

183
110

**Fast Half-Loop Maneuvers for the F/A-18 Fighter Aircraft
Using a Singular Perturbation Feedback Control Law**

by

Frederick Earl Garrett, Jr.

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of
Master of Science
in
Aerospace Engineering

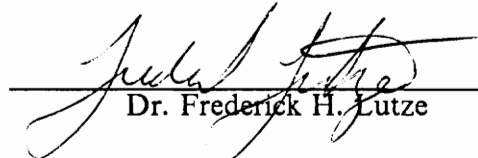
APPROVED:



Dr. Harold L. Stalford, Chairman



Dr. Eugene M. Cliff



Dr. Frederick H. Lutze

September, 1988

Blacksburg, Virginia

c.2

LD
5655
V855
1988
G377
c.2

Our nonlinear feedback control law for agile half-loop maneuvers consists of three parts. A feedback subroutine of table look-up values is used to generate our feedback control law for simulation results. The middle portion is an outer layer solution in which the stabilator angle is a nonlinear function of Mach number, flight path angle and altitude. The angle of attack is nearly constant at the stall value (approximately 36 to 38 degrees). The pitch rate, which is a function of Mach number, flight path angle and altitude, is the state that maximizes the rate of change of the flight path angle while providing equilibrium to the angle of attack and the pitch rate differential equations of motion. This portion of the control law is defined by surfaces (i.e. table look-up values) for the outer layer values of angle of attack, pitch rate and stabilator angle. A data file, consisting of look-up tables for linear interpolation, is used by the feedback subroutine to determine the outer layer values for a given Mach number, flight path angle and altitude. The interpolation nodes are given in Tables 2, 3 and 4 in Appendix A. This data is valid for angles of attack from zero to ninety (0 to 90) degrees; Mach numbers from 0.05 to 0.90; flight path angles from zero to one hundred eighty (0 to 180, using redefined flight path angle) degrees; and altitudes from zero to sixty thousand (0 to 60,000) feet. The feedback subroutine uses these values to determine the outer layer values of the feedback control law (i.e. the stabilator angle of the outer layer). This subroutine also returns the proper control setting for the first and third portions of the feedback control law.

The first and third portions of the feedback control law are transition layer solutions which minimize the time to bring the aircraft from the initial trim values to the outer layer and from the outer layer to the final specified values, respectively. In the transition layers, the stabilator angle feedback control is bang-bang. We emphasize that our feedback control law is not a pure singular perturbation solution since we use the tran-

sition layers to minimize the time in transition to and from the outer layer. But, we do use singular perturbation analysis for all three parts of the control law.

In comparison to simulations of half-loop maneuvers done at NASA Langley Research Center, the feedback control law developed does provide a fast half-loop maneuver. Our maneuver requires about nine (9) seconds to execute the longitudinal portion of the half-loop maneuver for an initial Mach number of 0.9 and an initial altitude of 15,000 feet. The NASA maneuver takes about fifteen (15) seconds to execute the half-loop with the roll included. For initial conditions of Mach 0.6 and 15,000 feet., the time required is approximately thirteen (13) seconds. The NASA simulation of the maneuver (including the roll) took twenty-two (22) seconds for the same initial conditions. In both cases, the aircraft could not execute the half-loop maneuver for an initial Mach number of 0.3 at an altitude of 15,000 feet. This is due to the lack of energy available for this maneuver at 0.3 Mach number. The thrust-to-weight ratio for the F/A-18 aircraft at an altitude of 15,000 feet varies from 0.6 at Mach 0.3 to 0.8 at Mach 0.9.

However, the pilot load factors encountered in some cases exceed those allowable in practice. In the Mach 0.9 case, our maneuver exceeded the pilot load factor limitation of 7.5 g's in the first second. This limitation was not exceeded in the other two cases (i.e. Mach 0.3 and 0.6), but the onset of high load factors are of concern in these maneuvers. For example, the pilot load factor increased from zero (0) g's to approximately six (6) g's in less than one (1) second. In all cases the load factor decreased to low levels after the first second of the maneuver. In order to avoid the high load factors encountered, the first portion of the maneuver could be executed at lower pitch rates over a longer period of time. Therefore, in the regions where load factor limitations are not exceeded,

our feedback control law does produce a relatively fast half-loop maneuver that is useful in a practical sense and as a good baseline for optimal time studies.

Acknowledgements

I would like to thank the following people for their help: Dr. Harold L. Stalford for his tremendous help on this project and for his constant encouragement, even when the going got tough, I will always be indebted to him for his help and support; Dr. E.M.Cliff and Dr. F.H.Lutze for their suggestions and for serving on my committee; and Doug Arbuckle of NASA for making the F/A-18 model available to me for use in this project. This work was partially supported by NASA Langley Research Center under the NASA Contract number NAG-1-873.

I would also like to thank my wife, Beth, for putting up with me during the work on this project and for all the help she gave me. She spent many lonely evenings and many cold dinners by herself. But through it all, she gave a lot of love to me when I most needed it. My friends have also been very helpful and encouraging. Also, thanks to my parents, Mr. and Mrs. Frederick Garrett, Sr. for their unwavering confidence in me.

Finally, my biggest thanks go to the two people who made it possible for my wife and I to attend graduate school. They are my wife's grandparents, Mr. and Mrs. Hugh

Ussery. They gave us the use of their home and made sure that we had everything we needed. This country needs more people who believe in making an education possible and who back it up with all their heart. It is to them that this work is dedicated.

Table of Contents

- 1.0 Introduction** 1
- 1.1 Unconstrained Half-Loop Maneuver 2
- 1.2 F/A-18 Fighter Escort Configuration 3

- 2.0 Problem Definition** 4

- 3.0 Singular Perturbation Approach** 6

- 4.0 F/A - 18 Model** 10

- 5.0 Outer Layer** 23
- 5.1 Analysis 23
- 5.2 Results 29
- 5.3 Pole Placement 53

- 6.0 Transition Region Analysis** 59
- 6.1 First Part of the Transition Region Analysis 60

6.2	Results: First Part of the Transition Region Analysis	63
6.3	Second Part of the Transition Region Analysis	65
7.0	Transition Regions	67
7.1	First Transition Region	68
7.2	Second Transition Region	70
8.0	Nonlinear Feedback Law	72
8.1	First Transition Region	72
8.1.1	Part I	73
8.1.2	Part II	74
8.1.3	Part III	75
8.2	Outer Layer	76
8.3	Second Transition Region	77
9.0	Results	78
10.0	Summary	150
11.0	References	154
Appendix A.	F/A - 18 Fighter Escort Model	156
A.1	Aerodynamic Model	156
A.1.1	Lift Coefficients	157
A.1.2	Drag Coefficient	158
A.1.3	Pitching Moment Coefficient	159
A.1.4	Flap Schedule	160
A.2	Thrust Model	160

A.3 F/A - 18 Constants	161
Appendix B. Modified F/A - 18 Model	162
B.1 Modifications	162
B.2 B - Spline Interpolation	165
B.2.1 Lift Coefficients	166
B.2.2 Drag Coefficient	167
B.2.3 Pitching Moment Coefficients	167
B.2.4 Node Sequences	168
Appendix C. Equations of Motion	173
Appendix D. Computer Programs	176
Vita	223

List of Tables

Table 1. Outer Layer Solution Nodes	30
Table 2. Stabilator Nodes (deg)	169
Table 3. Mach Number Nodes	170
Table 4. Angle of Attack Nodes (deg)	171
Table 5. Model Limits	172

List of Illustrations

Figure 1.	Unconstrained Half-Loop Maneuver	5
Figure 2.	Lift Coefficient (h = 15000 ft.)	14
Figure 3.	Drag Coefficient (h = 15000 ft.)	15
Figure 4.	Pitching Moment Coefficient (h = 15000 ft.)	16
Figure 5.	Pitch Rate Derivative of Lift Coefficient (h = 15000 ft.)	17
Figure 6.	Pitch Rate Derivative of Pitching Moment Coefficient (h = 15000 ft.)	18
Figure 7.	Angle of Attack Rate Derivative of Lift Coefficient (h = 15000 ft.)	19
Figure 8.	Angle of Attack Rate Derivative of Pitching Moment Coefficient (h = 15000 ft.)	20
Figure 9.	Thrust per Engine (h = 15000 ft.)	21
Figure 10.	Thrust in x-body Direction (h = 15000 ft.)	22
Figure 11.	Outer Layer Angle of Attack (h = 5000 ft)	32
Figure 12.	Outer Layer Angle of Attack (h = 15000 ft)	33
Figure 13.	Outer Layer Angle of Attack (h = 25000 ft)	34
Figure 14.	Outer Layer Angle of Attack (h = 35000 ft)	35
Figure 15.	Outer Layer Angle of Attack (h = 45000 ft)	36
Figure 16.	Outer Layer Angle of Attack (h = 55000 ft)	37
Figure 17.	Outer Layer Pitch Rate (h = 5000 ft)	38
Figure 18.	Outer Layer Pitch Rate (h = 15000 ft)	39
Figure 19.	Outer Layer Pitch Rate (h = 25000 ft)	40

Figure 20. Outer Layer Pitch Rate ($h = 35000$ ft)	41
Figure 21. Outer Layer Pitch Rate ($h = 45000$ ft)	42
Figure 22. Outer Layer Pitch Rate ($h = 55000$ ft)	43
Figure 23. Outer Layer Stabilator Deflection ($h = 5000$ ft)	44
Figure 24. Outer Layer Stabilator Deflection ($h = 15000$ ft)	45
Figure 25. Outer Layer Stabilator Deflection ($h = 25000$ ft)	46
Figure 26. Outer Layer Stabilator Deflection ($h = 35000$ ft)	47
Figure 27. Outer Layer Stabilator Deflection ($h = 45000$ ft)	48
Figure 28. Outer Layer Stabilator Deflection ($h = 55000$ ft)	49
Figure 29. Simulation of Outer Layer ($M = 0.6, h = 15000$ ft)	50
Figure 30. Simulation of Outer Layer ($M = 0.6, h = 15000$ ft)	51
Figure 31. Simulation of Outer Layer ($M = 0.6, h = 15000$ ft)	52
Figure 32. Eigenvalues of Fast System for Outer Layer Simulation	57
Figure 33. Eigenvalues of Fast System for Outer Layer Simulation	58
Figure 34. Half-Loop Simulation: Angle of Attack ($M(t=0)=0.9, h=15000$ ft) . .	79
Figure 35. Half-Loop Simulation: Pitch Rate ($M(t=0)=0.9, h=15000$ ft)	80
Figure 36. Half-Loop Simulation: Airspeed ($M(t=0)=0.9, h=15000$ ft)	81
Figure 37. Half-Loop Simulation: Mach Number ($M(t=0)=0.9, h=15000$ ft) . . .	82
Figure 38. Half-Loop Simulation: Flight Path Angle ($M(t=0)=0.9, h=15000$ ft) .	83
Figure 39. Half-Loop Simulation: Stabilator Angle ($M(t=0)=0.9, h=15000$ ft) . .	84
Figure 40. Half-Loop Simulation: Altitude vs Range ($M(t=0)=0.9, h=15000$ ft) .	85
Figure 41. Half-Loop Simulation: Load Factor ($M(t=0)=0.9, h=15000$ ft)	86
Figure 42. Half-Loop Simulation: Load Factor ($M(t=0)=0.9, h=15000$ ft)	87
Figure 43. Half-Loop Simulation: Dynamic Pressure ($M(t=0)=0.9, h=15000$ ft) .	88
Figure 44. Half-Loop Simulation: Fast System Eigenvalues ($M(t=0)=0.9,$ $h=15000$ ft)	89

Figure 45. Half-Loop Simulation: Pole Placement Gain ($M(t=0)=0.9$, $h=15000$ ft)	90
Figure 46. Half-Loop Simulation: Pole Placement Gain ($M(t=0)=0.9$, $h=15000$ ft)	91
Figure 47. Half-Loop Simulation: Angle of Attack ($M(t=0)=0.6$, $h=15000$ ft) . .	92
Figure 48. Half-Loop Simulation: Pitch Rate ($M(t=0)=0.6$, $h=15000$ ft)	93
Figure 49. Half-Loop Simulation: Airspeed ($M(t=0)=0.6$, $h=15000$ ft)	94
Figure 50. Half-Loop Simulation: Mach Number ($M(t=0)=0.6$, $h=15000$ ft) . . .	95
Figure 51. Half-Loop Simulation: Flight Path Angle ($M(t=0)=0.6$, $h=15000$ ft)	96
Figure 52. Half-Loop Simulation: Stabilator Angle ($M(t=0)=0.6$, $h=15000$ ft) . .	97
Figure 53. Half-Loop Simulation: Altitude vs Range ($M(t=0)=0.6$, $h=15000$ ft)	98
Figure 54. Half-Loop Simulation: Load Factor ($M(t=0)=0.6$, $h=15000$ ft)	99
Figure 55. Half-Loop Simulation: Load Factor ($M(t=0)=0.6$, $h=15000$ ft) . . .	100
Figure 56. Half-Loop Simulation: Dynamic Pressure ($M(t=0)=0.6$, $h=15000$ ft)	101
Figure 57. Half-Loop Simulation: Fast System Eigenvalues ($M(t=0)=0.6$, $h=15000$ ft)	102
Figure 58. Half-Loop Simulation: Pole Placement Gain ($M(t=0)=0.6$, $h=15000$ ft)	103
Figure 59. Half-Loop Simulation: Pole Placement Gain ($M(t=0)=0.6$, $h=15000$ ft)	104
Figure 60. Half-Loop Simulation: Angle of Attack ($M(t=0)=0.3$, $h=15000$ ft) .	105
Figure 61. Half-Loop Simulation: Pitch Rate ($M(t=0)=0.3$, $h=15000$ ft)	106
Figure 62. Half-Loop Simulation: Airspeed ($M(t=0)=0.3$, $h=15000$ ft)	107
Figure 63. Half-Loop Simulation: Mach Number ($M(t=0)=0.3$, $h=15000$ ft) . .	108
Figure 64. Half-Loop Simulation: Flight Path Angle ($M(t=0)=0.3$, $h=15000$ ft)	109
Figure 65. Half-Loop Simulation: Stabilator Angle ($M(t=0)=0.3$, $h=15000$ ft) .	110
Figure 66. Half-Loop Simulation: Altitude vs Range ($M(t=0)=0.3$, $h=15000$ ft)	111
Figure 67. Half-Loop Simulation: Load Factor ($M(t=0)=0.3$, $h=15000$ ft) . . .	112
Figure 68. Half-Loop Simulation: Load Factor ($M(t=0)=0.3$, $h=15000$ ft) . . .	113

Figure 69. Half-Loop Simulation: Dynamic Pressure ($M(t=0)=0.3$, $h=15000$ ft) 114

Figure 70. Half-Loop Simulation: Fast System Eigenvalues ($M(t=0)=0.3$, $h=15000$ ft) 115

Figure 71. Half-Loop Simulation: Pole Placement Gain ($M(t=0)=0.3$, $h=15000$ ft) 116

Figure 72. Half-Loop Simulation: Pole Placement Gain ($M(t=0)=0.3$, $h=15000$ ft) 117

Figure 73. Comparison with NASA: Angle of Attack ($M(t=0)=0.9$, $h=15000$ ft) 122

Figure 74. Comparison with NASA: Pitch Rate ($M(t=0)=0.9$, $h=15000$ ft) . . . 123

Figure 75. Comparison with NASA: Airspeed ($M(t=0)=0.9$, $h=15000$ ft) 124

Figure 76. Comparison with NASA: Mach Number ($M(t=0)=0.9$, $h=15000$ ft) 125

Figure 77. Comparison with NASA: Pitch Angle ($M(t=0)=0.9$, $h=15000$ ft) . . 126

Figure 78. Comparison with NASA: Stabilator Angle ($M(t=0)=0.9$, $h=15000$ ft) 127

