1977

Optimal and suboptimal numerical solutions to a class of optimal control problems with applications to sailplane dynamics

Lawrence James Genalo

Iowa State University

Follow this and additional works at: https://lib.dr.iastate.edu/rtd

Part of the Mathematics Commons

Recommended Citation
Genalo, Lawrence James, "Optimal and suboptimal numerical solutions to a class of optimal control problems with applications to sailplane dynamics" (1977). Retrospective Theses and Dissertations. 5825.
https://lib.dr.iastate.edu/rtd/5825

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.
GENALO, Lawrence James, 1946-
OPTIMAL AND SUBOPTIMAL NUMERICAL SOLUTIONS
TO A CLASS OF OPTIMAL CONTROL PROBLEMS WITH
APPLICATIONS TO SAILPLANE DYNAMICS.

Iowa State University, Ph.D., 1977
Mathematics

Xerox University Microfilms, Ann Arbor, Michigan 48106
Optimal and suboptimal numerical solutions
to a class of optimal control problems
with applications to sailplane dynamics

by

Lawrence James Genalo

A Dissertation Submitted to the
Graduate Faculty in Partial Fulfillment of
The Requirements for the Degree of

DOCTOR OF PHILOSOPHY

Department: Mathematics
Major: Applied Mathematics

Approved:

Signature was redacted for privacy.

In Charge of Major Work

Signature was redacted for privacy.

For the Major Department

Signature was redacted for privacy.

For the Graduate College

Iowa State University
Ames, Iowa
1977
# TABLE OF CONTENTS

| CHAPTER I. | INTRODUCTION AND PROBLEM DEVELOPMENT | 1 |
| CHAPTER II. | OPTIMAL SOLUTIONS | 18 |
| Section 2.1. | Conjugate Gradient Algorithm | 18 |
| Section 2.2. | Quasilinearization Algorithm | 26 |
| Section 2.3. | Sufficiency Conditions | 40 |
| CHAPTER III. | SUBOPTIMAL SOLUTIONS | 56 |
| Section 3.1. | Linearized Two-Point Boundary Value Problem | 56 |
| Section 3.2. | Linear/Quadratic Version | 57 |
| Section 3.3. | Matched Asymptotic Expansions | 67 |
| CHAPTER IV. | EXTENSIONS OF THE THERMAL MODEL | 87 |
| Section 4.1. | Distributed End-Point Thermal | 87 |
| Section 4.2. | Distributed Mid-Range Thermal | 94 |
| CHAPTER V. | CONCLUSIONS AND SUGGESTIONS FOR FURTHER STUDY | 105 |
LITERATURE CITED

ACKNOWLEDGEMENTS

APPENDIX A. MACCREDY EQUILIBRIUM GLIDE
TRAJECTORY

APPENDIX B. FORTRAN PROGRAM LISTING AND SAMPLE
RUN FOR THE QUASILINEARIZATION
ALGORITHM

APPENDIX C. FORTRAN PROGRAM LISTING AND SAMPLE
RUN FOR THE LINEAR/QUADRATIC
ALGORITHM

APPENDIX D. EQUATIONS OF MOTION FOR THREE-
DIMENSIONAL FLIGHT IN THE
PRESENCE OF A VERTICAL WIND

APPENDIX E. THE NECESSARY CONDITIONS FOR THE
PROBLEM WITH UNSPECIFIED, BUT EQUAL,
BOUNDARY VALUES
CHAPTER I. INTRODUCTION AND PROBLEM DEVELOPMENT

This research project grew out of an attempt to solve an interesting optimal control problem arising in sailplane flight path optimization. The key features of this problem are that the boundary conditions for the dynamic constraints at the initial and terminal times are fixed and equal, and the dynamic problem exhibits a close relationship to a well known sailplane parameter optimization problem. A more general class of problems, which includes this sailplane problem, will be stated at the close of this chapter.

The usual mode of cross-country flight is known as thermalling. In this mode the pilot seeks out rising air masses called thermals, enters into a circular flight path, and ascends in the thermal. This is assuming, of course, that the upward air velocity in the thermal exceeds the rate of sink of the sailplane in still air. After gaining sufficient altitude the sailplane exits from the thermal and glides down range until another thermal is found.

In practice one doesn't know where the next thermal is, but in this problem a fixed distance between thermals of equal strength is assumed. An experienced pilot can detect
thermals from a good distance away by studying meteorological and topographical conditions. For example, thermals will exist under cumulus clouds, under thunderclouds, and above barren fields on a hot day.

The problem considered here will be to minimize the time of flight from the peak altitude point of one thermal to the point in the next thermal at the same altitude. 

Equilibrium glide trajectories are considered first. An equilibrium glide trajectory is characterized by constant velocity, flight path angle, and lift coefficient. The equilibrium glide condition and the forces acting on the sailplane are depicted in Figure 1.1. The static equations which define an equilibrium glide are given by

\[ L = \frac{1}{2} \rho V^2 C_L S = mg \cos \gamma \tag{1.1} \]

\[ D = \frac{1}{2} \rho V^2 C_D S = -mg \sin \gamma \tag{1.2} \]

where \( L \) and \( D \) represent the aerodynamic forces lift and drag, respectively, and the other variables are defined as follows. The acceleration due to the force of gravity is
represented by \( g \), and \( m \) is the mass of the sailplane. \( S \) is a reference area, \( \gamma \) is the flight path angle of the sailplane, and \( \rho \) is the air density which is assumed to be constant. The usual assumption is made that the drag coefficient, \( C_D \), is a known function of the lift coefficient, \( C_L \).

The angle of attack, \( \alpha \), is the attitude of the wing relative to the incoming airstream. For each \( \alpha \) there is a distinct equilibrium glide trajectory described by the values for \( C_D \) and \( C_L \). The collection of all pairs of \( C_D \) and \( C_L \) values (each for a different equilibrium glide trajectory) defines a "drag polar". A typical drag polar is sketched in Figure 1.2. A drag polar depends on the size, shape, and surface condition of the sailplane, but not its weight.

Assuming that \( C_D \) is a known function of \( C_L \) reduces the number of unknowns in equations (1.1) and (1.2) to three. For a given lift coefficient, \( C_L \), these equations can be used to calculate the corresponding velocity, \( v \), and flight path angle, \( \gamma \), for that particular sailplane and equilibrium glide trajectory. This information can be used to graph a velocity polar as shown in Figure 1.3. The
horizontal component of velocity, $V_x$, and vertical component, $V_d$, form the axes for Figure 1.3. The minimum value for $V_d$, and hence maximum time aloft in still air, is labeled on the graph. The minimum negative flight path angle, and hence maximum range in still air, is also labeled.

Referring to Figure 1.4, the static MacCready problem can now be stated. Which equilibrium glide trajectory should be flown in still air from A to B in order to minimize the total time of flight from A to C? The circular thermalling motion has been approximated by a direct vertical "elevator" ascension at speed $V_T$ from B to C as shown in Figure 1.4. The problem statement can be written as

$$\min \left\{ \frac{X_f}{V_x} + \frac{V_d}{V_x} \cdot \frac{X_f}{V_T} \mid f(V_x, V_d) = 0, \ V_T \ \text{given} \right\}$$

(1.3)

$$= X_f \min \left\{ \frac{1}{V_x} + \frac{V_d}{V_x} \cdot \frac{1}{V_T} \mid f(V_x, V_d) = 0, \ V_T \ \text{given} \right\}$$

where $X_f$ is the range between equal strength thermals, and $V_T$ is the strength of the thermal minus the sailplane rate of sink. Note that the solution is independent of the range between thermals, $X_f$, as shown in equation (1.3).
The solution of this problem yields the MacCready speed. A derivation of this solution is shown in Appendix A.

The name MacCready is attached to this problem and its solution since Paul B. MacCready Jr. pioneered much of the work on it (see references [1] and [2]). In the first of these references, MacCready presents a design for a new sailplane instrument which has come to be known as the MacCready ring. Most sailplanes are equipped with a MacCready ring which allows the pilot to dial in an estimate for $V_T$ and determine the corresponding MacCready speed to be flown during the equilibrium glide between thermals.

A simple graphical interpretation of this solution can now be given. On the velocity polar the value of $V_T$ is plotted on the vertical component of velocity axis, $V_d$. The MacCready speed is then determined by constructing the line of tangency as shown in Figure 1.5. The average horizontal velocity, $V_{AVG}$, is then determined as the $V_x$ intercept of the line of tangency.

The dynamic MacCready problem, which allows trajectories other than equilibrium glides between thermals, will now be stated with reference to Figure 1.6. Determine the lift coefficient (control) time history which minimizes the total time of flight from A to B to C. The performance
index is given by

\[ J = t_f + \frac{[h(0) - h(t_f)]}{V_T} \]  \hspace{1cm} (1.4)
These equations are derived in Appendix D for the more general case of a distributed thermal (vertical) wind profile. Equations (1.5)-(1.8) are the result of setting the wind and its derivative equal to 0 in Appendix D. Four basic assumptions have been made in the above system. They are:

(i) uniform acceleration of gravity - the "flat earth" approximation

(ii) constant density atmosphere

(iii) point mass dynamics (no rotational dynamics)

(iv) planar flight in still air.

This is a free end time problem since $t_f$ is not specified. The problem can be changed to a fixed end time
problem while reducing the order of the system as follows. Note that altitude, \( h \), appears only in the performance index (1.4). Therefore, equation (1.7) may be integrated after determining the solution values for the other state variables. Note also that the range, \( x \), does not appear on the right hand sides of the differential equations (1.5)-(1.8). It is also assumed that range is a monotonically increasing function of time, \( t \). This assumption eliminates maneuvers such as loops in the flight path which would not be expected in a minimum time solution. Therefore, \( x \) will replace \( t \) as the independent variable and further reduce the order of the system. With these changes the following dynamic system is obtained

\[
\begin{align*}
\ddot{V}' &= \left[-\frac{D(V, C_L)}{m \cos \gamma} - g \tan \gamma\right]/\dot{V} \\
\gamma' &= \left[\frac{L(V, C_L)}{m \cos \gamma} - g\right]/\dot{V}^2
\end{align*}
\] (1.9) (1.10)

where the prime denotes a derivative with respect to \( x \).
The definitions in equations (1.1) and (1.2) can be used to determine lift and drag as functions of the lift coefficient and drag coefficient. Next, a parabolic drag polar is assumed so that the drag coefficient is given by

\[ C_D = a_0 + a_1 C_L + a_2 C_L^2 \]  

(1.11)

where the parameters \(a_0, a_1, a_2\) are determined from the velocity polar as obtained from the sailplane manufacturer or flight tests.

The performance index can be calculated as follows.

\[ J = \text{total flight time} = \text{glide time (A to B)} + \text{climb time (B to C)} \]

\[ = \int_0^{t_f} dt + \text{altitude loss/maximum climb rate in thermal} \]

\[ = \int_0^{\hat{h}_f} \frac{d\hat{x}}{\hat{V} \cos \gamma} + \frac{\hat{h}(0) - \hat{h}(t_f)}{\hat{V}_T} \]

\[ = \int_0^{\hat{h}_f} \frac{d\hat{x}}{\hat{V} \cos \gamma} + \left( \frac{-1}{\hat{V}_T} \right) \int_{\hat{h}(0)}^{\hat{h}(t_f)} d\hat{h} \]
\begin{align}
&= \int_0^{\hat{x}_f} \frac{d\hat{x}}{\hat{v} \cos \gamma} - \frac{1}{\hat{v}_T} \int_0^{\hat{x}_f} \tan \gamma \, d\hat{x} \\
&= \int_0^{\hat{x}_f} \frac{\hat{v}_T - \hat{v} \sin \gamma}{\hat{v} \hat{v}_T \cos \gamma} \, d\hat{x} \\
&\quad \text{(1.12)}
\end{align}

The problem is further simplified by using the non-dimensionalizing transformations

\begin{align}
\hat{v} &= \hat{v} / \sqrt{g \hat{x}_f} \\
\tau &= \hat{x} / X_f \\
\eta &= \frac{1}{2} \rho g X_f / (mg/s) = \rho X_f / 2m \\
&\quad \text{(1.13) (1.14) (1.15)}
\end{align}

and by defining the dimensionless parameter

Adopting the usual notation of control theory, the lift coefficient will be designated by $u$. 
After applying the transformations (1.13) and (1.14) and introducing the parameter (1.15), the final non-dimensional problem statement is given by the following.

Find the control \( u(\tau), 0 \leq \tau \leq 1 \), which minimizes

\[
J = \int_{0}^{1} \left[ \frac{v_{T} - V \sin \gamma}{v_{T} \cos \gamma} \right] d\tau
\]

subject to the dynamic constraints

\[
\frac{dv}{d\tau} = - (\eta c_{D} V^{2} + \sin \gamma)/(V \cos \gamma) \quad (1.17)
\]

\[
\frac{d\gamma}{d\tau} = \left( \frac{\eta u v^{2}}{\cos \gamma} - 1 \right)/v^{2} \quad (1.18)
\]

with boundary conditions

\[
v(0) = v(1) = V_{o}
\]

\[
\gamma(0) = \gamma(1) = \gamma_{o}
\]
The relationship between the static sailplane problem stated in equation (1.3) and the dynamic sailplane problem stated in equations (1.16), (1.17), and (1.18) can be generalized as follows. Let the static problem be given by

Problem S: \( \min \{L(x,u) | f(x,u) = 0 \} \)

and let the corresponding dynamic problem be given by

Problem D: \( \min \left\{ \int_0^1 L(x,u) \, d\tau \mid \dot{x} = f(x,u), x(0) = x(1) = x_o \right\} \)

where \( x \) and \( u \) are state and control vectors, respectively. Let \( x_S, u_S, \lambda_S \) be the solution to problem S where \( \lambda_S \) represents the associated Lagrange multiplier. The first-order necessary conditions of optimality for problem D (see Bryson and Ho, [3,p.49]) are

\[
\begin{align*}
\dot{x} &= f(x,u) \\
\dot{\lambda} &= - \frac{\partial T}{\partial x} \lambda - \frac{\partial L}{\partial x} \\
0 &= \frac{\partial L}{\partial u} + \frac{\partial T}{\partial \lambda}, \quad 0 \leq \tau \leq 1
\end{align*}
\]

\( x(0) = x(1) = x_o \)
Note that these conditions are met by setting $x(\tau) = x_S$, $u(\tau) = u_S$, and $\lambda(\tau) = \lambda_S$ providing $x_S = x_o$, since the necessary conditions of optimality for problem S are

$$0 = f(x,u)$$
$$0 = - f^T x - L^T x$$
$$0 = L^T u + f^T \lambda$$

This leads to speculation about the optimal solution to problem D when the boundary conditions do not match the static solution values. It will be seen that the optimal dynamic sailplane trajectories feature a broad mid-range of values close to the static MacCready solution values along with "boundary-layer" transients during which the state of the system is transferred between the static solution values and the boundary values.

Chapter II of this thesis will present several optimal solutions to the dynamic problem for various ranges between thermals which were obtained through the use of a conjugate gradient algorithm and a quasilinearization algorithm. Chapter III will present some suboptimal solutions using two
linearized models of the original problem. One of these models is obtained by solving the linear/quadratic version of the optimal control problem. The other is obtained by solving the linear version of the two-point boundary value which arises in the solution of an optimal control problem. A third suboptimal approximation is attempted through the use of matched asymptotic expansions and singular perturbations. Chapter IV exhibits an extension of the original problem to a more realistic thermal model. Chapter V concludes this thesis while the Appendices list several computer programs and pertinent derivations.
FIGURE 1.1: AN EQUILIBRIUM GLIDE TRAJECTORY

FIGURE 1.2: DRAG POLAR
**FIGURE 1.3: VELOCITY POLAR**

**FIGURE 1.4: STATIC MACCREADY PROBLEM**
FIGURE 1.5: GRAPHIC INTERPRETATION OF THE SOLUTION TO THE STATIC MacCREADY PROBLEM

FIGURE 1.6: DYNAMIC MacCREADY PROBLEM
CHAPTER II. OPTIMAL SOLUTIONS

Section 2.1: Conjugate Gradient Algorithm

The optimal control problem described by equations (1.16) - (1.18) along with the associated boundary conditions has been solved numerically by both conjugate gradient and quasilinearization algorithms. The conjugate gradient algorithm was used to produce solutions to problems with range between thermals, \( X_f \), of 1000, 2000, 5000, and 10000 meters. The quasilinearization algorithm, however, encountered stability problems for \( X_f \) greater than 2000 m.

The conjugate gradient algorithm used here is described by Pierson in Reference [4]. This algorithm is a direct method. That is, the algorithm provides a sequential alteration of the control history. The state equations are integrated forward, and then the costate equations are integrated backward. The gradient is formed by calculating the derivative of the Hamiltonian with respect to the control. An updated control vector is produced using conjugate directions of search, and the procedure is then repeated. The algorithm employs a projection of the gradient vector in order to enforce terminal constraint satis-
faction. The terminal constraints (on the state variables in this problem) must be satisfied at each iteration. A one-dimensional minimization is employed to select the step size parameter. This one-dimensional minimization takes place along the constraint surface using a parabolic interpolation scheme requiring the computation of function values only. An upper bound is placed on the step size in order to prevent an inordinate amount of computational effort in the constraint correction phase of the algorithm which might result from the selection of too large a step size. A constraint is also enforced on the maximum allowable change in a control element. The algorithm starts with negative projected gradient directions of search until an unconstrained iteration (one-dimensional minimization) takes places. Thereafter, unless a constrained iteration occurs, a cycle of one negative projected gradient direction of search step followed by some specified number of projected conjugate gradient direction of search steps proceeds until the solution is achieved.

The solution of the static MacCready problem is used as an initial guess for the optimal control to the dynamic problem with \( X_f = 1000 \) meters. The solution to the
$X_f = 1000\text{m}$ problem is then "lengthened" and used as an input to the $X_f = 2000\text{m}$ problem. The control history is lengthened by adding 1000 meters of constant control values to the middle of the $X_f = 1000\text{ meter}$ solution. The constant value selected is the solution value at $x = 500\text{ meters}$ (the midpoint of the 1000 meter solution). This procedure is continued as range increases in order to develop nominal starting values for the control history.

The numerical results presented here have been computed on an IBM 360-65 computer using the Fortran language. All calculations are done with double precision arithmetic, and fourth order Runge Kutta is the integration technique employed. The integration step size is selected so that there is one step for each 10 meters of range. The values for the constants in the quadratic drag polar (Equ. 1.11) are: $a_0 = 0.009278$, $a_1 = -0.009652$, and $a_2 = 0.022288$. These values vary with the configuration of the particular sailplane being used. The NIMBUS II, an open class sailplane, is the source for these values as well as for the value of the aerodynamic parameter $\eta$ (Equ. 1.15): $\eta = 9.81 \frac{x_f}{512.0}$. The termination criteria used is imposed on the change in the control history from one
iteration to the next. The algorithm terminates when the sum of the squares of the control changes is less than $0.5 \times 10^{-3}$. The terminal state constraints are

$$
\psi[V(1), \gamma(1)] = \begin{bmatrix}
V(1) - V_o \\
\gamma(1) - \gamma_o
\end{bmatrix} = 0
$$

and they are considered to have been satisfied if $\|\psi\| < \epsilon$. A value of $0.1 \times 10^{-4}$ was selected for $\epsilon$. The upper limit on the allowed amount of change in the control, $C_L$, per iteration is 0.1.

Figures 2.1, 2.2, 2.3, and 2.4 show the optimal trajectories for the 1000, 2000, 5000, and 10000 meter problems, respectively. Figures 2.5, 2.6, 2.7, and 2.8 indicate the optimal control histories for these same four problems. These figures demonstrate that the solution can be thought of in three separate pieces. First, the state of the sailplane changes rapidly from the prescribed boundary condition values to the solution values for the static MacCready problem. (The MacCready solution values are derived in Appendix A.) The state of the system then remains close to the static MacCready state until another
period of rapid change returns the state of the sailplane to the prescribed boundary values.

