AD-787 696

A SUBSONIC METHOD FOR PREDICTING THE SEPARATION TRAJECTORIES OF AIRCRAFT STORES IN THE PITCH PLANE

Joseph M. Manter

Air Force Institute of Technology
Wright-Patterson Air Force Base, Ohio

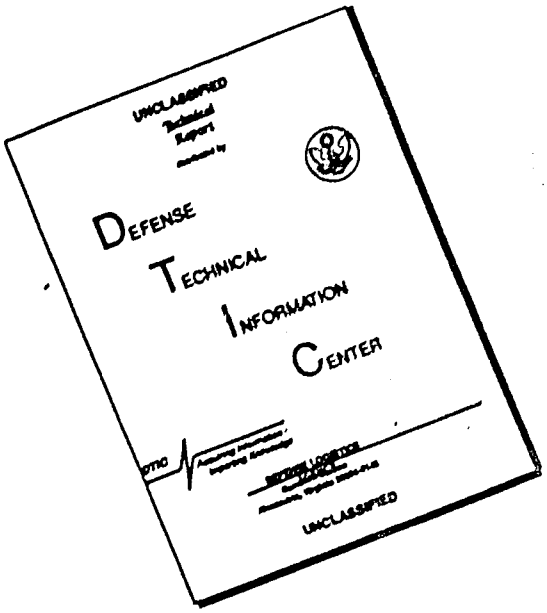
September 1974

DISTRIBUTED BY:

NTIS
National Technical Information Service
U. S. DEPARTMENT OF COMMERCE
5285 Port Royal Road, Springfield Va. 22151

PII Redacted

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

AD787696

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER GAE/AE/74S-4	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A SUBSONIC METHOD FOR PREDICTING THE SEPARATION TRAJECTORIES OF AIRCRAFT STORES IN THE PITCH PLANE		5. TYPE OF REPORT & PERIOD COVERED Thesis 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Joseph M. Manter		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Air Force Institute of Technology (AU) Wright-Patterson AFB, Ohio 45433		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE September 1974 13. NUMBER OF PAGES ~~58~~ 62
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Aeronautical Systems Division (ENFDA) Wright-Patterson AFB, Ohio 45433		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

NATIONAL TECHNICAL INFORMATION SERVICE

18. SUPPLEMENTARY NOTES

Approved for public release; IAW AFR/190-17

Jerry C. Hix, Capt, USAF
Director of Information

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

External Stores
Aerodynamic Interference
Fernandes Interference Method
Subsonic Flow
Field Flow
Store Trajectory Simulation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A subsonic three-degree-of-freedom method for predicting store separation trajectories is presented. The method combines the calculation of interference loading on aircraft stores due to F. Dan Fernandes with loads due to free air, ejector, and aerodynamic damping. Discussion of these loads is given, along with a presentation of the three-degree-of-freedom equations of motion and an approach to their solution. A computer program is

62

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

developed and used to calculate trajectories for the M-117 bomb as carried on the inboard pylon of the F-4E. Comparisons of these trajectory profiles with wind tunnel captive-trajectory test results and existing theory are made. A user's guide and computer listing, excluding those subroutines due to F. Dan Fernandes, are given. Recommendations are made for possible improvements to the newly developed method.

A SUBSONIC METHOD FOR PREDICTING THE
SEPARATION TRAJECTORIES OF AIRCRAFT
STORES IN THE PITCH PLANE

Thesis

GAE/AE/74S-4 Joseph M. Manter

Approved for public release; distribution unlimited.

ib

GAE/AE/74S-4

A SUBSONIC METHOD FOR PREDICTING THE SEPARATION TRAJECTORIES OF AIRCRAFT STORES IN THE PITCH PLANE

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by

Joseph M. Manter, B.S.A.A.E.
Graduate Aeronautical Engineering

September 1974

Approved for public release; distribution unlimited.

ic

Preface

My main objective in this study was to develop a subsonic three-degree-of-freedom store separation prediction method which could incorporate, when available, experimentally determined, free-air, store static stability data. I am very satisfied with the results I have obtained with this method and I am hopeful that this longitudinal analysis of the separation problem will serve as a precursor to a six-degree-of-freedom analysis.

I would like to express my sincere gratitude for the assistance I received during the course of this study. I especially would like to thank my independent study advisor, Dr. Milton Franke, for his advice and guidance. I would also like to acknowledge my other two faculty advisors, Major Carl Stolberg and Captain James Karam for their constructive criticism. I owe a special debt of gratitude to Mr. Cal Dyer and Mr. Jerry Jenkins, both of AFFDL/FGC, for introducing me to the store separation problem and assisting me in the computation and analysis of theoretical data from existing methods. Finally, I would like to express my deepest appreciation to my wife, Ruth, for her patience and understanding, and to my daughters Jenny Lynn and Jill Kathleen for the time I took from them during these past twelve months.

Joseph M. Manter

Contents

Preface ii

List of Figures v

List of Tables vi

List of Symbols vii

Abstract x

I. Introduction 1

Problem 1
Background 1
Objective 2
Approach 3

II. Discussion of Loads on the Store 5

Interference Loads 5
Free-Air Loads 6
Ejector Loads 7
Pitching Moment Due to Damping 7
Total Loads on the Store 8

III. Integration of Forces and Moments 10

Equations of Motion 10
Integration Techniques 13

IV. Computer Program 14

Computer Memory and Execution Time Requirements 14
Overview of Computer Program 14

V. Results and Discussions 16

Simulated M-117 Trajectories 16
Comparison of Store Trajectories 18
Comparison of Computer Execution Time . . . 25

VI. Conclusions and Recommendations 30

Conclusions 30
Recommendations 31

Bibliography 33

Appendix A: Computational Modes Available in the Computer Program 35

Appendix B: User's Guide to the Computer Program . . 37

Appendix C: Computer Program Listing 42

Vita 58

List of Figures

Figure		Page
1	Forces and Moments on the Store in the Store Axis System	9
2	Inertial Coordinate System Used for Trajector Calculations (taken from Ref 7)	11
3	Sketch of the M-117 All-Purpose Bomb . .	17
4	Ejector-Force Function for the M-117 Bomb	19
5	Simulated Separation Characteristics of the M-117 Bomb	20
6	Simulated Separation Characteristics of the M-117 Bomb	21
7	Simulated Separation Characteristics of the M-117 Bomb	22
8	Simulated Separation Characteristics of the M-117 Bomb	23
9	Comparison of Loads on the M-117 Bomb for the Low Speed Case	26
10	Comparison of Loads on the M-117 Bomb for the High Speed Case	27

List of Tables

Table		Page
I	Full-Scale M-117 Parameters	16
II	Comparison of CDC 6600 Computer Execution Time	28

List of Symbols

Symbol	Description
b	Store reference dimension, ft
C_A	Store axial force coefficient, axial force/$\frac{1}{2}\rho_\infty V^2_{store} S_{ref}$
C_m	Store pitching moment coefficient, referenced to store center of gravity, pitching moment/$\frac{1}{2}\rho_\infty V^2_{store} S_{ref} b$
$C_{m_{DAMP}}$	Store pitching moment coefficient due to aerodynamic damping
C_{mq}	Store pitch damping derivative, $dC_m/d(qb/2\ V_{store})$
C_N	Store normal force coefficient, normal force/$\frac{1}{2}\rho_\infty V^2_{store} S_{ref}$
F_A	Sum of forces acting on store in axial direction, lb
$F_{AERO,A}$	Sum of aerodynamic forces acting on store in axial direction, lb
$F_{AERO,N}$	Sum of aerodynamic forces acting on store in normal direction, lb
F_{fa}	Free-air force acting on store in normal direction, lb
F_{int}	Interference force acting on store in normal direction, lb
F_N	Sum of forces acting on store in normal direction, lb
F_{z1}, F_{z2}	Ejector force due to ejectors one and two, respectively, lb
F_ξ, F_η	Sum of forces acting on store in ξ and η directions, respectively, lb

Symbol	Description
g	Acceleration due to gravity, ft/sec/sec
I_{yy}	Store moment of inertia about store y-axis, slug-ft^2
k	Store radius of gyration about store y-axis, ft
ℓ_{x1}, ℓ_{x2}	Ejector one and ejector two piston location distances relative to store center of gravity position forward of store center of gravity, ft
M_{AERO}	Pitching moment acting on store taken about store center of gravity, positive nose up, due to static aerodynamic forces, ft-lb
M_{DAMP}	Pitching moment acting on store taken about store center of gravity, positive nose up, due to aerodynamic damping, ft-lb
m_s	Store mass, slugs
M_θ	Sum of pitching moments on store taken about store center of gravity, positive nose up, ft-lb
q	Store angular velocity about store y-axis, rad/sec
S_{ref}	Store reference area, ft^2
V_{store}	Store total velocity, ft/sec
V_∞	Free stream velocity, ft/sec
X	x-separation distance of store center of gravity in fuselage wind-axis system, measured relative to store carriage position, ft
X_{CG}	Store center of gravity position, measured from nose of store, ft
X_{L1}, X_{L2}	Ejector piston location of ejectors one and two, respectively, positive forward of store center of gravity, ft
X_S, Y_S, Z_S	Store coordinate axes, ft

Symbol	Description
Z	z-separation distance of store center of gravity in fuselage wind-axis system, measured relative to store carriage position, ft
$Z_{E_{MAX}}$	Maximum ejector stroke travel, ft
α_B	Aircraft angle of attack, degrees
γ_B	Aircraft flight path angle, degrees
γ_S	Store flight path angle, degrees
θ	Store pitch angle, positive nose up, degrees
$\ddot{\theta}$	Store pitching acceleration, $d^2\theta/dt^2$, rad/sec/sec
ξ,η	Coordinates of store center of gravity in fuselage coordinate system, see Fig. 2, ft
$\ddot{\xi},\ddot{\eta}$	Accelerations of store center of gravity, relative to fuselage, ft/sec/sec
ρ_∞	Air density at simulated altitude, slugs/ft^3

Abstract

A subsonic three-degree-of-freedom method for predicting store separation trajectories is presented. The method combines the calculation of interference loading on aircraft stores due to F. Dan Fernandes with loads due to free air, ejector, and aerodynamic damping. Discussion of these loads is given, along with a presentation of the three-degree-of-freedom equations of motion and an approach to their solution. A computer program is developed and used to calculate trajectories for the M-117 bomb as carried on the inboard pylon of the F-4E. Comparisons of these trajectory profiles with wind tunnel captive-trajectory test results and existing theory are made. A user's guide and computer listing, excluding those subroutines due to F. Dan Fernandes, are given. Recommendations are made for possible improvements to the newly developed method.

A SUBSONIC METHOD FOR PREDICTING THE SEPARATION TRAJECTORIES OF AIRCRAFT STORES IN THE PITCH PLANE

I. Introduction

Problem

Today's aircraft are required to carry, and subsequently to release, many types of stores under many different flight conditions. The first 0.5 to 1.0 seconds is usually the critical period in determining a successful ejection under a given flight condition. An analytical method of predicting store separation trajectories during this critical time period is needed to serve as a preliminary design tool and to augment existing prediction methods.

Background

Store separation has been a matter of concern since the first World War I pilot attempted to throw a projectile from his open-canopied aircraft onto his enemy below. However, because of the relatively slow speeds of early flight, clearance from parent aircraft posed no great threat until the advent of the jet. At this time in aerial history, pilots were required to release externally carried stores from aircraft flying at higher and higher speeds (Ref 12:1,2).

Prediction techniques which have been developed are generally divided into three categories: full-scale flight testing, wind tunnel, and theoretical methods. Full-scale

flight testing has an obvious drawback in high costs, but, more importantly, can be quite hazardous to the pilot and his aircraft. Wind tunnel methods, including the simulated trajectory (Ref 2) and grid techniques (Ref 1), have proved to provide satisfactory results. However, because the only Air Force tunnel equipped for these techniques is scheduled many months in advance and because the already expensive costs of operating any wind tunnel is increasing, these methods cannot always be applied to present-day problems.

Theoretical methods have only recently been considered as an alternative to the first two approaches. While many authors have developed analytical methods of calculating flow fields under an aircraft (Ref 12:5-19), Goodwin, Nielsen, and Dillenius (Ref 7) and Goodwin, Dillenius, and Nielsen (Ref 8) have applied these analytical methods in developing the most comprehensive technique for computing separation trajectories. Fernandes (Refs 3 and 4), has devised methods for determining the interference loading on an aircraft store in the flow field of a parent aircraft. His development is incorporated in the present work.

Objective

The objective of this study is to develop a three-degree-of-freedom method to predict the separation trajectory of an aircraft store upon release from a parent aircraft flying at subsonic speeds. While previously developed techniques (Refs 7 and 8) have been entirely analytical in nature,

a goal of this study is to permit the use of experimental static stability data, when available, to determine the free-air loads on the store.

Another intent of this study is to allow the simulation and, therefore, the integrated effects, of a two-point ejector system in calculating the trajectory of an aircraft store. This again contrasts previously developed techniques which do not simulate the ejector, but account for it in initial conditions which must be calculated before trajectory calculations can begin.

A shortcoming of previously developed methods is the large computer time necessary to simulate the separation trajectories of aircraft stores. A third purpose for undertaking this study is to write a FORTRAN computer program which will require reasonable computer execution time to simulate the trajectories of ejected stores.

Finally, it is expected that the three-degree-of-freedom store separation prediction method will serve as a precursor to a six-degree-of-freedom method.

Approach

The method used to calculate separation trajectories is straightforward. Interference normal forces and pitching moment coefficients on the ejected store are determined using portions of Fernandes' subsonic interference loading program (Ref 3). Predetermined free-air normal force and pitching moment coefficients, obtained either by experiment

or any suitable theory, are then added to the interference loads. This sum, along with a predetermined axial force coefficient which remains constant throughout the trajectory, constitutes the total aerodynamic loading on the store. Next, normal force and pitching moment coefficients due to the store ejector system, when operating, are determined. Finally the pitching moment coefficient due to damping is calculated using a predetermined, from experiment or theory, pitch damping coefficient. All force and moment coefficients are then summed and a new position is determined by integrating the three-degree-of-freedom equations of motion. The process is repeated until clearance from the parent aircraft is assured.

II. Discussion of Loads on the Store

Interference Loads

Interference pitching moment and normal force coefficients are calculated using portions of the F. Dan Fernandes subsonic interference loading program (Ref 3). The calculation of the interference flow field by this method entails a linear theory of source, vortex, and doublet distributions. The aircraft wing, pylon, inlet and fuselage nose are modeled using various combinations of these distributions, the strengths of which are calculated by satisfying boundary conditions (no flow normal to the body surface) at various control points on the body. Disturbance velocities are then calculated over the length of the store as functions of these strengths and distances to the field point under consideration. From these disturbance velocities an interference angle-of-attack field can be determined. Interference forces and moments are then calculated by integrating the effects of this variable interference angle-of-attack field over the length of the store using predetermined (by experiment or theory provided by Fernandes) free-air body loading coefficients per unit angle-of-attack per unit length. Concurrently, the interference static pressure field over the length of the store is calculated and then integrated to yield the loading due to buoyancy on the store (Ref 3:5, E-1). Compressibility effects are included by applying the Prandtl-Glauert rule (Ref 3:A-1). Details on the interference

loading method are given by Fernandes (Ref 3).

Although Fernandes' program calculates five interference force and moment coefficients (normal force, side force, pitching moment, rolling moment, and yawing moment coefficients), the present method uses only two (pitching moment and normal force coefficients) because it is a three-degree-of-freedom analysis only. Interference axial force coefficient is not calculated by the Fernandes method and so this interference force is not accounted for in the present method.