Figure 79. Comparison with NASA: Altitude ($M(t=0)=0.9$, $h=15000$ ft) 128

Figure 80. Comparison with NASA: Load Factor ($M(t=0)=0.9$, $h=15000$ ft) . . 129

Figure 81. Comparison with NASA: Angle of Attack ($M(t=0)=0.6$, $h=15000$ ft) 130

Figure 82. Comparison with NASA: Pitch Rate ($M(t=0)=0.6$, $h=15000$ ft) . . . 131

Figure 83. Comparison with NASA: Airspeed ($M(t=0)=0.6$, $h=15000$ ft) 132

Figure 84. Comparison with NASA: Mach Number ($M(t=0)=0.6$, $h=15000$ ft) 133

Figure 85. Comparison with NASA: Pitch Angle ($M(t=0)=0.6$, $h=15000$ ft) . . 134

Figure 86. Comparison with NASA: Stabilator Angle ($M(t=0)=0.6$, $h=15000$ ft) 135

Figure 87. Comparison with NASA: Altitude ($M(t=0)=0.6$, $h=15000$ ft) 136

Figure 88. Comparison with NASA: Load Factor ($M(t=0)=0.6$, $h=15000$ ft) . . 137

Figure 89. Comparison with NASA: Angle of Attack ($M(t=0)=0.3$, $h=15000$ ft) 138

Figure 90. Comparison with NASA: Pitch Rate ($M(t=0)=0.3$, $h=15000$ ft) . . . 139

Figure 91. Comparison with NASA: Airspeed ($M(t=0)=0.3$, $h=15000$ ft) 140

Figure 92. Comparison with NASA: Mach Number ($M(t=0)=0.3$, $h=15000$ ft) 141

Figure 93. Comparison with NASA: Pitch Angle ($M(t=0)=0.3$, $h=15000$ ft) . .	142
Figure 94. Comparison with NASA: Stabilator Angle ($M(t=0)=0.3$, $h=15000$ ft)	143
Figure 95. Comparison with NASA: Altitude ($M(t=0)=0.3$, $h=15000$ ft)	144
Figure 96. Comparison with NASA: Load Factor ($M(t=0)=0.3$, $h=15000$ ft) . .	145
Figure 97. Thrust-to-Weight Ratio ($M(t=0)=0.9$, $h=15000$ ft)	147
Figure 98. Thrust-to-Weight Ratio ($M(t=0)=0.6$, $h=15000$ ft)	148
Figure 99. Thrust-to-Weight Ratio ($M(t=0)=0.3$, $h=15000$ ft)	149

List of Symbols

α	Angle of Attack
q	Pitch Rate
g	Gravitational Acceleration
v	Total Velocity
γ	Flight Path Angle
\bar{q}	Dynamic Pressure
S	Wing Area
m	Mass
C_L	Lift Coefficient
T_x	Thrust in x-body direction
T_h	Total Thrust
C_D	Drag Coefficient
I_y	Moment of Inertia about y-axis
l_x	Thrust Offset from x-axis
l_x	Distance along x-axis from center of gravity to the aerodynamic center
l_z	Distance along z-axis from center of gravity to the aerodynamic center

θ	Pitch angle
h	Altitude
d_L	Lift Offset
d_D	Drag Offset
W	Weight
\bar{c}	Mean Aerodynamic Chord
b	Wing Span
\bar{x}	Distance along x-axis from center of gravity to the pilot
\bar{z}	Distance along z-axis from center of gravity to the pilot
δ_h	Stabilator Deflection
δ_{lef}	Leading Edge Flap Deflection
δ_{tef}	Trailing Edge Flap Deflection
δ_T	Throttle Setting
n_x	Load Factor (x-direction)
n_z	Load Factor (z-direction)
n_{x_p}	Pilot Load Factor (x-direction)
n_{z_p}	Pilot Load Factor (z-direction)
ε	Perturbation Parameter
J_{OL}	Outer Layer Performance Index
ΔJ_x	First Inner Layer Performance Index
ΔJ_q	Second Inner Layer Performance Index
t	Time
M	Mach Number
\mathcal{H}	Hamiltonian
f_α	Differential Equation for Angle of Attack
f_q	Differential Equation for Pitch Rate

A	System Matrix for the Fast Outer Layer Variables
B	Input Matrix for the Fast Outer Layer Variables
p_{OL}	Open-Loop Characteristic Equation
p_{CL}	Closed-Loop Characteristic Equation
U	Controllability Matrix
a_1	First Order Coefficient of the Open-Loop Characteristic Equation
a_2	Zerth Order Coefficient of the Open-Loop Characteristic Equation
\bar{a}_1	First Order Coefficient of the Closed-Loop Characteristic Equation
\bar{a}_2	Zerth Order Coefficient of the Closed-Loop Characteristic Equation
$\bar{\lambda}$	Open-Loop Eigenvalues
$\tilde{\lambda}$	Closed-Loop Eigenvalues
ω	Circular Frequency of the Butterworth Poles
W	Lower Toeplitz Matrix
K	Gain Matrix
$\Delta\alpha$	Change in Angle of Attack
$\Delta\gamma$	Change in Flight Path Angle
Δt	Time required to bring Angle of Attack and Pitch Rate to desired values

1.0 Introduction

We desire to develop a nonlinear state feedback control law that is time optimal for the half-loop maneuver of the F/A-18 aircraft. However, there is a tradeoff between constructing a nonlinear feedback control law and constructing one that is optimal. Time optimal control laws are usually open loop. Therefore, to construct an optimal feedback control law, we generally have to synthesize it from a family of optimal open loop control histories. The construction of an optimal feedback law for high angle of attack flight can be extremely difficult, if not impossible, and is therefore beyond the scope of this work. In order to establish a good baseline for a time optimal study, a suboptimal (i.e. fast) feedback control law for the longitudinal portion of a fast half-loop maneuver is developed. The response of the aircraft to the fast feedback control law can then be compared with the results of any time optimal solution obtained in the future. The tradeoff referred to here is thus the relative ease in which a suboptimal feedback control law can be constructed as compared with that obtained by synthesizing optimal open loop controls. The approach used to develop our fast feedback control law is singular perturbation theory. This method is similar to a study done for the T-2C aircraft by

Stalford and Hoffman (Ref. 12). The motivation and details of this approach are discussed in chapter 3.

1.1 Unconstrained Half-Loop Maneuver

The half-loop maneuver consists of a pitch-up in which the aircraft reverses its direction in the vertical plane (see Figure 1). This maneuver is almost entirely executed in the plane of symmetry of the aircraft except for the one hundred and eighty degree roll executed at the end of the maneuver. The roll is considered to be outside the scope of our work. The time to perform the roll, which can be executed rather quickly, is not considered herein.

This maneuver is particularly useful in head-on pass engagements at high Mach numbers when the opponent is at a higher altitude. Major limitations to the maneuver are the high load factors encountered at high Mach numbers and the large amount of thrust required to perform it. For high Mach numbers and low altitudes, the load factor may be more than that allowable for a pilot. A useable feedback law would necessarily have to take pilot limitations into account and constrain the aircraft response within allowable load factor limits.

In this study, a load factor constraint is not used in determining the feedback law. The maneuver being considered is therefore called the "unconstrained half-loop maneuver". However, the load factor is computed in order to expose the regions in which load factor limits are violated. In those regions where the load factor limit is not violated, the un-

constrained feedback law will be identical to the constrained one. The load factor resulting from using the feedback law is a function of Mach number and altitude. This means that there will be a region of Mach numbers and altitudes in which the unconstrained solution will satisfy pilot load factor limits.

1.2 F/A-18 Fighter Escort Configuration

The version of the F/A-18 used for this study is a single-seat aircraft used as an escort fighter. It is meant primarily as a replacement for the F-4 fighter, A-4 and A-7 attack aircrafts (Ref. 7). The escort configuration is assumed to have 60 percent fuel with tip missiles attached. The flight control system consists of horizontal stabilators, ailerons, two twin fin rudders, full span leading-edge flaps and single slot trailing-edge flaps (Ref. 7). For this study, the ailerons and rudders are not used. The leading-edge and trailing-edge flaps are used according to a flap schedule supplied by NASA Langley Research Center. Actuator dynamics and differential control surface deflections are not considered.

2.0 Problem Definition

We desire to design a nonlinear feedback law which will execute, relatively fast, the unconstrained half-loop maneuver. Our work will also establish a baseline for future research work on time optimal controls for this maneuver. The study will only encompass the longitudinal portion of the maneuver for subsonic initial Mach numbers.

A graphical representation of the half-loop maneuver is presented in Figure 1. The aircraft begins the maneuver in straight and level flight. The velocity vector is then rotated through one hundred and eighty degrees flight path angle. It is then necessary to bring the aircraft back to straight and level flight as shown. A one hundred and eighty degree roll can then be executed, at the end, to complete the maneuver.

Stalford and Hoffman (Ref. 12) worked on a similar problem for the T-2C trainer aircraft, but only for the special constant speed case. Their research involved developing a feedback control law that would point the aircraft according to a commanded pitch angle. Their maneuver is very similar in nature to our half-loop maneuver.

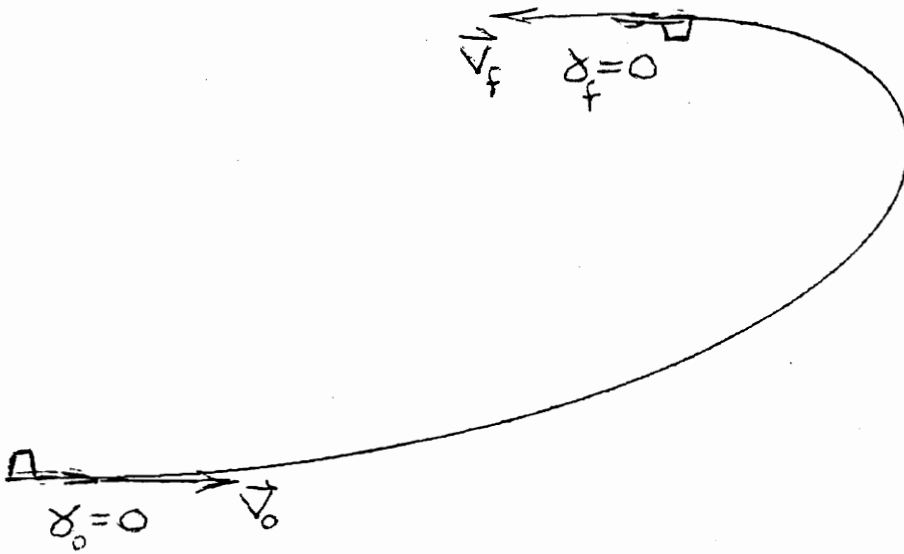


Figure 1. Unconstrained Half-Loop Maneuver

3.0 Singular Perturbation Approach

A singular perturbation approach is used to derive a nonlinear feedback law for the half-loop maneuver of the F/A-18 fighter aircraft. The approach is similar to that used in Stalford and Hoffman (Ref. 12). The problem is solved using several time scales or layers. First, in the outer layer solution the differential equations of angle of attack and pitch rate are reduced to algebraic equations making it possible to hold the states of angle of attack and pitch rate nearly constant on a particular time scale. The algebraic equations obtained are used to derive a feedback law which is valid under these conditions. The resulting stabilator feedback control law is a function of the remaining non-constant states of Mach number, altitude and flight path angle. This approach makes it possible to develop a nonlinear state feedback control law which rotates the velocity vector through one hundred and eighty degrees with nearly constant angle of attack and maximum equilibrium pitch rate. The resulting feedback control law is a function of flight path angle, Mach number and altitude.

The nonlinear feedback law of the half-loop maneuver is derived using three parts; an outer layer and two transition regions. On the outer layer, the angle of attack and pitch

rate are assumed to be fast compared to velocity, flight path angle and altitude. Angle of attack and pitch rate are assumed to be fast because a change in control has a direct effect on the corresponding differential equations of motion. Similarly, a change in control has little direct effect on the flight path angle, Mach number or altitude equations of motion. The changes in these slow states are a consequence of changes in the values of the fast states, angle of attack and pitch rate. In the transition region analysis, to be discussed later, a similar process is applied to determine that the pitch rate is fast compared to the angle of attack.

In order for the outer layer of this maneuver to be executed in "minimum time", we observe that the integrand of the performance criterion is minimized by maximizing the pitch rate. This follows since the angle of attack is being held constant. The right hand sides of the two differential equations governing the angle of attack and the pitch rate variables are set to zero, producing two algebraic equations. That is, the angle of attack and the pitch rate variables are assumed constant. The feedback law for the outer layer is found by solving these two algebraic equations for the values of the angle of attack and the stabilator angle which maximize the pitch rate. The resulting nonlinear control law consists of a set of surfaces for each altitude. These surfaces represent the variables angle of attack, pitch rate and stabilator deflection as a function of flight path angle and Mach number. The surface representing the stabilator deflection is the feedback control law for the outer layer. The surfaces which represent the angle of attack and the pitch rate define the values that these states must necessarily possess in order that the aircraft be in the outer layer. The outer layer solution is thus defined.

The next part in deriving a nonlinear feedback control law is to get the states from the initial conditions to the outer layer in minimum time (i.e. the first transition region).

The last part is the transition from the outer layer to a prescribed terminal state (i.e. the second transition region). This is also to be performed in minimum time. The control of these transitions, together with that of the outer layer, form our feedback control law. In the first transition region, the angle of attack and the pitch rate are again taken to be the fast variables as compared to Mach number, altitude and flight path angle. First, the pitch rate is considered to be fast compared to the angle of attack. The control law which maximizes the time rate of change of angle of attack is found first; then, the time rate of change of pitch rate is maximized to complete the analysis of the first transition region.

The overall feedback law is therefore determined by dividing the maneuver into the three portions discussed above. First there is the outer layer region, where the outer layer stabilator surface is used as a feedback law. Then there are the two transition regions at the beginning and end of the outer layer. The transition regions comprise the transfer of the states from the initial conditions to the outer layer and the transfer from the outer layer to the terminal state at the end of the maneuver. The control laws for the transition regions are derived from singular perturbation analysis.

We emphasize that our entire feedback control law is not a pure singular perturbation solution since in the transition regions, we are using singular perturbation analysis to minimize the time to transfer the aircraft to and from the outer layer. A pure singular perturbation solution would be more complex in the transition layers. A control law obtained by developing inner layer solutions and asymptotically matching these solutions to the derived outer layer (Ref. 3) would be very complicated and the result would be a degradation of time optimality in performing the transition regions. Our feedback control for the transition regions is bang-bang and appears to be nearly time optimal.

We do use singular perturbation analysis to derive the bang-bang control laws. The work done by Stalford and Hoffman (Ref. 12 and Ref. 15) for the T-2C aircraft gives us reason to suspect that our bang-bang control laws for the transition regions are nearly time optimal. That is, the results obtained in Stalford and Hoffman (Ref. 15) using the necessary conditions of optimal control (i.e. the maximum principle) are identical with those obtained in Stalford and Hoffman (Ref. 12) using the singular perturbation analysis in the transition regions.

