AFFDL-TR-65-20, Vol. I

A STUDY OF TIME-OPTIMAL RENDEZVOUS IN THREE DIMENSIONS (Vol. I)

TECHNICAL DOCUMENTARY REPORT No. AFFDL-TR-65-20

January 1965

AIR FORCE FLIGHT DYNAMICS LABORATORY
RESEARCH AND TECHNOLOGY DIVISION
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO

Project No. 8219, Task No. 821904

(Prepared under Contract No. AF 33(657)-11319 by Aeronautical Research Associates of Princeton, Inc., Princeton, New Jersey; B. H. Paiewonsky and P. J. Woodrow, authors)

NOTICES

FOREWORD

This report, consisting of two volumes, was prepared by Aeronautical Research Associates of Princeton, Inc. (ARAP) under USAF Contract Number AF33(657)-11319. The contract was initiated under Project No. 8219, "Stability and Control Investigation," Task No. 821904, "Systems Analysis and Optimization." The work was administered under the direction of Flight Dynamics Laboratory, FDCC, RTD, Wright-Patterson Air Force Base, Ohio, with Mr. Lawrence Schwartz, project engineer until June 1963, and Mr. Ronald Anderson, project engineer from June 1963.

This report covers work conducted from 15 April 1963 to 15 April 1964.

The research program at ARAP was under the direction of Dr. Bernard Paiewonsky. ARAP personnel participating in this study were Mr. Peter J. Woodrow and Mr. Tony Lauck. Thanks are due to Mr. Alonzo Connors of the Flight Dynamics Laboratory for help with computation arrangements at Wright Field.

The assistance of Mrs. Alita Boyden, Miss Helen Contor, and Mrs. Phyllis Bennet in preparing the manuscript is gratefully acknowledged.

ABSTRACT

The results of a study of time optimal rendezvous in three dimensions with bounds on the rocket thrust and the available propellant are described. The equations of motion are linearized and Neustadt's method is used to solve the two-point boundary value problem in the seven-dimensional state space. Three convergence acceleration schemes are studied. Fletcher and Powell's modification of Davidon's method was superior to Powell's method and a modified method of steepest ascent. Examples of terminal rendezvous paths are presented and discussed in terms of the magnitudes of the bounds on thrust and fuel. The dependence of terminal errors on initial measurement errors in position and velocity is also discussed. The range of initial values include position errors up to 25 miles and relative velocity errors of 200 ft/sec. The thrust accelerations of the rockets are on the order of 1 ft/sec^2; the propellant bounds (ideal characteristic velocities) range between 600 ft/sec and 250 ft/sec.

This report has been reviewed and is approved.

W.A. SLOAN, Jr.
Colonel, USAF
Chief, Flight Control Division
AF Flight Dynamics Laboratory

TABLE OF CONTENTS

SECTION PAGE

1. Introduction 1
2. Formulation of the Problem 5
3. The Computation of Optimal Controls 15
4. Rendezvous Studies 24
5. Summary and Conclusions 32
6. References 35

Appendix I. Accelerated Convergence Techniques for Finding Maxima and Minima of Functions of Several Variables 37

Appendix II. Computer Program for Finding the Extremum of a Multivariate Function 43

Figures 59

ILLUSTRATIONS

FIGURE PAGE

1. Coordinate System and Steering Angles.................. 6
2. Geometrical Aspects of Neustadt's Method.............. 19
3. Gradient Length and Stopping Time vs. Total Number of Steps for Powell's Method.......................... 59
4. Terminal Errors Position and Velocity vs. Iteration Number...................................... 60
5. Gradient Length and Stopping Time vs. Total Number of Steps for Modified Method of Steepest Ascent.......... 61
6. Terminal Errors vs. Iteration Number.................. 62
7. Gradient Length and Stopping Time vs. Total Number of Steps for Fletcher-Powell-Davidon Method.............. 63
8. Terminal Errors vs. Iteration Number.................. 64
9. Gradient Length and Stopping Time vs. Total Number of Steps for Unaccelerated Steepest Ascent using Optimal Steps.. 65
10. Rendezvous Trajectories and Control Variable Time Histories... 66
11. Rendezvous Trajectories and Control Variable Time Histories... 67
12. Projection of Optimal Thrust Direction on $x_1 - x_3$ Plane.. 68
13. Motion in $x_1 - x_3$ Plane for $A_{max} = 1.0$ ft./sec^2 and Varying Amounts of Fuel........................... 69
14. Plot of x_5 vs. x_6 for $A_{max} = 1.0$ ft./sec^2 and Varying Amounts of Fuel............................. 70
15. Fuel Time Histories, $A_{max} = 1.0$ ft./sec^2.............. 71

ILLUSTRATIONS (Cont'd.)

FIGURE		PAGE
16.	Steering Angle θ vs. Time	72
17.	Steering Angle ϕ vs. Time	73
18.	Optimal Time Tradeoffs	74
19.	Flow Diagram for a Recursive Procedure	75
20.	Example of Parallel Chords and Quadratic Functions	76
21.	Powell's Method in Three Dimensions	77

TABLES

TABLE		PAGE
1.	Matrix of Partial Derivatives	26
2.	Measurement Error Sensitivity Coefficients-- A_{max} = 1.0 ft./sec^2, ΔV = 600 ft./sec	28
3.	Measurement Error Sensitivity Coefficients-- A_{max} = 1.0 ft./sec^2, ΔV = 450 ft./sec	28
4.	Measurement Error Sensitivity Coefficients-- A_{max} = 1.0 ft./sec^2, ΔV = 250 ft./sec	29
5.	Measurement Error Sensitivity Coefficients-- A_{max} = .5 ft./sec^2, ΔV = 600 ft./sec	29
6.	Measurement Error Sensitivity Coefficients-- A_{max} = .5 ft./sec^2, ΔV = 450 ft./sec	30

SECTION 1

INTRODUCTION

The terminal phase of orbital rendezvous has been studied by many investigators. These studies range in scope from analyses of coordinate systems and simplified dynamical equations to large scale computer simulations of proposed guidance schemes. Our work has been directed towards the solution of the linearized three-dimensional time optimal rendezvous problem with bounded thrust and limited fuel. The main objective of this study was the generation of optimal paths and control laws. The computational scheme was also of interest as it represents one of the first applications of Neustadt's synthesis method (1,2) to a high order system. The time optimal problem with bounded fuel is similar in many respects to the minimum fuel problem with time fixed; the system equations are the same and the Euler equations are closely related. Neustadt's method is applicable to either problem.

We will discuss the previous research very briefly; extensive reviews have already appeared in orbital flight handbooks.

The type of guidance system used for terminal phase of rendezvous depends to a very large extent on the propulsion system of the maneuvering vehicle. The early rendezvous studies by Sears and Felleman (3), Clohessy and Wiltshire (4), Wheelon (5) and Hord (6) provided an analytical basis for many subsequent investigations. Studies of impulsive guidance schemes based on orbital mechanics have been made by Eggleston (7), Stapleford (8) and Hornby (9) among others. The results of Reference (8) show that those guidance schemes which assume instantaneous velocity changes will have significant

---

Manuscript released by the authors July 1964 for publication as an RTD Technical Documentary Report.

errors unless the thrust acceleration is relatively high; the burning times would then be sufficiently short to justify the assumption of impulsive velocity changes. Rendezvous using continuously burning or throttleable rockets has been studied by Cicolani (10), Carney (11), and Passera (12).

Fuel optimal and time optimal rendezvous maneuvers have been studied by Goldstein et al. (13), Kelley and Dunn (14), Hinz (15), and McIntyre and Crocco (16), but no synthesis procedures were developed and the thrust and fuel constraints were applied separately. Simulations of pilot-controlled rendezvous have been carried out by Brissenden et al. (17) and Beasley (18) among others.

The sections to follow contain the formulation of the optimization problem, a description of Neustadt's method, and a discussion of the results of the computational study which includes a comparison of three methods for minimizing functions of several variables.

The rendezvous maneuver is based on controlling the relative motion between two space vehicles. Let the non-maneuvering vehicle be used as the origin of a moving coordinate system. The gravitational terms in the equations of relative motion can be linearized using the assumption that the relative distance between the two vehicles is small compared to their distances to the Earth's center. The result is a set of linear differential equations for the relative position and velocity; i.e., $\dot{x}_i = \sum_j A_{ij} x_j + \sum_k B_{ik} u_k$ where x is a six-dimensional vector describing the state of the system (relative position and velocity) and u is a vector representing the rocket thrust acceleration. The matrices A and B describe the coupling between the different degrees of freedom of the system and the control. The solution of a rendezvous problem consists in finding an allowable control, $u(t)$, which brings the maneuvering vehicle into coincidence with the target with zero relative velocity. The problem as stated is a terminal control problem and is not an optimization problem. Specialization is achieved by putting constraints on the kinds of control actions which are permissible; e.g., impulsive thrusts, continuous thrusts. Further specialization of the terminal control problem is obtained by stating the kind of guidance law to be employed, e.g., proportional (homing) navigation (19) or exponentially weighted proportional guidance (20).

The problem of terminal control is changed to one of optimal control by asking for a path that satisfies not only the terminal conditions and the constraints on the control action but furthermore, gives an optimal value to a functional (usually time or fuel) taken along the path.

From a design standpoint, the adjustable parameters are rocket thrust levels and fuel allotment: the duration of the flight to rendezvous or the maneuvering time is frequently a secondary consideration, as long as it does not become excessively large. From an operational standpoint, however, the maneuvering vehicle has to reach the target, and the thrust levels and fuel on board are fixed and act as constraints on the possible solutions. In an actual rendezvous, it is necessary to find a control law, compatible with the real thrust and fuel limitations, which brings the maneuvering vehicle to the target. The time optimal controller can obtain a nominal path consistent with the propulsive capabilities of the system.

The time optimal rendezvous is formulated as a two-point boundary value problem in a straightforward way using the maximum principle.

The solution of this two-point boundary value problem involves searching for the optimal initial conditions for the adjoint equation. Neustadt's iterative method for the time optimal control problem finds the optimal initial conditions for the adjoint but this method requires a program for maximizing a function of several variables. This is a difficult maximization because the maximum is quite flat and the location of the maximum must be found precisely. Earlier studies (21) showed that Powell's convergence acceleration method worked well. In the present study, we compare Powell's method with a modified method of steepest ascent and also with a method described by Fletcher and Powell (22). The latter procedure was based on work done originally by Davidon (23).

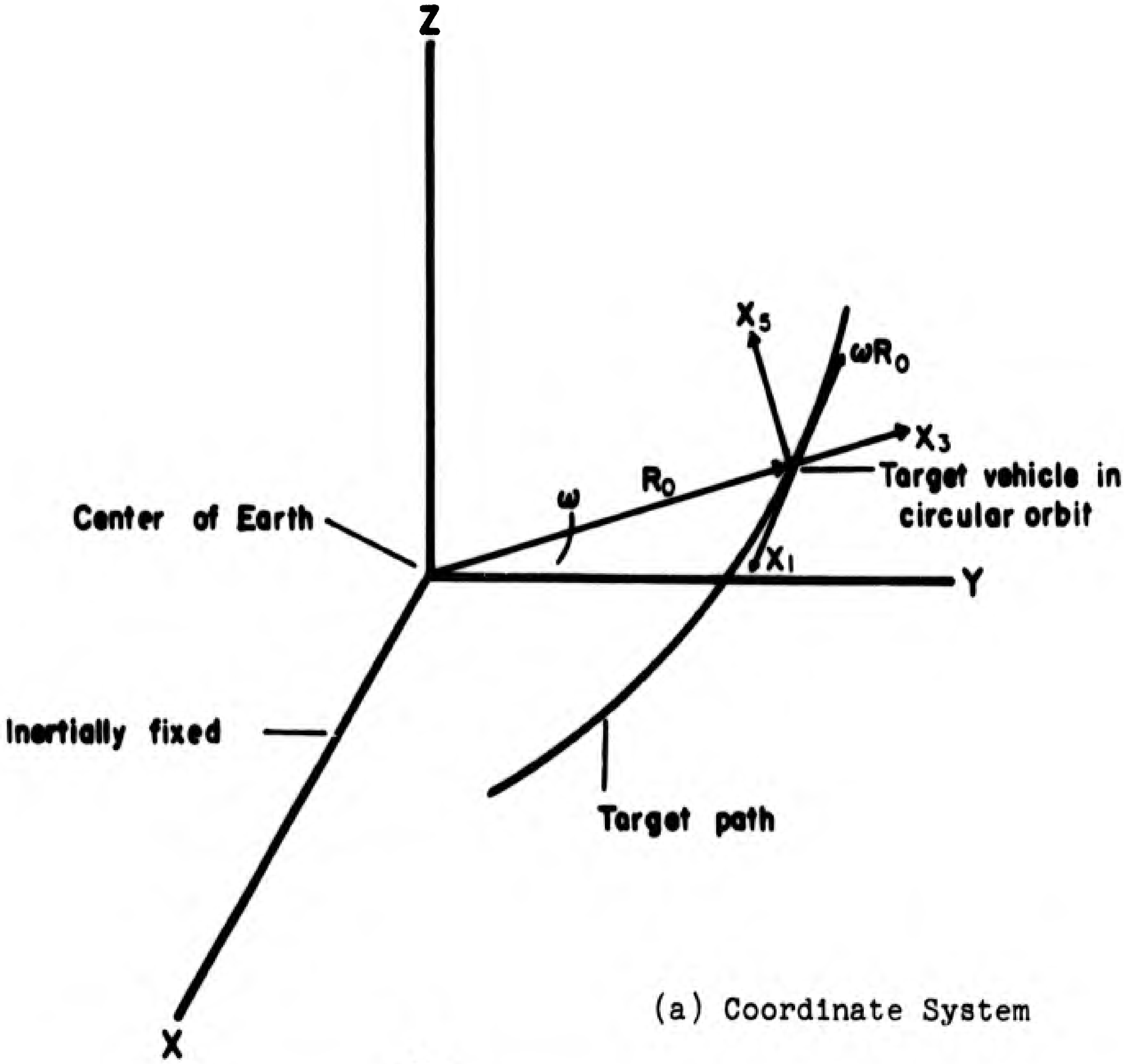
The optimal rendezvous paths for two different initial conditions and for several values of thrust level and fuel allocation are presented along with a discussion of the sensitivity of terminal errors to initial errors in position and velocity estimation.

SECTION 2

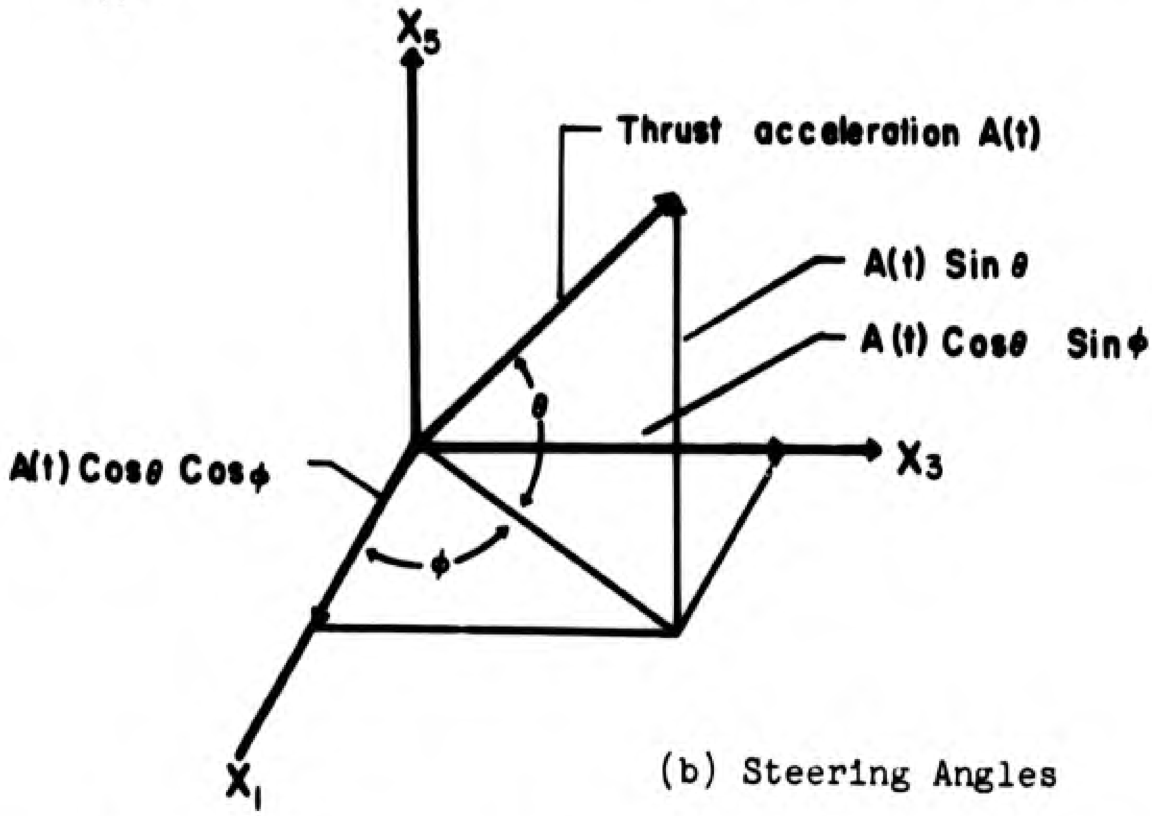
FORMULATION OF THE PROBLEM

The problem under consideration is the terminal phase of a space rendezvous maneuver. The object of the study is to examine minimum-time paths subject to restrictions on the maximum thrust levels and bounds on the propellant available for maneuvering. The thrust acceleration levels are moderately low; on the order of 1 ft/sec^2. The highest ΔV available in any example was 600 ft/sec; the least ΔV was 250 ft/sec. The total propellant available in all cases was less than 5% of the total vehicle mass. The dynamic equations do not include the effect of the time varying total mass because the inclusion of this effect makes the problem much more difficult to solve.

There is a duality between the minimum-time and minimum-fuel problems. The time-optimal and fuel-optimal rendezvous problems have both been studied before (13,14) in linearized form. The complete solution of these two-point boundary problems was not obtained by the previous investigations. We solve the two-point boundary problems by the application of Neustadt's method for the time-optimal case with constraints on the thrust and on the fuel. The computational techniques developed for the time-optimal case are applicable, with minor modifications, to the minimum-fuel problem. The three dimensional powered flight equations are linearized by assuming that the relative distance between the target and the maneuvering vehicle is small compared with the distance of target to the Earth's center. A uniformly rotating coordinate system is employed as shown in Figure 1(a). The rotating rectangular system with axes labeled x_1, x_3, and x_5 has



(a) Coordinate System



(b) Steering Angles

Figure 1. Coordinate System and Steering Angles

its origin at the nominal target radius and moves with the target's mean motion. The x_1 axis is in the orbital plane in the tangential direction, opposite to the direction of the rotation, the x_3 axis is in the outward radial direction and the x_5 axis is orthogonal to both the x_1 and x_3 axis. Figure 1(a) shows the x_1 x_3 x_5 trihedron located at the radius R_o. The trihedron XYZ is an earth centered non-rotating rectangular coordinate system used as a reference frame for the initial orbit.

A target in a circular orbit at the nominal radius will be stationary if placed at the origin of the rotating system. Target vehicles in orbits with eccentricity correspond to a rendezvous with an object moving with respect to the origin of the x_1 x_3 x_5 system; i.e., there will be relative motion between the target and the origin of the coordinate system. This study was limited to circular target orbits although the method employed in solving the optimal control problem can handle moving targets.

The derivation of the linearized equations is easily available in the literature* and will not be repeated here. It is important to note that the transformations leading to the linearized equations of relative motion are not unique (14). The equations resulting from the application of different transformations may be identical in form; however, the interpretation of the dependent variables and steering angles is different. We use the rectangular form in this study; it is clear that the synthesis method works in either case.

---

*See, for example, References 13 or 14.

The linearized equations of motion are:

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= 2\omega x_4 + u_1 \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= -2\omega x_2 + 3\omega^2 x_3 + u_2 \\
\dot{x}_5 &= x_6 \\
\dot{x}_6 &= -\omega^2 x_5 + u_3
\end{aligned} \tag{1}$$

where the dot indicates d/dt and $u_1 = A(t) \cos\theta \cos\phi$, $u_2 = A(t) \cos\theta \sin\phi$, and $u_3 = A(t) \sin\theta$. Figure 1(b) illustrates the definition of the steering angles, θ and ϕ. The thrust acceleration constraint is given by the equation $u_1^2 + u_2^2 + u_3^2 = A^2(t)$; $0 \leq A(t) \leq A_{max}$ and the propellant constraint is given by the requirement that $\frac{m_o}{c}\int_o^T A(t)dt \leq m_p(0)$ where $m_p(0)$ is the initial propellant mass, m_o is the total vehicle mass which is assumed to be constant, and c is the rocket effective exhaust velocity.

