H_{\lambda u}\delta u \tag{A.4}$$

$$\delta\dot{\lambda} = -H_{xx}\delta x - H_{xu}\delta u - H_{x\lambda}\delta\lambda \tag{A.5}$$

$$\delta H_{u}^{T} = H_{ux}\delta x + H_{uu}\delta u + H_{u\lambda}\delta\lambda \tag{A.6}$$

where the partial derivatives of the Hamiltonian H are evaluated along the nearby trajectory.

Making the assumption that the (mxm) matrix H_{uu} is non-singular, the control corrections can be obtained from Equation (A.6) as

$$\delta u = H_{uu}^{-1} \left(\delta H_u^T - H_{ux} \delta x - H_{u\lambda} \delta \lambda \right) \tag{A.7}$$

Using this expression to eliminate δu from Equations (A.4) and (A.5) then gives

$$\begin{pmatrix} \delta \dot{x} \\ \delta \dot{\lambda} \end{pmatrix} = \begin{pmatrix} A & B \\ -C & -A^T \end{pmatrix} \begin{pmatrix} \delta x \\ \delta \lambda \end{pmatrix} + \begin{pmatrix} v \\ -w \end{pmatrix} \tag{A.8}$$

where

$$A \triangleq -H_{\lambda u} H_{uu}^{-1} H_{ux} + H_{\lambda x}$$

$$B \triangleq -H_{\lambda u} H_{uu}^{-1} H_{u\lambda}$$

$$C \triangleq -H_{xu} H_{uu}^{-1} H_{ux} + H_{xx}$$

$$v \triangleq H_{\lambda u} H_{uu}^{-1} \delta H_u^T$$

$$w \triangleq H_{xu} H_{uu}^{-1} \delta H_u^T$$

Equation (A.8) represents the inhomogeneous set of linear perturbation equations used by second-order variational methods. For computational purposes, the following is used:

$$\delta H_u^T(t) = -\varepsilon_u H_u^t(t) \quad , \quad 0 < \varepsilon_u \leq 1 \tag{A.9}$$

BOUNDARY CONDITIONS

Boundary conditions for these equations are derived in Appendix B. They are summarized here as

$$\delta x(t_o) = \delta \bar{x}_o = 0 \qquad (A.10)$$

and

$$\delta\lambda(t_f) = (P_{xx})_f \, \delta x_f + M_{x_f}^T d\nu + \alpha_f dt_f + \tau_1 \qquad (A.11)$$

where the vectors α and τ_1 are also defined in Appendix B.

APPENDIX B

FIRST-ORDER PERTURBATION OF TERMINAL CONDITIONS

To allow for changes in the variable final time from iteration to iteration, the following linear approximation is used throughout this section. For an appropriate variable; e.g., v_f, assume that $dv_f = \delta v_f + \dot{v}_f \, dt_f$.

The transversality conditions on the terminal values of the Lagrange multipliers are expressed by the condition

$$\Sigma_f^T = P_{x_f} - \lambda_f^T \tag{B.1}$$

A first-order perturbation of this condition gives

$$d\Sigma_f = (P_{xx})_f dx_f + (P_{x\nu})_f d\nu + (P_{xt})_f dt_f - d\lambda_f \tag{B.2}$$

Replacing dx_f and $d\lambda_f$ using the linear approximation stated in the first paragraph above, and grouping terms yields

$$d\Sigma_f = (P_{xx})_f \delta x_f + (P_{x\nu})_f d\nu + \left(\frac{DP_x^T}{Dt} - \dot{\lambda} \right)_f dt_f - \delta\lambda_f \tag{B.3}$$

Replacing $\dot{\lambda}$ by use of $\dot{\lambda} = -H_x^T$ and transposing the $\delta\lambda_f$ term to the left gives

$$\delta\lambda_f = (P_{xx})_f \delta x_f + M_{x_f}^T d\nu + \left(\frac{DP_x^T}{Dt} + H_x^T \right)_f dt_f - d\Sigma_f \tag{B.4}$$

The required terminal conditions on the state variables are given by

$$M_f = M(x_f, t_f) = 0 \tag{B.5}$$

Similarly, a first-order perturbation in the variables yields

$$dM_f = M_{x_f}\delta_{x_f} + \dot{M}_f dt_f \tag{B.6}$$

The transversality condition on the final value of the Hamiltonian is given by

$$\Omega_f \triangleq [P_t + H(x,u,\lambda,t)]_{t_f} \tag{B.7}$$

After using the linear assumptions stated in the first paragraph of this section, a first-order perturbation in the variables gives

$$\begin{aligned} d\Omega_f = {} & [(P_{tx} + H_x)]_f \delta x_f + (P_{t\nu})_f d\nu + (H_u)_f \delta u_f + (H_\lambda)_f \delta\lambda_f \\ & + \left\{(P_{tt} + H_t) + (P_{tx} + H_x)\dot{x} + H_u\dot{u} + H_\lambda\dot{\lambda}\right\}_f dt_f \end{aligned} \tag{B.8}$$

Eliminating δu_f using Equation (A.7), $\delta\lambda_f$ using Equation (B.4), and collecting terms gives

$$\begin{aligned} d\Omega_f = {} & \left[P_{tx} + H_x - H_u H_{uu}^{-1} H_{ux} + \left(H_\lambda - H_u H_{uu}^{-1} H_{u\lambda}\right) P_{xx}\right]_f \delta x_f \\ & + \left[P_{t\nu} + \left(H_\lambda - H_u H_{uu}^{-1} H_{u\lambda}\right) P_{x\nu}\right]_f d\nu \\ & + \left[\frac{D}{Dt}(P_t + H) + \left(H_\lambda - H_u H_{uu}^{-1} H_{u\lambda}\right)\left(\frac{DP_x^T}{Dt} + H_x^T\right)\right]_f dt_f \\ & + \left(H_u H_{uu}^{-1} \delta H_u^T\right)_f - \left(H_\lambda - H_u H_{uu} H_{u\lambda}\right)_f d\Sigma_f \end{aligned} \tag{B.9}$$