4.0 F/A - 18 Model

The F/A - 18 Model is based on the NASA Langley Research Center Model described in Appendix A. Modifications made to the NASA Model are given in Appendix B. These modifications are necessary in order to create a smooth and continuous model. This was necessary because the NASA model is based on two different sets of data. One set of data is from wind tunnel tests for angles of attack less than forty degrees and the other set is for angles of attack greater than forty degrees. These two tests were done using different flap settings and thus the model has a discontinuity at forty degrees angle of attack. It is therefore necessary to include a flap schedule which does away with this discontinuity. Unfortunately, the flap schedule does not alleviate the discontinuity for all Mach numbers and altitudes. Therefore, it is necessary to add a correction factor for high Mach numbers and low altitudes which gives us a smooth model. The equations of motion for longitudinal flight are presented in Appendix C and are given as follows:

$$\dot{\alpha} = q + \frac{g}{V} \cos \gamma - \frac{\bar{q}S}{mV} C_L - \frac{T_x}{mV} \sin \alpha \quad (4.1)$$

$$\dot{V} = -g \sin \gamma - \frac{\bar{q}S}{m} C_D + \frac{T_x}{m} \cos \alpha \quad (4.2)$$

$$\dot{q} = \frac{1}{I_y} [l_z T_x + \bar{q} S (\bar{c} C_m + d_L C_L + d_D C_D)] \quad (4.3)$$

$$\dot{\theta} = q \quad (4.4a)$$

$$\dot{\gamma} = q - \dot{\alpha} \quad (4.4b)$$

$$\dot{h} = V \sin \gamma \quad (4.5)$$

where:

$$d_L(\alpha) = l_x \cos \alpha + l_z \sin \alpha \quad (4.6)$$

$$d_D(\alpha) = l_x \sin \alpha - l_z \cos \alpha \quad (4.7)$$

$$\gamma = \theta - \alpha \quad (4.8)$$

The state variables are angle of attack, pitch rate, Mach number, flight path angle and altitude. That is, for analysis purposes, we use (4.4b) instead of (4.4a) since pitch angle, θ , and flight path angle, γ are related by equation (4.8). Importantly, we use (4.4b) instead of (4.4a) since angle of attack, α is fast with respect to γ but not with respect to θ . For the F/A - 18 fighter escort configuration, the constants in the above equations are:

$$I_y = 151,293 \text{ slug ft}^2$$

$$W = 33,310 \text{ lb}$$

$$m = 1034.47 \text{ slugs}$$

$$\bar{c} = 11.52 \text{ ft}$$

$$b = 37.42 \text{ ft}$$

$$S = 400.0 \text{ ft}^2$$

$$l_{z_e} = 2.8 \text{ in}$$

$$l_x = -3.56 \text{ in}$$

$$l_z = 2.8 \text{ in}$$

$$g = 32.174 \text{ ft/s}^2$$

$$\bar{x} = 18 \text{ ft}$$

$$\bar{z} = 6 \text{ ft}$$

The control surface limits are:

$$-24^\circ \leq \delta_h \leq 10.5^\circ$$

$$-3^\circ \leq \delta_{lef} \leq 34^\circ$$

$$-8^\circ \leq \delta_{ref} \leq 45^\circ$$

The actuator no load rate limits are:

$$\text{Stabilator, } \delta_h \quad 40^\circ/s$$

$$\text{Leading edge flap, } \delta_{lef} \quad 18^\circ/s$$

$$\text{Trailing edge flap, } \delta_{ref} \quad 18^\circ/s$$

Note: Actuator dynamics are not used in this study, these limits are given as a reference.

Thrust is a function of altitude, Mach number, and throttle setting. The model is made up of look-up tables which are used by a subroutine to calculate the thrust per engine, T_h . The total thrust in the x-body direction is found by doubling the thrust per engine and taking into account the angle of the engines with respect to the x-body axis.

$$T_x = 2T_h \cos(1.98^\circ)$$

$$T_h = f(M, h, \delta_T)$$

where:

$$\delta_T = \begin{cases} 30^\circ & \text{idle} \\ 106.5^\circ & \text{military} \\ 130^\circ & \text{full afterburner} \end{cases}$$

The thrust in the y-body direction T_y is zero because the engines are mounted in such a way that the y-body components of thrust for each engine cancel each other out.

The pilot load factors are calculated as follows

$$n_{x_p} = \frac{1}{g} \left[\frac{\bar{q}S}{m} ((C_{L_0} + C_{L_q}q + C_{L_\alpha}) \sin \alpha - C_{D_0} \cos \alpha) - \bar{x}q^2 + \bar{z}\dot{q} \right] \quad (4.9)$$

$$n_{z_p} = \frac{1}{g} \left[\frac{\bar{q}S}{m} ((C_{L_0} + C_{L_q}q + C_{L_\alpha}\dot{\alpha}) \cos \alpha + C_{D_0} \sin \alpha) \right] + \bar{x}\dot{q} + \bar{z}q^2 \quad (4.10)$$

Figures 2-8 show the aerodynamic coefficients for an altitude of 15,000 feet. Figures 9 and 10 show the thrust per engine and the total thrust in the x-body direction respectively for an altitude of 15,000 feet.

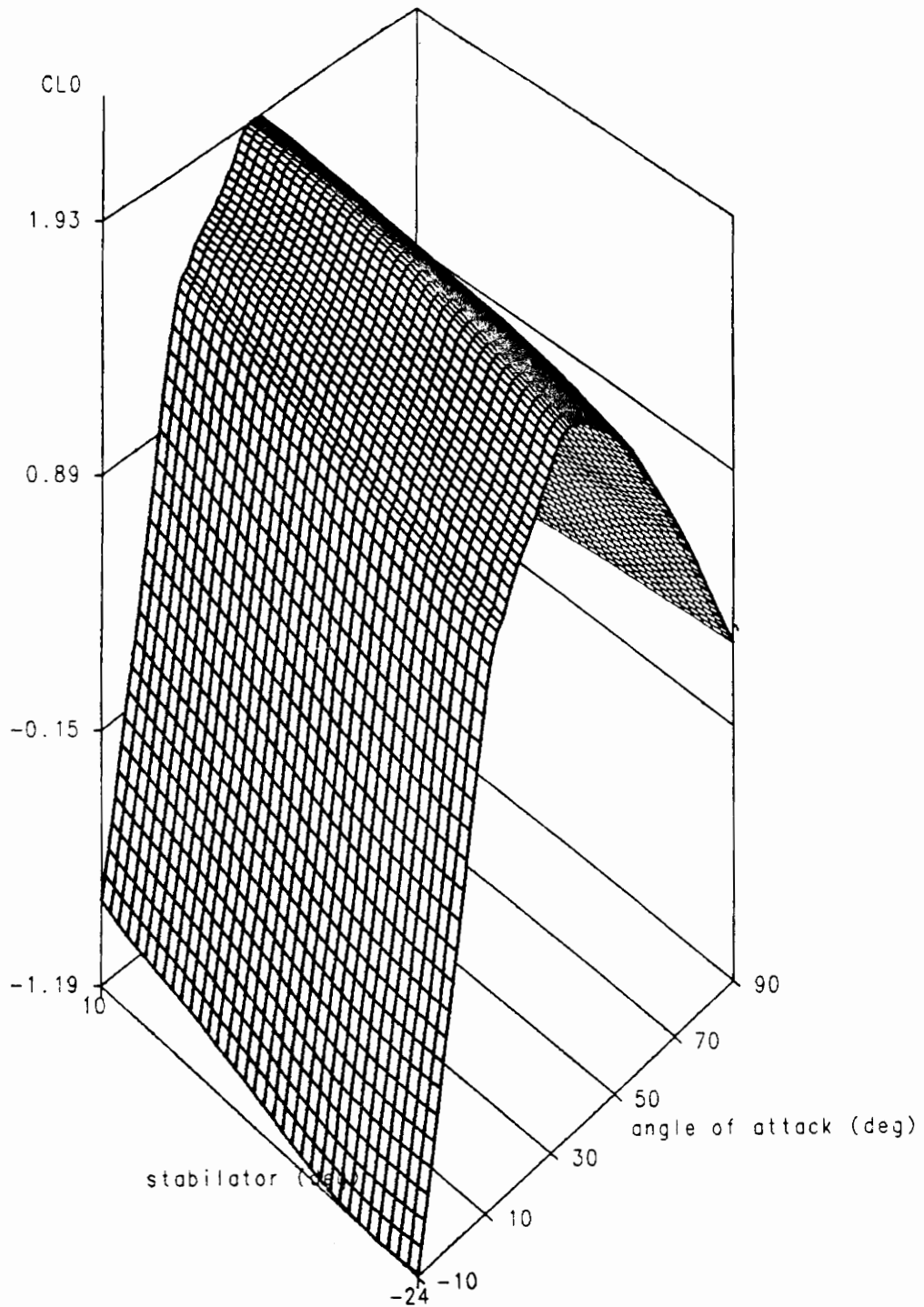


Figure 2. Lift Coefficient (h = 15000 ft.)

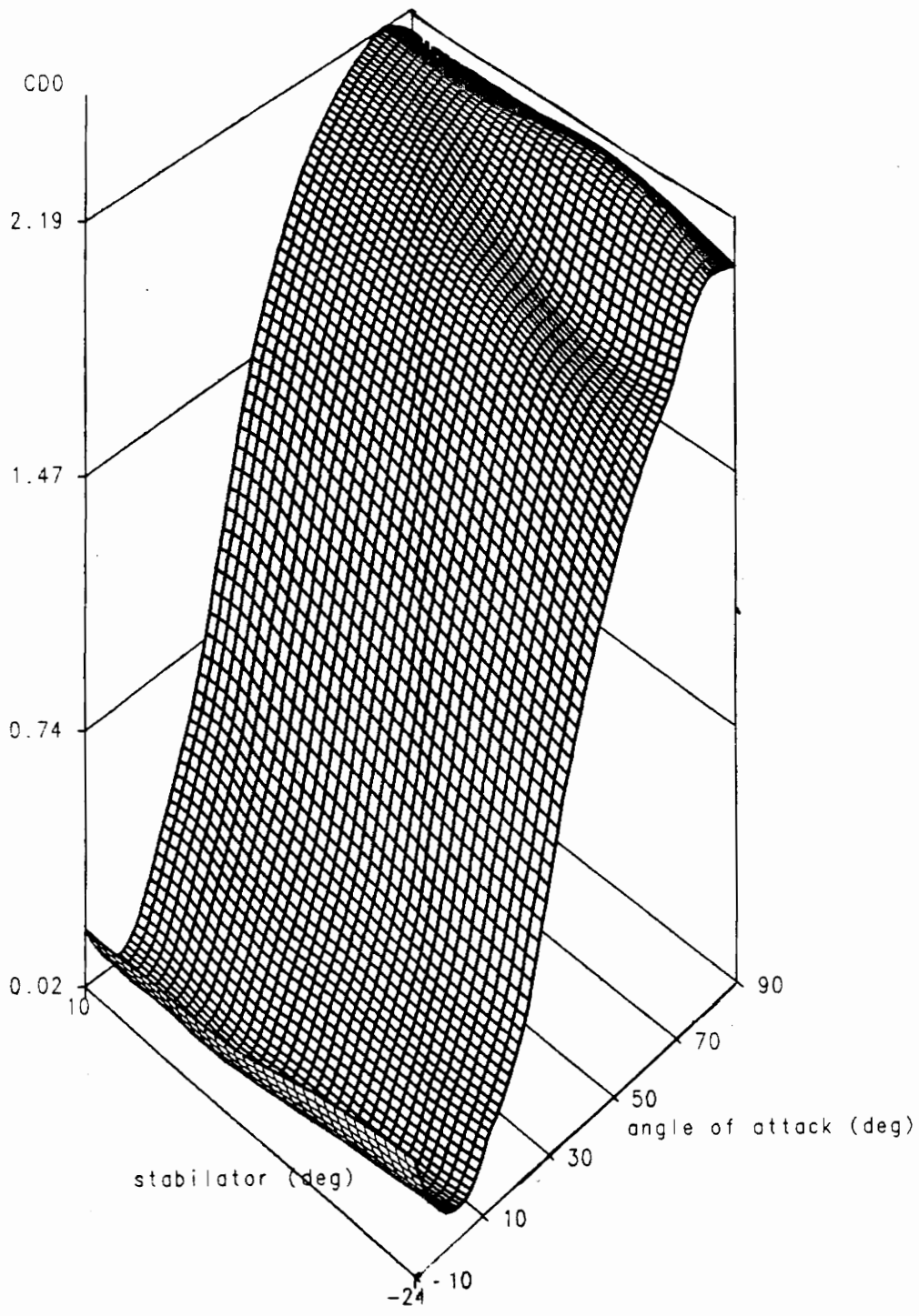


Figure 3. Drag Coefficient (h = 15000 ft.)

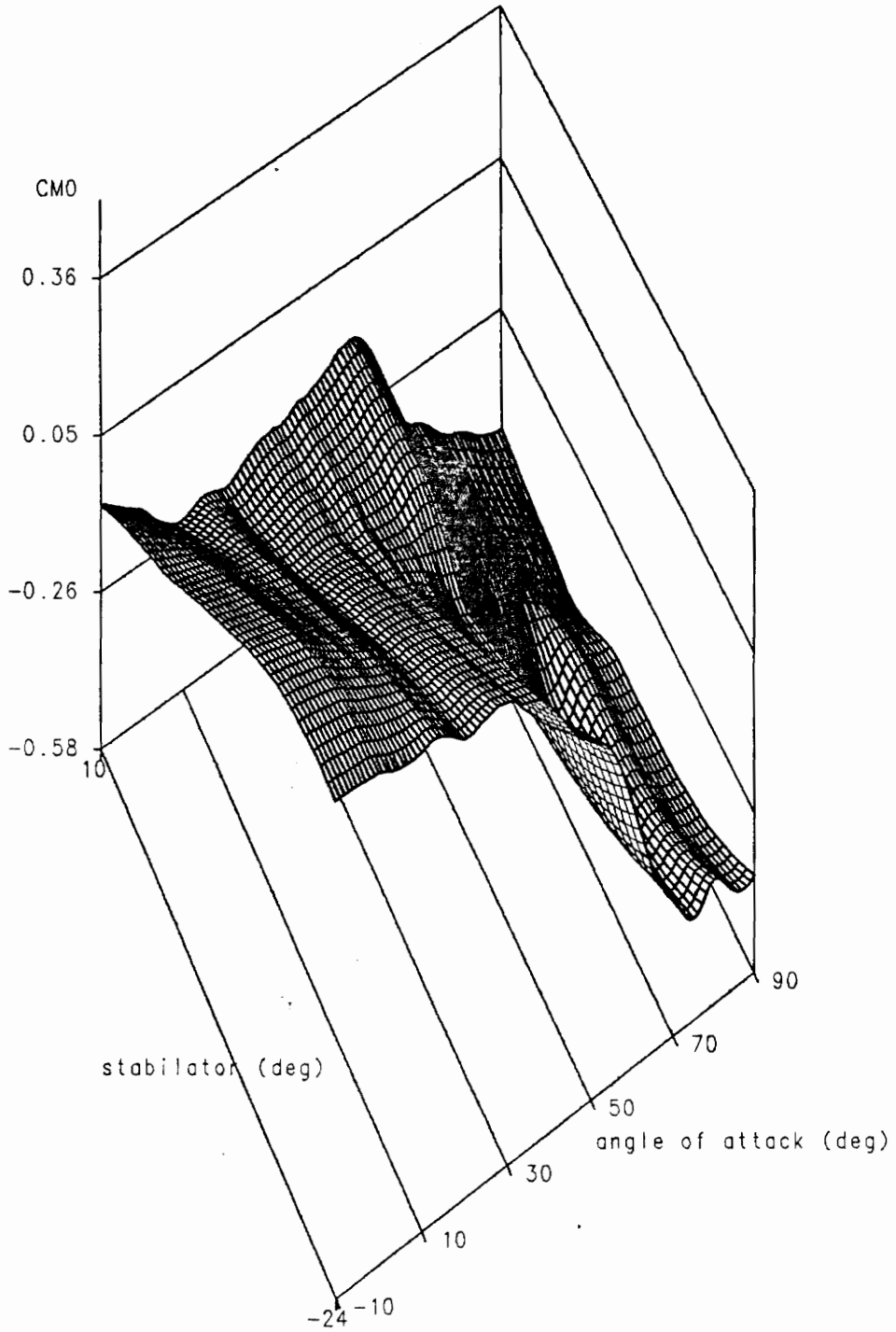


Figure 4. Pitching Moment Coefficient (h = 15000 ft.)