It will be convenient to make the following transformations of variables. The time is rescaled in terms of the angular velocity. Let $t' = \omega t$, and define

$$\begin{aligned}
y_1 &= \omega^2 x_1, & y_2 &= \omega x_2, & y_3 &= \omega^2 x_3 \\
y_4 &= \omega x_4, & y_5 &= \omega^2 x_5, & y_6 &= \omega x_6
\end{aligned} \tag{2}$$

The equations are rewritten below in terms of the new variables. The prime denotes d/dt'.

$$\begin{aligned}
\frac{dy_1}{dt'} &= y_1' = y_2 & \frac{dy_4}{dt'} &= y_4' = -2y_2 + 3y_3 + u_2 \\
\frac{dy_2}{dt'} &= y_2' = 2y_4 + u_1 & \frac{dy_5}{dt'} &= y_5' = y_6 \\
\frac{dy_3}{dt'} &= y_3' = y_4 & \frac{dy_6}{dt'} &= y_6' = -y_5 + u_3
\end{aligned} \tag{3}$$

The equations of motion can be put into a more compact form by using the matrix notation $y' = Ay + Bu$ where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -2 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The solution of these equations is given below in the well-known integral form:

$$y(t') = Y(t')\left[y(0) + \int_0^{t'} Y^{-1}(\tau)B(\tau)u(\tau)d\tau\right], \qquad \frac{dY}{dt'} = AY, \; Y(0) = I \tag{4}$$

where

$$Y(t') = \begin{bmatrix} 1 & 4\sin t' - 3t' & 6(t' - \sin t') & 2(1-\cos t') & 0 & 0 \\ 0 & 4\cos t' - 3 & 6(1-\cos t') & 2\sin t' & 0 & 0 \\ 0 & 2(\cos t' - 1) & -3\cos t' + 4 & \sin t' & 0 & 0 \\ 0 & -2\sin t' & 3\sin t' & \cos t' & 0 & 0 \\ 0 & 0 & 0 & 0 & \cos t' & \sin t' \\ 0 & 0 & 0 & 0 & -\sin t' & \cos t' \end{bmatrix} \tag{5}$$

$$Y^{-1}(t') = \begin{bmatrix} 1 & -4\sin t' + 3t' & 6(\sin t' - t') & 2(1-\cos t') & 0 & 0 \\ 0 & 4\cos t' - 3 & 6(1-\cos t') & -2\sin t' & 0 & 0 \\ 0 & 2(\cos t' - 1) & -3\cos t' + 4 & -\sin t' & 0 & 0 \\ 0 & 2\sin t' & -3\sin t' & \cos t' & 0 & 0 \\ 0 & 0 & 0 & 0 & \cos t' & -\sin t' \\ 0 & 0 & 0 & 0 & \sin t' & \cos t' \end{bmatrix} \tag{6}$$

The function $z(t', \eta)$ to be used in Neustadt's method (1) is now defined:

$$z(t',\eta) = -\int_0^{t'} Y^{-1}(\tau)B(\tau)u(\tau,\eta)d\tau \tag{7}$$

where $u(\tau,\eta)$ is the control obtained by maximizing the hamiltonian. The individual components of $z(t',\eta)$ are:

$$\begin{aligned}
z_1(t',\eta) &= -\int_0^{t'} \Big[(3\tau - 4\sin\tau)u_1 + 2(1-\cos\tau)u_2\Big]d\tau \\
z_2(t',\eta) &= -\int_0^{t'} \Big[4(\cos\tau - 3)u_1 - (2\sin\tau)u_2\Big]d\tau \\
z_3(t',\eta) &= -\int_0^{t'} \Big[2(\cos\tau - 1)u_1 - (\sin\tau)u_2\Big]d\tau \\
z_4(t',\eta) &= -\int_0^{t'} \Big[(2\sin\tau)u_1 + (\cos\tau)u_2\Big]d\tau \\
z_5(t',\eta) &= -\int_0^{t'} \Big[(-\sin\tau)u_3\Big]d\tau \\
z_6(t',\eta) &= -\int_0^{t'} \Big[(\cos\tau)u_3\Big]d\tau
\end{aligned} \tag{8}$$

The significance of this function will be made clear later on. It is introduced at this point simply to show its dependence on $Y^{-1}(t')$.

Now, introduce adjoint variables ψ_i, $i = 1 \ldots 6$, satisfying the differential equations $\psi' = -A^T\psi$. The solution of these equations is:

$$\psi = -\Big[Y^{-1}(t')\Big]^T \eta$$

where $\psi_i(0) = -\eta_i$.

Let

$$\alpha = 2\eta_1 - \eta_4, \qquad \beta = 2\eta_2 + \eta_3, \qquad \gamma = 3\eta_2 + 2\eta_3$$

$$a = 2\alpha, \qquad b = 2\beta, \qquad c = 3\eta_1$$

These expressions are substituted into the expanded equations to give the following relations between the components of ψ and η:

$$\begin{aligned}
\psi_1(t') &= -\eta_1 \\
\psi_2(t') &= a \sin t' - b \cos t' - ct' + \gamma \\
\psi_3(t') &= -3a \sin t' + 3\beta \cos t' + 2ct' - 2\gamma \\
\psi_4(t') &= \beta \sin t' + a \cos t' - 2\eta_1 \\
\psi_5(t') &= -\eta_6 \sin t' - \eta_5 \cos t' \\
\psi_6(t') &= \eta_5 \sin t' - \eta_6 \cos t'
\end{aligned} \tag{9}$$

A new variable, y_7, is introduced now to account for the fuel constraint. The variable y_7 satisfies the differential equation

$$\frac{dy_7}{dt'} = -\omega\frac{c}{m_o}\frac{dm_p}{dt'} = -A(t'), \qquad y_7(0) = \omega x_7(0) = \omega\Delta V \tag{10}$$

where $x_7(0) = -c\, \ell n(\frac{m_o - m_p}{m_o}) = \Delta V$, $\frac{dm_p}{dt'}$ is the propellant mass flow, c is the rocket effective exhaust velocity, and m_o is the total vehicle mass. The adjoint variable ψ_7 corresponding to y_7 satisfies the equation $(d\psi_7/dt') = 0$. The initial condition $\psi_7(0)$ is called $-\eta_7$ and is an additional unspecified parameter. We also define an additional component to $z(t',\eta)$, namely,

$$z_7(t',\eta) = -\int_0^{t'} A(\tau,\eta)d\tau \tag{11}$$

which is related to the other components of $z(t',\eta)$ in an obvious way. The thrust acceleration dependence on η will be obtained from the maximum principle.

To find the optimal control we form the hamiltonian $H = \sum_{i=1}^{n} \psi_i y_i'$ from the adjoint variables and the state velocity vector and then maximize H with respect to u subject to all constraints on u. The thrust acceleration is bounded so $\sqrt{u_1^2 + u_2^2 + u_3^2} \leq A_{max}$. The constraint on the available propellant may also be written as an inequality, i.e.

$$\int_0^T \sqrt{u_1^2 + u_2^2 + u_3^2}\, dt \leq c \,\ln \frac{1}{1 - \frac{m_p}{m_o}} = \Delta V = x_7(0)$$

where m_p is the total available propellant mass and m_o is the total mass at the initial time. The hamiltonian to be maximized can be separated into two parts; one part depends on the control and the other part is independent of the control. The hamiltonian for the time optimal problem is (24):

$$H = \sum_{\substack{i=1\\j=1}}^{6} \psi_i A_{ij} y_j + u_1\psi_2 + u_2\psi_4 + u_3\psi_6 - \psi_7 A(t') + 1$$

$$H = \sum_{\substack{i=1\\j=1}}^{6} \psi_i A_{ij} y_i \tag{12}$$

$$+ A(t')\left[\psi_2 \cos\theta \cos\phi + \psi_4 \cos\theta \sin\phi + \psi_6 \sin\theta - \psi_7\right] + 1$$

To find the optimal steering angles, we maximize the hamiltonian with respect to θ and ϕ:

$$\frac{\partial H}{\partial \phi} = \left[-\psi_2 \cos\theta \sin\phi + \psi_4 \cos\theta \cos\phi \right] A(t') = 0$$

$$\Rightarrow \tan\phi = \psi_4/\psi_2$$

$$\Rightarrow \sin\phi = \psi_4 \Big/ \sqrt{\psi_2^2 + \psi_4^2} \qquad (13)$$

$$\Rightarrow \cos\phi = \psi_2 \Big/ \sqrt{\psi_2^2 + \psi_4^2}$$

$$\frac{\partial H}{\partial \theta} = \left[-\psi_2 \sin\theta \cos\phi - \psi_4 \sin\theta \sin\phi + \psi_6 \cos\theta \right] A(t') = 0$$

$$= \left[-\sqrt{\psi_2^2 + \psi_4^2} \sin\theta + \psi_6 \cos\theta \right] = 0$$

$$\Rightarrow \tan\theta = \psi_6 \Big/ \sqrt{\psi_2^2 + \psi_4^2} \qquad (14)$$

$$\Rightarrow \sin\theta = \psi_6 \Big/ \sqrt{\psi_2^2 + \psi_4^2 + \psi_6^2}$$

$$\Rightarrow \cos\theta = \sqrt{\psi_2^2 + \psi_4^2} \Big/ \sqrt{\psi_2^2 + \psi_4^2 + \psi_6^2}$$

The optimal burning program $A^o(t')$ is found by maximizing the hamiltonian with respect to $A(t')$:

$$H = \sum_{\substack{i=1 \\ j=1}}^{6} \psi_i A_{ij} y_j + A(t') \left[\sqrt{\psi_2^2 + \psi_4^2 + \psi_6^2} - \psi_7 \right]$$

$$A^o(t') = \begin{cases} A_{max} : \text{when } \sqrt{\psi_2^2 + \psi_4^2 + \psi_6^2} - \psi_7 \geq 0 \\ 0 \quad : \text{when } \sqrt{\psi_2^2 + \psi_4^2 + \psi_6^2} - \psi_7 < 0 \end{cases} \qquad (15)$$

The optimal control components u_1^o, u_2^o, u_3^o, are given below in terms of the adjoint variables

$$\begin{aligned} u_1^o &= A^o(t')\psi_2/r \\ u_2^o &= A^o(t')\psi_4/r \\ u_3^o &= A^o(t')\psi_6/r \end{aligned} \tag{16}$$

$$\text{where} \quad r = \sqrt{\psi_2^2 + \psi_4^2 + \psi_6^2}$$

Note that ψ_7 is always ≥ 0 because $\frac{d\psi_7}{dt'} = 0 \Rightarrow \psi_7 =$ constant $= -\eta_7$ and

$$\eta_7 = \frac{\partial T^o}{\partial m_p} \leq 0 \qquad \text{(see Ref. 21)}$$

Define a function f as follows:

$$f = r - \psi_7$$

The function f is used in the program as part of the procedure for obtaining the times when the control goes on and off the boundary; i.e., find all times such that

$$f = r - \psi_7 = 0$$

The condition above yields transcendental equations. The zeros of derivatives of these functions cannot be determined analytically and a special numerical test was developed to find them.

SECTION 3

THE COMPUTATION OF OPTIMAL CONTROLS

A. Neustadt's Method. Algorithms for synthesizing minimum time and minimum effort controllers for linear systems have been developed by Neustadt (1 , 2). These algorithms provide convergent iterative procedures for solving the two-point boundary problems associated with the optimization problems. The computational difficulties associated with two-point boundary problems are well known. The purpose of this section is to discuss the computational aspects of Neustadt's method applied to the rendezvous problem.

A complete account of the mathematical details of the method is available in the literature. We will present a summary of the development with the stress placed on geometrical interpretations. The general idea will be illustrated using a time optimal controller as an example. Suppose we are given the linear system:

$$\dot{x} = A(t)x + B(t)u \tag{17}$$

where x is an n-vector

u is an r-vector

A and B are $n \times n$ and $n \times r$ matrices respectively.

The control u is to be a piecewise continuous function of time and is also restricted to a compact convex set of allowed functions, U. The initial conditions $x_1(0)$ are prescribed and the desired terminal state is also specified.

The objective of the control is to transfer the system state to the origin in the least time.

The conditions under which this problem has a solution have been discussed at length elsewhere (24,25). We assume that these conditions are satisfied. The hamiltonian for a general performance index, L(min $\int$ L dt), is

$$H = \sum_{i=1}^{n} \psi_i \left[\sum_j A_{ij} x_j + \sum_k B_{ik} u_k \right] + \psi_o L \tag{18}$$

In the time optimal problem, L = 1 and $\psi_o \equiv 1$. The form of the control is available directly from the maximum principle, i.e.,

$$\max_{u \in U} H = \max_{u \in U} \sum_{i=1}^{n} \sum_{k=1}^{r} \psi_i B_{ik} u_k \tag{19}$$

The control is given in terms of the adjoint variables $\psi_i(t)$. The adjoint variables satisfy the differential equations

$$\psi_i(t) = - \frac{\partial H}{\partial x_i} , \; \psi_i(0) = -\eta_i \tag{20}$$

Controls satisfying the maximum principle depend on η and are denoted as $u(\tau,\eta)$. The problem is to find the initial conditions on the adjoint so that the system goes from the initial state to the final state when the control $u^o(t,\eta)$, (derived from the maximum principle), is applied.

The solution of the system (17) can be written as

$$x(t) = X(t)\left[x(0) + \int_0^t X^{-1}(\tau)B(\tau) \; u(\tau)d\tau\right] \tag{21}$$

where $X(t)$ is the fundamental matrix solution of (17). Let the target, $x(T)$, be the origin. The set $C(t)(t \geq 0)$, defined as the set of all points swept out by the vectors

$$x(t) = - \int_0^t X^{-1}(\tau)B(\tau)u(\tau)d\tau$$

for a given positive t and for all admissible u, is called the reachable set with respect to the origin. This set comprises all points from which the origin can be reached within time t. The boundary of $C(t)$ for each t is an optimal isochrone. The conditions that U be compact and convex insure that $C(t)$ will be closed, bounded and convex. Furthermore, if $t < t'$ then $C(t)$ is contained in $C(t')$. It is shown in Refs. (1) and (2) that the normal to the support plane or tangent plane at a point of the boundary of $C(t)$ is the vector η, i.e. the initial condition for the adjoint. The function

$$z(t,\eta) = -\int_0^t X^{-1}(\tau)B(\tau)\,u(\tau,\eta)d\tau$$

for any $\eta \neq 0$ and $t > 0$ is a mapping of vectors η into vectors $z(t,\eta)$. This function generates the boundary of $C(t)$ for a given $t > 0$.

The method of solution will first be described, then explained further, then justified. The iterative search for the optimal η begins by guessing the slope of the support plane at $x(0)$, i.e., guessing a starting value for η (call it η^1). In all of our computational studies, we use a unit vector parallel to $x(0)$ as a starting value for η. The function $z(t,\eta^1)$ is generated as a function of t and the integration is stopped at the time when

$$f(t,\eta^1) \equiv \eta^1 \cdot \left[z(t,\eta^1) - x(0)\right] = 0$$

This time is called $F(\eta^1)$, i.e.

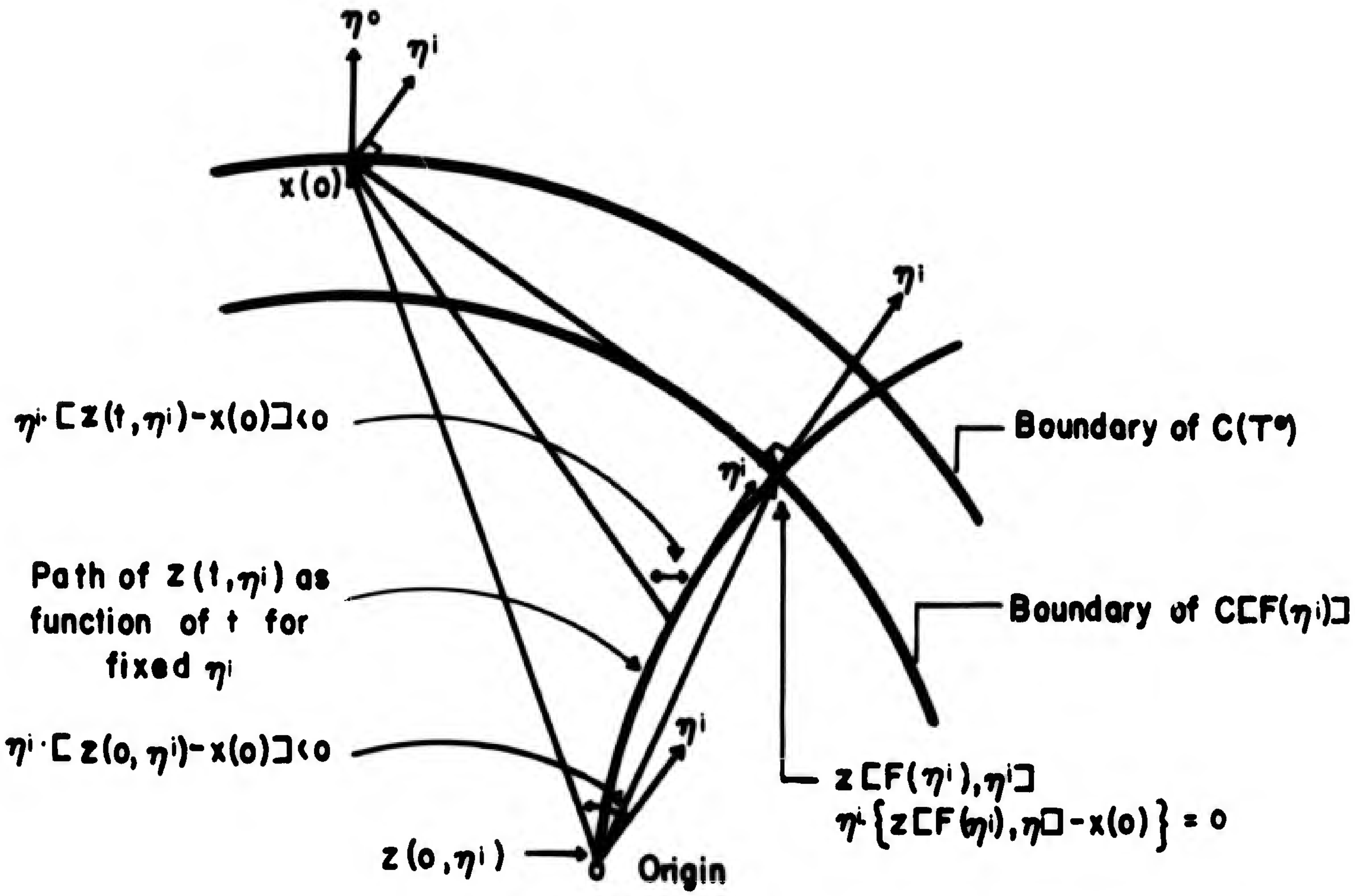
$$\eta^1 \cdot \left\{z\left[F(\eta^1),\eta^1\right] - x(0)\right\} = 0$$

It is shown in Reference 1 that $f(t,\eta)$ is a monotonic increasing function of time. It starts with a negative value $\eta \cdot (-x(0))$ and must vanish because $f(t,\eta)$ is continuous and monotonic increasing in t. The vector $z[F(\eta^i),\eta^i] - x(0)$ is recorded and provides the direction for the corrections to η^i. The rule is to take

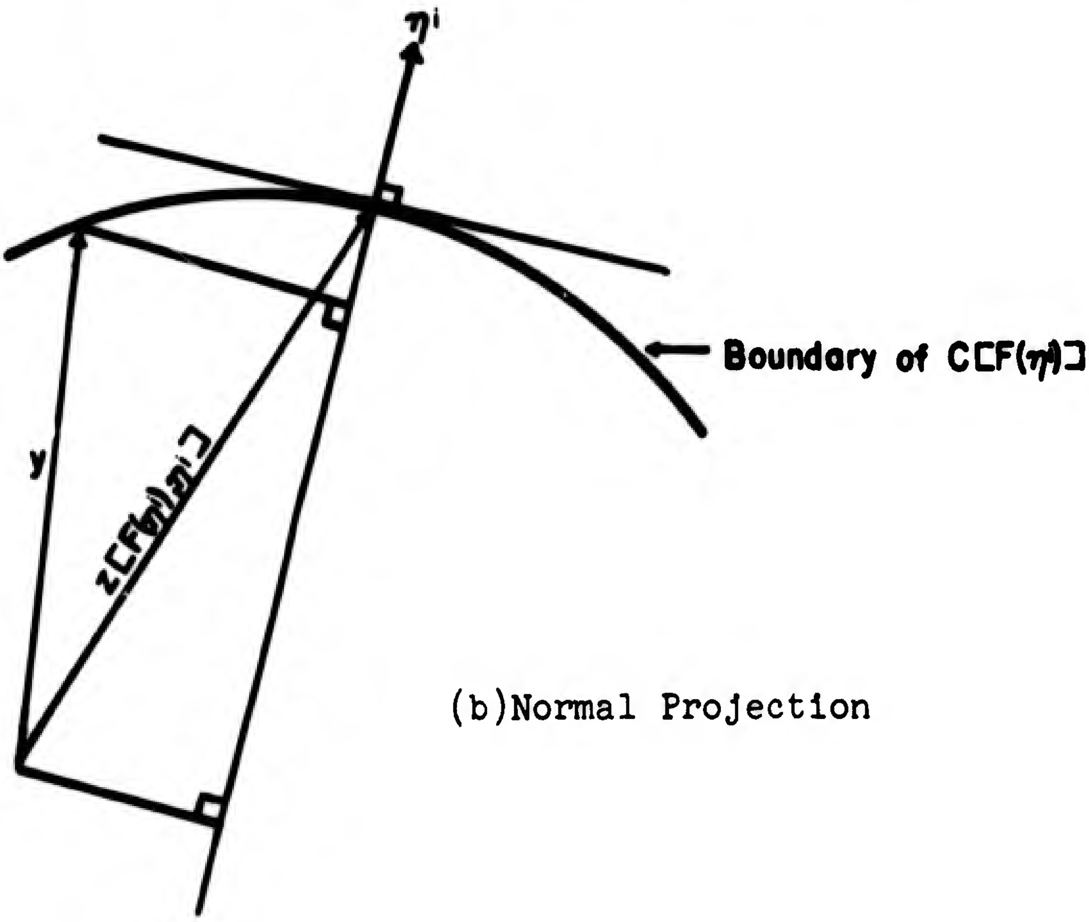
$$\eta^{i+1} = \eta^i + \frac{k\left\{z[F(\eta^i),\eta^i] - x(0)\right\}}{\|z[F(\eta^i), \eta^i] - x(0)\|}$$

where $\| \ \|$ denotes the Euclidean norm and k is a parameter used to adjust the step size. The goal of these iterations is to find the η^i that maximizes $F(\eta^i)$. This vector, η^o, causes the boundary conditions to be satisfied when $u^o(t,\eta^o)$ is applied to the system (17). The iterations are based on the idea of steepest ascent, and the gradient of $F(\eta^i)$ with respect to η can be shown to be the vector $\{z[F(\eta^i),\eta^i] - x(0)\}$.

