Improved Secant Method Applied to Boost Trajectory Optimization

R.G. Gottlieb*
Lewis Industries, Kansas City, Mo.
and
W.T. Fowler†
University of Texas at Austin, Austin, Texas

The present study presents a secant trajectory optimization procedure which, when combined with a min-H steepest-ascent algorithm, efficiently produces optimal lifting boost trajectories. The form of the secant is simpler than the secant procedure which has been used entensively at the Marshall Space Flight Center. Approximations of the Jacobian using a least squares procedure helps to speed convergence of the procedure. Results of a complex lifting boost trajectory optimization carried out using the improved secant are summarized.

Introduction

WITH the advent of the space shuttle program, a great deal of interest has been generated in optimal lifting-booster flight in the atmosphere. The synthesis of such trajectories requires the solution of a very complex two-point boundary-value problem.

Over the past fifteen years, a number of computational techniques have been proposed for the solution of complex two-point boundary-value problems. These can be characterized as gradient methods[1-6] and higher order methods.[7-12] The gradient methods are, relatively speaking, easier to implement on a computer, and give good convergence when far from the solution. On the other hand, although the higher order methods converge well when near the solution, they suffer from the disadvantage of being either much more complex and difficult to implement, or very difficult to initiate with trial solutions, or both. It was felt that the combination of the min-H[3,4] method and a numerical secant method[13,14] would lead to a technique which would be relatively easy to implement, easy to initiate, and in addition, efficient in its convergence properties.

The choice of control variables was felt to be a significant issue in this study. Atmospheric flight has generally been described in a velocity or noninertial coordinate system with angle-of-attack as a primary control variable[1,2]; whereas, in the Saturn V program, inertial attitude angles were used as control variables during the launch phase.[15] It has been observed that the use of angle-of-attack as a control variable in a gradient program is inherently unstable and that inertial attitude angle control variables are to be preferred. In addition, the use of inertial attitude angles makes it somewhat easier to define a nominal launch trajectory since the direction of thrust is uncoupled from the path flown. Consequently, the equations of motion herein are formulated in a Cartesian system using inertial attitude angles as control variables. It is shown that it is not difficult to describe aerodynamic forces and the condition of optimality in such a system, contrary to what might at first be assumed.

For the present study, two programs were constructed. Both were identical in terms of Earth model, coordinate systems, control variables, vehicle characteristics, and staging logic. The first program was a min-H steepest ascent program, and the second was a secant controlled program which satisfied the optimality conditions of the calculus-of-variations. Trajectories were initiated on the min-H program with final convergence provided by the secant-COV program. Both programs model the earth as an oblate rotating spheroid. Atmospheric properties are calculated using the Partick 1963 model atmosphere. Aerodynamic axial force and normal force are assumed to be functions of angle-of-attack and Mach number.

Presented at the AAS/AIAA Astrodynamics Specialist Conference, Nassau, Bahamas, July 28-30, 1975; (no preprint); submitted Nov. 5, 1975; revision received Oct. 13, 1976.

Index category: LV/M Trajectories.

*Vice President. Member AIAA.

†Associate Professor Department of Aerospace Engineering and Engineering Mechanics. Member AIAA.

Secant Method

The development of the secant method begins with the optimization problem stated in the usual form, i.e., extremize a performance index $\phi(\bar{X}_f, t_f)$ subject to the constraints

$$\dot{\bar{X}} = \bar{F}(\bar{X}, \bar{u}, \bar{\beta}, t)$$

$$\bar{X}(t_0) = \bar{X}_0 \qquad \bar{\psi}(\bar{X}_f, t_f) = 0 \tag{1}$$

where $\bar{X}$ is an n-vector of state variables, $\bar{u}$ is an m-vector of control variables, $\bar{\beta}$ is a q-vector of system parameters, and $\bar{\psi}$ is a p-vector of terminal constraint relations. Through standard operations[3,8,11,12] the problem can be reduced to a two point boundary value problem in $2n$ variables, $\bar{z}$, made up of the $\bar{X}$'s and n Lagrange multipliers, $\bar{\lambda}$. In the usual trajectory optimization problem[11,12], n of the quantities $\bar{z}$ are known at t_0 and $n(n+1$ if t_f is unknown) relations between the $\bar{z}$'s and t_f are known at the final time. The secant procedure basically calls for one to guess the unknown $\bar{z}$'s at t_0 (denote these quantities $\bar{x}$) and then to integrate a trajectory based on these guesses. At the end of the trajectory, the n or $n+1$ terminal conditions (call these $\bar{f}$) will not be satisfied. The quantities $\bar{f}$ are functions of the final values of $\bar{z}$ and t_f which in turn are functions of the guessed quantities, $\bar{x}$. Thus, each complete trajectory gives a set of values $\bar{f}(\bar{x})$, and the goal is to find $\bar{x}$ such that

$$\bar{f}(\bar{x}) = 0 \tag{2}$$

Since each value of $\bar{f}(\bar{x})$ requires integraion of a complete trajectory, and random guessing of the $\bar{x}$ vector will not generally lead to satisfaction of Eq. (2). Some type of convergent iteration procedure is required for solution of the problem.

