OPTIMIZED SPACE SHUTTLE TRAJECTORY SIMULATION

Louis Tramonti, Senior Aeroballistics Engineer
Trajectory Simulation, Vehicle Software Systems and Trajectories

Richard G. Brusch, Design Specialist
Optimization Technology, Launch Vehicle Programs

Convair Aerospace Division of General Dynamics
San Diego, California

INTRODUCTION

Current and anticipated aerospace vehicles have many trajectory degrees-of-freedom that can be optimized to maximize or minimize a performance measure such as payload, range, or cost. This feature is particularly true for lifting reusable space shuttle configurations where the inclusion of lift adds additional operational modes such as lifting dogleg maneuvers during ascent, aerodynamically-controlled entry maneuvers, and synergetic skipping maneuvers. Many trajectory constraints must be considered such as maximum heating, acceleration, angle of attack, and excluded or included overflight or impact regions. To synthesize lifting shuttle vehicle trajectories, it is necessary to have the capability to optimize highly constrained trajectories for all phases of flight.

A computer program, the General Trajectory Optimization Program (GTOP), is described in this paper. The program can handle the ascent, return, and synergetic maneuvers of lifting boosters and spacecraft. This generality also means that the GTOP program can be used for a large variety of aerospace vehicles and missions.

TRAJECTORY STRUCTURE

Trajectory sectioning is a method of subdividing the time history of a trajectory simulation into parts relevant to the description of the simulation. The simulation sections are defined to allow for the following types of changes in the simulation models:

1. Changes in the state equations (the flight equations).
2. Changes in the control model (those functions and parameters that the user is free to manipulate).
3. Changes in the various trajectory constraints.
4. Changes in the terms of the performance measure.
5. Requirement for an analytic trajectory segment (e.g. a weight discontinuity caused by jettisoning an expended stage or a conic trajectory arc).

A section is defined as any segment of the trajectory in which the mathematical model is of a given form and the state variables $x_{ij}(t)$, during the numerically integrated part, are continuous functions of time. Section endpoints are chosen to coincide with points at which the differential equations, the control model, or the trajectory constraints change form; or at which the state variables experience a discontinuity. Analytic trajectory segments are used whenever a state variable discontinuity is modeled or whenever it is possible to propagate the trajectory analytically, such as use of a conic trajectory. Trajectory sectioning provides a skeletal framework that may be molded by users with widely varying problems to facilitate the description of their particular problem to a general mathematical model.

Each trajectory section may consist of an analytic part and a numerically integrated part. The numerically integrated part of a section is governed by the set of nonlinear differential equations of motion (the state equations*) which must be numerically integrated in order to propagate the trajectory. The state equations have the form shown in Equation (1) in the illustration.

The analytic part of a section is that part for which analytic integrals of the motion may be used to propagate the trajectory. The four analytic methods currently available are: 1) state variable discontinuity by a specified value, 2) state variable discontinuity to a specified value, 3) conic trajectory propagation through a specified time, and 4) conic trajectory propagation through a specified central angle.

The state equations (to be discussed later) define the position, velocity, and weight of the vehicle.

STATE EQUATIONS

$$x_{ij}(t) = f_{ij} \left[x_{ij}(t),\ u_{kj}(t),\ d_p,\ t \right] \qquad (1)$$

$i = 1, 2, \ldots n_i$ (state variable index)

$j = 1, 2, \ldots n_j$ (simulation section index)

$k = 1, 2, \ldots n_k$ (time varying control parameter index)

$p = 1, 2, \ldots n_p$ (time invariant design parameter index)

where: $u_{kj}(t)$ is the kth time varying* control variable of the jth simulation section.

d_p is the pth time invariant design parameter

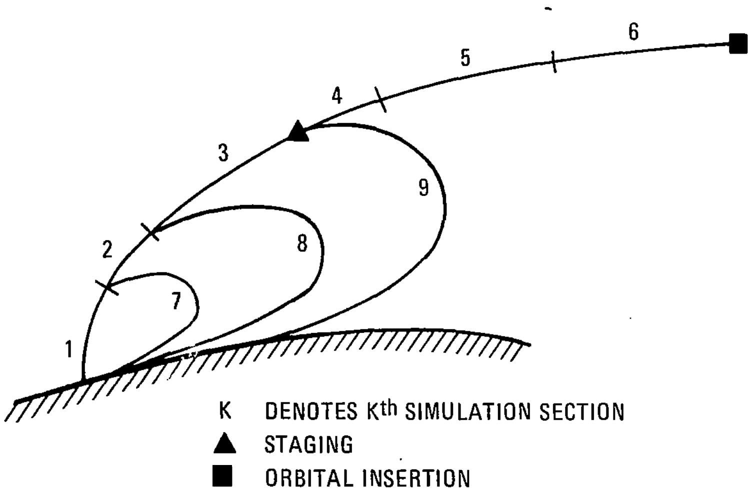
$x_{ij}(t)$ is the ith state variable of the jth simulation section

t is the time

**The procedure described here, and which is used in the GTOP Program, allows the use of any of a large number of parameters as the independent variable of a control function; however, these independent variables are themselves time varying; hence the representation $u_{ku}(t)$ is general*

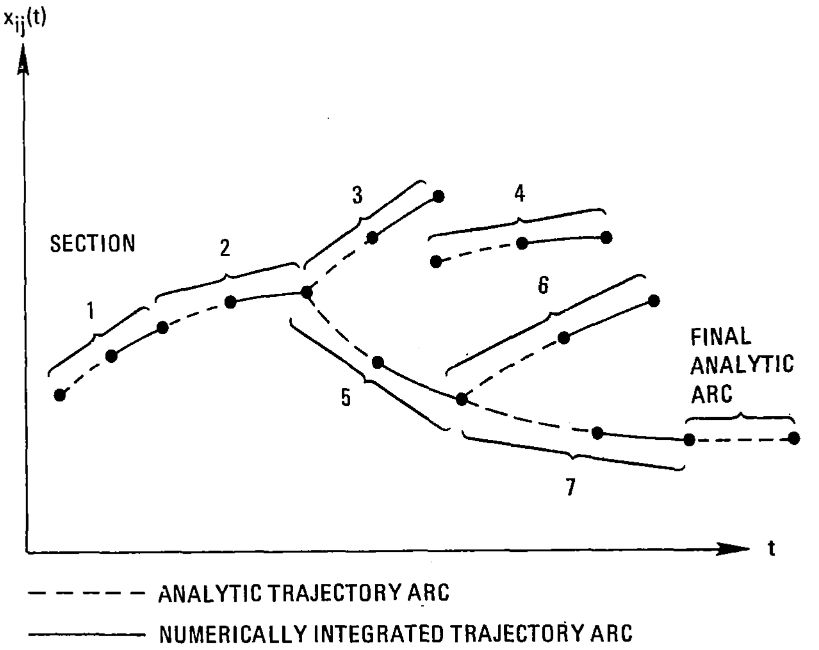
For some problems it is necessary to be able to simulate a trajectory which branches at one or more points. For example, the optimization of a space shuttle ascent trajectory in which constraints are imposed during the booster return flight and during possible abort return flights (left half of figure) requires the capability to simulate a trajectory with multiple branching points. After each ascent trajectory simulation, the normal booster return flight and hypothetical abort trajectories which can be initiated at the beginning or end of any ascent simulation section (as shown in left half of figure) are then simulated. To meet the return flight constraints, the ascent trajectory must be modified; thus the hypothetical abort and booster return trajectories directly influence the optimization of the ascent trajectory. A generalized branching capability is provided in the GTOP system. This procedure allows the initiation of any simulation section at either the beginning or end of any integrated arc. This trajectory sectioning and general branching capability is shown in the right half of the figure.

TRAJECTORY STRUCTURE



SIMULATION SECTION	FLIGHT PHASE
1 ——▶ 6	NOMINAL ASCENT
7	ABORT CASE 1
8	ABORT CASE 2
9	BOOSTER RETURN

TYPICAL ASCENT TRAJECTORY WITH BOOSTER ABORT/RETURN CONSTRAINTS


$x_{ij}(t)$
SECTION
1
2
3
4
5
6
7
FINAL ANALYTIC ARC
t
- - - - ANALYTIC TRAJECTORY ARC
——— NUMERICALLY INTEGRATED TRAJECTORY ARC

TRAJECTORY SECTIONING AND GENERAL BRANCHING CAPABILITY

Figure 1

GENERAL MATHEMATICAL PROBLEM

The mathematical problem is to extremize a performance measure which is a function of the vehicle and its trajectory when the vehicle/trajectory is constrained by up to six categories of constraints. The performance measure to be extremized has the form of Equation (2).

The problem, then, is to determine optimal $u_{kj}(t)$ and d_o such that J is extremized and such that the specified constraints are satisfied. The general mathematical model incorporates six types of problem-oriented equality and inequality constraints. Two fundamentally different types of constraints can be identified; those that are functions of the time-varying system variables (dynamic constraints), and those that are functions of the design parameters and the initial and final states of the trajectory sections (parametric constraints). For user convenience, two additional constraints depending upon integrals of the time-varying system variables are provided. These constraint categories are defined by equations (3) through (8).