Substituting $\dot{x} = H_\lambda^T$, grouping terms again, and rewriting gives

$$
\begin{aligned}
d\Omega_f \;=\; & \left\{\alpha^T - H_u H_{uu}^{-1} \left(H_{ux} + H_{u\lambda} P_{xx}\right)\right\}_f \delta x_f \\
& + \left\{\left(\frac{DM^T}{Dt}\right) - H_u H_{uu}^{-1} H_{u\lambda} M_x^T\right\}_f d\nu \\
& + \left\{\frac{D}{Dt}\left(P_t + H\right) + \dot{x}^T \frac{DP_x^T}{Dt} + \dot{x}^T H_x^T - H_u H_{uu}^{-1} H_{u\lambda}\alpha\right\}_f dt_f \\
& + \left\{H_u H_{uu}^{-1} \delta H_u^T - \dot{x}^T d\Sigma_f + H_u H_{uu}^{-1} H_{u\lambda}\; d\Sigma_f\right\}_f \qquad \text{(B.10)}
\end{aligned}
$$

where

$$\alpha \triangleq \left(\frac{DP_x^T}{Dt} + H_x^T\right)_{t_f}$$

Manipulating the coefficient for the dt_f term, it is possible to rewrite this coefficient as

$$\left\{\frac{D}{Dt}\left(\frac{DP}{Dt} + Q\right) - \Sigma_f^T \frac{Df}{Dt} - H_u H_{uu}^{-1} H_{u\lambda}\alpha\right\}_f$$

In matrix form, Equations (B.4), (B.6), and (B.10) can be written as

$$
\begin{bmatrix} \delta\lambda_f \\ dM_f \\ d\Omega_f \end{bmatrix}
=
\begin{bmatrix} (P_{xx})_f & M_{x_f}^T & \alpha_f \\ M_{x_f} & 0 & \dot{M}_f \\ \alpha_f^T + \tau_2 & \dot{M}_f^T + \tau_3 & \beta_f + \tau_4 \end{bmatrix}
\begin{bmatrix} \delta x_f \\ d\nu \\ dt_f \end{bmatrix}
+
\begin{bmatrix} \tau_1 \\ 0 \\ \tau_5 \end{bmatrix}
\qquad \text{(B.11)}
$$

where

$$\alpha_f \triangleq \left(\frac{DP_x^T}{Dt} + H_x^T\right)_{t_f}$$

$$\beta_f \triangleq \frac{D}{Dt}\left(\frac{DP}{Dt} + Q\right)_{t_f}$$

$$\tau_1 \triangleq -d\Sigma_f$$

$$\tau_2 \triangleq -\left[H_u H_{uu}^{-1}(H_{ux} + H_{u\lambda} P_{xx})\right]_f$$

$$\tau_3 \triangleq -\left[H_u H_{uu}^{-1} H_{u\lambda} M_x^T\right]_f$$

$$\tau_4 \triangleq -\left\{\Sigma_f^T \frac{Df}{Dt} + H_u H_{uu}^{-1} H_{u\lambda} \alpha\right\}_f$$

and

$$\tau_5 \triangleq \left\{H_u H_{uu}^{-1} \delta H_u^T\right\}_f - \left(\dot{x}^T - H_u H_{uu}^{-1} H_{u\lambda}\right)_f \; d\Sigma_f$$

APPENDIX C

A MODIFIED SWEEP METHOD HAMILTONIAN WITH FIRST AND SECOND PARTIALS OF APOLLO 3-D REENTRY PROBLEM

Problem Statement

Minimize the real functional

$$J = \int_0^{t_f} \left(\frac{\sqrt{L^2 + D^2}}{m} + \tilde{\lambda}_o \rho^{1/2} V^3 \right) dt$$

subject to

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

$$\dot{\theta} = \frac{V}{(R + h)} \frac{\cos \gamma \ \cos A}{\cos \Delta}$$

$$\dot{\Delta} = \frac{V}{(R + h)} \cos \gamma \sin A$$

$$\dot{V} = - \frac{\mu}{(R + h)^2} \sin \gamma - \frac{D}{m}$$

$$\dot{\gamma} = - \frac{\mu}{(R + h)^2} \frac{\cos \gamma}{V} + \frac{V}{(R + h)} \cos \gamma + \frac{L}{m} \frac{\cos \beta}{V}$$

$$A = - \frac{V}{(R + h)} \frac{\cos \gamma \cos A \sin \Delta}{\cos \Delta} - \frac{L}{m} \frac{\sin \beta}{V \cos \gamma}$$

and satisfying the end conditions

$$\bar{x}(0) = \bar{C}_o \qquad \text{(a constant vector)}$$

$$\bar{M}(x_f, t_f) = 0 \qquad \{\text{ a (qx1) vector }\}$$

where

$$\rho = \rho_o e^{-\overset{*}{\beta}h}$$

$$L = \frac{1}{2} \rho S V^2 C_L$$

$$D = \frac{1}{2} \rho S V^2 C_D$$

The variational Hamiltonian is

$$H = \frac{1}{2m} \rho_o e^{-\overset{*}{\beta}h} S V^2 \sqrt{C_L^2 + C_D^2} + \tilde{\lambda}_o \rho_o^{1/2} e^{-\overset{*}{\beta}(h/2)} V^3 + \lambda_1 V \sin \gamma$$

$$+ \frac{V \cos \gamma}{(R + h)} \left[\frac{\cos A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) + (\lambda_3 \sin A + \lambda_5) \right]$$

$$- \frac{\mu}{(R + h)^2} \left[\lambda_4 \sin \gamma + \lambda_5 \frac{\cos \gamma}{V} \right]$$

$$- \frac{1}{2m} \rho_o e^{-\overset{*}{\beta}h} S V \left[\lambda_4 C_D V - C_L \left(\lambda_5 \cos \beta - \lambda_6 \frac{\sin \beta}{\cos \gamma} \right) \right]$$