Both sets of figures further indicate that these periods of rapid change cover a length of about 300 meters, and this length remains approximately constant as the range between thermals increases. The lengthening of the range serves primarily to lengthen the distance traveled in the static MacCready state. The flight time along the optimal trajectory is compared to the flight time along a constant static MacCready flight path in Table 2.1. This again serves to indicate that the lengthening of range only increases the length of the static MacCready portion of the optimal path. The time "lost" on the optimal trajectory is the amount of extra time necessary to move the sailplane state from the prescribed boundary values to the MacCready state.

These optimal solutions indicate an almost symmetric optimal control as shown in Figures (2.5) – (2.8). A numerical check of the solution values indicates that the drag coefficient, $C_D$, is less than 0.012 for all $\tau$, $0 \leq \tau \leq 1$. Furthermore, the prescribed boundary conditions on the flight path angle are $\gamma(0) = \gamma(1) = -0.0209626$
23

radians. If these two already small values are set equal to zero, then equations (1.17) and (1.18) become,

\[ \frac{dv}{d\tau} = -\sin \gamma/(V \cos \gamma) \] (2.1)

and

\[ \frac{d\gamma}{d\tau} = \eta u/\cos \gamma - 1/V^2 \] (2.2)

The boundary conditions become

\[ V(0) = V(1) = V_0 \] (2.3)
\[ \gamma(0) = \gamma(1) = 0 \]

This leads to the following theorem.

**Theorem 2.1.**

Given a symmetric control, \( u(\tau) \), (that is \( u(\tau) = u(1-\tau) \) for all \( \tau, 0 \leq \tau \leq 1 \)) the system (2.1) and (2.2) along with the boundary conditions (2.3) possesses a solution \( (V(\tau), \gamma(\tau)) \) for \( 0 \leq \tau \leq 1 \) which exhibits the following type of symmetry;
\[ V(\tau) = V(1-\tau) \]
\[ \gamma(\tau) = - \gamma(1-\tau) \]  \hspace{1cm} (2.4)

Proof: Let \( W(\tau) = V(1-\tau) \)
\[ G(\tau) = - \gamma(1-\tau) \]

where \( V(\tau) \) and \( \gamma(\tau) \) are the unique solutions for (2.1) - (2.3) for the given control \( u(\tau) \). Then \( W(0) = W(1) = V_o \) and \( G(0) = G(1) = 0 \). That is, the "reverse time" variables \( W \) and \( G \) satisfy the same boundary conditions as the "forward time" variables \( V \) and \( \gamma \). Differentiating with respect to \( \tau \), it is seen that

\[ \frac{dW(\tau)}{d\tau} = - \frac{dv(1-\tau)}{d\tau} = - \frac{\sin(-G(\tau))}{W(\tau)\cos(-G(\tau))} \]  \hspace{1cm} (2.5)

\[ \frac{dG(\tau)}{d\tau} = \frac{d\gamma(1-\tau)}{d\tau} = \eta u(\tau)/\cos(-G(\tau)) - 1/W^2(\tau) \]  \hspace{1cm} (2.6)

or more simply
\[
\frac{dw}{d\tau} = -\sin G/(W \cos G) \quad (2.7)
\]

\[
\frac{dg}{d\tau} = \eta \frac{u}{\cos G} - 1/W^2 \quad (2.8)
\]

where all variables are understood to be evaluated at \( \tau \).

Since \( W \) and \( G \) satisfy the same differential equations and boundary conditions as \( V \) and \( \gamma \) do, the theorem is proved.

It must be remembered that this theorem does not prove that the optimal solution is symmetric. It only shows that given a symmetric control and assuming uniqueness of solution for the approximating equations (2.1) - (2.3), then the variables \( V \) and \( \gamma \) must also be symmetric. To show that the optimal solution is symmetric, the entire two-point boundary value problem must be employed in the proof. Unfortunately, with \( C_D = 0 \) the optimal control problem becomes singular as the following derivation will show.
Section 2.2: Quasilinearization Algorithm

The quasilinearization algorithm used (see, for example, Shipman and Roberts [5]) attempts to solve the nonlinear two-point boundary value problem encountered in the solution of an optimal control problem by solving a sequence of linearized versions of this problem. The two-point boundary value problem is determined by the state and costate dynamics along with the prescribed boundary conditions (see [6]). The state dynamics are given in equations (1.17) and (1.18). The costate dynamics are developed in the following manner. First define the following quantities.

\[ L(V, \gamma) = \frac{(V_T - V \sin \gamma)}{(V \cos \gamma)} \]  \hspace{1cm} (2.9)

\[ f(V, \gamma, u) = \begin{bmatrix} -\left(\eta C_D V^2 + \sin \gamma\right)/(V \cos \gamma) \\ \eta u/\cos \gamma - 1/V^2 \end{bmatrix} \]  \hspace{1cm} (2.10)

\[ \lambda(\tau) = \begin{bmatrix} \lambda_V(\tau) \\ \lambda_Y(\tau) \end{bmatrix} \]  \hspace{1cm} (2.11)
\( \lambda(\tau) \) is the costate vector composed of the Lagrange multipliers associated with the states \( V \) and \( \gamma \). Then the Hamiltonian, \( H \), is defined by

\[
H(V,\gamma,u,\lambda) = L(V,\gamma) + \lambda^T f(V,\gamma,u) \tag{2.12}
\]

The costate dynamics are then given by

\[
\dot{\lambda}_V = - \frac{\partial H}{\partial V} \tag{2.13}
\]

\[
\dot{\lambda}_\gamma = - \frac{\partial H}{\partial \gamma} \tag{2.14}
\]

To implement the quasilinearization algorithm, the optimality condition

\[
0 = \frac{\partial H}{\partial u} = \lambda^T f_u = - \frac{\eta V \lambda_v}{\cos \gamma} \frac{dC}{du} + \frac{\eta \lambda_\gamma}{\cos \gamma} \tag{2.15}
\]
must be used to remove $u$ from the state (equations (1.17) and (1.18)) and costate (equations (2.13) and (2.14))
equations. This is not always possible although it is in this problem since equation (2.15) is linear in $u$ (quad­
ratic drag polar (1-11) assumed). Notice that if $C_D = 0$,
equation (2.15) could not be solved for $u$. Once equation (2.15) is used to remove $u$, equations (1.17), (1.18), (2.13) and (2.14) along with the prescribed boundary con­
ditions become the nonlinear two-point boundary value problem as stated below.

\[
\dot{V} = -\eta C_D \frac{V}{\cos \gamma} - \frac{\sin \gamma}{V \cos \gamma} \quad (2.16)
\]

\[
\dot{\gamma} = \eta \frac{u}{\cos \gamma} - \frac{1}{V^2} \quad (2.17)
\]

\[
\lambda_V = \frac{1}{V^2 \cos \gamma} - \lambda_V \left( -\eta \frac{C_D}{\cos \gamma} + \frac{\sin \gamma}{V^2 \cos \gamma} \right) - \lambda_V \cdot \frac{2}{V^3} \quad (2.18)
\]
\[ \dot{\lambda}_V = \frac{(V - V_T \sin \gamma)}{(V_T V \cos^2 \gamma)} \]

\[ - \lambda_V \left(- \eta C_D \frac{V \sin \gamma}{\cos^2 \gamma} - \frac{1}{V \cos^2 \gamma}\right) \]  

\[ - \lambda_\gamma \left(\eta u \frac{\sin \gamma}{\cos^2 \gamma}\right) \]

\[ V(0) = V(1) = V_0 \]  

\[ \gamma(0) = \gamma(1) = \gamma_0 \]  

where \[ u = \lambda_{\gamma}/(2a_2 V \lambda_V) - a_1/(2a_2) \]

Note that (2.21) is the solution of equation (2.15).

The quasilinearization algorithm, in general, does not work well on sensitive problems (see [7]) because of the error in the guessed missing initial values for \( \lambda_V \) and \( \lambda_\gamma \). The method proceeds by choosing nominal profiles for the variables \( V, \gamma, \lambda_V, \) and \( \lambda_\gamma \) and linearizing equations (2.16) - (2.19) about these nominals. The resulting linear system is solved by guessing values for \( \lambda_V(0) \)
and \( \lambda^V(0) \) and integrating the equations (2.16) - (2.19) forward to \( \tau = 1 \). At this time a correction in the guesses for \( \lambda^V(0) \) and \( \lambda^Y(0) \) is made based on the amount of miss of the terminal state constraints. In this thesis the adjoint method (see [5]) is used to update the values \( \lambda^V(0) \) and \( \lambda^Y(0) \). When this two-point boundary value problem is solved (terminal state constraints accurate to within a given tolerance, \( 0.5 \times 10^{-7} \)) the solution is used as the current nominal, and the method is repeated. After each iteration, the sum of the squares of the changes in the state variables for the entire \( \tau \)-interval (an approximation of the \( L_2 \) norm) is evaluated, and the algorithm terminates when this value changes less than \( 0.5 \times 10^{-8} \).

The program itself is listed in Appendix B.

This problem lends itself to a procedure to calculate nominal values for \( \lambda^V(0) \) and \( \lambda^Y(0) \). This procedure allows a partial avoidance of the sensitivity of the problem to inaccuracies in these nominal values. Since the Hamiltonian, \( H \), (equation (2.12)) is autonomous (no explicit dependence on the independent variable, \( \tau \)) it will be constant along the optimal trajectory (see [8], p.254). The optimal trajectory is expected to contain a broad
midrange of solution values which approximate the static MacCready values. Therefore, the static MacCready values can be substituted into equation (2.12) to get an approximate value for $H$. An approximate value for $u(0)$, which turns out to be $-0.12065$, can also be calculated using equations (1.1) and (1.2) along with the prescribed boundary conditions for $V(0)$ and $\gamma(0)$ since these boundary conditions correspond to an equilibrium glide trajectory (the one which describes the minimum rate of sink). Finally, substituting the values approximated for $H$ and $u(0)$ along with the boundary conditions, $V(0)$ and $\gamma(0)$, into equations (2.12) and (2.15), nominal values for $\lambda_V(0)$ and $\lambda_\gamma(0)$ can be calculated.

For $x_f = 1000\text{m}$ and $V_T = 2\text{m/s}$, the values calculated as described are $\lambda_V(0) = -12.859$ and $\lambda_\gamma(0) = 0.047$. After convergence of the quasilinearization algorithm for the 1000 meter problem, the solution values are $\lambda_V(0) = -12.857$ and $\lambda_\gamma(0) = 0.048$. This procedure's ability to provide nominal values close to the solution values enabled the algorithm to converge on a problem whose sensitivity ordinarily might preclude convergence.
The solution trajectories for the \( X_f = 1000 \) meter problem are lengthened as for the conjugate gradient algorithm and used as inputs for the \( X_f = 2000 \) meter problem. The 1000 meter problem was started with constant nominal trajectories equal to the static MacCready values. This method of lengthening the 1000 meter solution essentially treats the extension of range between thermals as a continuation problem (see [5]). Since this is a sensitive problem, the continuation method, as might be expected, eventually fails. In this case convergence was not achieved for \( X_f > 2000 \) meters. A measure of this sensitivity is shown in Table 2.2. As was previously mentioned, the adjoint method was used to correct the initial costate values. In order to explain the adjoint method the following definitions are made.

\[
Z^T = [V, \gamma, \lambda_V, \lambda_\gamma] \quad (2.22)
\]
\[ F = \begin{bmatrix}
- \eta C_D V \cos \gamma - \sin \gamma / (V \cos \gamma) \\
\eta u / \cos \gamma - 1 / \sqrt{V^2} \\
1 / (V^2 \cos \gamma) - \lambda V (-\eta C_D / \cos \gamma + \sin \gamma / (V^2 \cos \gamma)) \\
- 2 \lambda \gamma / V^3 \\
(V - V_T \sin \gamma) / (V_T V \cos^2 \gamma) \\
- \lambda V (-\eta C_D V \sin \gamma / \cos^2 \gamma - 1 / (V \cos^2 \gamma)) \\
- \lambda \gamma (\eta u \sin \gamma / \cos^2 \gamma)
\end{bmatrix} \quad (2.23) \]

where \( u \) is defined by equation (2.21). The two-point boundary value problem can now be written as

\[ \dot{Z} = F(Z) \quad (2.24) \]

with boundary conditions \( Z_1(0) = Z_1(1) = V_0 \)

and \( Z_2(0) = Z_2(1) = \gamma_0 \)
The quasilinearization algorithm begins by linearizing \( F(Z) \) about the MacCready values to obtain the variational equation

\[
\delta Z = F_Z \delta Z \tag{2.25}
\]

where \( \delta Z \) represents the change in the vector \( Z \).

The adjoint correction equations are

\[
\dot{\Lambda} = - F_Z^T \Lambda \tag{2.26}
\]

where \( \Lambda \) represents the adjoint variables. The correction procedure requires two backward integrations of this equation with the initial values set first to 
\( \Lambda^T(l) = [1,0,0,0] \) and then \( \Lambda^T(l) = [0,1,0,0] \). In each case the initial values \( \Lambda(0) \) are saved in order to calculate \( \delta Z_3(0) \) and \( \delta Z_4(0) \), the correction in the initial costates. This calculation proceeds by noticing that

\[
(\delta Z^T\Lambda) = \delta Z^T\Lambda + \delta Z^T\dot{\Lambda} = (F_Z\delta Z)^T\Lambda + \delta Z^T(-F_Z^T\Lambda) = 0.
\]
Then integration yields

\[ \int_{0}^{1} (\delta Z^T \Lambda) d\tau = 0 \]

or

\[ \delta Z^T \Lambda \bigg|_{0}^{1} = 0 \quad (2.27) \]

After integrating the adjoint equations twice equation (2.27) can be used to solve for \( \delta Z(0) \), the corrections to the initial costates. That is

\[ \delta Z_3(1) = \Lambda_3^1(0) \delta Z_3(0) + \Lambda_4^1(0) \delta Z_4(0) \quad (2.28) \]

\[ \delta Z_4(1) = \Lambda_3^2(0) \delta Z_3(0) + \Lambda_4^2(0) \delta Z_4(0) \quad (2.29) \]

where \( \Lambda_3^1(0) \), for example, is the third element of the adjoint variable vector found in the first backward integration when evaluated at zero. Equations (2.28) and (2.29) may be written as
Then define

\[
\begin{bmatrix}
\delta Z_3(1) \\
\delta Z_4(1)
\end{bmatrix} =
\begin{bmatrix}
\Lambda_3^1(0) & \Lambda_4^1(0) \\
\Lambda_3^2(0) & \Lambda_4^2(0)
\end{bmatrix}
\begin{bmatrix}
\delta Z_3(0) \\
\delta Z_4(0)
\end{bmatrix} \quad (2.30)
\]

The corrections to the initial costates can then be calculated as

\[
P =
\begin{bmatrix}
\Lambda_3^1(0) & \Lambda_4^1(0) \\
\Lambda_3^2(0) & \Lambda_4^2(0)
\end{bmatrix} \quad (2.31)
\]

The condition number (see [9]) of \( P \) is displayed in Table 2.2. The condition number, \( C \), is computed as follows.

If \( A \) is the matrix whose condition number is to be computed, then the condition number is the ratio of the largest to the smallest eigenvalue of \( A \). For a \( 2 \times 2 \) matrix,
define \( T = \text{trace of } A, \quad D = \text{determinant of } A, \) and
\[
S = \sqrt{T^2 - 4D}.
\]
Then \( C = (T + S)/(T - S). \)

Also shown in Table 2.2 is the magnitude of the eigenvalues of the system matrix encountered when the original two-point boundary value problem, equations (2.16) - (2.20), is linearized about the MacCready values. This system matrix is found by calculating the derivative of the matrix, \( F \), found in equation (2.23) with respect to \( Z \) and evaluating this derivative on the MacCready trajectory. Define the result to be \( \frac{\partial F}{\partial Z} \equiv F_Z \). The increasing magnitude of the eigenvalues of this matrix indicate an increasing sensitivity of the optimal control problem. All four of the eigenvalues of \( F_Z \) have the same magnitude because they are complex and because of the following theorem which is found in reference [10] (see also [11]).

**Theorem 2.2.**

Given a matrix, \( C \), which has the block form
\[
C = \begin{bmatrix} F & A \\ B & -F_T \end{bmatrix},
\]
where each block is \( n \times n \) in dimension,
then if \( \mu \) is an eigenvalue of \( C \), so is \(-\mu\).
Proof: Define $S = \begin{bmatrix} -B^T & F \\ F & A \end{bmatrix}$ and $J = \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix}$ where $I_n$ denotes an identity matrix which is $n \times n$ in dimension.

Then notice that $JS = C$, $\det|J| = (-1)^n$ and

$$(\mu I_{2n} - JS)^T = (\mu I_{2n} + SJ) = J(-\mu I_{2n} - JS)J.$$ Denoting the characteristic polynomial of $C$ as $P$, one completes the proof by noting that

$$P(\mu) = \det|\mu I_{2n} - C| = \det|\mu I_{2n} - JS| = \det|(\mu I_{2n} - JS)^T| =$$

$$\det|\mu I_{2n} + SJ| = \det|J(-\mu I_{2n} - JS)J| = \det|-\mu I_{2n} - JS| = P(-\mu).$$

If $C$ in the above theorem has all real entries, then if $\mu$ is complex, the conjugates of $\pm \mu$ are also eigenvalues of $C$. A Hamiltonian system has the form

$$\dot{x} = H_\lambda$$

$$\dot{\lambda} = -H_x$$

where $x$ and $\lambda$ are the state and costate vectors,
respectively, $H$ is the Hamiltonian as in equation (2.12), and the subscripts denote partial differentiation. If this system is linearized about an equilibrium point (in this problem, the static MacCready values), one obtains a system matrix with the form

$$
\begin{bmatrix}
H_{\lambda x} & H_{\lambda \lambda} \\
-\bar{H}_{xx} & -\bar{H}_{x\lambda}
\end{bmatrix}
$$

where all partials are evaluated at the equilibrium point. If $H$ possesses sufficient smoothness (for example, one continuous derivatives), then $H_{\lambda x} = H_{x\lambda}$ and $H_{x\lambda}$ is symmetric. Therefore, this matrix meets the hypotheses of Theorem 2.2 and has eigenvalues in groups of four. Some of these groups of four may degenerate if an eigenvalue is real or equal to zero.

The quasilinearization solutions which were computed for $X_f = 1000\text{m}$ and $2000\text{m}$ showed good agreement with the conjugate gradient solutions as is shown in Table 2.3. This table shows the agreement in both the performance index and the optimal trajectories.

The major advantage of the quasilinearization algorithm is the relatively low cost. For example, the 1000 meter solution required only 6.71 seconds of CPU time,
while the corresponding conjugate gradient solution re-
quired 70.34 seconds of CPU time. The major disad-
vantage is the sensitivity to the initial costate values and
corresponding inability to produce solutions for
$X_f > 2000m$.

Section 2.3: Sufficiency Conditions

This chapter concludes with an indication of the
validity of these optimal solutions. Sufficiency conditions
for a broad class of optimal control problems are developed
in [3, pp.179-181] by Bryson and Ho as follows. Define the
following matrices:

\[
A = f_{x} - f_{x} H^{-1} H_{x}
\]
\[
B = f_{u} H^{-1} H_{u}
\]
\[
C = H_{xx} - H_{xx} H^{-1} H_{x}
\]

where the subscript denotes partial differentiation. Then
compute the matrices $S$, $Q$, and $R$ as solutions to the
differential equations
\[ \dot{S} = -SA - AT S + SBS - C \]
\[ \dot{R} = - (AT - SB) R \]
\[ \dot{Q} = R^T BR \]

subject to the boundary conditions (for this particular problem)

\[ S(l) = 0 \]
\[ R(l) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \]
\[ Q(l) = 0 \]

The dot (\( \cdot \)) means differentiation with respect to the independent variable, \( \tau \).

Then the sufficiency conditions are

\[ H_{uu}(\tau) > 0, \ 0 \leq \tau \leq l \]  \hspace{2cm} (2.33)

\[ Q(\tau) < 0, \ 0 \leq \tau \leq l \]  \hspace{2cm} (2.34)
\[ S(\tau) - R(\tau) Q^{-1}(\tau) R^T(\tau) \text{ finite for } 0 \leq \tau < 1 \] (2.35)

These conditions are known in optimal control theory as the convexity condition or strengthened Legendre-Clebsch condition, the normality condition, and the Jacobi condition (no conjugate points on the optimal path), respectively. These conditions were checked numerically using the solution trajectories from the conjugate gradient solution of the 1000 meter problem with \( v_T = 2 \text{m/s} \). These calculations, as depicted in Figures 2.9, 2.10, and 2.11 demonstrate that the sufficiency criteria for optimality have been fulfilled.