The dynamic inequality constraints include the familiar state and control variable inequality constraints as a subset. The parametric equality constraints allow the specification of the boundary conditions on the differential equations as well as constraints on the time invariant design parameters. It is noted, however, that neither dynamic nor integral constraints can be defined on an analytic arc.

PERFORMANCE INDEX

$$J = \sum_{m=1}^{n_\phi} \Phi_m (t_j^o, t_j^f, x_{ij}^o, x_{ij}^f, d_p) + \sum_{j=1}^{n_j} \sum_{\ell=1}^{n_L} \int_{t_j^o}^{t_j^f} L_\ell \left[x_{ij}(t), u_{kj}(t), d_p, t \right] dt \qquad (2)$$

where Φ_m is the mth parametric performance term (e.g., net weight)

L_ℓ is the ℓth dynamic performance term (e.g., heating parameter)

n_ϕ is the number of parametric performance terms

n_L is the number of dynamic performance terms

and the superscripts "o" and "f" denote evaluation at the initial and final values of time during the corresponding section.

EQUATIONS FOR CONSTRAINT CATEGORIES

Dynamic inequality (e.g. specification of a maximum dynamic pressure during an interval):

$$\xi_{sj}\left[x_{ij}(t),\ u_{kj}(t),\ d_p,\ t\right] \geq 0 \qquad s = 1, 2, \ldots, n_\xi \tag{3}$$

Dynamic equality (e.g. specification of flight at a constant altitude during an interval):

$$\eta_{sj}\left[x_{ij}(t),\ u_{kj}(t),\ d_p,\ t\right] = 0 \qquad s = 1, 2, \ldots, n_\eta \tag{4}$$

Parametric inequality (e.g. exclusion of areas on the earth's surface from predicted impact for spent stages):

$$\zeta_s\,(t_j^o,\ t_j^f,\ x_{ij}^o,\ x_{ij}^f,\ d_p) \geq 0 \qquad s = 1, 2, \ldots, n_\zeta \tag{5}$$

Parametric equality (e.g. orbital insertion conditions):

$$\psi_s\,(t_j^o,\ t_j^f,\ x_{ij}^o,\ x_{ij}^f,\ d_p) = 0 \qquad s = 1, 2, \ldots, n_\psi \tag{6}$$

Integral inequality (e.g. specification of a maximum total heating parameter):

$$\sum_{j=1}^{n_j} \int_{t_j^o}^{t_j^f} Q_{sj} \left[x_{ij}(t), \mu_{kj}(t), d_p, t\right] dt - D_s \geqq 0 \qquad s = 1, 2, \ldots n_Q \tag{7}$$

Integral equality (e.g. total impulse):

$$\sum_{j=1}^{n_j} \int_{t_j^o}^{t_j^f} P_{sj} \left[x_{ij}(t), \mu_{kj}(t), d_{pj}, t\right] dt - C_s = 0 \qquad s = 1, 2, \ldots n_p \tag{8}$$

where ξ_{si}, η_{sj}, Q_{sj}, and P_{sj} denote independent functions applicable during simulation section j.

NONLINEAR PROGRAMMING TECHNIQUE

Nonlinear programming is a term applied to any algorithmic procedure seeking to extremize a function of N independent variables that are restricted to some subspace of Euclidian N-space. The problem then is to minimize J(y) as shown by Equations 9, 10, and 11.

The trajectory optimization problem is a nonlinear programming problem except for one feature: a continuous function of time is sought as the solution to the trajectory optimization problem, whereas the solution to the nonlinear programming problem is represented by a point in Euclidian N-space. This dissimilarity is resolved by approximating the continuous function of time by a function of n independent parameters. These n parameters then become a subset of the N independent variables in the nonlinear programming formulation.

There are two ways in which the n parameters can conveniently be chosen to describe the control. The n parameters can specify the control magnitude at specified points in time. An interpolation device, such as simple linear interpolation can be used to define the control at intermediate points in time. Alternately, the parameters can be regarded as coefficients of some mathematical model that is a function of time. The dynamic optimal control can be approximated to any desired accuracy by refining the parameterization of the control function.

The method of Fiacco-McCormick,[1-8] which is currently judged to be the best interior penalty function method* was selected as the nonlinear programming method to be employed to solve the highly constrained trajectory optimization problems. This method transforms the highly constrained trajectory optimization problem to a related sequence of unconstrained optimization problems in which the sequence is described by the positive real number r_k. The penalty function is the sum of the performance index J(y) and the Fiacco-McCormick penalty terms for inequality and equality constraints.

The algorithm begins at a point $y^{(0)}$ in Euclidian N-space at which all inequalities are satisfied (a feasible point). A systematic method for obtaining an initial feasible solution is given by Fiacco-McCormick[3] and is described later. The constant r_1 is selected and the point $y^{(1)}$ is found such that $P(y^{(1)}, r_1)$ is a minimum. A new value $r_2 < r_1$ is then selected and a point $y^{(2)}$ is found so that $P(y^{(2)}, r_2)$ is a minimum, etc. The limit approaches the solution to the nonlinear programming problem, Equations (9) to (11).

*An interior penalty function method starts with a solution estimate in the feasible region, that is in a region in which none of the inequality constraints are violated. A penalty term is added to the performance index, which keeps the solution in the feasible region at each stage in the iteration.

P — PROBLEM

Minimize J(y) (9)

subject to

$$g_i(y) \geq 0 \qquad i = 1,\ 2,\ \ldots,\ m \tag{10}$$

$$h_j(y) = 0 \qquad j = 1,\ 2,\ \ldots,\ p \tag{11}$$

where $y = (y_1, y_2, \ldots, y_N)^T$ and the functions $g_i(y)$ and $h_j(y)$ are single-valued functions of y. To maximize J(y), it is sufficient to minimize -J(y).

$$P(y, r_k) = J(y) + r_k \sum_{i=1}^{m} \frac{1}{g_i(y)} + r_k^{-1/2} \sum_{j=1}^{p} h^2{}_j(y) \tag{12}$$

$$\lim_{r_k \to 0} \left[\min_{y} P(y^{(k)}, r_k) \right] \tag{13}$$

$$P(y^{(k)}, r_k) > P(y^{(k+1)}, r_{k+1}) \quad \text{for} \quad r_k > r_{k+1} \tag{14}$$

Each iteration requires minimization of the function P for a given value of r. This problem is termed the P-problem, and its solution is fundamental to the method of Fiacco-McCormick. The computational algorithm which was implemented minimizes the function P, using a Fletcher-Powell[9] function minimization algorithm in conjunction with the simultaneous golden section search/cubic curve fit, one dimensional search of Johnson and Meyers.[10]

It can be seen from the form of the penalty term that a minimization technique will avoid points y that cause $g_i(y)$ to go to zero and become negative since $1/g_i(y)$ would increase without bound. Clearly, the initial point $y^{(0)}$ must be feasible. It is also clear that the minimization of $P(y, r_k)$ will force $h_j \rightarrow 0$, since otherwise these terms would increase without bound as $r^{1/2}$ goes to zero.

A sequence of minimizations is performed with $r_1 > r_2 \ldots > r_k \ldots > r_f$, rather than just one minimization of $P(y, r_k)$, because the latter minimization problem is very difficult to solve from a numerical standpoint. The solution of the P-problem at each stage then provides a good initial estimate for the solution of the P-problem at the following stage.

Fiacco and McCormick[2] prove that their method is computationally stable for the inequality constrained problem because of Equation (14).

GENERATION OF A FEASIBLE SOLUTION – This section reviews the algorithm developed by A. V. Fiacco for generating initial feasible solutions. An arbitrary initial design vector y^o will result in general in $s<m$ of the inequality constraints being satisfied while the remainder will be in violation. With no loss in generality, assume the constraints to be reordered so that the first s constraints are the satisfied constraints. Constraint $s + 1$ can be brought into the feasible set while simultaneously guaranteeing that constraints $1, 2,\ldots,s$ remain in the feasible set by solving the following nonlinear programming problem:

$$\text{minimize} \quad J(y) \equiv -g_{s+1}(y)$$

$$\text{subject to} \quad g_j(y) \geqslant 0 \quad j = 1,\ldots,s$$

using the Fiacco-McCormick algorithm previously described. The minimization is terminated as soon as g_{s+1} becomes positive. This constraint is then a member of the feasible set and the procedure is repeated with the next violated constraint until all inequality constraints are satisfied. Note that both the equality constraints, Equation 11, and the original performance index, Equation 9, are temporarily ignored during the generation of the feasible solution.

CONSTRAINED OPTIMAL CONTROL AS A NONLINEAR PROGRAMMING PROBLEM

Implementation of the method of Fiacco-McCormick requires a solution to the P-problem defined in the previous section. Before exhibiting the function P for the trajectory optimization problem, it is convenient to transform the integral constraints, Equations (7) and (8), into parametric constraints on the final values of the following pseudo-state variables, Equations 15 through 18.

Now the P function for the general trajectory optimization problem may be formed by adding to the objective function, Equation 2, penalty terms for constraints, Equations 3 to 6, 16, and 18, in accordance with Equation 12, to yield Equation 19.