The key to understanding the significance of the various steps involved in this method can be found in the theory of convex sets. There are three steps in the preceding development that are essential. Let the trial value of η be η^i. Construct the path $z(t,\eta^i)$ as a function of t and find the point where this path crosses the trial support plane through $x(0)$ with normal η^i as shown in Figure 2(a). The scalar product $\eta^i \cdot [z(t,\eta^i) - x(0)]$ is a monotonic increasing function of time. It starts from a negative value $\eta^i \cdot (-x(0))$ and vanishes when $z(t,\eta^i)$ crosses the hyperplane through $x(0)$ with normal η^i. The location of this point on the hyperplane is also shown in Figure 2(a). This η^i is the optimal η for the point $z[F(\eta^i),\eta^i]$, so η^i is the normal to the boundary $C[F(\eta^i)]$ at the point $z[F(\eta^i),\eta^i]$. The time $F(\eta^i)$ is less than the optimal



(a) Path Visualization



(b)Normal Projection

Figure 2. Geometrical Aspects of Neustadt's Method

time T^o because $C[F(\eta^1)]$ is contained in $C(T^o)$.

The justification for these assertions is based on the convexity of $C(t)$. Observe that $x(0)$ lies in the boundary of the set $C(T^o)$. The convexity of $C[F(\eta^1)]$ assures us that $\eta^1 \cdot z[F(\eta^1),\eta^1] > \eta^1 \cdot y$ for all vectors y, $y \neq z[F(\eta^1),\eta^1]$ in $C[F(\eta^1)]$. This is easy to see by looking at the projection of $z[F(\eta^1),\eta^1]$ onto the normal to the hyperplane as shown in Figure 2(b). But $\eta^1 \cdot x(0) = \eta^1 \cdot z[F(\eta^1),\eta^1]$ by definition of $F(\eta^1)$. Therefore $x(0)$ is not in $C[F(\eta^1)]$ but lies outside it and $T^o > F(\eta^1)$, unless $z[F(\eta^1),\eta^1] = x(0)$; in that case $F(\eta^1) = T^o$. This means that $F(\eta)$ is maximized when $z[F(\eta),\eta] = x(0)$. Neustadt has shown, in the references cited, that $\nabla F(\eta^1)$ is the error vector $z[F(\eta^1),\eta^1] - x(0)$, and the iteration rule for steepest ascent follows directly from this. The two-point boundary value problem has been transformed into the problem of locating the maximum value of a function of several variables. It is important to note that it is the location of the maximum value of $F(\eta)$ which determines the optimal control. The maximum value itself is not used in the solution of the two-point boundary value problem.

Earlier studies (21,26) on the synthesis of optimal controls by Neustadt's method showed that the problem of finding the vector η^o, which maximized $F(\eta)$, could be difficult. In these studies of third order systems, Powell's convergence acceleration method was successful in bringing the terminal conditions into the desired values.

Some research into the computational aspects of this problem was carried out as part of the rendezvous studies. We hoped to find a computational scheme that would be easier to program than Powell's method and would be at least as good in bringing in the boundary conditions. The main problem is simply that $F(\eta)$ is a very flat function of η. The value of the optimal time is easily obtained but the end conditions are frequently in error by large amounts.

In the rendezvous studies, we tried the following methods:

1) Powell's Method
2) Modified Steepest Ascent--special logic for testing steps (see Refs. 21, 26, and Appendices II--VII)
3) Fletcher-Powell modification of Davidon's variable metric method

Brief descriptions of these methods are contained in the Appendices. For more detailed descriptions of Powell's, Fletcher-Powell's, or Davidon's method, the original papers (27,22,23) should be consulted.

A set of figures to illustrate the relative convergence rates for these three maximization methods has been prepared. The gradient of the function being maximized is available at each stage of the computation, making it possible to do the job without a numerical partial differentiation. The function itself is so flat that its use is very undesirable in any of the computation, except perhaps for some very simple logical tests. The optimal step, in all of the methods tested, is computed using a criterion based on an orthogonality property of the gradient (28).

Figure 3 shows the sequence of steps in convergence of $F(\eta)$ to T^o and $\|z[F(\eta),\eta] - x(0)\|$ to zero using Powell's method. Each point plotted corresponds to an optimum step; the abscissa, N, corresponds to total number of steps. The quantity $\|z[F(\eta),\eta] - x(0)\|$ is the length of the gradient vector in the nondimensionalized y-system. The optimal time is established by the hundredth step but $\|z[F(\eta),\eta] - x(0)\|$ is not satisfactory until the last step. The optimal time is an important quantity, but the really significant items are the errors in the boundary conditions in the physical space. The decision to stop the computation is made on the basis of the physical boundary conditions and not on the $\|z[F(\eta),\eta] - x(0)\|$ errors. Figure 4 shows the terminal error sequence for the relative range $R(T)$ and relative speed $|V(T)|$. These are the terminal errors which would result from application of optimal control $u(t,\eta^1)$ evaluated at $T = F(\eta^1)$. There is a "bouncing" character to these graphs; the major reduction in error occurs at the last few steps in a cycle. Thirty-seven optimum steps are taken, requiring 154 separate calculations of $z(t,\eta^1)$.

The modified method of steepest ascent was applied to this example with reasonable success. The convergence of the sequences of $F(\eta)$ to T^0 and $\|z[F(\eta),\eta] - x(0)\|$ to zero is shown in Figure 5. The optimal time is established quite soon but the boundary conditions shown in Figure 6 are not satisfied until 167 steps are made. The trend of the convergence of terminal conditions to the desired values is exponential at approximately 70 steps per order of magnitude reduction. There is a big reduction in error at the last step shown. The errors in the boundary conditions were acceptable at this point and the program automatically stopped. The number of steps, 167, is comparable to the 154 required by Powell's method.

The results of the Fletcher-Powell version of Davidon's variable metric method applied to this example are shown in Figures 7 and 8. This is the most rapidly convergent scheme that we have tried so far. The errors in the boundary conditions are reduced by three order of magnitude in 13 optimum steps requiring 64 evaluations of $z(t,\eta)$. Figure 9 shows an unaccelerated steepest ascent maximization using optimal steps (see Ref. 21).

SECTION 4

RENDEZVOUS STUDIES

Rendezvous paths have been generated for two sets of initial conditions. The thrust level and fuel allocations were used as parameters. The first example is typical of a class of nearly planar problems. The initial position is $x_1 = -100,000$ ft., $x_3 = 50,000$ ft., $x_5 = 25,000$ ft., and the initial velocity components are $x_2 = 200$ ft./sec., $x_4 = 50$ ft./sec., $x_6 = -20$ ft./sec. The motion out of the target orbital plane is small; the projection of the thrust vector in the $x_1 - x_3$ plane lies in the general direction of the line of sight.

Figure 10 shows the superposition of graphs of the relevant trajectory variables for several values of thrust. The bound on the fuel is fixed at $\Delta V = 350$ ft./sec. As the upper bound on the thrust acceleration is raised, there is a gradual transition from a continuous burning program to a program with a single coast period. The optimal time is increased from 10 min. to some point beyond 16 min. as the thrust acceleration is reduced from 1.0 ft./sec.^{2} to .25 ft./sec.^{2}. The steering programs change slightly.

A set of trajectories with varying thrust levels with the bound on fuel fixed by $\Delta V = 400$ ft./sec. is shown in Figure 11. There is a considerable amount of information contained in these figures. The steering angle, θ, is always small and has a switching type characteristic. An examination of the $x_5 - x_6$ phase portrait shows a motion resembling that of a two-dimensional switching system with fuel constraint. In this instance, the coupling between the $x_1 - x_3$ motions and the x_5 motion is not very strong. The fluctuations in $\cos\theta$ are small and the interaction between the steering laws, for motion in the target orbital plane and motion normal to the target orbital plane, is weak. Figure 12 shows the $x_1 - x_3$ plane and the projection of the thrust vector onto this plane. The thrust vector is nearly orthogonal to the line of sight in the beginning and swings around opposite to the velocity vector during the final braking maneuver.

Examination of the steering laws for the other thrust levels suggests the use of simplified guidance laws based on the ideas of homing systems as described in the references cited earlier. The optimal controller does have the advantage of predicting whether a rendezvous is possible with a given amount of fuel. In cases of marginal fuel supply, a homing system may exhaust the fuel before the rendezvous is completed. A closed-loop optimal system might also do this.

The possibility of allowing the system to run open-loop for considerable periods of time, as well as the actual utility of the optimal controller in any circumstance, depends on the sensitivity of the end conditions to errors in initial estimates of position and velocity. Sets of partial derivatives have been obtained for some of the trajectories selected as examples. Table 1 shows the matrix of partials for the case $A_{max} = .5$ ft/sec^2, $x_7(0) = 600$ ft/sec. Errors in the measurements of initial velocity have a large effect on the terminal errors. This effect depends strongly on the nature of the burning program.

TABLE 1

MATRIX OF PARTIAL DERIVATIVES $\left[\frac{\partial x_i(T)}{\partial x_j(0)}\right]$

j \ i	1	2	3	4	5	6	7
1	-.95	-.0001	.03	.0012	-.03	.0004	0
2	-360.96	.3807	593.01	1.8042	34.38	.1022	0
3	-.58	-.0022	-2.05	-.0039	.06	.0003	0
4	-610.20	-1.5034	-670.36	-.8906	8.59	-.0486	0
5	0.	.0001	.02	-.0004	-.68	.0008	0
6	17.19	.0321	0	-.1379	-653.17	-.6724	0
7	0	0	0	0	0	0	0

These partial derivatives represent the effect of applying the incorrect control, based on estimates of the system state vector, to the vehicle located at the nominal initial point. The propagation of actual initial velocity and position errors for a given control law can be calculated easily from the fundamental matrix solution. The coefficients given here involve the change in the optimal control with change in initial state vector; a somewhat more difficult calculation.

The second example is typical of a class of abort maneuvers and provides a more interesting class of trajectories than the first example. The maneuvering vehicle is initially moving away from the target; it becomes necessary to return to the target vehicle as soon as possible.

The initial conditions are $x_1=x_3=x_5= 60{,}000$ ft . $x_2=x_4=x_6= 100$ ft/sec .

The variety of optimal paths found, for the thrust levels and fuel allocations studied range from continuous burning programs with no coast periods to triple burn, two coast period programs. Some of these multiple coast arcs have very short burn times, giving rise to near impulsive thrust transfers.

Figure 13 shows the $x_1 - x_3$ projections of the motions. The thrust level is fixed at 1 ft/sec^2 and the fuel allocation is reduced from $\Delta V = 600$ to $\Delta V = 250$ ft/sec. The optimal time varies from 17 min. for the case $\Delta V = 600$ ft/sec. to 55 min. for the case $\Delta V = 300$ ft/sec. and 67 min. for $\Delta V = 250$ ft/sec. As the fuel allocation is reduced the excursions in x_1 increase to a maximum of 195,000 ft. The out-of-plane motions are depicted in an $x_5 - x_6$ plot in Figure 14. Figure 15 shows the propellant time histories $x_7(t)$. The steering angles θ and ϕ are given in Figures 16 and 17. The motion and optimal steering program for these maneuvers are clearly three dimensional in nature. In the absence of the tidal forces the time-optimal policy for these initial conditions would be to thrust directly back at the target along the velocity vector, coast if necessary, and then apply braking thrust. An inspection of the trajectories and the steering angles shows that this field-free analysis is very far from the time-optimal program. The out-of-plane correction is initially small, $\theta \sim 20^\circ$; the initial in-plane corrections are nearly perpendicular to the velocity and line of sight vectors. The out-of-plane errors are reduced during the coasting phase as are the planar errors. The final braking maneuver divides the thrust between the in-plane and out-of-plane components, i.e., $\theta \sim 40^\circ$. The three burn case is different. The intermediate pulse is directed mainly in the x_5 direction and sharply reduces the out-of-plane velocity at a time when the out-of-plane error is small. The final pulse is mainly in the $x_1 - x_3$ plane, as shown in Figure 16.

The error sensitivities for typical trajectories of this class have also been obtained. Tables 2, 3 and 4 show the matrices of partial derivatives for the initial condition $x_1 = x_3 = x_5 = 60{,}000$ ft., $x_2 = x_4 = x_6 = 100$ ft/sec., $A_{max} = 1.0$ ft/sec^2 and for $\Delta V = 600$, 450 and 250 ft/sec., respectively.

TABLE 2

MEASUREMENT ERROR SENSITIVITY COEFFICIENTS $\left[\frac{\partial x_i(T)}{\partial x_j(0)}\right]$

$A_{max} = 1.0$ ft/sec^2 $\Delta V = 600$ ft/sec

j \ i	1	2	3	4	5	6	7
1	-.95	-.0036	0	.0008	-.05	-.0025	.0045
2	-137.51	1.0593	1048.51	1.9201	25.78	-.3143	.5273
3	-1.57	-.0089	-2.88	-.0022	.03	-.0034	.0058
4	-1100.08	-4.3326	-764.90	.2983	0	-1.7650	3.1170
5	-.03	-.0018	-.02	.0004	-.30	-.0002	.0022
6	-42.97	-1.1476	0	.2894	-747.71	-1.2348	1.4368
7	-8.59	.9495	-8.59	-.2459	0	.6724	-2.2075

TABLE 3

MEASUREMENT ERROR SENSITIVITY COEFFICIENTS $\left[\frac{\partial x_i(T)}{\partial x_j(0)}\right]$

$A_{max} = 1.0$ ft/sec^2 $\Delta V = 450$ ft/sec

j \ i	1	2	3	4	5	6	7
1	-.96	-.0047	.07	.0039	.01	-.0044	.0075
2	653.17	2.9518	1753.25	2.1082	0	-.1307	.1831
3	-3.56	-.0148	-4.08	.0025	.01	-.0070	.0120
4	-1753.25	-6.1922	-825.06	3.4236	34.38	-3.8879	6.6691
5	-.02	-1.7092	.05	.0017	.06	-.0008	.0033
6	8.59	-.0020	34.38	1.2890	850.84	-1.4426	2.4873
7	-17.19	2.4996	0	-1.9638	-8.59	2.2843	-4.9296

TABLE 4

MEASUREMENT ERROR SENSITIVITY COEFFICIENTS $\left[\frac{\partial x_i(T)}{\partial x_j(0)}\right]$

$A_{max} = 1.0$ ft/sec^2 $\Delta V = 250$ ft/sec

j \ i	1	2	3	4	5	6	7
1	-1.04	.0002	0	.0022	-.02	.0017	.0028
2	15,581.59	2.6177	1727.47	-6.4954	17.19	-3.5556	-5.8168
3	-34.16	-.0055	-4.02	.0194	-.05	.0129	.0207
4	-1667.31	3.0149	885.22	9.2189	42.97	7.4381	11.9647
5	.05	.0009	.04	.0082	.01	.0054	.0105
6	8.59	.5799	-8.59	6.6294	842.25	5.4075	8.6040
7	-120.32	-1.3787	-77.35	-8.7561	0	-7.0444	-12.5179

The sensitivity to initial velocity measurement errors is very large, especially in the case of the triple burn path.

The effect of reducing the thrust level can be seen by comparing Tables 5 and 6 with Tables 2 and 3.

TABLE 5

MEASUREMENT ERROR SENSITIVITY COEFFICIENTS $\left[\frac{\partial x_i(T)}{\partial x_j(0)}\right]$

$A_{max} = .5$ ft/sec^2 $\Delta V = 600$ ft/sec

j \ i	1	2	3	4	5	6	7
1	-.95	-.0015	.07	.0014	-.01	-.0012	.0024
2	1314.94	4.1585	2131.40	1.7832	17.19	.1383	-.3354
3	-5.06	-.0121	-4.69	-.0006	.03	-.0025	.0051
4	-2122.81	-3.8029	-807.87	1.7860	8.59	-1.3727	2.7873
5	-.06	-.0004	.04	.0004	.27	.0007	.0006
6	8.59	-.4945	42.97	.4876	-842.25	.1871	.8185
7	-8.59	.3257	-8.59	-.2867	0	.2471	-1.5098

TABLE 6

MEASUREMENT ERROR SENSITIVITY COEFFICIENTS $\left[\frac{\partial x_i(T)}{\partial x_j(0)}\right]$

$A_{max} = .5$ ft/sec^2 $\Delta V = 450$ ft/sec

i \ j	1	2	3	4	5	6	7
1	-.94	-.0014	.07	.0026	-.01	-.0019	.0036
2	2827.55	5.7803	2715.82	.8806	0	.5349	-1.0447
3	-8.19	-.0151	-5.77	.0037	.01	-.0049	.0092
4	2707.23	-3.8176	-721.93	4.0860	0	-2.5895	4.8871
5	-.06	-.0005	.02	.0009	.54	.0002	.0012
6	25.78	-.4343	42.97	.8942	-687.55	-.0933	1.2045
7	-17.19	.8442	0	-1.3820	0	1.0235	-2.9248

The reduction in thrust acceleration level causes a general increase in the sensitivity to measurement errors. We see that a low thrust acceleration system does not necessarily result in a system with low sensitivity to errors.

A collision between the rendezvous vehicle and the target vehicle can occur if the initial planar errors are zero, i.e., $x_1 = x_2 = x_3 = x_4 = 0$. If no control is applied there will be a collision at the line of nodes of the two orbits. The equations for the out-of-plane motion in this particular case are simply

$$\ddot{x}_5 + \omega^2 x_5 = A(t)\, u_3$$

The time optimal control for this system with a fuel constraint is of the bang-off-bang type. The possibility of collision in the optimally controlled case depends on the initial conditions; specifically the value of the integral of the uncontrolled motion, $x_5^2 + (\frac{x_6}{\omega})^2$. For large values of this quantity there is no way to manipulate the control, u_3, to avoid a collision.

The difficulty is easily corrected by using u_1 and u_2, i.e., ϕ, to change the relative planar motions. Automatic terminal rendezvous systems, whatever the basis of the control and guidance designs, should provide for the situation described above. It seems unlikely that the two orbits would be established so perfectly that the relative local planar motions would coincide; but the possibility of this happening is there.