In the perturbation method for solving the n nonlinear equations $\bar{f}(\bar{x}) = 0$, a Jacobian matrix $\partial \bar{f}/\partial \bar{x}$, is generated after each n iterates (the usual value of n is 1). The fact that

the Jacobian is based upon a single trial solution plus n linear variations about the trial solution implies that information available from earlier trial solutions is ignored.

It is possible to develop methods predicated upon the supposition that information gleaned from successive trial solutions may be sufficient to provide a reasonable estimate of the direction in which the solution lies. In place of generating n linear perturbations about a single nonlinear trial solution, one may generate $n+1$ trial solutions (using the nonlinear equations of motion) and then calculate an estimate of the Jacobian based on the end conditions of the $n+1$ trial solutions. Once $n+1$ trial solutions have been generated, a new Jacobian can be calculated after each successive trial solution. The new Jacobian can be calculated by two distinct procedures. The "best" $n+1$ trial solutions can be employed to construct the Jacobian in a least squares sense. If more than $n+1$ trial solutions are used in conjunction with least squares, the Jacobian matrix defines a plane cutting through the surface $\bar{f}(\bar{x})$ near which the trial solutions lie.

In the perturbation method, the Jacobian defines a plane tangent to the surface $\bar{f}(\bar{x})$ on which $\bar{f}$ is linearly related to $\bar{x}$ for small variations about the tangent point. In the methods where $n+1$ trial solutions are utilized, the Jacobian matrix defines a plane cutting through the surface $\bar{f}(\bar{x})$ (a sort of average tangent plane, or, more precisely, a secant plane) on which the trial solutions lie.

A secant method for solving Eq. (2) has been developed at the NASA Marshall Space Flight Center (MSFC) and has been used in both trajectory optimization and guidance. A brief derivation of the improved secant procedure will now be given. The MSFC secant procedure will be identified prior to the simplifications which improve the procedure.

Consider the case in which Eq. (2) is an n-vector. We choose $n+1$ sets of points $\bar{x}^i (i=1,2,\ldots,n+1)$ and integrate the trajectories to obtain the vectors $\bar{f}^i$ $(i=1,2,\ldots,n+1)$. Now we fit a "plane" in n space through the $n+1$ values of f_1, another "plane" through the $n+1$ values of f_2, etc. The equation of a plane in n space is $f_i = a_{ij}x_j + b_i$. The $n+1$ relations for f_1 corresponding to the $n+1$ data sets $\bar{x}^i$ give $n+1$ linear equations in the $n+1$ unknowns (a_{1j}, b_1) $(j=1,n)$. Similarly, the equations for f_2, f_3, $\ldots f_n$ give $(n-1)\times(n+1)$ additional linear equations in the $(n-1)\times(n+1)$ variables (a_{ij}, b_j) $(i=2,\ldots n, j=1,\ldots n)$. The solution of this set of linear equations can be written as

$$G = Y^{-1}F \tag{3}$$

where G is an $(n+1)\times(n)$ matrix given by

$$G = \begin{bmatrix} a_{ij} \\ \text{------} \\ b_j \end{bmatrix} \quad (i,j=1,\ldots n)$$

Y is a square matrix of dimension $n+1$ given by

$$Y = \begin{bmatrix} x_1^1 & x_2^1 & x_3^1 & \ldots & x_n^1 & 1 \\ \vdots & & & & \vdots & \vdots \\ x_1^{n+1} & x_2^{n+1} & x_3^{n+1} & \ldots & x_n^{n+1} & 1 \end{bmatrix}$$

and F is an $(n+1)\times n$ matrix given by

$$F = \begin{bmatrix} f_1^1 & f_2^1 & \ldots & f_n^1 \\ \vdots & & & \vdots \\ f_1^{n+1} & & & f_n^{n+1} \end{bmatrix}$$

Note that the superscripts on the x's denote the initial condition set (point in n-space) used to generate the trjaectory and those on the f_i's denote that those values of the terminal conditions result from that set of initial conditions.