PSEUDO-STATE VARIABLES

$$\dot{x}_{(n+i),\, j} = P_{ij}\left[x_{ij}(t),\ u_{kj}(t),\ d_p\ ,\ t\right] \qquad x^o_{(n+1),\, j} = 0 \qquad i = 1,\ 2,\ \ldots,\ n_p \tag{15}$$

with corresponding parametric constraints

$$x^f_{(n+i),} \ - C_i = 0 \qquad i = 1,\ 2,\ \ldots,\ n_p \tag{16}$$

and

$$\dot{x}_{(n+n_p+i),\, j} = Q_{ij}\left[x_{ij}(t),\ u_{ij}(t),\ d_p\ ,\ t\right] \qquad x^o_{(n+n_p+i),\, j} = 0 \qquad i = 1,\ 2,\ \ldots,\ n_Q \tag{17}$$

with corresponding parametric constraints

$$x^f_{(n+n_p+i),\, j} - D_i \geq 0 \qquad i = 1,\ 2,\ \ldots,\ n_Q \tag{18}$$

$$P = G(t_j^o,\ t_j^f,\ x_{ij}^o,\ x_{ij}^f,\ d_p) + \sum_{j=1}^{n_j} \int_{t_j^o}^{t_j^f} L_j^* \left[x_{ij}(t),\ u_{kj}(t),\ d_{pj},\ t \right] dt \tag{19}$$

where

$$\left.\begin{aligned}
G = {} & \sum_{m=1}^{n_\phi} \Phi_m (t_j^o,\ t_j^f,\ x_{ij}^o,\ x_{ij}^f,\ d_p) && \text{parametric performance} \\
& + r^{-1/2} \sum_{s=1}^{n_\Psi} \Psi_s^2 (t_j^o,\ t_j^f,\ x_{ij}^o,\ x_{ij}^f,\ d_p) && \text{parametric equality} \\
& + r^{-1/2} \sum_{i=1}^{n_p} \left[x_{(n+i),\ell}^f - C_i \right]^2 && \text{transformed integral quality} \\
& + r \sum_{s=1}^{n_\zeta} 1/\zeta_s (t_j^o,\ t_j^f,\ x_{ij}^o,\ x_{ij}^f,\ d_p) && \text{parametric inequality} \\
& + r \sum_{i=1}^{n_Q} 1\Big/\left[x_{(n+n_p+i),\ell}^f - D_i \right] && \text{transformed integral inequality}
\end{aligned}\right\} \tag{20}$$

and

$$\left.\begin{array}{ll}
L_j^* = \sum_{\ell=1}^{n_L} L_\ell\left[x_{ij}(t), u_{kj}(t), d_p, t\right] dt & \text{integral performance} \\
+ r^{-1/2} \sum_{s=1}^{n_\eta} \eta_{sj}^2 \left[x_{ij}(t), u_{kj}(t), d_p, t\right] & \text{dynamic equality} \\
+ r \sum_{s=1}^{n_\xi} 1/\xi_{sj} \left[x_{ij}(t), u_{kj}(t), d_p, t\right] & \text{dynamic inequality}
\end{array}\right\} \quad (21)$$

GENERATING A FEASIBLE SOLUTION FOR STATE AND CONTROL VARIABLE VARIABLE INEQUALITY CONSTRAINTS[12]

The state and control variable inequality constraints, ξ_{sj} in Equation 3 must be satisfied over a continuous range of time.

An initial design estimate represented by the initial control function estimate $u^o(t)$ and the initial design parameter estimate d_p^o will result in general in $s < n_\xi$ of the dynamic inequality constraints being satisfied, while the remainder are violated during some time interval.

As before, assume the constraints to be reordered so that the first s constraints are those satisfied for the pertinent time interval, while the remainder are those in violation. Dynamic constraint s + 1 can then be brought into the feasible set while simultaneously guaranteeing that the first s constraints remain in the feasible set by solving the problem to the right.

The Fiacco-McCormick minimization algorithm is terminated as soon as $\xi_{(s+1)} \leqslant 0$ for all pertinent time intervals. This constraint is then a member of the feasible set and the procedure is repeated for the next violated dynamic inequality constraint until all inequality constraints have been satisfied.

NUMERICAL CONSIDERATIONS – The first term in Equation 22 is the integral of the current inequality constraint being satisfied, over all time at which it is in violation, (see figure). This term has minimum value of 0, which will be achieved when the constraint trajectory is driven out of the violated region. The second term prevents the constraint from going into violation at a time point at which the constraint was initially satisfied. The third term prevents constraints already in the feasible set from becoming violated.

Obviously it is impossible to integrate the second term to the time at which $\xi = 0$. To do so numerically causes trapped points during the minimization process. As ξ at point A in the figure moves the epsilon distance from being just violated to being just feasible the second penalty term experiences a sudden increase, although the move was in a desirable direction. The function minimization algorithm becomes trapped. To alleviate this problem the time intervals for the evaluation of the second term are fixed at integration time points just outside the violated region at the start of the function minimization.

Theoretically, these boundaries should remain frozen throughout the function minimization. Experience indicates, however, that the integration limits can be updated at the end of each one dimensional search. Such frozen limits on the first term are not required.

A DYNAMIC INEQUALITY CONSTRAINT TRAJECTORY

$$\underset{u(t)\ d_p}{\text{minimize}} \sum_{j=1}^{n_j} \int_{t_j^o}^{t_j^f} \Bigg\{ - \xi_{(s+1),j} U(- \xi_{(s+1),j}) + r_k U(\xi_{(s+1),j}) \Big/ \xi_{(s+1),j} + r_k \sum_{i=1}^{s} 1/\ \xi_{ij} \Bigg\} dt \qquad (22)$$

$$\text{where } U(x) = \begin{Bmatrix} 1 \text{ if } x > 0 \\ 0 \text{ if } x \leqq 0 \end{Bmatrix}$$

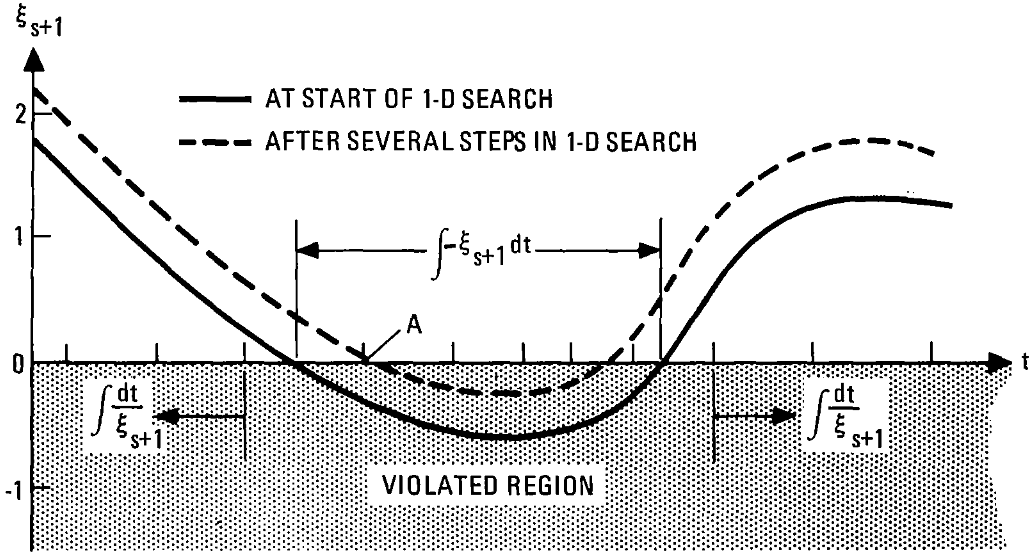


Figure 2

TRAJECTORY SIMULATION

To simulate and optimize Space Shuttle ascent and entry trajectories, a General Trajectory Optimization Program (GTOP) was developed.[11] This general-purpose computer program combines a flexible, high-speed, trajectory simulation module with a highly reliable nonlinear programming optimization driver. GTOP represents an implementation of all features of the general mathematical model presented previously.

The flight of the shuttle vehicle is described by a set of seven differential equations written in a relative velocity coordinate system. The first three elements of the state vector describe the current position of the vehicle with respect to a rotating earth: geocentric radius vector, r; geocentric latitude, θ; and geocentric longitude, ϕ.