The relationships between rendezvous time, thrust acceleration level, and fuel are shown in Figure 18. For a given amount of fuel the time decreases as the thrust increases. The time decreases as the fuel increases keeping the thrust acceleration fixed. Neither result is very surprising. The sensitivities to initial velocity errors discussed previously generally increase in magnitude as the time increases. The tabulated results show that this is not true for all cases studied.

The three burn cases(300 and 250 in Figures 14, 15, 16, and 17) are nearly three inpulse transfers and are extremely sensitive to initial velocity errors.

SECTION 5

SUMMARY AND CONCLUSIONS

We have studied a linearized three-dimensional time optimal rendezvous with bounds on the thrust and on the fuel. The problem is formulated using the maximum principle to find the relations between the adjoint variables and the optimal steering laws and burning program. The solution of the two-point boundary value problem is obtained by applying Neustadt's method. This method requires the maximum value of a function of several variables to be found. The location of the maximum point must be found precisely if the boundary conditions are to be satisfied accurately.

In the rendezvous studies, three methods for maximizing functions were tried. These are:

(1) Powell's method

(2) Modified Steepest Ascent

(3) Fletcher-Powell modification of Davidon's method

The Fletcher-Powell-Davidon method gave the best results in terms of the number of steps required for convergence to the desired values of the terminal conditions. These results show that it is possible to obtain the optimal control law and corresponding solutions to the two-point boundary problem in a reasonable number of iterations. The next task is to find ways to use these solutions to control a rendezvous maneuver.

The use of time optimal control in a closed feedback loop with an inequality constraint on the fuel can lead to trouble if there is any noise in the measurement of the state variables. Filtering the measurements will reduce the problem but will not overcome it. The main difficulty is the possibility of errors in burning program, e.g., spurious pulses being applied during a coast phase. The fuel remaining after the termination of a burning period may be just sufficient to complete the rendezvous, so that a spurious pulse may put the rendezvous out of reach or make a collision inevitable. If the nominal path uses up all the fuel allocated, there is no margin for errors. If the braking pulse is inadequate, the vehicles collide. Pulses cannot be prohibited a priori because certain optimal trajectories do call for short duration pulses as part of the control program. One proposal is to hold back some of the fuel and use it later for corrections. The optimal controller can still run closed-loop, but the propellant <u>allocated</u> for the maneuver is adjusted to fit the situation.

One plan is to use the optimal path as a nominal trajectory and steer about this using a conventional controller. The simplest scheme is to let the system run open-loop until a final correction is made. If the nominal paths can be generated rapidly on board, it should be possible for a pilot to allocate fuel (less than the total available), find the time-optimal path, and repeat the calculation using fuel as a parameter until he finds a fuel-time combination that he likes. This has an advantage over the dual formulation of picking a time and minimizing fuel. The minimum fuel corresponding to an arbitrary choice of rendezvous time may exceed the fuel available, so that the computation will have to be repeated with an increased time; this can be avoided by making the initial choice of time very large. If the rendezvous problem has a solution at all, then the time-optimal controller finds one on the first choice of fuel, i.e., fuel allocated = fuel available.

A combination of the two schemes is appealing. The dual problems are so similar computationally that the same computer could handle either with minor modifications to the main program. The optimal time, T^o , can be found by using the fuel available as a constraint. A nominal path can be selected using the minimum fuel program with a time $t > T^o$.

Section 6

References

1. Neustadt, L. W., Jour. Math. Analysis and Application, Vol. 1, No. 3-4, December 1960.

2. Neustadt, L. W., S.I.A.M. Jour. on Control, Vol. 1, No. 1, 1963.

3. Sears, N. E. Jr., and Felleman, P. G., A.R.S. Jour., Vol. 30, No. 8, August 1960.

4. Clohessy, W. H., and Wiltshire, R. S., Jour. Aerospace Sciences, Vol. 27, No. 9, September 1960.

5. Wheelon, A. D., Ch. 26 Space Technology Wiley, N. Y., 1959.

6. Hord, R. A., NACA TN 4399, 1958.

7. Eggleston, J., NASA TN D-1029, February 1962.

8. Stapleford, R. L., ASD TDR-62-83, July 1962.

9. Hornby, H., NASA TN D-1207, March 1962.

10. Cicolani, L. S., NASA TN D-772, April 1961.

11. Carney, T. M., NASA TN D-923, 1962.

12. Passera, A. L., IRE - PGANE, Vol. 7, No. 4, December 1960.

13. Goldstein, A. D., Greene, A. H., Johnson, A. T., and Seidman, T. I., AIAA paper No. 63-354, August 1963.

14. Kelley, H. J., and Dunn, J. C., Proc. 2nd IFAC Congress on Automatic Control, Basle, September 1963.

15. Hinz, H. K., AIAA Journal 1, pp. 1367-1371 (1963).

16. McIntyre, J., and Crocco, L., AIAA Preprint 64-29 (January 1964).

17. Brissenden, R. F., etal., NASA TN D-747, 1961.

18. Beasley, G. P., NASA TN D-1613, March 1963.

19. Steffan, K. F., ARS Jour., Vol. 31, No. 11, November 1961.

20. Green, W. G., IRE PGANE, June 1961, p. 59.

21. Paiewonsky, B., et al. ASD TDR-63-239, 1963.

22. Fletcher R. and Powell, M. J. D., The Computer Journal, July 1963, p. 163.

23. Davidon, W. L., Variable Metric Method for Minimization,ARGONNE Nat. Lab. Rept. ANL 5990-Rev. May 1959.

24. Pontryagin, L., Boltyanskii, V. G., Gamkrelidze, R. V., and Mishchenko, E. F., The Mathematical Theory of Optimal Processes, Interscience, 1962.

25. LaSalle, J. P., Proc. Nat. Acad. Sci. USA 46 (1959) pp. 573-577.

26. Paiewonsky B. H., Woodrow, P. J., AIAA paper 63-224, June 1963.

27. Powell, M. J. D., The Computer Journal, July 1962.

28. Woodrow, P. J., Princeton Univ. Dept. of Elec. Eng. Control Systems Lab. TR No. 4, November 1963.

APPENDIX I

ACCELERATED CONVERGENCE TECHNIQUES FOR FINDING MAXIMA AND MINIMA OF FUNCTIONS OF SEVERAL VARIABLES

1 - Powell's Method

Powell's method is a recursive procedure for finding local extrema of multivariate functions. It is an extension of the accelerated-gradient method (27,28) and will always converge to the extremum of any unimodal function. In order to understand Powell's method, it is necessary to understand the philosophy of a recursive procedure. The difference between a recursive procedure and an iterative one is basically the difference between starting at the top and working down and starting at the bottom and working up. For example, if $f(n) = n!$, then $f(n)$ can be written recursively, $f(n) = nf(n-1)$, or iteratively

$$f(n) = \prod_{m=1}^{n} m$$

A flow diagram of a recursive procedure to compute $n!$ would resemble that in Figure 19.

Powell's method in much the same way uses the location of the extremum in a constrained (n-1)-dimensional space to locate the extremum in an n-dimensional space. Since the extremum in a one-dimensional space can be located by any number of methods (e.g., the method of false position to locate the zero of the derivative of the function), then the extremum can be located in an arbitrary n-dimensional space. The series of steps used to locate a maximum in n-dimensional space using Powell's method are:

1. Start with an n-variable problem and an initial point (A).
2. Compute the gradient at A.
3. Step in the direction of the gradient until the local maximum (B) in that direction is located (a single variable problem in terms of distance along a line through A, parallel to the gradient).
4. In the (n-1)-dimensional hyperplane (P), perpendicular to the gradient (or to the line AB) at B, locate the local maximum (C) of the function (an (n-1)-variable problem).
5. If step four can be done, locate the local maximum (D) on the line AC (another single-variable problem).
6. If step four cannot be done, consider the (n-1)-dimensional local maximization as a new problem and repeat steps two to four with A' now the point B for a solution to step four (an (n-1)-variable problem).
7. etc.

It is clear that eventually the hyperplane in step four will reduce to a one-dimensional line and the one-variable problem can be solved as mentioned above. The point D(=C) for the one-dimensional problem is then used in the two-dimensional problem to find D there. This point D is in turn used in the three-dimensional problem and so on until the point D is found in the (n-2)-dimensional hyperplane. This is used as the maximum in the (n-1)-dimensional hyperplane of step four.

If the function is a quadratic form, then point D can be shown to be the location of the maximum of the function. If the function is unimodal, then point D can be shown to be an improvement over point A, the improvement being greater as point A approaches the extremum point, where a quadratic form is usually a good approximation to the function. If the function is multimodal, the steps above have to be modified so that the closest maximum is always selected.

If the function in question is a quadratic form in two variables, $f(x_1,x_2)$, then Powell's method will operate as shown in Figure 20. The contours of equal functional value are concentric ellipses with center at the extremum. It is easy to show that the line connecting the two points, where a pair of parallel hyperplanes (lines in the two-variable case) are tangent to two equal function contours (two ellipses in the two-variable case) for the quadratic form, passes through the extremum values of the variables. If the function in question is a quadratic form in three variables, $f(x_1,x_2,x_3)$, then the real recursive nature of Powell's method can be shown as in Figure 21. The unprimed letters refer to the original n(3)-dimensional problem, the primed letters to the (n-1)(2)-dimensional problem in the hyperplane P through B, perpendicular to the gradient of the function at A, and the double-primed letters to the one-dimensional problem, the last step down in the recursive process. The contours of equal functional value are now concentric ellipsoids. The two-dimensional "cut-out" shows how the maximum in the two-dimensional plane, perpendicular to the gradient, is found.

The problem of computing gradients in constrained hyperplanes has not been discussed because it is quite complex and is adequately treated in (28). The problem of finding the maximum along a line (steps 2 and 4) is essentially the same as that of determining the extremum in one-dimensional space and is also treated in (28). It is worthwhile to note that the derivative of the function along a line at a point ζ is given by the dot-product of the "direction" of the line and the gradient at point ζ. Then step three above could have been written as "... until the point (B) is reached, where the derivative of the function with respect to distance traveled is zero ..." or "... until the point (B) is reached, where the gradient of the function is perpendicular to the gradient of the function at the point A...".

2 - Fletcher-Powell-Davidon Method

The geometrical interpretation of Davidon's variable metric method will be sketched briefly here. A complete description can be found in the original papers referenced earlier. The method is an iterative gradient technique. It consists in calculating the gradient of $f(x)$ for successive values of the n-vector x and attempting to find the places where $\nabla f = 0$, and $\left\| \frac{\partial^2 f}{\partial x_i \partial x_j} \right\|$ is positive definite. If the function being minimized is quadratic, then $\left\| \frac{\partial^2 f}{\partial x_i \partial x_j} \right\|$ is a matrix of constants. If this matrix (the Hessian) is known then the minimum point can be found in one step

$$x_o - x = - \left[\frac{\partial^2 f}{\partial x_i \partial x_j} \right]^{-1} \nabla f(x)$$

For a general function the Hessian is not a constant but is a function of position. Furthermore the Hessian is often not known explicitly and can be obtained only by calculation.

In Davidon's method the inverse of the Hessian is not computed directly but successive approximations are made. A positive definite symmetric matrix H is chosen initially and is modified at each step according to a prescribed rule. That is, instead of stepping in the direction opposite the gradient at all times (as in the steepest descent method) the direction of the step is modified so that the step at the ith stage is taken in the direction $H^{(i)} . \nabla f^{(i)}$. The unit matrix is a satisfactory first choice for H . The sequence of operations at the ith stage is listed below. The superscript indicates the stage of the iterations.

1. Find the optimum step in the direction $-H^{(i)} \nabla f(x^i)$
2. Evaluate $f(x^{i+1})$ and $\nabla f(x^{i+1})$
3. Set $y^i = \nabla f(x^{i+1}) - \nabla f(x^i)$
4. Modify $H : H^{i+1} = H^i + A^i + B^i$
   The matrices A^i and B^i are obtained by calculation according to these definitions

$$A^i = \frac{\alpha^i[-H^i \nabla f(x^i)] \otimes \alpha^i[-H^i \nabla f(x^i)]}{[-\alpha^i H^i \nabla f(x^i)] \cdot y^i}$$

$$B^i = \frac{-H^i \cdot y^i \otimes y^i \cdot H^i}{[y^i \cdot H^i \cdot y^i]}$$

The symbol $a \otimes b$ means that a linear operator D_{ij} is formed from vectors a and b ; the matrix elements of this operator are: $D_{ij} = a_i b_j$.

α^i is a positive constant corresponding to the length of the optimal step at the ith stage.

Fletcher and Powell show that the process is stable and that the minimum of a quadratic form in n variables is obtained in n iterations.

APPENDIX II

COMPUTER PROGRAM FOR FINDING THE EXTREMUM OF A MULTIVARIATE FUNCTION

The following sections (A--F) will enable the reader to use the program developed for this contract to solve the problem of finding the extremum of a multivariate function. Although the subroutines were used in a specific application, they were written in a sufficiently general way so that any individual could use them for his own application. FORTRAN II source program IBM cards are available from either ARAP or WPAFB.

The theory behind the various methods is discussed at some length in Reference (28) except for the Fletcher-Powell method, the theory of which is discussed in Reference (22). A set of operational notes describing the program inputs and outputs (but not internal operation) is available at ARAP for two versions of the whole rendezvous program used in the simulation study. These programs were designed for a specific purpose and, unless a possible user has the same purpose in mind, the programs will probably be unusable. One version injects random additive measurement noise and attempts to compensate by feedback, and the other computes partial derivatives of the terminal state with respect to trajectory. FORTRAN II source program IBM cards are available from WPAFB.

A. Explanation of Subroutines

1. This section attempts to explain the workings and usage of a set of subroutines for the problem of finding the extremum of a multivariate function. The various additional subroutines that are necessary are listed and the manner of linking these to the available ones is explained. The various inputs that must be provided are listed and explained in some detail. Due to the peculiarities of each machine installation and to the particular application in mind, there are some changes that may need to be made in the given subroutines, but these are minor and associated only with the output of the various quantities. All necessary modifications are noted and explained.

SEARCH — This is the main subroutine of the package and is the one which is called upon to seek the extrema. The function of this subroutine is to execute all major initialization, to perform the necessary recursion, and to find the gradient in the hyperplane given either the directional derivatives or the actual gradient at a point. Since the program is capable of being used both for problems in which directional derivatives can be determined directly and for problems in which the gradient must always be found, SEARCH provides the logic for differentiating between these and performing the necessary computation for each.

ØPSTEP — The function of this subroutine is to determine a sequence of steps in a direction given by SEARCH such that the sequence converges to the optimum step. An optimum step is defined as that step in the given direction such that the gradient is perpendicular to the given direction at the end of the step.

MKSTEP — The function of this subroutine is to actually make a step along a given direction, check to see whether the step was acceptable (i.e., did not overstep any imposed boundaries), check to see whether the step was "optimum" to the desired tolerance, check to see whether the maximum number of permissible steps has been exceeded and execute all desired printing.

VVP — The function of this subroutine is to compute the dot product of two given vectors.

NØRM — The function of this subroutine is to compute the norm of a given vector as well as the normalized version of the vector.

MVØVMP — The function of this subroutine is to compute the product of a matrix times a vector or a vector times a matrix as called for and output the resultant vector.

B. Additional Necessary Subroutines

(MAIN) — (The name may be anything and in fact this may be the main program). The function of this program is to do the necessary inputing and to call SEARCH.

GRAD — The function of this subroutine is to compute the gradient or directional derivatives as called for by SEARCH and MKSTEP.

PAGECK — (This may be just a dummy subroutine). The function of this subroutine is to provide ejecting to a fresh page when necessary and desired page heading.

PRNTSP — The function of this subroutine is to print any additional variables desired beyond those printed by MKSTEP. Printing occurs following printing by MKSTEP.

C. Points of Interest to the User

1. The one parameter of SEARCH, NØKFL indicates the success of the convergence:

NØKFL = 1 - Convergence successful

= 2 - Convergence unsuccessful in specified number of cycles

= 3 - Somebody goofed

a. The first guess is not acceptable.
b. A guess is acceptable to GRAD and the same guess is not acceptable on a later call.

= 4,5,etc. May be used by GRAD to terminate convergence and indicate reasons.

2. If an appropriate PAGECK is provided and some printing is desired, the first printing will take place on a fresh page and page ejection and heading will be provided on subsequent pages.

3. NPRTFL offers numerous forms of printing. The thousands digit keys the printing at the end of each step, the hundreds at the end of each optimum step, the tens at the end of each cycle, and the units at the end of the convergence (terminated by success or by too many cycles). The highest order digit greater than 0 controls the MKSTEP printing. If, for example, the hundreds digit is 1 or greater and the thousands digit is 0, then printing will occur after every optimum step. Use of digits greater than 1 will be explained in the section on the construction of GRAD.

4. There are two basic modes of operation and five auxiliary modes controlled by the input MØDE. If MØDE is odd (1,3,5,7,9), then the SEARCH package presumes that GRAD will calculate directional derivatives directly. If MØDE is even (2,4,6,8,10), then the SEARCH package

presumes that GRAD always computes the gradient. If

MØDE = 1,2 - Normal Powell's method

= 3,4 - Powell's method without optimum steps during dimension reduction part of cycle.

= 5,6 - Just optimum steps

= 7,8 - Just standard steps with the requirement that a good step be made and with provision for doubling the step-size for the next step whenever the directional derivative at the end of a step is greater than 50% of the derivative at the start of the step. (Same provision holds for MØDE = 3,4).

= 9,10 - Fletcher-Powell-Davidon

(N.B. Use of all five modes of operation on one representative seven-dimensional problem revealed that modes 1,2, 7,8, or 9,10 were practical with the first two modes requiring almost the same number of steps and 9,10 almost half. For MØDE 1 or 7, MØDE 1 would have required less computation.)

5. The SEARCH package is presently written so that GRAD must check to see whether convergence has been accomplished and to inform the SEARCH package. In addition, GRAD must inform the SEARCH package whenever a "point" lies outside any of the imposed boundaries.

6. VVP may be used by any other subroutine. The arguments are V1, V2, V1V2, N respectively. V1 is one of the 10-dimensional vectors, V2 the other, V1V2 the dot product result, and N the actual number of dimensions.

7. NØRM may be used by any other subroutine. The arguments are V, VN, VL, N respectively. V is the 10-dimensional input vector, VN the resultant normalized version of V, VL the norm of V, and N the actual number of dimensions.

8. MVØVMP may be used by any other subroutine. The arguments are AM, V1, V2, NR, NC, NCØDE respectively. AM is the 10 × 10 matrix, V1 the vector multiplying AM or multiplied by AM, V2 the resultant vector, NR the actual number of rows in AM, NC the actual number of columns in AM, and NCØDE indicating whether matrix-vector or vector-matrix multiplication is desired. If NCØDE=1, matrix-vector multiplication is performed and, if NCØDE=2, vector-matrix multiplication is performed.

9. During the dimension reduction part of a Powell cycle, if MØDE is odd, SEARCH forms an orthogonal matrix along whose columns GRAD determines directional derivatives. SEARCH (via MVØVMP) then multiplies the vector of directional derivatives by the transpose of the orthogonal matrix to obtain the actual gradient in the hyperplane. If MØDE is even, SEARCH computes the gradient in the hyperplane by subtracting from the actual gradient the components of the actual gradient that lie along the previous higher-dimensional hyperplane gradients.

10. One of the inputs to the SEARCH package is STEPMN. If, during the process of trying to take a step such that the final product is at least positive, the ratio of the difference of two successive steps to the length of the position vector is less than STEPMN, then one of the following takes place: If the Fletcher-Powell mode is being used, it is restarted--if the desired step is still less than STEPMN, the special section below takes over; if the dimensional reduction section of a Powell's cycle is underway, it is stopped and dimensional increase started; if dimensional increase is underway, it is continued; if the dimensional reduction has just started (the step is taking place in the

full dimensional space), then the special section of SEARCH takes over. The special section examines the present gradient and starts stepping each of the components of the position vector starting with the one corresponding to the largest component of the present gradient. This process continues until a successful step is made along one of the components. At this point the normal procedure is applied. If no successful step can be made, the convergence process stops with NØKFL=3.

11. The optimum step subroutine, ØPSTEP, is completely different from that described in Reference 28. In particular, the initial guess for the optimum step size is chosen by means of a complex rule (except for Fletcher-Powell when the last optimum step is used). The step-size, STEP, is first set to the maximum of the previous optimum step-size (arbitrarily .05 to start) and STEPMN times the norm of the position vector. SMX is then set equal to the maximum of STEP and 10% of the norm and SMN to the minimum of the two quantities. STEP is then computed such that

$$\mathrm{SMX} \cdot 2^{-\alpha} = \mathrm{SMN} \cdot 4^{\alpha}$$

The logic behind this computation is somewhat abstruse and may become clearer through an explanation of the workings of ØPSTEP.