The sufficiency conditions were also checked on the static MacCready values (Appendix A) for \( v_T = 2 \text{m/s} \). The results are depicted in Figures 2.12 and 2.13. The following theorem then relates the value of the performance index (equation (1.16)) when evaluated on the optimal trajectory to the value of this performance index when evaluated on the static MacCready values. The Legendre-Clebsch condition is also met on the MacCready trajectory as can be seen analytically as follows. By setting equations (2.18) and (2.19) equal to zero and rearranging terms, one obtains
\[
\frac{1}{V^2 \cos \gamma} = \lambda_V ((-\eta C_D V^2 + \sin \gamma)/(V^2 \cos \gamma)) + \lambda_\gamma (2/V^3) \quad (2.36)
\]

\[
\frac{V - V_T \sin \gamma}{V_T V \cos^2 \gamma} = \lambda_V ((-\eta C_D V^2 \sin \gamma - 1)/(V \cos^2 \gamma)) + \lambda_\gamma (\eta u \sin \gamma / \cos^2 \gamma) \quad (2.37)
\]

Since \(-\eta C_D V^2 = \sin \gamma \) (equation A.1), velocity is positive, flight path angle is between 0° and -90°, and drag coefficient is positive on an equilibrium glide trajectory in still air the coefficients on the right hand sides of equations (2.36) and (2.37) are all negative except for \(2/V^3\), which is positive. For the same reasons the left hand sides are positive. It can be seen, therefore, that \(\lambda_V\) must be negative on the MacCready trajectory.

Returning to the optimality condition (equation (2.15)), one can calculate

\[
\frac{d^2 H}{du^2} = H_{uu} = -\frac{d^2 C_D}{du^2} \eta V \lambda_V \cos \gamma = -2a_2 \eta V \lambda_V \cos \gamma \quad (2.38)
\]
In view of the fact that $a_2$ is positive (see Appendix A) and the above analysis, one sees that the Legendre-Clebsch condition, $H_{uu} > 0$, is met on the MacCready trajectory.

**Theorem 2.3.**

Define an optimal control problem,

$$D: \min_{t_0} J = \int_{t_0}^{t_f} L(x,u) \, dt \tag{2.39}$$

subject to \[ \dot{x} = f(x,u) \]

\[ x(t_0) = x(t_f) = \bar{x} \text{ (specified)} \tag{2.40} \]

which has a unique solution, \((x_D,u_D,\lambda_D)\), and assume $f(x,u)$ has an equilibrium point, \((x_S,u_S,\lambda_S)\), which is the solution to the parameter optimization problem

$$\min \ L(x,u)$$

$$\text{subject to} \quad 0 = f(x,u).$$

Further assume that \((x_S,u_S,\lambda_S)\) satisfies the sufficiency conditions for optimality for problem D. Then
Clearly, equality holds in (2.41) if $\bar{x} = x_s$.

Proof: First, note that if $H = L + \lambda^T f$ is the Hamiltonian of Problem D, than the necessary conditions for optimality are

\begin{align}
\dot{x} &= H_x = f \\
\dot{\lambda} &= -H_x^T \\
x(t_o) &= x(t_f) = \bar{x}
\end{align}

Next, define a new optimal control problem

\begin{align}
\bar{D}: \min \mathcal{J} &= \int_{t_o}^{t_f} L(x,u) dt \\
\dot{x} &= f(x,u) \\
x(t_o) &= x(t_f) \text{ (unspecified)}
\end{align}
The necessary conditions are derived in references [12, p.344] and [13] and include equations (2.42), (2.43), and (2.45) along with the conditions that $x(t_o) = x(t_f)$ and $\lambda(t_o) = \lambda(t_f)$. A demonstration of how to apply the theorem in reference [12] is given in Appendix E. Therefore, the necessary conditions for this problem are satisfied by $(x_s, u_s, \lambda_s)$. Since $(x_s, u_s, \lambda_s)$ is the solution to problem $\tilde{D}$ which admits a wider choice of possible solutions than does problem $D$, the result $J(x_s, u_s, \lambda_s) \leq J(x_D, u_D, \lambda_D)$ is obtained.
Table 2.1. Time of Flight Comparison

<table>
<thead>
<tr>
<th>$x_f$ (m)</th>
<th>Conjugate Gradient Time of Flight (Sec)</th>
<th>Static MacCready Time of Flight (Sec)</th>
<th>Difference Between Times of Flight</th>
<th>Percentage Difference $\frac{Col<del>2-Col</del>3}{Col~2} \times 100$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>40.382</td>
<td>38.203</td>
<td>2.179</td>
<td>5.704</td>
</tr>
<tr>
<td>2000</td>
<td>78.583</td>
<td>76.406</td>
<td>2.177</td>
<td>2.849</td>
</tr>
<tr>
<td>5000</td>
<td>193.186</td>
<td>191.015</td>
<td>2.171</td>
<td>1.137</td>
</tr>
<tr>
<td>10000</td>
<td>384.190</td>
<td>382.030</td>
<td>2.160</td>
<td>0.565</td>
</tr>
</tbody>
</table>

Table 2.2. Sensitivity of the Optimal Control Problem When Solved by Quasilinearization

<table>
<thead>
<tr>
<th>$x_f$ (m)</th>
<th>Condition Number of P</th>
<th>Magnitude of the Eigenvalues of $F_Z$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>26.460</td>
<td>11.305</td>
</tr>
<tr>
<td>2000</td>
<td>25.769</td>
<td>22.613</td>
</tr>
<tr>
<td>3000</td>
<td>84.116</td>
<td>33.916</td>
</tr>
</tbody>
</table>
Table 2.3. Consistency of Quasilinearization and Conjugate Gradient Solutions

<table>
<thead>
<tr>
<th>Performance Index (Sec)</th>
<th>Max Flight Velocity (m/Sec)</th>
<th>Max Flight Path Angle (Rad)</th>
<th>Min Flight Path Angle (Rad)</th>
<th>Max C_L</th>
</tr>
</thead>
<tbody>
<tr>
<td>QL 1000</td>
<td>40.399</td>
<td>41.556</td>
<td>0.369</td>
<td>-0.405</td>
</tr>
<tr>
<td>CGI 1000</td>
<td>40.382</td>
<td>41.579</td>
<td>0.371</td>
<td>-0.407</td>
</tr>
<tr>
<td>QL 2000</td>
<td>78.600</td>
<td>41.491</td>
<td>0.369</td>
<td>-0.405</td>
</tr>
<tr>
<td>CGI 2000</td>
<td>78.583</td>
<td>41.489</td>
<td>0.370</td>
<td>-0.407</td>
</tr>
</tbody>
</table>

QL = Quasilinearization Method

CGI = Conjugate Gradient Algorithm
FIGURE 2.1: ALTITUDE LOSS
TRAJECTORY $X_f = 1000\text{m}, V_f = 2\text{m/s}$

FIGURE 2.2: ALTITUDE LOSS
TRAJECTORY $X_f = 2000\text{m}, V_f = 2\text{m/s}$
FIGURE 2.3: ALTITUDE LOSS
TRAJECTORY $X_f = 5000\, \text{m}$, $V_f = 2\, \text{m/s}$

FIGURE 2.4: ALTITUDE LOSS
TRAJECTORY $X_f = 10000\, \text{m}$, $V_f = 2\, \text{m/s}$
FIGURE 2.5: CONTROL HISTORY $X_f = 1000\, m$, $V_T = 2\, m/s$

FIGURE 2.6: CONTROL HISTORY $X_f = 2000\, m$, $V_T = 2\, m/s$
Figure 2.7: Control History \( X_f = 5000 \text{m} \), \( V_f = 2 \text{m/s} \)

Figure 2.8: Control History \( X_f = 10000 \text{m} \), \( V_f = 2 \text{m/s} \)
Figure 2.9: Legendre-Clebsch Condition

Figure 2.10: Normality Condition via Sylvester's Criterion
FIGURE 2.11: JACOBI CONDITION
DETERMINANT NONDIMENSIONAL RANGE, $X/X_f$

FIGURE 2.12: THE NORMALITY CONDITION ON THE MAC CREADY TRAJECTORY (EQUATION (2.29))

DETERMINANT OF $Q$

NONDIMENSIONAL RANGE, $\tilde{X}/X_f$

FIGURE 2.13: THE JACOBI CONDITION ON THE MAC CREADY TRAJECTORY (EQUATION (2.30))
CHAPTER III. SUBOPTIMAL SOLUTIONS

Section 3.1. Linearized Two-Point Boundary Value Problem

Since the optimal solutions discussed in Chapter II become expensive to produce as range increases, and since a reasonable idea of the shape of the solution can be "guessed" by using Theorem 2.3, it is natural to investigate the possibility of suboptimal solutions. The first suboptimal solution to be discussed will be a shortened version of the quasilinearization algorithm described in Chapter II. In this method, the original nonlinear two-point boundary value problem, (2.16) - (2.20), is linearized about the MacCready values, and then this linear two-point boundary value problem is solved. The resulting solution is suboptimal in the sense that the necessary conditions (nonlinear two-point boundary value problem) are only approximately satisfied.

Solutions have been computed for the $X_f = 1000$ and 2000 meters problems. Sensitivity problems, as previously demonstrated in Table 2.2, preclude solutions for greater ranges. Certain cost and performance factors for the optimal and suboptimal quasilinearization algorithms are
compared in Table 3.1. The cost is measured in terms of CPU time, while the time of flight values are also listed to compare the accuracy of the two algorithms. This table indicates a substantial savings in CPU time at the expense of a small error in the time of flight.

A comparison of the solutions indicates that the major disadvantage of the suboptimal algorithm arises in the lack of good prediction for optimal altitude loss. The maximum altitude loss in the suboptimal solution for the \( x_f = 1000 \) meters problem was 91.420 meters as compared to 76.856 meters in the optimal solution. This represents an increase of 18.95%. The complete trajectories for the \( x_f = 1000 \) meters problem are shown in Figure 3.1 for \( v_T = 2 \text{m/s} \).

Section 3.2. Linear/Quadratic Version

The next suboptimal algorithm is based on a linear/quadratic version of the original optimal control problem. This differs from the quasilinearization algorithm in the following way. In the linear/quadratic version the dynamics, equations (1.17) and (1.18), are linearized about the static MacCready trajectory before the optimality
condition, equation (2.15), is applied to remove the control from the two-point boundary value problem. Also, the performance index, equation (1.16), is expanded to second order about the MacCready values. This proceeds as follows.

Using the notation established in equations (2.9), (2.10), and (2.11), equations (1.16), (1.17), and (1.18) can be rewritten as

\[ \min J = \int_{0}^{1} L(V, \gamma) d\tau \]  \hspace{1cm} (3.1)

\[ \dot{\begin{bmatrix} V \\ \gamma \end{bmatrix}} = \begin{bmatrix} f(V, \gamma, u) \end{bmatrix} \]  \hspace{1cm} (3.2)

with the boundary conditions on \( V \) and \( \gamma \) remaining as stated previously.

Denoting partial differentiation by a subscript the expansions of (3.1) and (3.2) yield

\[ \delta^2 J = \int_{0}^{1} \left\{ L_V \delta V + L_\gamma \delta \gamma + \frac{1}{2} \begin{bmatrix} \delta V \\ \delta \gamma \end{bmatrix} \begin{bmatrix} L_{VV} & L_{V\gamma} \\ L_{\gamma V} & L_{\gamma\gamma} \end{bmatrix} \begin{bmatrix} \delta V \\ \delta \gamma \end{bmatrix} \right\} d\tau \]  \hspace{1cm} (3.3)
(which is the second variation of the performance index) and the linearized dynamics

\[
\begin{bmatrix}
\delta \dot{V} \\
\delta \gamma
\end{bmatrix} = [f_V, f_\gamma] \begin{bmatrix}
\delta V \\
\delta \gamma
\end{bmatrix} + [f_u] \delta u
\]  

(3.4)

The boundary values are specified as the difference between the original boundary values and the MacCready values. All the partial derivatives in equations (3.3) and (3.4) are evaluated at the MacCready values for \( V, \gamma, \) and \( u. \)

For convenience the following quantities are defined.

\[ A^T = [L_V, L_\gamma] \]  

(3.5)

\[ C = \begin{bmatrix}
L_{VV} & L_{V\gamma} \\
L_{V\gamma} & L_{\gamma\gamma}
\end{bmatrix} \]  

(3.6)

\[ D = [f_V, f_\gamma] \]  

(3.7)

\[ E = [f_u] \]  

(3.8)

\[ X = \begin{bmatrix}
\delta V \\
\delta \gamma
\end{bmatrix} \]  

(3.9)
\( \gamma = \delta u \quad (3.10) \)

The specified boundary values are given by \( \hat{x} = x(0) = x(1) \).

With this notation the new approximate optimal control problem becomes

\[
\begin{align*}
\min J &= \int_{0}^{1} \left( A^T x + \frac{1}{2} x^T C x \right) dt \\
\text{subject to} & \quad \dot{x} = D x + E y \\
& \quad x(0) = x(1) = \hat{x}
\end{align*}
\]

(3.11, 3.12)

This is the linear/quadratic version of the optimal control problem stated in equations (3.1) and (3.2). The Hamiltonian for this new problem is

\[
H = A^T x + \frac{1}{2} x^T C x + \lambda^T (D x + E y) \quad (3.13)
\]

where \( \lambda \), the costate variables, satisfy
This problem is known as a singular arc problem (see [14]) due to the fact that the control, \( y \), appears linearly in the Hamiltonian. This linear dependence forces a change in the solution procedure. The optimality condition

\[
\frac{\partial H}{\partial y} = E^T \lambda = 0, \quad 0 < \tau < 1
\]  

(3.15)

can not be used to eliminate the control, \( y \), from the two-point boundary value problem stated in equations (3.12) and (3.14). However, since \( \frac{\partial H}{\partial y} = 0 \) along the singular arc, the derivative of this expression with respect to \( \tau \) must also equal zero.

\[
\frac{d}{d\tau} \frac{\partial H}{\partial y} = E^T \dot{\lambda} = E^T (-A - Cx - D^T \lambda) = 0
\]  

(3.16)

This equation still does not yield any information about the control, \( y \). The procedure is repeated to get
\[
\frac{d^2}{d\tau^2} \frac{\partial H}{\partial y} = - E^T C \dot{x} - E^T D^T \lambda = 0
\] (3.17)

If (3.12) and (3.14) are used to substitute for \( \dot{x} \) and \( \dot{\lambda} \), (3.17) becomes

\[
0 = - E^T C (Dx + Ey) - E^T D (-A-Cx-D^T \lambda)
\] (3.18)

This allows the following solution for the control on a singular arc

\[
y = \left[ (-E^T CD + E^T D^T C)x + E^T D^T D^T \lambda + E^T D^T A \right]/E^T CE
\] (3.19)

Of course, \( E^T CE \neq 0 \), as was verified by a numerical check. In fact,

\[
- \frac{\partial}{\partial y} \frac{d^2}{d\tau^2} \frac{\partial H}{\partial y} = E^T CE > 0.
\] (3.20)

Equation (3.20) is the generalized convexity condition for linear/quadratic singular arc problems (see [3], p.246).
Further use can be made of equations (3.15) and (3.16) since they form the following system.

\[
\begin{bmatrix}
    E^T \\
    E^TD^T
\end{bmatrix} \lambda = \begin{bmatrix}
    0 \\
    -E^T(A+Cx)
\end{bmatrix}
\]

(3.21)

This system can be used to solve for \( \lambda \) provided the inverse of \( \begin{bmatrix}
    E^T \\
    E^TD^T
\end{bmatrix} \) exists. This condition, which was again numerically checked and found to hold, is equivalent to Kalman’s controllability condition (see [15]) that the matrix \([E,DE]\) must have rank 2.

Let \( \Delta = \text{determinant of } \begin{bmatrix}
    E^T \\
    E^TD^T
\end{bmatrix} \) and enumerate the elements of \( E^T = [e_1, e_2] \). Substitute (3.19) and the solution for \( \lambda \) in (3.21) into (3.12) to obtain

\[
\dot{x} = \left\{ D - (1/E^TCE)[EE^T(CD-D^TC) + EE^TD^TD^T/\Delta[e_2]^T]E^T\right\} x
\]

\[
+ (1/E^TCE)[EE^TD^TD^T/\Delta[e_2]^TA - EE^TD^TA],
\]
or more simply

\[ \dot{x} = Sx + F \]  \hspace{1cm} (3.22)

This is a linear, constant coefficient, ordinary differential equation whose initial conditions, the point of entry onto the singular arc, are yet to be determined.

The linear/quadratic problem solved here allows for an unbounded control. This means that an impulse in the control moves the state of the system instantaneously from the boundary conditions to the singular arc and another impulse returns the state of the system to the boundary conditions. By examining (3.12) it is seen that impulses in \( y \) will move the system along lines of constant

\[ e_{2} x_{1} - e_{1} x_{2} \]

value.

If \( 0^{+} \) and \( 1^{-} \) are used to designate the entry and exit times (\( \tau \) values) for the singular arc, the following equations can then be written.

\[ e_{2} x_{1}(0^{+}) - e_{1} x_{2}(0^{+}) = e_{2} \dot{x}_{1} - e_{1} \dot{x}_{2} \]  \hspace{1cm} (3.23)
Along a singular arc we may write $H$, equation (3.13), as a function of $x$ alone through the use of equations (3.19) and (3.21). $x_1(1^-)$ and $x_2(1^-)$ may be written in terms of $x_1(0^+)$ and $x_2(0^+)$ by solving equation (3.22). Eliminating $x_1(1^-)$ and $x_2(1^-)$ from (3.23) and (3.24), a solution can be found for $x_1(0^+)$ and $x_2(0^+)$ in terms of $\hat{x}$ and $E$. This enables equation (3.22) to be solved.

An algorithm was devised to compute these solutions numerically. A program listing of that algorithm is contained in Appendix C. Solutions were computed for the $X_f = 1000, 2000, 3000, 4000, \text{ and } 5000$ meters problems.

The algorithm essentially does the computations involved to produce an analytical solution. For ease of presentation, that analytical solution is printed out at a selected step size. The computations involve solving for the eigenvalues of the matrix in equation (3.22) as well as the constants of integration. The two eigenvalues involved grew so large (as $X_f$ increases) that the computer can no longer handle the arithmetic involved even though

$$e_2x_1(1^-) - e_1x_2(1^-) = e_2\hat{x}_1 - e_1\hat{x}_2 \quad (3.24)$$
double precision was used. The eigenvalues, both real, are shown for increasing values of \( X_f \) in Table 3.2.

The solution of this two-dimensional problem cannot, of course, show the oscillation about the MacCready values apparent in the optimal solutions. Instead the state, costate, and control remain fairly constant through a broad midrange area. The control histories for the \( X_f = 5000 \) meters problem for the suboptimal linear/quadratic method and the optimal conjugate gradient method are compared in Figure 3.2.

A further comparison of the linear/quadratic and conjugate gradient results using some selected data points is made in Table 3.3. This table highlights the major discrepancies in the suboptimal solution and further points out some of the advantages. The advantages are a reasonable agreement with the state and control trajectories, a close approximation of the minimum time of flight, and a greatly reduced cost.

The discussion of the linear/quadratic method will be concluded with the presentation of Figure 3.3. The state space \((V,\gamma)\) plane trajectory of the solution to the \( X_f = 1000 \) meters suboptimal problem is shown in this
figure. It also shows the path of motion of the state variables when acted upon by an impulse in the control.

Section 3.3. Matched Asymptotic Expansions

In order to understand the next attempt at suboptimal solutions which will be presented, it is necessary to review the theory of singular perturbations and matched asymptotic expansions (see references [16] through [18] and especially [19]). Most of the applications of this theory have dealt with boundary value problems for finite order scalar differential equations or with initial value problems for first order systems. Optimal control problems result in a non-linear, in general, two-point boundary value problem. To begin with the theory will be reviewed for initial value problems.