DIFFERENTIAL EQUATIONS OF MOTION

$$\left.\begin{array}{l}
\boxed{\dot{r} = v \sin\gamma} \\[2ex]
\boxed{\dot{v} = \left[g_\theta (\cos\beta \cos\gamma) - g_r (\sin\gamma) + \frac{F_v g_0}{W} + \omega^2 r (\cos\theta)(\sin\gamma \cos\theta - \cos\beta \cos\gamma \sin\theta) \right]} \\[2ex]
\boxed{\dot{\gamma} = \frac{v}{r} (\cos\gamma) + \left[\frac{1}{v}\right] \left[-g_\theta (\cos\beta \sin\gamma) - g_r (\cos\gamma) + \frac{F_\gamma g_0}{W} + 2\omega V (\sin\beta \cos\theta) + \omega^2 r (\cos\theta)(\cos\gamma \cos\theta + \cos\beta \sin\gamma \sin\theta) \right]} \\[2ex]
\boxed{\dot{\beta} = \frac{1}{v(\cos\gamma)} \left[\frac{v^2}{r} (\sin\beta \cos^2\gamma \tan\theta) - g_\theta (\sin\beta) + \frac{F_\beta g_0}{W} + 2\omega V (\cos\gamma \sin\theta - \cos\beta \sin\gamma \cos\theta) + \omega^2 r (\sin\beta \sin\theta \cos\theta) \right]} \\[2ex]
\boxed{\dot{\theta} = \frac{V \cos\beta \cos\gamma}{r}} \\[2ex]
\boxed{\dot{\phi} = \frac{V \sin\beta \cos\gamma}{r \cos\theta}}
\end{array}\right\} \quad (23)$$

$$\boxed{\dot{w} = \text{tabular or analytic model}}$$

where

$$\left.\begin{array}{l}
F_\beta = (+\cos\lambda \sin\alpha \sin\sigma + \sin\lambda \cos\sigma) F_\xi + (-\sin\lambda \sin\alpha \sin\sigma + \cos\lambda \cos\sigma) F_\eta + (-\cos\alpha \sin\sigma) F_\zeta \\
F_v = (+\cos\lambda \cos\alpha) F_\xi + (-\sin\lambda \cos\alpha) F_\eta + (+\sin\alpha) F_\zeta \\
F_\gamma = (+\cos\lambda \sin\alpha \cos\sigma - \sin\lambda \sin\sigma) F_\xi + (-\sin\lambda \sin\alpha \cos\sigma - \cos\lambda \sin\sigma) F_\eta + (-\cos\alpha \cos\sigma) F_\zeta
\end{array}\right\} \quad (24)$$

g_r = the radial component of the gravitational acceleration vector

g_θ = the horizontal component of the gravitational acceleration vector (positive in local due north direction)

F = vector of aerodynamic and thrusting forces.

The velocity vector is described in a relative velocity coordinate system. The relative velocity coordinate system ($\hat{i}_\beta$ - $\hat{i}_V$ - $\hat{i}_\gamma$ axes) has its origin at the current position of the aerospace vehicle point mass and its axes aligned as shown in the left side of the figure.

The associated elements of the state vector (elements 4, 5, and 6) are the relative velocity magnitude* v; and, defining the orientation of the relative velocity vector, the relative flight path angle, γ, and the relative azimuth,β.

The seventh element of the state vector is the vehicle weight ($\dot{w}$). Analytic and Tabular models are available to define w.

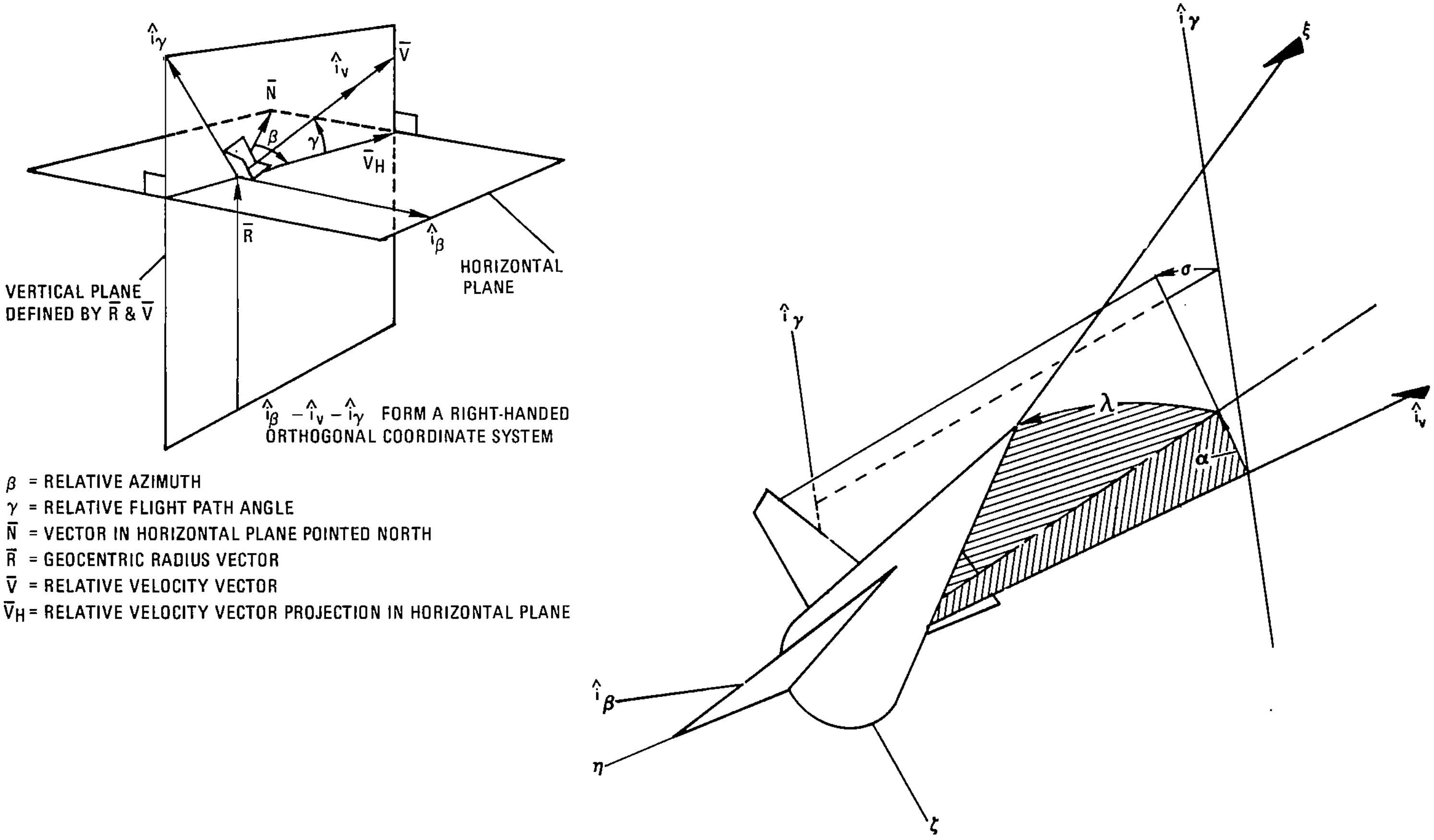
Vehicle attitude is defined by three attitude angles: roll (bank) angle, σ; pitch angle of attack, α; and yaw angle of attack, λ. If the vehicle axes ($\hat{\xi}$, $\hat{\eta}$, $\hat{\zeta}$) shown in the right side of the figure are initially aligned with the relative velocity axes ($\hat{i}_V$, $\hat{i}_\beta$, and $\hat{i}_\gamma$, respectively) then the vehicle attitude is defined by a sequenced rotation around the $\hat{\xi}$-$\hat{\eta}$-$\hat{\zeta}$ axes, consisting of:

1. A first rotation around the roll ($\hat{\xi}$) axis through the bank roll angle (σ).
2. A second rotation about the pitch axis ($\hat{\eta}$) axis through the pitch angle of attack (α).
3. A final rotation about the yaw axis ($\hat{\zeta}$) axis through the yaw angle of attack (λ).

A typical rotation is shown in the right side of the figure.

Relative here means relative to an earth fixed rotating coordinate system.

COORDINATE SYSTEMS AND ATTITUDE ANGLES



β = RELATIVE AZIMUTH

γ = RELATIVE FLIGHT PATH ANGLE

$\bar{N}$ = VECTOR IN HORIZONTAL PLANE POINTED NORTH

$\bar{R}$ = GEOCENTRIC RADIUS VECTOR

$\bar{V}$ = RELATIVE VELOCITY VECTOR

$\bar{V}_H$ = RELATIVE VELOCITY VECTOR PROJECTION IN HORIZONTAL PLANE

Figure 3

The central body total gravitational acceleration vector (g) is obtained directly from the central body gravitational potential energy function (U) by taking its gradient (∇U). The expression for U is shown in Equation 25.

The assumed gravity model includes up through the fourth harmonic term (N = 4) and assumes symmetry about the rotational axis. The corresponding expression for the gravitational components is $g_r = \partial U/\partial r$ and $g_\theta = -(\partial U/\partial \theta)/r$.

The local surface radius of the central body is defined by assuming the central body is an oblate spheroid with symmetry about the rotational axis and the equatorial plane shown at the bottom of the figure. This surface is described mathematically as a symmetric ellipsoid. The local surface radius (R_s) is then expressed as a function of the geocentric latitude (θ) from Equation 26 that defines the elliptic cross-section containing the rotational axis.

The atmosphere model is assumed to be static, that is, its characteristics are invariant with time. Two modeling methods are available. The first method utilizes n^{th} degree continuous expressions for the natural logarithm of the atmospheric density, the velocity of sound, and the atmospheric temperature, each as a function of altitude. The second method utilizes tables to define these quantities.