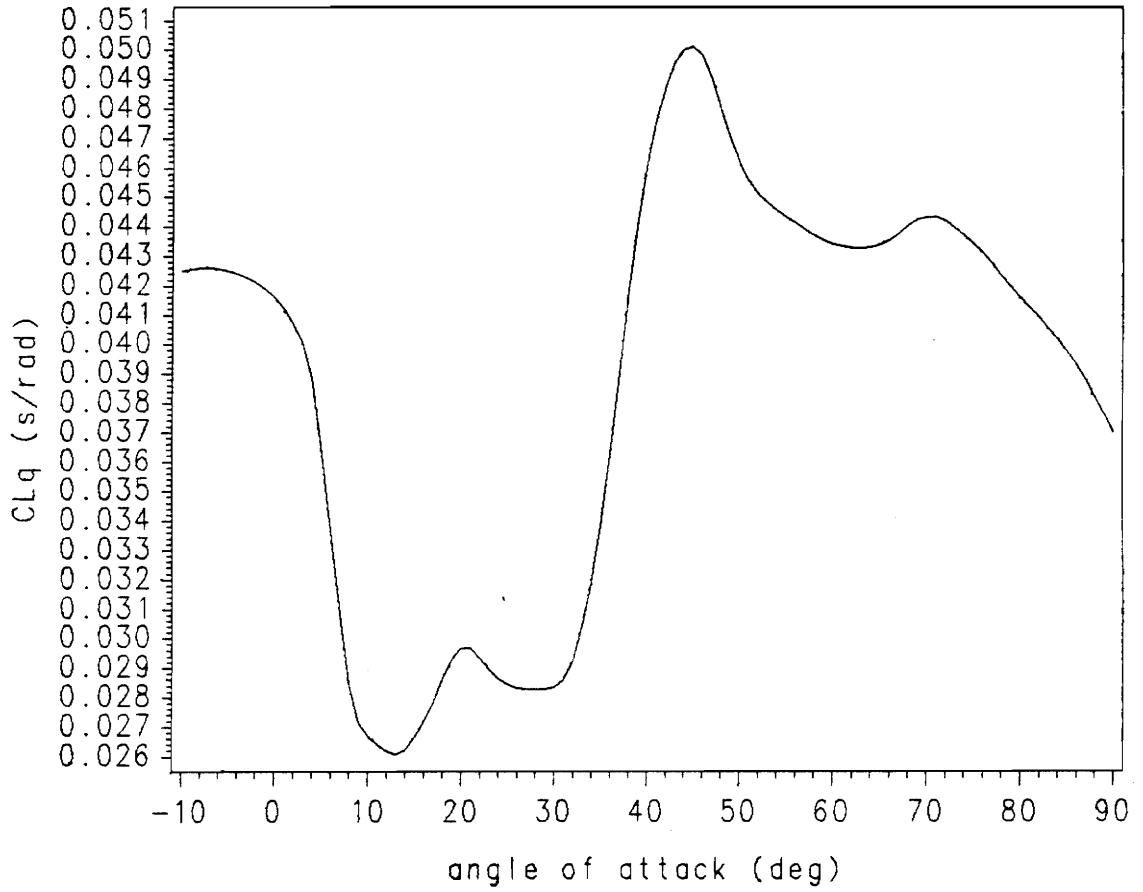


Figure 5. Pitch Rate Derivative of Lift Coefficient (h = 15000 ft.)

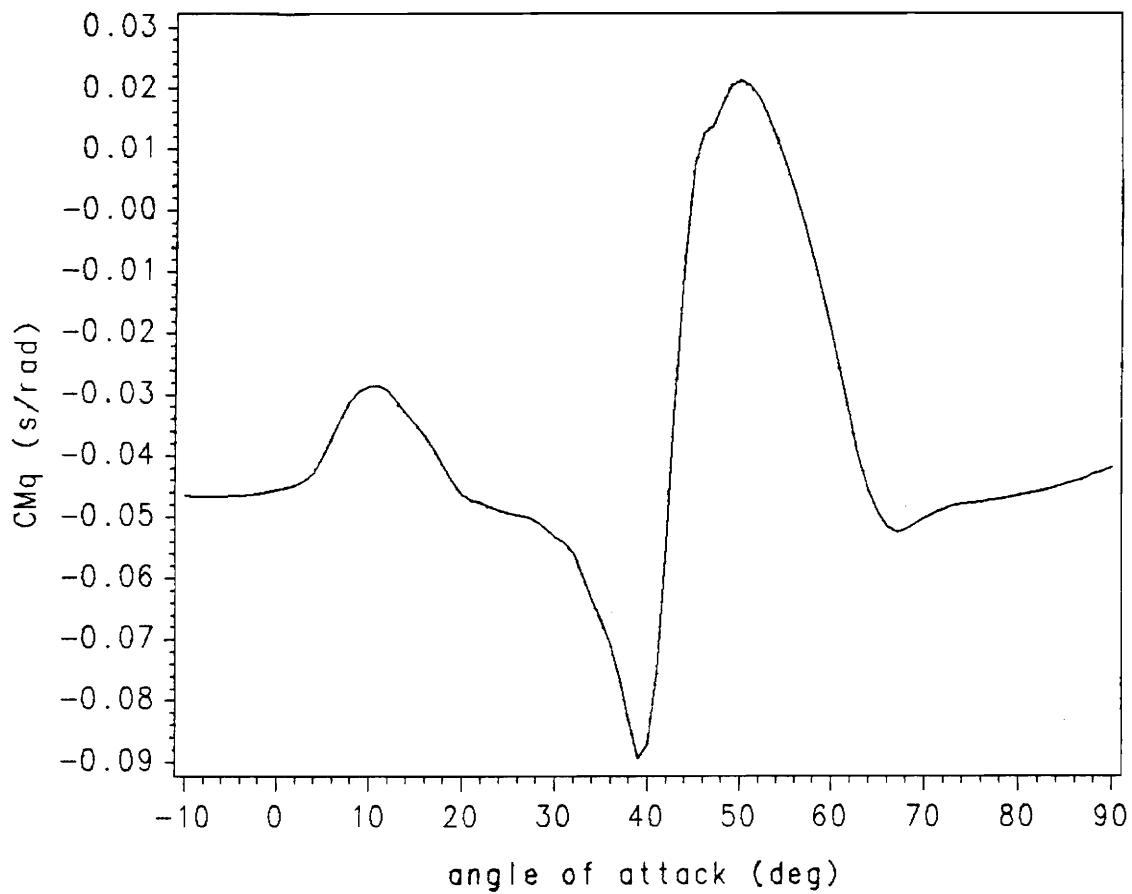


Figure 6. Pitch Rate Derivative of Pitching Moment Coefficient (h = 15000 ft.)

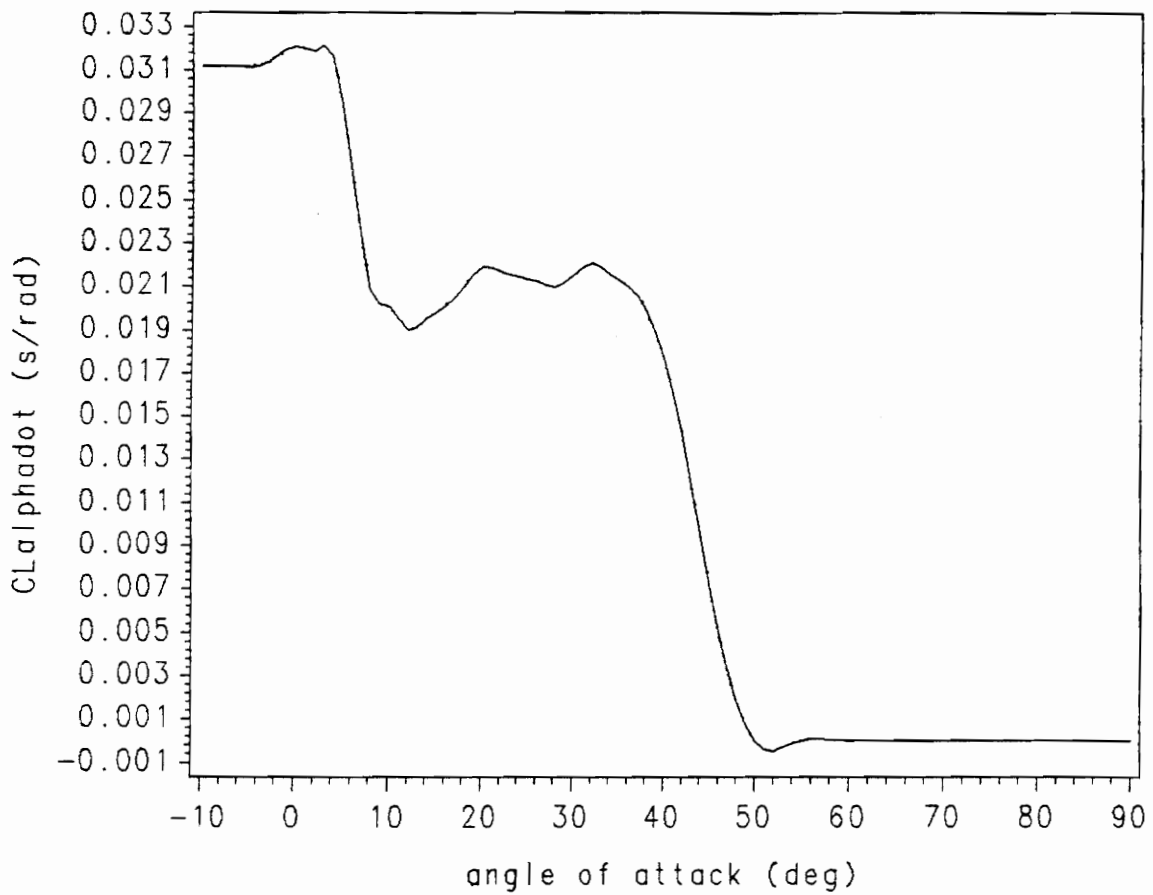


Figure 7. Angle of Attack Rate Derivative of Lift Coefficient (h = 15000 ft.)

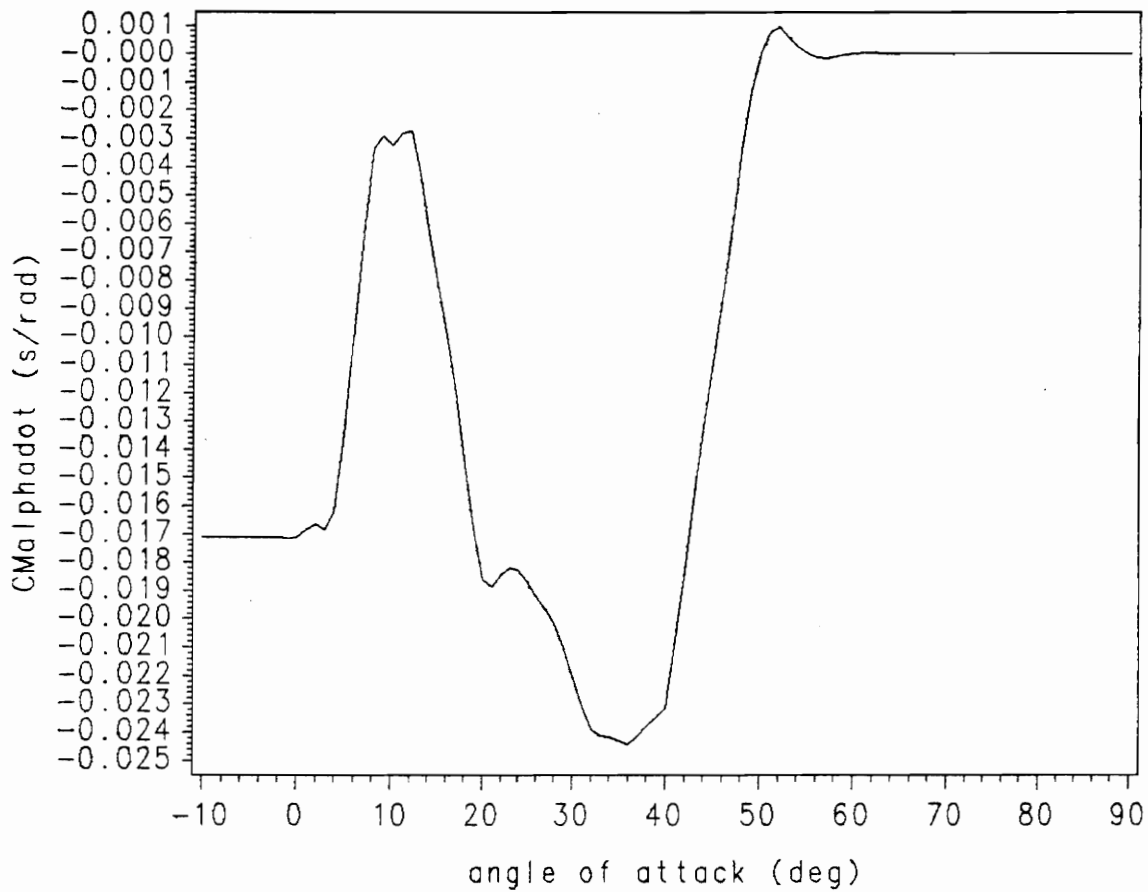


Figure 8. Angle of Attack Rate Derivative of Pitching Moment Coefficient ($h = 15000 \text{ ft.}$)

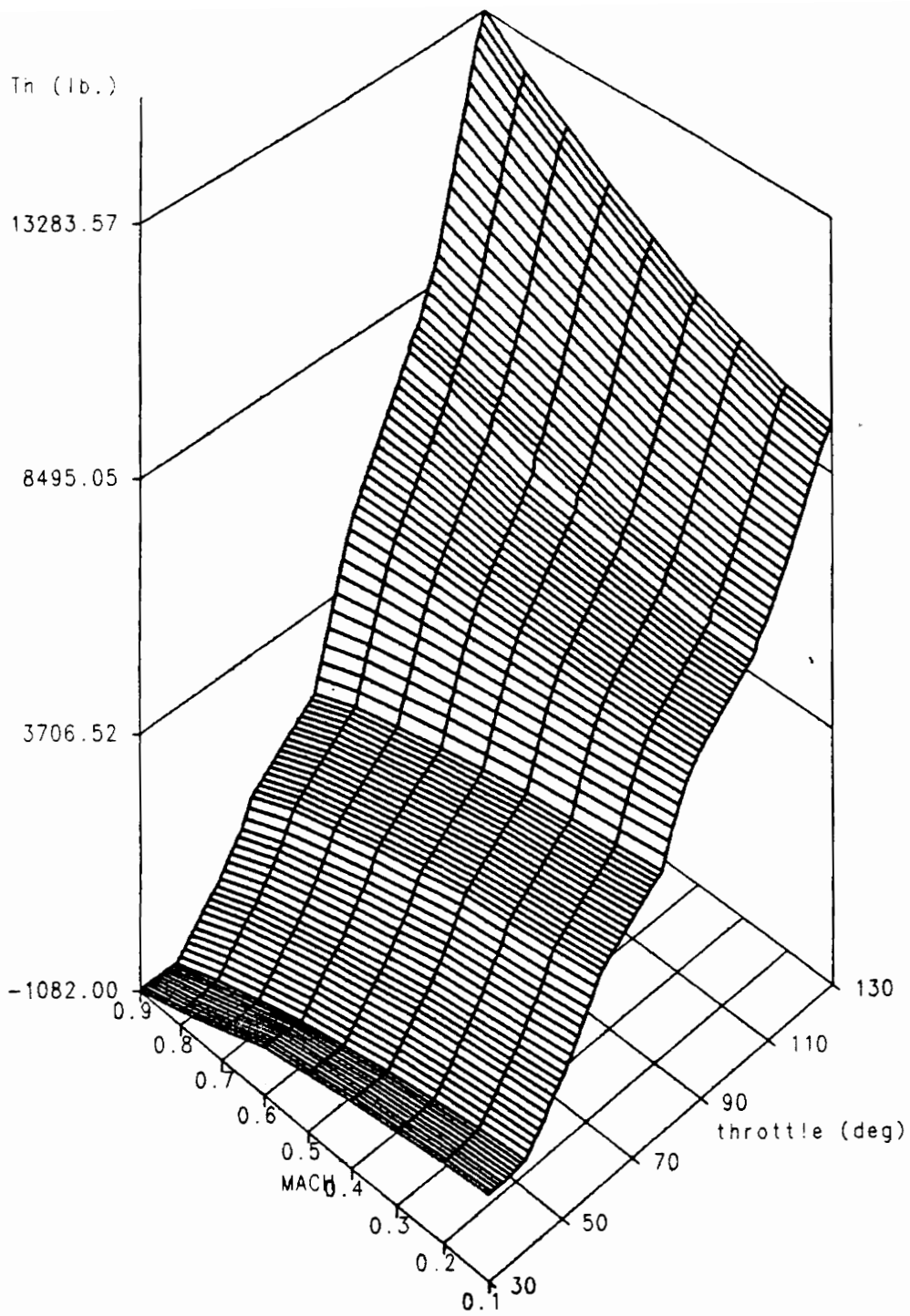


Figure 9. Thrust per Engine (h = 15000 ft.)