We now seek a new vector $(\bar{x}^{\text{new}}, 1)$ which has the property that it lies at the intersection of the n planes in n space, i.e.

$$[\bar{f}^{\text{new}}]^T = [(\bar{x}^{\text{new}})^T \vdots 1]G \tag{4}$$

Equation (4) has $2n$ unknown quantities, the $\bar{x}^{\text{new}}$ vector and the $\bar{f}^{\text{new}} = \bar{f}(\bar{x}^{\text{new}})$ vector. We obviously cannot solve Eq. (4) unless more information is available. At MSFC, decreasingly small values are prescribed for the $\bar{f}^{\text{new}}$ components in order to achieve convergence.

Since the last element in the new vector is 1, we can rewrite Eq. (4) as

$$[(\bar{f}^{\text{new}})^T \vdots 1] = [(\bar{x}^{\text{new}})^T \vdots 1] \begin{bmatrix} G & \vdots & 0 \\ & \vdots & \vdots \\ & \vdots & 0 \\ & \vdots & 1 \end{bmatrix} \tag{5}$$

in order to make the matrix containing G square. Solving Eq. (5) for the unknown vector and substituting for G from Eq. (3) we get

$$[(\bar{x}^{\text{new}})^T \vdots 1] = [(\bar{f}^{\text{new}})^T \vdots 1] \begin{bmatrix} F & \vdots & 1 \\ & \vdots & \vdots \\ & \vdots & 1 \end{bmatrix}^{-1} Y \tag{6}$$

This is the form of the secant implemented at MSFC. The details of the derivation between Eq. (5) and Eq. (6) are given in Ref. 16.

An improvement in the secant can be obtained by the obvious choice of $\bar{f}^{\text{new}} \equiv 0$ in Eq. (6). The fact that the row vector $[(\bar{f}^{\text{new}})^T \vdots 1]$ now has only one nonzero element means that only the last row of the matrix following it is needed. Let F_1 denote the matrix inverted in Eq. (6) and let F_1 be partitioned into four submatrices with the lower right-hand submatrix being a scalar. Then F_1 and F_1^{-1} can be written as

$$F_1 = \begin{bmatrix} A & \vdots & B \\ \cdots & \vdots & \cdots \\ C & \vdots & 1 \end{bmatrix} \qquad F_1^{-1} = \begin{bmatrix} P & \vdots & Q \\ \cdots & \vdots & \cdots \\ R & \vdots & S \end{bmatrix} \tag{7}$$

where P, Q, R, and S are given by

$$P = (A - BC)^{-1} \tag{8}$$

$$Q = -A(A - BC)^{-1}B \tag{9}$$

$$R = -C(A - BC)^{-1} \tag{10}$$

$$S = 1 + C(A - BC)^{-1}B \tag{11}$$

Thus, after using Eqs. (7)-(11) in Eq. (6), we get

$$[(\bar{x}^{\text{new}})^T \vdots 1] = [R \;\; S]Y \tag{12}$$

Since, from Eq. (10) and (11), $S = 1 - RB$, Eq. (12) can be rewritten in component form as

$$x_i^{\text{new}} = x_i^{n+1} + R \begin{bmatrix} x_i^1 \cdots x_i^{n+1} \\ \vdots \\ x_i^n \cdots x_i^{n+1} \end{bmatrix} \quad (i=1,\ldots,n) \tag{13}$$

Inserting the quantities making up R into Eq. (13), and then substituting for A, B, and C in terms of the f values, Eq. (13)

becomes

$$x_i^{\text{new}} = x_i^{n+1} - (f^{n+1})^T \begin{bmatrix} \Delta f_1^1 \cdots \Delta f_n^1 \\ \vdots \\ \Delta f_1^n \quad \Delta f_n^n \end{bmatrix}^{-1} \begin{bmatrix} \Delta x_i^1 \\ \vdots \\ \Delta x_i^n \end{bmatrix} \quad (14)$$

for $(i=1,2,\ldots n)$ where $\Delta f_k^j = \Delta f_k^j - \Delta f_k^{n+1}$ $(j,k=1,2,\ldots n)$ and $\Delta x_i^j = x_i^j - x_i^{n+1}$ $(j=1,2,\ldots n)$. Note that the improved form of the secant, Eq. (14), is in the form of an update to x_i^{n+1} and that a percentage of the update can be used to control convergence.