Consider the initial value problem

\[
\frac{dx}{dt} = f(x, y) \quad x(\varepsilon, 0) = \bar{x} \tag{3.25}
\]

\[
\varepsilon \frac{dy}{dt} = g(x, y) \quad y(\varepsilon, 0) = \bar{y} \tag{3.26}
\]
where $x(\epsilon, t)$ and $y(\epsilon, t)$ are scalars, $\epsilon$ is positive, and $\bar{x}$ and $\bar{y}$ are constants. If $\epsilon$ is a "small" parameter, then $\frac{dy}{dt}$ may be relatively large when compared to $\frac{dx}{dt}$. Therefore, $x$ and $y$ are referred to as "slow" and "fast" variables, respectively.

To solve (3.25) and (3.26) it is natural to set $\epsilon = 0$, and solve the resulting problem with the hope that a good approximation of the solution to the original problem will be obtained. This is the reduced, or free stream, problem whose solution is denoted by $x^r_o(t)$ and $y^r_o(t)$. These variables satisfy the following.

$$\frac{dx^r_o}{dt} = f(x^r_o, y^r_o) \quad x^r_o(0) = \bar{x} \quad (3.27)$$

$$0 = g(x^r_o, y^r_o) \quad (3.28)$$

Since this is a first order system only one boundary condition can be retained. The boundary condition is retained on the slow variable and the boundary condition is met on the fast variable by allowing a discontinuity at $t = 0$. 
The best that can be hoped for is that $x_0^r$ is a good approximation to the true solution, $x(t)$, everywhere and that $y_0^r$ is a good approximation everywhere except near $t = 0$. This assumes, of course, that the exceptional case of $0 = g(x, y)$ is excluded.

To study the behavior of $y(t)$ near $t = 0$, the time scale is "stretched" by using the transformation $\tau = \frac{t}{\epsilon}$ in order to obtain

\[
\frac{dx}{d\tau} = \epsilon f(x, y) \quad x(\epsilon, 0) = x \tag{3.29}
\]

\[
\frac{dy}{d\tau} = g(x, y) \quad y(\epsilon, 0) = y \tag{3.30}
\]

These are referred to as boundary layer equations. Once again an approximate solution is found by setting $\epsilon = 0$ which yields

\[
x_0^b(\tau) = x \tag{3.31}
\]
It should be noted that \( x^r_0, y^r_0 \) provides an equilibrium point for the zeroth order boundary layer equations. The crucial condition for "matching" the boundary layer and reduced solutions is the stability of the zeroth order boundary layer equations (see [19]).

If a more accurate approximation to the true solution is desired, the method of matched asymptotic expansions may be tried. Assume that \( x^r \) and \( y^r \) may be represented asymptotically by

\[
x^r(t) = x^r_0(t) + x^r_1(t)\epsilon + x^r_2(t)\epsilon^2 + \ldots \tag{3.33}
\]

\[
y^r(t) = y^r_0(t) + y^r_1(t)\epsilon + y^r_2(t)\epsilon^2 + \ldots \tag{3.34}
\]

and similarly
\[ x^b (\tau) = x^b_0 (\tau) + x^b_1 (\tau) \epsilon + x^b_2 (\tau) \epsilon^2 + \ldots \quad (3.35) \]

\[ y^b (\tau) = y^b_0 (\tau) + y^b_1 (\tau) \epsilon + y^b_2 (\tau) \epsilon^2 + \ldots \quad (3.36) \]

To solve (3.25) and (3.26), put the expansions (3.33) and (3.34) into \( f \) and \( g \) and expand about \( \epsilon = 0 \) to obtain

\[
\begin{align*}
  f(x^r, y^r) &= f(x^r_0, x^r_1 \epsilon + x^r_2 \epsilon^2 + \ldots, y^r_0, y^r_1 \epsilon + y^r_2 \epsilon^2 + \ldots) \\
  &= f(x^r_0, y^r_0) + \left( \frac{\partial f}{\partial x} x^r_1 \epsilon + \frac{\partial f}{\partial y} y^r_1 \epsilon \right) + \ldots \\
  &= f^r_0 + f^r_1 \epsilon + \ldots
\end{align*}
\]

where, for example, \( f^r = \frac{\partial f}{\partial x} (x^r_0, y^r_0) \). Similarly,

\[
\begin{align*}
  g(x^r, y^r) &= g^r_0 + g^r_1 \epsilon + \ldots \quad (3.38)
\end{align*}
\]

Substituting (3.33), (3.34), (3.37) and (3.38) into (3.25) and (3.26) yields
Equating like powers of $\varepsilon$ yields a succession of problems

\begin{align*}
\frac{dx^r}{dt} + \frac{dx^r}{dt} \varepsilon + \ldots &= f^r_o + f^r_1 \varepsilon + \ldots \\
\frac{dy^r}{dt} + \frac{dy^r}{dt} \varepsilon + \ldots &= g^r_o + g^r_1 \varepsilon + \ldots
\end{align*}  

(3.39)

(3.40)

\begin{align*}
\frac{dx^r}{dt} &= f^r_o; \quad 0 = g^r_o \quad \text{for } i = 0 \\
\frac{dx^r_i}{dt} &= f^r_i; \quad \frac{dy^r_i}{dt} = g^r_i \quad \text{for } i > 0
\end{align*}  

(3.41)

(3.42)

Each of these problems is first order and for $i > 0$ they are linear. The constants of integration, $c^r_i$, are as yet undetermined.

A similar development in the boundary layer system leads to a succession of problems

\begin{align*}
\frac{dx^b_o}{dt} &= 0; \quad \frac{dy^b_o}{dt} = q^b_o; \quad x^b_o(0) = x; \quad y^b_o(0) = y \quad \text{for } i = 0
\end{align*}  

(3.43)
\[ \frac{dx_i^b}{d\tau} = f_i^b; \quad \frac{dy_i^b}{d\tau} = g_i^b; \quad x_i^b(0) = 0; \quad y_i^b(0) = 0 \quad \text{for} \quad i > 0 \]

(3.44)

As in the reduced system, each of the boundary layer problems is first order and for \( i > 0 \) they are linear.

The matching principle (see [19]) can now be applied to determine the constants \( c_i^r, i = 0,1,... \). This principle, stated simply, is

\[ \lim_{\epsilon \to 0} \lim_{t \to 0} \lim_{\tau \to \infty} [x^r(\epsilon,t) - x^b(\epsilon,\tau)] = 0 \]

(3.45)

where it is noted that \( \epsilon \to 0 \) faster than \( t \to 0 \) since \( \tau = \frac{t}{\epsilon} \). If the matching is accomplished for \( x \), then this will force \( y \) to be matched and therefore no matching condition is applied to \( y \). In order to apply (3.45), the behavior of \( x^r(\epsilon,t) \) for small \( t \) and \( x^b(\epsilon,\tau) \) for large \( \tau \) must be known. Expanding the coefficients of (3.33) in a power series about \( t = 0 \) yields
\[
x_i^r(t) = x_i^r(0) + \frac{dx_i^r}{dt}(0)t + ... \\
= c_i^r + f_i^r(0)t + ...
\]

where (3.42) has been used.

To determine the behavior of \( x^b(\varepsilon, \tau) \) for large \( \tau \)
notice that (3.43) shows \( \frac{dy^b_o}{d\tau} = g(x, y^b_o(\tau)) \) and the assumed
asymptotic stability of the boundary shows that
\[
\lim_{\tau\to0} y^b_o(\tau) = \emptyset(x), \text{ where } \emptyset(x) \text{ is determined from }
0 = g(x, \emptyset(x)). \text{ Hence } f^b_o(\tau) = f(x^b_o, y^b_o) = f(x, y^b_o)
\]
approaches \( f(x, \emptyset(x)) \) as \( \tau \to \infty \). To proceed, further
knowledge of the functions \( f(x, y) \) and \( g(x, y) \) is needed.

The application of this technique to two-point boundary
value problems is reasonably straightforward. Boundary
layers are required at both the initial and terminal times.
Difficulty is often encountered in optimal control problems
because the matching principle, equation (3.45), must
specify constants of integration which suppress unstable
modes in the boundary layer equations and therefore produce
stable boundary layers which match with the reduced solution. Generally the boundary layers and reduced solutions can only be obtained numerically which makes the matching particularly difficult. Furthermore, an optimal control problem is well suited for solution by this method if some of the variables are "slow", that is, they change slowly over the entire domain of the independent variable, and some variables are "fast", and hence change rapidly near the boundaries but more slowly over the rest of the interval of solution.

A check of the optimal trajectories calculated in Chapter II demonstrates that the two state variables, $V$ and $\gamma$, both behave like "fast" variables in the problem described by equations (1.16) - (1.18). For this reason, along with those of the previous paragraph, the usual procedure of matched asymptotic expansions (one slow and one fast variable) is not applicable.

A numerical procedure was attempted which would produce a solution equivalent to the zeroth order composite (matched boundary layers and reduced solutions) solution obtained by treating both $V$ and $\gamma$ as fast variables. In applying matched asymptotic expansions to optimal control
problems, the costates variables, equation (2.11), which arise in the two-point boundary value problem are treated as slow (fast) variables if the corresponding state variable is slow (fast). Hence all variables in the two-point boundary value problem are treated as fast variables. By placing an $\epsilon$ in front of the derivative terms of equations (2.16) - (2.19) and setting $\epsilon = 0$, one obtains

$$0 = \eta \frac{c^r}{D_o} \frac{\nu^r}{\cos \gamma^r_o} - \sin \gamma^r_o \left( \nu^r_o \cos \gamma^r_o \right)$$  \hspace{1cm} (3.47)$$

$$0 = \eta u^r_o \cos \gamma^r_o - \frac{1}{\nu^r_o}$$  \hspace{1cm} (3.48)$$

$$0 = \frac{1}{(\nu^r_o \cos \gamma^r_o)} - \frac{\nu^r}{\nu^r_o} \left( - \eta \frac{c^r}{D_o} / \cos \gamma^r_o \right) + \frac{\sin \gamma^r_o}{(\nu^r_o \cos \gamma^r_o)} \frac{(\nu^r_o \cos \gamma^r_o)^2}{(\nu^r_o \cos \gamma^r_o)} - 2\frac{\lambda^r}{\nu^r_o}$$ \hspace{1cm} (3.49)$$

$$0 = \frac{(\nu^r - \nu_T \sin \gamma^r_o)}{(\nu_T \nu^r_o \cos \gamma^r_o)} - \frac{\nu^r}{\nu^r_o} (- \eta \frac{c^r}{D_o} \nu^r \sin \gamma^r_o / \cos^2 \gamma^r_o)$$

$$- \frac{1}{(\nu^r_o \cos^2 \gamma^r_o)} - \frac{\nu^r}{\nu^r_o} \eta u^r_o \sin \gamma^r_o / \cos^2 \gamma^r_o$$ \hspace{1cm} (3.50)$$
These equations along with the optimality condition, equation (2.21), defines the reduced solution which yields the static MacCready values (see Appendix A).

After applying time stretching transformations and setting \( \varepsilon = 0 \), the boundary layer equations are seen to be exactly the same as the original two-point boundary value problem. The composite (matched) zeroth order approximate solution would then consist of an initial boundary layer solution transferring the state of the system from the initial point to the static MacCready values, a constant reduced solution of static MacCready values, and a terminal boundary layer returning the system to the boundary conditions. The matching conditions can be stated very precisely since the reduced solution can be found analytically and is constant. The difficulty which was encountered here was the selection of initial (terminal) values for the costate variables which would suppress unstable modes and allow the initial (terminal) boundary layer solution to reach the matching values.

For the algorithm which was devised here, initial guesses are selected for \( \lambda^{b}_{\text{inc}}(0) \) and \( \lambda^{b}_{\text{term}}(0) \) exactly as was described in Chapter II. The initial boundary layer
equations are then integrated forward until sensitivity is encountered. Sensitivity is defined in the algorithm as the velocity growing larger than the MacCready value by some prescribed tolerance. Once sensitivity is encountered, the adjoint method is used to correct the initial values for $\lambda_V$ and $\lambda_Y$. To do this the "final time" for the boundary layer is set at the point where sensitivity is encountered, the "final time" conditions are imposed (all variables equal to the MacCready values), and backward integration of the adjoint equations provide an update on the $\lambda_V$ and $\lambda_Y$ initial values. This procedure is then continued by integrating forward until sensitivity is encountered once again.

Solutions were attempted for the $X_f = 1000$ and 2000 meter problems with $V_T = 2m/sec$. In the $X_f = 1000$ meter problem, the continuation procedure failed after a range of 260 meters. Initial boundary layer length was determined, in Figures 2.1 and 2.5, to be about 300 meters. A comparison of the values obtained at 260 meters in the initial boundary layer, at 300 meters in the conjugate gradient algorithm of Chapter II, and the static MacCready values is shown in Table 3.4. This table indicates that
the flight path angle, $\gamma$, shows a good deal of inaccuracy in the current suboptimal algorithm while all other variables have converged reasonably well. Since $\gamma$ has not converged to the MacCready value no matched, composite solution is possible.
Table 3.1. Cost and Performance of the Quasilinearization Algorithms

<table>
<thead>
<tr>
<th>Type of Algorithm</th>
<th>( X_f ) (m)</th>
<th>Time of Flight (sec)</th>
<th>% of Increase in Time of Flight</th>
<th>CPU Time (sec)</th>
<th>% Decrease in CPU Time</th>
<th>Control ( (C_L) ) at Mid-Range Value(^a)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimal</td>
<td>1000</td>
<td>40.399</td>
<td>—</td>
<td>6.71</td>
<td>—</td>
<td>0.296147</td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>78.600</td>
<td>—</td>
<td>33.46</td>
<td>—</td>
<td>0.300472</td>
</tr>
<tr>
<td>Suboptimal</td>
<td>1000</td>
<td>40.558</td>
<td>0.394</td>
<td>2.54</td>
<td>62.15</td>
<td>0.302316</td>
</tr>
<tr>
<td>(First Iteration Only)</td>
<td>2000</td>
<td>78.755</td>
<td>0.197</td>
<td>2.53</td>
<td>92.44</td>
<td>0.300426</td>
</tr>
</tbody>
</table>

\(^a\)Note that the static MacCready value is 0.300417
Table 3.2. Eigenvalues of the Linear/Quadratic System Matrix ($s$, eqn. (3.22)) Along the Singular Arc

<table>
<thead>
<tr>
<th>$x_f$</th>
<th>Positive Eigenvalue</th>
<th>Negative Eigenvalue</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>4.624</td>
<td>- 7.129</td>
</tr>
<tr>
<td>2000</td>
<td>8.966</td>
<td>- 18.676</td>
</tr>
<tr>
<td>3000</td>
<td>14.388</td>
<td>- 36.001</td>
</tr>
<tr>
<td>4000</td>
<td>21.468</td>
<td>- 59.678</td>
</tr>
<tr>
<td>5000</td>
<td>30.427</td>
<td>- 89.932</td>
</tr>
<tr>
<td>5250</td>
<td>32.973</td>
<td>- 98.535</td>
</tr>
<tr>
<td>5500</td>
<td>35.643</td>
<td>- 107.556</td>
</tr>
<tr>
<td>5750</td>
<td>38.439</td>
<td>- 116.996</td>
</tr>
<tr>
<td>6000</td>
<td>41.360</td>
<td>- 126.855</td>
</tr>
<tr>
<td>6250</td>
<td>44.407</td>
<td>- 137.133</td>
</tr>
<tr>
<td>6500</td>
<td>47.581</td>
<td>- 147.832</td>
</tr>
<tr>
<td>6750</td>
<td>50.882</td>
<td>- 158.952</td>
</tr>
<tr>
<td>7000</td>
<td>54.310</td>
<td>- 170.492</td>
</tr>
<tr>
<td>10000</td>
<td>105.495</td>
<td>- 341.916</td>
</tr>
</tbody>
</table>
Table 3.3. Comparison of Linear/Quadratic and Conjugate Gradient Results

<table>
<thead>
<tr>
<th>$X_f$ (m)</th>
<th>Method</th>
<th>Time of Flight (sec)</th>
<th>Max Alt Loss (m)</th>
<th>Max Speed (m/sec)</th>
<th>Max $^a$ DBM EGT (m)</th>
<th>CPU Time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>L/Q</td>
<td>41.418</td>
<td>78.574</td>
<td>39.187</td>
<td>65.419</td>
<td>1.56</td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>40.382</td>
<td>76.969</td>
<td>41.579</td>
<td>58.842</td>
<td>70.34</td>
</tr>
<tr>
<td>2000</td>
<td>L/Q</td>
<td>79.802</td>
<td>111.205</td>
<td>41.197</td>
<td>77.085</td>
<td>2.16</td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>78.583</td>
<td>104.935</td>
<td>41.489</td>
<td>58.433</td>
<td>102.57</td>
</tr>
<tr>
<td>5000</td>
<td>L/Q</td>
<td>195.412</td>
<td>230.561</td>
<td>41.275</td>
<td>108.507</td>
<td>5.17</td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>193.186</td>
<td>188.666</td>
<td>41.494</td>
<td>58.466</td>
<td>122.51</td>
</tr>
</tbody>
</table>

$^a_{DBM EGT} = \text{distance below MacCready equilibrium glide trajectory}$
Table 3.4. Matching of Reduced Solution and the Initial Boundary Layer Solution

<table>
<thead>
<tr>
<th>Matched Asymptotic Expansions, Initial Boundary Layer x = 260m</th>
<th>Numerical (Conjugate gradient) solution of the optimal control problem x = 300m</th>
<th>Static MacCready Values (Reduced Solution)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity (m/sec)</td>
<td>41.265</td>
<td>41.279</td>
</tr>
<tr>
<td>Flight Path Angle (radians)</td>
<td>-0.129</td>
<td>-0.055</td>
</tr>
<tr>
<td>$\lambda_{V}$ (non-dim)</td>
<td>-20.681</td>
<td>$^a$</td>
</tr>
<tr>
<td>$\lambda_{\gamma}$ (non-dim)</td>
<td>-0.035</td>
<td>$^a$</td>
</tr>
<tr>
<td>Lift Coef. (non-dim)</td>
<td>0.307</td>
<td>0.323</td>
</tr>
</tbody>
</table>

$^a$Since terminal state constraints are met by projection in the conjugate gradient algorithm costate variables are not the same as those calculated in the other numerical methods encountered.
FIGURE 3.1 TRAJECTORIES FOR

$X_f = 1000$ m AND $V_f = 2$ m/s

VIA QUASILINEARIZATION
FIGURE 3.2: OPTIMAL CONTROL HISTORIES FOR

\[ X_f = 5000 \text{ m} \quad \text{AND} \quad V_f = 2 \text{ m/s} \]
Figure 3.3: State space trajectory for linear/quadratic solution to $x_f = 1000\text{m}, v_f = 2\text{m/s}$ problem
CHAPTER IV. EXTENSIONS OF THE THERMAL MODEL

Section 4.1. Distributed End-Point Thermal

Up to this point the thermals have been modeled as vertical winds of constant strength occurring at a single point (zero width). In this chapter a more realistic extension of this model is pursued. This model (see [20]) assumes an exponential distribution of the vertical wind profile as sketched in Figure 4.1. That is, each of the thermals show a peak updraft value, \( C \), which occurs at the center of the thermal (\( \tau = 0 \) for the initial thermal and \( \tau = 1 \) for the terminal thermal), and zero updraft values at \( \tau = R \) and \( \tau = 1 - R \) for the initial and terminal thermals, respectively. The vertical wind profile is the sum of the initial and terminal thermals and is given by

\[
\omega(\tau) = Ce^{-\left(\frac{T}{R}\right)^2} \left[1 - \left(\frac{T}{R}\right)^2\right] + Ce^{-\left(\frac{1-T}{R}\right)^2} \left[1 - \left(\frac{1-T}{R}\right)^2\right]
\]

(4.1)

Note that this wind distribution is assumed to be independent of altitude. The derivative with respect to \( \tau \)
\[
\frac{dw(\tau)}{d\tau} = Ce^{-\left(\frac{\tau}{R}\right)^2} \left(\frac{-2\tau}{R^2}\right) \left[2 - \left(\frac{\tau}{R}\right)^2\right] + Ce^{-\left(\frac{1-\tau}{R}\right)^2} \left(\frac{2(1-\tau)}{R^2}\right) \left[2 - \left(\frac{1-\tau}{R}\right)^2\right]
\]

(4.2)

where \( C = V_T + \text{minimum rate of sink of the sailplane} \). It should be pointed out that the thermals are of equal strength, and the "elevator motion" is still retained. That is, once the sailplane has reached the range \( \tau = 1 \), it rises vertically to the prescribed altitude, \( h(0) \).