Partials of the Apollo 3-D Hamiltonian With Respect to the Roll Angle

$$H_\beta = \frac{\partial H}{\partial \beta} = \frac{1}{2m} \rho_o e^{-\overset{*}{\beta}h} S V C_L \left[-\lambda_5 \sin \beta - \lambda_6 \frac{\cos \beta}{\cos \gamma} \right] = 0$$

This implies that

$$\lambda_5 \sin \beta + \lambda_6 \frac{\cos \beta}{\cos \gamma} = 0$$

or

$$\tan \beta = - \frac{\lambda_6}{\lambda_5 \cos \gamma}$$

Thus

$$\sin \beta = \frac{\lambda_6}{\pm\sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma}}$$

and

$$\cos \beta = \frac{-\lambda_5 \cos \beta}{\pm\sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma}}$$

Sufficiency Condition

$$H_{\beta\beta} = - \frac{1}{2m} \rho_o e^{-\overset{*}{\beta} h} SVC_L \left(\lambda_5 \cos \beta - \frac{\lambda_6 \sin \beta}{\cos \gamma} \right)$$

We require

$$H_{\beta\beta} > 0 \qquad \text{for a local minimum}$$

Thus

$$\left(\lambda_5 \cos \beta - \lambda_6 \frac{\sin \beta}{\cos \gamma} \right) < 0$$

or, using $H_\beta \doteq 0$

$$\left(- \frac{1}{\cos \gamma} \right) \left(\pm\sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma} \right) < 0$$

Requiring $- \frac{\pi}{2} < \gamma < \frac{\pi}{2}$ yields

$$0 < \cos\gamma < 1$$

Therefore, the "+" sign of the radical must be chosen for optimality in β.

Finally,

$$\sin\beta_{OPT} = \frac{\lambda_6}{\sqrt{\lambda_6^2 + \lambda_5^2 \cos^2\gamma}}$$

$$\cos\beta_{OPT} = \frac{-\lambda_5 \cos\gamma}{\sqrt{\lambda_6^2 + \lambda_5^2 \cos^2\gamma}}$$

The Optimal Hamiltonian With Respect to β

Eliminating $\sin\beta$ and $\cos\beta$ then yields

$$\tilde{H} = \frac{1}{2m}\rho_o e^{-\overset{*}{\beta}h} SV^2\sqrt{C_L^2 + C_D^2} + \tilde{\lambda}_o \rho_o^{1/2} e^{-\overset{*}{\beta}(h/2)} V^3$$

$$+ \lambda_1 V \sin\gamma + \frac{V\cos\gamma}{(R + h)}\left\{\frac{\cos A}{\cos\Delta}(\lambda_2 - \lambda_6 \sin\Delta)\right.$$

$$\left. + (\lambda_3 \sin A + \lambda_5)\right\} - \frac{\mu}{(R + h)^2}\left\{\lambda_4 \sin\gamma + \lambda_5 \frac{\cos\gamma}{V}\right\}$$

$$- \frac{1}{2m}\rho_o e^{-\overset{*}{\beta}h} SV\left\{\lambda_4 C_D V + \frac{C_L}{\cos\gamma}\sqrt{\lambda_6^2 + \lambda_5^2 \cos^2\gamma}\right\}$$

First Partials of $\tilde{H}$ With Respect to $\bar{x}$ and $\bar{\lambda}$

The Euler-Lagrange equations for the MSM are obtained from the first partial derivatives of the Hamiltonian $\tilde{H}$. These equations are given below, where

$$\dot{\lambda}_i = -\left(\frac{\partial \tilde{H}}{\partial x_i}\right) \quad , \quad \dot{x}_i = \left(\frac{\partial \tilde{H}}{\partial \lambda_i}\right) \quad \text{and} \quad i,j = 1, \ldots 6.$$

$$\tilde{H}_{x_1} = \frac{\partial \tilde{H}}{\partial h} = -\overset{*}{\beta}\,\frac{\rho_o}{2m}\,e^{-\overset{*}{\beta}h}SV^2\sqrt{C_L^2 + C_D^2} - \frac{\overset{*}{\beta}}{2}\,\tilde{\lambda}_o\rho_o^{1/2}e^{-\overset{*}{\beta}(h/2)}V^3$$

$$- \frac{V\cos\gamma}{(R+h)^2}\left[\frac{\cos A}{\cos\Delta}\,(\lambda_2 - \lambda_6 \sin\Delta) + (\lambda_3 \sin A + \lambda_5)\right]$$

$$+ \frac{2\mu}{(R+h)^3}\left[\lambda_4 \sin\gamma + \frac{\lambda_5 \cos\gamma}{V}\right]$$

$$+ \overset{*}{\beta}\,\frac{\rho_o}{2m}\,e^{-\overset{*}{\beta}h}SV\left[\lambda_4 C_D V + \frac{C_L}{\cos\gamma}\sqrt{\lambda_6^2 + \lambda_5^2\cos^2\gamma}\right]$$

$$\tilde{H}_{x_2} = \frac{\partial \tilde{H}}{\partial \theta} = 0$$

$$\tilde{H}_{x_3} = \frac{\partial \tilde{H}}{\partial \Delta} = \frac{V\cos\gamma}{(R+h)}\left[\frac{\cos A}{\cos^2\Delta}\,(\lambda_2 \sin\Delta - \lambda_6)\right]$$

$$\tilde{H}_{x_4} = \frac{\partial \tilde{H}}{\partial V} = \frac{\rho_o}{m}\,e^{-\overset{*}{\beta}h}SV\sqrt{C_L^2 + C_D^2} + 3\tilde{\lambda}_o\rho_o^{1/2}e^{-\overset{*}{\beta}(h/2)}V^2$$

$$+ \lambda_1 \sin\gamma + \frac{\cos\gamma}{(R+h)}\left[\frac{\cos A}{\cos\Delta}\,(\lambda_2 - \lambda_6 \sin\Delta)\right.$$

