AIAA PAPER
NO. 76-153

IMPROVED CONVERGENCE OF GRADIENT-TYPE METHODS
INVOLVING FREE FINAL TIME AND PENALTY FUNCTIONS

by
W. F. POWERS and C. J. SHIEH
The University of Michigan
Ann Arbor, Michigan

AIAA PAPER 76-153

AIAA 14th Aerospace Sciences Meeting

WASHINGTON, D.C. / JANUARY 26-28, 1976

M76-10731

IMPROVED CONVERGENCE OF GRADIENT-TYPE METHODS INVOLVING FREE FINAL TIME AND PENALTY FUNCTIONS*

W. F. Powers+ and C. J. Shieh++
The University of Michigan
Ann Arbor, Michigan

Abstract

Gradient-type methods which employ penalty functions typically have difficulties in solving optimal control problems with variable final time. Methods for improving the rate of convergence are presented. The first approach involves simply guaranteeing that the initial iterate final time is less than the optimal final time. Heuristic reasons for this approach are presented along with a simple example which illustrates why such an approach is effective. The second approach involves the use of a two-dimensional search routine, with a final time parameter and the control vector correction length as the parameters. The number of function evaluations is reasonable because of the availability of $J(0,0)$, $J_\alpha(0,0)$, and $J_T(0,0)$, where $J(\alpha, T)$ is the two-parameter function to be minimized, and since a recursive quadratic surface fitting procedure is employed. The approach is illustrated on two aerospace trajectory optimization problems.

1. INTRODUCTION

When a gradient-type technique is employed to solve an optimal control problem, one must continually confront the problems associated with terminal equality constraints and variable final time. Usually for flexibility and ease of programming, a penalty function approach is preferred to a projected gradient approach, especially if an accelerated gradient technique such as the conjugate gradient method is employed. However, numerous investigators have found that accelerated gradient methods which employ penalty functions have serious difficulties with tightly constrained problems, especially those involving variable final time. In Ref. 1 it was noted that problems involving three or more terminal equality constraints were extremely difficult to solve with the conjugate gradient method. In Ref. 2, which involved a comparison of various optimization techniques, it was shown that the standard conjugate gradient method was the only method of those compared which could not solve a minimum time orbital transfer problem with three terminal equality constraints.

A number of approaches have been employed to improve the rate of convergence of methods which involve variable final time and/or penalty functions. Hestenes[3] proposed the multiplier method to overcome the ill-conditioning associated with large penalty function coefficients, and a time-optimal problem was solved successfully with this method in Ref. 4. References 5 and 6 treat variable final-time problems by employing the transformation introduced by Long[7]. Moyer[6] employs this idea along with a modified gradient procedure to successfully solve a time-optimal atmospheric flight problem.

In this paper two approaches for improving the convergence of free final time problems which utilize gradient-type methods with penalty functions are presented. In Section 2 a relatively simple approach involving the initial final time estimate and Long's transformation is presented. Heuristic reasons for this approach are presented along with a simple example which illustrates why such an approach is effective. In Section 3 the approach of Section 2 is augmented by an efficient two-dimensional search procedure. The resultant procedure typically requires only a few more function evaluations per iterate than the standard one-dimensional search, while improving the operational rate of convergence considerably. Section 4 presents simulations of the method on an orbital transfer problem and an atmospheric flight problem, both involving free final time and penalty functions.

2. TERMINAL TIME CONSIDERATIONS

Consider the optimal control problem

$$\text{Minimize: } \tilde{J} = \tilde{\phi}(t_f, x_f) + \int_{t_o}^{t_f} \tilde{L}(t, x, u)\, dt \tag{1}$$

$$\text{Subject to: } \frac{dx}{dt} = \tilde{f}(t, x, u), \quad x(t_o) = x_o \tag{2}$$

$$u \in U \tag{3}$$

where x is an n-vector, u is an m-vector, and U is the set of admissible controls. If terminal constraints are present, it is assumed that they have been incorporated into the function $\phi(t_f, x_f)$ by the method of penalty functions.

Since variable $-t_f$ problems are of primary interest here, the following parameterization due to Long[7] is employed to alleviate some of the difficulties involved with variable t_f, especially extrapolation problems. (References 4, 5, and 7 discuss these aspects). Let s be the new independent variable defined by

$$t = Ts + t_o, \quad T = t_f - t_o \quad s \in [0, 1] \tag{4}$$

*This work was supported by the National Science Foundation under Grant ENG 74-21618.

+Associate Professor, Department of Aerospace Engineering

++Graduate Student, Department of Aerospace Engineering

Usually $t_o = 0$ and Eq. (4) becomes

$$t = T s, \; T = t_f, \; s \in [0,1]. \tag{5}$$

Without loss of generality, $t_o = 0$ will be assumed in the following developments. In this case Eqs. (1) and (2) are transformed into

$$J = \phi(T, x_f) + \int_0^1 T L(T s, x, u) \, d s \tag{6}$$

$$\dot{x} = T f(T s, x, u), \; x(0) = x_o \tag{7}$$

$$(\dot{\ }) \equiv \frac{d(\ )}{d s}, \tag{8}$$

where T is a parameter to be optimized along with the control vector u.

Let $T^{(0)}$ be the initial estimate for T in a gradient-type iteration scheme. Our experience has shown that the relation of $T^{(0)}$ to the optimal value of T, say T^*, can greatly affect the rate of convergence of the algorithm, especially when penalty functions are present in $\phi(t_f, x_f)$. In particular, we have found that $T^{(0)} < T^*$ gives more rapid convergence than $T^{(0)} > T^*$ in all cases simulated to date. Although a mathematical proof of this approach is not available, heuristic reasons for its effectiveness are given below.