Free-Air Loads

The calculation of total static aerodynamic loads on the store is completed with the addition of free-air loads to those interference loads determined by the Fernandes method. Free-air normal force and pitching moment coefficients are not calculated by the present method and, therefore, must be obtained before using it, usually as functions of store angle of attack and store Mach number. The free-air axial force coefficient must also be obtained before using the present method and is assumed constant throughout the trajectory simulation. Data may be obtained from experimental sources (as was done for the sample case considered in this report--see Chapter V) or from any suitable analytical method, such as USAF DATCOM (Ref 10, Sec 4).

Ejector Loads

Calculation of ejector normal forces and pitching moments is quite similar to the method used by Christopher and Carleton (Ref 2:32-33). Modifications were necessary to allow ejector force curves either as a function of ejector foot distance or as a function of time. In addition calculation of ejector normal force and pitching moment coefficients are made to permit comparisons of those ejector coefficients with corresponding interference and free-air coefficients.

The present method requires the ejector force-distance or force-time curves to be in the form of a fifth degree polynomial, the same form used by Christopher and Carleton (Ref 2:32). Two such curves are allowed to accommodate a two-point ejector system. In addition, the distance away from the center-of-gravity of the store at which each ejector is assumed to act must be known to calculate ejector moments on the store.

Pitching Moment Due to Damping

To account for the change in pitching moment due to aerodynamic damping, a pitch damping coefficient, required by the present method, must be known. According to Christopher and Carleton, the consideration of pitching moment due to aerodynamic damping is usually a second order effect, so any reasonable estimate for pitch damping coefficient, C_{mq}, should satisfy the requirements of the present method. Should the store exhibit large oscillatory motions,

however, this effect could become significant in which case consideration should be given to obtaining experimental values for $C_{m\dot{q}}$ or running trajectory sensitivity studies varying possible values for C_{mq} in order to determine its significance on the trajectory of the store under study (Ref 2:9).

The actual calculation of pitching moment coefficient due to aerodynamic damping is a simple one once C_{mq}, $dC_m/d(qb/2\ V_{store})$, is known:

$$C_{m_{DAMP}} = C_{mq} \cdot qb/2\ V_{store} \tag{1}$$

where

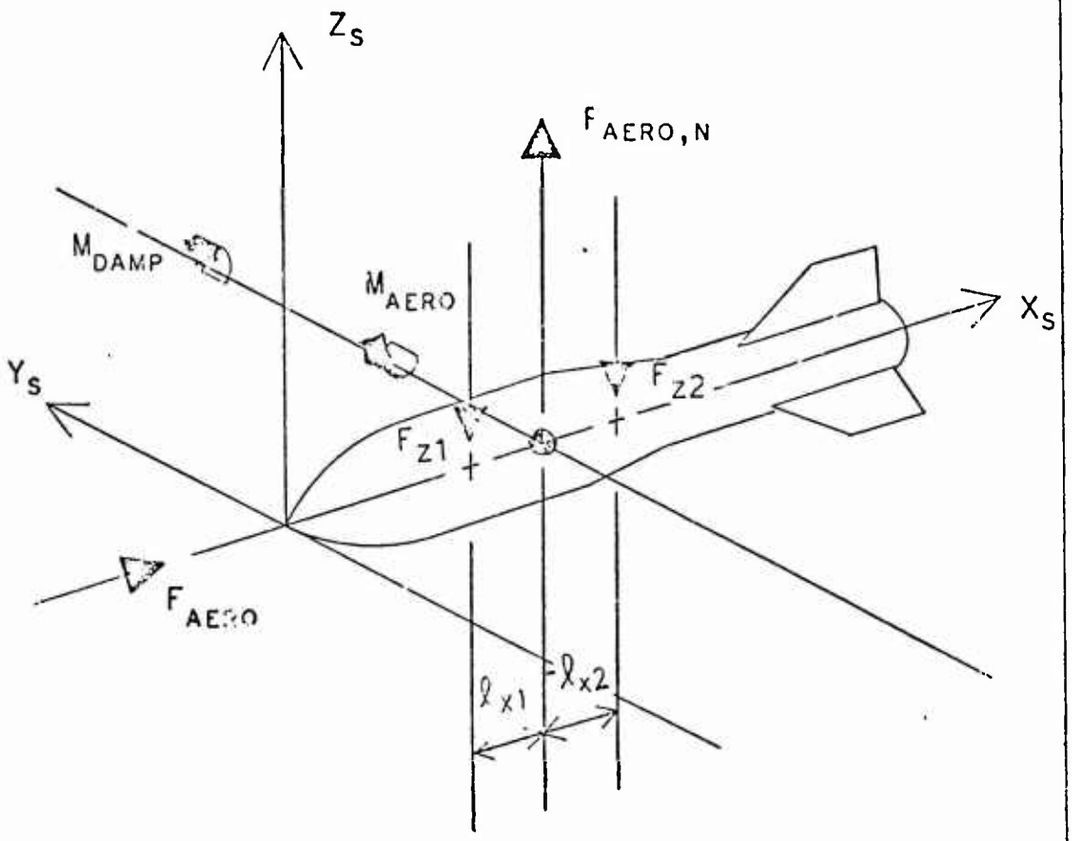
q = store pitch rate

b = store reference length

V_{store} = absolute store total velocity.

Total Loads on the Store

The total normal force load on the store is given by the sum of the loads due to interference, free-air, and ejector systems. The total axial force load on the store is computed using the predetermined axial force coefficient. The total pitching moment on the store is simply the sum of the moments due to interference, free-air, ejector system, and aerodynamic damping. These loads are illustrated in Fig. 1, in the coordinate system fixed in the ejected store.



WHERE

$F_{AERO,N}$ = sum of interference and free-air aerodynamic forces in the normal direction

$F_{AERO,A}$ = free-air aerodynamic force in the axial direction

F_{z1} = force due to ejector one

F_{z2} = force due to ejector two

M_{AERO} = pitching moment due to interference and free-air aerodynamic loads

M_{DAMP} = pitching moment due to damping

Fig. 1. Forces and Moments on the Store in the Store Axis System.

III. Integration of Forces and Moments

Equations of Motion

The form of the three-degree-of-freedom equations of motion presented in this chapter was taken from Goodwin, Nielsen, and Dillenius (Ref 7:78-84). In place of the Goodwin method for calculating forces and moments on the store, however, the calculation of forces and moments by the present method, including free-air loading, interference loading, aerodynamic damping, and ejector loading, is used.

The inertial frame (ξ,η) for this treatment is fixed with the aircraft, hence the flight conditions must be non-accelerating. The aircraft, therefore, is assumed to fly at a constant flight path angle, γ_B, free stream velocity, V_∞, and angle of attack, α_B. Figure 2 illustrates this inertial coordinate system and the initial orientation of the store in it.

The longitudinal equations of motion in this coordinate system are:

$$m_s\ddot{\xi} = F_\xi \tag{2}$$

$$m_s\ddot{\eta} = F_\eta \tag{3}$$

$$m_s k^2\ddot{\theta} = M_\theta \tag{4}$$

Now the forces on the store in the ξ-direction and η-direction, F_ξ and F_η, can be found by properly resolving

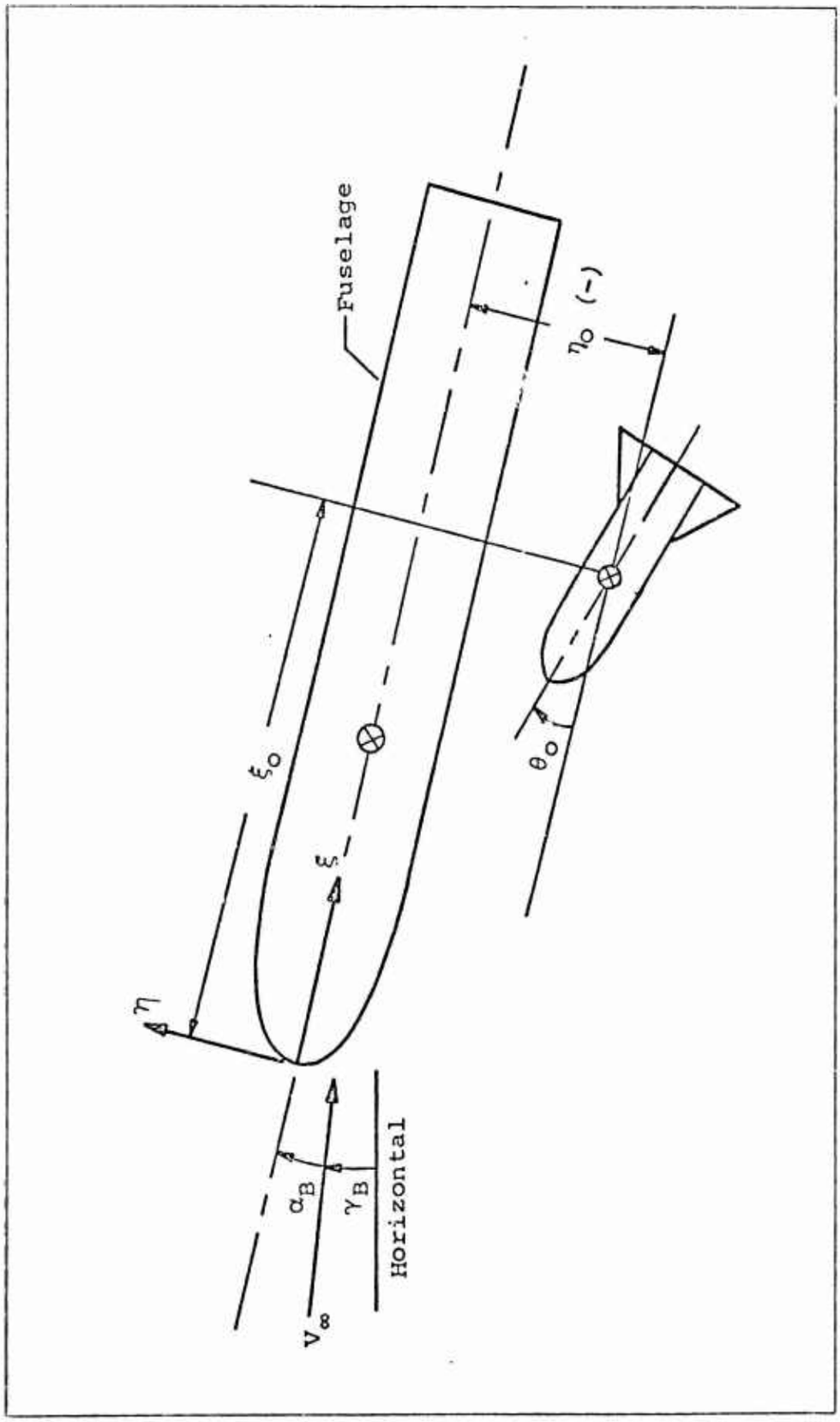


Fig. 2. Inertial Coordinate System Used for Trajector Calculations (taken from Ref 7).

forces along the store axis system into the inertial frame. The force normal to the store is simply

$$F_N = F_{int} + F_{fa} + F_{z1} + F_{z2} \tag{5}$$

The axial force on the store is given by

$$F_A = C_A \frac{1}{2} \rho_\infty V^2_{store} S_{ref} \tag{6}$$

The resolved forces are, then

$$F_\xi = F_N \sin\theta + F_A \cos(\alpha_B+\gamma_B-\gamma_S) + m_s g \sin(\alpha_B+\gamma_B) \tag{7}$$

$$F_\eta = F_N \cos\theta + F_A \sin(\alpha_B+\gamma_B-\gamma_S) - m_s g \cos(\alpha_B+\gamma_B) \tag{8}$$

where γ_s is the flight path angle of the store, which is different from γ_B if wind tunnel captive trajectory tests are to be simulated (Ref 7:84-85).

Noting that $C_N = F_N/(\frac{1}{2} \rho_\infty V^2_{store} S_{ref})$ and substituting Eqs (7) and (8) into Eqs (2) and (3), the first two equations of motion become

$$\ddot{\xi} = \frac{1}{2} \rho_\infty V^2_{store} (S_{ref}/m_s)[C_N \sin\theta + C_A \cos(\alpha_B+\gamma_B-\gamma_S)] + g \sin(\alpha_B+\gamma_B) \tag{9}$$

$$\ddot{\eta} = \frac{1}{2} \rho_\infty V^2_{store} (S_{ref}/m_s)[C_N \cos\theta + C_A \sin(\alpha_B+\gamma_B-\gamma_S)] - g \cos(\alpha_B+\gamma_B) \tag{10}$$

Substitution of $C_m = M_\theta/(\frac{1}{2} \rho_\infty V^2_{store} S_{ref} b)$ into Eq (9) yields the third longitudinal equation of motion

$$\ddot{\theta} = \frac{1}{2} \rho_\infty V^2_{store} (S_{ref} b / m_s k^2) C_m \qquad (11)$$

where C_m is the sum of the interference, free-air, ejector one, ejector two, and pitch damping moment coefficients. Equations (9), (10), and (11) are, except for slight changes in notation, exactly those equations derived by Goodwin, Nielsen, and Dillenius (Ref 7:79-80).

Integration Techniques

Equations (9), (10), and (11) must be integrated in order to calculate new store positions as well as angular and linear velocities and accelerations of the store at each time of interest. The method used for integration is a standard fourth-order predictor-corrector technique, utilizing a Runge-Kutta scheme to calculate intermediate steps, and was taken from Goodwin, et al. The method is introduced in their three-degree-of-freedom report (Ref 6:26) and explained in detail in their six-degree-of-freedom report (Ref 9:139-142, 219-221).

Actual integration of the equations of motion may be started at any desired time, provided the correct initial conditions, store position and linear and angular velocities, are known. The forces and moments are calculated using methods outlined in Chapter II, then a new position is found by integrating Eqs (9), (10), and (11) into which have been substituted the calculated values for C_m and C_N. The process is then repeated until clearance is assured.

IV. Computer Program

Computer Memory and Execution Time Requirements

A FORTRAN computer program was developed for the present method. It is operational on the CDC 6600 computer with the SCOPE 3.4 operating system and library tape and requires about 60000 octal storage registers for loading. Execution time varies greatly, depending on how much detail is taken in describing the geometry of the parent aircraft, how many store sections are considered for computing interference loading, the time step size chosen for integration of the equations of motion, and the computational mode desired for this integration. Execution time for the trajectory simulations of the M-117 bomb from an F-4E are listed in Chapter V and discussion of computational modes in Appendix A.

Overview of Computer Program

The bulk of the computer program consists of twenty-six subroutines taken from Fernandes (Ref 5), all necessary in computing the interference loading on the aircraft stores. In addition, the predictor-corrector subroutine necessary for integration of the equations of motion, subroutine ADAMS, was taken from Goodwin (Ref 6).

The user of the computer program must supply his own subroutine, FREAIR, whose purpose is to calculate the free-air pitching moment and normal force coefficients on the store, given the store angle of attack and Mach number.

The main program whose primary purpose is to organize calculations by calling necessary subroutines, calls two other subprograms, TRREAD and TRAJEC. The purpose of subroutine TRREAD is to read in values of store geometry, store mass characteristics, ejector characteristics, pitch damping coefficient, axial force coefficient, and aircraft flight conditions necessary for separation calculations. This subroutine also initiates values of the dependent variables, store position and pitch angle, and linear and angular velocities required for integration of the equations of motion.