$$+ \left.(\lambda_3 \sin A + \lambda_5)\right] + \frac{\mu}{(R + h)^2} \frac{\lambda_5 \cos \gamma}{V^2}$$

$$- \frac{1}{2m} \rho_o e^{-\overset{*}{\beta} h} S \left[2\lambda_4 C_D V + \frac{C_L}{\cos \gamma} \sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma}\right]$$

$$\tilde{H}_{x_5} = \frac{\partial \tilde{H}}{\partial \gamma} = \lambda_1 V \cos \gamma - \frac{V \sin \gamma}{(R + h)} \left[\frac{\cos A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta)\right.$$

$$+ \left.(\lambda_3 \sin A + \lambda_5)\right] - \frac{\mu}{(R + h)^2} \left[\lambda_4 \cos \gamma - \frac{\lambda_5}{V} \sin \gamma \right]$$

$$- \frac{1}{2m} \rho_o e^{-\overset{*}{\beta} h} SVC_L \left[\frac{\lambda_6^2 \sin \gamma}{\cos^2 \gamma \sqrt{\lambda_6^2 + \cos^2 \gamma}}\right]$$

$$\tilde{H}_{x_6} = \frac{\partial \tilde{H}}{\partial A} = \frac{V \cos \gamma}{(R + h)} \left[- \frac{\sin A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) + \lambda_3 \cos A\right]$$

$$\tilde{H}_{\lambda_1} = \left(\frac{\partial \tilde{H}}{\partial \lambda_1}\right) = V \sin \gamma$$

$$\tilde{H}_{\lambda_2} = \left(\frac{\partial \tilde{H}}{\partial \lambda_2}\right) = \frac{V \cos \gamma \cos A}{(R + h) \cos \Delta}$$

$$\tilde{H}_{\lambda_3} = \left(\frac{\partial \tilde{H}}{\partial \lambda_3}\right) = \frac{V \cos \gamma \sin A}{(R + h)}$$

$$\tilde{H}_{\lambda_4} = \left(\frac{\partial \tilde{H}}{\partial \lambda_4}\right) = - \frac{\mu}{(R + h)^2} \sin \gamma - \frac{1}{2m} \rho_o e^{-\overset{*}{\beta} h} SV^2 C_D$$

$$\tilde{H}_{\lambda_5} = \left(\frac{\partial \tilde{H}}{\partial \lambda_5}\right) = \frac{V \cos \gamma}{(R + h)} - \frac{\mu}{(R + h)^2} \frac{\cos \gamma}{V}$$

$$- \frac{1}{2m} \rho_o e^{-\overset{*}{\beta} h} SVC_L \frac{\lambda_5 \cos \gamma}{\sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma}}$$

$$\tilde{H}_{\lambda_6} = \left(\frac{\partial \tilde{H}}{\partial \lambda_6}\right) = - \frac{V \cos \gamma \cos A \sin \Delta}{(R + h) \cos \Delta}$$

$$- \frac{1}{2m} \rho_o e^{-\overset{*}{\beta} h} \frac{SVC_L}{\cos \gamma} \frac{\lambda_6}{\sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma}}$$

<u>$H_{\lambda\lambda}$ Matrix</u>

$$\tilde{H}_{\lambda\lambda} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a & b \\ 0 & 0 & 0 & 0 & b & c \end{pmatrix}$$

where

$$a = - \frac{\rho_o}{2m} \frac{C_L SVe^{-\overset{*}{\beta} h}}{\left(\lambda_6^2 + \lambda_5^2 \cos^2 \gamma\right)^{3/2}} \lambda_6^2 \cos \gamma$$

$$b = + \frac{\rho_o}{2m} \frac{C_L SVe^{-\overset{*}{\beta} h}}{\left(\lambda_6^2 + \lambda_5^2 \cos^2 \gamma\right)^{3/2}} \lambda_5 \lambda_6 \cos \gamma$$

$$c = - \frac{\rho_o}{2m} \frac{C_L SVe^{-\overset{*}{\beta} h}}{\left(\lambda_6^2 + \lambda_5^2 \cos^2 \gamma\right)^{3/2}} \lambda_5^2 \cos \gamma$$

$\tilde{H}_{\lambda x}$ Matrix

$$\tilde{H}_{\lambda_1 x_1} = 0$$

$$\tilde{H}_{\lambda_1 x_2} = 0$$

$$\tilde{H}_{\lambda_1 x_3} = 0$$

$$\tilde{H}_{\lambda_1 x_4} = \sin \gamma$$

$$\tilde{H}_{\lambda_1 x_5} = V \cos \gamma$$

$$\tilde{H}_{\lambda_1 x_6} = 0$$

$$\tilde{H}_{\lambda_2 x_1} = - \frac{\dot{V}}{(R + h)^2} \cos \gamma \frac{\cos A}{\cos \Delta}$$

$$\tilde{H}_{\lambda_2 x_2} = 0$$

$$\tilde{H}_{\lambda_2 x_3} = \frac{V}{(R + h)} \frac{\cos \gamma \cos A \sin \Delta}{\cos^2 \Delta}$$

$$\tilde{H}_{\lambda_2 x_4} = \frac{\cos \gamma \cos A}{(R + h) \cos \Delta}$$

$$\tilde{H}_{\lambda_2 x_5} = - \frac{V}{(R + h)} \frac{\sin \gamma \cos A}{\cos \Delta}$$

$$\tilde{H}_{\lambda_2 x_6} = - \frac{V}{(R + h)} \frac{\cos \gamma \sin A}{\cos \Delta}$$

$$\tilde{H}_{\lambda_3 x_1} = - \frac{V}{(R + h)^2} \cos \gamma \sin A$$

$$\tilde{H}_{\lambda_3 x_2} = 0$$

$$\tilde{H}_{\lambda_3 x_3} = 0$$

$$\tilde{H}_{\lambda_3 x_4} = \frac{\cos \gamma}{(R + h)} \sin A$$

$$\tilde{H}_{\lambda_3 x_5} = - \frac{V}{(R + h)} \sin \gamma \sin A$$

$$\tilde{H}_{\lambda_3 x_6} = \frac{V}{(R + h)} \cos \gamma \cos A$$

$$\tilde{H}_{\lambda_4 x_1} = \frac{2\mu}{(R + h)^3} \sin \gamma + \overset{*}{\beta} \frac{\rho_o}{2m} SV^2 C_D e^{-\overset{*}{\beta} h}$$