Consider a time optimal control problem with terminal constraints which is treated by the method of penalty functions. Suppose $T^{(0)} < T^*$. Then, it is impossible for the initial trajectory to meet the boundary conditions, and $T^{(1)}$ must be greater than $T^{(0)}$ to decrease the error on the terminal constraints. Thus, the optimal solution has the unique characteristic of being the closest trajectory to the initial iterate, with respect to final time, which satisfies the terminal constraints. Conversely, if $T^{(0)} > T^*$, then there exist infinitely many nearby solutions which satisfy the terminal constraints. Since with penalty functions terminal constraint satisfaction is a major part of the performance index, there exists the tendency to "lock-in" on the terminal conditions at $T > T^*$. That is, the optimal solution no longer possesses the unique property of being the closest trajectory which satisfies the boundary conditions. Apparently this lack of uniqueness affects adversely the rate of convergence.

With respect to the relationship of $T^{(0)}$ and T^*, it is instructive to consider the convergence characteristics of the following variant of Zermelo's problem which contains both singular and bounded control subarcs:

$$\text{Minimize: } J = T^2 + P[x(1) - 4]^2 + P[y(1) - 3]^2 \tag{9}$$

$$\text{Subject to: } \begin{aligned} \dot{x} &= T\cos\theta, & x(0) &= 0 \\ \dot{y} &= T\sin\theta, & y(0) &= 0 \\ \dot{\theta} &= T u, & \theta(0) &= 0 \\ |u| &\le 0.5 \end{aligned} \tag{10}$$

(This is a penalty function approximation to the time optimal transfer to $x_f = 4$, $y_f = 3$ problem; T^2 is used in the performance index because with T-linear in the performance index along with penalty functions, $T \rightarrow -\infty$ is a possibility which is to be avoided).

This problem was solved numerically with a bounded control version of the conjugate gradient method [5, 8] for various initial estimates of $T^{(0)}$. It was found that a useful way of studying the convergence characteristics for the various initial estimates is to plot the gradient with respect to $T^{(N)}$ versus $T^{(N)} - T^*$ for $N = 0, 1, 2, \ldots$ (where N ≡ iterate number). Such a plot is shown in Figure 1 for a given set of penalty coefficients. (Three initial control estimates, i.e., $u^{(0)}(s)$, were employed so that points in all four regions of the plane could be generated. Trajectories 1, 2, 5, and 6 employ $u^{(0)}(s) = 0.5(1-s)$, Trajectories 3 and 4 employ $u^{(0)}(s) = 0.5$, and, Trajectories 7 and 8 employ $u^{(0)}(s) = 0.05 + 0.45s$.)

Trajectories with initial points in Regions I and IV have the tendency to "lock-in" at points where $T^{(N)} - T^* > 0$ and nonoptimal trajectories are obtained. In Region II, the trajectories do not converge directly; however, they move to Region III and converge rapidly thereafter. In Region III the trajectories converge rapidly (even with a physically unreasonable $T^{(0)} < T^*$, such as Trajectory No. 6). Thus, Figure 1 shows (at least for this example) that the value of $T^{(0)}$ is operationally cirtical to the rate of convergence of the scheme. Such a choice appears to allow one to set the penalty coefficients relatively high initially, which avoids the problems associated with raising the penalty terms during the iterative process.

3. A TWO-DIMENSIONAL SEARCH ALGORITHM

The use of Long's transformation and requiring $T^{(0)} < T^*$ are simple procedures which improve the convergence of variable final time gradient algorithms. However, convergence is sometimes still slow because of the sensitivity of the gradient $\partial J/\partial T$, which is a direct function of the penalty coefficients. Thus, further improvements might be possible by optimizing the parameter T directly as opposed to using the possibly sensitive or misleading gradient $\partial J/\partial T$. At first glance this does not appear to be a promising idea because the algorithm then contains two free parameters, i.e., T and the length of the control (u) search direction. However, it will be shown in this section that after the first iterate there exists enough information about the function of two parameters to make a two-dimensional search feasible. Furthermore, in the simulations of Section 4 the two-dimensional search requires only a few more function evaluations per iterate than a corresponding one-dimensional search while improving considerably the convergence rate.

For the problem defined by Eqs. (6) - (8), define

$$\ell(s,x,u,T) \equiv T\,L\,(T\,s,x,u) \quad (11)$$

$$F(s,x,u,T) \equiv T\,f\,(T\,s,x,u) \quad (12)$$

$$H(s,x,u,T,\lambda) \equiv \ell(s,x,u,T) + \lambda^T F(s,x,u,T) \quad (13)$$

The basic steps for a gradient-type algorithm with a two-dimensional search are as follows:

BASIC ALGORITHM: Specify $u^{(0)}(s)$, $s \in [0,1]$, $T(0)$.

1. Given $u^{(N)}(s)$, $T^{(N)}$, integrate Eq. (7) forward from $s = 0$ to $s = 1$. Define

$$\lambda^{(N)}(1) = \partial\,\phi/\partial x_f\,\Big|^{(N)}, \quad (14)$$

and integrate

$$\dot{\lambda}_i = -\partial H/\partial x_i \;(i = 1, \ldots, n) \quad (15)$$

backwards with $\lambda^{(N)}(1)$ to form $\lambda^{(N)}(s)$. Calculate and store the gradient $\partial H/\partial u\,|^{(N)}$.