In computing the optimum step, two separate sections of ØPSTEP are usually encountered. The first section contains the programming necessary to accommodate a positive dot product of the gradient and direction vector. As long as the dot product remains positive, computation remains in the first section. In this section the next guess for STEP is ›uted by linearly extrapolating the previous two points to find STEP such that the dot product would be zero. If this linear extrapolation results in a negative step-size or in a step-size greater than FACMX (an input usually set equal to 4) times the previous step, FACMX times the previous step is chosen as the next guess.

Once the dot product goes negative, the computation enters the second section of ØPSTEP and remains there. In this section the present guess is always bracketed by two previous guesses--one for a smaller step-size (with a positive dot product), one for a larger (with a negative dot product). Both the interpolated and extrapolated guesses are computed. If both lie in the interval in which the actual solution is known to lie, and the difference between them is less than 25% of that interval, quadratic interpolation is used. If this is not the case, then, if quadratic interpolation yields a guess which is no further from the center point of that interval than 25% of the interval, then the quadratic interpolation guess is again used. Otherwise the center point of that interval is chosen.

12. MKSTEP is the subroutine which actually checks to see whether the optimum step has been found. The criterion for an acceptable optimum step is that the dot-product for that step be positive and less than ANGMN, the maximum of ANGAMN (an input depending on the problem) and GFAC (an input usually set equal to .01) times the dot product at the start of the step (i.e., for a step-size of zero). If any step is not acceptable to GRAD, then STEP is set equal to the average of STEP and the previous acceptable step (initially zero).

If the ratio of the difference between the present unacceptable step and the previous acceptable step to the norm of the present position vector ever falls below STEPMN, the most recent step which resulted in a positive dot product is accepted, regardless of the size of the dot product. The same procedure is followed whenever the number of guesses for an optimum step ever exceeds NSTPMX (an input usually set equal to 15). If, however, no previous acceptable step resulted in a positive dot product, then the present step-size is continually halved until either an acceptable step with a positive dot product has been found or until the ratio of the step-size to the norm of the position vector falls below STEPMN. In the latter case, STEP is set equal to zero and the search for an acceptable step is terminated. In addition, SEARCH is notified of this difficulty so that it may take special steps, if needed (see 10).

13. The quantities normally printed by MKSTEP when called for are the following:

NCYC - The number of complete Powell's cycles that have been completed (for MØDE=5,6,7, 8,9,10, NCYC is the number of (optimum) steps that have been completed).

ND - The number of dimensions of the present hyperplane while dimensional reduction is underway and lowest number reached while dimensional increase is underway.

NU - Equals zero during dimensional reduction and equals the number of dimensions of the present hyperplane during dimensional increase.

NSTEP - The number of steps that have been made in the present search for an acceptable (optimum) step.

NSTEPL - The maximum number of steps that have ever been required to find an acceptable (optimum) step during the present call of SEARCH. For reasons explained in (12) this number may exceed NSTPMX.

STEPI - The step-size at the start of the present step.

STEP - The step-size at the end of the present step (will only be different from STEPI if a boundary is encountered or if NSTPMX has been exceeded).

ANG - The dot product at the end of the present step.

ANGMN - The required tolerance for the present optimum step (see 12).

DDO - The dot product at the start of the present (optimum) step (i.e., for a step-size equal to zero).

SGNG - To find the actual direction vector for stepping, the components of DC below should be multiplied by this (remember that for minimization, steps are taken along the negative of the direction vector).

ETA(I) - The components of the present position vector.

DD(I) - The directional derivative(s). If MØDE is odd, only one number is printed which has the same magnitude as ANG. If MØDE is even, these numbers represent the present gradient as given by GRAD.

DC(I) - The direction cosines of the vector being stepped along (see SGNG above). The formation of these is explained in Ref. 28.

Following the above four lines of output will be any additional output desired by the user and executed in PRNTSP.

D. Instructions on the Preparation of the Additional Subroutines

1. Common Storage - In order to shorten the calls to various subroutines in the SEARCH package, most of the variables that are shared are stored in common. For any subroutine that is not involved in the SEARCH process the two statements

```
DIMENSIØN  WWWWW (279)
CØMMØN  WWWWW
```

will allow additional variables to be stored in common without interfering with the SEARCH package.

2. (MAIN) - This program must input (or compute) the 95 items needed by the SEARCH package, namely:

FMTSCH(36) - If PAGECK is written as intended, this array should contain the format corresponding to the heading used for printout of the SEARCH variables described in Section IV, 13. It may be read in using the format (3(12A6/)) and should resemble a FØRMAT statement except that the word FØRMAT is omitted, i.e., the format begins with a left parenthesis.

FMTPSP(36) - If PAGECK is written as intended, this array should contain the format corresponding to the heading used for the printout of any additional information in PRNTSP. Otherwise it is the same as FMTSCH above.

NSCHPL - The number of lines printed by MKSTEP (unless MKSTEP is changed, this should be 5).

NPSPL - The number of lines printed by PRNTSP.

MØDE - Explained above.

NDIM - The number of dimensions involved (the number of components in the position vector).

NPRTFL - Partially explained above and partially explained below in (3).

NCYCMX - The maximum permissible number of Powell cycles (for MØDE=5,6,9,10, the number of optimum steps; for MØDE=7,8, the number of "good" steps).

NSTPMX - The maximum permissible number of steps (or guesses) to find the optimum (or "good") step.

FMXMN - 1.0 for maximizing a function.
-1.0 for minimizing a function.

FACMX - Explained above.

GFAC - Explained above.

RADMN - If, during dimensional reduction with ND < NDIM, the norm of the gradient projected onto the hyperplane ever falls below RADMN, dimensional reduction is stopped and dimensional increase starts.

ANGAMN - The minimum criterion for an optimum step (see IV, 12 above).

STEPMN - Explained above.

ETAO(10) - The initial guess for the position vector. If this is not acceptable to GRAD, searching immediately terminates with NØKFL=3.

The "main" program must also call SEARCH using the statement CALL SEARCH (NØKFL). The values of NØKFL returned have been explained above. When SEARCH returns to the calling program the final position vector is given in double-precision by ETA(I) in common. In addition, DD(I) is the array of directional derivatives and NCYC holds the number of cycles completed.

3. GRAD - This program must actually compute either the gradient or specified directional derivatives as well as the function, if desired. The pertinent inputs to this program in common are:

NPRTFL - The thousands digit of this number may be used to key various types of printing. If, for instance, a trajectory must be computed in order to find the gradient, then if the thousands digit is 0 or 1, no printing takes place in GRAD; if it is 2, printing occurs only at a few specified points and, if it is 3, printing occurs at every point. There may be as many as nine different options including no printing and these in turn may vary depending on whether an optimum step has just been completed or a cycle, or convergence achieved, Returning to the example, the input to SEARCH may be 1023 for

NPRTFL. In this case, the normal SEARCH printing would take place at the end of every step and GRAD would see NPRTFL as 1023. At the end of a Powell cycle, MKSTEP would call GRAD once more with the same position vector and NPRTFL temporarily set equal to 2300. When convergence had been achieved or NCYC had exceeded NCYCMX, MKSTEP would call GRAD once more with the same position vector and NPRTFL temporarily set equal to 3000. This special printing at the end of a cycle and at the end of the convergence process would occur before the natural MKSTEP printout.

ETA(10) - The double-precision position vector. For many problems, the most significant part of each word is sufficient, but for those problems where it is not, the double-precision makes a big difference.

The GRAD heading should read something like SUBRØUTINE GRAD (NDD, NFINFL, DDM, NACCFL, NØKFL). NDD and DDM are meaningful only in case MØDE is odd (but DDM must always be dimensioned as DDM(10,10)). In this case NDD is the number of directional derivatives desired by the calling subroutine in the SEARCH package and the columns of DDM contain the direction cosines of the directions wanted. In other words if NDD=2, then two directional derivatives are called for and the NDIM direction-cosines of the first are in the first column and of the second in the second column of DDM. NFINFL, NACCFL, and NØKFL, are all output quantities of GRAD, specifically:

NFINFL - 1 if convergence not complete.
2 if convergence criteria satisfied.

NACCFL - 1 if present position vector acceptable.
2 if present position vector unacceptable (e.g., outside some imposed boundary).

NØKFL - 1 if convergence is to proceed naturally, 4 and up if convergence is to be immediately terminated. This value of NØKFL is communicated back to the "main" program.

Provided that NACCFL=1, NØKFL=1, the array DD(I) in common should contain either the NDD directional derivatives in order or the NDIM components of the gradient depending upon whether MØDE is odd or even respectively.

4. PRNTSP - whenever this is called, it is called by MKSTEP with the statement CALL PRNTSP (DD,2) or with CALL PRNTSP (DD,3). The latter call occurs only when a Powell cycle has been completed, convergence achieved, or NCYC > NCYCMX, and when sense switch 6 on the 7090 console has been depressed. Hence this call might cause PRNTSP to print out some information on-line in case the user is present at the 7090 and interested in such output. The CALL PRNTSP (DD,2) occurs whenever MKSTEP prints and just following such printout (depending on NPRTFL). Except that DD (or whatever dummy variable is used in PRNTSP) must be dimensioned DD(10) and that there must be two parameters, PRNTSP may be anything the user desires. In one case, PRNTSP was used to print the end point of a trajectory involved and in other just to print the function.

5. PAGECK - This routine may be a dummy just to satisfy FØRTRAN or it may be programmed as intended--to provide page headings on every new page. There are two types of calls of PAGECK in the SEARCH package, neither of them actually occuring unless NPRTFL > 0. The first call is CALL PAGECK (FMTSCH, FMTPSP, 0, 2) and was intended to start a fresh page each time SEARCH was called. The second call is CALL PAGECK (FMTSCH, FMTPSP, NSCHPL + NPSPL, 1) and it was intended that

PAGECK check to see whether NSCHPL + NPSPL lines could fit on the present page and if not, to eject to the next page and provide page headings. The PAGECK routine which is used at ARAP and which might provide an example calls a subroutine CHRON that might not be locally available. The function of CHRON is to interrogate the on-line timer and to yield two six Hollerith (alphanumeric) character words containing the date and time. PAGECK may be used, as is the ARAP version, to number pages successively, to provide page ejection with heading and to provide for initializing the page counter. The one feature of PAGECK that is mandatory is that it properly dimension the first two parameters as 36 word floating point arrays.

E. Possible Changes in SEARCH Package

The only changes that should ever be required in the SEARCH package are those in MKSTEP to provide an acceptable output format for ETA and DD. The statement numbered 1002 now provides the format for ETA, DD, and DC except that if any component of ETA falls below .001 or exceeds 100., then 1003 is used to print ETA. These two numbers are set by the first two executable statements of MKSTEP as ETMIN and ETMAX, respectively.

In general the WRITE ØUTPUT TAPE statements will have to be changed in MKSTEP to conform with a specific installation so that the tape called for (3 in the present version) is the correct one.

F. Afterthoughts

1. When MØDE is even, SEARCH provides one accuracy check. If, during dimensional reduction, the dot product of the gradient projected on the hyperplane with the

gradient is ever ≤ 0, then dimensional reduction stops and dimensional increase starts. The dot product should never actually be negative, but if there are a large number of dimensions, it is possible that sufficient loss of accuracy may occur so that it is in fact computed to be negative.

2. If there are no special criteria that have to be satisfied for over-all convergence, a useful one is that the norm of the gradient should fall below some experimentally determined number. In general, RADMN should also be set equal to this number and hence can be considered as an additional input to GRAD in common.

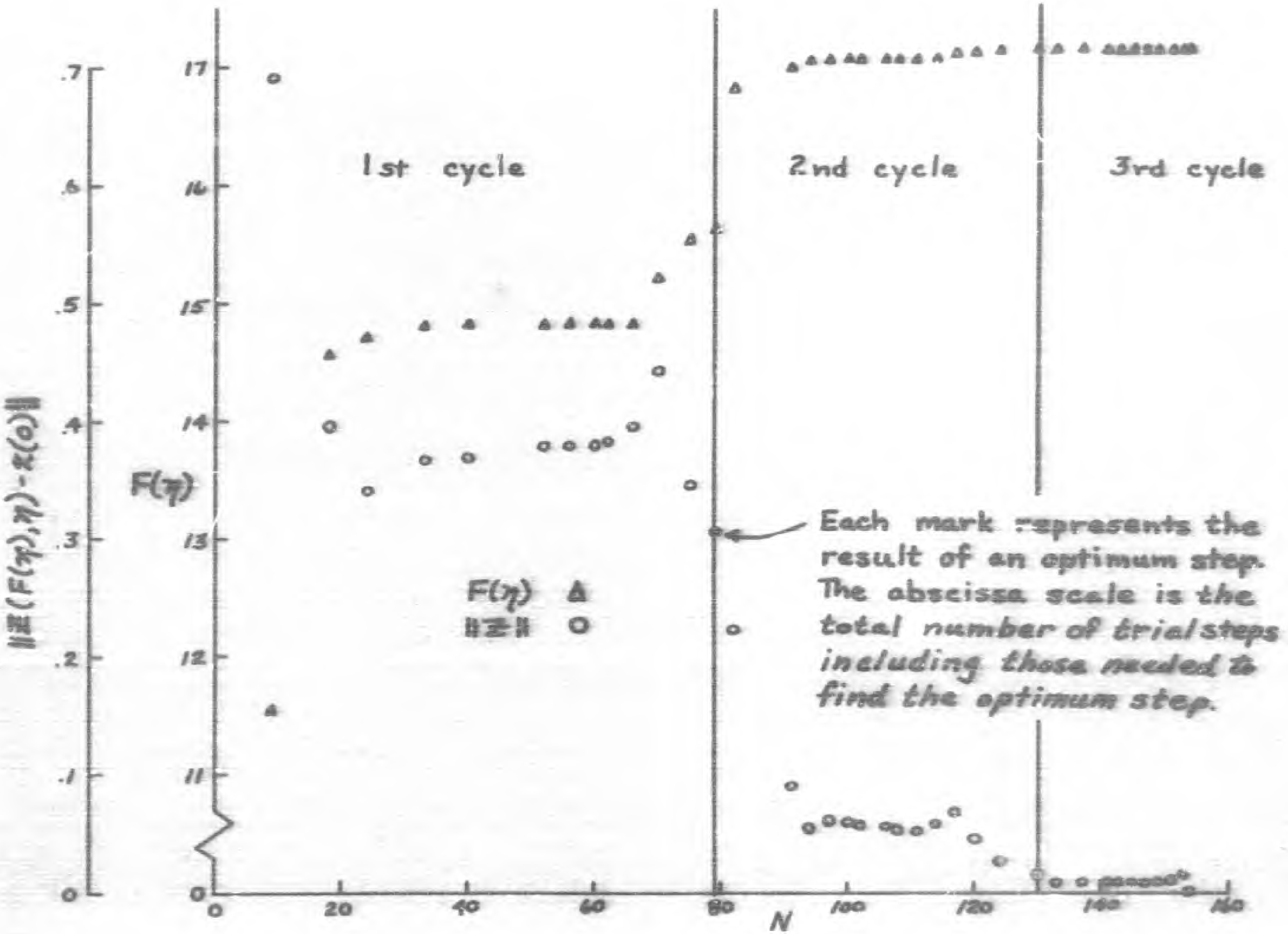


Figure 3. Gradient Length and Stopping Time vs. Total Number of Steps for Powell's Method.

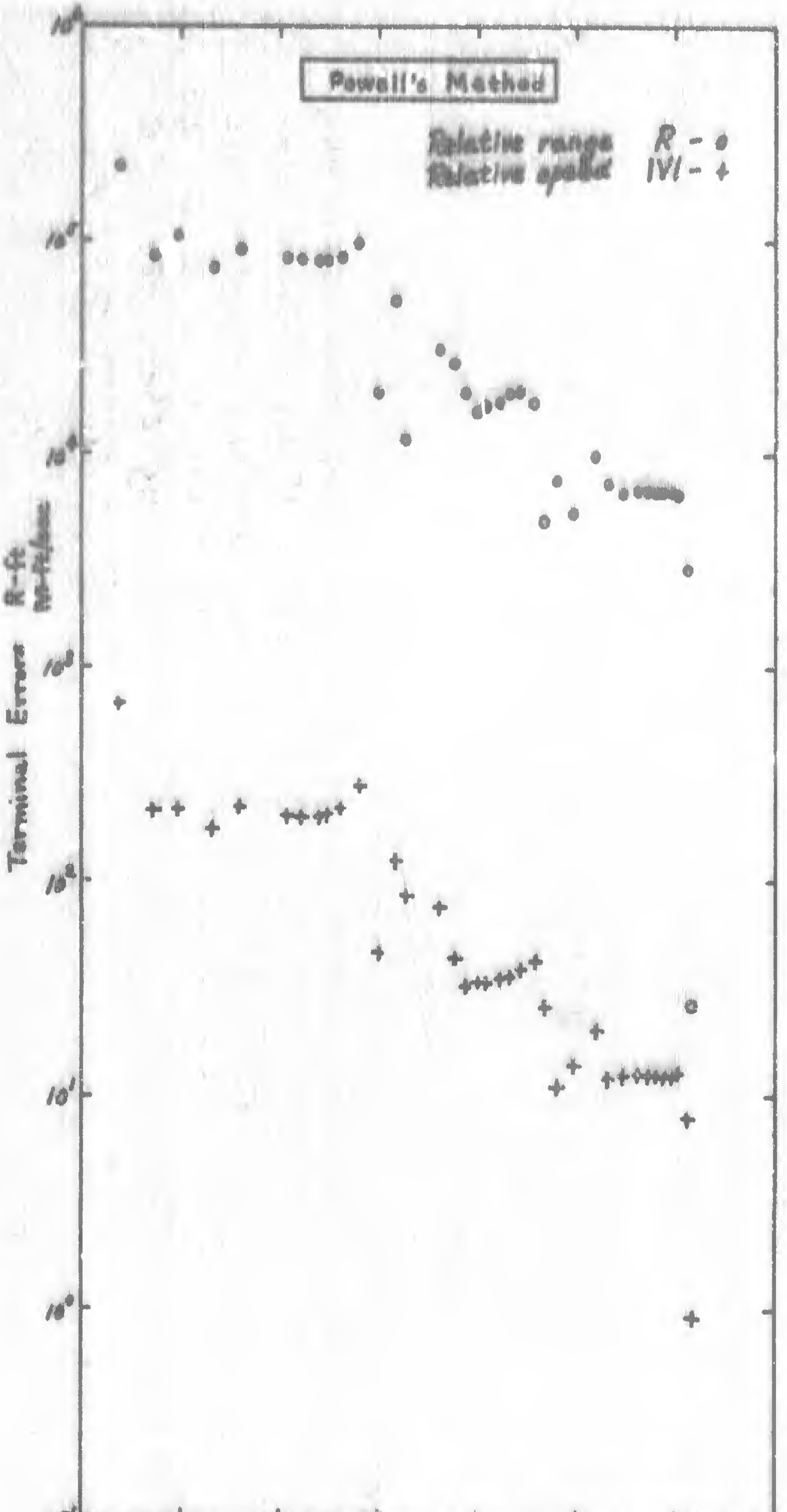


Figure 4. Terminal Errors Position and Velocity vs. Iteration Number

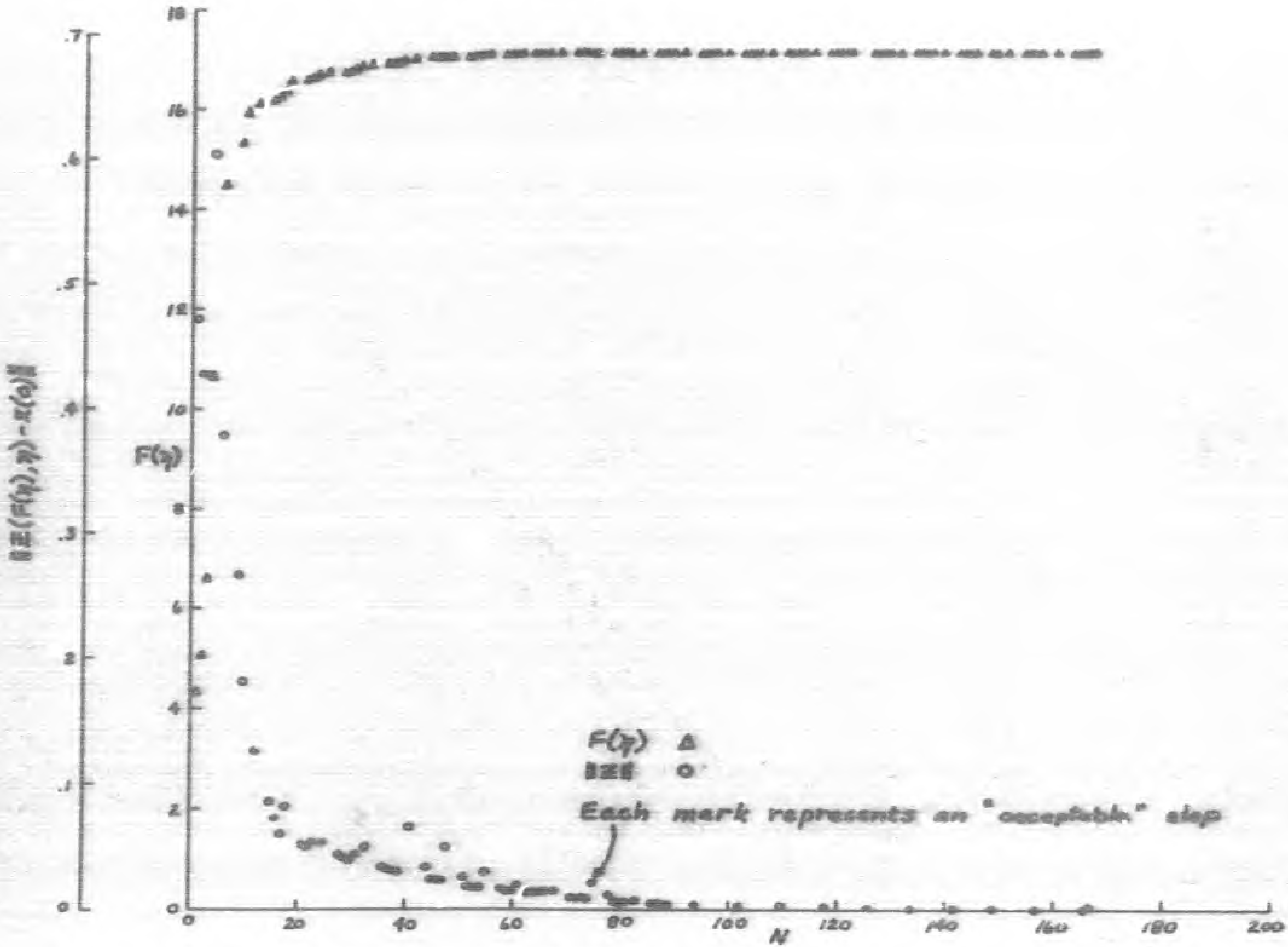


Figure 5. Gradient Length and Stopping Time vs. Total Number of Steps for Modified Method of Steepest Ascent