For suborbital velocities, the aerodynamic force coefficients are assumed to be dependent on Mach number (M_V) and the appropriate angle of attack (pitch or yaw angle of attack, α or λ, respectively). For near, or above, orbital velocities, the effects of viscosity can significantly affect the aerodynamic force. Examples of the latter occur during entry from orbit, synergetic maneuvers, and the terminal phases of ascent when the velocity is near orbital velocity. A realistic aerodynamic model that accounts for the effects of viscosity is available, as well as the standard velocity models. This model assumes that the aerodynamic force coefficients are dependent on the viscous parameter (P_V) and the appropriate angle of attack (α, or $\bar{\lambda}$). The corresponding aerodynamic forces F_k are

$$F_k = C_k q A_k \qquad \text{for } k = \xi, \eta, \zeta$$

where A_k is the reference area

q is the dynamic pressure

C_k is the aerodynamic force coefficient

Several propulsion models are available. The models range from simplified models in which the vacuum thrust and propellant flow rate are constant to models in which the vacuum thrust, propellant flow rate, and engine throttling coefficient are tabular functions of generalized independent variables. Of particular use to the space shuttle ascent simulation is a model which automatically provides throttling to meet thrust acceleration constraints. For all models, the effective thrust is computed as a function of the atmospheric pressure.

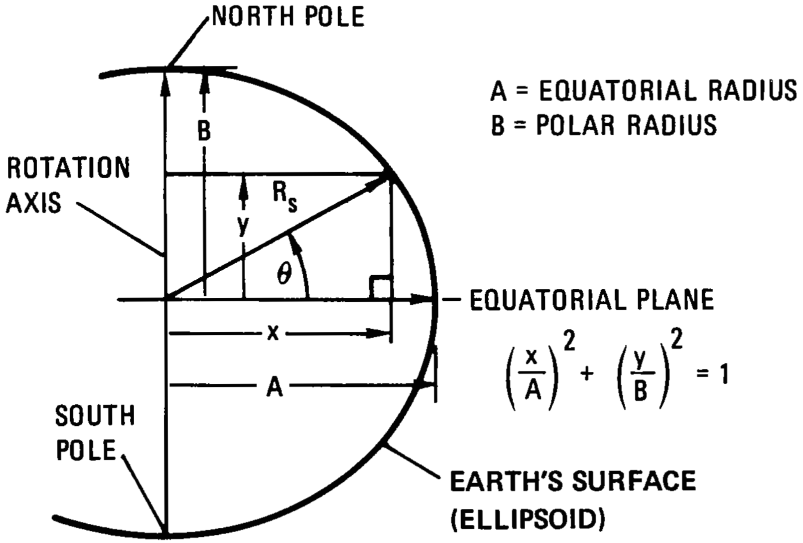
CENTRAL BODY SURFACE GEOMETRY

$$U = -\frac{\mu_k}{r}\left[1 - \sum_{n=2}^{N} J_n \left(\frac{A}{R}\right)^n P_n\right] \tag{25}$$

where

A is the earth's equatorial radius
R is the geocentric radius magnitude
J_n is the n^{th} gravitational harmonic coefficient
P_n is the n^{th} Legendre polynominal in sin θ terms
μ_k is the gravitational field constant of the central body

$$R_s = \left[\left(\frac{\mathrm{Cos}\,\theta}{A}\right)^2 + \left(\frac{\mathrm{Sin}\,\theta}{B}\right)^2\right]^{-1/2} \tag{26}$$



A = EQUATORIAL RADIUS
B = POLAR RADIUS

$$\left(\frac{x}{A}\right)^2 + \left(\frac{y}{B}\right)^2 = 1$$

Figure 4

SPACE SHUTTLE BOOSTER RETURN EXAMPLE

An optimal space shuttle booster return trajectory was determined by using the GTOP computer program.[13] The trajectory was initiated at the staging point of a reference ascent trajectory. The following conditions were assumed:

Altitude = 74610.3 meters (244,784 ft.)

Velocity (relative) = 3299.2 meters/sec (10,824 fps)

Gamma (relative) = 5.654 deg.

Heading Azimuth (relative) = 182.495 deg.

Latitude = 32.788 deg.

Longitude = 239.343 deg.

Wing Area = 785.12 sq. meters (8,451 sq. ft.)

Weight = 349,942.9 kg (771,492 lb.)

Wing Loading (W/S) = 445.715 kg/sq. meter (91.29 psf)

Staging Time = 216.36 sec.

The aerodynamic force coefficients, C_ξ and C_ζ which were used are shown in the figure.

The problem is to minimize the distance of the Space Shuttle booster from the launch site after the booster has completed re-entry (altitude = 6096.0 meters (20,000 feet)). Minimizing this flyback distance minimizes the jet fuel required to execute the powered return to the launch site. Clearly, the less flyback fuel required, the grea he payload injected into orbit.

The entry trajectory for the Space Shuttle booster configuration was constrained by five state variable and control variable inequality constraints:

Total acceleration load factor $\leq$ 4g

Dynamic pressure $1/2\,\rho V^2 \leqslant 2441.2$ kg/sq. meter (500 lb./sq. ft.)

Instantaneous heating parameter $1/2\,\rho V^3 \leqslant 4.762 \times 10^6$ kg/sec. meter (3.2×10^6 lb./sec. ft.)

Angle of attack $\geqslant$ 0 deg.

Angle of attack $\leq$ 90 deg.

The last two constraints were due to the unavailability of aerodynamic data outside of this region.

C_ξ AND C_ζ AS FUNCTIONS OF MACH NUMBER AND ANGLE OF ATTACK

α		-90.	0.0	4.0	8.0	12.0	16.0	20.0	25.0	30.0	40.0	50.0	60.0	70.0	90.0
0	C_ζ	2.2761	-.0264	-.1524	-.2819	-.4240	-.5679	-.6911	-.8473	-1.0127	-1.3349	-1.6549	-1.9581	-2.2761	-2.2761
	C_ξ	.0000	-.0408	-.0309	-.0101	.0274	.0785	.1186	.1469	.1459	.1671	.1909	.1916	.1719	.0000
	C_ζ	2.2761	-.0264	-.1524	-.2819	-.4240	-.5679	-.6911	-.8473	-1.0127	-1.3349	-1.6549	-1.9581	-2.2761	-2.2761
.6	C_ξ	.0000	-.0408	-.0309	-.0101	.0274	.0785	.1186	.1469	.1459	.1671	.1909	.1916	.1719	.0000
	C_ζ	2.8706	-.0240	-.1879	-.3501	-.5183	-.6909	-.8914	-1.1200	-1.3066	-1.6810	-2.0687	-2.4506	-2.8706	-2.8706
.9	C_ξ	.0000	-.0580	-.0638	-.0698	-.0768	-.0838	-.0925	-.1067	- .1232	- .1364	- .1171	- .0754	- .0073	.0000
	C_ζ	2.9920	-.0490	-.2183	-.3892	-.5736	-.7704	-.9860	-1.2103	-1.3993	-1.7694	-2.1582	-2.5534	-2.9920	-2.9920
1.2	C_ξ	.0000	-.1555	-.1543	-.1540	-.1543	-.1509	-.1397	-.1583	- .1736	- .1796	- .1661	- .1174	- .0248	.0000
	C_ζ	2.7028	-.0400	-.1533	-.2704	-.3986	-.5344	-.6788	-.8645	-1.0648	-1.4892	-1.8972	-2.2940	-2.7028	-2.7028
3.0	C_ξ	.0000	-.0900	-.0995	-.1135	-.1197	-.1224	-.1254	-.1265	- .1243	- .1211	- .1037	- .0866	- .0153	.0000
	C_ζ	2.5590	.0000	-.0764	-.1743	-.2826	-.4039	-.5372	-.7231	-. .9265	-1.3656	-1.7869	-2.1521	2.5590	2.5590
5.0	C_ξ	.0000	-.0500	-.0598	-.0765	-.0933	-.1026	-.1163	-.1262	- .1348	- .1464	- .1574	- .1225	- .0449	.0000
	C_ζ	2.3935	.0000	-.0710	-.1561	-.2639	-.3848	-.5216	-.7065	- .9004	-1.3335	-1.7276	-2.0652	-2.3935	-2.3935
8.0	C_ξ	.0000	-.0400	-.0552	-.0689	-.0870	-.0977	-.1028	-.1119	- .1095	- .1081	- .0724	- .0630	- .0901	.0000
	C_ζ	2.3935	.0000	-.0710	-.1561	-.2639	-.3848	-.5216	-.7065	- .9004	-1.3335	-1.7276	-2.0652	-2.3935	-2.3935
50.0	C_ξ	.0000	-.0400	-.0552	-.0689	-.0870	-.0977	-.1028	-.1119	- .1095	- .1081	- .0724	- .0630	- .0901	.0000

Reference Area = 785.12 sq. meters (8451 sq. ft.)

The initial angle of attack and bank angle were each modeled by 12 parameters as a function of Mach number, a monotonic function of time for the entry. This independent variable yields nearly uniform sensitivities of the performance index (flyback distance) to variations in the control modeling parameters. Time is unsatisfactory in this respect as an independent variable. The controls were tabulated at the following Mach numbers: 0, 0.6, 0.9, 1.2, 2, 3, 4, 5, 6, 7, 8, 9, 11, and 50.