2. Two-Dimensional Search: With

$$u^{(N+1)}(s) = u^{(N)}(s) - \alpha^{(N)}\,p^{(N)}(s) \quad (15)$$

$$\Delta T^{(N)} = T^{(N+1)} - T^{(N)} \quad (16)$$

perform a two-dimensional search to determine the values of $\alpha^{(N)} > 0$ and $T^{(N+1)}$ which minimize

$$J^{(N+1)}(\alpha^{(N)}, \Delta T^{(N)}) = \phi\,(T^{(N+1)}, x_f^{(N+1)}) + \int_0^1 T^{(N+1)}\,L(T^{(N+1)}s, x^{(N+1)}, u^{(N+1)})\,ds \quad (17)$$

3. Check convergence criterion, e.g., $|J^{(N+1)} - J^{(N)}| \le \epsilon$. If yes, stop; if no, set $N = N + 1$ and return to Step 1.

In the algorithm above the search direction $p^{(N)}(s)$ determines which kind of method is employed, e.g., gradient, conjugate gradient, etc. The unique feature of the algorithm is Step 2, and this step requires a separate subroutine. Before listing the two-dimensional search algorithm, the approach to its development will be discussed.

Consider the problem of determining a minimum of a function of two parameters, say $J(\alpha, \Delta T)$, when only function and derivative data are available, e.g., $J_1 = J(\alpha_1, \Delta T_1)$, $J_2 = J(\alpha_2, \Delta T_2)$, $\partial J/\partial\alpha\,(\alpha_1, \Delta T_1)$, etc. In such a case, either a direct search or curve-fitting procedure is necessary to approximate the minimum. If a curve-fitting procedure is employed, then one must assume at least a quadratic function in the parameters:

$$\bar{J} = c_{00} + c_{10}\alpha + c_{01}\Delta T + c_{20}\alpha^2 + c_{11}\alpha\,\Delta T + c_{02}\Delta T^2 \quad (18)$$

To determine the coefficients c_{ij}, six data points concerning $J(\alpha, \Delta T)$ are necessary. However, in the trajectory optimization problem three data points are readily available after the first iterate

$$c_{00} = J(0, 0) \quad (19)$$

$$c_{01} = \frac{\partial J}{\partial\alpha}\Big|_{(0,0)} \equiv J_\alpha(0, 0) = -\int_0^1 H_u^{(N)}(s)\,p^{(N)}(s)\,ds \quad (20)$$

$$c_{10} = \frac{\partial J}{\partial T}\Big|_{(0,0)} \equiv J_T(0, 0) = \left(\phi_T + \frac{H_f}{T}\right)\Big|_{(0,0)} \quad (21)$$

The proofs of Eqs. (20) and (21) are presented in Appendix A.

With the information supplied by Eqs. (19) - (21), only three function evaluations are necessary to obtain the three remaining coefficients in Eq. (18), i.e., c_{20}, c_{11}, and c_{02}. Experience with one-dimensional search routines served as a guide for the choice of parameter values to use for the three evaluations.

First function evaluation: Compute

$$J^1 = J[\alpha_1^{(N)}, \Delta T_1^{(N)}]$$

with

$$\alpha_1^{(N)} = \alpha^{(N-1)}, \quad \Delta T_1^{(N)} = 0,$$

i.e., use the converged values of the two parameters from the previous iterate. (The value of the correction length α is in many cases of the same order of magnitude of the previous iterate).

Second function evaluation: Compute

$$J^2 = J[\alpha_2^{(N)}, \Delta T_2^{(N)}]$$

with

$$\alpha_2^{(N)} = 0, \quad T_2^{(N)} = T^{(N-1)} + \Delta T_2^{(N)},$$

where

$$\Delta T_2^{(N)} = -.01\,T^{(N-1)}\,sgn(\partial J/\partial\Delta T_{(0,0)}) \quad (22)$$

That is, select the second point along the T direction in the direction of a cost decrease.

Third function evaluation: With $J(0,0)$, $J_\alpha(0,0)$, and J^1, perform a quadratic fit for α while holding $\Delta T^{(N)} = 0$, which implies a value $\alpha_3^{(N)}$ for a minimum of $J(\alpha^{(N)}, 0)$. With $J(0,0)$, $J_T(0,0)$, and $J^2(0, \Delta T_2^{(N)})$, perform a quadratic fit for $\Delta T_3^{(N)}$ while holding $\alpha = 0$. A value ΔT_3 for a minimum of $J(0, \Delta T^{(N)})$ is obtained. Compute

$J^3 = J[\alpha_3^{(N)}, \Delta T_3^{(N)}]$.

If the fit is not successful or a more accurate fit is required, then more than three function evaluations are necessary. In such a circumstance, the points as shown in Fig. 2 will serve, respectively, as the 4th, 5th and 6th additional function evaluations (assuming sgn $(\partial J/\partial \Delta T_{(0,0)}) < 0$).

It should be noted that, for each additional function evaluation, the value of $\partial J/\partial \Delta T$ at the point where the function is evaluated is readily obtained (without additional integration) and can be employed as supplemental information in the search scheme. The proposed two dimensional search scheme is presented below, and a complete flow chart of this procedure is given in Fig. 3.

SURFACE FITTING SEARCH METHOD (SFSM)

Specify c_{00}, c_{10}, c_{01} as defined in Eq. (19)-(21).

1) Evaluate the first three functions in the manner noted above, and denote by J^i, $\partial J^i/\partial \Delta T = \partial J/\partial \Delta T[\alpha_i^{(N)}, \Delta T_i^{(N)}]$, $i = 1, 2, 3$.

2) Fit Eq. (18); if good, go to 3); if not, for $i = 1, 2, 3$ replace c_{01} by $J^i_{\Delta T}$ and fit (18); if good, go to 3); if not; select 4th point $(\alpha_4^{(N)}, 0)$ and go to 3).