$$\tilde{H}_{\lambda_4 x_2} = 0$$

$$\tilde{H}_{\lambda_4 x_3} = 0$$

$$\tilde{H}_{\lambda_4 x_4} = - \frac{\rho_o}{m} SVC_D e^{-\overset{*}{\beta} h}$$

$$\tilde{H}_{\lambda_4 x_5} = - \frac{\mu}{(R + h)^2} \cos \gamma$$

$$\tilde{H}_{\lambda_4 x_6} = 0$$

$$\tilde{H}_{\lambda_5 x_1} = - \frac{V \cos \gamma}{(R + h)^2} + \frac{2\mu}{(R + h)^3} \frac{\cos \gamma}{V} + \overset{*}{\beta} \frac{\rho_o}{2m} C_L SVe^{-\overset{*}{\beta} h} \frac{\lambda_5 \cos \gamma}{\sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma}}$$

$$\tilde{H}_{\lambda_5 x_2} = 0$$

$$\tilde{H}_{\lambda_5 x_3} = 0$$

$$\tilde{H}_{\lambda_5 x_4} = \frac{\cos \gamma}{(R + h)} + \frac{\mu}{(R + h)^2} \frac{\cos \gamma}{V^2} - \frac{\rho_o}{2m} C_L Se^{-\overset{*}{\beta} h} \frac{\lambda_5 \cos \gamma}{\sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma}}$$

$$\tilde{H}_{\lambda_5 x_5} = - \frac{V \sin \gamma}{(R + h)} + \frac{\mu}{(R + h)^2} \frac{\sin \gamma}{V} - \frac{\rho_o}{2m} C_L SVe^{-\overset{*}{\beta} h} \lambda_5 \tau_2$$

$$\tilde{H}_{\lambda_5 x_6} = 0$$

where

$$\tau_2 = -\frac{\lambda_6^2 \sin\gamma}{\left(\lambda_6^2 + \lambda_5^2 \cos^2\gamma\right)^{3/2}}$$

$$\tilde{H}_{\lambda_6 x_1} = \frac{V \cos\gamma}{(R + h)^2} \frac{\cos A \sin\Delta}{\cos\Delta} + \overset{*}{\beta} \frac{\rho_o}{2m} \frac{C_L SVe^{-\overset{*}{\beta}h}}{\cos\gamma} \frac{\lambda_6}{\sqrt{\lambda_6^2 + \lambda_5^2 \cos^2\gamma}}$$

$$\tilde{H}_{\lambda_6 x_2} = 0$$

$$\tilde{H}_{\lambda_6 x_3} = -\frac{V}{(R + h)} \frac{\cos\gamma \cos A}{\cos^2\Delta}$$

$$\tilde{H}_{\lambda_6 x_4} = -\frac{\cos\gamma}{(R + h)} \frac{\cos A \sin\Delta}{\cos\Delta} - \frac{\rho_o}{2m} \frac{C_L Se^{-\overset{*}{\beta}h}}{\cos\gamma} \frac{\lambda_6}{\sqrt{\lambda_6^2 + \lambda_5^2 \cos^2\gamma}}$$

$$\tilde{H}_{\lambda_6 x_5} = \frac{V}{(R + h)} \frac{\sin\gamma \cos A \sin\Delta}{\cos\Delta} - \frac{\rho_o}{2m} C_L SVe^{-\overset{*}{\beta}h} \lambda_6 \tau_3$$

$$\tilde{H}_{\lambda_6 x_6} = \frac{V}{(R + h)} \frac{\cos\gamma \sin A \sin\Delta}{\cos\Delta}$$

where

$$\tau_3 = \frac{\sin\gamma}{\cos^2\gamma} \frac{\left(\lambda_6^2 + 2\lambda_5^2 \cos^2\gamma\right)}{\left(\lambda_6^2 + \lambda_5^2 \cos^2\gamma\right)^{3/2}}$$

$\tilde{H}_{xx}$ Matrix

$$\tilde{H}_{x_1x_1} = \overset{*}{\beta}{}^2 \frac{\rho_o}{2m} e^{-\overset{*}{\beta}h} SV^2 \sqrt{C_L^2 + C_D^2} + \frac{\overset{*}{\beta}{}^2}{4} \tilde{\lambda}_o \rho_o^{1/2} e^{-\overset{*}{\beta}(h/2)} V^3$$

$$+ \frac{2V \cos \gamma}{(R + h)^3} \left[\frac{\cos A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) + (\lambda_3 \sin A + \lambda_5) \right]$$

$$- \frac{6\mu}{(R + h)^4} \left[\lambda_4 \sin \gamma + \lambda_5 \frac{\cos \gamma}{V} \right]$$

$$- \overset{*}{\beta}{}^2 \frac{\rho_o}{2m} e^{-\overset{*}{\beta}h} SV \left[\lambda_4 C_D V + \frac{C_L}{\cos \gamma} \sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma} \right]$$

$$\tilde{H}_{x_1x_2} = 0$$

$$\tilde{H}_{x_1x_3} = - \frac{V \cos \gamma}{(R + h)^2} \left[\frac{\cos A}{\cos^2 \Delta} (\lambda_2 \sin \Delta - \lambda_6) \right]$$

$$\tilde{H}_{x_1x_4} = -\overset{*}{\beta} \frac{\rho_o}{m} e^{-\overset{*}{\beta}h} SV \sqrt{C_L^2 + C_D^2} - \frac{3}{2} \overset{*}{\beta} \tilde{\lambda}_o \rho_o^{1/2} e^{-\overset{*}{\beta}(h/2)} V^2$$