The purpose of subroutine TRAJEC is to accept store position and interference loads, calculate free-air loads, ejector loads, and pitch damping, sum all loads and output them. It then integrates the equations of motion and returns a new store position and orientation to the main program.

A user's guide for the computer program is presented in Appendix B and a listing of the program, with the exception of those 26 subroutines taken from Fernandes (Ref 5), is provided in Appendix C. A few minor changes to those Fernandes subroutines, necessary to pass interference coefficients to the main program and to avoid extraneous output, are also listed in Appendix C. The computer codes for the Fernandes subroutines themselves are available through COSMIC (Computer Software Management and Information Center). Requests should be directed to: COSMIC, University of Georgia 30601.

V. Results and Discussions

Simulated M-117 Trajectories

Sample trajectories were calculated using the present method and the Goodwin method (Ref 8) for the M-117 all-purpose bomb ejected from the F-4E aircraft on the 81.50 (inboard) pylon. A sketch of the M-117 bomb is shown in Fig. 3 and its aerodynamic, mass, and geometric characteristics used in theoretical calculations are listed in Table I (Ref 10:25,100).

Table I

Full-Scale M-117 Parameters

Parameter Name	Parameter Value
m_s, slugs	25.45
X_{CG}, feet	2.333
X_L, feet	-0.3417
$Z_{E_{max}}$, feet	0.275
S_{ref}, sq ft	1.395
b, ft	1.333
C_A	0.1
I_{YY}, slugs-sq ft	50.0
C_{mq}, per rad	-70.0

The M-117 bomb was chosen for a sample calculation because experimental data on free-air stability characteristics (Ref 13:25) were available and, under certain flight

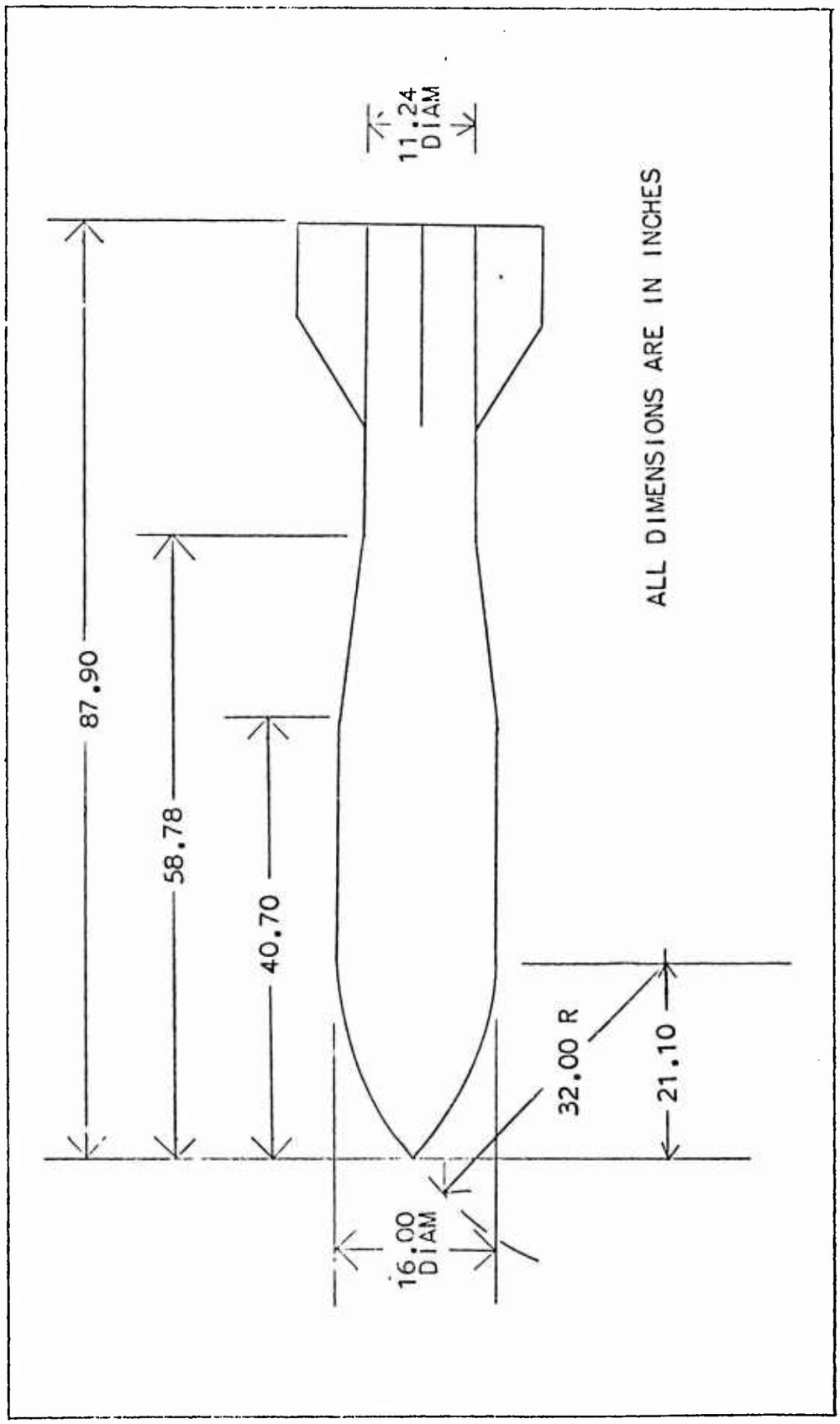


Fig. 3. Sketch of the M-117 All-Purpose Bomb.

conditions and ejector loads, the M-117 exhibited small lateral (yaw angle and y-direction) excursions.

Experimental trajectories were taken from an AEDC wind tunnel report whose author investigated the effects of wing leading-edge slats on the separation characteristics of various stores as carried on the F-4E (Ref 11). That same wind tunnel investigation modeled separation trajectories from the F-4E aircraft without leading edge modifications and it is those baseline trajectories to which the theoretical results are compared.

The ejector force-distance curve used in the present method was taken from the same AEDC wind tunnel report and is shown in Fig. 4.

Comparison of Store Trajectories

Figures 5, 6, 7 and 8 exhibit X and Z coordinates, relative to carriage position, of the center of gravity of the store, as well as the absolute pitch angle of the store as seen in the fuselage wind-axis system. Results from experiment and both the present method as well as the method due to Dillenius, et al. (Ref 8) are shown.

As illustrated in Fig. 5 both methods predicted the experimental results remarkably well for the low speed, moderate angle-of-attack case (Mach = 0.332, α = 7.40), with the present method doing slightly better in position prediction and the previously developed method predicting pitch angle excursions more accurately. In Fig. 6, however, while linear

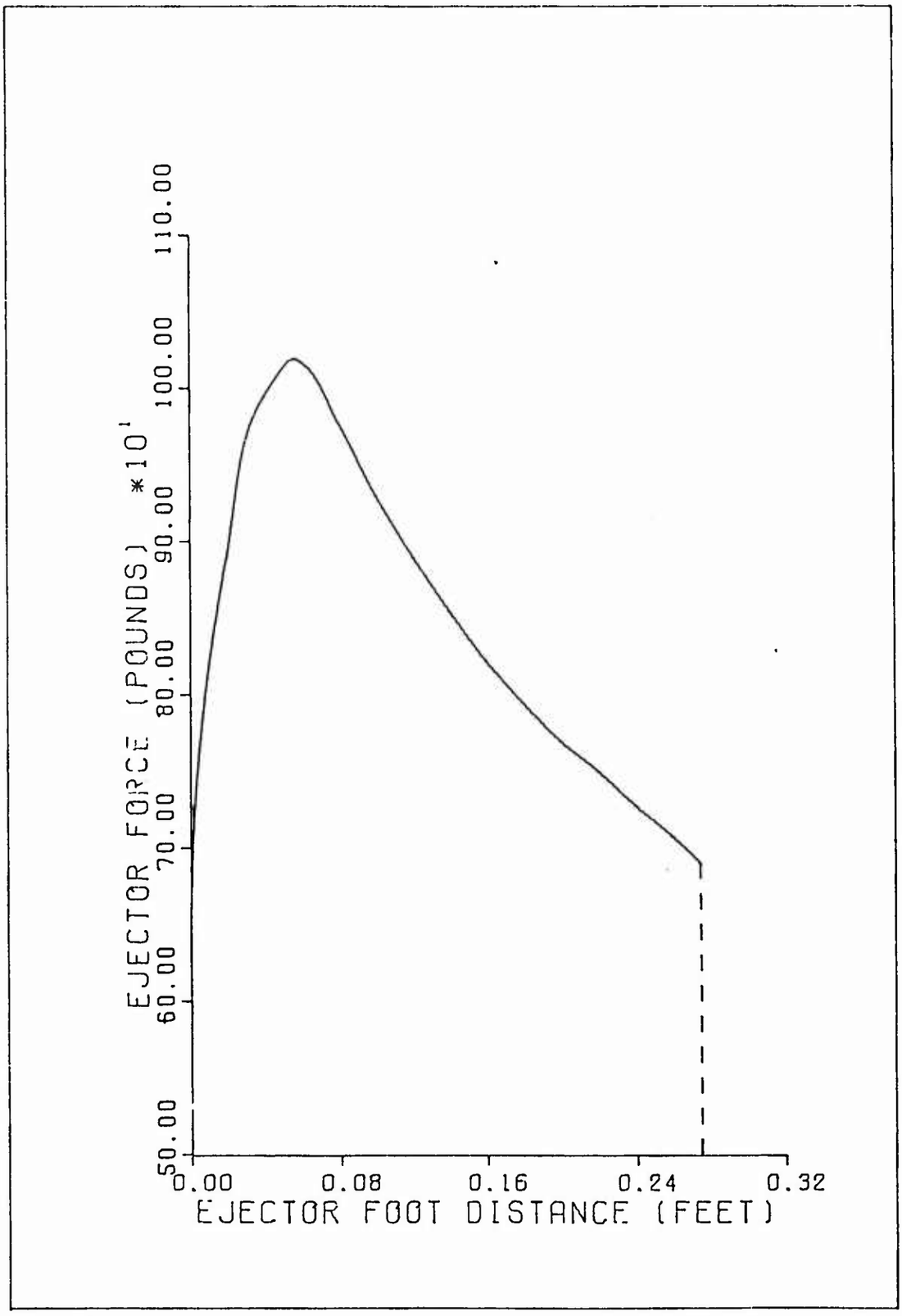


Fig. 4. Ejector-Force Function for the M-117 Bomb.

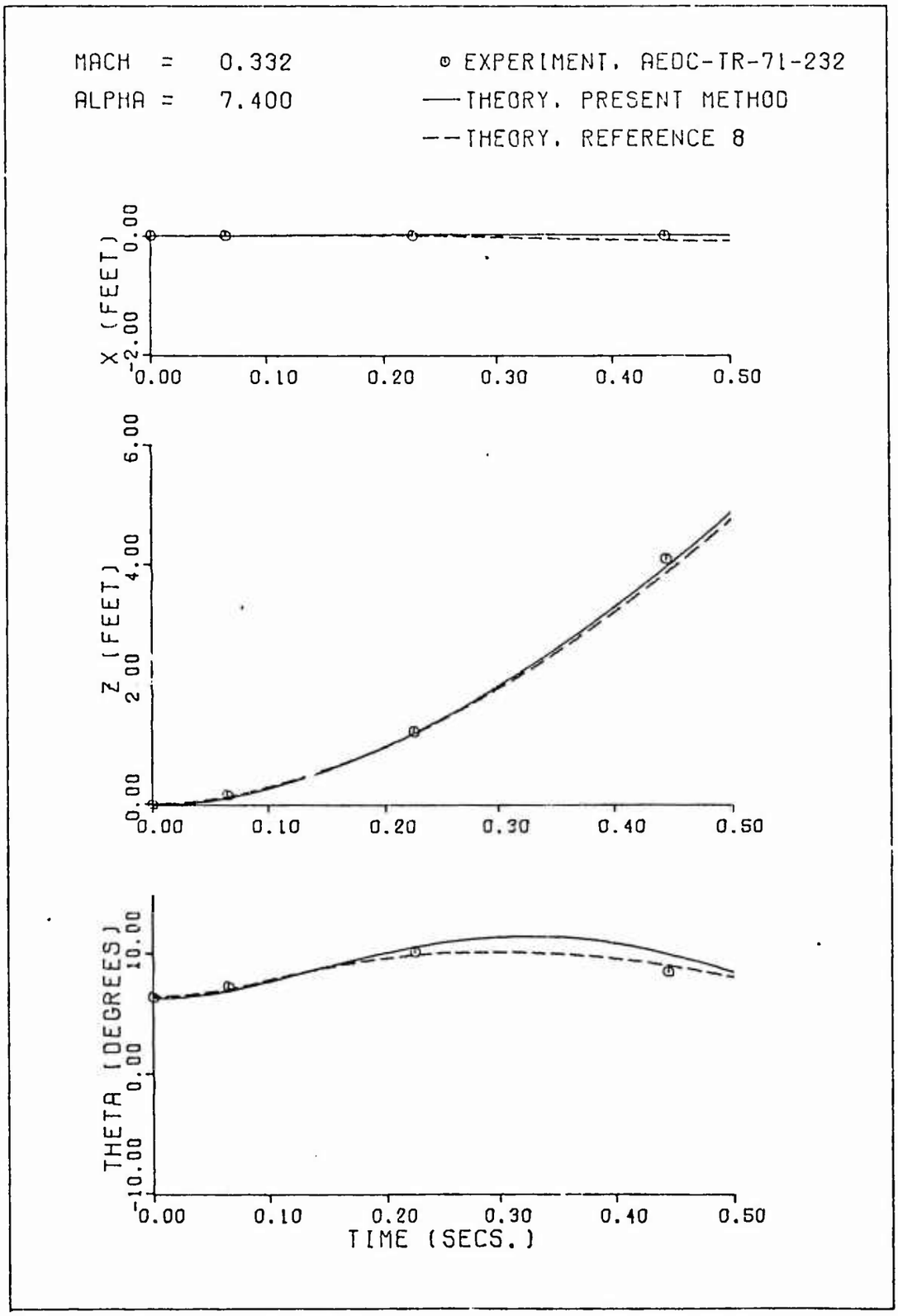


Fig. 5. Simulated Separation Characteristics of the M-117 Bomb.