The initial guess on the optimal control histories was intentionally poor (figures 5 and 6) to demonstrate the insensitivity of the algorithm to initial guesses. The trajectory corresponding to the initial control estimate resulted in a flyback distance of 756.91 km (408.7 n.mi.) and experienced a maximum acceleration load factor of 5.0g. Two one-dimensional minimization searches were required to obtain a solution satisfying all constraints (called "initial feasible solution" in figures 5 and 6. Since no attempt is made to maximize performance when generating the initial feasible solution, the flyback distance for the initial feasible solution was 862.29 km (465.6 n.mi.). However, after one unconstrained function minimization ($r = 1.0$) the flyback distance was reduced to 732.10 km (395.3 n.mi.).

After three more unconstrained function minimizations (with $r = 0.1$, 0.01, and 0.001, respectively) the final control history shown in figures 5 and 6 was obtained with a corresponding flyback range of 684.87 km (369.8 n.mi.). The relevant trajectory parameters are shown in figures 7 to 10. Note that both the acceleration load factor and the dynamic pressure state variable constraints are simultaneously active. The ripples in the load factor constraint are due to the discretization of the control.

ANGLE OF ATTACK VERSUS MACH NUMBER

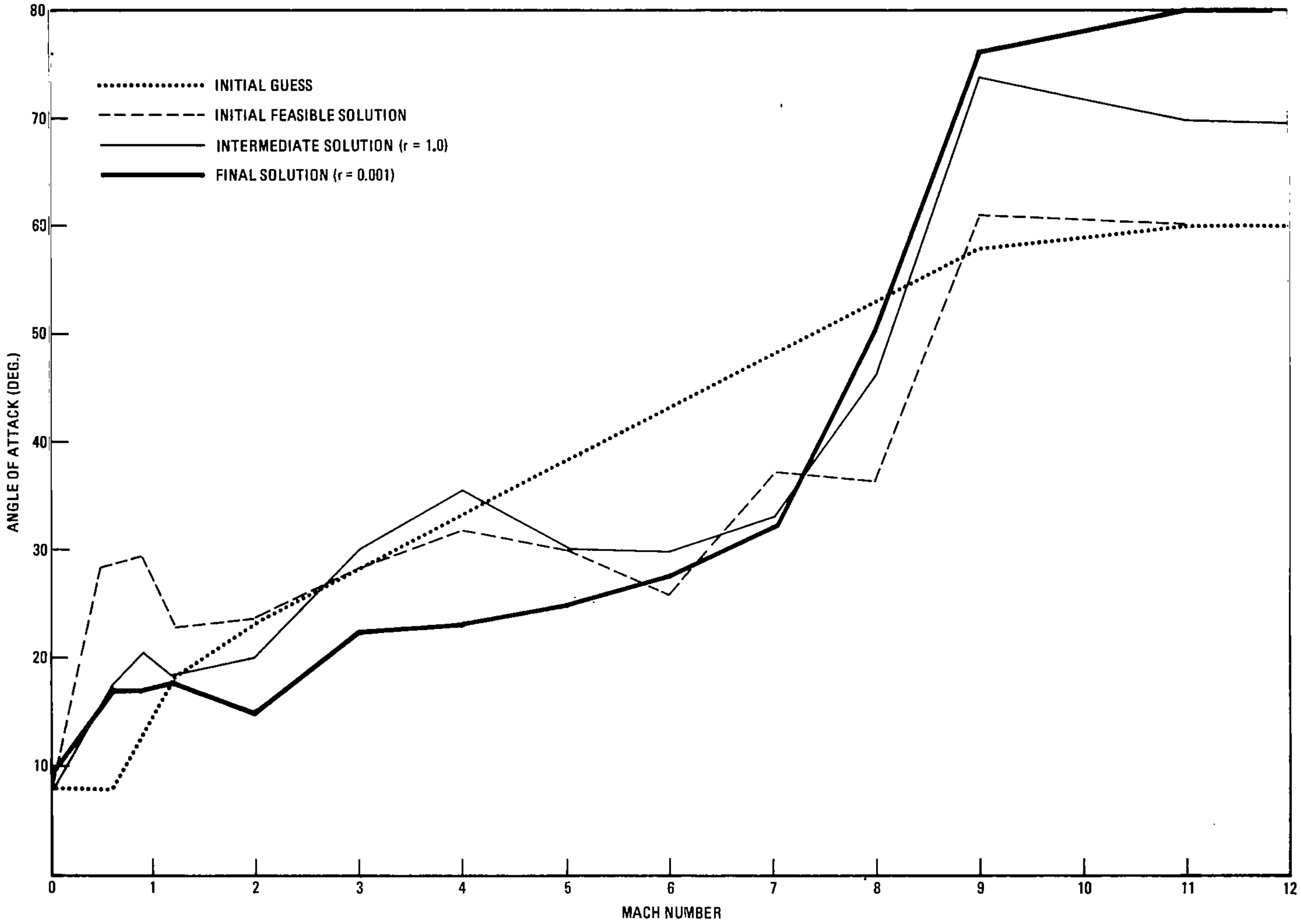