$$- \frac{\cos \gamma}{(R + h)^2} \left[\frac{\cos A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) + (\lambda_3 \sin A + \lambda_5) \right]$$

$$- \frac{2\mu}{(R + h)^3} \frac{\lambda_5 \cos \gamma}{V^2}$$

$$+ \ \overset{*}{\beta} \frac{\rho_o}{2m} e^{-\overset{*}{\beta} h} S \left[2\lambda_4 C_D V + \frac{C_L}{\cos \gamma} \sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma} \right]$$

$$\tilde{H}_{x_1 x_5} = \frac{V \sin \gamma}{(R + h)^2} \left[\frac{\cos A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) + (\lambda_3 \sin A + \lambda_5) \right]$$

$$+ \ \frac{2\mu}{(R + h)^3} \left[\lambda_4 \cos \gamma - \frac{\lambda_5 \sin \gamma}{V} \right]$$

$$+ \ \overset{*}{\beta} \frac{\rho_o}{2m} S V e^{-\overset{*}{\beta} h} C_L \left[\frac{\lambda_6^2 \sin \gamma}{\cos^2 \gamma \sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma}} \right]$$

$$\tilde{H}_{x_1 x_6} = - \frac{V \cos \gamma}{(R + h)^2} \left[- \frac{\sin A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) + \lambda_3 \cos A \right]$$

$$\tilde{H}_{x_2 x_1} = 0$$

$$\tilde{H}_{x_2 x_2} = 0$$

$$\tilde{H}_{x_2 x_3} = 0$$

$$\tilde{H}_{x_2 x_4} = 0$$

$$\tilde{H}_{x_2 x_5} = 0$$

$$\tilde{H}_{x_2 x_6} = 0$$

$$\tilde{H}_{x_3 x_1} = -\frac{V \cos \gamma}{(R + h)^2} \left[\frac{\cos A}{\cos^2 \Delta} (\lambda_2 \sin \Delta - \lambda_6) \right]$$

$$\tilde{H}_{x_3 x_2} = 0$$

$$\tilde{H}_{x_3 x_3} = \frac{V \cos \gamma}{(R + h)} \left[\frac{\cos A}{\cos^3 \Delta} \left[\lambda_2 (1 + \sin^2 \Delta) - 2\lambda_6 \sin \Delta \right] \right]$$

$$\tilde{H}_{x_3 x_4} = \frac{\cos \gamma}{(R + h)} \left[\frac{\cos A}{\cos^2 \Delta} (\lambda_2 \sin \Delta - \lambda_6) \right]$$

$$\tilde{H}_{x_3 x_5} = -\frac{V \sin \gamma}{(R + h)} \left[\frac{\cos A}{\cos^2 \Delta} (\lambda_2 \sin \Delta - \lambda_6) \right]$$

$$\tilde{H}_{x_3 x_6} = -\frac{V \cos \gamma}{(R + h)} \left[\frac{\sin A}{\cos^2 \Delta} (\lambda_2 \sin \Delta - \lambda_6) \right]$$

$$\tilde{H}_{x_4 x_1} = -\overset{*}{\beta} \rho_o SV \frac{\sqrt{C_L^2 + C_D^2}}{m} e^{-\overset{*}{\beta} h} - \frac{3}{2} \overset{*}{\beta} \tilde{\lambda}_o \rho_o^{1/2} V^2 e^{-\overset{*}{\beta}(h/2)}$$

$$- \frac{\cos \gamma}{(R + h)^2} \left[\frac{\cos A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) + (\lambda_3 \sin A + \lambda_5) \right]$$

$$- \frac{2\mu}{(R + h)^3} \frac{\lambda_5 \cos \gamma}{V^2}$$

$$+ \overset{*}{\beta} \frac{\rho_o S e^{-\overset{*}{\beta} h}}{2m} \left[2\lambda_4 C_D V + \frac{C_L}{\cos \gamma} \sqrt{\lambda_6^2 + \lambda_5^2 \cos^2 \gamma} \right]$$

$$\tilde{H}_{x_4x_2} = 0$$

$$\tilde{H}_{x_4x_3} = \frac{\cos\gamma}{(R + h)} \left[\frac{\cos A}{\cos^2 \Delta} (\lambda_2 \sin \Delta - \lambda_6) \right]$$

$$\begin{aligned}
\tilde{H}_{x_4x_4} &= \rho_o S \frac{\sqrt{C_L^2 + C_D^2}}{m} e^{-\overset{*}{\beta} h} + 6\tilde{\lambda}_o \rho_o^{1/2} V e^{-\overset{*}{\beta}(h/2)} \\
&- \frac{2\mu}{(R + h)^2} \left(\frac{\lambda_5 \cos\gamma}{V^3} \right) - \frac{\rho_o}{m} S e^{-\overset{*}{\beta} h} \left(\lambda_4 C_D \right)
\end{aligned}$$

$$\begin{aligned}
\tilde{H}_{x_4x_5} &= \lambda_1 \cos\gamma - \frac{\sin\gamma}{(R + h)} \left\{ \frac{\cos A}{\cos\Delta} (\lambda_2 - \lambda_6 \sin\Delta) \right. \\
&+ \left. (\lambda_3 \sin A + \lambda_5) \right\} - \frac{\mu}{(R + h)^2} \frac{\lambda_5 \sin\gamma}{V^2} \\
&- \frac{\rho_o}{2m} S e^{-\overset{*}{\beta} h} C_L \left[\frac{\lambda_6^2 \sin\gamma}{\cos^2\gamma \sqrt{\lambda_6^2 + \lambda_5^2 \cos^2\gamma}} \right]
\end{aligned}$$

$$\tilde{H}_{x_4x_6} = \frac{\cos\gamma}{(R + h)} \left[- \frac{\sin A}{\cos\Delta} (\lambda_2 - \lambda_6 \sin\Delta) + \lambda_3 \cos A \right]$$