The performance index (equation (1.16)) undergoes the following change. Beginning with equation (1.4),

\[
J = t_f + \frac{[h(0) - h(t_t)]}{V_T}
\]

\[
= \int_0^1 \frac{dx}{V \cos \gamma} - \frac{1}{V_T} \int_0^1 \frac{\psi(\tau) + V \sin \gamma}{V \cos \gamma} d\tau
\]

\[
= \int_0^1 \frac{V_T - \psi(\tau) - V \sin \gamma}{V V_T \cos \gamma} d\tau
\]

(4.3)
where the nondimensionalizing transformations found in equations (1.13) and (1.14) have been used. The only change in the performance index (1.16) is the $w(\tau)$ term in equation (4.3).

The differential equations (1.17) and (1.18) become

\[
\frac{dv}{d\tau} = - \left[ (1 + v \frac{dw}{d\tau} \cos \gamma) \sin \gamma + \eta c_D V^2 \right] / (V \cos \gamma) \quad (4.4)
\]

\[
\frac{dv}{d\tau} = \left[ - (1 + \frac{dw}{d\tau} \cos \gamma) \cos \gamma + \eta u V^2 \right] / (V^2 \cos \gamma) \quad (4.5)
\]

These equations are essentially those derived in Appendix D for three-dimensional flight. The difference is in the choice of nondimensionalizing units. The boundary conditions remain the same as before. The revised problem can now be summarized as

\[
\min J = \int_0^1 \frac{V_T - w(\tau) - V \sin \gamma}{V V_T \cos \gamma} \, d\tau \quad (4.6)
\]
subject to

$$\frac{dv}{d\tau} = - \left[ (1 + V \frac{dw}{d\tau} \cos \gamma) \sin \gamma + \eta C_D V^2 \right]/(V \cos \gamma)$$ (4.7)

$$\frac{dy}{d\tau} = - \left[ (1 + \frac{dw}{d\tau} \cos \gamma) \cos \gamma + \eta u V^2 \right]/(V^2 \cos \gamma)$$ (4.8)

with $V(0) = V(1) = V_o$  
$\gamma(0) = \gamma(1) = \gamma_o$

the revised Hamiltonian, $H$, is given by

$$H = \frac{V_T - w(\tau) - V \sin \gamma}{V V_T \cos \gamma}$$

$$+ \lambda_V \left[ -(1 + V \frac{dw}{d\tau} \cos \gamma) \sin \gamma + \eta C_D V^2 \right]/(V \cos \gamma) \right]$$

$$+ \lambda_\gamma \left[ -\left[ (1 + \frac{dw}{d\tau} \cos \gamma) \cos \gamma + \eta u V^2 \right]/(V^2 \cos \gamma) \right]$$

(4.9)

where $\lambda_V$ and $\lambda_\gamma$ are the costate values defined by the
differential equations
\[ \frac{d\lambda}{d\tau} = \frac{-\partial H}{\partial \lambda} = \frac{1}{(V^2 \cos \gamma)} - \frac{w(\tau)}{(V_T^2 \cos \gamma)} \]

\[ - \lambda_V \left( - \eta \frac{C_D}{\cos \gamma} + \sin \gamma/(V^2 \cos \gamma) \right) \]

\[ - \lambda_Y \left( 2 + 2 \frac{dw}{d\tau} \cos \gamma \right)/V^3 \]  

(4.10)

\[ \frac{d\lambda}{d\tau} = \frac{-\partial H}{\partial \lambda} = \frac{(V - V_T \sin \gamma + w(\tau) \sin \gamma)}{(V_T \cos^2 \gamma)} \]

\[ - \lambda_V \left( - \eta \frac{C_D}{\cos \gamma} \sin \gamma/\cos^2 \gamma - 1/(V \cos^2 \gamma) - \frac{dw}{d\tau} \cos \gamma \right) \]

\[ - \lambda_Y \left( \eta u \sin \gamma/\cos^2 \gamma + \frac{dw}{d\tau} \sin \gamma/V^2 \right) \]  

(4.11)

where \( u \) is defined by the optimality condition which has not changed and is given in equation (2.21). Equations (4.7), (4.8), (4.10), and (4.11), along with the prescribed boundary conditions and the optimality condition, describe the revised two-point boundary value problem.

This revised problem has been solved numerically by both the quasilinearization and conjugate gradient
algorithms described in Chapter II. The quasilinearization algorithm was used to solve the problem with various values of the parameters $V_T$ and $R$. With $V_T$ set equal to 2, solutions were obtained for $R = 0.05, 0.075, 0.1, 0.125, 0.15,$ and, 0.2. Then with $R$ set equal to 0.1, solutions were obtained for $V_T = 0.5, 1, 2, 3,$ and 4. All of the problems solved in this chapter had a range between thermals, $X_f$, of 1000 meters. The conjugate gradient algorithm, being significantly more expensive than the quasilinearization algorithm, was used only to verify the accuracy of the results obtained with the quasilinearization algorithm.

The optimal trajectories and control histories for several of these problems are depicted in Figures 4.2, 4.3, 4.4, and 4.5. In Figures 4.2 and 4.3 $V_T = 2$ remains constant, while in Figures 4.4 and 4.5 $R = 0.1$ remains constant. The changes in the thermal radius, $R$, cause a change in the trajectories near the boundary as is evident in Figures 4.2 and 4.3. On the other hand, a change in the thermal strength, $V_T$, causes a change during the mid-range portion of the flight. A comparison of all the solutions obtained appears in Table 4.1 and Table 4.2.
These tables show that the changes in the thermal strength cause larger changes in the performance index than do changes in the thermal radius. These tables also permit a comparison with the $X_f = 1000$ meter, zero-width thermal solution of Chapter II. The time spent in the "elevator" motion at the terminal value of range is slightly greater in the distributed thermal problems than in the zero-width thermal problem. For example, the final altitude loss for the $V_T = 2m/s$, zero-width thermal is 25.99m, while the final altitude loss for the $V_T = 2m/s$, $R = 0.1$, distributed end-point thermal is 26.74m. It should be noted that the insertion of distributed end-point thermals did not greatly alter the results obtained with the zero-width thermals unless an unrealistically large value of the thermal radius, $R$, was used.

This section will be closed with a comparison of the quasilinearization and conjugate gradient solutions for $V_T = 2m/s$, $R = 0.1$, and $X_f = 1000m$. This is done in an effort to verify the accuracy of the results already cited and to provide a further performance comparison between the two methods. This comparison is done in Table 4.3. The two methods produce very similar results while the
quasilinearization method required considerably less CPU time. Both methods used the $X_f = 1000$ meters, $V_T = 2m/s$, zero-width thermal model solution as starting data.

Section 4.2. Distributed Mid-range Thermal

This section deals with another model of vertical wind distribution. An exponential distribution given by

$$w(\tau) = Ce^{-(0.5-\tau)^2} \left[1 - \left(\frac{0.5 - \tau}{R}\right)^2 \right]$$

is inserted into the problem. Zero-width thermals and the "elevator motion" are retained at the end-points. The value of $C$ in equation (4.12) is plus the minimum rate of sink of the sailplane. The derivative of this vertical wind profile with respect to $\tau$ is given by

$$\frac{dw(\tau)}{d\tau} = 2Ce^{-(0.5-\tau)^2} \left(\frac{0.5 - \tau}{R}\right) \left[2 - \left(\frac{0.5 - \tau}{R}\right)^2 \right]$$
This thermal model is adopted in order to calculate the optimal trajectory for the sailplane when an "unexpected", weaker thermal is encountered during the mid-range portion of the flight. The technique employed by most sailplane pilots is to do a pull-up while in a rising thermal current and a dive while in a down draft ([21], p.26) in order to get the most benefit from the mid-range thermal. For this reason the term Dolphin soaring is applied to the problem of planar flight through alternate upcurrents and down currents between thermals.

The solutions obtained by the conjugate gradient algorithm were for end-point thermal strengths of 4m/s and 2m/s. In both cases the mid-range thermal strength was 1m/s with a thermal radius of \( R = 0.2 \). The results for the \( V_T = 4m/s \) end-point thermal are displayed in Table 4.4 and Table 4.5. These results indicate that in the mid-range thermal case a steeper descent occurs just prior to the range \( = 500m \) point. Also, a more rapid ascent occurs just after the range \( = 500m \) point. The final altitude loss is approximately the same (0.41m less for the mid-range thermal), but the mid-range thermal effects are evident in the fact that the performance index
shows a value of 32.66 seconds for the mid-range thermal problem and a value of 33.15 seconds without the mid-range thermal.
Table 4.1. The Effects of Varying Thermal Strength \((V_T)\) via the Quasilinearization Method \((X_f = 1000m, R = 0.1)\)

<table>
<thead>
<tr>
<th>Flight Time (sec)</th>
<th>Percent Diff. (^a)</th>
<th>Max Speed (m/s)</th>
<th>Max C(_L)</th>
<th>Min C(_L)</th>
<th>Max Altitude Loss (m)</th>
<th>Max DBMEGT (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CGI Method (V_T = 2m/s, R = 0)</td>
<td>40.382</td>
<td>0.0</td>
<td>41.579</td>
<td>0.603</td>
<td>-0.197</td>
<td>76.969</td>
</tr>
<tr>
<td>QL Method (V_T)</td>
<td>72.867</td>
<td>80.444</td>
<td>32.747</td>
<td>0.715</td>
<td>0.395</td>
<td>41.878</td>
</tr>
<tr>
<td>0.5</td>
<td>50.889</td>
<td>26.019</td>
<td>36.082</td>
<td>0.722</td>
<td>0.295</td>
<td>53.600</td>
</tr>
<tr>
<td>1</td>
<td>38.828</td>
<td>-3.848</td>
<td>41.773</td>
<td>0.759</td>
<td>0.091</td>
<td>76.777</td>
</tr>
<tr>
<td>2</td>
<td>34.257</td>
<td>-15.168</td>
<td>45.999</td>
<td>0.806</td>
<td>-0.116</td>
<td>96.809</td>
</tr>
<tr>
<td>3</td>
<td>31.733</td>
<td>-21.418</td>
<td>49.094</td>
<td>0.859</td>
<td>-0.325</td>
<td>113.347</td>
</tr>
</tbody>
</table>

\(^a\)Flight Time - CGI Flight Time / CGI Flight Time \times 100
Table 4.2. The Effects of Varying Thermal Radius (R) via the
Quasilinearization Method \( (X_f = 1000\text{m}, V_T = 2\text{m/s}) \)

<table>
<thead>
<tr>
<th>Method</th>
<th>Flight Time (sec)</th>
<th>Percent Diff. (^a)</th>
<th>Max Speed (m/s)</th>
<th>Max CL</th>
<th>Min CL</th>
<th>Max Altitude Loss (m)</th>
<th>Max DBMEGT (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CGI Method</td>
<td>V(_T) = 2m/s, R = 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40.382</td>
<td>0.0</td>
<td>41.579</td>
<td>0.603</td>
<td>-0.197</td>
<td>76.969</td>
<td>58.842</td>
<td></td>
</tr>
<tr>
<td>QL Method</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.050</td>
<td>39.805</td>
<td>-1.429</td>
<td>41.619</td>
<td>0.741</td>
<td>0.364</td>
<td>76.728</td>
<td>58.994</td>
</tr>
<tr>
<td>0.075</td>
<td>39.366</td>
<td>-2.516</td>
<td>41.678</td>
<td>0.757</td>
<td>0.180</td>
<td>76.800</td>
<td>59.397</td>
</tr>
<tr>
<td>0.100</td>
<td>38.828</td>
<td>-3.848</td>
<td>41.773</td>
<td>0.759</td>
<td>0.091</td>
<td>76.777</td>
<td>59.881</td>
</tr>
<tr>
<td>0.125</td>
<td>38.238</td>
<td>-5.309</td>
<td>41.909</td>
<td>0.745</td>
<td>0.053</td>
<td>76.648</td>
<td>60.472</td>
</tr>
<tr>
<td>0.150</td>
<td>37.628</td>
<td>-6.820</td>
<td>42.063</td>
<td>0.721</td>
<td>0.043</td>
<td>76.500</td>
<td>61.093</td>
</tr>
<tr>
<td>0.200</td>
<td>36.417</td>
<td>-9.819</td>
<td>42.092</td>
<td>0.676</td>
<td>0.067</td>
<td>75.682</td>
<td>61.067</td>
</tr>
</tbody>
</table>

\(^a\) (Flight Time - CGI Flight Time) \times 100

\(\frac{\text{Flight Time}}{\text{CGI Flight Time}}\) x 100
Table 4.3. A Comparison of $V_T = 2\text{m/s}$, $R = 0.1$, $X_f = 1000\text{m}$ Solutions Using Both Conjugate Gradient and Quasilinearization Methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>CGI</th>
<th>QL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time of Flight (sec)</td>
<td>38.814</td>
<td>38.828</td>
</tr>
<tr>
<td>Max Altitude Loss (m)</td>
<td>77.051</td>
<td>76.777</td>
</tr>
<tr>
<td>Max DBMEGT (m)</td>
<td>60.038</td>
<td>59.881</td>
</tr>
<tr>
<td>Max Speed (m/s)</td>
<td>41.775</td>
<td>41.773</td>
</tr>
<tr>
<td>Max $C_L$</td>
<td>0.765</td>
<td>0.759</td>
</tr>
<tr>
<td>Min $C_L$</td>
<td>0.078</td>
<td>0.091</td>
</tr>
<tr>
<td>CPU Time (sec)</td>
<td>42.16</td>
<td>5.53</td>
</tr>
</tbody>
</table>
Table 4.4. A Comparison of the Distributed Mid-Range Thermal Solution and the Zero-Width End-Point Thermal Solution Via the Conjugate Gradient Algorithm

<table>
<thead>
<tr>
<th>Range (m)</th>
<th>Velocity (m/s)</th>
<th>Flight Path Angle (RAD)</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
<td>B</td>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>0</td>
<td>23.56</td>
<td>23.56</td>
<td>-0.02</td>
<td>-0.02</td>
</tr>
<tr>
<td>100</td>
<td>38.10</td>
<td>38.38</td>
<td>-0.41</td>
<td>-0.42</td>
</tr>
<tr>
<td>200</td>
<td>44.48</td>
<td>45.17</td>
<td>-0.20</td>
<td>-0.22</td>
</tr>
<tr>
<td>300</td>
<td>47.34</td>
<td>47.86</td>
<td>-0.15</td>
<td>-0.12</td>
</tr>
<tr>
<td>400</td>
<td>49.48</td>
<td>48.91</td>
<td>-0.13</td>
<td>-0.07</td>
</tr>
<tr>
<td>500</td>
<td>50.47</td>
<td>49.18</td>
<td>-0.04</td>
<td>-0.04</td>
</tr>
<tr>
<td>600</td>
<td>49.40</td>
<td>48.89</td>
<td>0.06</td>
<td>-0.01</td>
</tr>
<tr>
<td>700</td>
<td>47.20</td>
<td>47.81</td>
<td>0.08</td>
<td>0.04</td>
</tr>
<tr>
<td>800</td>
<td>44.35</td>
<td>45.11</td>
<td>0.14</td>
<td>0.15</td>
</tr>
<tr>
<td>900</td>
<td>38.10</td>
<td>38.41</td>
<td>0.35</td>
<td>0.37</td>
</tr>
<tr>
<td>1000</td>
<td>23.56</td>
<td>23.56</td>
<td>-0.02</td>
<td>-0.02</td>
</tr>
</tbody>
</table>

A = Distributed Mid-Range Thermal ($V_T = 1$ m/s, $R = 0.1$) plus Zero-Width End-Point Thermal ($V_T = 4$ m/s)

B = Zero-Width End-Point Thermal ($V_T = 4$ m/s)
Table 4.5. A Comparison of the Distributed Mid-Range Thermal Solution and the Zero-Width End-Point Thermal Solution Via the Conjugate Gradient Algorithm

<table>
<thead>
<tr>
<th>Range (m)</th>
<th>Altitude Loss (m)</th>
<th>Lift Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>100</td>
<td>47.87</td>
<td>48.95</td>
</tr>
<tr>
<td>200</td>
<td>77.93</td>
<td>81.11</td>
</tr>
<tr>
<td>300</td>
<td>94.74</td>
<td>97.43</td>
</tr>
<tr>
<td>400</td>
<td>108.95</td>
<td>106.42</td>
</tr>
<tr>
<td>500</td>
<td>117.94</td>
<td>111.65</td>
</tr>
<tr>
<td>600</td>
<td>116.52</td>
<td>114.06</td>
</tr>
<tr>
<td>700</td>
<td>109.37</td>
<td>112.53</td>
</tr>
<tr>
<td>800</td>
<td>99.46</td>
<td>103.32</td>
</tr>
<tr>
<td>900</td>
<td>76.35</td>
<td>78.02</td>
</tr>
<tr>
<td>1000</td>
<td>32.95</td>
<td>33.36</td>
</tr>
</tbody>
</table>

A = Distributed Mid-Range Thermal ($V_T = 1\text{ m/s}, R = 0.1$) plus Zero-Width End-Point Thermal ($V_T = 4\text{ m/s}$)

B = Zero-Width End-Point Thermal ($V_T = 4\text{ m/s}$)
Figure 4.1: Thermal Model, \( W(T) = \text{sum of thermals} \)

Figure 4.2: Altitude Loss Trajectory
\[ X_f = 1000 \text{m}, \ V_f = 2 \text{m/s} \]
**Figure 4.3: Optimal Control History**

X_f = 1000 m, V_f = 2 m/s

**Figure 4.4: Altitude Loss Trajectories**

X_f = 1000 m, R = 0.1
FIGURE 4.5: OPTIMAL CONTROL HISTORIES

$X_f = 1000 \text{m}, R = 0.1$
CHAPTER V. CONCLUSIONS AND SUGGESTIONS FOR FURTHER STUDY

The minimum time between thermals, dynamic sailplane problem has been investigated. The static problem related to this dynamic problem was used as a vehicle to aid in understanding the dynamic problem. This relationship between the static and dynamic problems formed the central idea behind Theorem 2.3 which proved that the MacCready values would yield a lower value for the performance index than the optimal solution does. This relationship is evident in the optimal trajectories calculated in Chapter II, and the suboptimal method based on matched asymptotic expansions which is explored in Chapter III is an attempt to exploit this relationship.

Both the conjugate gradient and quasilinearization algorithms were used to calculate optimal solutions for various values of the range between thermals. The conjugate gradient algorithm produced solutions, although costly, to all ranges, $x_f$. The quasilinearization algorithm produced solutions for short ranges, but failed on longer ranges because of the inherent sensitivity of the optimal control problem. A method for determining nominal
"guesses" for the missing initial costate values, $\lambda_V(0)$ and $\lambda_\gamma(0)$, was developed and is also based on the static MacCready trajectories. This procedure is explained in Chapter II and aids the quasilinearization algorithm in converging on solution values for a sensitive problem which would be difficult to solve with "poor" nominal values. The quasilinearization algorithm proved to be vastly less expensive than the conjugate gradient algorithm for those problems where it reached convergence.

All the methods used in this report attempt to find a solution which satisfies the necessary conditions for optimality (equations (2.16) - (2.21)). Of course, there is no guarantee that a solution found in this manner is a minimizing solution unless it is shown to also satisfy the sufficiency conditions (equations (2.28) - (2.30)). This was done in Chapter II for both the static and dynamic solutions.

Several suboptimal solutions were calculated in an effort to find a more economical approximation of the true solution. A linearized version of the two-point boundary value problem (equations (2.16) - (2.21)) was solved only for the smaller values of $x_f$ as this method suffers from the
same instability encountered in the quasilinearization algorithm. The linear/quadratic version of the optimal control problem is solved in Chapter III. This method produces interesting results because the performance index (equation (1.16)) does not depend on the optimal control, \( u \). When the linear/quadratic problem is derived (equations (3.11) and (3.12)) the Hamiltonian (equation (3.13)) is a linear function of the control, \( y = \delta u \). Therefore, the optimality condition, \( \frac{\delta H}{\delta y} = 0 \), can not be solved for \( y \).