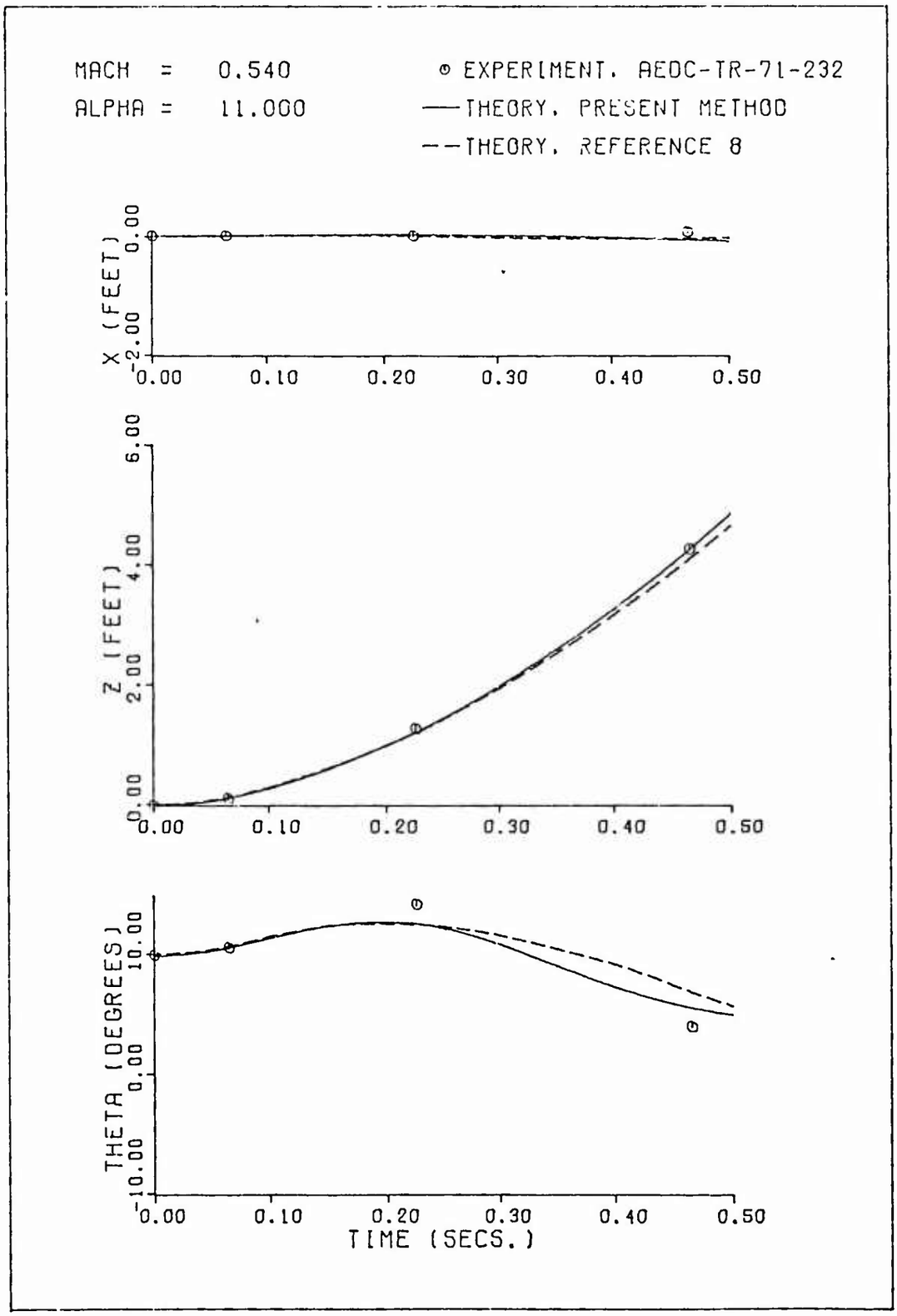


Fig. 6. Simulated Separation Characteristics of the M-117 Bomb.

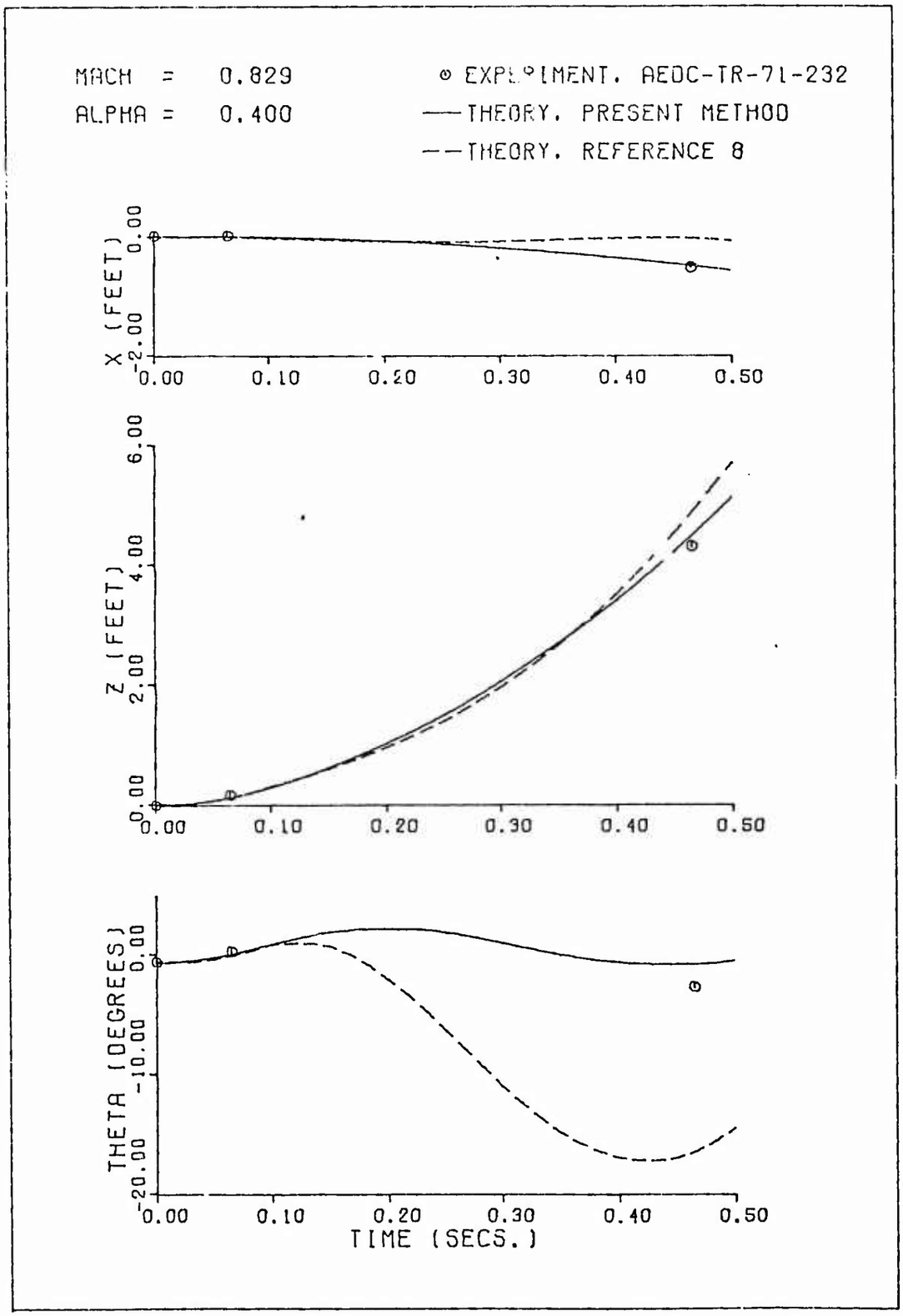


Fig. 7. Simulated Separation Characteristics of the M-117 Bomb.

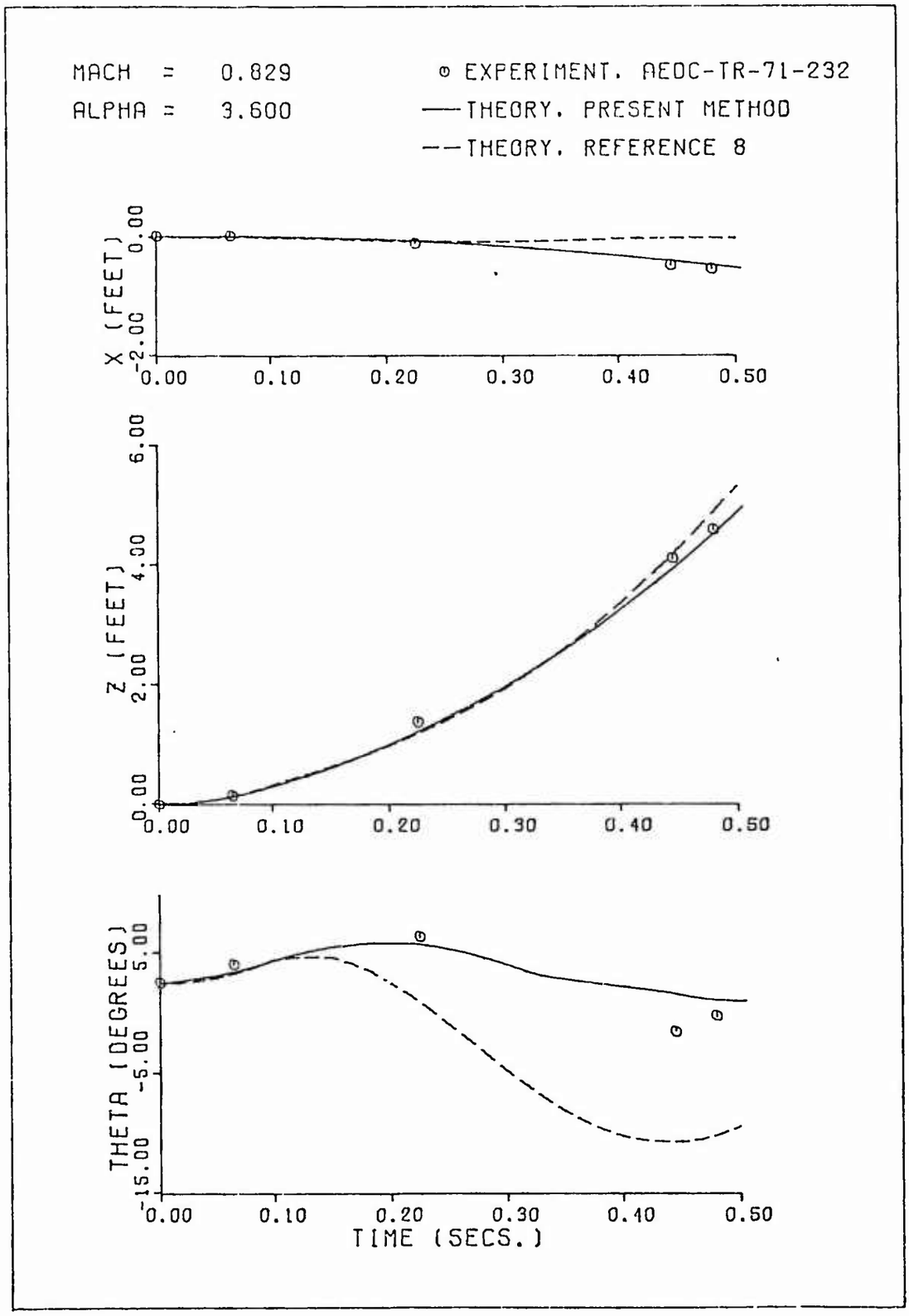


Fig. 8. Simulated Separation Characteristics of the M-117 Bomb.

distance vs. time curves are again estimated quite well, the pitch angle calculation is not as close to experiment as in the previous case. It is difficult to conjecture which method predicts pitch angle better, since both indicate the damped sinusoidal pitch angle profile common to stable stores, but the trajectory due to the present method has a shorter period of pitch oscillation than does the trajectory due to the Goodwin method (Ref 8).

For the high speed case (Mach = 0.829), Figs. 7 and 8 show interesting results. While the present method again predicts linear travel quite well and pitching angle reasonably well, the method obtained from Ref 8 shows a definite deviation from experiment slightly after ejector time cut-off (about 0.08 seconds). A Mach number equal to 0.829 results, more than likely, in a supercritical flow-field situation, thus violating a basic assumption of the method (Ref 8), that conditions modeled are to be at subcritical speeds only, thereby rendering the results calculated at the high speeds invalid.

The application of the present method to relatively high Mach numbers (0.829) might also be construed as a violation of the subcritical speed restriction placed on the Fernandes method. While technically this is true, there is no such restriction on the other sources of store loading, so the present method can be pushed past the normal critical speed (about Mach = 0.7 to 0.8) with relative safety provided the free-air loading is known with some confidence.

Reasons for the excellent agreement of the present method with experiment are many, one of which is certainly the acceptable results the Fernandes method for calculating interference forces and moments seems to produce (Ref 3:13-21). Good results for total aerodynamic loading is assured by the fact that the experimental free-air data was available to add to the analytically predicted interference loads.

Store loading due to ejector forces and due to aerodynamic damping are modeled in the present work exactly as they are in the AEDC tunnel tests and as a result should introduce no error into calculations of store trajectories. In the case of the ejector forces at low speeds, this effect can be quite significant because, when operating, the ejector forces tend to overwhelm the aerodynamic forces. Figure 9 illustrates this point for the low speed (Mach = 0.332) case. At higher aircraft speeds, however, the aerodynamic forces are greater and do not, in general, dominate the ejector forces, as indicated in Fig. 10 for the Mach = 0.829 case.

A final reason for satisfactory results is the fact that the M-117 is a very dense store. Dense stores have higher inertial-to-aerodynamic-loads ratios than do lighter weight stores of the same general shape and so are not "blown around" as easily as their lighter weight counterparts.

Comparison of Computer Execution Time

The execution times required by the computer program utilizing the present method can be directly compared to that

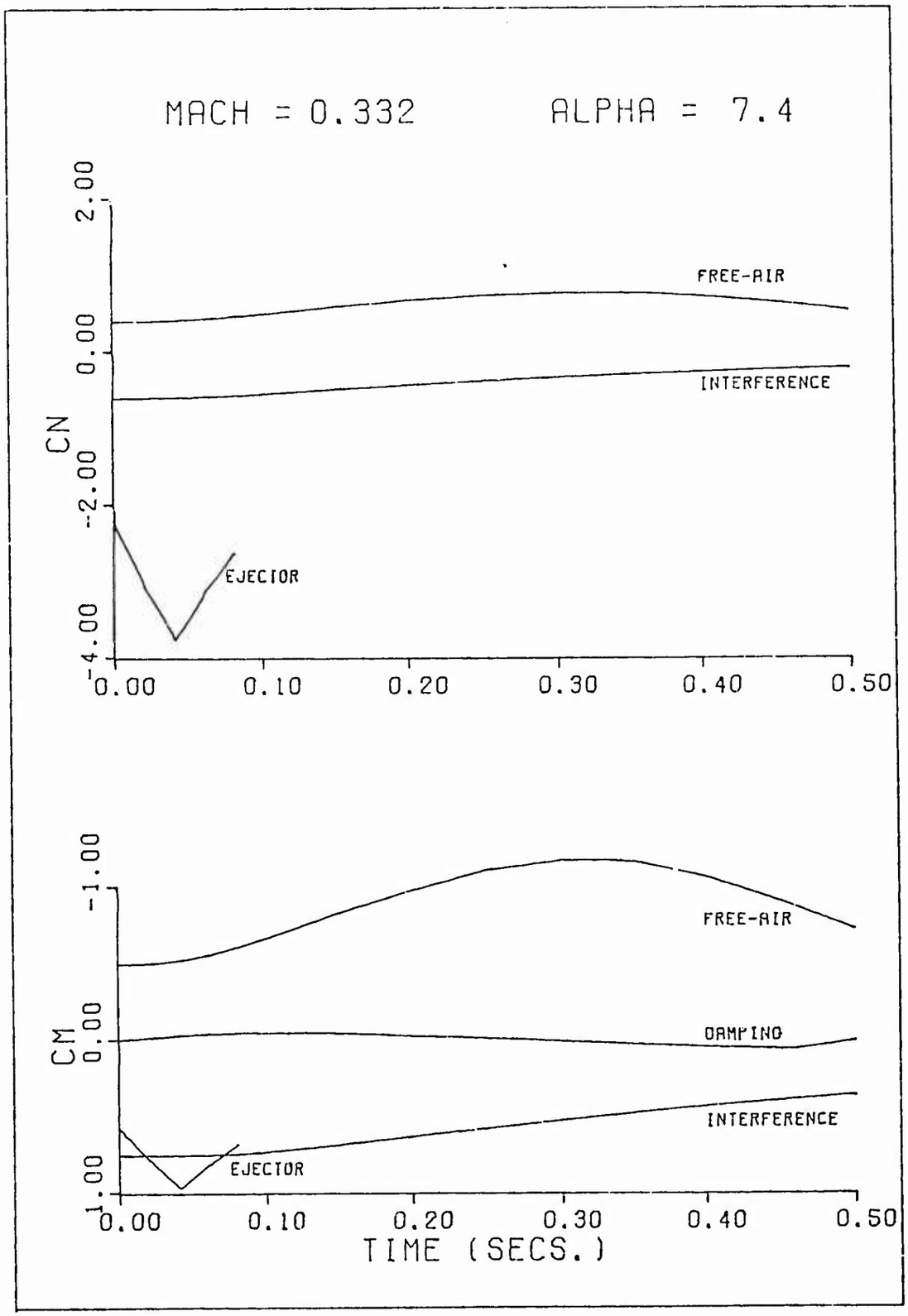


Fig. 9. Comparison of Loads on the M-117 Bomb for the Low Speed Case.