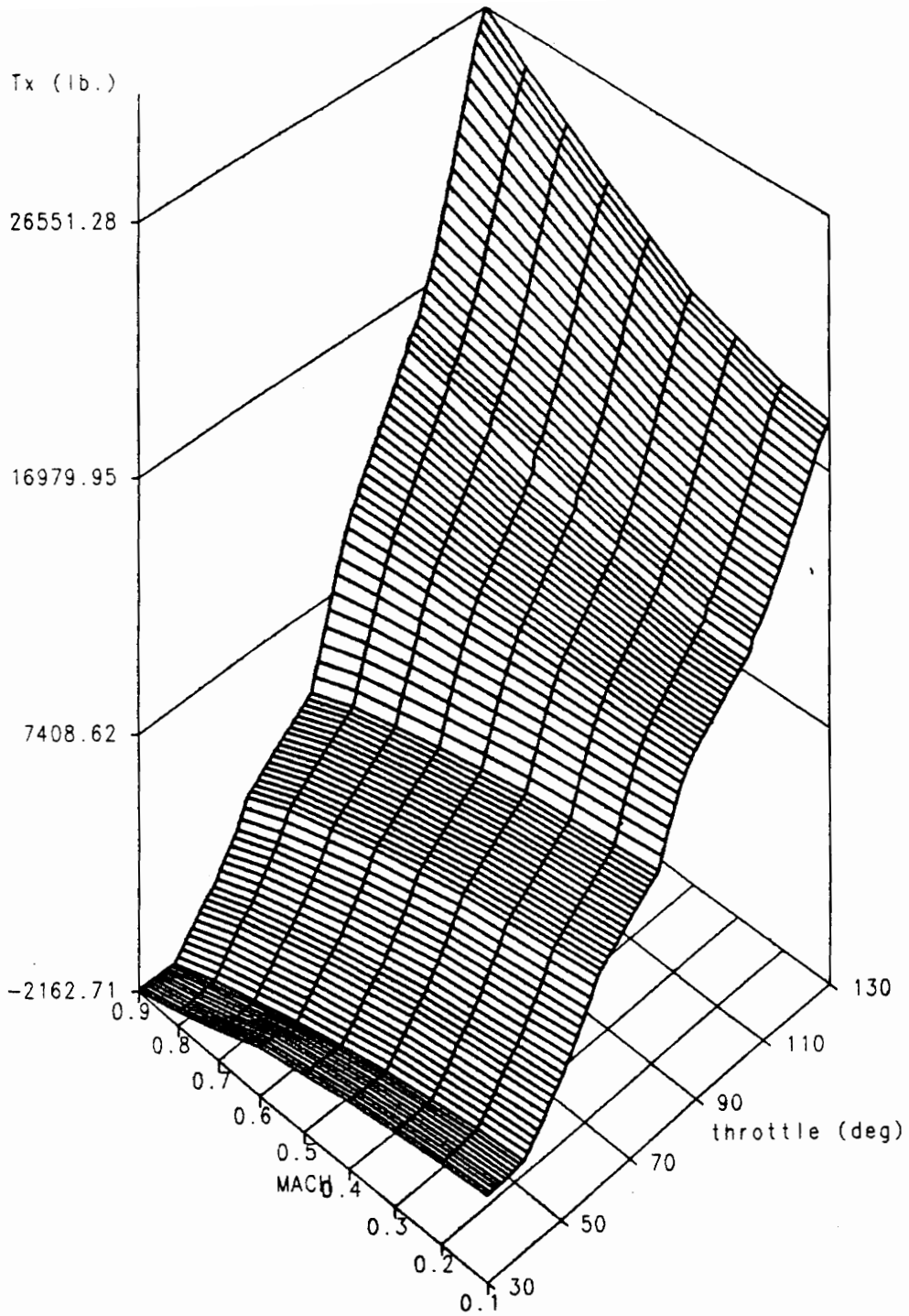


Figure 10. Thrust in x-body Direction (h = 15000 ft.)

5.0 Outer Layer

5.1 Analysis

The outer layer portion of the maneuver consists of the rotation of the velocity vector through 180° in flight path angle, assuming the angle of attack and the pitch rate are fast variables compared to Mach number, flight path angle, and altitude. By this assumption, the equations of motion become Eq. (5.1) through Eq. (5.5) for small values of ε .

$$\varepsilon \dot{\alpha} = q + \frac{g}{V} \cos \gamma - \frac{\bar{q}S}{mV} c_L - \frac{T_x}{mV} \sin \alpha \quad (5.1)$$

$$\dot{V} = -g \sin \gamma - \frac{\bar{q}S}{m} C_D + \frac{T_x}{m} \cos \alpha \quad (5.2)$$

$$\varepsilon \dot{q} = \frac{1}{I_y} [l_z T_x + \bar{q}S(\bar{c}C_m + d_L C_L + d_D C_D)] \quad (5.3)$$

$$\dot{\theta} = q \quad (5.4a)$$

$$\dot{\gamma} = q - \dot{\alpha} \quad (5.4b)$$

$$\dot{h} = V \sin \gamma \quad (5.5)$$

We seek to minimize the time required to execute the outer layer. The outer layer performance index J_{OL} is given by

$$J_{OL} = t_f^{OL} - t_0^{OL} = \int_{t_0^{OL}}^{t_f^{OL}} dt \quad (5.6)$$

$$\gamma = \theta - \alpha \quad (5.7)$$

$$\dot{\gamma} = \dot{\theta} - \dot{\alpha} \quad (5.8)$$

Therefore, for nearly constant angle of attack, $\dot{\alpha} = 0$

$$\dot{\gamma} = \frac{d\gamma}{dt} = \dot{\theta} = q \quad (5.9)$$

Therefore,

$$dt = \frac{d\gamma}{q} \quad (5.10)$$

$$J_{OL} = \int_{\gamma(t_0^{OL})}^{\gamma(t_f^{OL})} \frac{d\gamma}{q} \quad (5.11)$$

This equation indicates that the integrand of the outer layer performance index is minimized when the instantaneous pitch rate is maximized. Therefore, the values of the angle of attack and the stabilator deflection for the outer layer are those which maximize pitch rate subject to Eq. (5.1) - (5.5) as ε approaches zero.

$$0 = q + \frac{g}{V} \cos \gamma - \frac{\bar{q}S}{mV} - \frac{T_x}{mV} \sin \alpha \quad (5.12)$$

$$\dot{V} = -g \sin \gamma - \frac{\bar{q}S}{m} C_D + \frac{T_x}{m} \cos \alpha \quad (5.13)$$

$$0 = \frac{1}{I_y} [l_z T_x + \bar{q}S(\bar{c}C_m + d_L C_L + d_D C_D)] \quad (5.14)$$

$$\dot{\theta} = q \quad (5.15)$$

$$\dot{h} = V \sin \gamma \quad (5.16)$$

Therefore by maximizing the pitch rate in Eq. (5.12) and Eq. (5.14), a feedback control law is derived as a function of the slow states which are flight path angle, Mach number and altitude. For a given altitude, this control law consists of a surface defined by the stabilator deflection for each flight path angle and Mach number. Surfaces for the fast variables that maximize pitch rate are determined as well.

$$q^*(\gamma, M, h) = \max_{\alpha, \delta_h} q(\alpha, \delta_h, \gamma, M, h) \quad (5.17)$$

$\alpha^*(\gamma, M, h)$ and $\delta_h^*(\gamma, M, h)$ are those arguments in Eq. (5.17) which maximize pitch rate subject to equations (5.12) and (5.14). Our numerical procedure generates unique maximizing arguments.

In order to find these surfaces, parameter optimization is used. This is done using equations (5.12), (5.14) and (5.17). First, the aerodynamic coefficients are rewritten in a form which explicitly shows their dependence on pitch rate.

$$C_L(\alpha, \delta_h, M, q) = C_{L_0}(\alpha, \delta_h, M) + C_{L_q}(\alpha, M)q \quad (5.18)$$

$$C_m(\alpha, \delta_h, M, q) = C_{m_0}(\alpha, \delta_h, M) + C_{m_q}(\alpha, M)q \quad (5.19)$$

$$C_D(\alpha, \delta_h, M) = C_{D_0}(\alpha, \delta_h, M) \quad (5.20)$$

Equation (5.12) becomes

$$0 = q + \frac{g}{V} \cos \gamma - \frac{\bar{q}S}{mV} (C_{L_0} + C_{L_q}q) - \frac{T_x}{mV} \sin \alpha \quad (5.21)$$

and equation (5.14) becomes

$$\begin{aligned} 0 &= \frac{1}{I_y} [l_z T_x + \bar{q}S[\bar{c}(C_{m_0} + C_{m_q}q) + d_L(C_{L_0} + C_{L_q}q) + d_D C_{D_0}]] \\ &= f(\alpha, \delta_h, \gamma, M, h, q) \end{aligned} \quad (5.22)$$

Then, solving equation (5.21) for q yields

$$q(\alpha, \delta_h, \gamma, M, h) = \frac{-\frac{g}{V} \cos \gamma + \frac{\bar{q}S}{mV} C_{L_0} + \frac{T_x}{mV} \sin \alpha}{1 - \frac{\bar{q}S}{mV} C_{L_q}} \quad (5.23)$$

For a given flight path angle, Mach number and altitude, the problem is now to maximize equation (5.23) subject to equation (5.22) over the domain (α, δ_h) .

$$\max_{\alpha, \delta_h} q(\alpha, \delta_h, \gamma, M, h)$$

$$\text{subject to } f(\alpha, \delta_h, \gamma, M, h, q) = 0$$

or

$$\begin{aligned} & \min_{\alpha, \delta_h} -q(\alpha, \delta_h, \gamma, M, h) \\ & \text{subject to } f(\alpha, \delta_h, \gamma, M, h, q) = 0 \end{aligned}$$

This problem is solved by defining a Hamiltonian

$$\mathcal{H}(\alpha, \delta_h, \gamma, M, h) = -q(\alpha, \delta_h, \gamma, M, h) + \lambda f(\alpha, \delta_h, \gamma, M, h, q) \quad (5.24)$$

And applying the necessary conditions for a stationary point on $-q$:

$$f(\alpha^*, \delta_h^*, q^*, \gamma, M, h) = 0 \quad (5.25)$$

$$\frac{d\mathcal{H}}{d\alpha}(\alpha^*, \delta_h^*, q^*, \gamma, M, h) = 0 \quad (5.26)$$

$$\frac{d\mathcal{H}}{d\delta_h}(\alpha^*, \delta_h^*, q^*, \gamma, M, h) = 0 \quad (5.27)$$

substituting equation (5.24) into equations (5.26) and (5.27) it follows that:

$$f(\alpha^*, \delta_h^*, q^*, \gamma, M, h) = 0 \quad (5.28)$$

$$-\frac{dq}{d\alpha}(\alpha^*, \delta_h^*, q^*, \gamma, M, h) + \lambda \frac{df}{d\alpha}(\alpha^*, \delta_h^*, q^*, \gamma, M, h) = 0 \quad (5.29)$$

$$-\frac{dq}{d\delta_h}(\alpha^*, \delta_h^*, q^*, \gamma, M, h) + \lambda \frac{df}{d\delta_h}(\alpha^*, \delta_h^*, q^*, \gamma, M, h) = 0 \quad (5.30)$$

where:

$$\frac{dq}{d\alpha} = \frac{\frac{\bar{q}S}{mV} \frac{dC_{L_0}}{d\alpha} + \frac{T_x}{mV} \cos \alpha}{1 - \frac{\bar{q}S}{mV} C_{L_q}} + \frac{-\frac{g}{V} \cos \gamma + \frac{\bar{q}S}{mV} C_{L_0} + \frac{T_x}{mV} \sin \alpha}{\left(1 - \frac{\bar{q}S}{mV} C_{L_q}\right)^2} \left(\frac{\bar{q}S}{mV} \frac{dC_{L_q}}{d\alpha} \right) \quad (5.31)$$

$$\frac{dq}{d\delta_h} = \frac{\frac{\bar{q}S}{mV} \frac{dC_{L_0}}{d\delta_h}}{1 - \frac{\bar{q}S}{mV} C_{L_q}} \quad (5.32)$$

$$\frac{df}{d\alpha} = \frac{\bar{q}S}{I_y} \left[\bar{c} C_{m_0} + d_L \left(C_{d_0} + \frac{dC_{L_0}}{d\alpha} \right) + d_D \left(\frac{dC_{D_0}}{d\alpha} - C_{L_0} \right) + \left(\bar{c} \frac{dC_{m_q}}{d\alpha} + d_L \frac{dC_{L_q}}{d\alpha} - d_D C_{L_q} \right) q + \left(\bar{c} C_{m_q} + d_L C_{L_q} \right) \frac{dq}{d\alpha} \right] \quad (5.33)$$

$$\frac{df}{d\delta_h} = \frac{\bar{q}S}{I_y} \left[\bar{c} \frac{dC_{m_0}}{d\delta_h} + d_L \frac{dC_{L_0}}{d\delta_h} + d_D \frac{dC_{D_0}}{d\delta_h} \right] \quad (5.34)$$

The solution to this problem is a set of three surfaces for each altitude considered. These surfaces define the control setting (δ_h^*), the angle of attack (α^*) and the pitch rate (q^*), for a given flight path angle and Mach number, which minimize the outer layer performance index (J_{OL}).

5.2 Results

The problem, represented by equation (5.17) in Section 1, is solved numerically using the computer program "SLLSQP" (Sequential Linear Least Squares Programming) written by Dieter Kraft and Klaus Schittkowski (Ref. 10). We checked this solution for uniqueness and optimality using a global grid method. We selected an initial grid spanning angles of attack from zero (0) to ninety (90) degrees and stabilator angles from negative twenty-four (-24) to ten and one-half (10.5) degrees using one (1) degree intervals for angle of attack and one half (0.5) degree intervals for the stabilator. For a selection of flight path angles, Mach numbers and altitudes corresponding to Table 1, this grid of angles of attack and stabilator angles in the domain of the F/A-18 model was processed. The pitch rate corresponding to each node on this grid was calculated using equation (5.23). The node corresponding to the maximum pitch rate on this grid was found by sorting the pitch rates found at each node on the grid. This node became the center node on a finer grid which was generated as before. The node corresponding to the maximum pitch rate was again found by sorting. This process was repeated until the grid size was suitable for finding the solution to an acceptable number of significant digits. The angle of attack and stabilator angle corresponding to the node yielding the maximum pitch rate was compared to the solution using SLLSQP. We found that the result obtained using SLLSQP was indeed unique and provided the optimal pitch rate. The outer layer surfaces are generated for the Mach numbers, flight path angles and altitudes given in Table 1. We use nineteen (19) nodes for flight path angle, six (6) nodes for Mach number and six (6) nodes for altitude.