Relation to Newton's Method

The relation of the secant method to Newton's method is easy to see from Eq. (14). Let us write the Δf_j^i quantities as

$$\Delta f_j^i = \frac{\partial f_j}{\partial x_1} \Delta x_1 + \ldots + \frac{\partial f_j}{\partial x_n} \Delta x_n \quad (i,j=1,\ldots n) \quad (15)$$

and then rewrite this relation in matrix form as

$$\begin{bmatrix} \Delta f_1^1 \ldots \Delta f_1^n \\ \vdots \\ \Delta f_n^1 \ldots \Delta f_n^n \end{bmatrix} = J \begin{bmatrix} \Delta x_1^1 \ldots \Delta x_1^n \\ \vdots \\ \Delta x_n^1 \ldots \Delta x_n^n \end{bmatrix} \quad (16)$$

where

$$J = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} \cdots \frac{\partial f_1}{\partial x_n} \\ \vdots \quad \vdots \\ \frac{\partial f_n}{\partial x_1} \quad \frac{\partial f_n}{\partial x_n} \end{bmatrix} = \begin{bmatrix} \Delta f_1^1 \ldots \Delta f_1^n \\ \vdots \\ \Delta f_n^1 \quad \Delta f_n^n \end{bmatrix} \begin{bmatrix} \Delta x_1^1 \ldots \Delta x_1^n \\ \vdots \quad \vdots \\ \Delta x_n^1 \ldots \Delta x_n^n \end{bmatrix}^{-1} \quad (17)$$

Newton's method in n dimensions using $\bar{x}^{n+1}$ as the base point can be written as

$$\bar{x}^{\text{new}} = \bar{x}^{n+1} - J^{-1} \bar{f}^{n+1} \quad (18)$$

where J is computed based on $\bar{x}^1$ through $\bar{x}^n$. If the form for J in Eq. (17) is substituted into Eq. (18), and the equation is transposed, we get

$$(\bar{x}^{\text{new}})^T =$$

$$(\bar{x}^{n+1})^T - (\bar{f}^{n+1})^T \begin{bmatrix} \Delta f_1^1 \ldots \Delta f_n^1 \\ \vdots \quad \vdots \\ \Delta f_1^n \ldots \bar{\Delta} f_n^n \; \nu' \end{bmatrix}^{-1} \begin{bmatrix} \Delta x_1^1 \ldots \Delta x_n^1 \\ \vdots \quad \vdots \\ \Delta x_1^n \ldots \Delta x_n^n \end{bmatrix} \quad (19)$$

However, Eq. (19) can be written as a form precisely like Eq. (14). Thus, the secant method is basically a Newton's method with numerical partials. The fact that good numerical partials require small intervals indicates that a good strategy for discarding $\bar{x}$'s might be to ignore the one farthest from x_i^{new}, providing a good definition of "far" can be found.

Least Squares Utilization of all Data

As soon as the minimum number of data sets has been attained and an iteration is made, an extra data set is available. Several strategies are then possible.

1) Discard the "worst" data set and use the new one in its place. This has been the policy heretofore.

2) Replace each data set *in turn* by the new data set and calculate another $\bar{x}^{\text{new}}$. This will lead to a number of candidate points for $\bar{x}^{\text{new}}$. For the case of a linear fit and one extra data set, n such points can be calculated. We then choose the "best" $\bar{x}^{\text{new}}$, $\bar{x}^b$, to be the one that minimizes the quadratic form

$$\phi \sum_{j=1}^{N} (x_i^j - x_i^b)^2 \quad (N = \text{candidate points}) \quad (20)$$

To do this we form

$$\frac{\partial \phi}{\partial \bar{x}^b} = 0 = -2 \sum_{j=1}^{N} (x_i^j - x_i^b) \quad (21)$$

Therefore

$$x_i^b = \frac{1}{N} \sum_{j=1}^{N} x_i^j \quad (22)$$

According to Eq. (22), the best estimate, x_i^b, in the least squares sense, is just the average of the candidate x_i.

Instead of finding the best estimate of x_i using candidates calculated from different sets of function coefficients, a very efficient algorithm for handling large numbers of excess points can be developed by finding the "best" set of function coefficients in a least squares sense and then using these to find a unique x_i^b. The equations relating $\bar{f}$ and $\bar{x}$, when available, formed as $k > n$ data sets, are

$$\begin{bmatrix} \Delta f_1^1 \\ \Delta f_1^2 \\ \vdots \\ \Delta f_1^k \end{bmatrix} = \begin{bmatrix} \Delta x_1^1 \quad \Delta x_2^1 \\ \Delta x_1^2 \quad \Delta x_2^2 \\ \vdots \quad \vdots \\ \Delta x_1^k \quad \Delta x_2^k \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \quad (23)$$

and

$$\Delta f_2 = \Delta x A_2 \quad (24)$$

where

$$\Delta x = \begin{bmatrix} \Delta x_1^1 \quad \Delta x_2^1 \\ \Delta x_1^2 \quad \Delta x_2^2 \\ \vdots \quad \vdots \\ \Delta x_1^k \quad \Delta x_2^k \end{bmatrix} \quad \Delta f_2 = \begin{bmatrix} \Delta f_2^1 \\ \Delta f_2^2 \\ \vdots \\ \Delta f_2^k \end{bmatrix} \quad A_2 = \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \quad (25)$$