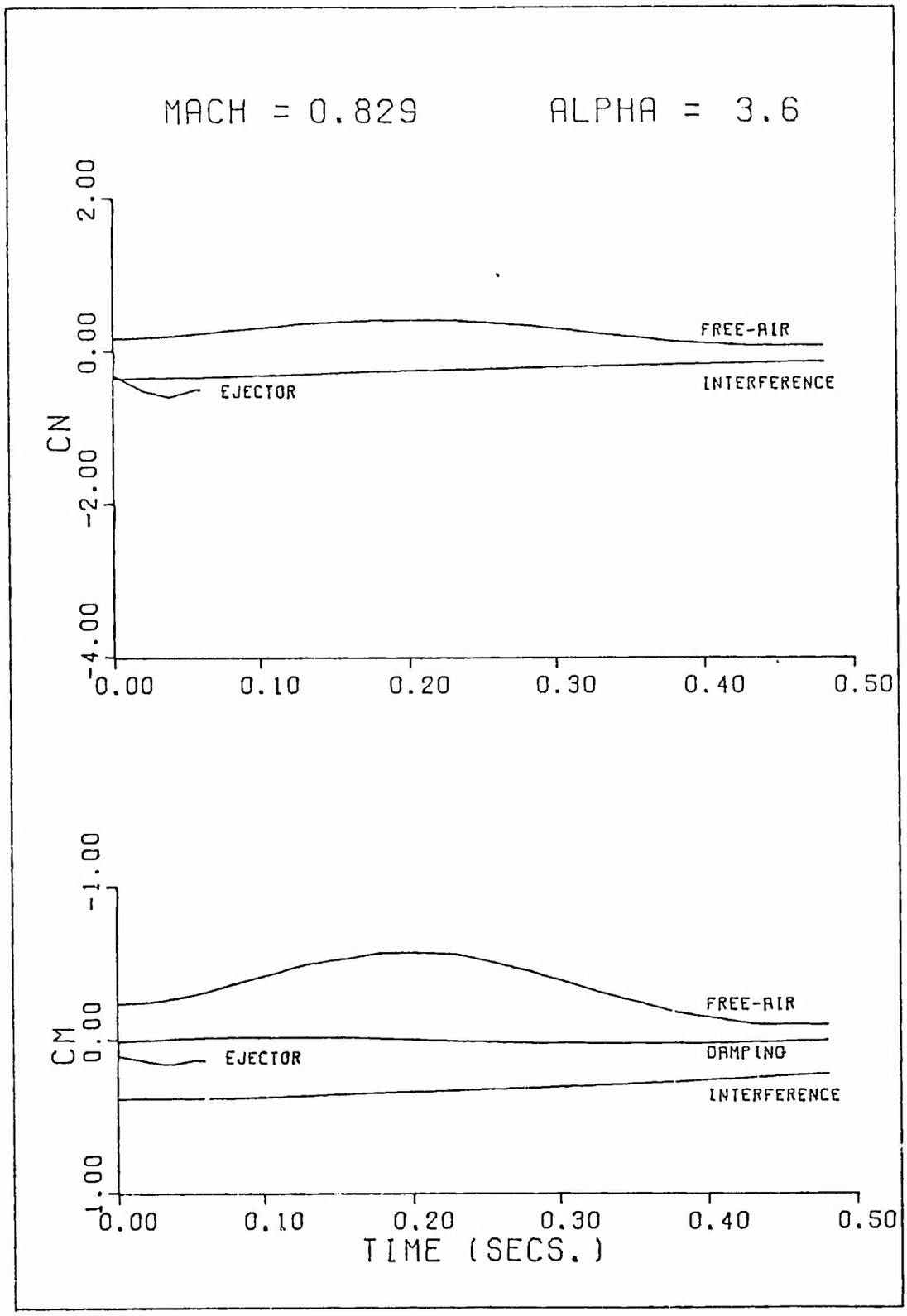


Fig. 10. Comparison of Loads on the M-117 Bomb for the High Speed Case.

of the computer program written by Goodwin and Dillenius (Ref 9). Two different execution times are listed in Table II for the present method, illustrating the two computational modes available in its corresponding computer program. These

Table II

Comparison of CDC 6600 Computer Execution Time

			Computer Execution Time (secs)		
			Present Method		Ref 8 Method
Case	Mach	Alpha	KSTABLE=0	KSTABLE=1	
1	0.332	7.4	98	30	205
2	0.540	11.0	98	67	204
3	0.829	0.4	142	46	209
4	0.829	3.6	142	43	210

computational modes, KSTABLE=0 and KSTABLE=1, yield essentially the same store trajectories and are explained in detail in Appendix A. Examination of Table II reveals that the present method yields some dramatic savings in computer execution time, while losing no noticeable accuracy in trajectory simulation at low and moderate speeds, and actually extending the speed regimes for reasonably accurate trajectory simulation to higher subsonic speeds than those allowed by the Goodwin method (Ref 8).

Both computer programs are operational on the CDC 6600 computer with the SCOPE 3.4 operating system and library

tape. The program due to Goodwin (Ref 9), which was modified to restrict movement to the pitch plane only, requires about 114000 octal storage registers for loading. The program written using the present method requires about 60000 octal store registers for loading.

VI. Conclusions and Recommendations

Conclusions

Results from the present prediction method compared well with experimental results for store trajectories which exhibit minimal lateral characteristics. Although previously developed trajectory prediction techniques (Refs 7 and 8) and the method used in the present work to calculate interference loads are limited to the subcritical speed regime, the present method may be used with some confidence past the subcritical cut-off because experimental data used to calculate free-air loads on the store is usually available for supercritical speeds. In addition, the method used to calculate the loads due to the ejector system is not limited to any aircraft speed regime, so the method used to calculate ejector loads may also be used supercritically.

A substantial improvement in computer execution time was realized with the present method over the method due to Goodwin, Dillenius, and Nielsen (Ref 8). This savings was even greater for the KSTABLE=1 mode, discussed in Appendix A, although all store trajectory simulations may not be suitable to allow the use of that time-saving feature.

Regardless of how well the present method predicts the three-degree-of-freedom trajectory of an aircraft store, its engineering practicality is limited because it is a three, not a six, degree-of-freedom analysis. Should the store trajectory under investigation not be suited to a pitch-plane

analysis, or should experimental store static stability data not be available, the six-degree-of-freedom analysis due to Goodwin, Dillenius, and Nielsen (Ref 8) should be used.

Recommendations

As alluded to above, the first obvious extension of the present method would be one to accommodate the other three degrees of freedom (side force, yawing moment, and rolling moment). While this sounds straightforward, consideration of all six degrees of freedom compounds problems immensely and should not be viewed lightly. Goodwin, Dillenius, and Nielsen (Ref 8) have completed this task using their theory for calculating store loading, so much of their work could again be utilized. However, as yet, the Fernandes method does not allow consideration of a yawed store so considerable modification to his method must be completed.

In addition, improvement to the Fernandes method itself could be attempted. Of particular interest would be to allow some consideration of wing-body interference, addition of an arbitrarily shaped representation of the fuselage, and the addition of other bodies to account for store-to-store interference. A method similar to Goodwin, et al. (Ref 7:20-22) would be the simplest to employ in order to get a first order effect for wing-fuselage and store-to-store interference. In that method, an induced camber on the wing is calculated to cancel the effect of the fuselage changing the boundary

conditions on the wing when the two independent solutions, wing and body, are superimposed.

Thirdly, the rack could be modeled, possibly as a distribution of axisymmetric sources (Ref 8:20) or, as suggested by Goodwin, et al. (Ref 8:20), a combination distribution of axisymmetric sources to account for rack thickness and a small system of vortices to account for the short wing-like stubs which normally protrude from most bomb racks.

Another area which typically gets little consideration is the elastic effect on the rack and, consequently, the net force felt by the store due to the extremely high ejector forces common to most store-ejector systems.

Finally, consideration should be made to the development of a supersonic three or six-degree-of-freedom trajectory program. Fernandes (Ref 4) has written a companion program to his subsonic interference loading program which will compute interference loads on a store in the flow field of an aircraft flying at supersonic speeds. His supersonic method could be incorporated into a new technique in a manner quite similar to that developed here.

Bibliography

1. Bamber, M. J. Store Separation Investigation by Grid Method Using Wind Tunnel Data. David Taylor Model Basin, Washington DC, April 1966. (DTMB Report 2202, AD 633 565).

2. Christopher, J. P. and W. E. Carleton. Captive-Trajectory Store-Separation System of the AEDC-PWT 4-Foot Transonic Tunnel. Arnold Engineering Development Center, Tullahoma, Tennessee, September 1968. (AEDC-TR-68-200, AD 839 743).

3. Fernandes, F. D. Theoretical Prediction of Interference Loading on Aircraft Stores; Part I - Subsonic Flow. General Dynamics, Pomona, California, June 1972. (NASA CR-112065-1).

4. Fernandes, F. D. Theoretical Prediction of Interference Loading on Aircraft Stores; Part II - Supersonic Flow. General Dynamics, Pomona, California, June 1972. (NASA CR-112065-2).

5. Fernandes, F. D. Theoretical Prediction of Interference Loading on Aircraft Stores; Part III - Programmer's Manual. General Dynamics, Pomona, California, June 1972. (NASA CR-112065-3).

6. Goodwin, F. K. Fortran Programs for Computing Three-Degree-of-Freedom Store Separation Trajectories at Speeds up to the Critical Speed. Nielsen Engineering and Research, Inc., Palo Alto, California, September 1971. (NEAR Report TR 30).

7. Goodwin, F. K., J. N. Nielsen, and M. F. E. Dillenius. A Method for Predicting Three-Degree-of-Freedom Store Separation Trajectories at Speeds up to the Critical Speed. Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, July 1971 (AFFDL-TR-71-81).

8. Goodwin, F. K., M. F. E. Dillenius, and J. N. Nielsen. Prediction of Six-Degree-of-Freedom Store Separation Trajectories at Speeds up to the Critical Speed, Vol. I - Theoretical Methods and Comparisons with Experiment. Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, 1972. (AFFDL-TR-72-83, Vol. I).

9. Goodwin, F. K. and M. F. E. Dillenius. Prediction of Six-Degree-of-Freedom Store Separation Trajectories at Speeds up to the Critical Speed, Vol. II - User's Manual for the Computer Programs. Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, October 1973. (AFFDL-TR-72-83, Vol. II).

10. Hoak, D. E., Project Engineer, USAF Stability and Control Datcom. Project No. 8219. Wright-Patterson Air Force Base, Ohio: Air Force Flight Dynamics Laboratory, October 1960 (rev. September 1970).

11. Myers, J. R. Separation Characteristics of the SUU-41, M-117, BLU-1/B, LAU-3/A, and 370-Gal Fuel Tank from the F-4E Aircraft Equipped with Wing-Leading-Edge Slats. Arnold Engineering Development Center, Tullahoma, Tennessee, October 1971. (AEDC-TR-71-232).

12 Ryan, B. M., R. E. Meeker, and L. W. Seeley. Store Separation: State-of-the-Art. Naval Weapons Center, China Lake, California, June 1972. (NWC TP 5530).

13. Summers, W. E. Aerodynamic Loads Data on the M-117 Bomb in the Flow Field of the Triple Ejection Rack at Mach Numbers from 0.5 to 1.3. Arnold Engineering Development Center, Tullahoma, Tennessee, June 1972. (AEDC-TR-72-81).

Appendix A

Computational Modes Available in the Computer Program

A standard fourth-order predictor-corrector technique was used to integrate the equations of motion. At the onset, and at each subsequent change in step size of the integration routine, Runge-Kutta calculations are made to determine intermediate values of the dependent variables. Large computer execution times were introduced because at each major and each intermediate time step all forces and moments must be calculated according to the methods outlined in Chapter II. Calculation of interference forces and moments, as might be expected, required the most amount of time, on the order of two seconds per store position for a typical fighter-bomber. To alleviate this problem a parameter, KSTABLE, may be input with the value of 1. This will suppress the calculation of interference loadings at each intermediate time step, thereby reducing computational time significantly. The resultant savings in computational time for the trajectory simulation examined in this study is listed in Table II of Chapter V.

Some care should be exercised in choosing which computational mode, the KSTABLE=0 mode or the KSTABLE=1 mode, is used. In the example of the M-117 bomb trajectory simulation, interference loads did not change much for intermediate time steps. This was probably due to the high weight and

good stability of the bomb causing small pitch oscillations at these intermediate time steps. This may not be the case, however, for unstable or lighter weight stores, so trajectory sensitivity studies should be run, using both computational modes, at extreme Mach numbers and angles of attack. A comparison of trajectories calculated from each mode should enable the determination of the suitability of the time-saving KSTABLE=1 mode for the aircraft/store under consideration.

Appendix B

User's Guide to the Computer Program

Most of the inputs to the computer program describe the aircraft geometry, store loading, coefficients, and parameters to determine the number of flow singularities used to represent the interference flow field as seen by the store. These items are input exactly as described by Fernandes (Ref 3:26-47) and will not be repeated here. The only change from the original Fernandes inputs is his control parameter for run stacking, IGO. This final Fernandes input is eliminated altogether for the present computer program.

The remaining inputs required by the present computer program will now be presented.

CARD 1 (after Fernandes inputs) consists of four control parameters input in format 4I5.

KSTABLE is the computational mode parameter discussed in Appendix A. KSTABLE=0 causes calculation of interference forces at all time steps; KSTABLE=1 causes calculation of interference forces at major time steps only.

NEJECT is the ejector curve control parameter. NJECT=0 implies the ejector curve to be input is Force-distance; NEJECT=1 implies Force-time.

NGAM is the index controlling flight path angle. NGAM=0 causes the trajectory simulation to be free-air; NGAM=1 causes a wind tunnel captive-store simulation.

NCNMAX is the index indicating the maximum number of major time steps allowed during integration.

CARD 2 contains four store parameters input in 4F10.5 format.

SMASS is store mass. (slugs)

RGYRAY is store radius of gyration about y-axis. (feet)

XCG is the store center of gravity measured behind store nose. (feet)

SRMAX is store maximum radius. (feet)

CARD 3 contains these four parameters input in 4E12.4 format.

VINF is aircraft flight velocity. (feet per second)

GAMF is aircraft flight path angle. (degrees)

RHO is air density at simulated altitude. (slugs per cubic foot)

G is acceleration due to gravity. (feet per second per second)

CARD 4 requires these four inputs in format 4E12.4.

CA is the store axial force coefficient, assumed constant throughout the simulation.