Figure 5

BANK ANGLE VERSUS MACH NUMBER

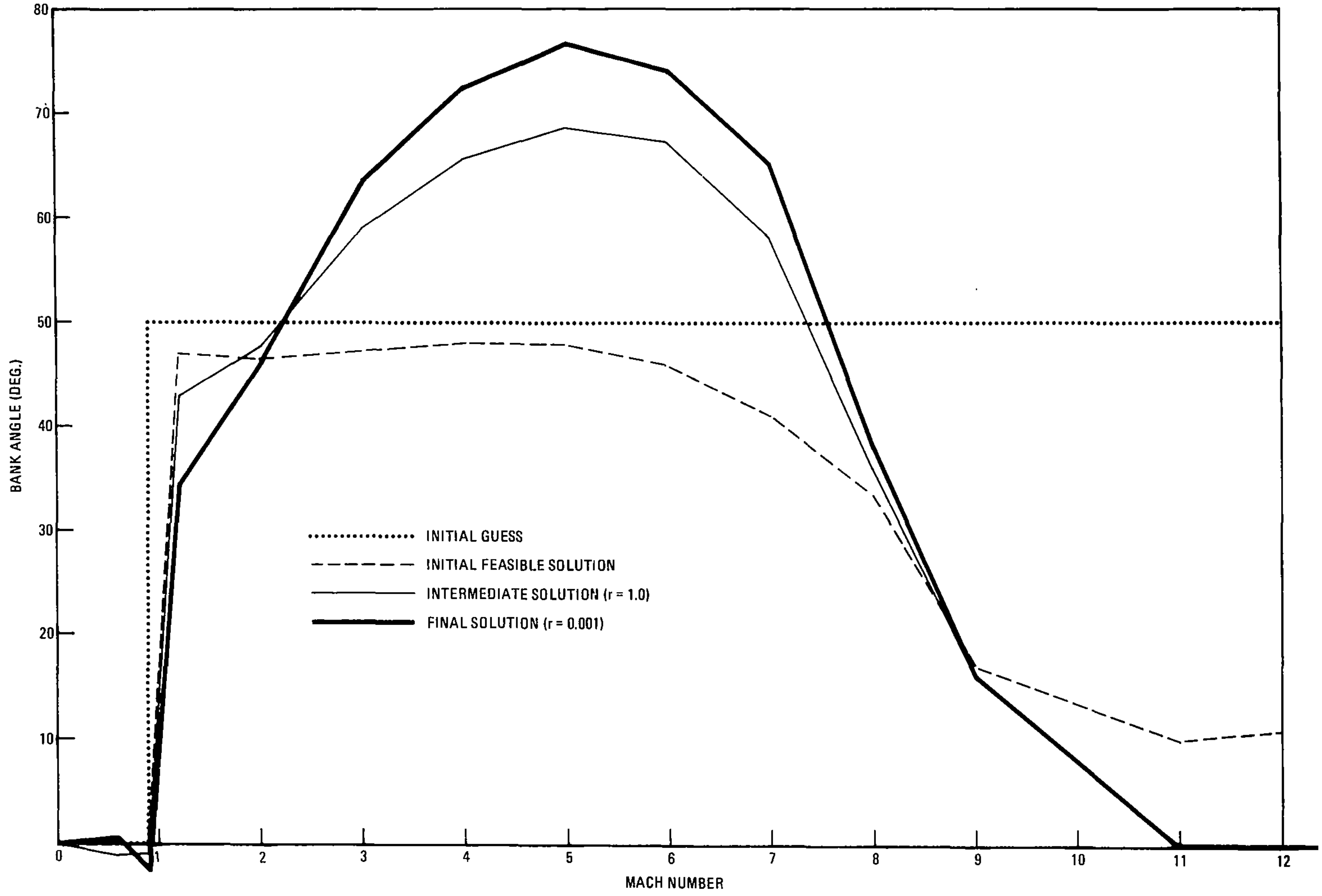


Figure 6

OPTIMAL MACH NUMBER AND CONTROL TIME HISTORIES

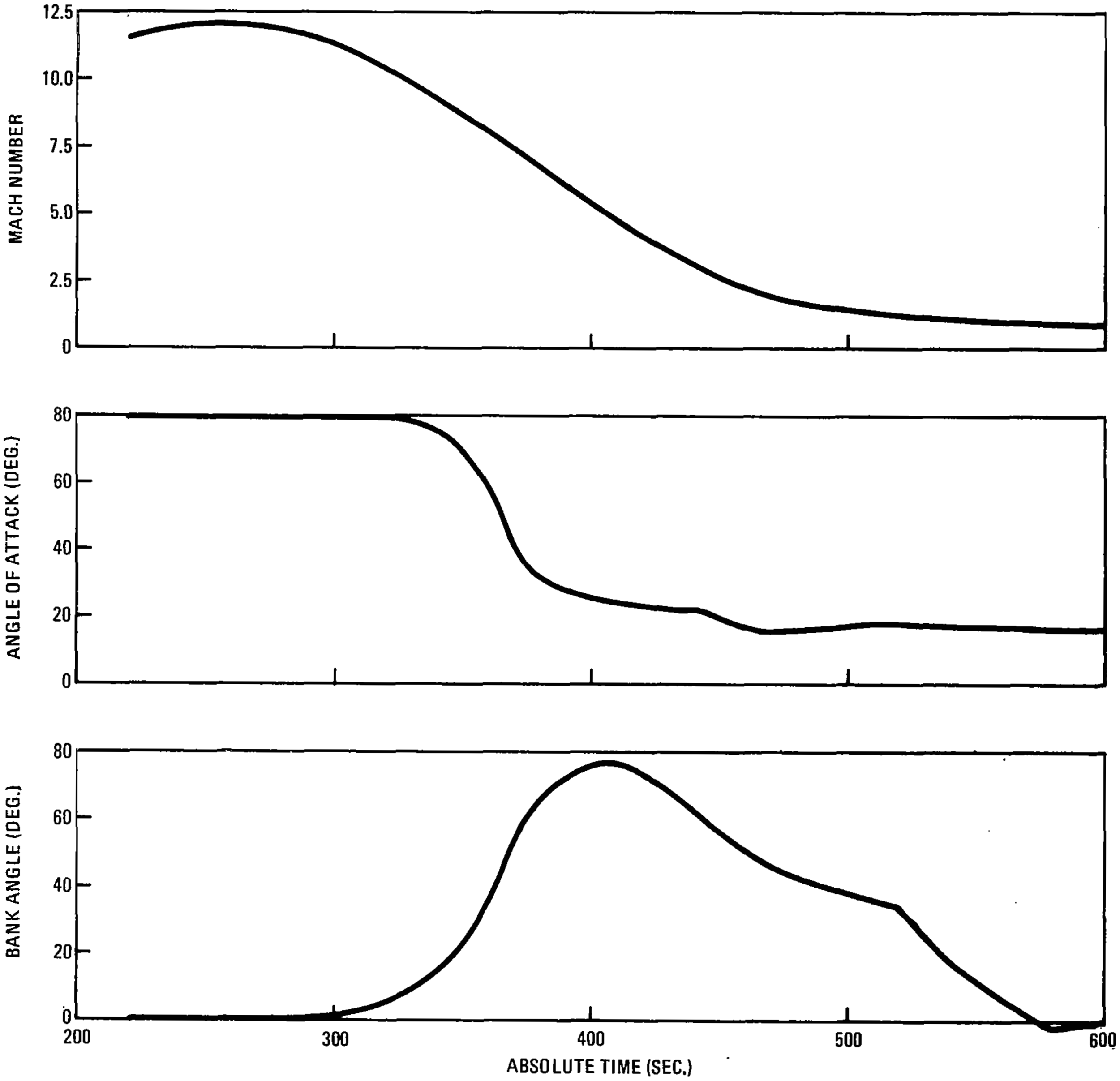


Figure 7

ACCELERATION LOAD FACTOR AND DYNAMIC PRESSURE CONSTRAINTS

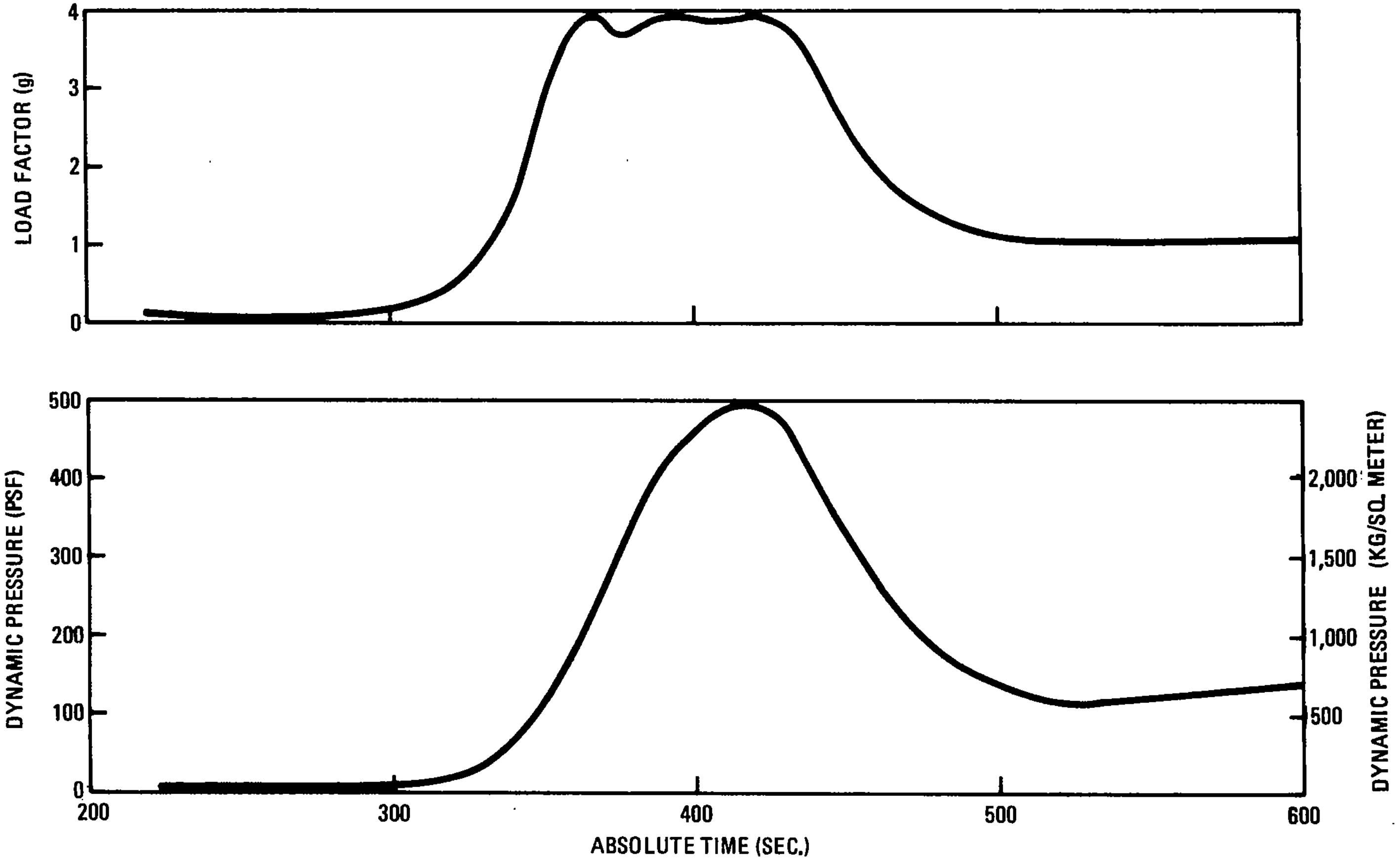


Figure 8

OPTIMIZED ALTITUDE AND RELATIVE VELOCITY TIME HISTORY

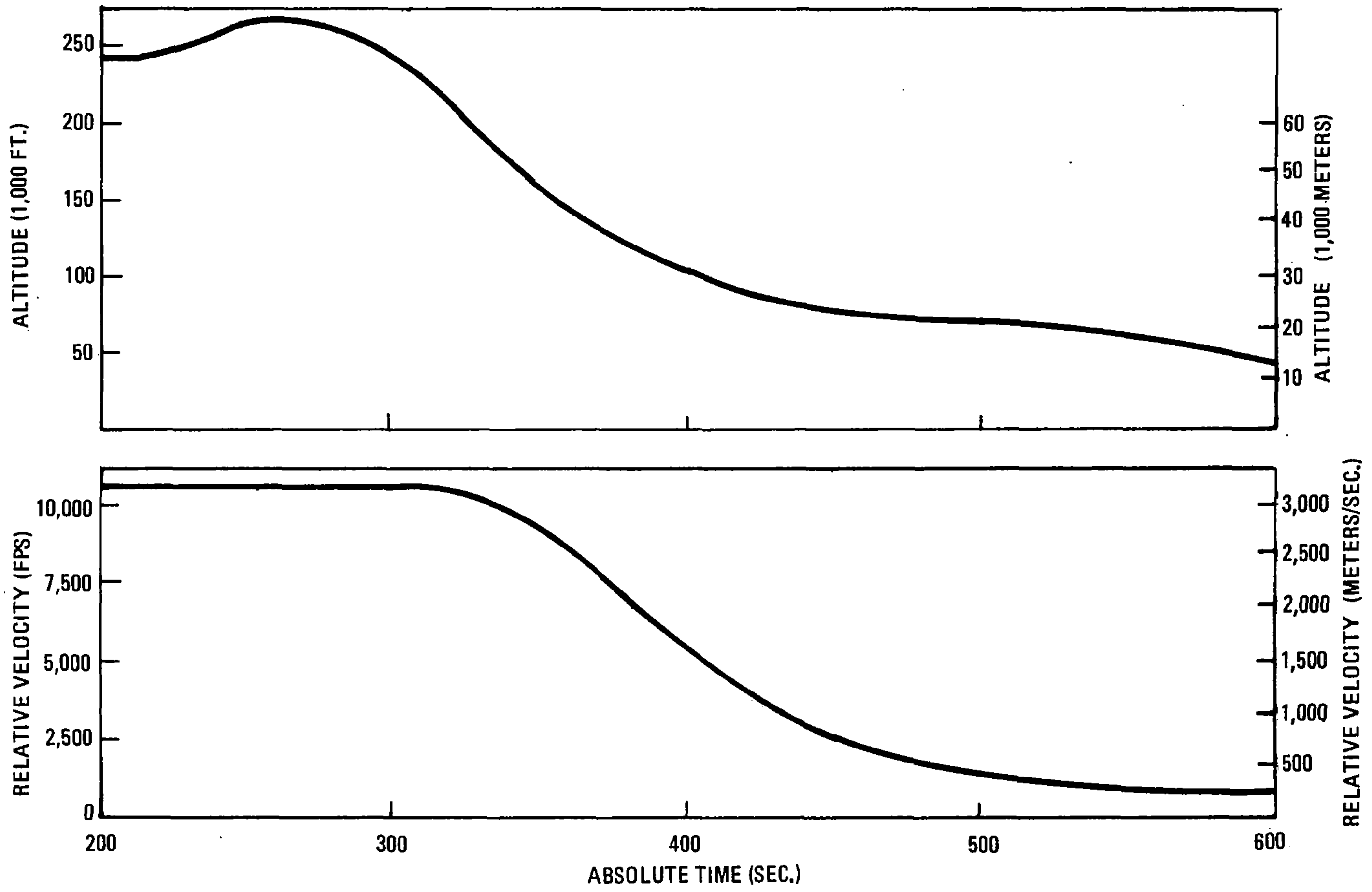


Figure 9

OPTIMIZED RELATIVE FLIGHT PATH ANGLE, AZIMUTH, AND DOWNRANGE DISTANCE

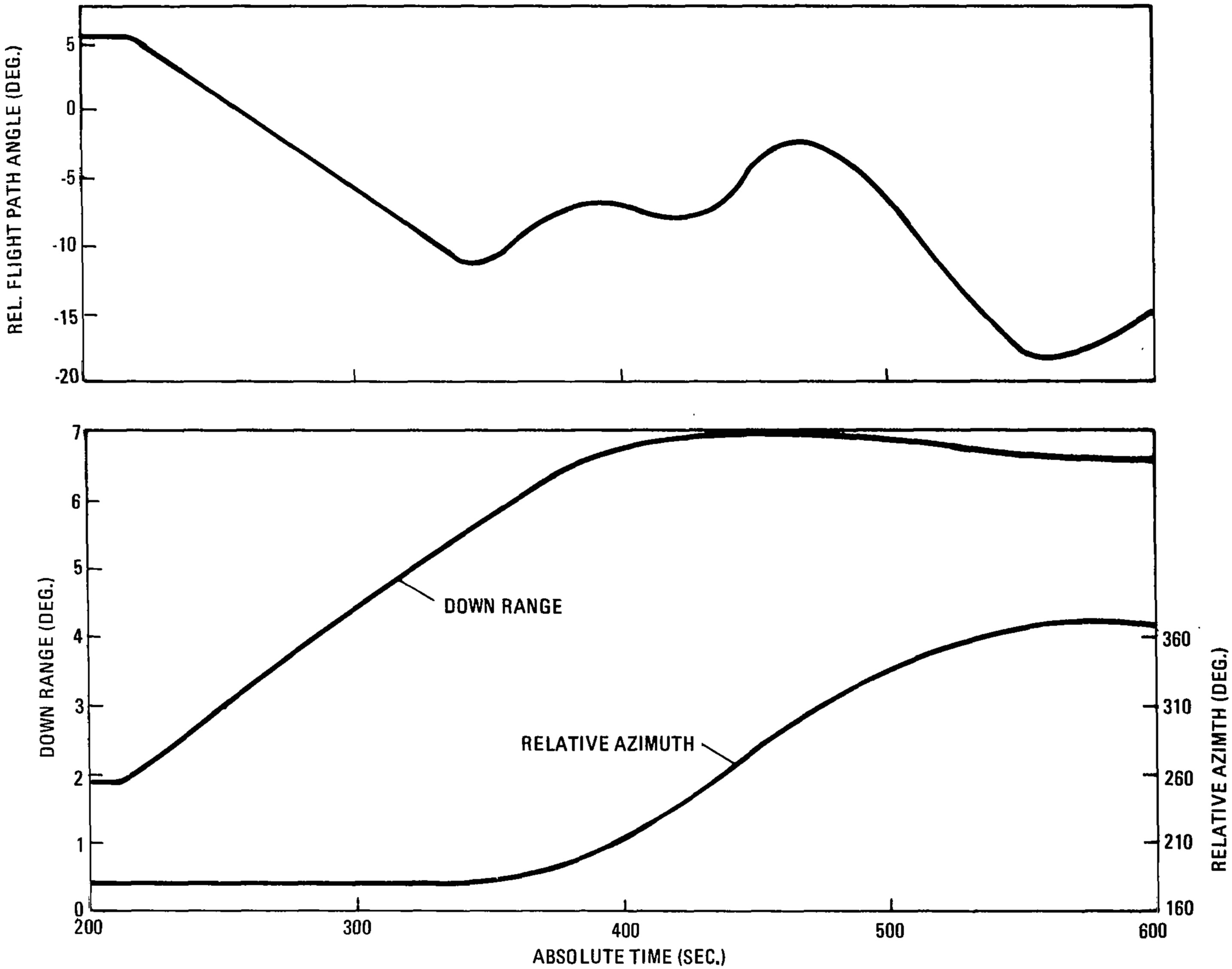


Figure 10

REFERENCES

1. Fiacco, A. V. and McCormick, G. P., *Nonlinear Programming: Sequential Unconstrained Minimization Techniques,* John Wiley & Sons, Inc., New York, 1968.
2. Fiacco, A. V., and McCormick, G. P., *Programming under Nonlinear Constraints by Unconstrained Minimization: A Primal-Dual Method,* RAC-TP-96, Research Analysis Corporation, September 1963.
3. Fiacco, A. V., and McCormick, G. P., "The Sequential Unconstrained Minimization Technique for Nonlinear Programming: A Primal-Dual Method," *Mgt. Sci.,* 10(2): 360-66 (1964).
4. Fiacco, A. V., and McCormick, G. P., "Computational Algorithm for the Sequential Unconstrained Minimization Technique for Nonlinear Programming," *Mgt. Sci.,* 10(4): 601-17 (1964).
5. Fiacco, A. V., and McCormick, G. P., "The Sequential Unconstrained Minimization Technique (SUMT) without Parameters," *Opns. Res.,* 15(5): 820-27 (1967).