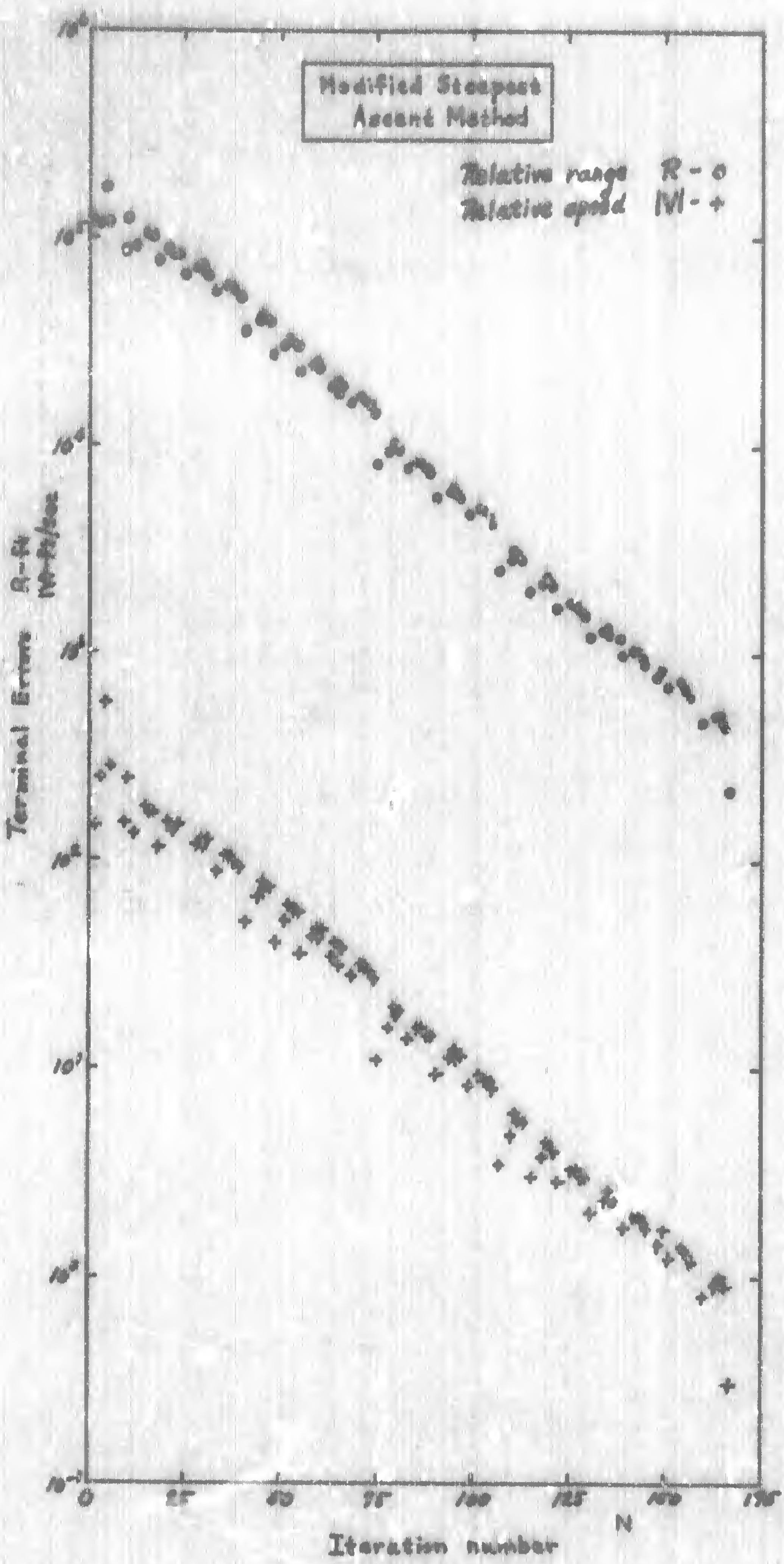


Figure 6. Terminal Errors vs. Iteration Number

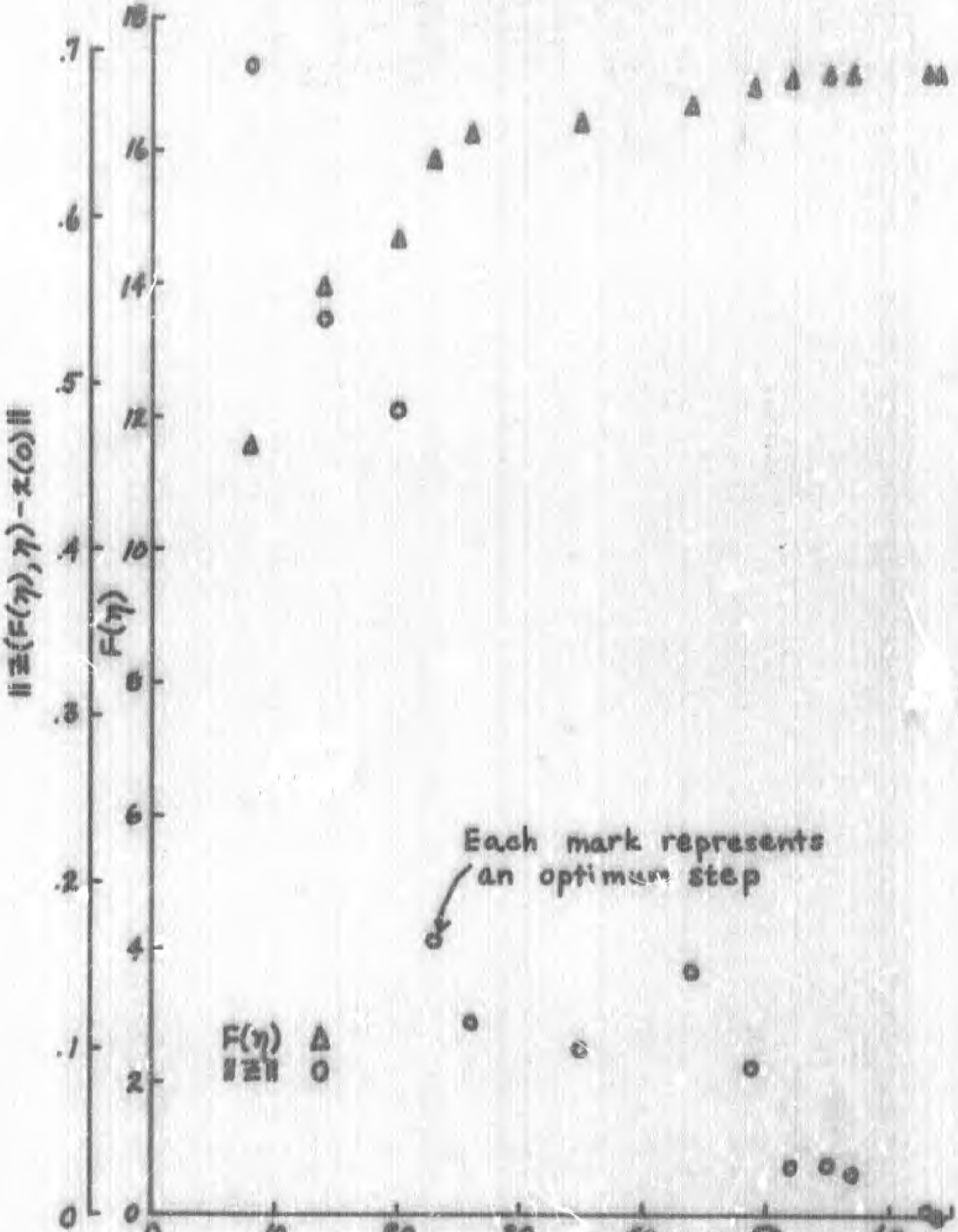


Figure 7. Gradient Length and Stopping Time vs. Total Number of Steps for Fletcher-Powell-Davidon Method

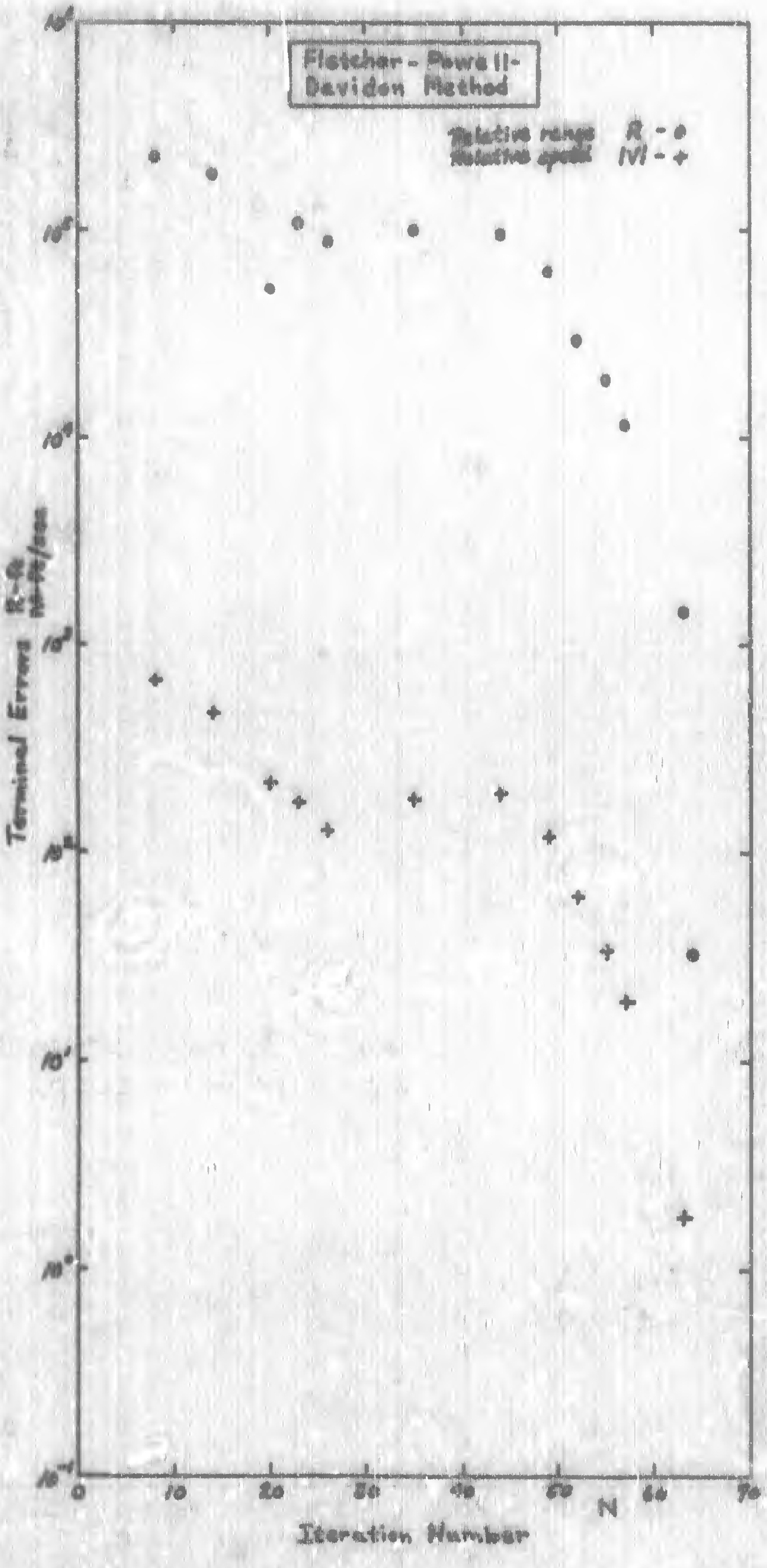


Figure 8. Terminal Errors vs. Iteration Number

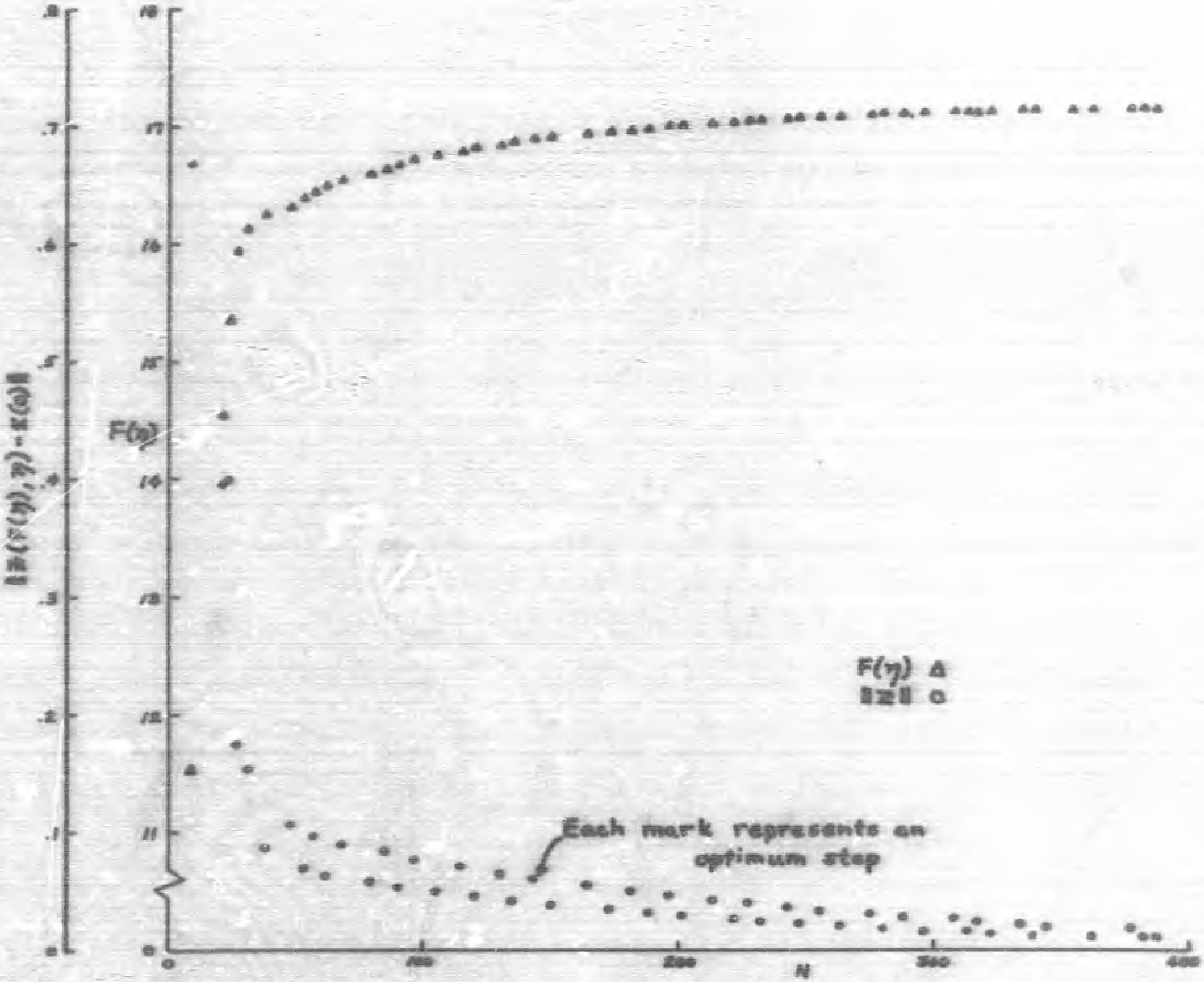


Figure 9. Gradient Length and Stopping Time vs. Total Number of Steps for Unaccelerated Steepest Ascent using Optimal Steps