VZERO is the store initial translational motion. Direction is normal to store longitudinal axis. (feet per second)

VAR(6) is the store initial pitching velocity about y-axis. Positive is nose up. (radians per second).

Note that VZERO and VAR(6) should be input as zero unless ejector simulation is not desired.

CARD 5 contains a one-dimensional array of order six whose input is 6E12.4.

C(1), C(2), ... C(6) are the coefficients, low-to-high order, of the fifth order polynomial curve fit of the ejector Force-time or Force-distance curves of ejector one.

CARD 6 also contains a one-dimensional array of order six whose input is 6E12.4.

D(1), D(2), ... D(6) are the coefficients, low-to-high order, of the fifth order polynomial curve fit of the ejector Force-time or Force-distance curves of ejector two.

CARD 7 consists of six parameters input in format 6F10.5.

CMQ is the pitch damping coefficient. (per radian)

XL1 is the ejector one piston location relative to the store center of gravity, positive forward of store center of gravity. (feet)

XL2 is the ejector two piston location relative to the store center of gravity, positive forward of store center of gravity. (feet)

EJEND1 is the ejector one cut-off argument. (feet or seconds, depending on NEJECT)

EJEND2 is the ejector two cut-off argument. (feet or seconds, depending on NEJECT)

EJANGL is the angle from the vertical in the aircraft y-z plane, at which each ejector is assumed to act. (degrees)

Note that in this pitch-plane analysis only that component of the ejector force curve in the aircraft z-direction is considered.

CARD 8 contains four time parameters input in format 4E12.4.

DTIME is the initial integration interval. (seconds)

TIMEI is the initial time of trajectory. (seconds)

TIMEF is the final time of trajectory. (seconds)

DTIME2 is the integration interval after both ejectors have stopped. (seconds)

If no ejector is present input DTIME2 equal to DTIME. If at least one ejector is present, DTIME on the order of 0.02 seconds and DTIME2 around 0.05 seconds have proved to be satisfactory. A TIMEF of 0.5 to 0.7 seconds is satisfactory for all but unusual trajectories.

CARDS 9 and 10 contain six inputs necessary to start any trajectory whose initial time is not equal to zero. These items are output from the computer program and are input in 5E14.7 format.

VAR(1) is the x-location of the store center of gravity in the fuselage coordinate system. (feet)

VAR(2) is the z-location of the store center of gravity in the fuselage coordinate system. (feet)

VAR(3) is the store pitch angle about the fuselage y-axis. Positive is nose up. (degrees)

VAR(4) is the x-velocity of the store center of gravity in the fuselage coordinate system. (feet per second)

VAR(5) is the z-velocity of the store center of gravity in the fuselage coordinate system. (feet per second)

VAR(6) is the store pitch rate about the fuselage y-axis. Positive is nose up. (radians per second)

Appendix C

Computer Program Listing

Explanation of Listed Programs

In this appendix are listed the main program and all subroutines with the exception of those 26 subroutines taken from Fernandes (Ref 5). Subroutine ADAMS was taken directly from Goodwin (Ref 6:76). The subroutine FREAIR, listed here as an example, was composed specifically for the M-117 bomb and, of course, will be different for other stores. Subroutine TBLNDC, available from the WPAFB computer library is called from the specific FREAIR listed in this appendix, and is included for completeness.

Changes to the Fernandes Subroutines

Changes were necessary to the Fernandes subroutines to pass the interference coefficients to the main program and to avoid extraneous output.

The Fernandes program has card identifiers, in columns 75 through 79, sequenced in intervals of 10. Cards listed below whose identifiers end in the number "0" are to replace those in the Fernandes subroutines with the same identifiers. All other cards are to be inserted into the Fernandes subroutines in the sequential order implied by their identifiers.

Subroutine INFLUN

```
COMMON /TRAJ/                                   KRUN,KSTOP,CNZF,CMZR      E  85
IF(KRUN .EQ. 0)WRITE (6,90) ALFH,ALFN                                     E 110
IF(KRUN .EQ. 0)WRITE (6,93) POLL                                          E 130
IF(KRUN .EQ. 0)WRITE (6,110) X,CNZ,CNY,CMZ,CMY,CL,CNZP,CNYP,CMZP,C        E 590
1MYP                                                                      E 595
```