This is a so-called "singular arc" problem which is solved in Chapter III. The final suboptimal technique is also described in Chapter III and is based on matched asymptotic expansions and the static MacCready trajectory.

A more realistic thermal model which allows for a distributed vertical wind profile is analyzed in Chapter IV. The results demonstrate that increasing the end-point thermal strengths causes a greater decrease in flight time than does increasing thermal radius. Several more extensions of the thermal model might deserve future analysis. One of these models would allow the initial and terminal thermals to have different values for both thermal...
strength and radius. One question that could then be answered would be whether it is faster to fly from a "weak" to a "strong" thermal than it is to fly from a "strong" to a "weak" thermal. Dolphin soaring is also investigated in Chapter IV. Further work is necessary in this area as well.

The sailplane dynamics might also be modeled differently. Circular flight in a thermal is the usual method of ascending in a sailplane. In this report an "elevator" motion was assumed in the thermal. Of course, in order to allow circular flight one must use three-dimensional equations of motion. The addition of rotational dynamics may be useful as well.

Besides changing the model for the sailplane or the thermal, the problem itself could be changed. There are several dynamic sailplane problems which deserve attention. Two of these problems are the maximum altitude launch and the short landing problem. The maximum altitude launch problem (see [22]) involves controlling the sailplane and a winch-launch mechanism in order to maximize altitude at a prescribed point in the launch process. The short landing problem attempts to minimize horizontal range
traveled while moving a sailplane from a specified initial state to a specified final state (with lower altitude).

One last area which deserves further consideration is that of designing sailplanes and sailplane instrumentation (see [23]) which will produce increased sailplane performance. An example of this is the design for a variable flap control presented in [24].
LITERATURE CITED


ACKNOWLEDGEMENTS

I would like to thank Dr. Bion L. Pierson and Dr. A. M. Fink for their assistance and guidance throughout my graduate work and especially during the preparation of this thesis. I would also like to thank the remainder of my graduate committee for their assistance in this project. Further thanks go to Mrs. Madonna Blanshan who typed this manuscript and Mrs. Rose Laugerman who prepared the figures. Special gratitude and appreciation goes to my wife, Toni, and sons, Larry and Frankie, whose willingness to live with less has made me more.
APPENDIX A. MACCREADY EQUILIBRIUM GLIDE TRAJECTORY

By setting the derivative terms in equations (1.17) and (1.18) equal to zero, the dimensionless equilibrium glide equations can be written as

\[ - \eta c_Dv^2 = \sin \gamma \quad (A.1) \]
\[ \eta c_Lv^2 = \cos \gamma \quad (A.2) \]

By squaring and adding, one obtains

\[ v = \left( \frac{1}{\eta^2 (c_D^2 + c_L^2)} \right)^{1/4} \quad (A.3) \]

The MacCready trajectory is obtained by minimizing the time of flight given by equation (1.3) as

\[ X_f \left( \frac{1}{V_x} + \frac{V_d}{V_x V_T} \right) = X_f \left( \frac{1}{V \cos \gamma} - \frac{\sin \gamma}{V T \cos \gamma} \right) \]

\[ = \frac{X_f}{V_T} \left( \frac{V_T - V \sin \gamma}{V \cos \gamma} \right) \]
This is equivalent to minimizing

\[ \frac{V_T - V \sin \gamma}{V \cos \gamma} \]  

(A.4)

Notice that this implies the MacCready values are range independent. Using equations (A.1) and (A.2) one obtains

\[ \frac{V_T - V \sin \gamma}{V \cos \gamma} = \frac{V_T}{V \cos \gamma} + \frac{C_D}{C_L} = \frac{V_T}{\eta C_L V^3} + \frac{C_D}{C_L} \]

\[ = \frac{V_T}{\eta C_L \left( \frac{1}{\eta^2 (C_D^2 + C_L^2)} \right)^{3/4}} + \frac{C_D}{C_L} \]

\[ = \frac{V_T \sqrt{\eta (C_D^2 + C_L^2)^{3/4}}}{C_L} + \frac{C_D}{C_L} \]  

(A.5)

Therefore, for minimum time,
\[
\frac{d}{dc_L} \left\{ \sqrt{\frac{V}{\pi}} \left[ \frac{1}{c_L} \left( \frac{2}{c_L^2 + c_D^2} \right)^{3/4} + c_D \right] \right\} = 0
\]

or

\[
0 = \sqrt{\frac{V}{\pi}} \left[ \frac{3}{2} \left( \frac{c_L^2 + c_D^2}{c_L} \right)^{3/4} - \frac{c_L^2}{c_D} \right] + \left( \frac{c_L}{c_D} - c_D \right) \left( c_L^2 + c_D^2 \right)^{1/4}
\]

which reduces to
Equation (A.7) is quite cumbersome but has been solved numerically by the method of false position by Bion Pierson (Aerospace Engineering Department, Iowa State University, unpublished notes, 1974).

The initial data used was

\begin{align*}
  &a_0 = 0.009278 \\
  &a_1 = -0.009652 \\
  &a_2 = 0.022288 \\
  &\eta = (9.81 \times X_f)/512.0 \text{ (with } X_f = 1000\text{m}, \eta = 19.160156) \\
\end{align*}

Various values of the parameter \( \frac{v_T}{\eta} \) produced the results displayed in Table A.1. To find the MacCready values for the costate variables, \( \lambda_V \) and \( \lambda_Y \), equations (2.18) and
(2.19) are set equal to zero and the MacCready values for the other variables are inserted. For \( V_T = 2 \text{m/s} \) this results in the nondimensional values \( \lambda_V = -20.679 \) and \( \lambda_Y = -0.032225 \) on the MacCready trajectory. A more detailed explanation of the MacCready trajectory is available in reference [21].

Table A.1. MacCready Values

<table>
<thead>
<tr>
<th>( V_T ) (m/s)</th>
<th>Lift Coefficient</th>
<th>Drag Coefficient</th>
<th>Velocity (m/s)</th>
<th>Angle (RAD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.403211</td>
<td>0.0090098</td>
<td>35.6299</td>
<td>-0.0223413</td>
</tr>
<tr>
<td>2</td>
<td>0.300417</td>
<td>0.0083898</td>
<td>41.2751</td>
<td>-0.0279202</td>
</tr>
<tr>
<td>3</td>
<td>0.243678</td>
<td>0.0082495</td>
<td>45.8250</td>
<td>-0.0338409</td>
</tr>
<tr>
<td>4</td>
<td>0.207411</td>
<td>0.0082349</td>
<td>49.6648</td>
<td>-0.0396824</td>
</tr>
<tr>
<td>5</td>
<td>0.182034</td>
<td>0.0082596</td>
<td>53.0073</td>
<td>-0.0453426</td>
</tr>
</tbody>
</table>
APPENDIX B.

FORTRAN PROGRAM LISTING AND

SAMPLE RUN FOR THE

QUASILINEARIZATION ALGORITHM
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION ST1(101),ST2(101),ST3(101),ST4(101),ST5(101),DEL1(101),
DEL2(101),DEL3(101),DEL4(101),F11(101),F12(101),F13(101),F14(101),
F1(101),F2(101),F3(101),F4(101),ALT(101),DFMEG(101),GFOR(101),
2F21(101),F22(101),F23(101),F24(101),F31(101),F32(101),F33(101),
2W(101),D(101),
JF34(101),F41(101),F42(101),F43(101),F44(101),R1(4),R2(4),R3(4),
4R4(4),P(3)
C THIS IS A QUASI-LINEARIZATION PROGRAM FOR THE MIN-TIME BETWEEN
C THERMALS PROBLEM. IT USES THE SOLUTION FROM THE PRECEDING PROBLEM
C AS STARTING GUESSES FOR THE MISSING COSTATE VALUES OF THE CURRENT ONE
READ(5,110) NINT,NCOCS,NSTOP,MIF
READ(5,120) AC,A1,A2,XF
READ(5,130) VZ,GAMMAZ,CVZ,CGZ
READ(5,140) VMAC,GMAC,UMAC,VT
READ(5,150) EPS,TEST,R
WRITE(6,150) NINT,NCOCS,NSTOP,MIF
WRITE(6,160) A0,A1,A2
WRITE(6,170) XF,VT
WRITE(6,180) VZ,GAMMAZ,CVZ,CGZ
WRITE(6,190) EPS,TEST,R
ETA=9.81D0*XF/512.00
VDIM=DSQRT(9.81D0*XF)
VZ=VZ/VDIM
VT=VT/VDIM
C=VT+VZ*DSIN(GAMMAZ)
H=1.00/DFLOAT(NINT)
H6=H/6.00
P(1)=0.5DC
P(2)=0.5DC
P(3)=1.0C
ITF=NINT+1
ISTOP=0

ALCS=H*XF*DSIN(GMAC)/DCOS(GMAC)

C COMPUTE VALUES FOR W(X) AND ITS DERIVATIVE (EOTH DIMENSIONLESS)

TM1R=1.00/R
TM2=-TM1R*TM1R

W(1)=C+C*DEXP(TM2)*(1.00+TM2)

DO 1 I=2,NINT

TR=DFLOAT(I-I)*H/R
TM1R=(1.00-DFLOAT(I-I)*H)/R
TR2=-TR*TR

W(I)=C*DEXP(TR2)*(1.00+TR2)+C*DEXP(TM2)*(1.00+TM2)

1 DO(I)=-C*DEXP(TM2)*2.00*TR*(2.00+TR2)/R+C*DEXP(TM2)*2.00

TR=1.00/R
TR2=-TR*TR

W(ITF)=C*DEXP(TR2)*(1.00+TR2)+C

DW(ITF)=-C*DEXP(TM2)*2.00*TR*(2.00+TR2)/R+C*DEXP(TM2)*2.00

C CALCULATE THE MACHREADY VALUES FOR THE COSTATES (NON-DIMENSIONAL)

VMAC=VMAC/VDIM

CVMAC=1.00*(2.00*DSIN(GMAC)+DCOS(GMAC))*(A1+2.00*A2*UMAC))

CGMAC=VMAC*CVMAC*(A1+2.00*A2*UMAC)

WRITE(6,180) VMAC,GMAC,CVMAC,CGMAC,UMAC
READ(5,100) (S1(I), I=1,MITF)
READ(5,100) (S2(I), I=1,MITF)
READ(5,1CC) (ST3(I),I=1,MITF)
READ(5,10C) (ST4(I),I=1,MITF)
DO 20 I=1,MITF

20 ST1(I)=ST1(I)/VDIM

IF(MITF.EQ.ITF) GO TO 28

MID=(MITF-1)/2
DO 22 I=1,MID
K=ITF-I+1
KM=MITF-I+1
ST1(K)=ST1(KM)
ST2(K)=ST2(KM)
ST3(K)=ST3(KM)

22 ST4(K)=ST4(KM)

MID1=MID+1
STR1=ST1(MID1)
STR2=ST2(MID1)
STR3=ST3(MID1)
STR4=ST4(MID1)

MID2=MID+2
IMID=ITF-MID
DO 24 I=MID2,IMID
ST1(I)=STR1
ST2(I)=STR2
ST3(I)=STR3

24 ST4(I)=STR4

28 DEL1(I)=VZ
DEL2(I)=GAMMAZ
DEL3(I)=ST3(I)
DEL4(I)=ST4(I)

C LINEARIZE ABOUT CURRENT NOMINAL TRAJECTORY
2 DO 25 I=1.ITF
78    T1=ST1(I)
79    T2=ST2(I)
80    T3=ST3(I)
81    T4=ST4(I)
82    T5=T4 /(2 * D0 * A2 + T3 * T1) - A1/(2 * D0 * A2)
83    T6=W(I)
84    T7=DW(I)
85    C0=A0 + A1 * T5 + A2 * T5**2
86    C0=DCOS(T2)
87    SI=DSIN(T2)
88    PUV=-T4/(2 * D0 * A2 * T3 * T1**2)
89    PUCV=-T4/(2 * D0 * A2 * T1 * T3**2)
90    PUCG=1 * D0 /(2 * D0 * A2 * T1 * T3)
91    PCDV=A1 * PUCV + 2 * D0 * A2 * PUV * T5
92    PCDVC=A1 * PUCV + 2 * D0 * A2 * PUCV * T5
93    PCDCG=A1 * PUCG + 2 * D0 * A2 * PUCG * T5
94    F11(I)=(-ETA*(CD+T1*PCDv)+SI/T1**2)/C0
95    F12(I)=-(ETA*CD*T1*SI+1*D0/T1)/C0**2-T7*C0
96    F13(I)=-ETA*T1*PCDCV/C0
97    F14(I)=-ETA*T1*PCDCG/C0
98    F21(I)=ETA*PUV/C0+2*D0/T1**3+T7*C0/T1**2
99    F22(I)=ETA*T5*SI/C0**2+T7*SI/T1
100   F23(I)=ETA*PUCV/C0
101   F24(I)=ETA*PUCG/C0
102   F31(I)=(-2*D0/T1**3+ETA*T3*PCDV+2*DC*T3*SI/T1**3)/C0+6*D0*T4/T1**4
1+2*D0*T6/(T1**3*VT*C0)+2*D0*T4*T7*C0/T1**3
103   F32(I)=(SI/T1**2+ETA*CD*T3*SI-T3/T1**2)/C0**2-T6*SI/(T1**2*CC**2
1*VT)+T4*T7*SI/T1**2
104   F33(I)=(ETA*CD-SI/T1**2+ETA*T3*PCDCV)/C0
F34(I) = ETA*T3*PC0CG/CC-2*D0/T1**3-T7*C0/T1**2
F41(I) = (SI/T1**2+ETA*CD*T3*SI+ETA*T3*T1*SI*PCDv-T3/T1**2-ETA*T4*
1SI*PV)/CO**2-T6*SI/(T1**2*VT*CO**2)+T4*T7*SI/T1**2
F42(I) = (-VT-VT*SI**2+2*D0*T1*SI)/(VT*T1*CO**3)+ETA*CD*T1*T3*(1.0C+
1SI**2)/CO**3+2*D0*T3*SI/(T1*CO**3)-ETA*T4*T5*(1.0C+S1**2)/CO**3
2+T6*(SI**2+1*D0)/(T1*VT*CO**3)-T3*T7*SI-T4*T7*CC/T1
F43(I) = (ETA*CO*T1*SI+1*D0/T1+ETA*T3*T1*SI*PCDv-ETA*T4*SI*PV)/
1CO**2+T7*CO
F44(I) = ETA*SI*(T3*T1*PCDCG-T5-T4*PVCG)/CO**2-T7*SI/T1
F1(I) = -(ETA*CD*T1**2+SI)/(T1*CO)-T7*SI
F2(I) = (ETA*T5*T1**2-C0)/(T1**2*CC)-T7*CO/T1
F3(I) = (1.0C+T3*(ETA*CD*T1**2-SI))/(T1**2*C0)-2*D0*T4/T1**3
1-T6/(T1**2*VT*CO)-T4*T7*CC/T1**2
25 F4(I) = (T1-VT*SI)/(VT*T1*CO**2)+T3*(ETA*CD*T1**2*SI+1.0C)/(T1*CC**2
1)-T4*ETA*T5*SI/CO**2+T6*SI/(T1*VT*CO**2)+T3*T7*CO-T4*T7*SI/T1
1CCCS=0
114 I=1
115 C perform forward integration
116 5 IF8=1
117 1TI=H
118 1T1=ST1(I)
119 1T2=ST2(I)
120 1T3=ST3(I)
121 1T4=ST4(I)
122 RHS1=F1(I)
123 RHS2=F2(I)
124 RHS3=F3(I)
125 RHS4=F4(I)
126 F1V =F11(I)
127 F16 =F12(I)
128  \[ F_{1CV} = F_{13}(I) \]
129  \[ F_{1CG} = F_{14}(I) \]
130  \[ F_{2V} = F_{21}(I) \]
131  \[ F_{2G} = F_{22}(I) \]
132  \[ F_{2CV} = F_{23}(I) \]
133  \[ F_{2CG} = F_{24}(I) \]
134  \[ F_{3V} = F_{31}(I) \]
135  \[ F_{3G} = F_{32}(I) \]
136  \[ F_{3CV} = F_{33}(I) \]
137  \[ F_{3CG} = F_{34}(I) \]
138  \[ F_{4V} = F_{41}(I) \]
139  \[ F_{4G} = F_{42}(I) \]
140  \[ F_{4CV} = F_{43}(I) \]
141  \[ F_{4CG} = F_{44}(I) \]
142  \[ J = 1 \]
143  \[ X_1 = \text{DEL}_1(I) \]
144  \[ X_2 = \text{DEL}_2(I) \]
145  \[ X_3 = \text{DEL}_3(I) \]
146  \[ X_4 = \text{DEL}_4(I) \]
147  \[ S_{X1} = X_1 \]
148  \[ S_{X2} = X_2 \]
149  \[ S_{X3} = X_3 \]
150  \[ S_{X4} = X_4 \]
151  \[ R_1(J) = \text{RHS}_1 + F_{1V}(X_1-T_1) + F_{1G}(X_2-T_2) + F_{1CV}(X_3-T_3) + F_{1CG}(X_4-T_4) \]
152  \[ R_2(J) = \text{RHS}_2 + F_{2V}(X_1-T_1) + F_{2G}(X_2-T_2) + F_{2CV}(X_3-T_3) + F_{2CG}(X_4-T_4) \]
153  \[ R_3(J) = \text{RHS}_3 + F_{3V}(X_1-T_1) + F_{3G}(X_2-T_2) + F_{3CV}(X_3-T_3) + F_{3CG}(X_4-T_4) \]
154  \[ R_4(J) = \text{RHS}_4 + F_{4V}(X_1-T_1) + F_{4G}(X_2-T_2) + F_{4CV}(X_3-T_3) + F_{4CG}(X_4-T_4) \]
155  \[ \text{GO TO} (83, 30, 84, 95), J \]
156  \[ \text{DEL}_1(IP1) = S_{X1} + H_6(R_1(1) + 2*0_0*R_1(2) + 2*0_0*R_1(3) + R_1(4)) \]
157  \[ \text{DEL}_2(IP1) = S_{X2} + H_6(R_2(1) + 2*0_0*R_2(2) + 2*0_0*R_2(3) + R_2(4)) \]
DEL3(IPL) = SX3 + H6*(R3(1) + 2*D0*R3(2) + 2*DC*R3(3) + R3(4))
DEL4(IPL) = SX4 + H6*(R4(1) + 2*D0*R4(2) + 2*DC*R4(3) + R4(4))

I = I + 1
IF (I .LT. IF) GO TO 15
C ENSATISFACTION ENDS ONE QUASI-LINEARIZATION STEP
IF (DABS(DEL1(IF) - DEL1(I)) .LT. EPS) GO TO 30

CCS = CCS + 1
IF (CCS.GT.NCCS) GO TO 270
GO TO 220
C TEST SatisFaction ENDS ENTIRE PROBLEM
PSG = 0.0
DO 301 I = 1, ITF
PSG = PSG + (ST1(I) - DEL1(I))**2 + (ST2(I) - DEL2(I))**2
ST1(I) = DEL1(I)
ST2(I) = DEL2(I)
ST3(I) = DEL3(I)
ST4(I) = DEL4(I)
STOP = STOP + 1
WRITE(6, 210) STOP
WRITE(6, 215) PSG
IF (PSG .GT. TEST AND STOP .LT. NSTOP) GO TO 2
C CALCULATE THE PERFORMANCE INDEX
F = -(VT - W(1) - ST1(1) * DSIN(ST2(1))) / (VT * ST1(1) * DCOS(ST2(1)))
DO 320 K = 2, NINT
F = F + 2*D0*(VT - W(K) - ST1(K) * DSIN(ST2(K))) / (VT * DCOS(ST2(K)) * ST1(K))
F = H*(F + (VT - W(IF) - ST1(IF) * DSIN(ST2(IF))) / (VT * ST1(IF) * DCOS(ST2(IF)))) / 2.0
PERF = F * DSQRT(XF / 9.81)
ST5(I) = (ST4(I) / (ST1(I)*ST3(I)) - A1) / (2.0*D0*A2)
THE ADJOINT METHOD IS NOW USED TO UPDATE THE MISSING INITIAL VALUES