Table 1. Outer Layer Solution Nodes

$\gamma(\text{deg})$	M	$h(\bar{r}_i)$
0	0.2	5000
10	0.3	15000
20	0.4	25000
30	0.5	35000
40	0.6	45000
50	0.7	55000
70		
80		
90		
100		
110		
120		
130		
140		
150		
160		
170		
180		

The values corresponding to these nodes are stored in a file named "HALF_LOOP_OL.DAT". The feedback subroutine reads these data into memory upon initialization. A three dimensional linear interpolation is performed in order to calculate values not located on the nodes.

If initialization is done with outer layer values for the angle of attack and pitch rate, the outer layer portion of the maneuver can be simulated. Figures 29-31 show the results of the simulation for an initial Mach number of 0.6 and an altitude of 15,000 feet.

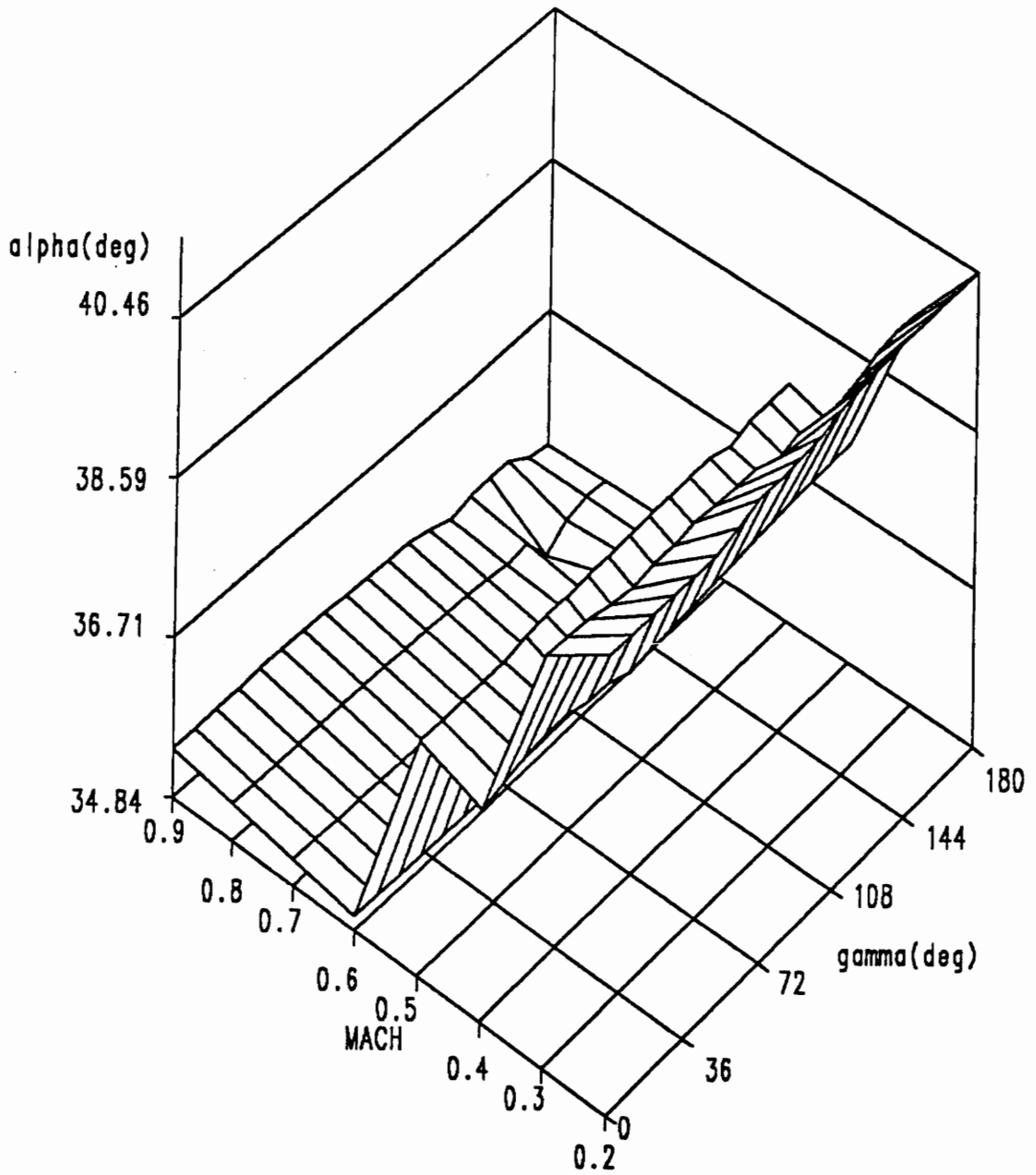


Figure 11. Outer Layer Angle of Attack ($h = 5000$ ft)

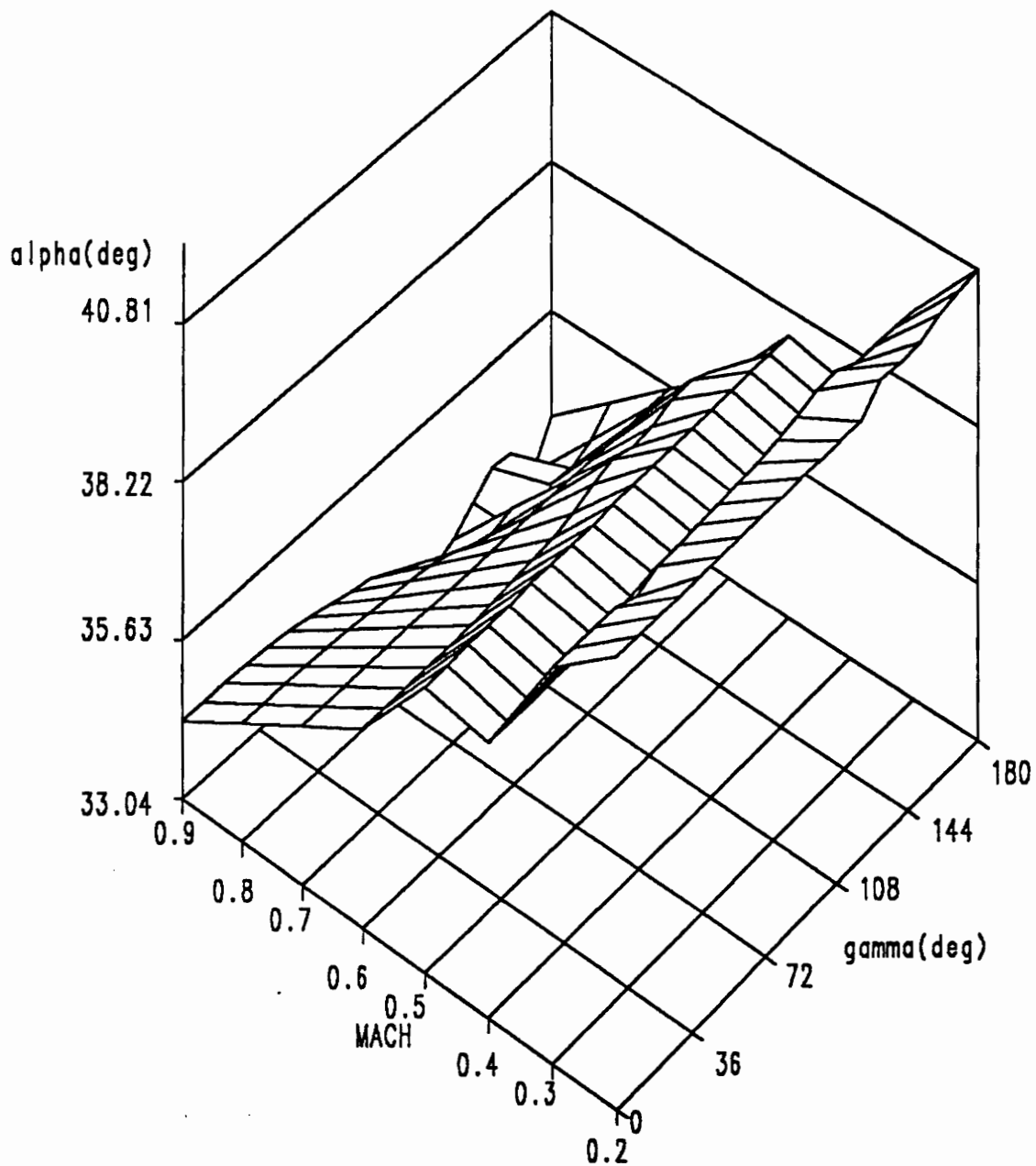


Figure 12. Outer Layer Angle of Attack ($h = 15000$ ft)

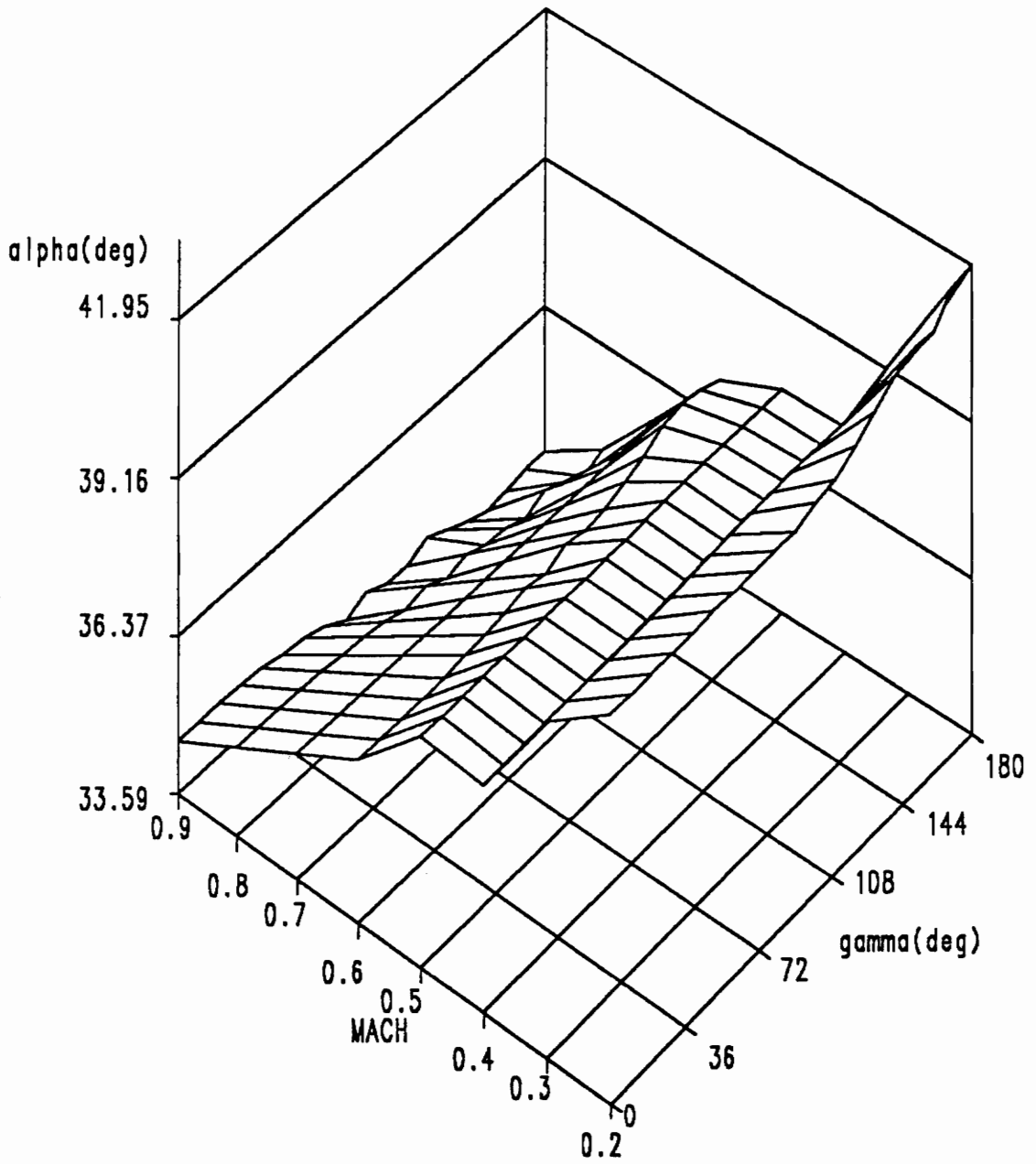


Figure 13. Outer Layer Angle of Attack ($h = 25000$ ft)

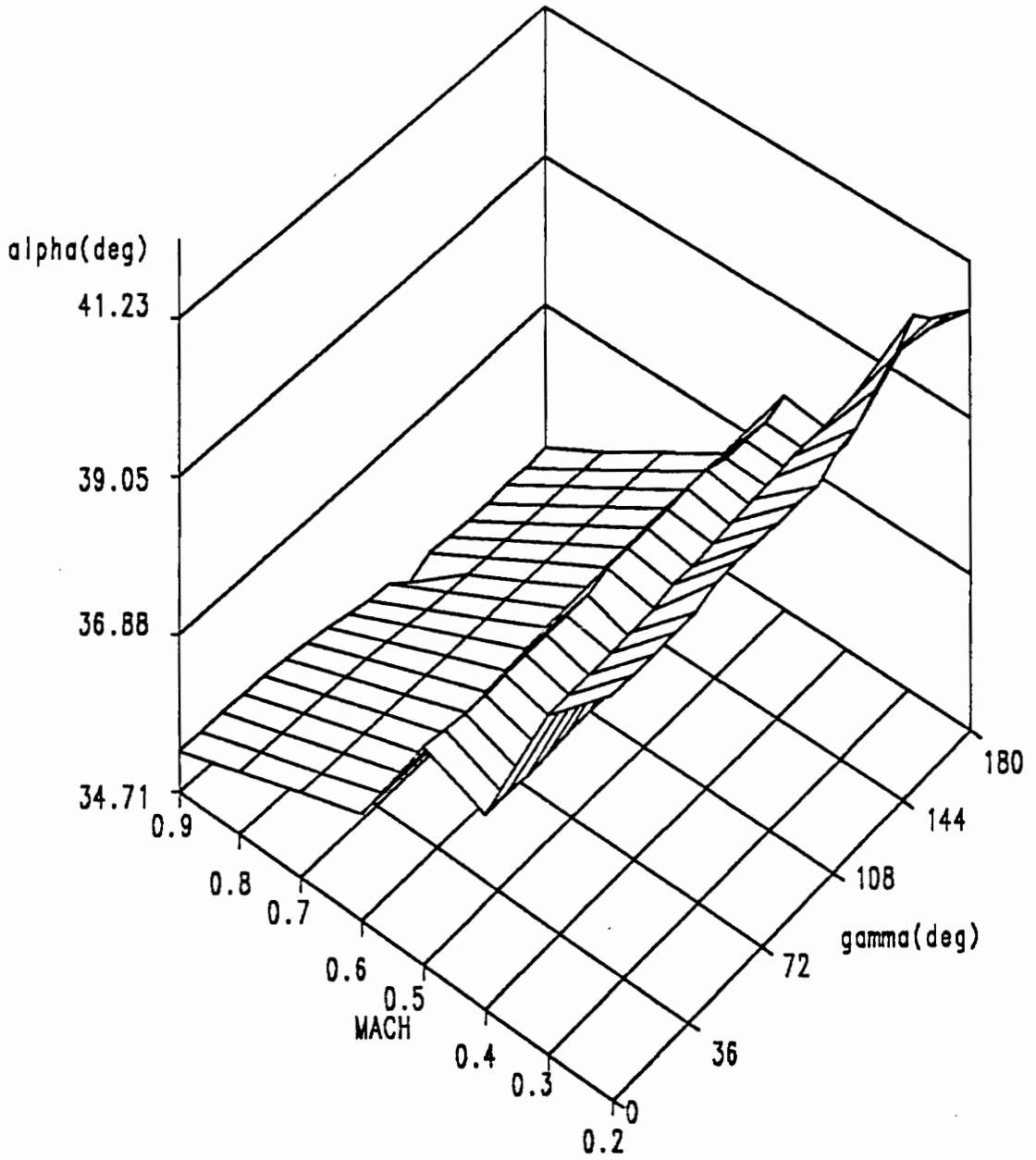


Figure 14. Outer Layer Angle of Attack (h = 35000 ft)