Subroutine DISTURB

```
COMMON /TRAJ/                                   KRUN,KSTOP,CNI,CMI        D  85
IF(KRUN .EQ. 0)WRITE (6,130) XM(L),YM(L),ZM(L),XT(L),ALFMM(L)             D 110
IF(KRUN .EQ. 0)WRITE (6,160) IP,IA,IN,NK,FIM,DIA                          D 260
IF(KRUN .EQ. 0)WRITE (6,150)                                              D 650
IF(KRUN .NE. 0)RETURN                                                     D 725
```

```
PROGRAM DROPIT      74/74    OPT=0   TRACE                    FTN 4.1+P373        08/27/74

                PROGRAM DROPIT(INPUT,OUTPUT,PUNCH=60,TAPE5=INPUT,TAPE6=OUTPUT)
          C
          C
          C     PROGRAM DROPIT COMPUTES THE 3-DOF SEPARATION TRAJECTORIES OF
          C     AIRCRAFT STORES
          C
          C     INTERFERENCE LOADING COEFFICIENTS CALCULATED USING METHOD (AND
          C     SUBROUTINES) DUE TO F. DAN FERNANDES (SEE NASA-CR-112065-1)
          C
          C     INTEGRATION OF EQUATIONS OF MOTION TAKEN FROM GOODWIN, NIELSEN,
          C     AND DILLENIUS (SEE NEAR TP 30 AND AFFDL-TR-71-81)
          C
          C     CARDS IDENTIFIED IN COLUMNS 73-80 WITH "A XXX", WHERE "XXX" IS
          C     A THREE-DIGIT NUMBER, WERE TAKEN DIRECTLY FROM THE FERNANDES MAIN
          C     PROGRAM
          C
          C     CARDS IDENTIFIED IN COLUMNS 73-80 WITH "TRAJXXX" OR "TPJLXXX",
          C     WHERE "XXX" IS A THREE-DIGIT NUMBER, WERE TAKEN DIRECTLY FROM THE
          C     GOODWIN MAIN PROGRAM AND SUBROUTINES (SEE NEAR TP 30)
          C
          C
                COMMON /TRAJ/                                  NRUN,KSTOP,CNI,CMI
                COMMON /RC/ RAD,PI,BETA,PM,IGO,BETA2,F5                              A 300
                COMMON /OTC/ XP(1),YP(1),ZP(1),XT(1),ALFRW(1)                        A 310
                DIMENSION ALFR(5)                                                    A 320
                DATA (RAD=0.01745329),(PI=3.14159),(NRUN=0),(IGO=3)                  A 330
                KSTOP=0
                KODE=0
                WRITE (6,90)                                                         A 340
          10    CONTINUE                                                             A 350
                CALL TITLES (2,0,0)                                                  A 360
                CALL TIMEX                                                           A 370
                READ (5,120) BM                                                      A 380
                BETA2=ABS (1.0-BM**2)                                                A 390
                BETA=SQRT(BETA2)                                                     A 400
                WRITE (6,100) BM,BETA                                                A 410
                IF (BM.EQ.1.0) GO TO 70                                              A 420
                CALL AIRCRT                                                          A 430
                CALL TIMEX                                                           A 440
          20    CONTINUE                                                             A 450
                IF (IGO.EQ.2) GO TO 30                                               A 460
                CALL TITLES (0,2,0)                                                  A 470
                CALL STORE                                                           A 480
          30    READ(5,110)NTL
                CALL TITLES(0,0,0)
                READ(5,120)XM(1),YM(1),ZM(1),XT(1),ALFRW(1)
                READ(5,115)NALF,IR,IA,IW
                READ(5,120)ALFR(1)
                CALL TITLES(1,1,1)
                WRITE(6,130)BM
                IF(NRUN .EQ. 0)CALL TPREAD(ALFR(1),KODE,KSTABLE)
          35    CALL DISTUR3(IR,IA,IW)
                IW=0
                CALL TIMEX
                CALL INFLUN(ALFR(1),ALFRW(1),XT(1),IW)
                CALL TIMEX
                CALL TRAJEC(BM,KSTABLE)
```

```
PROGRAM DROPIT      74/74    OPT=0    TRACE                    FTN 4.1+P373        06/27/74

                IF(KSTOP .EQ. 1)STOP
                NRUN=NRUN+1
                GO TO 35
          70    WRITE (6,150)                                                        A 710
          80    CALL EXIT                                                            A 720
          C                                                                          A 730
          C     -END STOPLD (MAIN PROGRAM)                                           A 740
          C                                                                          A 750
          C                                                                          A 760
          C                                                                          A 770
          90    FORMAT (1H1,59HTHIS PROGRAM PREDICTS AERO INTERFERENCE ON AIRCRAFT  A 780
               1 STORES,//25H METHOD FORMULATED AND PROGRAMMED BY M DAN FERNANDES)  A 790
         100    FORMAT (1H0,12HMACH NUMBER=,F6.3,5X,FMCTAR=,F7.4)                   A 800
         110    FORMAT (7I10)                                                       A 810
         120    FORMAT (7F10.4)                                                     A 820
         130    FORMAT (1H0,12HMACH NUMBER,F6.3,5X,11HRESULTS FOR)                  A 830
         140    FORMAT (20X,10HEND OF RUN,5XI2)                                     A 840
         150    FORMAT (1H0,23HINPUT ERROR, BK OR NALF)                             A 850
                END                                                                A 860-
```

```
SUBROUTINE TRPEAD      74/74    OPT=0    TRACE                    FTN 4.1+672        08/27/74

      SUBROUTINE TRPEAD(LJMMY,NCOF,ISTABLE)
C
C
C     CARDS IDENTIFIED IN COLUMN 73-80 WITH "TRJXXXX", WHERE "XXXX" IS
C     A THREE-DIGIT NUMBER, ARE READ DIRECTLY THE ROCKETA MAIN PROGRAM
C     AND SUBROUTINES (SEE WARP TR 33)
C
C
      COMMON /C1C/ XX(1),YM(1),ZM(1),VT(1),ALFMN(1)
      COMMON /1C2/ TX,SCMAX,VMAX,DMCOS,CIC(11),CSIGMY,YMSMIN,VINF,GAME,
     1RHO,G,CT,CL,FS,CAFM(1),C(1),CT(1),CM,CLI,CZ,EJECT1,EJECT2,DTIME,
     2ITMS1,T1,T2,NC,ITMX,ICTIME,CTIME,CTIMEF,CTIMEM,CTIMEP,
     3XIMPS,CMMX,CMMY,CALP,CMI,CD,CLMPF,CMPF,VCOPP,VTCAP,CDAR(6),
     4ANGSTR,NCTR,KP0,IMIN,TJCT,CAM,NCFG,VTCOR,FACTOR,ZTHEPT
      DIMENSION XMRC(1),ZMRC(1)
      DATA NTIP/1/
  415 FORMAT(8F12.4)                                                    TRJA 51
  416 FORMAT(/14 )                                                      TRJA 64
  434 FORMAT(///1X,26HAIRCRAFT FLIGHT CONDITIONS/15X,10HVELOCITY =,1PE1TRJA 98
     11.4,7 [illegible] 15X,17HMACH NUMBER =,1PE7.3,4X,3HEG/1X,17HAIR TRJA 99
     2DENSITY =,1PE11.4,1X,3HSLUG/CU FT/15X,23HGRAVITATIONAL ACCEL:PAIGTRJA102
     3N =,1PE11.4,1X,13HFEET/SEC/SEC)                                   TRJA101
  435 FORMAT(/10X,21HZERO LIFT COEFFICIENT AT ZERO LIFT/15X,6HCMO =,1PE1TRJA102
     111.4/,12X,4HCIJE INDICA, [illegible] TO AIRCRAFT/1X,20HJOTRJA103
     2MOMENT RATIO  MIT [illegible] 1PE11.4,5X,1PE11.4)                 TRJA104
  436 FORMAT(/1X,26HCONTROL F [illegible] IES USED IN FORCE AND MOMENT CALCULTRJA107
     1ATION/10X,11HAREA, S[illegible] 1C LENGTH, FT/14X,1PE11.4,5X,1PE11.4TRJA108
     2) .                                                               TRJA109
 666* FORMAT(5F14.7)                                                    TRJA.6
C                                                                       TRJA653
C     INITIALIZE FOR TRAJECTORY CALCULATION                             TRJA654
C                                                                       TRJA655
 1105 PI=3.1415927                                                      TRJA656
      NCONTR=1
      IF(KONT .EQ. 1)GO TO 871
      READ(5,5)ISTABLE,NEJECT,NGAM,NONMAX
    5 FORMAT(6I5)
      WRITE(6,100)
  100 FORMAT(1H1,  *THIS RUN USES ALL INPUTS AND PARAMETERS NECESSARY
     1FOR TRAJECTORY SIMULATION*/,4X  *THESE INPUTS ARE ALL IN ADDITION
     2TO THOSE REQUIRED BY THE REMARKES PROGRAM.*)
      WRITE(6,110)ISTABLE
  110 FORMAT(  /,1H0,  *ISTABLE = *,I2,  *, -- 0 CALCULATES INTERFERENCE
     1 FORCES AT ALL INTERMEDIATE STEPS,*/,13X *  -- 1 CALCULATES INTERF
     2ERENCE FORCES AT MAJOR TIME STEPS ONLY.*)
      WRITE(6,120)NEJECT
  120 FORMAT(  /,1H0,*NEJECT = *I2*, -- 0 IMPLIES EJECTION CURVE IS FORC
     1E-DISTANCE,*,/,12X,  *  -- 1 IMPLIES EJECTION CURVE IS FORCE-TIME
     2.*)
      WRITE(6,130)NGAM
  130 FORMAT(  /,1H0, *NGAM = *,I2,  *, -- 0 IMPLIES FREE-AIR SIMULATION
     1,*,/12X, *  -- 1 IMPLIES CAPTIVE TRAJECTORY SIMULATION.*)
      WRITE(6,13 ) NONMAX
  135 FORMAT(  /,1H0,*NONMAX = *,I3,*, -- NONMAX IS THE MAXIMUM NUMBER O
     1F MAJOR TIME STEPS ALLOWED DURING TRAJECTORY SIMULATION.*)
      ANGSTR=DUMMY
      RAD=180./PI
```

```
SUBROUTINE TRDEAD     74/74    OPT=0    TRACE                 *              FTN 4.1+P373          09/27/74

                 READ(5,10)FMEX,XM1,GMASS,RGYRAY
                 WRITE(6,140)FMEX,XM1,GMASS,RGYRAY
140              FORMAT(///,1H0,  *SLUG MASS RADIUS IN FEET = *,F10.4,/,
                1    * SLUG OF REMOTE MASS IN SLUGS = *,F10.4,/,
                2    * SLUG MASS X-AXIS IN SLUGS = *,F10.4,/,
                3    * SLUG Y-AXIS OF ROTATION IN FEET = *,F10.4)
                 YMOMIN=GMASS*RGYRAY*RGYRAY
10               FORMAT(8F10.5)
                 READ (5,430) VINF,GAMF,RHO,G                                          TRJA657
                 WRITE (6,430) VINF,GAMF,RHO,G                                         TRJA658
                 READ (5,4.5) OPS,VZERO,VAR(6)                                         TRJA659
                 READ(5,150)C
                 READ(5,4250)D
                 READ(5,10)CM1,XL1,XL2,EJCND1,EJCND2,EJANGL
                 EJANGL=COS(EJANGL/RAD)
                 WRITE(6,150)C
150              FORMAT(///,* EJECTOR ONE COEFFICIENTS = *,6(1XE12.4))
                 WRITE(6,160)D
160              FORMAT(* EJECTOR TWO COEFFICIENTS = *,6(1XE12.4))
                 WRITE(6,170)CJCND1,EJCND2
170              FORMAT(///,* EJECTOR ONE CUT-OFF ARGUMENT = *,F10.4,
                1/,* EJECTOR TWO CUT-OFF ARGUMENT = *,F10.4)
                 WRITE(6,180)EJANGL
180              FORMAT(* EJECTOR ANGLE IN DEGREES = *,F10.4)
                 READ(5,435)                                                           TRJA660
                 READ (5,435) OMG,VZERO,VAR(6)                                         TRJA661
                 STP(1)=-ALPHA(1)
                 VAR(3)=SIN(STP)/RAD                                                   TRJA662
                 VAR(1)=XM(1)
                 VAR(2)=-ZM(1)
                 VAR(4)=-VZERO*SIN(VAR(3))                                             TRJA665
                 VAR(5)=-VZERO*COS(VAR(3))                                             TRJA666
                 THTH=VAR(3)                                                           TRJA667
                 READ(5,405) DTIME,TIMEI,TIMEF,DTIME2
                 NEQ=6                                                                 TRJA669
                 DDTIME=DTIME                                                          TRJA670
                 TIME=TIMEI                                                            TRJA671
                 ZINERT=VAR(2)
                 IF (TIME.LE.0.0) GO TO 870                                            TRJA672
                 READ (5,665A) VAR
                 VAR(3)=VAR(3)/RAD                                                     TRJA674
                 XM(1)=VAR(1)
                 ZM(1)=-VAR(2)
                 ALPHA(1)=-VAR(3)*RAD
870              NDIFEQ=1                                                              TRJA675
                 GO TO 872
871              DTIME=DTIME2
                 DDTIME=DTIME
                 EJFNC1=100.
                 EJFNC2=100.
                 NDIFEQ=1
                 DO 70 I=1,6
                 C(I)=0.
                 D(I)=0.
70               CONTINUE
                 CALL ADAMS(DTIME,DDTIME,VAR,DVAR,NEQ,NDIFEQ,TIME)
                 NOUT=0
```

```
SUBROUTINE TRPEAD      74/74    OPT=0    TRACE      .                FTN 4.1+P573        08/27/74

115                 RETURN
            872     CALL ADAMS(TIME,DDTIME,XX,DVAR,NEQ,NDIFEQ,IISE)
                    NOMT=1                                                                  TRJA677
                    REFARE=PI*SRMAX*SRMAX                                                   TRJA678
                    REFLGH=SRMAX*SRMAX                                                      TRJA679
120                 VIRG=VINF*COS(GAMF/RAD)                                                 TRJA680
                    VIRSF=VINF*SIN(GAMF/RAD)                                                TRJA681
                    CONS=0.5*RHOINF*REFARE/SMASS                                            TRJA682
                    CONS=CONS*REFLGH/(RGYRAY**2)                                            TRJA683
            899     WRITE (6,426) REFARE,REFLGH                                             TRJA693
125                 CALP=COS(ANGA1K/RAD)                                                    TRJA700
                    SALP=SIN(ANGA1K/RAD)                                                    TRJA701
                    VIN=VAR(1)*CALP+VAR(2)*SALP                                             TRJA702
                    VIN=VAR(2)*CALP-VAR(1)*SALP                                             TRJA703
                    CGFAF=COS(GAMF+ANGA1K/RAD)                                              TRJA704
130                 SGFAF=SIN(GAMF+ANGA1K/RAD)                                              TRJA705
                    VIGAF=VIN*COS(ANGA1K/RAD)                                               TRJA706
                    VIGAF=VIN*SIN(ANGA1K/RAD)                                               TRJA707
            C                                                                               TRJA708
            C       CALCULATE FORCES AND MOMENTS                                            TRJA709
135         C                                                                               TRJA710
                    KODE=0
                    RETURN
                    END
```

```
SUBROUTINE TRAJEC      74/74    OPT=0    TRACE                    FTN 4.1+P375          08/27/74

      SUBROUTINE TRAJEC(XMACH,KSTABLE)
C
C
C     CARDS IDENTIFIED IN COLUMNS 73-80 WITH "TRJAXXX", WHERE "XXX" IS
C     A THREE-DIGIT NUMBER, ARE TAKEN DIRECTLY THE GOODWIN MAIN PROGRAM
C     AND SUBROUTINES (SEE MEAP IP 33)
C
C
      COMMON /TRAJ/                                  NRUN,KSTOP,CNI,CMI
      COMMON /CSIZ/ XM(1),YM(11,7M(1),XT(1),(LEM(1)
      COMMON /T ERD/ GRAV,XOS,GMA,GSIN(1),RGYRAY,XMOMIN,VINF,GAMF,
     1RHO,C,CDA,V EFF,,AP(6),TIE),Z(1),CMS,XL1,XL2,EJ,RM1,EJ,RM2,DTIME,
     2TIMEI,TIMLI,TEF,TIME,DRIMT,WSTEO,XMT,TFAC,RSFLOW,VISOF,
     3VISGF,CDMI,CDMI,CALP,SALF,RN,RIN,RGFAM,SGFAF,SIGAF,VISAT,D.AP(6),
     4ANGAIR,NCN,NFA,RAN,THIN,NCHECK,NRAM,NSEG,VISSF,FACTOR,ZINERT
  437 FORMAT(1H0,6X,6HTIME =,1PE11.4,1X,7HSECONDS)                    TRJA106
  440 FORMAT(//5X,7HLOCATION OF ST AE CENTER OF GRAVITY AND STORE PITCHTRJA112
     1 ANGLE IN EQUIVALENT WIND-AXIS SYSTEM/10X,8HMEASURED RELATIVE TO THTRJA113
     2E LOCATION THE C.G. WOULD HAVE BEEN IN ACTUAL ATTITUDE TH/10X,  TRJA114
     3HSTORE WOULD HAVE HAD HAD IT REMAINED ATTACHED                  TRJA115
     42HT TO THE FLIGHT ATTITUDE/10X,5HXRIT,FT,                       TRJA116
     57X,9HZRIT,FT,9X,7HPS, DEG/(1X),Z(1PE10.4))                      TRJA117
  441 FORMAT(//5X,57HLOCATION OF THE CENTER OF GRAVITY AND STORE PITCHTRJA118
     1 ANGLE IN EQUIVALENT COORDINATE SYSTEM/10X,8HLOCATION MEASURED RELATRJA119
     2TIVE TO EQUIVALENT NOSE/10X,6HPITCH ANGLE IS ANGLE BETWEEN EQUIVALTRJA120
     3 AXIS AND SLIPE OF AXIS//20X,5HXRE, FT,10X,6HZRE, FT,8X,10HPSI, DEG, TRJA121
     4/14X,7(1PE10.4))                                                TRJA122
  442 FORMAT(//5X,6HVELOCITIES AND ACCELERATIONS OF STORE RELATIVE TO FTRJA123
     1USELAGE MOTION//20X,11HVXRE, FT/SEC,8X,11HVZRE, FT/SEC,8X,15HTRJA124
     2 RAD/SEC/19X,1PE11.4,8X,E11.4,11X,E11.4//17X,1PE22.4E, FT/SEC/SEC, TRJA125
     33X,19HDZRE, FT/SEC/SEC,16,20HDTHETA, RAD/SEC/SEC/19X,E11.4,8X,   TRJA126
     4E11.4,11X,E11.4/)                                               TRJA127
  443 FORMAT(//5X,5HSTORE/7X,22HFREE STREAM VELOCITY =,1PE12.4,7H FT/SEC/TRJA128
     17X,19HFLIGHT PATH ANGLE =,1PE12.4,4H DEG//7X,17HANGLE OF ATTACK =, TRJA129
     21PE12.4,4H DEG)                                                 TRJA130
  862 VARZD=VAR(4)                                                    TRJA711
      VARED=VAR(5)                                                    TRJA712
      VSTORE=SQRT((VTGAF-VARZD)**2+(-VISAF+VARED)**2)                 TRJA713
      VSTOSQ=VSTORE*VSTORE                                            TRJA714
      VVERT=VISGF-VARZD*SGFAF+VARED*CGFAF                             TRJA715
      VHORT=VTGGF-VARZD*CGFAF-VARED*SGFAF                             TRJA716
      GAMS=ATAN(VVERT/VHORT)                                          TRJA717
      IF (NGAM.NE.1) GAMS=GAMF                                        TRJA718
      ALPS=GAMS+ATAN(-VARZD*VAR(3)-GAMS                               TRJA719
      IF (NCHI.EQ.1) WRITE (6,437) TIME                               TRJA720
      CALL FRGAIR(ALPS,XMACH,CNS,CMS)
C     CALL FCRGF                                                      TRJA721
      FZ1=FZ2=0.
      EJARG=TIME
      IF(EJCT .EQ. 0)EJARG=ABS(VAR(2)-ZINERT)/FACTOR
      IF(EJARG .LE. EJEND1)CALL EJCTR(0,EJARG,FZ1,FACTOR)
      IF(EJARG .LE. EJEND2)CALL EJCTR(0,EJARG,FZ2,FACTOR)


      XEMOM1=-FZ1*XL1
      XEMOM2=-FZ2*XL2
      QSR=CDMI*XMOMIN*VSTOSQ
```

```
SUBROUTINE TRAJTC      74/74    OPT=0    TRACE                    FTN 4.1+P373        08/27/74

                 OS=OSN/PERLGH
                 OMEJ1=EMEJ1/OSP
                 OMEJ2=EMEJ2/OSP
                 OMEJ1=E/170.
                 OMEJ2=E/22/OS
                 OMPAMP=OMS*(VAR(4)*PERLGH/(2.*VSTOPE))
                 OMT=OMI+OMS+OMEJ1+OMEJ2
                 OMT=OMI+OMS+OMEJ1+OMEJ2+OMPAMP
                 IF(NOUT .EQ. 0)GO TO 100
                 WRITE(6,787)OMI,OMT,OMS,OMEJ1,OMEJ2,OMEJ2
                 WRITE(6,788)OMP,OMT,OMT
          100    CONTINUE
          787    FORMAT(// 5X,18HPROPES AND EMENTS/26X,2HON,11X,2HOM/7X,12HINTERME
                1DIATE,2(1PE12.4)/7X,8HFEED 1E6,4X,2(1PE12.4)//7X,12HEJECTOR ONE ,2(
                21PE12.4)/7X,12HEJECTOR TWO 2(1PE12.4))
          788    FORMAT(7X,7HDAMPING,17X,1PE12.4/7X,5HTOTAL,7X,2(1PE12.4))
                 DVAR(1)=VAR(4)                                                    TRJA722
                 DVAR(2)=VAR(5)                                                    TRJA723
                 DVAR(3)=VAR(6)                                                    TRJA724
                 DVAR(4)=OMPAMP*VSTOSQ*(( OMT )*SIN(VAR(3))+GRO*COS((ANGATK+GAMF-GA
                1MS)/RAD))+GSOM*SF                                                 TRJA725
                 DVAR(5)=OMPAMP*VSTOSQ*(( OMT )*COS(VAR(3))+GRO*SIN((ANGATK+GAMF-GA
                1MS)/RAD))+GSOM*CF                                                 TRJA726
                 DVAR(6)=OMPAMP*VSTOSQ*(( OMT )                                    TRJA727
                 IF (NOUT.EQ. 1 GO TO 460                                          TRJA730
                 DUMA=RADD(VAR(3)-TPIN)                                            TRJA731
                 DUMB=VAR(1)*COLP-VAR(2)*SALP+ZIN                                  TRJA732
                 DUMC=VAR(2)*COLP-VAR(1)*SALP-ZIN                                  TRJA733
                 WRITE (6,465) DUMB,DUMC,DUMA                                      TRJA734
                 DUMA=RAD*VAR(3)                                                   TRJA735
                 WRITE (6,461) VAR(1),VAR(2),DUMA                                  TRJA736
                 WRITE (6,462) DVAR                                                TRJA737
                 WRITE (6,463) VSTOPE,GAMS,ALPS                                    TRJA738
                 NA=NSEG/5                                                         TRJA739
                 NOUT=0                                                            TRJA749
                 IF(NCOLTR .GE. NONMAX)GO TO 49
                 NCONTR=NCONTR+1
                 IF((EJARG .LT. EJEND1) .OR. (EJARG .LT. EJEND2))GO TO 40
                 KODE=1
                 CALL TREAD(ANGATK,KODE,KSTABLE)
                 GO TO 860
          40     IF (TIME+1.0E-06-TIMEF) 860,49,49                                 TRJA750
          49     KSTOP=1
                 RETURN
          860    CALL ADAMS (DTIME,DDTIME,VAR,DVAR,NEQ,NDIFEQ,TIME)                TRJA751
          880    IF (NDIFEQ.LE.1) GO TO 49                                         TRJA752
                 IF (NDIFEQ.GT.7) NOUT=1                                           TRJA753
                 XN(1)=VAR(1)
                 ZN(1)=-VAR(2)
                 ALPDN(1)=-VAR(3)*RAD
                 IF(KSTABLE .EQ. 0)RETURN
                 IF(NOUT .LE. 1)RETURN
                 GO TO 862
                 END
```

```
SUBROUTINE EJECTR     74/74    OPT=0    TRACE                    FTN 4.1+P373        08/27/74

                SUBROUTINE EJECTR(C,ARG,FZ,FAC)
                DIMENSION C(6)
                FZ=C(1)
                DO 10 I=2,6
                TEMP=ARG**(I-1)
                FZ=FZ+C(I)*TEMP
           10   CONTINUE
                FZ=FZ*FAC
                RETURN
                END
```

```
SUBROUTINE FREAIR     74/74    OPT=0    TRACE                    FTN 4.1+P373        08/27/74

      SUBROUTINE FREAIR(THETA,XMACH,CN,CM)
      DIMENSION TRXMTH(17),TRYCN(76),TRYCM(76),NA(2),XA(2)
      DATA NA,K,N/4,9,1,3/
      DATA TRXMTH/0.,0.5,0.8,0.9,0.,.2,.4,.6,.8,.10,.12,.14,.15./
      DATA TRYCN/0.,0.12,0.24,0.37,0.52,0.68,0.84,1.03,1.14,
     1           0.,0.12,0.24,0.38,0.53,0.69,0.84,1.03,1.14,
     2           0.,0.13,0.28,0.41,0.56,0.72,0.94,1.13,1.24,
     3           0.,0.13,0.25,0.45,0.6 ,0.74,0.87,1.21,1.31/