We now wish to choose the function coefficients so as to minimize

$$\phi_i = [\Delta f_i - \Delta x A_i]^T [\Delta f_i - \Delta x A_i] \quad (i=1,\ldots,n) \quad (26)$$

$$\frac{\partial \phi}{\partial A_i} = 0 = -2[\Delta f_i - \Delta x A_i]^T$$

$$\Delta x = \Delta f_i^T \Delta x - A_i^T \Delta x^T \Delta x \quad (27)$$

The coefficients satisfying Eq. (27) are

$$\begin{bmatrix} a_1 \quad a_2 \\ b_1 \quad b_2 \end{bmatrix} = [\Delta x^T \Delta x]^{-1} \Delta x^T \begin{bmatrix} \Delta f_1^1 \quad \Delta f_2^1 \\ \vdots \quad \vdots \\ \Delta f_1^k \quad \Delta f_2^k \end{bmatrix} = J^T \quad (28)$$

If $k=n$, this reduces to Eq. (14). This last method seems to have several advantages. First of all, only one inverse is required instead of one for each candidate x_i. Second, and more importantly, this form is not likely to suffer from ill-conditioned inverses.

4) Since the best estimate of J is not necessarily the inverse of the best estimate of J^{-1}, a way of utilizing the two

estimates is examined. Define χ and Φ as

$$\chi \equiv \left[\begin{Bmatrix} \Delta x_1 \\ \vdots \\ \Delta x_n \end{Bmatrix}_1 \cdots \begin{Bmatrix} \Delta x_1 \\ \Delta x_n \end{Bmatrix}_m \right] \qquad \Phi = \left[\begin{Bmatrix} f_1 \\ \vdots \\ f_n \end{Bmatrix}_1 \cdots \begin{Bmatrix} f_1 \\ \vdots \\ f_n \end{Bmatrix}_m \right] \tag{29}$$

where $m > n$.

If the Jacobian, J, exists then we should have

$$\Phi = J\chi \tag{30}$$

Define ϵ^{-T} as

$$\bar{\epsilon}^T \equiv J\chi - \Phi \tag{31}$$

and choose J to minimize

$$I \equiv \bar{\epsilon}^T \bar{\epsilon} = (J\chi - \Phi)(\chi^T J^T - \Phi^T) \tag{32}$$

The best estimate of J, $\hat{J}$, satisfies

$$J\chi\chi^T - \Phi\chi^T = 0 \tag{33}$$

Thus

$$\hat{J} = \Phi\chi^T(\chi\chi^T)^{-1} \tag{34}$$

and

$$\hat{J}^{-1} = (\chi\chi^T)(\Phi\chi^T)^{-1} \tag{35}$$

Define P as the best estimate of J^{-1}, i.e., $P = \hat{J}^{-1}$. Now let us define $\bar{\gamma}^T$ as

$$\bar{\gamma}^T \equiv P\Phi - \chi \tag{36}$$

Now let us choose P so as to minimize the product $\bar{\gamma}^T\bar{\gamma}$. The best estimate of P, $\hat{P}$, is given by

$$\hat{P} = (\chi\Phi^T)(\Phi\Phi^T)^{-1} \tag{37}$$

Note that if $m = n$ both P and J^{-1} reduce to

$$P = J^{-1} = \chi\Phi^{-1} \tag{38}$$

but otherwise, they are different. The extent of the difference is a measure of the correlation of the data. In fact, the product of $(\hat{J})$ and $(\hat{J}^{-1})$ can be thought of as a generalized correlation coefficient for the data involved. The correlation matrix is

$$C = (\Phi\chi^T)(\chi\chi^T)^{-1}(\chi\Phi^T)(\Phi\Phi^T)^{-1} \tag{39}$$

If the matrices are square and invertible C reduces to the identity matrix. If all the data lie exactly in an n-dimensional plane, C will again reduce to the identity matrix. If the data are scattered out of an n-dimensional plane, C will not reduce to the identity matrix, and the extent of the difference is a measure of the scatter. The least squares estimates of J^{-1}, given two separate estimates of J^{-1}, is the average of the two. Therefore, the best we can do is

$$(\hat{J}^{-1}) = (\Delta X^T/2)[\Delta X(\Delta F^T\Delta x)^{-1} + \Delta F(\Delta F^T\Delta F)^{-1}] \tag{40}$$

Sample Problem

The example problem chosen was a two stage lifting boost trajectory to a 50 naut mile circular orbit starting from a 60° launch azimuth at Cape Canaveral. There was no terminal constraint on azimuth, but full three-dimensional control was available on both stages. The Earth was modeled as an oblate spheroid rotating about its polar axis at a constant rate. A Patrick 1963 model atmosphere which rotated with the Earth (no winds) was employed.