$$\begin{aligned}
\tilde{H}_{x_5x_1} &= \frac{V \sin\gamma}{(R + h)^2} \left\{ \frac{\cos A}{\cos\Delta} (\lambda_2 - \lambda_6 \sin\Delta) + (\lambda_3 \sin A + \lambda_5) \right\} \\
&+ \frac{2\mu}{(R + h)^3} \left(\lambda_4 \cos\gamma - \frac{\lambda_5 \sin\gamma}{V} \right)
\end{aligned}$$

$$+ \quad \overset{*}{\beta}\rho_o \frac{SVe^{-\overset{*}{\beta}h}}{2m} C_L \left[\frac{\lambda_6^2 \sin\gamma}{\cos^2\gamma \sqrt{\lambda_6^2 + \lambda_5^2 \cos^2\gamma}} \right]$$

$$\tilde{H}_{x_5x_2} = 0$$

$$\tilde{H}_{x_5x_3} = -\frac{V \sin\gamma}{(R+h)} \left[\frac{\cos A}{\cos^2\Delta} (\lambda_2 \sin\Delta - \lambda_6) \right]$$

$$\tilde{H}_{x_5x_4} = \lambda_1 \cos\gamma - \frac{\sin\gamma}{(R+h)} \left[\frac{\cos A}{\cos\Delta} (\lambda_2 - \lambda_6 \sin\Delta) \right.$$

$$\left. + \quad (\lambda_3 \sin A + \lambda_5) \right] - \frac{\mu}{(R+h)^2} \left[\frac{\lambda_5 \sin\gamma}{V^2} \right]$$

$$- \quad \frac{\rho_o}{2m} Se^{-\overset{*}{\beta}h} C_L \left[\frac{\lambda_6^2 \sin\gamma}{\cos^2\gamma \sqrt{\lambda_6^2 + \lambda_5^2 \cos^2\gamma}} \right]$$

$$\tilde{H}_{x_5x_5} = -\lambda_1 V \sin\gamma - \frac{V \cos\gamma}{(R+h)} \left[\frac{\cos A}{\cos\Delta} (\lambda_2 - \lambda_6 \sin\Delta) \right.$$

$$\left. + \quad (\lambda_3 \sin A + \lambda_5) \right] + \frac{\mu}{(R+h)^2} \left[\lambda_4 \sin\gamma + \frac{\lambda_5 \cos\gamma}{V} \right]$$

$$- \quad \frac{\rho_o}{2m} SVe^{-\overset{*}{\beta}h} C_L \tau_4$$

where

$$\tau_4 = \frac{\lambda_6^2}{\cos\gamma \left(\lambda_6^2 + \lambda_5^2 \cos^2\gamma \right)^{3/2}} \left[\left(\lambda_6^2 + \lambda_5^2 \cos^2\gamma \right) \right.$$

$$\cdot \quad (1 + 2 \tan^2 \gamma) + \lambda_5^2 \sin^2 \gamma \Big]$$

$$\tilde{H}_{x_5 x_6} = - \frac{V \sin \gamma}{(R + h)} \left[- \frac{\sin A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) + \lambda_3 \cos A\right]$$

$$\tilde{H}_{x_6 x_1} = - \frac{V \cos \gamma}{(R + h)^2} \left[- \frac{\sin A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) + \lambda_3 \cos A \right]$$

$$\tilde{H}_{x_6 x_2} = 0$$

$$\tilde{H}_{x_6 x_3} = \frac{V \cos \gamma}{(R + h)} \left[- \frac{\sin A}{\cos^2 \Delta} (\lambda_2 \sin \Delta - \lambda_6)\right]$$

$$\tilde{H}_{x_6 x_4} = \frac{\cos \gamma}{(R + h)} \left[- \frac{\sin A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) + \lambda_3 \cos A\right]$$

$$\tilde{H}_{x_6 x_5} = - \frac{V \sin \gamma}{(R + h)} \left[- \frac{\sin A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) + \lambda_3 \cos A\right]$$

$$\tilde{H}_{x_6 x_6} = \frac{V \cos \gamma}{(R + h)} \left[- \frac{\cos A}{\cos \Delta} (\lambda_2 - \lambda_6 \sin \Delta) - \lambda_3 \sin A\right]$$

BIBLIOGRAPHY

1. Bliss, G.A., Lectures on the Calculus of Variations. The University of Chicago Press, 1946.

2. Breakwell, J.V., and Y.C. Ho., "On the Conjugate Point Condition for the Control Problem," International Journal of Engineering Science, Vol. II, Pergamon Press, Ltd., 1965.

3. Breakwell, J.V., J.L. Speyer, and A.E. Bryson., "Optimization and Control of Nonlinear Systems Using the Second Variation," S.I.A.M. Journal on Control 1:193-233,1963.

4. Bryson, A.E., and W.F. Denham., "A Steepest-Ascent Method for Solving Optimum Programming Problems," Raytheon Report BR-1303, Raytheon Company, Missile and Space Division, Bedford, Massachusetts, August 1961.

5. Courant, R., "Variational Methods for the Solution of Problems of Equilibriums and Vibrations," Bull. Am. Math. Soc., 49:1-23,1943.

6. Gelfand, I.M., and S.V. Fomin., Calculus of Variations. Trans. R.A. Silverman, Englewood Cliffs: Prentice-Hall, 1966.

7. Goodman, T.R., and G.N. Lance., "The Numerical Integration of Two-Point Boundary Value Problems," Math Tables and Other Aids to Computation, Vol. X, 1956, p. 82.

8. Hestenes, M.R., Variational Theory and Optimal Control Theory, Computing Methods in Optimization Problems, Chapter 1, Academic Press, New York, 1964.

9. Jacobson, D.H., New Second-Order and First-Order Algorithms for Determining Optimal Control: A Differential Dynamic Programming Approach, Technical Report 551, Division of Engineering and Applied Physics, Harvard University, February, 1968.

10. Jazwinski, A.H., "Optimal Trajectories and Linear Control of Nonlinear Systems," A.I.A.A. Journal, Vol. II, No. 8, August 1964, p. 1371.