3) Evaluate** J^4, and replace $J_{\Delta T}(0, 0)$ by J^4; fit (18); if good, go to 4); if not, for $i = 1, 2, 3, 4$, replace $J_\alpha(0, 0)$ by $J^i_{\Delta T}$ and fit (18); if good, go to 4); if not, select the fifth point $(0, \Delta T_2^{(N)}/2)$ and go to 4).

4) Evalute** J^5 and fit Eq. (18) with six data points; if good, go to 5); if not, go to 6)

5) Evalute** J^j, $j \geq 6$, compute and select the 6 points with the least costs of the j+1 data points; fit (18); if not good or if $|(J^j - J^{j-1})/J^{j-1}| < \epsilon$ go to 6), otherwise repeat 5) for higher j.

6) Select the point with the least cost and stop.

At least three function evaluations are necessary for a complete search. The number of points for the fit depends upon the accuracy that is required for a specific problem, and typically this search requires six to eight function evaluations. Mathematically, Eq. (18) is a quadratic surface which approximates $J[\alpha, T]$ in the neighborhood of the initial point $J[0, T^{(N-1)}]$. The fitting process should tend to be more consistent near the optimum, and this has been the experience in our simulations. In addition to the search mentioned above, a relatively stable procedure[11] has been developed for implementation as a backup routine. It can also be operated independently, but experience has indicated that it has slower convergence and requires more function evaluations than the method above.

** Note that when the surface fit is good, an approximation of the minimum of J is obtained and used as the next point to be evaluated.

With respect to convergence aspects of the algorithm, note that since J is minimized with respect to T on each iterate, the sequence of iterates possesses the property that $\frac{\partial J}{\partial T}|^{(N)} = 0$ (N=1, 2, . . .) (whereas with a one-dimensional search $\frac{\partial J}{\partial T}|^{(N)} \to 0$ as $N \to \infty$). This means that one of the transversality conditions for the optimal control problem is satisfied by each iterate, i.e.,

<u>PROPERTY:</u> Assuming t_f unspecified in the problem defined by Eqs. (1) - (3), $\tilde{\Phi}_{t_f} + \tilde{H}_f = 0$ (where $\tilde{H} = \tilde{L} + \lambda^T \tilde{f}$) is a necessary condition of optimality. If the two-dimensional search is employed, then $\tilde{\Phi}_{t_f} + \tilde{H}_f = 0$ on each iterate.

<u>Proof:</u> The optimization is performed on the transformed problem defined by Eqs. (5) - (8), and on each iterate $\partial J/\partial T = 0$. But, from Eq. (A-8) in Appendix A $\partial J/\partial T = \phi_T + \tilde{H}_f = 0$. Since $\phi_T = \tilde{\phi}_{t_f}$, the property is proved.

4. SIMULATION RESULTS

In this section examples are selected to demonstrate the performance of the algorithm. One of the examples, an orbital transfer problem which could not be solved by the conjugate gradient method in Ref. 2, is solved successfully with the proposed method.

Problem 1) Orbital Transfer Problem

$$\text{Minimize } J = t_f \tag{23}$$

Subject to

$$\begin{aligned} \frac{dx_1}{dt} &= x_2 \\ \frac{dx_2}{dt} &= x_3^2/x_1 - \mu/x_1^2 + \frac{\tau}{m_0 + \dot{m}t}\sin u \\ \frac{dx_3}{dt} &= -x_2x_3/x_1 + \frac{\tau}{m_0 + \dot{m}t}\cos u \end{aligned} \tag{24}$$

where x_i are the state variables, u is the pitch angle control, and μ, τ , m_0, $\dot{m}$ are constants. The boundary conditions are, after Long's transformation,

$$x_1(0) = 1,\ x_2(0) = 0,\ x_3(0) = 1 \tag{25}$$

$$x_1(1) = 1.525,\ x_2(1) = 0,\ x_3(1) = .8098 \tag{26}$$

These parameters correspond to a nondimensional Earth-Mars transfer with continuous thrust[2,10].

The problem was solved numerically by the conjugate gradient method with two different search routines. Method No. 1 is defined as the standard conjugate gradient method with a one dimensional search routine, while method No. 2 employs the proposed two-dimensional search. The resultant data are presented in Table 1.

Method No. 2 converged in nine iterates with errors of less than 0.2 percent (while the standard method, No. 1, did not converge). The ninth iterate control is very close to optimal (see Fig. 4), and the deviation from optimal is typical of function space gradient-type techniques. Method No. 1 performed smoothly in the first six iterations, however difficulties occurred in the improvement of the final time after the sixth iterate. A number of sets of penalty functions were tested in an attempt to move the final time toward t_f^*, but no significant improvement in t_f was obtained. From the data obtained here and in Ref. 2, it is evident that the standard conjugate gradient method has operational difficulty in converging to the optimum.

Problem 2) Minimum Time Turning Flight With Specified Range.

$$\text{Minimize} \quad J = t_f \tag{27}$$

Subject to:

$$x' = u \cos \psi$$

$$y' = u \sin \psi$$

$$u' = \frac{\eta}{m}\left[\tau - (1+\lambda^2)u^2\right] \tag{28}$$

$$\psi' = \frac{\sqrt{\lambda^2 u^4 - m^2}}{u\, m}$$

$$m' = -\eta' \tau$$

The equations represent the turning flight of a vehicle in a horizontal plane. Here, x and y are the longitudinal and lateral ranges, respectively. The remaining state variables are the velocity magnitude u, the heading angle ψ, and the mass m. The control variables are lift control $\lambda = C_L/C_L^*$, where C_L is the lift coefficient and C_L^* is the lift coefficient for maximum lift-to-drag ratio, and a nondimensionalized thrust variable, τ. Both controls are founded as follows

$$0 \le \lambda \le 3.066$$

$$0 \le \tau \le 9.059 \tag{29}$$

The constant parameter η is defined as $\eta = C_{DO}/C_L^*$ ($\doteq$ .181), where C_{DO} is the zero-lift drag coefficient (for more details see Ref. 11). The problem is to find the time-optimal controls from $t_0 = x_0 = y_0 = \psi_0 = 0$, $u_0 = .264$, $m_0 = .03914$ to $x_f = 0$, $y_f = .0525$, $m_f = .01973$.