6. Fiacco, A. V., and McCormick, G. P., "Extensions of SUMT for Nonlinear Programming: Equality Constraints and Extrapolation," *Mgt. Sci.,* 12(11): 816-29 (1966).
7. Fiacco, A. V., *Sequential Unconstrained Minimization Methods for Nonlinear Programming,* Ph.D. dissertation, Northwestern University, Evanston, Ill., June 1967.
8. Fiacco, A. V., *Penalty Methods for Mathematical Programming in E(N) with General Constraint Sets,* Research Analysis Corp. RAC-TP-385, 1969.
9. Fletcher, R., and Powell, M. J. D., "A Rapidly Convergent Descent Method for Minimization," *Computer Journal,* June 1963.
10. Guilfoyl, G., Johnson, I., and Wheatly P., "One-Dimensional Search Combining Golden Section and Cublc Techniques" *NASA* CR-65994, Jan. 1967.
11. Tramonti, L. G., Brusch, R. G., Schappelle, R. H., *Hypersonic Vehicle Trajectory Optimization,* Convair Aerospace Report GDC-ERR-1604, 1970.
12. Brusch, R. G. and Schappelle, R. H., "Obtaining Feasible Solutions to Inequality Constrained Control Problems," to be presented at the Fifth Hawaii International Conference on Systems Sciences, Honolulu, Hawaii Jan 11-13, 1972.
13. Brusch, R. G., "A Nonlinear Programming Approach to Space Shuttle Trajectory Optimization," *XXII Congress of the International Astronautical Federation,* Brussels, Belgium, Sept. 1971.