183  \text{GFOR}(I) := (\text{ETA} \times \text{ST5}(I) \times \text{ST1}(I) \times 2 - \text{DCCS}(\text{ST2}(I))) / (\text{ST1}(I) \times \text{DCOS}(\text{ST2}(I)))

184  \text{DBMEGT}(I) := 0.00

185  \text{ALT}(I) := 0.00

186  \text{OO} 335  I = 2 \times \text{ITF}

187  \text{ST5}(I) := (\text{ST4}(I) / (\text{ST1}(I) \times \text{ST3}(I)) - \text{A1}) / (2.0 \times \text{A2})

188  \text{ALT}(I) := \text{ALT}(I - 1) - H \times \text{XF} \times (\text{DSIN}(\text{ST2}(I - 1)) / \text{DCOS}(\text{ST2}(I - 1)))^2

189  \text{DBMEGT}(I) := \text{ALOS} \times \text{DFLOAT}(I - 1) \times \text{ALT}(I)

190  \text{GFOR}(I) := \text{ETA} \times \text{ST5}(I) \times \text{ST1}(I) \times 2 - \text{DCCS}(\text{ST2}(I))

191  335  \text{ST1}(I) := \text{ST1}(I) \times \text{VDIM}

192  \text{ST1}(I) := \text{ST1}(I) \times \text{VDIM}

193  \text{WRITE}(6,400) \text{PERF}

194  \text{WRITE}(6,420) (K, \text{ST1}(K), \text{ST2}(K), \text{ALT}(K), \text{DBMEGT}(K), \text{ST3}(K), \text{ST4}(K),

195  \text{GFOR}(K), \text{ST5}(K), K = 1, \text{ITF})

196  \text{WRITE}(7,100) (\text{ST1}(I), I = 1, \text{ITF})

197  \text{WRITE}(7,100) (\text{ST2}(I), I = 1, \text{ITF})

198  \text{WRITE}(7,100) (\text{ST3}(I), I = 1, \text{ITF})

199  \text{WRITE}(7,100) (\text{ST4}(I), I = 1, \text{ITF})

200  \text{GO TO} 410

201  270  \text{WRITE}(6,280) \text{ITF}

202  \text{WRITE}(6,440) \text{ITF}, \text{DEL1}(\text{ITF}), \text{DEL2}(\text{ITF}), \text{DEL3}(\text{ITF}), \text{DEL4}(\text{ITF})

203  \text{GO TO} 410

THE ADJOINT METHOD IS NOW USED TO UPDATE THE MISSING INITIAL VALUES

204  \text{IFB} = 2

205  \text{WRITE}(6,440) \text{ITF}, \text{DEL1}(\text{ITF}), \text{DEL2}(\text{ITF}), \text{DEL3}(\text{ITF}), \text{DEL4}(\text{ITF})

206  \text{T1} := -1

207  \text{Z3} := \text{VZ} - \text{DEL1}(\text{ITF})

208  \text{Z4} := \text{GAMMAZ} - \text{DEL2}(\text{ITF})

209  \text{IAD} = 3
128

210  80  X3=0.00
211     X4=0.00
212     IF(IAC.EQ.4) GO TO 50
213     X1=1.00
214     X2=0.00
215     GO TO 60
216  50  X1=0.00
217     X2=1.00
218  60  K=1 TF
219     F1V=F11(K)
220     F1G=F12(K)
221     F1CV=F13(K)
222     F1CG=F14(K)
223     F2V=F21(K)
224     F2G=F22(K)
225     F2CV=F23(K)
226     F2CG=F24(K)
227     F3V=F31(K)
228     F3G=F32(K)
229     F3CV=F33(K)
230     F3CG=F34(K)
231     F4V=F41(K)
232     F4G=F42(K)
233     F4CV=F43(K)
234     F4CG=F44(K)
235  40  J=1
236     SX1=X1
237     SX2=X2
238     SX3=X3
239     SX4=X4
70 RI(J) = -F1V*X1 - F2V*X2 - F3V*X3 - F4V*X4
241 R2(J) = -F1G*X1 - F2G*X2 - F3G*X3 - F4G*X4
242 R3(J) = -F1CV*X1 - F2CV*X2 - F3CV*X3 - F4CV*X4
243 R4(J) = -F1CG*X1 - F2CG*X2 - F3CG*X3 - F4CG*X4
244 GO TO (93, 30, 94, 95), J
245 X1 = SX1 + HM6*(R1(1) + 2.D0*R1(2) + 2.D0*R1(3) + R1(4))
246 X2 = SX2 + HM6*(R2(1) + 2.D0*R2(2) + 2.D0*R2(3) + R2(4))
247 X3 = SX3 + HM6*(R3(1) + 2.DC*R3(2) + 2.D0*R3(3) + R3(4))
248 X4 = SX4 + HM6*(R4(1) + 2.D0*R4(2) + 2.D0*R4(3) + R4(4))
249 K=K-1
250 IF(K.GT.1) GO TO 40
251 IF(IAC.EQ.4) GO TO 90
252 XAD33 = X3
253 XAD34 = X4
254 IAC=4
255 GO TO 80
256 XAD43 = X3
257 XAD44 = X4
258 DELCO = XAD44*XAD33 - XAD34*XAD43
259 CORCV = (XAD44*X3 - XAD34*X4) / DELCO
260 CORCG = (-XAD43*X3 + XAD33*X4) / DELCO
261 DEL3(1) = DEL3(1) + CORCV
262 DEL4(1) = DEL4(1) + CORCG
263 I=1
264 GO TO 5
C LINEAR INTERPOLATION PACKAGE
265 83 IP1=I+1
266 F1V=(F1V+F1(1(IP1)))/2.D0
267 F1G=(F1G+F12(IP1))/2.D0
268 F1CV=(F1CV+F13(IP1))/2.D0
269 \( F_{1CG} = (F_{1CG} + F_{14}(IP_1))/2.00 \)
270 \( F_{2V} = (F_{2V} + F_{21}(IP_1))/2.00 \)
271 \( F_{2G} = (F_{2G} + F_{22}(IP_1))/2.00 \)
272 \( F_{2CV} = (F_{2CV} + F_{23}(IP_1))/2.00 \)
273 \( F_{2CG} = (F_{2CG} + F_{24}(IP_1))/2.00 \)
274 \( F_{3V} = (F_{3V} + F_{31}(IP_1))/2.00 \)
275 \( F_{3G} = (F_{3G} + F_{32}(IP_1))/2.00 \)
276 \( F_{3CV} = (F_{3CV} + F_{33}(IP_1))/2.00 \)
277 \( F_{3CG} = (F_{3CG} + F_{34}(IP_1))/2.00 \)
278 \( F_{4V} = (F_{4V} + F_{41}(IP_1))/2.00 \)
279 \( F_{4G} = (F_{4G} + F_{42}(IP_1))/2.00 \)
280 \( F_{4CV} = (F_{4CV} + F_{43}(IP_1))/2.00 \)
281 \( F_{4CG} = (F_{4CG} + F_{44}(IP_1))/2.00 \)
282 \( RHS_1 = (RHS_1 + F_1(IP_1))/2.00 \)
283 \( RHS_2 = (RHS_2 + F_2(IP_1))/2.00 \)
284 \( RHS_3 = (RHS_3 + F_3(IP_1))/2.00 \)
285 \( RHS_4 = (RHS_4 + F_4(IP_1))/2.00 \)
286 \( T_1 = (T_1 + ST_1(IP_1))/2.00 \)
287 \( T_2 = (T_2 + ST_2(IP_1))/2.00 \)
288 \( T_3 = (T_3 + ST_3(IP_1))/2.00 \)
289 \( T_4 = (T_4 + ST_4(IP_1))/2.00 \)
290 \( GO T_4 30 \)
291 \( 84 \ F_1V = F_{11}(IP_1) \)
292 \( F_1G = F_{12}(IP_1) \)
293 \( F_1CV = F_{13}(IP_1) \)
294 \( F_1CG = F_{14}(IP_1) \)
295 \( F_2V = F_{21}(IP_1) \)
296 \( F_2G = F_{22}(IP_1) \)
297 \( F_2CV = F_{23}(IP_1) \)
298 \( F_2CG = F_{24}(IP_1) \)
F3V = F31(IP1)
F3G = F32(IP1)
F3CV = F33(IP1)
F3CG = F34(IP1)
F4V = F41(IP1)
F4G = F42(IP1)
F4CV = F43(IP1)
F4CG = F44(IP1)
RHS1 = F1(IP1)
RHS2 = F2(IP1)
RHS3 = F3(IP1)
RHS4 = F4(IP1)
T1 = ST1(IP1)
T2 = ST2(IP1)
T3 = ST3(IP1)
T4 = ST4(IP1)
GO TO 30

K1 = K - 1

F1V = (F1V + F11(KM1))/2.0
F1G = (F1G + F12(KM1))/2.0
F1CV = (F1CV + F13(KM1))/2.0
F1CG = (F1CG + F14(KM1))/2.0
F2V = (F2V + F21(KM1))/2.0
F2G = (F2G + F22(KM1))/2.0
F2CV = (F2CV + F23(KM1))/2.0
F2CG = (F2CG + F24(KM1))/2.0
F3V = (F3V + F31(KM1))/2.0
F3G = (F3G + F32(KM1))/2.0
F3CV = (F3CV + F33(KM1))/2.0
F3CG = (F3CG + F34(KM1))/2.0
F4V = (F4V + F41(KM1))/2.D0
F4G = (F4G + F42(KM1))/2.D0
F4CV = (F4CV + F43(KM1))/2.D0
F4CG = (F4CG + F44(KM1))/2.D0
GO TO 30
F1V = F11(KM1)
F1G = F12(KM1)
F1CV = F13(KM1)
F1CG = F14(KM1)
F2V = F21(KM1)
F2G = F22(KM1)
F2CV = F23(KM1)
F2CG = F24(KM1)
F3V = F31(KM1)
F3G = F32(KM1)
F3CV = F33(KM1)
F3CG = F34(KM1)
F4V = F41(KM1)
F4G = F42(KM1)
F4CV = F43(KM1)
F4CG = F44(KM1)

C STANDARD FOURTH ORDER RUNGE KUTTA INTEGRATION: IFB=1 MEANS FORWARD
C INTEGRATION. IFB=2 MEANS BACKWARD (ADJOINT) INTEGRATION

CRK = P(J)
X1 = SX1 + CRK * T1 * R1(J)
X2 = SX2 + CRK * T1 * R2(J)
X3 = SX3 + CRK * T1 * R3(J)
X4 = SX4 + CRK * T1 * R4(J)
J = J + 1
GO TO (10,70), IF3
410 CONTINUE
STOP
100 FORMAT (4, 2, 13)
110 FORMAT (4I5)
130 FORMAT (1H, 5H, A0=, D13.5, 5H, A1=, D13.5, 5H, A2=, D13.5)
140 FORMAT (1H, 6H, XF=, D13.5, 5H, VT=, D13.5)
150 FORMAT (1H, 25HNO. OF INTEGRATION STEPS=, I5, 29H ALLOWABLE NO. OF QUASI-LINEARIZATIONS=, I5
18H MITF =, I5)
180 FORMAT (1H, 24H MACREADY VALUES ARE V=, D13.5, 4H, G=, D13.5, 5H, CV=, D13.5, 5H, CG=, D13.5)
190 FORMAT (1H, 23H INITIAL VALUES ARE V=, D13.5, 4H, G=, D13.5, 5H, CV=, D13.5, 5H, CG=, D13.5)
200 FORMAT (1H, 8H EPS =, D13.5, 9H, TEST =, D13.5, 6H, R =, D13.5)
210 FORMAT (1H, 10H ISTOP =, I5)
215 FORMAT (1H, 8H PSQ =, D13.5)
280 FORMAT (1H, 21H TOC MANY SHOTS ITF=, I5)
400 FORMAT (1H, 9H F =, D15.6)
420 FORMAT (6H, INDEX, 9X, 1H, V, 12X, 5H, GAMMA, 6X, 13H, ALTITUDE LOSS, 7X, 6H, MEGT
18X, 7H, LAMBDAA1, 8X, 7H, LAMBDAA2, 9X, 5H, GFCR, 12X, 2H, CL/1H, 12X, 5H, (M/S), 26X
23H(M), 14X, 3H, (M)/(1H, 15, 1X, 8D15.6))
440 FORMAT (1H, 34H CURRENT TERMINAL VALUES ARE ITF=, I5, 5H, V=, D13.5, 5H, G=, D13.5, 6H, CV=, D13.5, 6H, CG=, D13.5)
END
<table>
<thead>
<tr>
<th>INDEX</th>
<th>X</th>
<th>GAMMA</th>
<th>ALTITUDE LOSS</th>
<th>EMEGNT</th>
<th>LAMBDA1</th>
<th>LAMBDA2</th>
<th>GFDR</th>
<th>CL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>5.355550</td>
<td>0.00050000</td>
<td>0.1272400</td>
<td>0.1101500</td>
<td>0.3589500</td>
<td>0.1349300</td>
<td>0.00</td>
<td></td>
</tr>
</tbody>
</table>

**Table Note:**

- **X** represents the index.
- **GAMMA** is the gamma value.
- **ALTITUDE LOSS** is the altitude loss.
- **EMEGNT** is the EMEGNT value.
- **LAMBDA1** and **LAMBDA2** are lambda values.
- **GFDR** is the GFDR value.
- **CL** is the classification.

**Additional Details:**

- **Initial Values: V = 3.233550, G = -0.209620, C = -0.128500, D = 0.476520**
APPENDIX C.

FORTRAN PROGRAM LISTING AND SAMPLE RUN FOR THE LINEAR/QUADRATIC ALGORITHM
C LINEAR/QUADRATIC SINGULAR ARC CALCULATIONS FOR THE MIN-TIME SAILPLANE
C --MVBS II-- WITH UNBOUNDED CONTROL
1  IMPLICIT REAL*(A-H,O-Z)
2  DIMENSION VEL(101),GAM(101),CL(101),ALT(101),DBMEGT(101),GFOR(101)
3  READ (5,1CC) ETA,U,V,VT
4  READ (5,1CC) AU0,AU1,AU2,XF
5  READ (5,1CC) V2,GAMMAZ
6  READ (5,1CC) NINT
7  WRITE (6,105) ETA,U,V,VT
8  WRITE (6,106) AU0,AU1,AU2,XF
9  WRITE (6,107) V2,GAMMAZ
10  WRITE (6,107) NINT
11  S=-C,0279202D 0)
12  ITF=NINT+1
13  H=1,0/DFLOAT(NINT)
14  AL=S/H*XF*DSIN(G)/DCOS(G)
15  ETA=9,81*XF/ETA
16  VDIM=DSQRT(XF*9.81D0)
17  V=V/VDIM
18  VT=VT/VDIM
19  VZ=V?/VDIM
20  PMAC=(VT-V*DSIN(G))/(VT*V*DCOS(G))
21  CD=AU0+AU1*U+AU2*U**3
22  AI=-1.00/(ETA*U*V**4)
23  A2=-(1.00+VT*ETA*CD*V)/(VT*ETA**2*U**2*V**4)
24  C11=2.00/(ETA*U*V**5)
25  C12= CD/(U**2*V**4)
26  C21=C12
27  C22=(VT+VT*ETA**2*CD**2*V**4+2.00*ETA*CD*V**3)/(VT*ETA**3
1*U**3)*V**7)
\[
2\text{CONDET}^*A_1 + ((-D_{12}E_2^2 + D_{22}E_1E_2)/\text{CONDET} - 1)D_0)A_1 + \\
1(-D_{11}E_2 + D_{21}E_1E_2)/\text{CONDET}A_2 + E_2\text{DE}_2^*((-D_{12}E_2E_1 + D_{22}E_1^2)^
2\text{CONDET}^*A_1 + ((-D_{12}E_2^2 + D_{22}E_1E_2)/\text{CONDET} - 1)D_0)A_2)/ECE
\]

\[
D_1 = S_{11}S_{22} - S_{12}S_{21}
\]

\[
\text{WRITE} (6,120) S_{11}, S_{12}, S_{21}, S_{22}, D_1 \text{ DETS}
\]

\[
\text{WRITE} (6,130) F_1, F_2
\]

C ASYMPTOTIC STABILITY OF THE UNCONTROLLED SYSTEM ALONG WITH
C CONTROLLABILITY CHECK TELLS US THAT S INVERSE EXISTS. THAT IS,
C THE PARTICULAR NON-HOMOGENEOUS SOLUTION X = -S INVERSE*F

\[
X_1 = (-S_{22}F_1 - S_{12}F_2)/D_1 \text{ DETS}
\]

\[
X_2 = (-S_{21}F_1 + S_{11}F_2)/D_1 \text{ DETS}
\]

C COMPUTE E. VALUES AND E. VECTORS OF S (ONLY FOR REAL E. VALUES)

\[
\text{DISC} = (-S_{11} - S_{22})^2 - 4C_0D_0D_1 \text{ DETS}
\]

\[
A_1 = ? D_0
\]

\[
A_2 = 0 D_0
\]

\[
\text{REAL}1 = (S_{11} + S_{22} + \sqrt{\text{DISC}})/2D_0
\]

\[
\text{REAL}2 = (S_{11} + S_{22} - \sqrt{\text{DISC}})/2D_0
\]

\[
\text{WRITE} (6,140) \text{REAL}1, A_1, \text{REAL}2, A_2
\]

\[
\text{EVEC}R1 = (\text{REAL}1 - S_{11})/S_{12}
\]

\[
\text{EVEC}R2 = (\text{REAL}2 - S_{11})/S_{12}
\]

\[
\text{WRITE} (6,150) \text{EVEC}R1, \text{EVEC}R2
\]

C NOW SOLVE FOR X1(0+) AND X2(0+)

\[
\text{START} = E_2*(V2-V) - E_1*(\text{GAMMA}2 - \text{G})
\]

\[
P_{11} = E_2
\]

\[
P_{12} = -E_1
\]

\[
\text{DEL}1 = D_0/(\text{EVEC}R2 - \text{EVEC}R1)
\]

\[
\text{EX}1 = \text{EXP}(\text{REAL}1)
\]

\[
\text{EX}2 = \text{EXP}(\text{REAL}2)
\]

\[
P_{21} = \text{DEL1*EX}1*\text{EVEC}R2*(E_2 - E_1*\text{EVEC}R1) + \text{EX}2*\text{DEL1*EVEC}R1*(E_1*\text{EVEC}R2 - E_2)
\]
P22=DEL*EX1*(E1*EVECR1-E2)+EX2*DEL*(E2-E1*EVECR2)
91=DEL*EX1*EVECR2*(E1*EVECR1-E2)+DEL*EX2*EVECR1*(E2-E1*EVECR2)+E2
32=DEL*EX1*(E2-E1*EVECR1)+DEL*EX2*(E1*EVECR2-E2)-E1
DETP=P11*F22-P12*P21
WRITE (6,170) DETP
X1=(START*P22-D12*(START-B1*XP1-B2*XP2))/DETP
C1=DEL*(EVECR2*(X10-XP1)-(X20-XP2))
C2=DEL*(EVECR1*(XP1-X10)+(X20-XP2))
WRITE (6,120) XP1,XP2,C1,C2
VEL(1)=X1:
SAV(1)=X2:
CL(1)=((REAL1*C1+REAL2*C2-D11*X10-D12*X20)*E1
DO 5) I=2,IT
T=**FLCAT(I-1)
D1=DEXP(REAL1*T)
D2=DEXP(REAL2*T)
VEL(1)=C1*D1+C2*D2+XP1
GAY(I)=C1*EVECR1*D1+XP2+D2*C2*EVECR2
50 CL(1)=(REAL1*C1*D1-D11*VEL(I)-D12*GAM(I)+D2*C2*REAL2)/E1
PEQP=(DEXP(REAL1)-1.00)*(A1*C1+A2*C1*EVECR1+2.00*C1*XP1*C11+
12*D0*C1*EVECR1*XP2*C22+C1*XP2*2.00*C12*C1*XP1*EVECR1*2.00*C12)/
2.00*C12+DEXP(REAL1)-1.00)*(A1*C2+A2*EVECR2*2.00*C22*XP1+C11+
12*D0*C2*EVECR2*XP2+C22*XP2*2.00*C12+C2*XP1*EVECR2*2.00*C12)/
4.00*9+((DEXP(2.00*REAL1)-1.00)*(C1**2+C11+C1**2*EVECR1*2.00*C22+9
SC1**2*EVECR1*2.00*C12)/(2.00*REAL1)+(DEXP(2.00*REAL2)-1.00)*(C2**2+
5*C11+C2**2*EVECR2**2+C22+C2**2*EVECR2**2*00*C12)/(2.00*REAL2)+
7(DDEXP(REAL1+REAL2)-1.00)*(2.00*C1*C2*C11+2.00*C1*C2*EVECR1*
8*EVECR2+C2+C1*C2*EVECR2**2.00*C12+C12+C2*C1*EVECR1*2.00*C12)/(REAL1+
9REAL2)+A1*XP1+A2*XP2+C11*XP1**2+C22*XP2**2+2.00*C12*XP1*XP2+PMAC
PERF = PERF * CSQRT(XF/9.81D0)