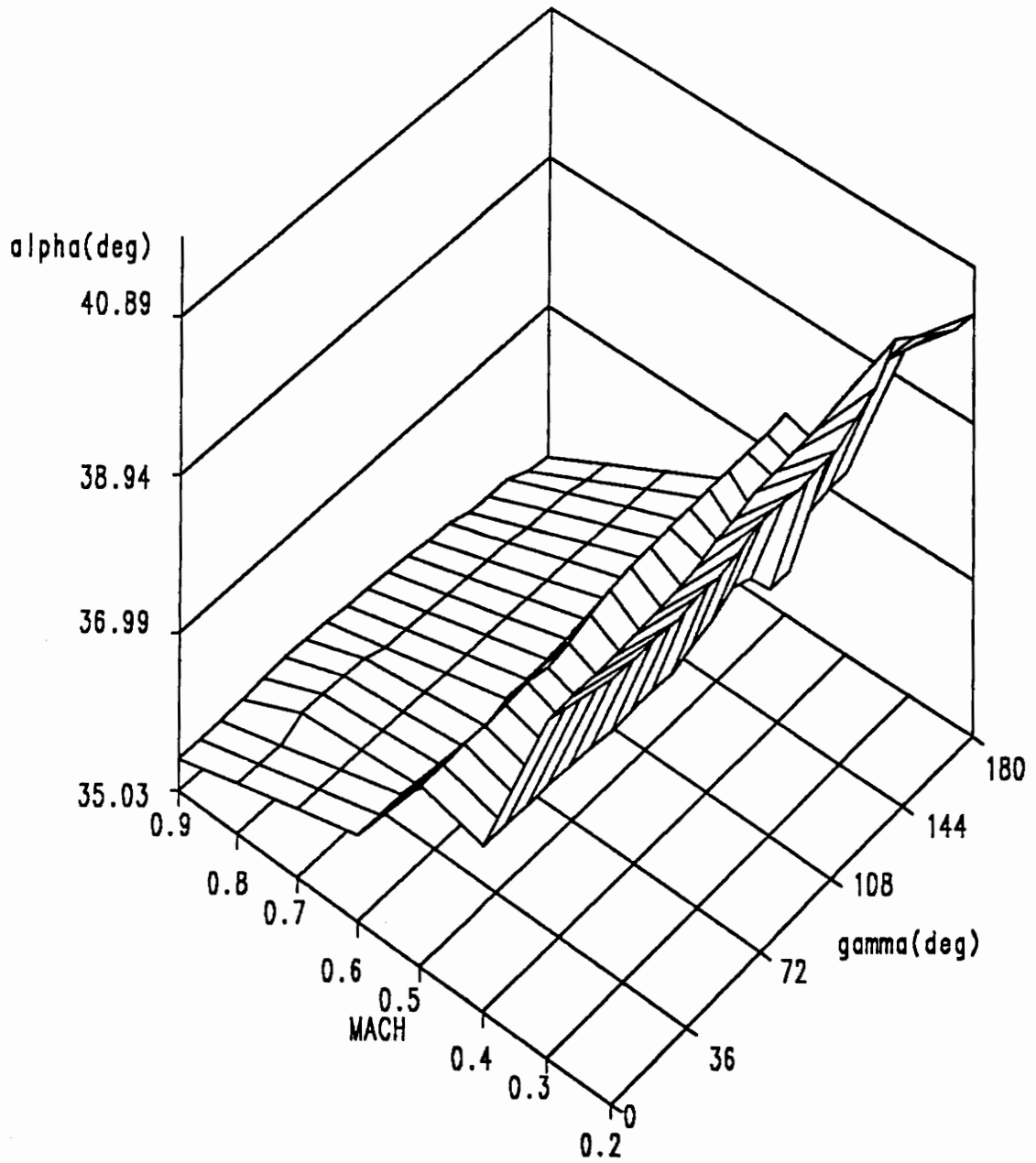


Figure 15. Outer Layer Angle of Attack ($h = 45000$ ft)

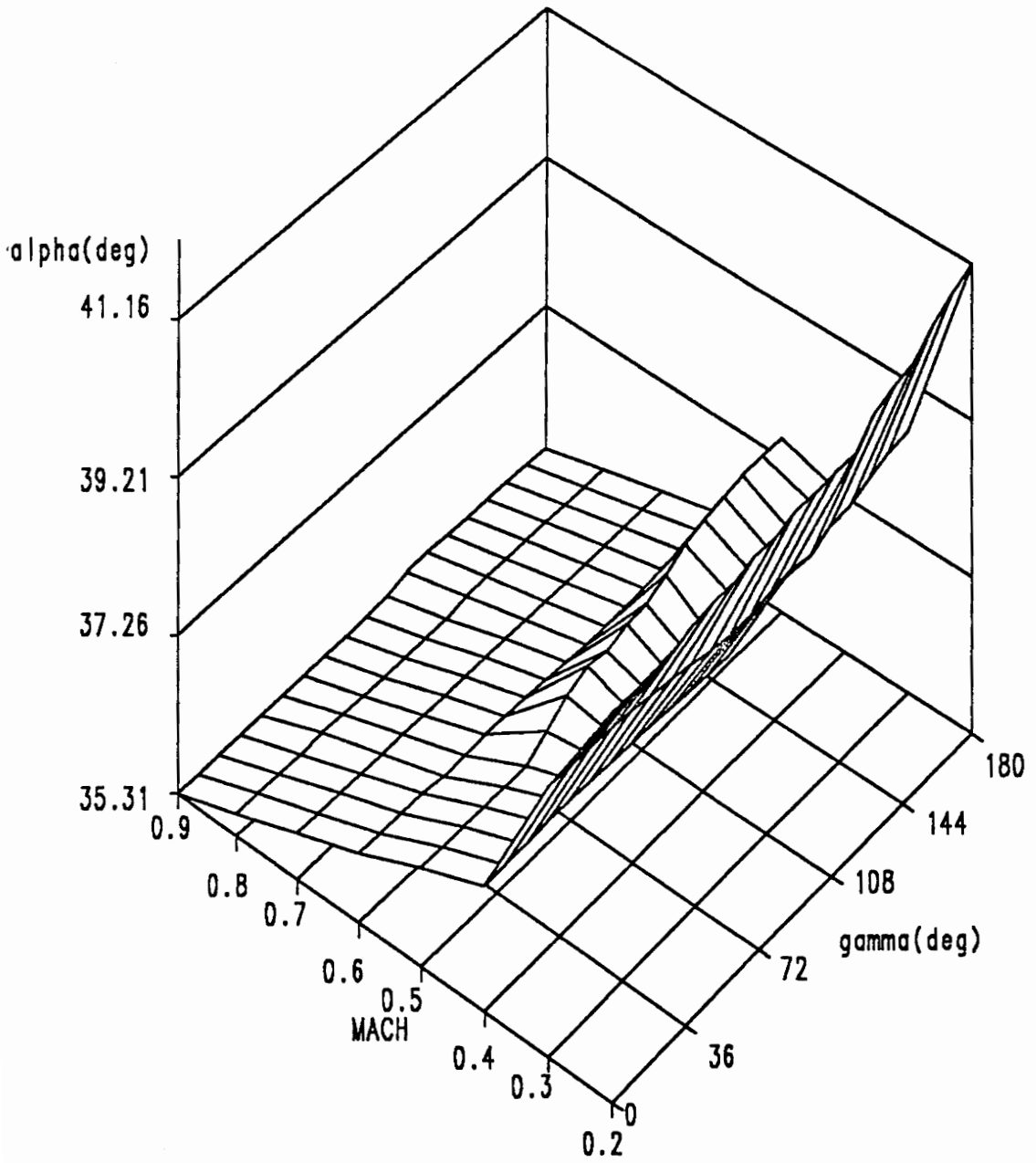


Figure 16. Outer Layer Angle of Attack (h = 55000 ft)

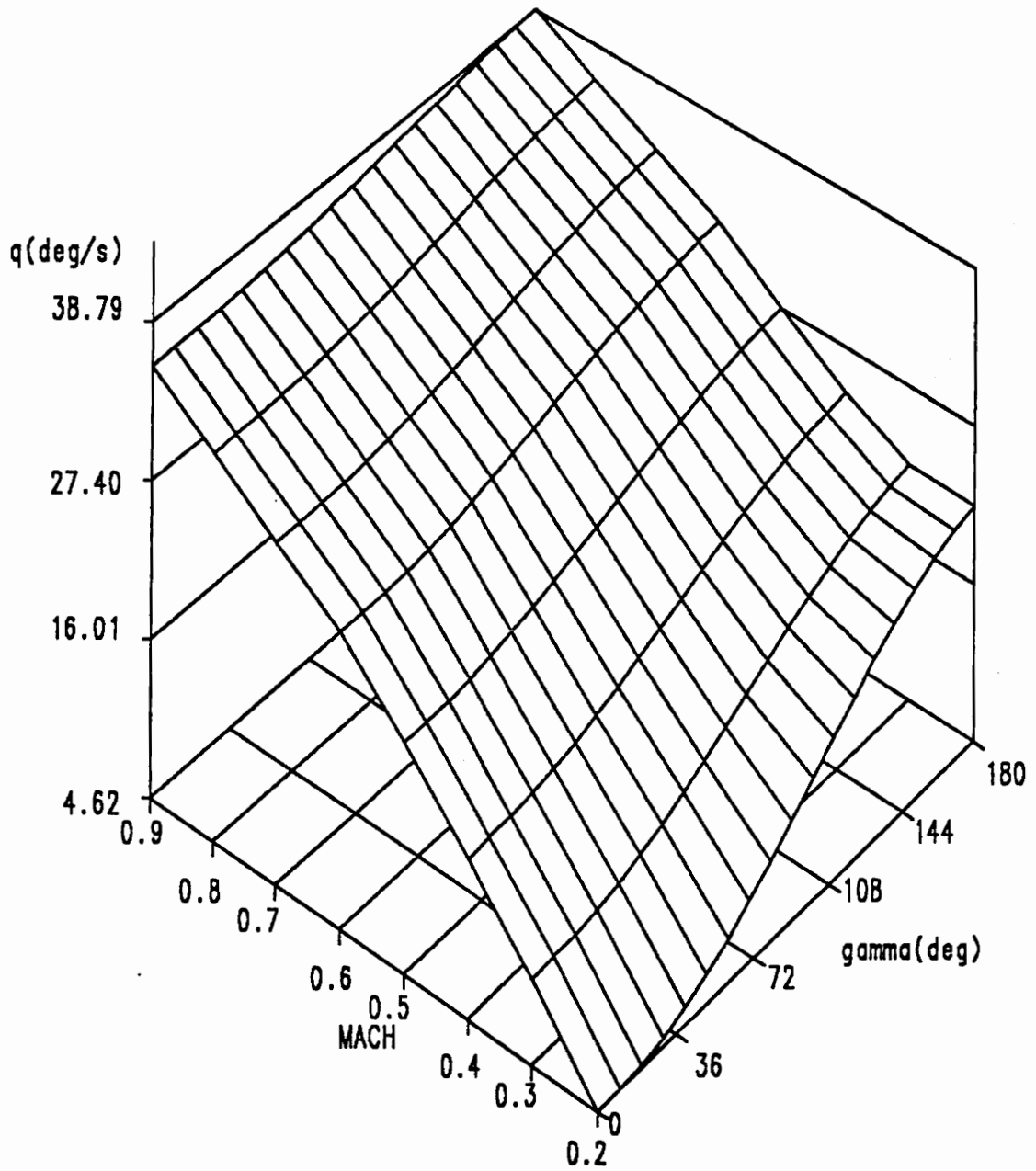


Figure 17. Outer Layer Pitch Rate ($h = 5000 \text{ ft}$)

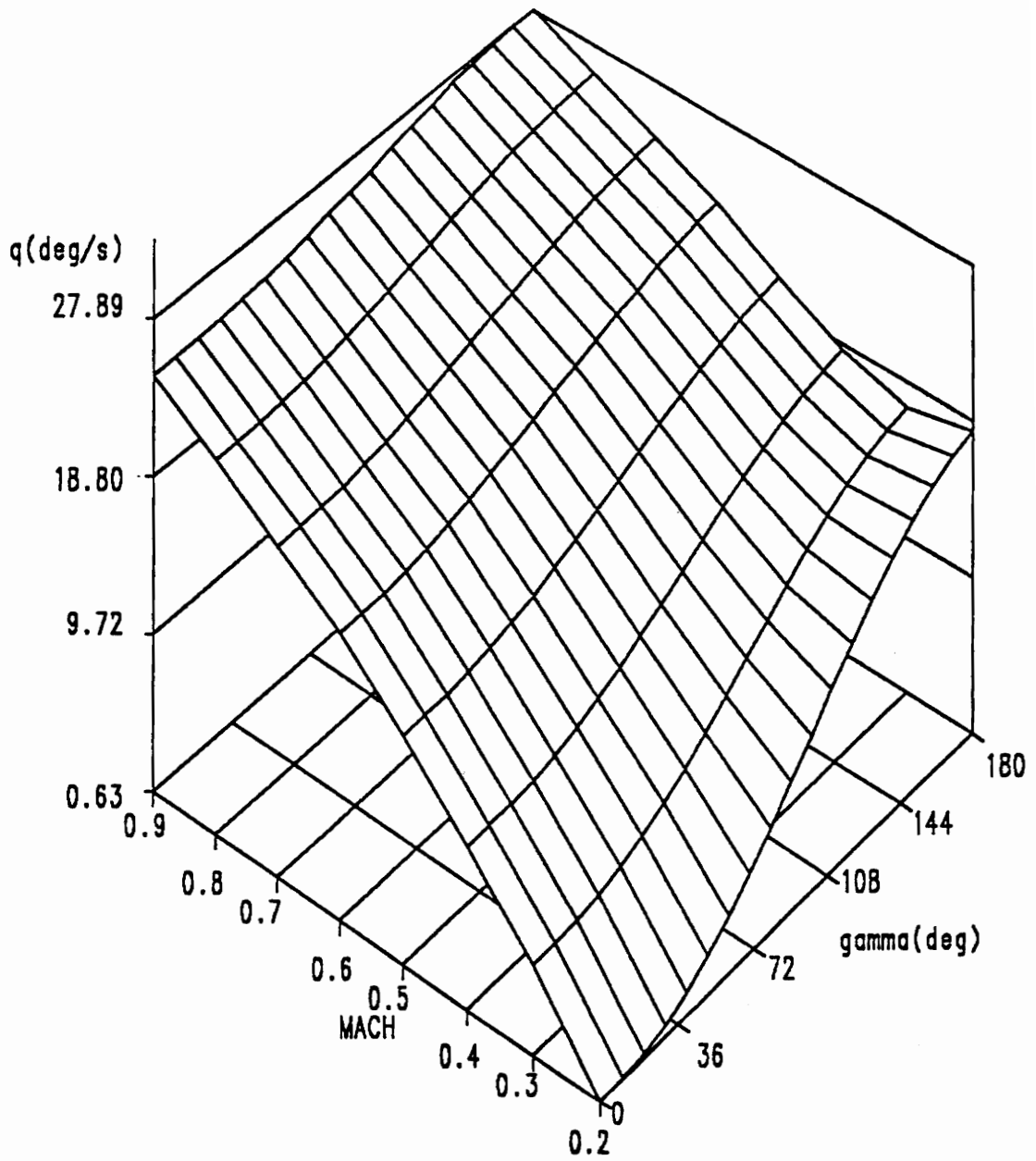


Figure 18. Outer Layer Pitch Rate ($h = 15000$ ft)