With the initial control shown in Fig. 5(a), (b), and initial final time estimate $t_f^{(0)} = 0.1$, a maximum-variable-minimum thrust profile was obtained for the optimal trajectory after twenty iterations. It also appears that the lift control is tending to a maximum-variable-minimum type. The optimal time obtained is $t_f^{(20)} = .122$ ($\approx$ 30.5 seconds) with the relative terminal errors reduced to the range of 0.1 percent. The bank angle control corresponding to the optimal lift control and the optimal trajectory are shown in Fig. 6(a),(b), and relevant data of the computations are presented in Table 2.

A suboptimal solution with the thrust magnitude control of a maximum-minimum type was noted in the simulations. Such a suboptimal might be a more easily implemented control policy even though its performance is not as good as the optimal solution presented above(i.e., t_f = 34 seconds on the suboptimal as opposed to $t_f \approx$ 30.5 seconds on the optimal). The lift control for the suboptimal case is a maximum-minimum-maximum type.

5. CONCLUSIONS

Methods for the improved convergence of gradient-type algorithms on problems involving free final time and penalty functions are presented. A method involving the estimate and characterization of the final time parameter along with an efficient two-dimensional search procedure appears to improve the operational rate of convergence of gradient-type algorithms considerably. The method is coupled with the conjugate gradient algorithm to solve efficiently two aerospace trajectory optimization problems, one of which was not previously solvable with a conjugate gradient type algorithm.

APPENDIX A

PROOFS OF EQUATIONS (20) AND (21)

Given

$$J = \phi\,(T, x_f) + \int_0^1 [\, TL(Ts, x, u) + \lambda^T (Tf - \dot{x})]\, ds \tag{A-1}$$

$$H = T(L + \lambda^T f) = T\tilde{H} \tag{A-2}$$

$$H_u = T(L_u + f_u^T \lambda) \tag{A-3}$$

$$u^{(N+1)} = u^{(N)} - \alpha^{(N)} p^{(N)} \tag{A-4}$$

Subsitute (A-2), (A-4) into (A-1), and take the derivative of (A-1) with respect to α

$$\frac{\partial J}{\partial \alpha} = \int_0^1 T[(\frac{\partial L}{\partial u})^T \frac{\partial u}{\partial \alpha} + \lambda^T f_u \frac{\partial u}{\partial \alpha}]\, ds$$

$$= \int_0^1 T[L_u^T + \lambda^T f_u] \frac{\partial u}{\partial \alpha}\, ds$$

$$= \int_0^1 H_u^T (-p^{(N)})\, ds, \tag{A-5}$$

which is Eq. (20).

To develop Eq. (21), first differntiate Eq. (A-1):

$$\partial J/\partial T = \phi_T + \phi_{x_f}^T x_{f_T} + \frac{\partial}{\partial T}\int_0^1 [\, TL + \lambda^T (Tf - \dot{x})]\, ds. \tag{A-6}$$

But, $t_f = T$, $\phi_{x_f} = \tilde{\phi}_{x_f} = \lambda_f$, and $x_{f_T} = \dot{x}_f$, which upon substitution in Eq. (A-6) implies

$$\partial J/\partial T = \phi_T + \lambda_f^T \, dx_f/dt_f + \frac{\partial}{\partial T}\int_0^1 [TL + \lambda^T (Tf - \dot{x})] ds \qquad \text{(A-7)}$$

Since $t_f = T$ and $t = T s$:

$$\frac{\partial}{\partial T}\int_0^1 [TL + \lambda^T(Tf - \dot{x})] ds = \frac{\partial}{\partial t_f}\int_0^{t_f} [\tilde{L} + \lambda^T(\tilde{f} - \frac{dx}{dt})] dt$$

$$= \tilde{L}(t_f, x_f, u_f) + \lambda_f^T [\tilde{f}(t_f, x_f, u_f) - \frac{dx_f}{dt_f}]$$

where $\tilde{f} - dx/dt = 0$ at every point on the trajectory. Upon substitution into Eq. (A-7):

$$\frac{\partial J}{\partial T} = \phi_T + \lambda_f^T \tilde{f}(t_f, x_f, u_f) + \tilde{L}(t_f, x_f, u_f) = \phi_T + \tilde{H}_f \qquad \text{(A-8)}$$

and the desired result is obtained since $\tilde{H}_f = H_f/T$ by Eq. (A-2).

REFERENCES

1. Mehra, R.K., and Bryson, A.E., "Conjugate Gradient Methods with an Application to V/STOL Flight-Path Optimization", AIAA Journal of Aircraft, Vol. 6, No. 2, pp. 123-128, 1969.

2. Pu, C.L., "Numerical Solution of Dynamic Optimization Problems", Report No. E-2549, MIT Charles Stark Draper Lab., February 1971.

3. Hestenes, M.R., "Multiplier and Gradient Methods", Journal of Optimization Theory and Applications, Vol. 4, No. 5, pp. 303-320, 1969.