WRITE (6, 240) PERF

DO 62 I = 1, ITF

VEL(I) = VEL(I) + V

GAM(I) = GAM(I) + G

62 CL(I) = CL(I) + U

GF3R(I) = (ETA * CL(I) * VEL(I) ** 2 - DCOS(GAM(I))) / (VEL(I) * DCOS(GAM(I)))

DBVEST(I) = C * DC

ALT(I) = 0.00

DO 335 I = 2, ITF

ALT(I) = ALT(I-1) - H*XF*(DSIN(GAM(I-1))/DCS(GAM(I-1)) + DSIN(GAM(I))/DCS(GAM(I)))/2.0

DBVEST(I) = ALTS*DFLOAT(I-1) + ALT(I)

GF3R(I) = ETA * CL(I) * VEL(I) ** 2 - DCOS(GAM(I))

335 VEL(I) = VEL(I) * VD1M

VEL(I) = VEL(I) * VD1M

4RIT(6, 260)(I, VEL(I) * GAM(I), ALT(I), CBVEST(I), GFOR(I), CL(I), I = 1, ITF)

100 FORMAT(4C2C.13)

101 FORMAT (IE)

102 FORMAT (1F, 7HINT = , I5)

103 FORMAT (1HC, 6HETA = , D20.13/1H, 4HU = , D20.13/1H, 4HV = , D20.13/1H, 5HVT = , D20.13)

104 FORMAT (1HC, 31HC, COEFFICIENTS IN DRAG POLAR ARE , 3D20.13/1H, 5HFXF = , 1D20.13)

105 FORMAT (1HC, 70HC, CONTROLLABILITY CHECK / DET = , D20.13)

106 FORMAT (1HC, 6HS1I = , D20.13, 1CH, 512 = , D20.13/1H, 6HS21 = , 1D20.13, 1CH, 522 = , D20.13/1H, 11H, DETS = , D20.13)

107 FORMAT (1HC, 5HF1 = , D20.13/10H, F2 = , D20.13)
114 140 FORMAT(1HC,16HEIGENVALUES ARE D2C.13,3H + D2C.13,1HI/1H, 
116H \text{ AND } D2C.13,3H + D2C.13,1HI)

115 170 FORMAT(1HC,7HDETP = D20.13)

116 220 FORMAT(1HC,43HSECOND COMPONENTS OF E.VECTORS ARE E.V.1 = D2C.13, 
117H \text{ AND E.V.2 = D20.13})

117 230 FORMAT(1HC,5HVZ = D20.13,12H GAMMAZ = D20.13)

118 240 FORMAT(1HC,29HVALUE OF PERFORMANCE INDEX = D2C.13)

119 260 FORMAT(6HCINDEX,9X,1HV,12X,5HGMMA,6X,13H-ALTITUDE LOSS,7X,6HDEMECT 
1,9X,5H GFCR,12X,?HCL/1H,12X,5H(M/S),26X,3H(M),14X,3H(M)//(1H , 
215,1X,6015.6))

120 STOP
121 END
**SENTRY**

<table>
<thead>
<tr>
<th>INDEX</th>
<th>V(M/S)</th>
<th>GAMMA</th>
<th>ALTITUDE LOSS (M)</th>
<th>CBMEXT</th>
<th>GFCR</th>
<th>CL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2363520</td>
<td>-0.5300180</td>
<td>0.0000000</td>
<td>0.0000000</td>
<td>-0.2589560</td>
<td>0.7419240</td>
</tr>
<tr>
<td>2</td>
<td>0.2482870</td>
<td>-0.4950540</td>
<td>0.5629200</td>
<td>0.5349920</td>
<td>-0.2274180</td>
<td>0.7119440</td>
</tr>
<tr>
<td>3</td>
<td>0.2593910</td>
<td>-0.4624760</td>
<td>0.1082140</td>
<td>0.1026280</td>
<td>0.3576880</td>
<td>0.6840460</td>
</tr>
<tr>
<td>4</td>
<td>0.2697210</td>
<td>-0.4321200</td>
<td>0.1562000</td>
<td>0.1478220</td>
<td>0.2698650</td>
<td>0.6580670</td>
</tr>
<tr>
<td>5</td>
<td>0.2793300</td>
<td>-0.4038330</td>
<td>0.2006250</td>
<td>0.1894540</td>
<td>0.4650800</td>
<td>0.6339350</td>
</tr>
<tr>
<td>90</td>
<td>0.3056487</td>
<td>22</td>
<td>0.107830</td>
<td>0C</td>
<td>0.5146920</td>
<td>C2</td>
</tr>
<tr>
<td>91</td>
<td>0.3006150</td>
<td>22</td>
<td>0.2012350</td>
<td>0C</td>
<td>0.4942350</td>
<td>C2</td>
</tr>
<tr>
<td>92</td>
<td>0.2557427</td>
<td>22</td>
<td>0.2121750</td>
<td>0C</td>
<td>0.4738640</td>
<td>C2</td>
</tr>
<tr>
<td>93</td>
<td>0.2893170</td>
<td>22</td>
<td>0.2236240</td>
<td>0C</td>
<td>0.4517220</td>
<td>C2</td>
</tr>
<tr>
<td>94</td>
<td>0.2242287</td>
<td>22</td>
<td>0.2356120</td>
<td>0C</td>
<td>0.4283470</td>
<td>C2</td>
</tr>
<tr>
<td>95</td>
<td>0.2779540</td>
<td>22</td>
<td>0.2451610</td>
<td>0C</td>
<td>0.4036740</td>
<td>C2</td>
</tr>
<tr>
<td>96</td>
<td>0.2715132</td>
<td>22</td>
<td>0.2512970</td>
<td>0C</td>
<td>0.3776350</td>
<td>C2</td>
</tr>
<tr>
<td>97</td>
<td>0.2649540</td>
<td>22</td>
<td>0.2755510</td>
<td>00</td>
<td>0.3501540</td>
<td>C2</td>
</tr>
<tr>
<td>98</td>
<td>0.2579273</td>
<td>22</td>
<td>0.2894540</td>
<td>0C</td>
<td>0.3311520</td>
<td>C2</td>
</tr>
<tr>
<td>99</td>
<td>0.2555763</td>
<td>22</td>
<td>0.3043270</td>
<td>00</td>
<td>0.2905470</td>
<td>C2</td>
</tr>
<tr>
<td>100</td>
<td>0.2431237</td>
<td>22</td>
<td>0.3263130</td>
<td>00</td>
<td>0.2692450</td>
<td>C2</td>
</tr>
<tr>
<td>101</td>
<td>0.2359140</td>
<td>22</td>
<td>0.3368420</td>
<td>00</td>
<td>0.2241490</td>
<td>C2</td>
</tr>
</tbody>
</table>
APPENDIX D. EQUATIONS OF MOTION FOR THREE-DIMENSIONAL FLIGHT IN THE PRESENCE OF A VERTICAL WIND

Referring to Figure D.1 the following variables are defined.

\[ \beta = \text{heading angle} \]
\[ \gamma = \text{Flight path angle (if there is no wind)} \]
\[ \dot{\mathbf{r}} = \text{inertial velocity vector of the sailplane} \]

\[ \mathbf{w} = \begin{bmatrix} 0 \\ 0 \\ w(x,y,z) \end{bmatrix} \]

\( w(x,y,z) \) - frame = inertial coordinate frame

The relative wind is given by

\[ \mathbf{V} = \mathbf{w} - \dot{\mathbf{r}} \]  \hspace{1cm} (D.1)

In the inertial coordinate frame, the translational equations of motion for unpowered flight in a uniform gravitational field can be written as (see [25])

\[ m\ddot{\mathbf{r}} = \mathbf{L} + \mathbf{D} + m\mathbf{g} = m(\dot{\mathbf{w}} - \dot{\mathbf{V}}) \]  \hspace{1cm} (D.2)
where \( L \) is the lift, \( D \) the drag, and \( mg \) the weight of the sailplane. Solving equation (D.2) for \( \vec{V} \), one obtains

\[
- \vec{V} = - \vec{\omega} + \frac{\vec{L}}{m} + \frac{\vec{D}}{m} + \vec{g}
\]

or

\[
- \vec{V} = \begin{bmatrix}
0 \\
0 \\
- \frac{\partial w}{\partial x} \vec{x} - \frac{\partial w}{\partial y} \vec{y} \\
\end{bmatrix} + \frac{\vec{L}}{m} + \frac{\vec{D}}{m} + \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix}
\]  

(D.4)

Primarily because of the definitions of lift and drag, it is desirable to write equation (D.4) with respect to a rotating coordinate frame ("trajectory axes") defined by the following unit vectors

\[
\hat{\xi} = - \frac{\vec{V}/||\vec{V}||}{||\vec{V}||} = - \frac{\vec{V}}{V}
\]

\[
\hat{\eta} = \hat{\xi} \times \hat{\xi}
\]

\[
\hat{\zeta} = \frac{\vec{L}/||\vec{L}||}{||\vec{L}||} \text{ for zero bank angle (} \emptyset = 0 \text{)}
\]
The inertial time derivative of the vector $-\vec{V}$ (as required by Newton's second law in equation (D.4)) with respect to the rotating frame is given by

$$\frac{d}{dt}(-\vec{V}) = (-\vec{V})\left(\hat{\xi}, \hat{\eta}, \hat{\zeta}\right) + \vec{w} \times (-\vec{V})$$

$$= -\ddot{\vec{V}} + (-\gamma \hat{\eta} + \beta \cos \gamma \hat{\zeta} + \beta \sin \gamma \hat{\xi}) \times (-\vec{V})$$

$$= -\ddot{\vec{V}} - \nu \hat{\xi} - \nu \beta \cos \gamma \hat{\eta} \quad \text{(D.6)}$$

Note that the rotating and inertial frames are related by the following rotation matrix.

$$\begin{bmatrix}
\cos \gamma & 0 & \sin \gamma \\
0 & 1 & 0 \\
-\sin \gamma & 0 & \cos \gamma
\end{bmatrix}
\begin{bmatrix}
\cos \beta & \sin \beta & 0 \\
-\sin \beta & \cos \beta & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
\xi \\
\eta \\
\zeta
\end{bmatrix}
= \begin{bmatrix}
\cos \gamma \cos \beta & \cos \gamma \sin \beta & \sin \gamma \\
-\sin \beta & \cos \beta & 0 \\
-\sin \gamma \cos \beta & -\sin \gamma \sin \beta & \cos \gamma
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
z
\end{bmatrix}
\quad \text{(D.7)}$$
Using equation (D.7) one obtains

\[ \dot{w} + g = - \left[ \frac{\partial w}{\partial x} x + \frac{\partial w}{\partial y} \dot{y} + g \right] \left[ \sin \gamma + \cos \gamma \right] \]  \hspace{1cm} (D.8)

Using equations (D.6) and (D.8), equation (D.4) can be written in scalar form as follows,

\[ - \dot{V} = - \left[ \frac{\partial w}{\partial x} x + \frac{\partial w}{\partial y} \dot{y} + g \right] \sin \gamma - \frac{\rho V^2 C_D s}{2m} \]  \hspace{1cm} (D.9)

\[ - V \dot{Y} = - \left[ \frac{\partial w}{\partial x} x + \frac{\partial w}{\partial y} \dot{y} + g \right] \cos \gamma + \frac{\rho V^2 C_L s}{2m} \cos \phi \]  \hspace{1cm} (D.10)

\[ - V \dot{\beta} \cos \gamma = \frac{\rho V^2 C_L s}{2m} \sin \phi \]  \hspace{1cm} (D.11)

where \( \phi \) denotes the bank angle.

From equation (D.1) and the inverse relationship of equation (D.7), the kinematic equations are given by

\[ \dot{x} = - V \cos \gamma \cos \beta \]  \hspace{1cm} (D.12)
\[
\dot{y} = - V \cos \gamma \sin \beta \quad (D.13)
\]
\[
\dot{z} = w - V \sin \gamma \quad (D.14)
\]

The following three steps are now performed.

(i) nondimensionalize with respect to a velocity unit 
\[(2mg/\rho s)^{1/2}\] and a time unit \[(2m/\rho sg)^{1/2}\]

(ii) using equations (D.12) and (D.13), substitute for \(\dot{x}\) and \(\dot{y}\) in equations (D.9) and (D.10)

(iii) replace \(- V\) with \(V\) (the true airspeed)

The equations of motion may now be written.

\[
\dot{V} = - \left( \frac{\partial w}{\partial x} \cos \beta + \frac{\partial w}{\partial y} \sin \beta \right) V \cos \gamma + l \right) \sin \gamma - V^2 C_D \quad (D.15)
\]

\[
\dot{\gamma} = - \left( \frac{\partial w}{\partial x} \cos \beta + \frac{\partial w}{\partial y} \sin \beta \right) \cos \gamma + \frac{l}{V} \right) \cos \gamma + V C_L \cos \phi \quad (D.16)
\]

\[
\dot{\beta} = V C_L \sin \phi / \cos \gamma \quad (D.17)
\]
\[ \dot{x} = V \cos \gamma \cos \beta \] (D.18)

\[ \dot{y} = V \cos \gamma \sin \beta \] (D.19)

\[ \dot{z} = w(x,y,z) + V \sin \gamma \] (D.20)

Once a drag polar, \( C_D = C_D(C_L) \), is chosen, the controls become lift coefficient, \( C_L \), and bank angle, \( \phi \).

For the case of planar flight, set \( \beta = \phi = 0 \) and \( w(x,y,z) \) reduces to \( w(x,z) \), to obtain

\[ \dot{V} = - \left( \frac{\partial w}{\partial x} V \cos \gamma + \frac{1}{V} \right) \sin \gamma - V^2 C_D \] (D.21)

\[ \dot{\gamma} = - \left( \frac{\partial w}{\partial x} \cos \gamma + \frac{1}{V} \right) \cos \gamma + V C_L \] (D.22)

\[ \dot{x} = V \cos \gamma \] (D.23)

\[ \dot{z} = w(x,z) + V \sin \gamma \] (D.24)
FIGURE D.1: THE VELOCITY DIAGRAM
APPENDIX E. THE NECESSARY CONDITIONS FOR
THE PROBLEM WITH UNSPECIFIED, BUT EQUAL,
BOUNDARY VALUES

The following theorem is taken from reference \[12\].
The notation used in that reference is that \( \mathcal{S} \) is the
class of arcs \( x: x^i(t), \ b^\sigma, \ u^k(t) \ (t^0 \leq t \leq t') \)

\((i = 1, \ldots, n; \ \sigma = 1, \ldots, m; \ k = 1, \ldots, q)\)

having its elements \((t, x(t), u(t))\) in a prescribed set \( \mathcal{P}_0 \)
in \( t \times u \)-space, having \( b \) in a given open set \( B \) and
satisfying conditions of the form

\[ \dot{x}^i = f^i(t, x, u) \quad (E.1) \]

\[ t^s = T^s(b), \ x^i(t^s) = X^i_s(b) \ (s = 0, 1) \quad (E.2) \]

\[ I_r(x) \leq 0 (1 \leq r \leq p'), I_r(x) = 0 (p' < r \leq p) \quad (E.3) \]

where
The problem considered is minimizing

\[ I_0(x) = g_0(b) + \int_{t_0}^{t'} L_0(t, x(t), u(t)) dt \]  

(E.5)

on the class \( \mathcal{G} \).

It is assumed that the function \( f^i \) and \( L_r \) are of class \( C' \) on a region \( \mathcal{R} \) of txu-space and that \( \mathcal{R}_0 \) is a subset of \( \mathcal{R} \). The functions \( T^s, X^i, g^o \) are of class \( C' \) in \( b = (b', \ldots, b^m) \) on the domains to be considered.

Assume a solution

\[ x_o: x_o(t), b_o, u_o(t), (t_0 \leq t \leq t') \]  

(E.6)

to the given problem. It is assumed that \( x_o \) possesses a program \( u(t, x) \) (see [12]). Setting
the first order necessary conditions for a minimum are as follows.

**Theorem E.1.**

Suppose $x_o$ is a solution to the given problem. Then there exist multipliers $\lambda_o \geq 0, \lambda_1, \ldots, \lambda_p, p_1(t), \ldots, p_n(t)$, not all vanishing simultaneously, and functions

$$H(t, x, u, p) = \sum_{i=1}^{n} p_i f_i^i - \sum_{r=0}^{P} \lambda_r L_r \text{, } G(b) = \sum_{r=0}^{P} \lambda_r g_r(b) \quad (E.8)$$

such that

(i) the inequality $\lambda_r \geq 0 (1 \leq r \leq p')$ holds with $\lambda_r = 0$ if $L_r(x_o) < 0$

(ii) the functions $p_i(t) (t^0 \leq t \leq t')$ are continuous and satisfy the differential equations

$$\dot{p}_i = - H \frac{\partial f_i^i}{\partial x_i} - H \frac{\partial f_i^i}{\partial u} \quad (E.9)$$

where

$$V_i^k(t) = \frac{\partial u_i^k(t, x_o(t))}{\partial x_i} \quad (E.7)$$
along \( x_o \), where \( v_i \) is given by (E.7)

(iii) the end values of \( x_o \) are such that the equation

\[
dG + \left[ - H^S dT^S + p_i(t^S)dx^iS \right]_{s=0}^{s=1} = 0 \tag{E.10}
\]

is an identity in \( db',...,db^m \) where

\[
H^S = H(t^S, x_o(t^S), u_o(t^S), p(t^S)) \quad (s = 0,1)
\]

(iv) the inequality

\[
H(t,x_o(t),u,p(t)) \leq H(t,x_o(t),u_o(t),p(t)) \text{ holds.} \tag{E.11}
\]

To apply this theorem to the problem given in Theorem 2.3, set

\[
x^{is}(b) = b \quad (s = 0,1), \quad (\text{where } b \text{ is unspecified}) \tag{E.11}
\]

\[
g_r(b) = 0 \quad (r = 0,1,...,p) \tag{E.12}
\]
\[ L_r(t,x(t),u(t)) = 0 \ (r = 1, \ldots, p) \]  
\[ (E.13) \]

and consider only those \( f^i \) and \( L_0 \) which are autonomous to obtain the problem

\[
\text{minimize} \quad I_0(x) = \int_{t_0}^{t_1} L_0(x(t),u(t))dt \\
\text{subject to} \quad x^i = f^i(x,u) \ (i = 1, \ldots, n) \\
\]  
\[ (E.14), (E.15) \]

with \[ x^i(t^s) = b \quad (s = 0,1) \]  
\[ (E.16) \]

Since \( H \) is autonomous, the optimal solution will produce a constant \( H \). Therefore, \( H^0 = H^1 \). Furthermore, since \( G = 0 \), then \( \frac{dG}{dt} = 0 \). Also, \( \frac{dX}{dt} = \frac{db}{dt} \) since \( x^i(b) = b \). Examining equation \((E.10)\), one obtains \( p_i(t^o) = p_i(t^\prime) \). In Theorem 2.3 this corresponds to \( \lambda(t^o) = \lambda(t_f) \).