All turns were assumed to be coordinated (no sideslip) and the thrust vector was assumed to be coincident with the vehicle's longitudinal axis. A 3g limit was employed on both stages to limit the axial acceleration. This limit forced throttling in each stage. The equations of motion for the boost vehicle in nonthrottling flights are

$$\dot{\bar{v}} = \left[\frac{T_v}{m} - \frac{A_e}{m}p_a + q\frac{S}{M}(b+c)\right]\hat{c} - q(S/m)(b\cos\alpha + a)\hat{V}_R + \bar{g}$$

$$\dot{\bar{r}} = \bar{v}$$

$$\dot{\mu} = -(\sigma g_a/T_v)m$$

In the equations above, $\bar{r}$ is position, $\bar{v}$ is velocity, μ is a mass parameter to be explained below, T_v is vacuum thrust, A_e is exit area, m is mass, p_a is ambient atmospheric pressure, $\hat{c}$ is a unit vector along the body axis, q is dynamic pressure, S is reference area, and $\hat{V}_R$ is a unit vector along the direction of the vehicle velocity relative to the atmosphere. The quantities, a, b, and c are tabular aerodynamic coefficients which are functions of the Mach number, which give aerodynamics representative of the space shuttle. Also, $\bar{g}$ is the gravitational acceleration vector, σ is fuel mass flow rate through the engine, and $\bar{g}_a$ is the axial acceleration limit. The seventh state variable, μ, is defined as the fuel plus the payload so that $m = \mu + m_a$ where m_a is the sum of all masses jettisioned during the flight. By this artifice the seventh state variable, μ is made to be continuous at all times, including those times at times which mass is discontinuous.

The control variables are σ, $\hat{c}$, and $\hat{V}_R$. The unit vector $\hat{c}$ (along the vehicle centerline) and $\hat{V}_R$ (along the relative wind direction) are defined by inertial angles references to launch site coordinates at liftoff. For further details of the simulation, the reader should refer to Ref. 17.

In order to obtain starting values for the secant procedure which were within its envelope of convergence, a min-H procedure was employed. Fifteen iterations were sufficient to get within the convergence envelope of the secant. The

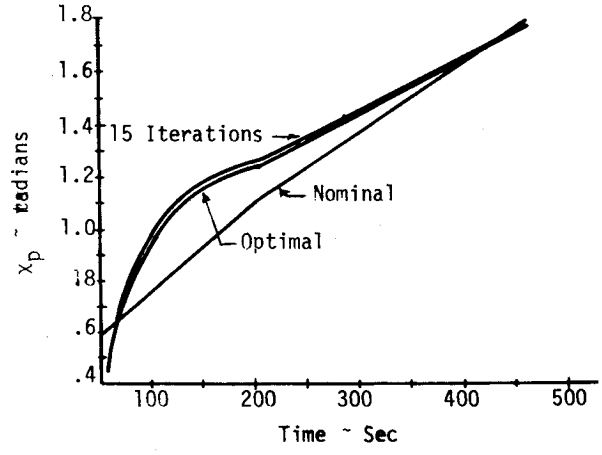

Fig. 1 χ_p time histories.

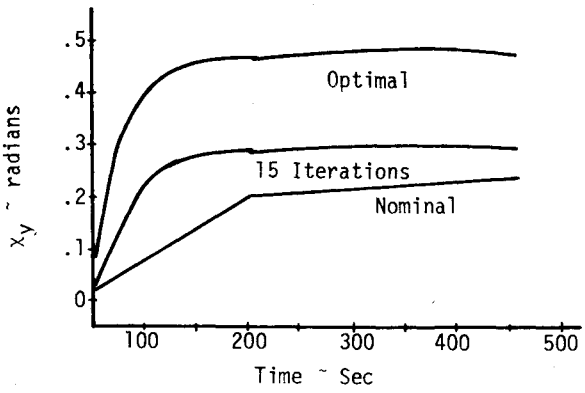

Fig. 2 χ_y **time histories.**

controls iterated upon were the two inertial angles which defined $\hat{c}$ and $\hat{V}_R$, χ_p and χ_y (χ-pitch and χ-yaw). The initial guesses for χ_p and χ_y, the min-H result after fifteen iterations, and the optimal values for an example boost trajectory are shown in Figs. 1 and 2. The min-H results had not converged at fifteen iterations and were continuing to improve the performance index at the time that the min-H process was stopped and the secant procedure was initiated.