4. Tripathi, S.S., and Narendra, K.S., "Constrained Optimization Problems by the Multiplier Method", Journal of Optimization Theory and Applications, Vol. 9, No. 1, pp. 59-70, 1972.

5. Edge, E., Shieh, C.J., and Powers, W.F., "Techniques for Shuttle Trajectory Optimization"; University of Michigan, Report No. 011482-F, Chapter 5, for NASA Johnson Space Center, Contract NAS 9-12872, December 1973.

6. Moyer, H.G., "Improved Gradient Algorithm for Two-Point Boundary Value Problems", AIAA Journal, Vol. 3, No. 7, pp. 1351-1352, 1965.

7. Long, R.S., "Newton-Raphson Operator; Problems with Undetermined End Points", AIAA Journal, Vol. 3, No. 7, pp. 1351-1352, 1965.

8. Edge, E.R., and Powers, W.F., "Function Space Quasi-Newton Algorithms for Optimal Control Problems with Bounded Controls and Singular Arcs", to appear in Journal of Optimization Theory and Applications, 1975.

9. Busemann, A., Vinh, N.X., and Culp, R.D., "Optimum Maneuvers of Hypervelocity Vehicles Using Thrust and/or Aerodynamic Force", USAF Aerospace Research Laboratory Report No. ARL 74-0104, Vol. 4, 1974.

10. Moyer, G., and Pinkham, G., "Several Trajectory Optimization Techniques, Part II, Applications", in Computing Methods in Optimization Problems, edited by A. V. Balakrishnan, L. Neustadt, Academic Press, pp. 91-105, 1964.

11. Shieh, C.J., "Theory and Improved Gradient Type Algorithms for Optimal Atmospheric Trajectories", PhD Thesis, University of Michigan, 1975.

Method	Fcn. Eval. Per Iter.	CPU Time Per Iter.	No. of Iter.	Terminal Errors	t_f
1	5	0.766	17*	.0005 .1500 .0610	3.242
2	8	0.997	9	.0019 .0015 .0013	3.318

*Penalty coefficients were adjusted on 7th, 11th, 14th iterates in an attempt to improve convergence; $P_1 = P_2 = P_3 = 1000$ on all iterates in Method 2.

Table 1. Orbital Transfer Problem Results ($t_f^{(0)} = 2.5$, $t_f^* = 3.32$)

Iter. no.	Relative Δx_f	terminal Δy_f	errors(%) Δm_f	Cost	Final time
0*	18.4	87.2	38.9	21.80	.0980
1	7.1	42.2	16.3	5.119	.0976
2	12.9	38.1	18.1	4.307	.0976
3	18.5	24.7	4.05	1.887	.0995
4	34.6	6.67	15.6	0.647	.1001
5	24.0	2.67	13.1	0.355	.1101
9	3.03	0.47	6.13	0.136	.1185
10**	0.99	2.86	0.37	0.146	.1212
12	0.20	0.09	3.20	0.125	.1214
15	0.14	0.22	2.28	0.124	.1220
18	0.11	0.51	1.77	0.123	.1220
19+	0.43	0.53	1.67	0.123	.1219
20	0.12	0.22	0.17	0.123	.1219

(P_x, P_y, P_m) = *$(10^3, 10^3, 10^4)$; ** $(10^4, 10^4, 10^4)$ +$(10^4, 10^4, 10^5)$

Table 2. Computational Results for Problem No. 2

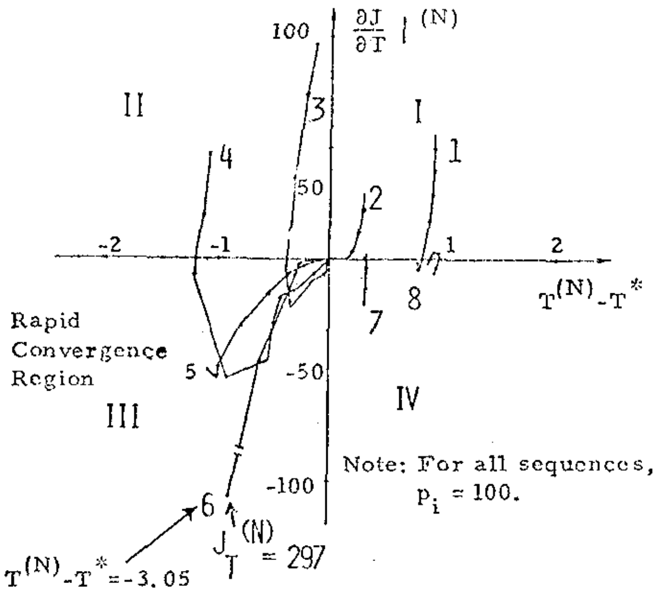


Figure 1. Convergence of Trajectory Sequences for Zermelo's Problem.

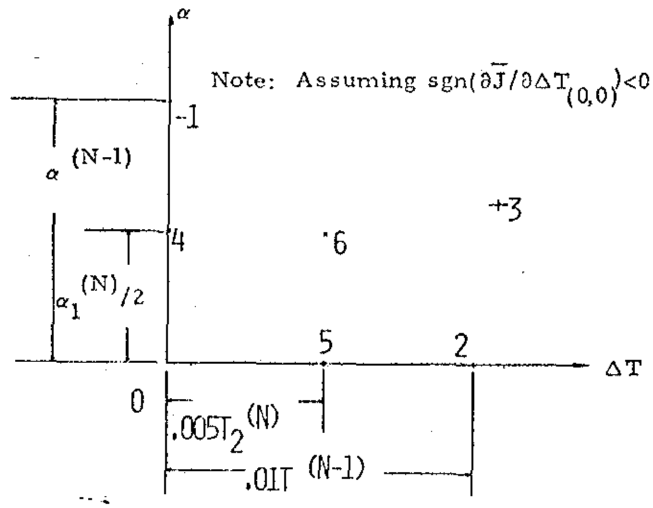


Figure 2. Configuration of Function Evaluations for Two-Dimensional Search Algorithm.