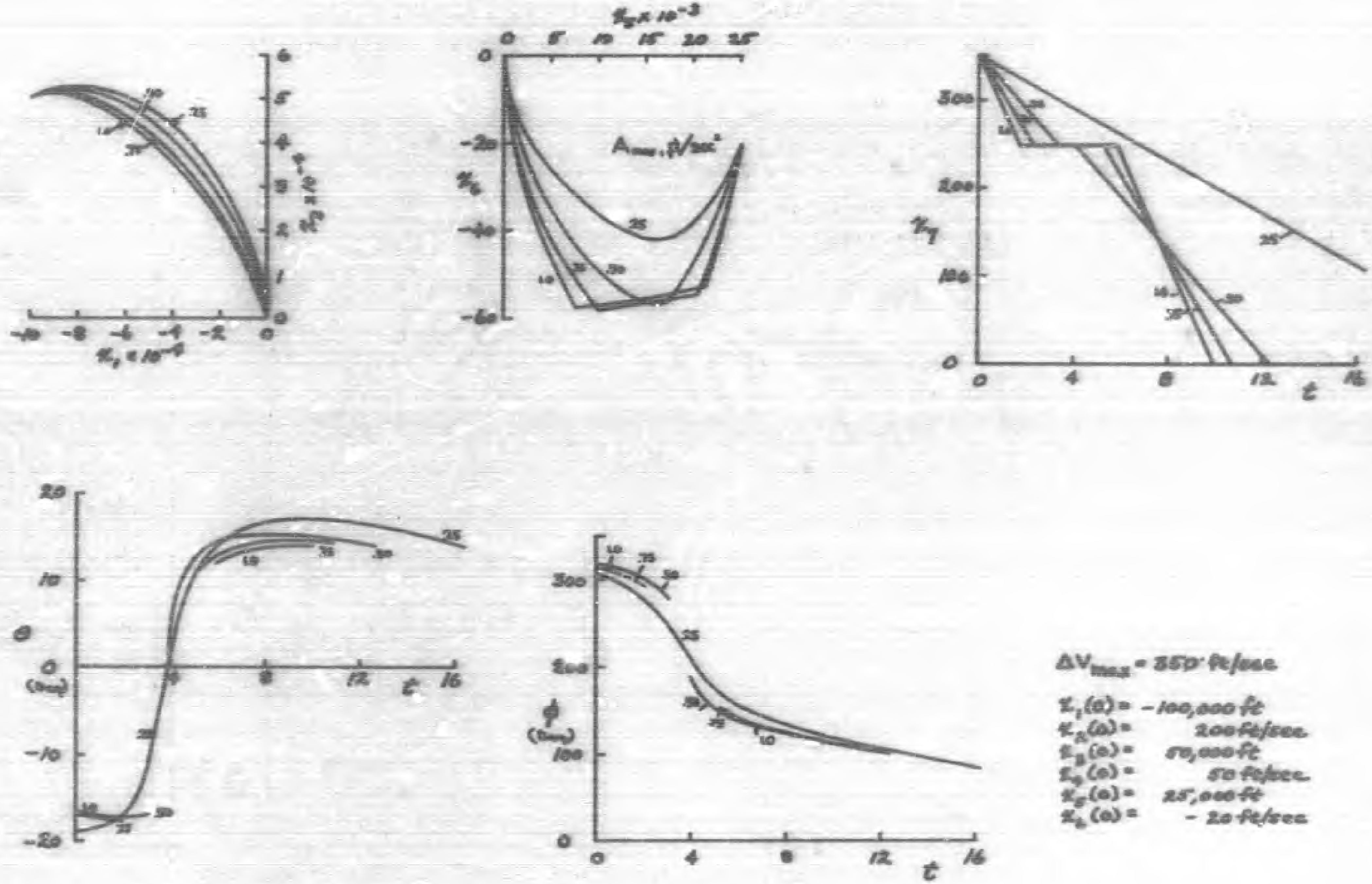


Figure 10. Rendezvous Trajectories and Control Variable Time Histories

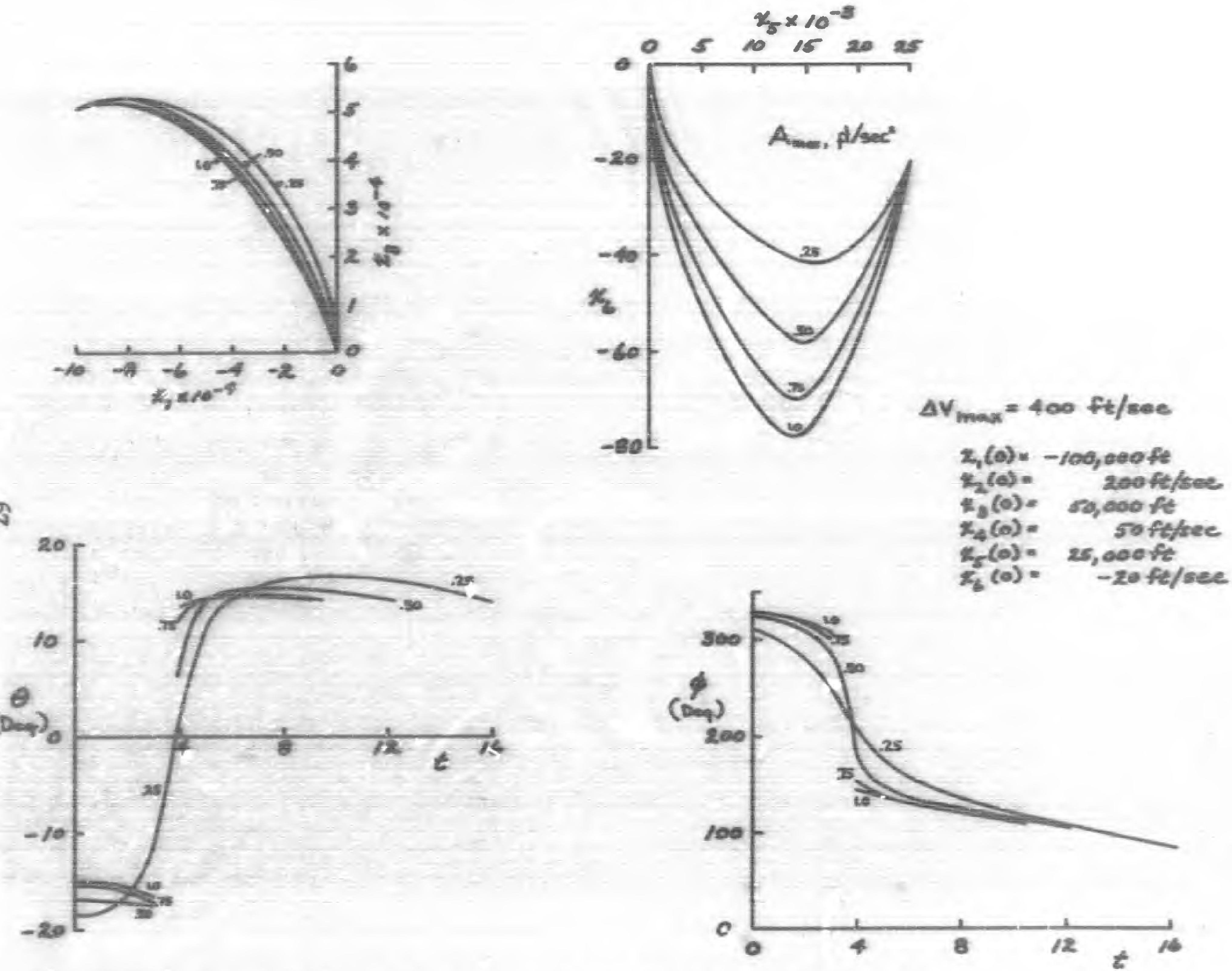


Figure 11. Rendezvous Trajectories and Control Variable Time Histories

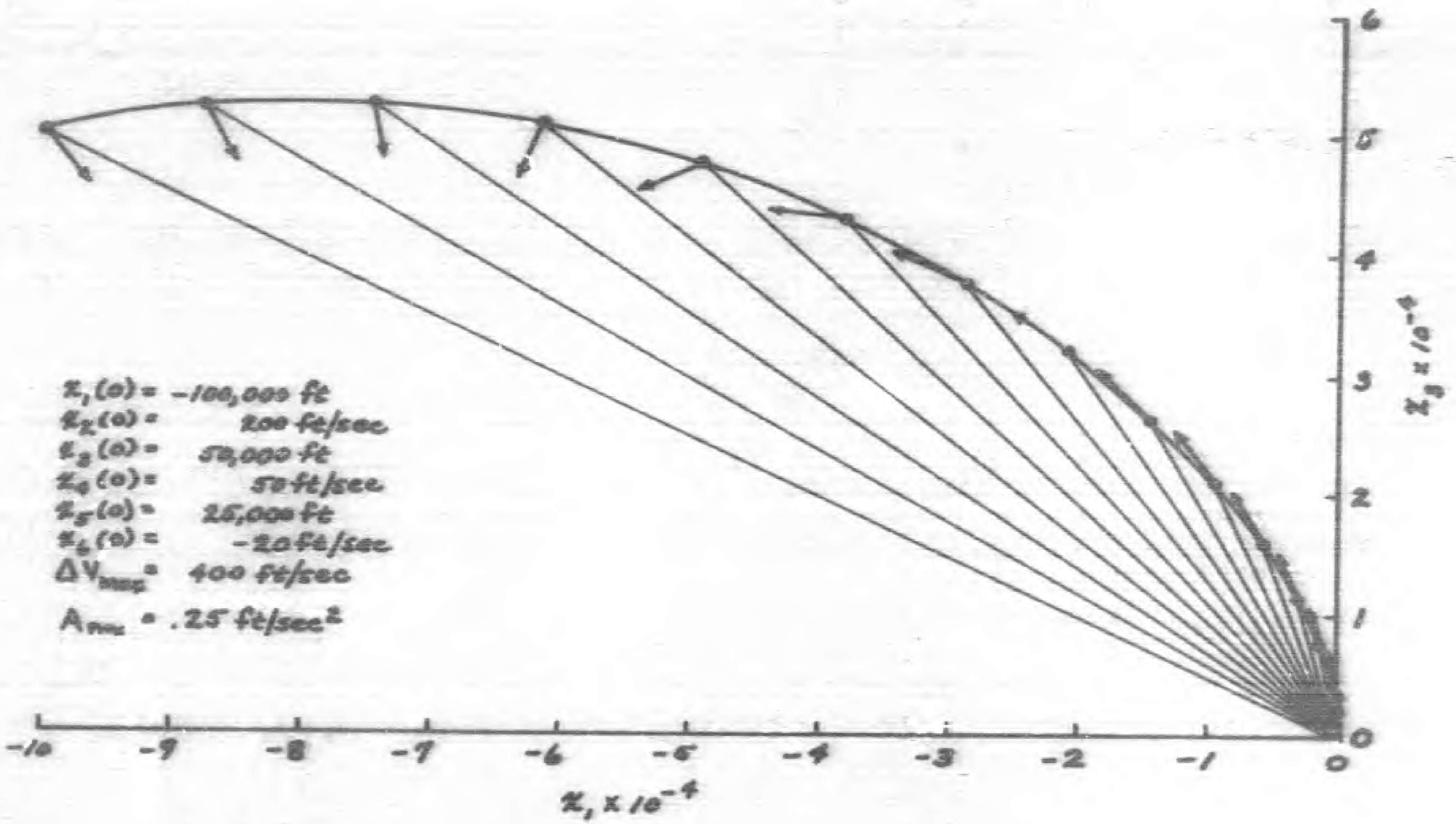


Figure 12. Projection of Optimal Thrust Direction on $x_1 - x_3$ Plane

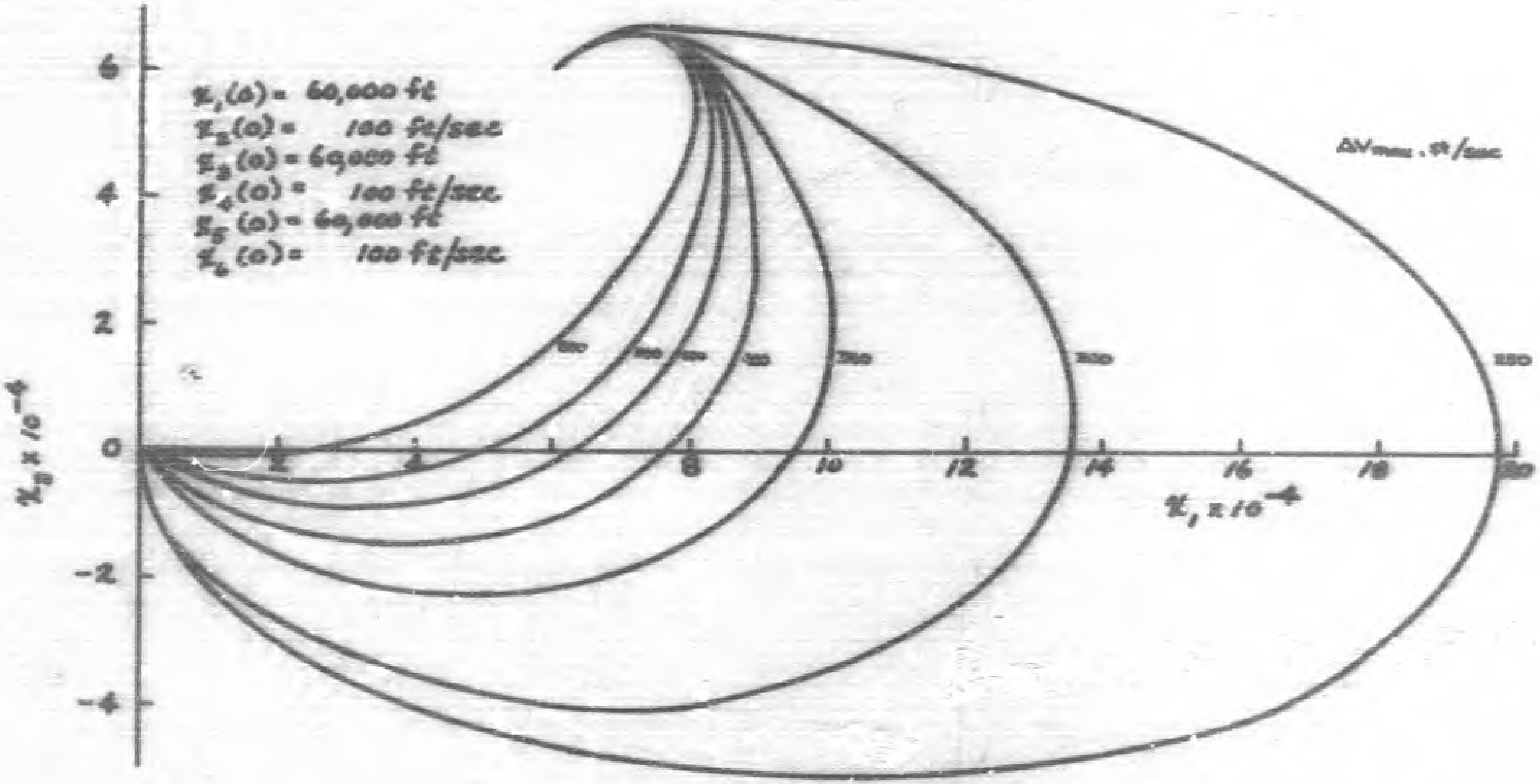


Figure 13. Motion in $x_1 - x_3$ Plane for $A_{max} = 1.0$ ft./sec.^{2} and Varying Amounts of Fuel

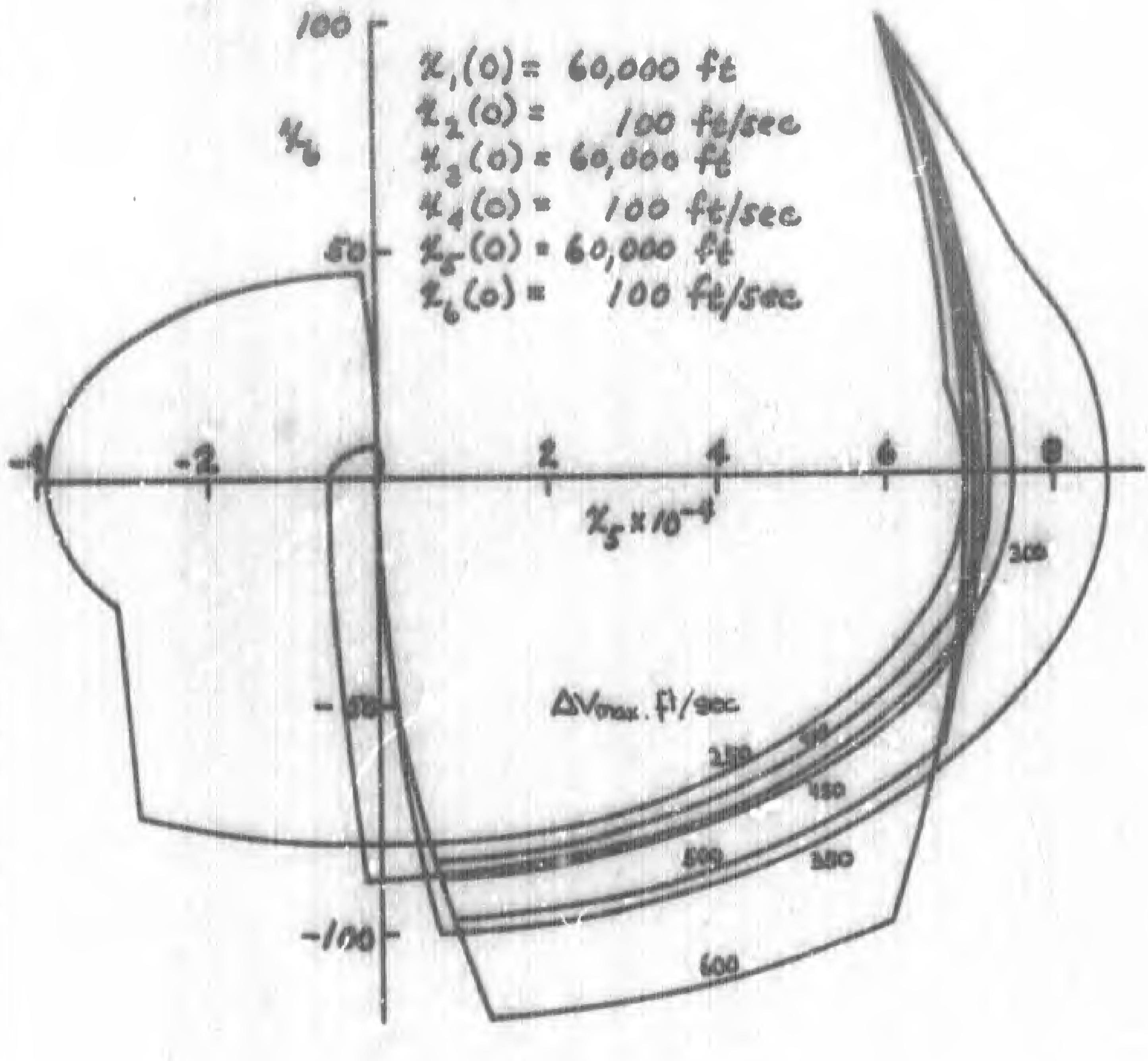


Figure 14. Plot of x_5 vs. x_6 for $A_{max} = 1.0$ ft./sec.^{2} and Varying Amounts of Fuel

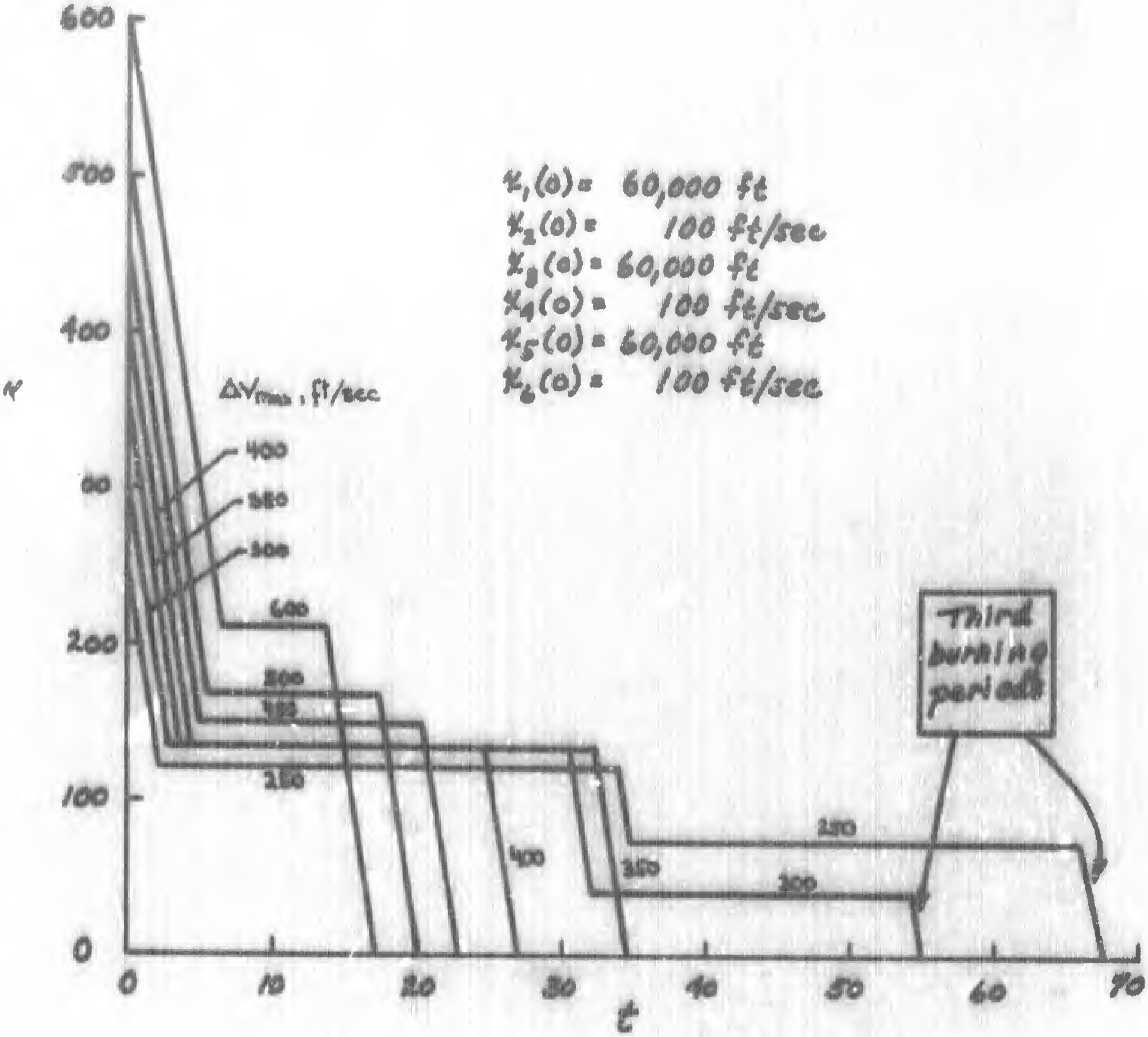


Figure 15. Fuel Time Histories, $A_{max} = 1.0$ ft./sec^2.