      DATA TRYCM/0.,-.172,-.304,-.455,-.614,-.778,-1.426,-1.634,-1.858,
     1           0.,-.137,-.304,-.454,-.614,-.778,-1.426,-1.634,-1.858,
     2           0.,-.153,-.305,-.611,-.911,-1.085,-1.445,-1.825,-1.972,
     3           0.,-.193,-.412,-.627,-.873,-1.178,-1.456,-1.566,-1.780/
      XA(1)=XMACH
      XA(2)=ABS(THETA)
      CN=TSLNOD(K,N,TRYCN,TRXMTH,NA,XA)
      CM=TSLNOD(K,N,TRYCM,TRXMTH,NA,XA)
      CN=CN*THETA/XA(2)
      CM=CM*THETA/XA(2)
      RETURN
      END
```

```
FUNCTION TBLNOC      74/74    OPT=0    19ACE                    FTN 4.1+P373        06/27/74

      FUNCTION TBLNOC(NEXTR,ND,Z,Y,NA,XA)                                       TBLN0001
      DIMENSION X(1),NA(1),XS(1),NS(6),NJ(22),ITOT(6),NGROUP(6),                TBLN0045
     1ITOT(6),Z(1)                                                              TBLN0046
      IF(ND.LE.6)GO TO 1                                                        TBLN0047
      PRINT 2                                                                   TBLN0048
    2 FORMAT(1H0,10X,20HERROR CONDITION-TBLND ROUTINE)                          TBLN0049
      PRINT 5, ND                                                               TBLN0050
    5 FORMAT(11X,30HDIMENSION OF TABLE LOOK-UP (M=,                             TBLN0051
     1I2,19H) IS GREATER THAN 6)                                                TBLN0052
    1 L1=2                                                                      TBLN0053
      LF=ND-1                                                                   TBLN0054
      DO 3 I=1,LF                                                               TBLN0055
      L2=L1+NA(I)-2                                                             TBLN0056
      FOUND=0.                                                                  TBLN0057
      DO 4 J=L1,L2                                                              TBLN0058
      IF(X(J).GT.X(J-1)) GO TO 6                                                TBLN0059
      PRINT 2                                                                   TBLN0060
      PRINT 40, I                                                               TBLN0061
   40 FORMAT(11X,23HINDEPENDENT VECTOR NO. ,I2,20H IS NOT IN ASCENDING          TBLN0062
     27H ORDER.)                                                                TBLN0063
      CALL SYSTEM(200,0)                                                        TBLN0064
      STOP                                                                      TBLN0065
    6 IF (FOUND.NE.0.) GO TO 4                                                  TBLN0066
      IF (XA(I)-X(J-1))8,8,4                                                    TBLN0067
    8 IF (J.EQ.L1) GO TO 10                                                     TBLN0068
      IF (NS(I).EQ.0) GO TO 37                                                  TBLN0069
      FOUND=1.                                                                  TBLN0070
      NS(I)=L1-1                                                                TBLN0071
      GO TO 4                                                                   TBLN0072
   10 FOUND=1.                                                                  TBLN0073
      NS(I)=J-2                                                                 TBLN0074
    4 CONTINUE                                                                  TBLN0075
      IF (FOUND) 11,12,11                                                       TBLN0076
   12 IF (XA(I)-X(L2))13,13,14                                                  TBLN0077
   14 IF (NEXTR.NE.0) GO TO 13                                                  TBLN0078
   37 PRINT 2                                                                   TBLN0079
      PRINT 41, I                                                               TBLN0080
   41 FORMAT(11X,28HINDEPENDENT PARAMETER NO. ,I2,17H IS OUT OF RANGE           TBLN0081
     24HOF CORRESPONDING INDEPENDENT VECTOR AND K=0.)                           TBLN0082
      CALL SYSTEM(200,0)                                                        TBLN0083
      STOP                                                                      TBLN0084
   13 NS(I)=L2-1                                                                TBLN0085
   11 L1=L2+2                                                                   TBLN0086
    3 CONTINUE                                                                  TBLN0087
      DO 15 I=1,LF                                                              TBLN0090
      K=NS(I)                                                                   TBLN0091
      PATIO(I)=(XA(I)-X(K))/(X(K+1)-X(K))                                       TBLN0092
   15 CONTINUE                                                                  TBLN0094
      NGROUP(1)=NS(1)                                                           TBLN0095
      NSUM=NJ(1)                                                                TBLN0096
      DO 16 I=2,LF                                                              TBLN0097
      NGROUP(I)=NS(I)-NSUM                                                      TBLN0098
      NSUM=NSUM+NA(I)                                                           TBLN0099
   16 CONTINUE                                                                  TBLN0100
C     IN NGROUP(I) IS THE SUBSCRIPT OF THE ITH VARIABLE SUCH                    TBLN0101
      ITOT(LF)=1                                                                TBLN0105
      DO 17 I=2,LF                                                              TBLN0107
```

```
FUNCTION TBLNDC      74/74    OPT=0    TRACE                    FTN 4.1+P373        08/27/74

      J=LF-I+1                                                                    TBLND108
      ITOT(J)=ITOT(J+1)*NA(J+1)                                                   TBLND109
   17 CONTINUE                                                                    TBLND110
      KF=2**LF                                                                    TBLND113
      MM=-2                                                                       TBLND114
      DO 22 I=1,KF,2                                                              TBLND115
      IFIRST=1                                                                    TBLND116
      MM=MM+2                                                                     TBLND117
      DO 21 J=1,LF                                                                TBLND118
      MN=2**(J-1)                                                                 TBLND119
      IF(AND(MN,MM).EQ.0)GO TO 18                                                 TBLND120
      IMON=IGROUP(J)+1                                                            TBLND121
      GO TO 19                                                                    TBLND122
   18 IMON=NGROUP(J)                                                              TBLND123
   19 IFIRST=IFIRST+(IMON-1)*ITOT(J)                                              TBLND124
   21 CONTINUE                                                                    TBLND125
      ISEC=IFIRST+ITOT(1)                                                         TBLND126
      WJ(I)=Z(IFIRST)                                                             TBLND127
      WJ(I+1)=Z(ISEC)                                                             TBLND128
   22 CONTINUE                                                                    TBLND129
      DO 24 I=1,LF                                                                TBLND130
      KF=KF/2                                                                     TBLND131
      DO 24 J=1,KF                                                                TBLND132
   24 WJ(J)=WJ(2*J-1)+(WJ(2*J)-WJ(2*J-1))*RATIO(I)                                TBLND133
      TBLNDC=WJ(1)                                                                TBLND134
      RETURN                                                                      TBLND135
      END                                                                         TBLND136
```

```
SUBROUTINE ADAMS      74/74    OPT=0    TRACE                    FTN 4.1+P373        09/27/74

      SUBROUTINE ADAMS (H,DS,N,DY,NEQ,NDIFEQ,S)                          TRJL  3
C
C
C     ADAMS INTEGRATION ROUTINE                                          TRJL  1
C                                                                        TRJL  2
C     SUBROUTINE ADAMS TAKEN FROM GORDON (CLS NEAR IR 30)
C
C
      DIMENSION  Y(6),DY(6)                                              TRJL  4
      DIMENSION Y1(11),F1(11),Y5(11),F5(11),Y3(11),DY1(11),Y2(11),DY(11)TRJL  5
     1,F2(11),DY2(11)                                                    TRJL  6
      DATA PI/3.1415927,[illegible],[illegible],[illegible],[illegible],[illegible]/  TRJL  7
      GO TO (100,200,300,400,500,600,700,800,900),NDIFEQ                 TRJL  8
C                                                                        TRJL  9
C     START BY RUNGE-KUTTA                                               TRJL 10
C                                                                        TRJL 11
  100 H=DS*DS                                                            TRJL 12
 1101 JR=1                                                               TRJL 13
      DO 101 I=1,NEQ                                                     TRJL 14
  101 Y1(I)=Y(I)                                                         TRJL 15
  102 SX1=S                                                              TRJL 16
  103 NDIFEQ=2                                                           TRJL 17
      RETURN                                                             TRJL 18
  200 DO 201 I=1,NEQ                                                     TRJL 19
      DY3(I)=DY(I)                                                       TRJL 20
      Y5(I)=Y(I)                                                         TRJL 21
      TEMP=DY(I)                                                         TRJL 22
      Y(I)=Y(I)+TEMP+Y5(I)                                               TRJL 23
  201 F(I)=TEMP                                                          TRJL 24
      S=S+H*S                                                            TRJL 25
      NDIFEQ=3                                                           TRJL 26
      RETURN                                                             TRJL 27
  300 DO 301 I=1,NEQ                                                     TRJL 28
      TEMP=DY(I)                                                         TRJL 29
      Y(I)=Y(I)+.5*TEMP+Y5(I)                                            TRJL 30
  301 F(I)=F(I)+2.0*TEMP                                                 TRJL 31
      NDIFEQ=4                                                           TRJL 32
      RETURN                                                             TRJL 33
  400 DO 401 I=1,NEQ                                                     TRJL 34
      TEMP=DY(I)                                                         TRJL 35
      Y(I)=Y5(I)+TEMP                                                    TRJL 36
  401 F(I)=F(I)+2.0*TEMP                                                 TRJL 37
      S=S+H*S                                                            TRJL 38
      NDIFEQ=5                                                           TRJL 39
      RETURN                                                             TRJL 40
  500 DO 501 I=1,NEQ                                                     TRJL 41
  501 Y(I)=Y5(I)+F(I)*.1666666667*H                                      TRJL 42
      GO TO (502,503,504,505),JB                                         TRJL 43
  502 DO 503 I=1,NEQ                                                     TRJL 44
  503 Y3(I)=Y(I)                                                         TRJL 45
      S=S-H                                                              TRJL 46
 5503 H=.5*H                                                             TRJL 47
      TEMP=S+H                                                           TRJL 48
      IF (TEMP-S) 504,504,505                                            TRJL 49
  504 NDIFEQ=1                                                           TRJL 50
      WRITE (6,750)                                                      TRJL 51
  750 FORMAT(//44H  INTEGRATION INTERVAL HAS BECOME TOO SMALL.//62H  WHNTRJL 52
```

```
SUBROUTINE ADAMS      74/74    OPT=0    TRACE                    FTN 4.1+P373        08/27/74

                1 IT IS ADDED TO INDEPENDENT VARIABLE, NO CHANGE RESULTS.)          TRJL 53
                  RETURN                                                            TRJL 54
            515 JR=2                                                                TRJL 55
           5505 DO 506 I=1,NEQ                                                      TRJL 56
            506 Y(I)=Y1(I)                                                          TRJL 57
                GO TO 102                                                           TRJL 58
            507 DO 508 I=1,NEQ                                                      TRJL 59
                DY1(I)=DY3(I)                                                       TRJL 60
            508 Y2(I)=Y(I)                                                          TRJL 61
                JR=3                                                                TRJL 62
                GO TO 103                                                           TRJL 63
            509 S=S-H                                                               TRJL 64
          C                                                                         TRJL 65
          C     ERROR CHECKING                                                      TRJL 66
          C                                                                         TRJL 67
           5509 TEST=0.0                                                            TRJL 68
                DO 510 I=1,NEQ                                                      TRJL 69
                VX4=Y(I)                                                            TRJL 70
                TEMP= ABS(Y3(I)-VX4)                                                TRJL 71
                IF(TEMP .LE. 0.)GO TO 510
                VX4= ABS(VX4)                                                       TRJL 72
                IF(VX4 .LE. 0.)GO TO 512
                IF(ABS(Y3(I)) .LE. 0.)GO TO 512
          C                                                                         TRJL 73
          C     CHECK FOR RELATIVE ERROR                                            TRJL 74
          C                                                                         TRJL 75
                IF (VX4*RTEST-TEMP) 512,511,511                                     TRJL 76
            511 TEMP=TEMP/VX4                                                       TRJL 77
                GO TO 519                                                           TRJL 78
          C                                                                         TRJL 79
          C     CHECK FOR ABSOLUTE ERROR                                            TRJL 80
          C                                                                         TRJL 81
            512 IF (ATEST-TEMP) 514,514,513                                         TRJL 82
            513 TEMP=TEMP*RATIO                                                     TRJL 83
                GO TO 519                                                           TRJL 84
          C                                                                         TRJL 85
          C     BOTH TESTS FAIL. HALVE INTEGRATION INTERVAL.                        TRJL 86
          C                                                                         TRJL 87
            514 CONTINUE                                                            TRJL 88
            515 S=S-H                                                               TRJL 89
                IF (JR-5) 517,514,516                                               TRJL 90
            516 JR=1                                                                TRJL 91
                GO TO 5505                                                          TRJL 92
            517 DO 518 II=1,NEQ                                                     TRJL 93
            518 Y3(II)=Y2(II)                                                       TRJL 94
                GO TO 5503                                                          TRJL 95
            519 IF (TEST-TEMP) 520,510,510                                          TRJL 96
            520 TEST=TEMP                                                           TRJL 97
            510 CONTINUE                                                            TRJL 98
          C                                                                         TRJL 99
          C     OUTPUT OF RUNGE-KUTTA                                               TRJL100
          C                                                                         TRJL101
                IF (JR-4) 521,802,802                                               TRJL102
            521 NDIFEQ=3                                                            TRJL103
                DO 522 I=1,NEQ                                                      TRJL104
                PX(I)=Y(I)                                                          TRJL105
                PF(I)=DY(I)                                                         TRJL106
```

```
SUBROUTINE ADAMS      74/74    OPT=0    TRACE                    FTN 4.1+P373        05/27/74

      522 Y(I)=Y2(I)                                                              TRJL107
          RETURN                                                                  TRJL108
      600 DO 801 I=1,NEQ                                                          TRJL109
          Y(I)=RY(I)                                                              TRJL110
          DY(I)=RF(I)                                                             TRJL111
      801 DY2(I)=DY3(I)                                                           TRJL112
          JB=4                                                                    TRJL113
          S=S+H                                                                   TRJL114
      802 NDIFEQ=9                                                                TRJL115
          RETURN                                                                  TRJL116
      900 IF (JB-5) 103,901,702                                                   TRJL117
      901 CON=0.041666667*H                                                       TRJL118
          JB=6                                                                    TRJL119
     9901 NDIFEQ=6                                                                TRJL120
          GO TO 600                                                               TRJL121
      902 JB=5                                                                    TRJL122
          TEST=FLD                                                                TRJL123
          GO TO 802                                                               TRJL124
C                                                                                 TRJL125
C     ADAMS INTEGRATION                                                           TRJL126
C                                                                                 TRJL127
      600 DO 601 I=1,NEQ                                                          TRJL128
          Y1(I)=Y(I)                                                              TRJL129
          Y(I)=Y1(I)+CON*(55.0*DY(I)-59.0*DY3(I)+37.0*DY2(I)-9.0*DY1(I))          TRJL130
          DY1(I)=DY2(I)                                                           TRJL131
          DY2(I)=DY3(I)                                                           TRJL132
          DY3(I)=DY(I)                                                            TRJL133
      601 Y3(I)=Y(I)                                                              TRJL134
          S=S+H                                                                   TRJL135
          NDIFEQ=7                                                                TRJL136
          RETURN                                                                  TRJL137
      700 DO 701 I=1,NEQ                                                          TRJL138
      701 Y(I)=Y1(I)+CON*(9.0*DY(I)+19.0*DY3(I)-5.0*DY2(I)+DY1(I))                TRJL139
          GO TO 5509                                                              TRJL140
C                                                                                 TRJL141
C     TEST FOR DOUBLING OF INTEGRATION INTERVAL                                   TRJL142
C                                                                                 TRJL143
      702 IF (TEST-FLD) 703,9901,9901                                             TRJL144
      703 H=4.0*H                                                                 TRJL145
          GO TO 1101                                                              TRJL146
          END                                                                     TRJL147
```

Vita

Joseph M. Manter was born [redacted] PII Redacted He attended The Ohio State University from June 1966 to December 1970, receiving his Bachelor of Science degree in Aeronautical and Astronautical Engineering. In March 1971 he began working at Wright-Patterson Air Force Base, Ohio, in the Airframe Directorate of Systems Engineering, Aeronautical Systems Division. In July 1973 he was given the opportunity to enroll in the Graduate Aeronautical Engineering program at the Air Force Institute of Technology.

Permanent Address: [redacted]

PII Redacted

This thesis was typed by Jane Manemann.