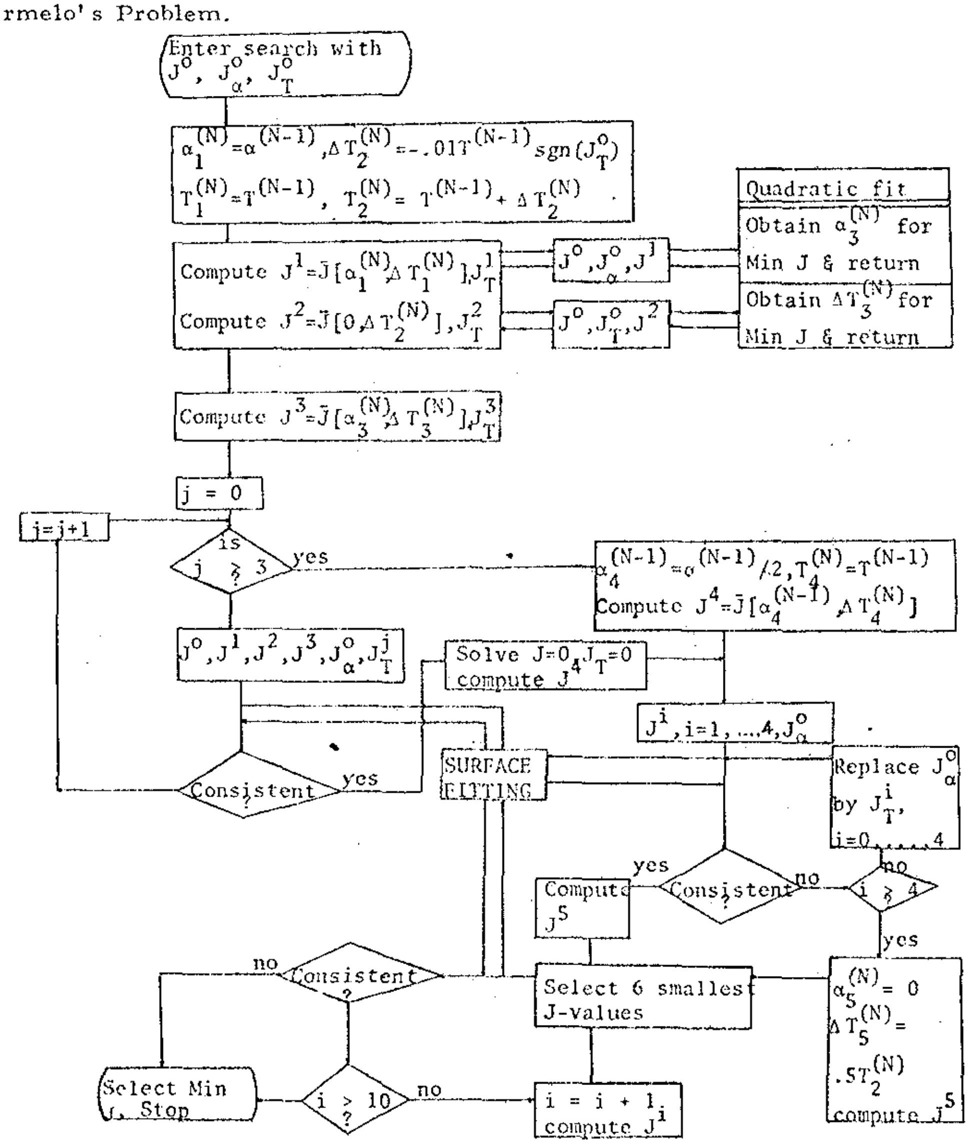


Figure 3. Flow Diagram of Two-Dimensional Search Algorithm

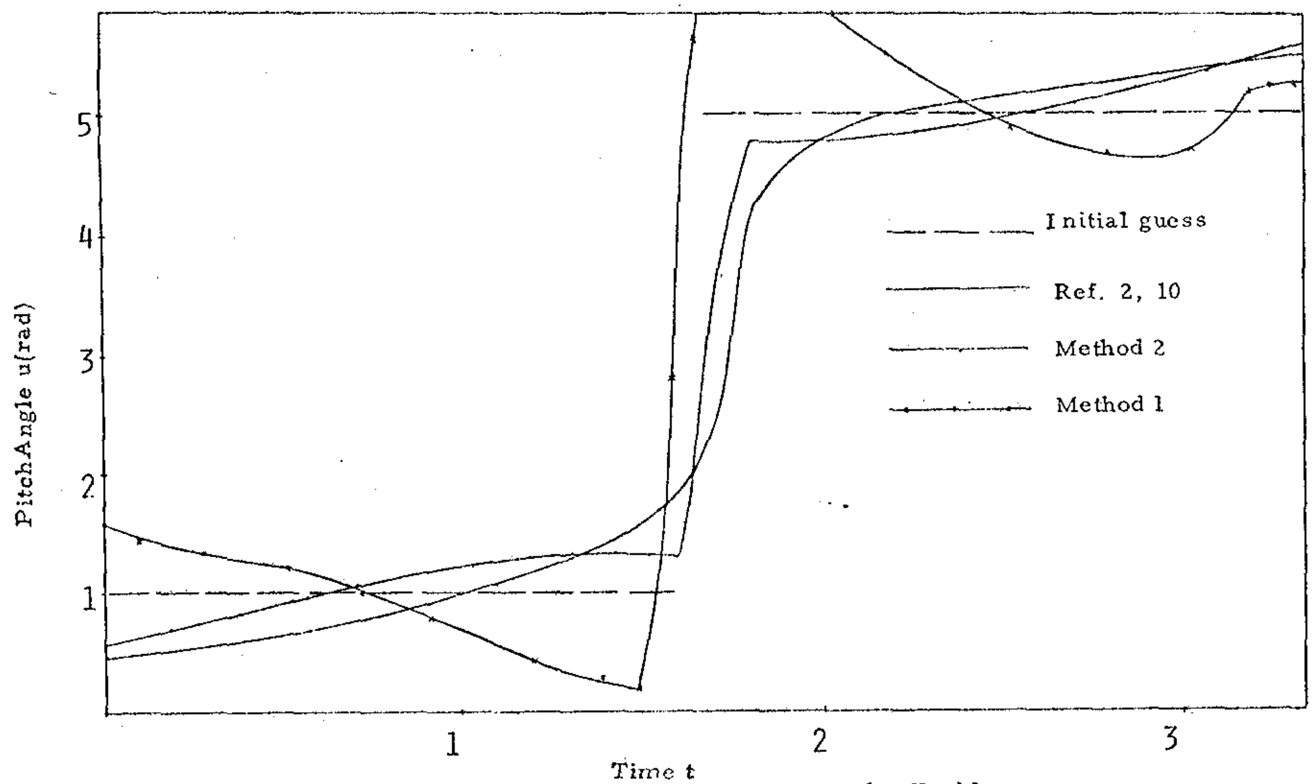
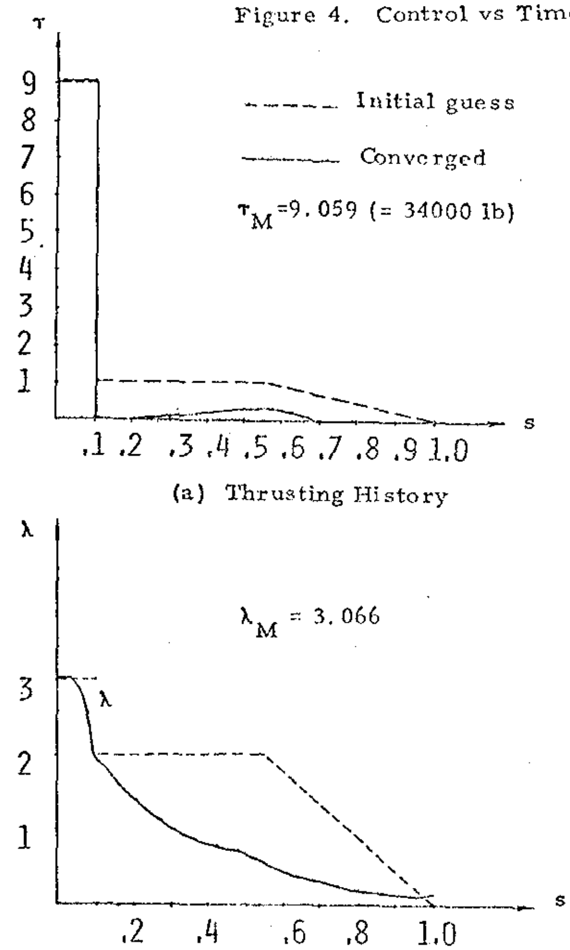


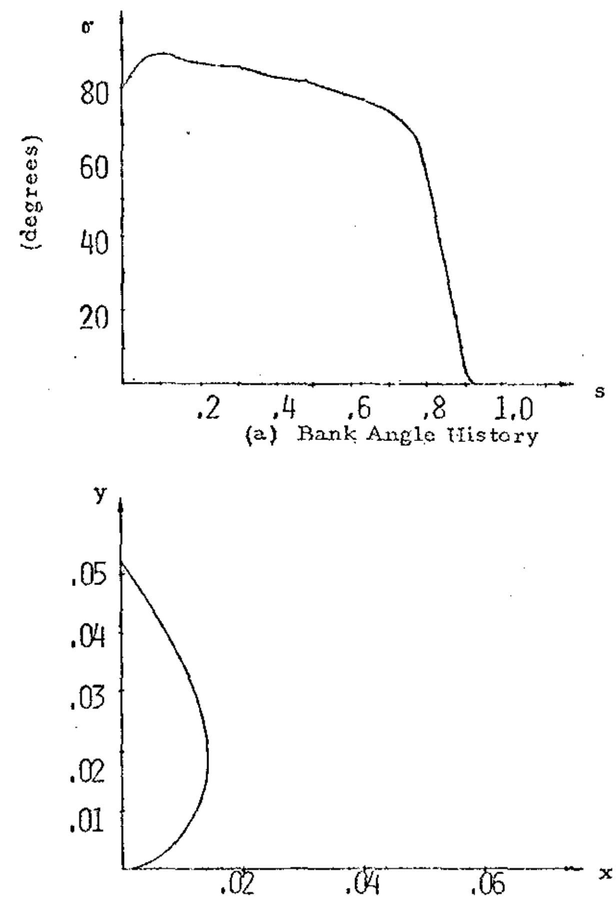
Figure 4. Control vs Time for Orbital Transfer Problem



(a) Thrusting History

(b) Lift Control History

Figure 5. Optimal Controls for Atmospheric Flight Problem.



(a) Bank Angle History

(b) Optimal Trajectory

Figure 6. Bank Angle and Trajectory for Atmospheric Flight Problem.