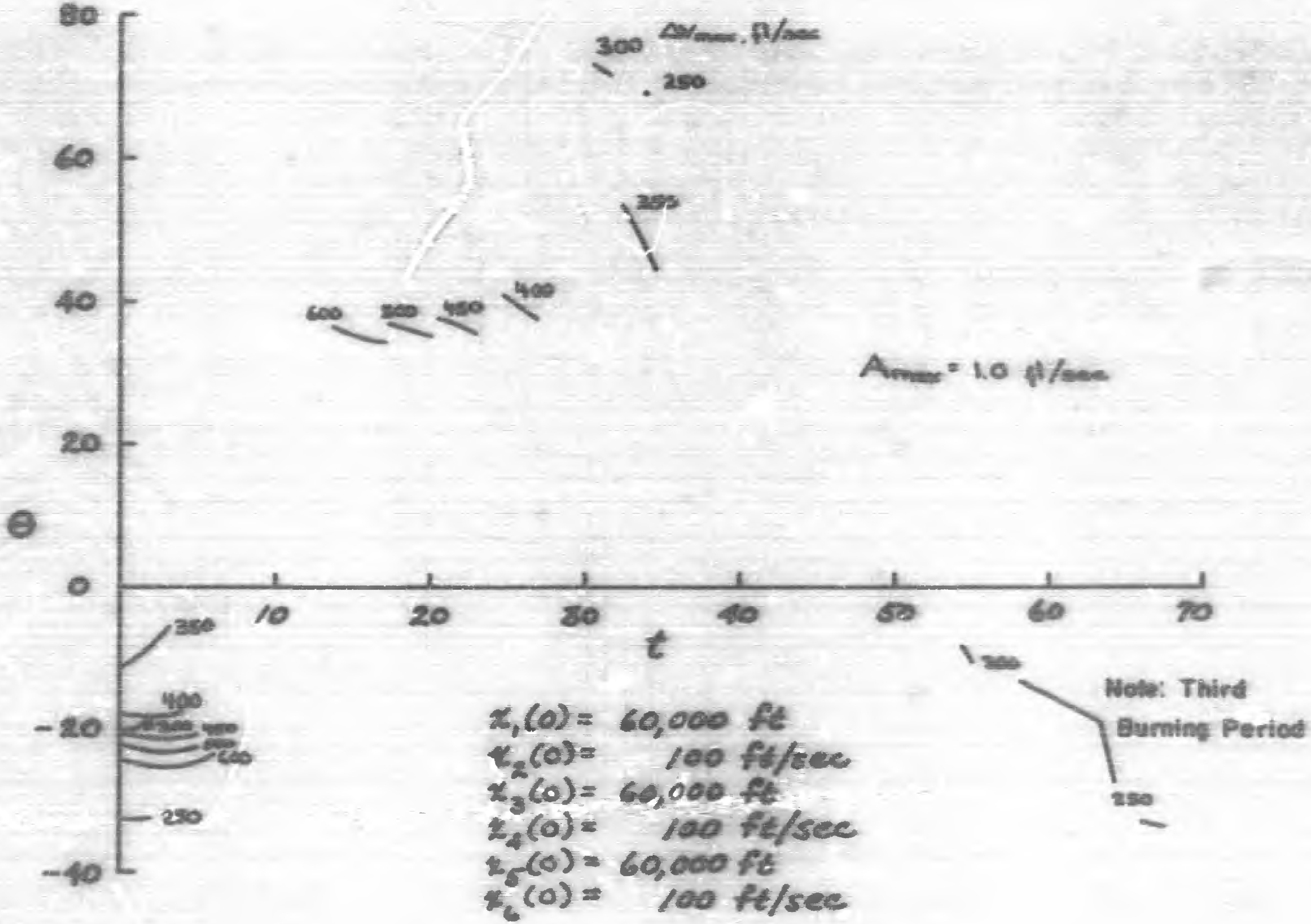


Figure 16. Steering Angle θ vs. Time

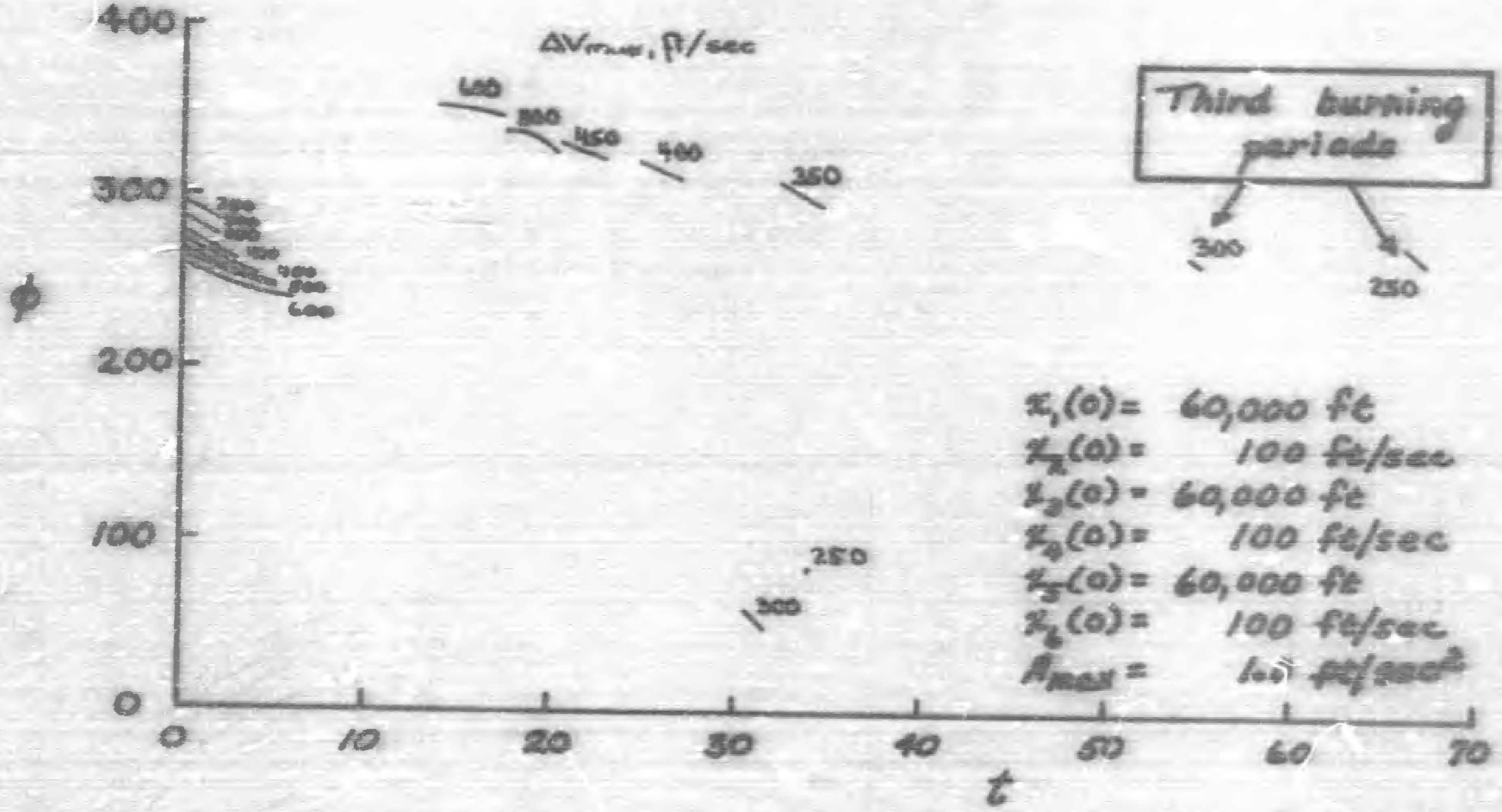
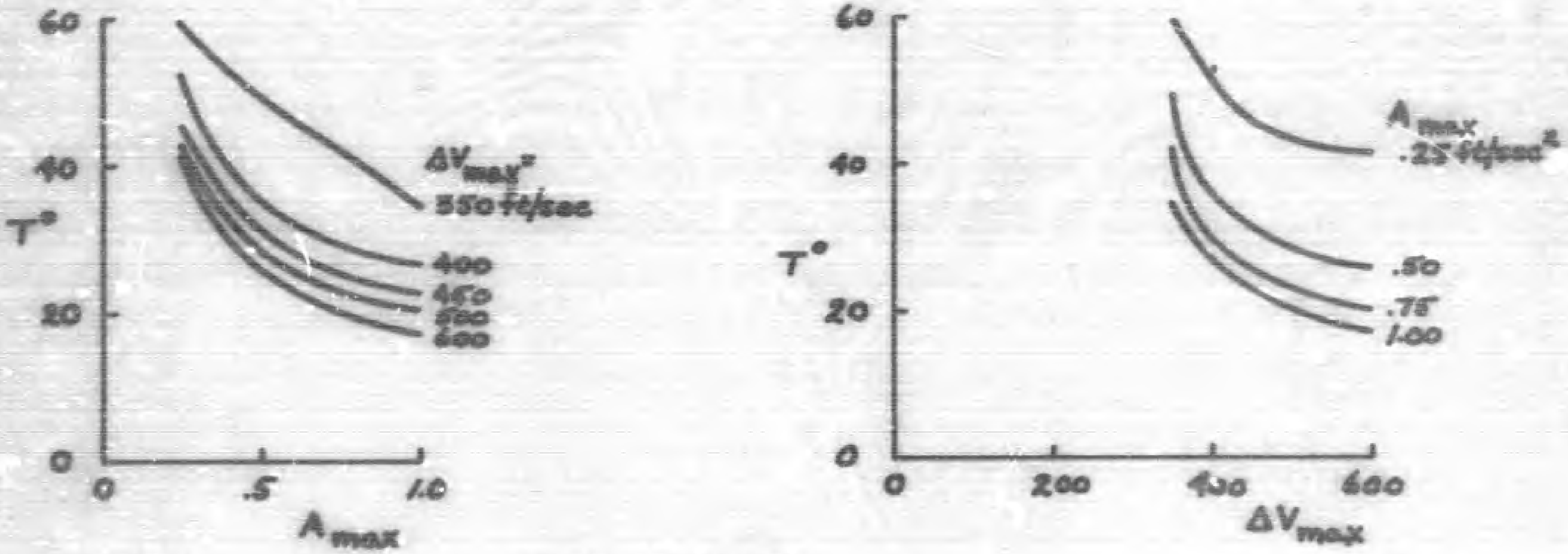


Figure 17. Steering Angle ϕ vs. Time



(a) Optimal Time
vs. Maximum Acceleration

$x_1(0) = 60{,}000$ ft
$x_3(0) = 60{,}000$ ft
$x_5(0) = 60{,}000$ ft
$x_2(0) = 100$ ft/sec
$x_4(0) = 100$ ft/sec
$x_6(0) = 100$ ft/sec

(b) Optimal Time
vs. Maximum Propellant Allocation

Figure 18. Optimal Time Tradeoffs

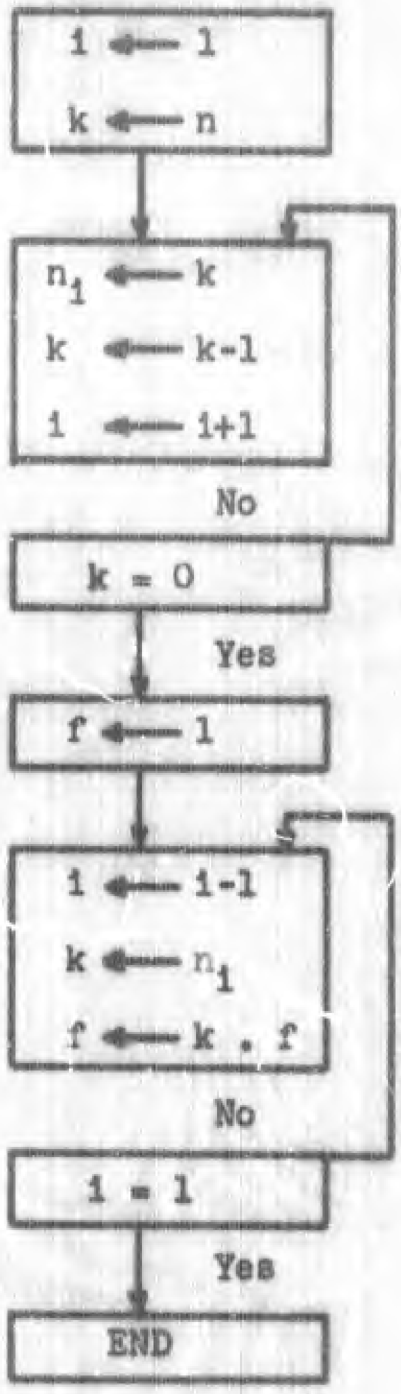


Figure 19. Flow Diagram for a Recursive Procedure

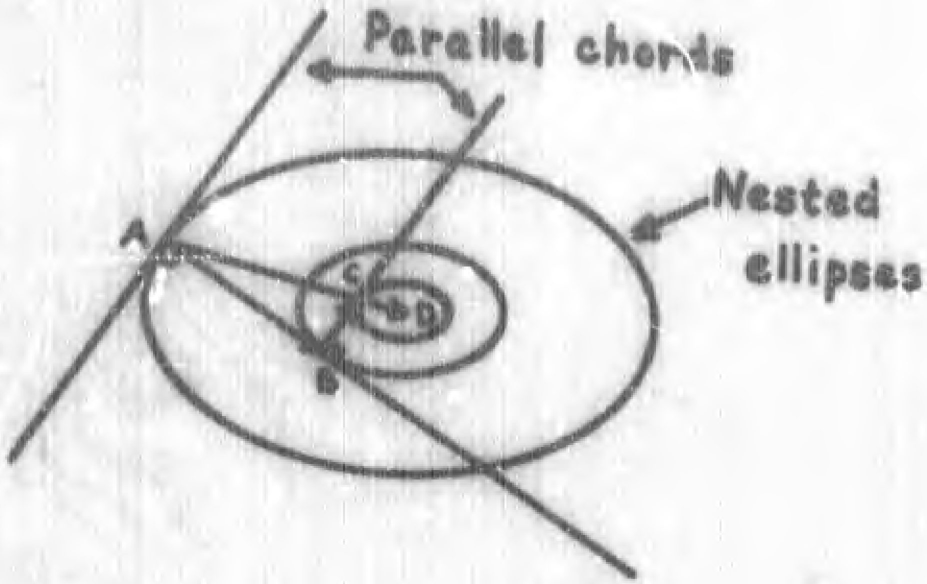


Figure 20. Example of Parallel Chords and Quadratic Functions

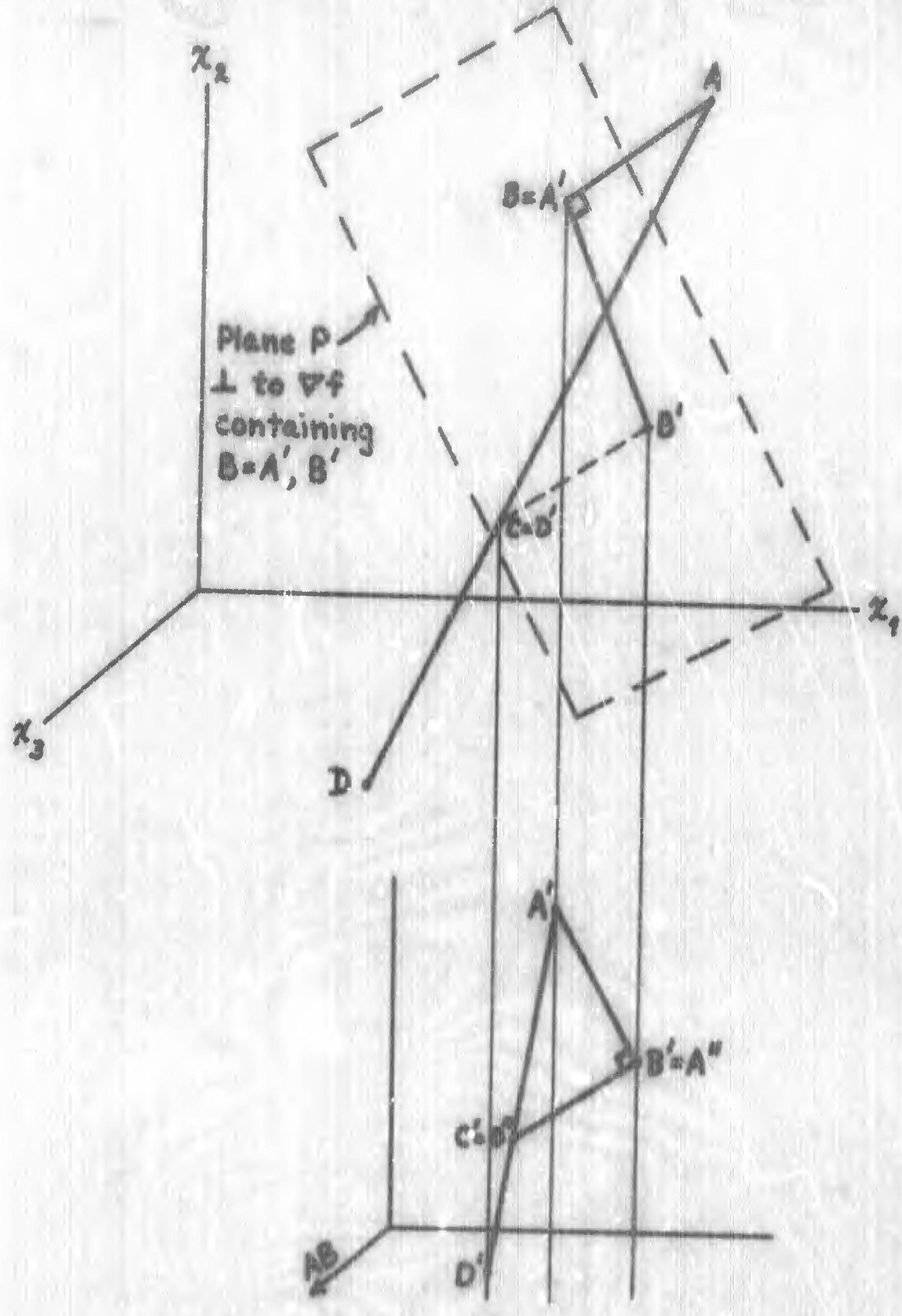


Figure 21. Powell's Method in Three Dimensions

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Aeronautical Research Associates of Princeton, Inc. 50 Washington Road (P.O. Box 229) Princeton, New Jersey 08540	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP N/A

3. REPORT TITLE

A Study of Time-Optimal Rendezvous in Three Dimensions (Vol. I)

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Final report

5. AUTHOR(S) (Last name, first name, initial)

Paiewonsky, Bernard H.
Woodrow, Peter J.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
January 1965	83	28

8a. CONTRACT OR GRANT NO. AF33(657)-11319 b. PROJECT NO. 8219 c. Task No. 821904 d.	9a. ORIGINATOR'S REPORT NUMBER(S) ARAP Report No. 69, Vol. I 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) AFFDL-TR-65-20, Vol. I

10. AVAILABILITY/LIMITATION NOTICES

Qualified requestors may obtain copies of this report from DDC, and the report may also be obtained from the Office of Technical Services, Department of Commerce.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
None	Air Force Flight Dynamics Laboratory Wright-Patterson AFB, Ohio 45433

13. ABSTRACT

The results of a study of time-optimal rendezvous in three dimensions with bounds on the rocket thrust and the available propellant are described. The equations of motion are linearized and Neustadt's method is used to solve the two-point boundary value problem in the seven-dimensional state space. Three convergence acceleration schemes are studied. Fletcher and Powell's modification of Davidon's method was superior to Powell's method and a modified method of steepest ascent. Examples of terminal rendezvous paths are presented and discussed in terms of the magnitudes of the bounds on thrust and fuel. The dependence of terminal errors on initial measurement errors in position and velocity is also discussed. The range of initial values includes position errors up to 25 miles and relative velocity errors of 200 ft/sec. The thrust accelerations of the rockets are on the order of 1 ft/sec^2; the propellant bounds (ideal characteristic velocities) range between 600 ft/sec and 250 ft/sec.

DD FORM 1 JAN 64 1473

Unclassified
Security Classification