11. Jurovics, S.A., and J.E. McIntyre. "The Adjoint Method and Its Application to Trajectory Optimization," ARS Journal, September 1962, p. 1354.

12. Kelley, H.J., "Gradient Theory of Optimal Flight Paths," ARS Journal (now A.I.A.A. Journal), October 1960, p. 947.

13. Kelley, H.J., R.E. Kopp, and H.G. Moyer, "A Trajectory Optimization Technique Based Upon the Theory of the Second Variation," A.I.A.A. Astrodynamics Conference, Yale University, 1963.

14. Lastman, G.J., "On the Equivalence of Three Trajectory Optimization Schemes," EMRL Report. RM 1020, The University of Texas,Austin, February 1967.

15. Lastman, G.J., and W.T. Fowler, "FORTRAN Subroutines for the Numerical Integration of First-Order Ordinary Differential Equations," EMRL Report. RM 1024, The University of Texas, Austin, March 1967.

16. Lewallen, J.M., "An Analysis and Comparison of Several Trajectory Optimization Methods." Ph.D. Dissertation, The University of Texas, Austin, 1966.

17. Long, R.S., "Newton-Raphson Operator Problems With Undetermined End Points," A.I.A.A. Journal, Vol. III, No. 7, July 1965, p. 1352.

18. Longmuir, A.G., and E.V. Bohn, "Second-Variation Methods in Dynamic Optimization," Journal of Optimization Theory and Applications, Vol. III, No. 3, 1969, pp. 164-173.

19. Mayne, D., "A Second-Order Gradient Method for Determining Optimal Trajectories of Non-Linear Discrete-Time Systems," International Journal on Control, Vol. III, No. 1, 1966, pp. 85-95.

20. McGill, R., and P. Kenneth, "Solution of Variational Problems by Means of a Generalized Newton-Raphson Operator," A.I.A.A. Journal, Vol. II, No. 10, October 1964.

21. McGill, R., and P. Kenneth, "A Convergence Theorem on the Iterative Solution of Nonlinear Two-Point Boundary Value Systems." Paper presented at the XIV International Astronautical Congress, Paris, September 25, 1963.

22. McGregor, C.O., "Numerical Optimization of Atmospheric Reentry Trajectories by the Sweep Method." Ph.D. Dissertation, The University of Texas, Austin, 1969.

23. McReynolds, S.R., "A Successive Sweep Method for Solving Optimal Programming Problems." Ph.D. Dissertation, Harvard University, Cambridge, 1965.

24. McReynolds, S.R., "The Successive Sweep Method and Dynamic Programming," Journal of Mathematical Analysis and Applications, Vol. XIX, No. 3, September 1967.

25. McReynolds, S.R., and A. E. Bryson, "A Successive Sweep Method for Solving Optimal Programming Problems," Cruft Technical Report No. 463, March 2, 1965.

26. Merriam, C.W., "An Algorithm for the Iterative Solution of a Class of Two-Point Boundary Value Problems," Journal of the Society of Industrial and Applied Mathematics Control, Ser. A, Vol. II, No. 1, 1964, pp. 1-10.

27. Mitter, S.K., "Successive Approximation Methods for the Solution of Optimal Control Problems," Automatica, Vol. III, Pergamon Press, 1966, pp. 135-149.

28. Narha, J.E., and W. H. Berry, "The Riccati Second-Variation Optimization Method," Vol. I of TRW Note No. 69 FMT-722. Acc. No. 11176-6027-RO-00, TRW Systems, Houston, Texas.

29. Omicioli, G.J., "The Riccati Second-Variation Optimization Program," Vol. II of TRW Note No. 69 FMT-723. Acc. No. 11176-6028-RO-00, TRW Systems, Houston, Texas, January 1969.

30. Pontryagin, L.S., V.G. Boltyanskii, R.V. Gamkrelidze, and G.F. Mishchenko, The Mathematical Theory of Optimal Processes, John Wiley and Sons, New York, 1962.

31. Rybicki, G.B., and P.D. Usher, "The Generalized Riccati Transformation as a Simple Alternative to Invariant Imbedding," The Astrophysical Journal, Vol. CXLVI, No. 3, December 1966.

32. Schley, C.H., and I. Lee, "Optimal Control Computation by the Newton-Raphson Method and the Riccati Transformation," IEEE Transactions on Automatic Control, Vol. AC-XII, No. 2, April 1967, p. 139.

33. Schwaush, O.A., "Subroutine STEPER - A FORTRAN Subroutine for the Numerical Integration of First-Order Ordinary Differential Equations Using Either a Fixed or Variable Integration Step-Size." Catalogue No. 186, Lockheed Electronics Company, Houston, Texas, April 1969.

34. Speyer , J.L., and A.E. Bryson, "A Neighboring Optimum Feedback Control Scheme Based on Estimated Time to go With Applications to Reentry Flight Paths," A.I.A.A. Journal, Vol. VI, No. 5, May 1968, pp. 769-776.

35. Sutherland, J.W., and E.V. Bohn, "A Numerical Trajectory Optimization Method Suitable for a Computer of Limited Memory," IEEE Transactions on Automatic Control, Vol. AC-XI, No. 3, July 1966, pp. 440-447.

36. Sylvester, R.J., and F. Meyer, "Two-Point Boundary Problems by Quasilinearization," Jour. Soc. Indust. Appl. Math, Vol. XIII, No. 2, June 1965.

37. Tapley, B.D., W.T. Fowler, and W.E. Williamson, "The Computation of Optimal Apollo-Type Reentry Trajectories." Paper presented at the Joint Automatic Control Conference, Boulder, Colorado, August 1969.

38. Tapley, B.D., and J.M. Lewallen, "Comparison of Several Numerical Optimization Methods," Journal of Optimization Theory and Application, Vol. I, No. 1, July 1967.

39. Williamson, W.E., Private communications about Ph.D. dissertation. (To be published.)

Houston, Texas