The function $\bar{f}(\bar{x})=0$ used in the secant was a 7-vector defined as follows.

$$f(1)=v-7848.5949 \text{ m/sec}=0$$

$$f(2)=\bar{r}\cdot\bar{v}=u\cdot x+v\cdot y+w\cdot z=0$$

$$f(3)=r-6470762.\text{m}=0$$

$$f(4)=v\lambda^w-w\lambda^v+y\lambda^z-z\lambda^y=0$$

$$f(5)=w\lambda^u-u\lambda^w+z\lambda^x-x\lambda^z=0$$

$$f(6)=u\lambda^v-v\lambda^u+x\lambda^y-y\lambda^x=0$$

$$f(7)=\bar{\lambda}^v\cdot\dot{\bar{v}}+\bar{\lambda}^r\cdot\dot{\bar{r}}+\lambda^m\dot{m}=0$$

In the simulation the final value of the Hamiltonian was included in the constraint set, and λ_0^m was varied by the secant process. In each of the terminal necessary conditions, sums of terms are required to be zero. The largest component part of each sum, in absolute value, was divided into each $f(i)$ thus normalizing each $f(i)$ to 1. For example

$$f(4)=f(4)/\max\{|v\lambda^w|,\ |w\lambda^v|,\ |y\lambda^z|,\ |z\lambda^y|\}$$

In this way the closeness of each error term to zero can be monitored in terms of comparable significant digits. A composite error term E was defined as

$$E=\sum_{i=1}^{7}|f(i)|$$

The "best" point was defined as having the smallest value of E, the "worst" data point was defined as having the largest value of E.

The secant process begins by running a nominal and then varying each λ_i and t_f by a small amount in order to construct as nearly as possible a good Jacobian matrix, J, where

$$J=\partial f/\partial(\bar{\lambda}_0,t_f)$$

The first update to the variables is possible after the eighth trajectory. Two excess data points were accumulated for the least squares process, and the number of data sets was then limited to 10 with "worst" points thereafter being discarded. The average of two estimated Jacobians was employed in subsequent iterations.

The secant program was initiated with the initial conditions given earlier, taking t_f to be 456.8 sec. The terminal $\hat{\epsilon}$ vector resulting from the nominal choice of $\bar{\lambda}_0$ and t_f was

$$(\bar{f})^T=[.013644,\ -.009440,\ -.322791,$$

$$-.0470097,\ .0875401,\ .0030747,\ .0656752]$$

The $\delta\bar{\lambda}_0,\ dt_f$ vector with which the initial Jacobian was constructed was

$$[\delta\lambda^u\delta\lambda^w\delta\lambda^x\delta\lambda^y\delta\lambda^z\delta\lambda^m dt_f]=$$

$$[10^{-3}10^{-4}10^{-5}10^{-4}10^{-5}10^{-4}10^{-3}]$$

The convergence criteria used for f was

$$|(\bar{f})^T|<[10^{-4}10^{-4}10^{-4}10^{-3}10^{-3}10^{-3}10^{-2}]$$

Convergence was obtained after a total of 20 trajectories were run, including the eight necessary to start the process. The final value of f was

$$(\bar{f})^T=[-.788\times10^{-5},.148\times10^{-5},-.503\times10^{-5},$$

$$.911\times10^{-5},-.212\times10^{-5},.616\times10^{-6},.344\times10^{-5}]$$

In order to avoid variations that in some sense are too large, the variations in the parameter set were scaled so that the variation in λ^u, in absolute value, did not exceed .01. Obviously a great many other criteria for scaling can be imagined. However, this seemed to work reasonably well for this simulation. Table 1 contains a summary of the 20 trajectories required to converge this problem. The second column contains the composite error function for each trajectory, and the third column contains the scale factor necessary to insure $|\delta\lambda^u|\leq.01$. Since at least eight trajectories are required before the first update can be made, the scale required to insure $|\delta\lambda^u|\leq.01$ is not applicable for the first eight trajectories. The converged $\bar{\lambda}$ at t_T is given

$$\bar{\lambda}^v=\begin{vmatrix}0.515764\\1.\\0.16316\end{vmatrix}\qquad\bar{\lambda}^r=\begin{vmatrix}.00012609\\.026029\\.000086608\end{vmatrix}\qquad\lambda^m=.0040149$$

Table 1 Values of trajectory number, E, and scale

Trajectory number	E	Scale
1	.549	-
2	.509	-
3	.548	-
4	.548	-
5	.551	-
6	.543	-
7	.543	-
8	.549	-
9	.625	.76
10	.609	1.
11	.289	.357
12	.104	.845
13	.0814	1.
14	.0527	1.
15	.0158	1.
16	.00269	1.
17	.00126	1.
18	.00772	1.
19	.00452	1.
20	.0000453	1.

The values can be compared to the nominal values given previously. The converged trajectory resulted in a payload improvement of 330 kg out of approximately 170,000 kg. The plots for the secant converged values of χ_p and χ_y are the optimal values shown in Figs. 1 and 2.

If the secant process appears to start diverging at any point due to the stringing out of data points and subsequent loss of rank in the coefficient matrix, the process can be restarted merely by looping back through the initialization loop. In fact, if this is done after each iteration, a standard batch processed numerical Newton's method results. Therefore, any secant routine with minor logic modifications contains a numerical Newton method as a special case and hence can be made to approximate the results obtainable with the "perturbation method."

Conclusions

An updated version of the secant method has been presented in which an excess data point is kept and a least squares procedure is used to approximate the Jacobian. The relation and near equivalency of the secant to Newton's method is shown. The least squares procedure seems to aid convergence and makes the procedure less likely to encounter ill conditioned matrices during the iteration process. In addition, if an ill conditioned matrix is encountered, the logic can be programmed to add additional trajectories to those being used (one at a time) until the ill conditioning is eliminated. The improvements presented will hopefully make the secant method a more reasonable optimization procedure to use.

References

[1]Bryson, A.E. and Denham, W.F., "A Steepest-Ascent Method for Solving Optimum Programming Problems," *Journal of Applied Mechanics,* Vol. 29, 1962, pp. 247–247.

[2]Denham, W.F., "Steepest Ascent Solution of Optimal Programming Problems," Ph.D. Dissertation, Harvard University, 1973.

[3]Gottlieb, R.G., "Rapid Convergence to Optimum Solutions Using a Min-H Strategy," *AIAA Journal,* Vol. 5, Feb. 1967, pp. 322–329.

[4]Kelly, H.J., Kopp. R.E., and Moyer, H.G., "Successive Approximation Techniques for Trajectory Optimization," *Proceedings of the Vehicle Systems Optimization Symposium IAS,* New York, 1961, pp. 10–25.

[5]Speyer, J.L., Kelly, H.J., Levine, N., and Denham, W.F., "Accelerated Gradient Projection Technique with Application to Rocket Trjaectory Optimization," *Automatica,* Vol. 7, Pergamon Press, New York, 1971, pp. 37–43.

[6]Stancil, R.T., "A New Approach to Steepest Ascent Trajectory Optimization," *AIAA Journal,* Vol. 2, Aug. 1964, pp. 1365–1370.

[7]Breakwell, J.V., Speyer, J.L., and Bryaon, A.E., "Optimization and Control of Nonlinear Systems Using the Second Variation," *SIAM Journal of Control,* Vol. 1, 1963, pp. 193–223.

[8]Jazwinski, A.H., "Optimal Trajectories and Linear Control of Non-Linear Systems," *AIAA Journal,* Vol. 2, Aug. 1964, pp. 1371–1379.

[9]Kelly, J.J., Kopp, R.E., and Moyer, H.G., "A Trajectory Optimization Technique Based on the Theory of the Second Variation," *AIAA Progress in Astronautics and Aeronautics: Celestial Mechanics and Astrodynamics,* Vol. 14, edited by V.G. Szebehely, Academic Press Inc., New York, 1964, pp. 559–582.

[10]McGill, R and Kenneth, P., "Solution of Variational Problems by Means of Generalized Newton-Raphson Operator," *AIAA Journal,* Vol. 2, Oct. 1964, pp. 1761–1766.

[11]Tapley, B.D. and Lewallen, J.M., "Comparison of Several Numerical Optimization Methods," *Journal of Optimization Theory and Application,* Vol. I, July 1967, pp. 1–32.

[12]Kopp, R.E. and Moyer, H.G., *Trajectory Optimization Techniques, Advances in Control Systems*, Vol. 4, edited by C.T. Leondes.

[13]Burrows, R.R. and McDaniel, G.A., "A Method of Trajectory Analysis with Multi-Mission Capability and Guidance Application," AIAA Paper 68-844, Pasadena, Calif., 1968.

[14]Wolfe, P.H., "The Secant Method for Simultaneous Nonlinear Equations," *Communications of ACM,* Vol. 2, Dec. 1959.

[15]Martin, D.T., Sievers, R.F., O'Brian, R.M., and Rice, A.F., "Saturn V Guidance, Navigation and Targeting," *Journal of Spacecraft and Rockets,* Vol. 4, July 1967, pp. 891–898.

[16]Gottlieb, R.G. and Fowler, W.T., "Three-Dimensional *g*-Constrained Shuttle Boost Trajectories Using a Min-H/Secant Hybrid Algorithm," AAS Paper 75-034 presented at the AAS/AIAA Astrodynamics Specialist Conference, Nassau, Bahamas, 1975.

[17]Gottlieb, R.G., "Optimal Three-Dimensional Boost Trajectories for Shuttle-Type Vehicles," Ph.D. Dissertation, The University of Texas at Austin, Austin, Texas, Dec. 1972.