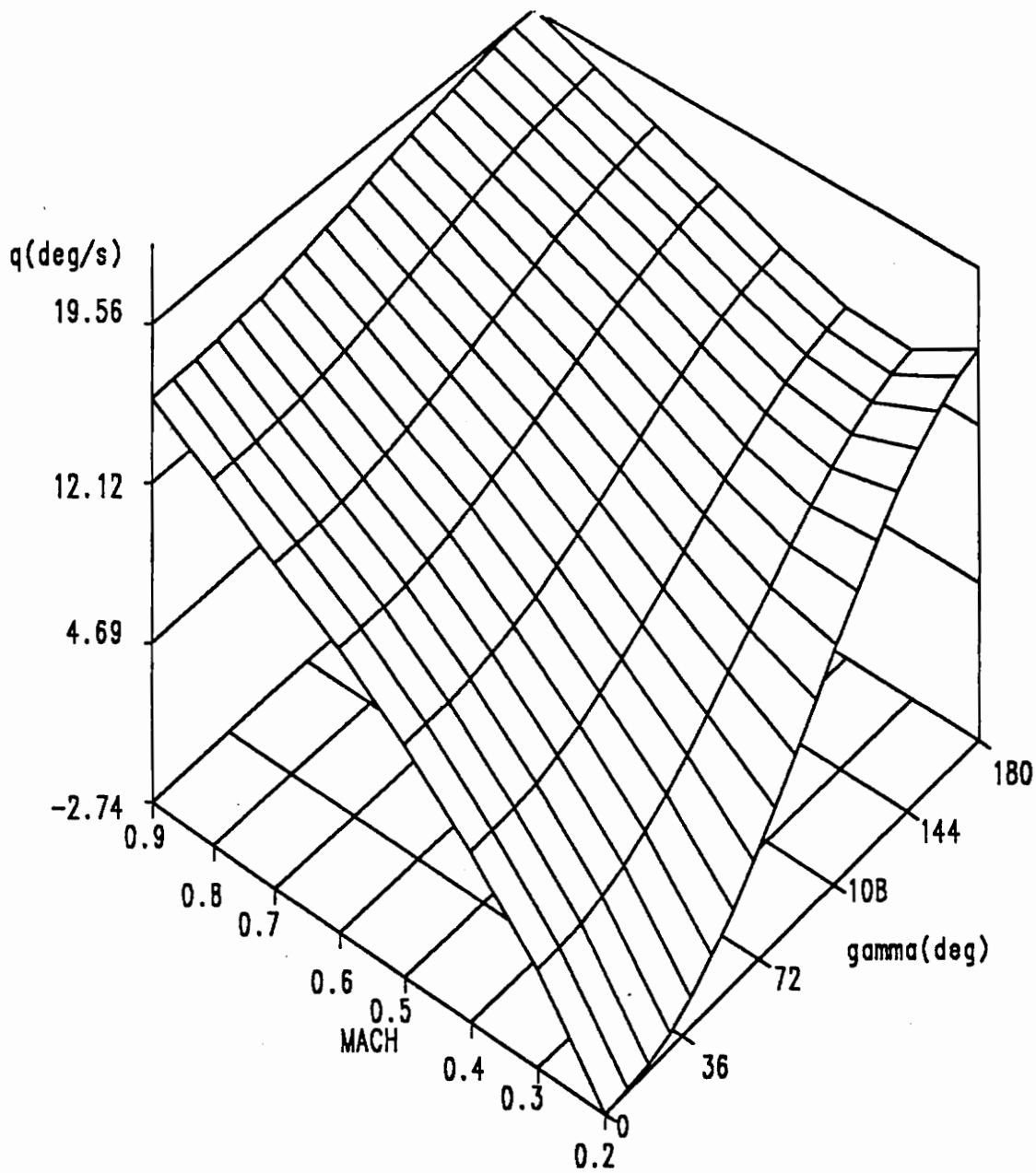


Figure 19. Outer Layer Pitch Rate ($h = 25000$ ft)

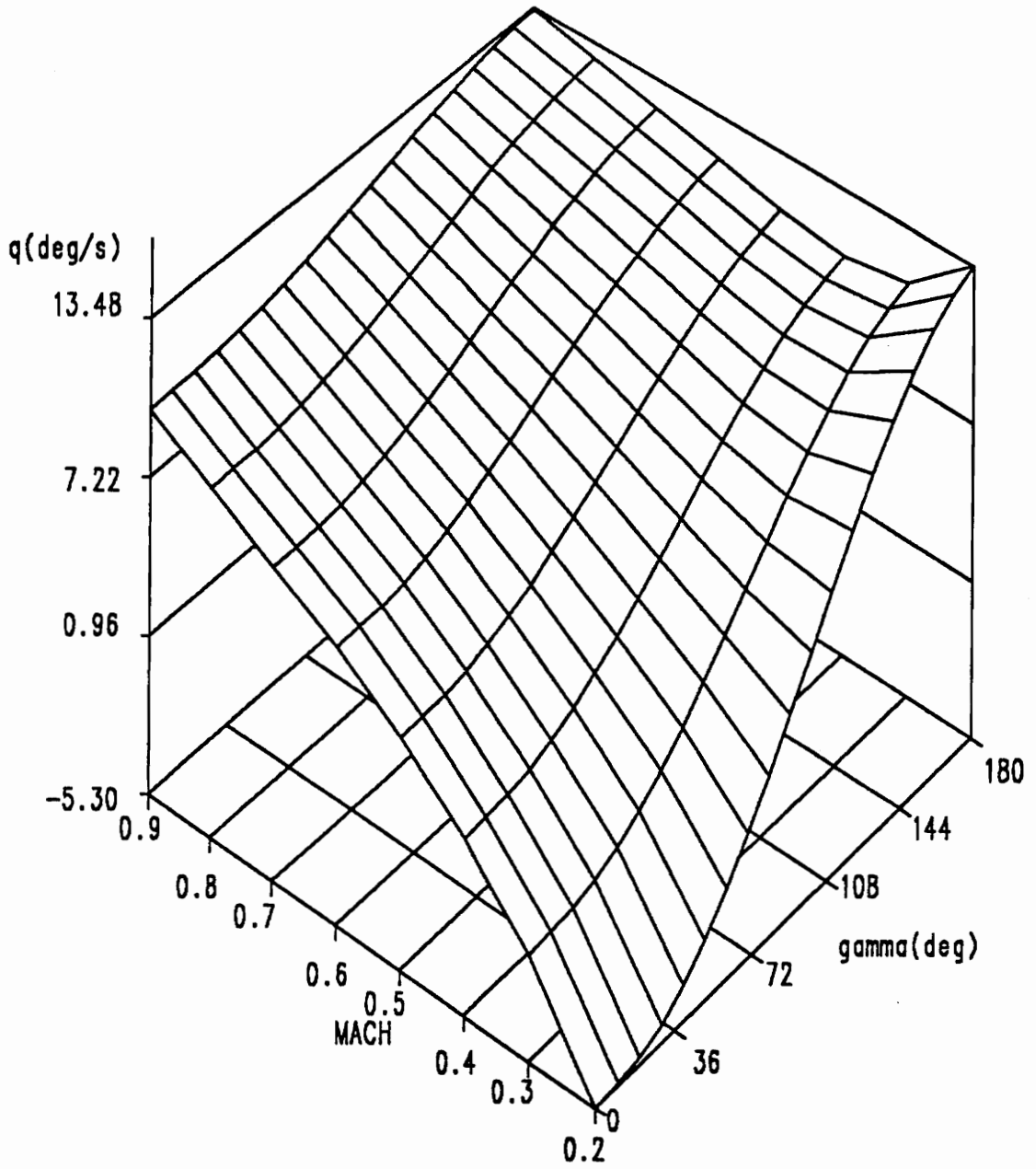


Figure 20. Outer Layer Pitch Rate ($h = 35000$ ft)

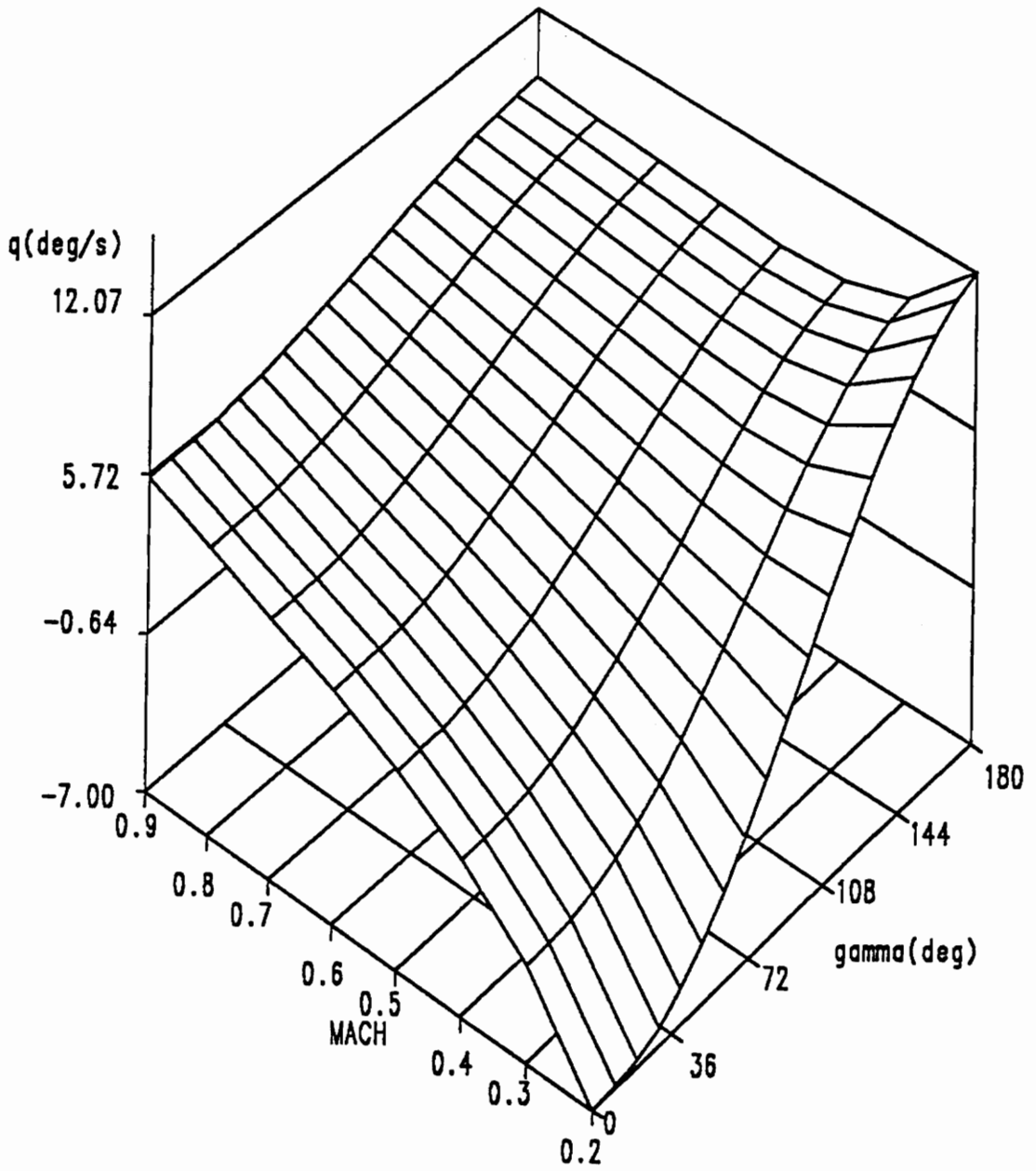


Figure 21. Outer Layer Pitch Rate ($h = 45000$ ft)

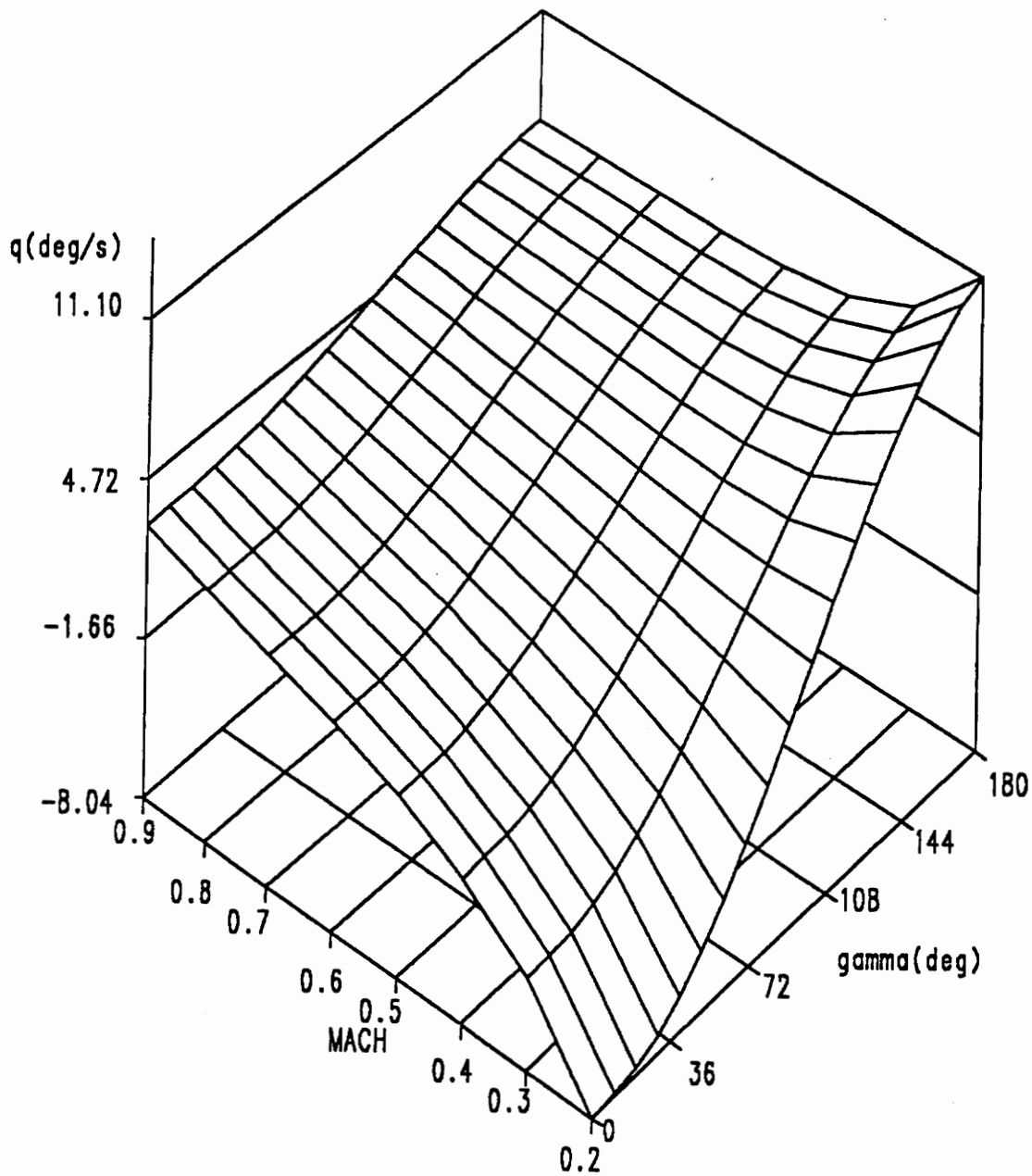


Figure 22. Outer Layer Pitch Rate ($h = 55000$ ft)

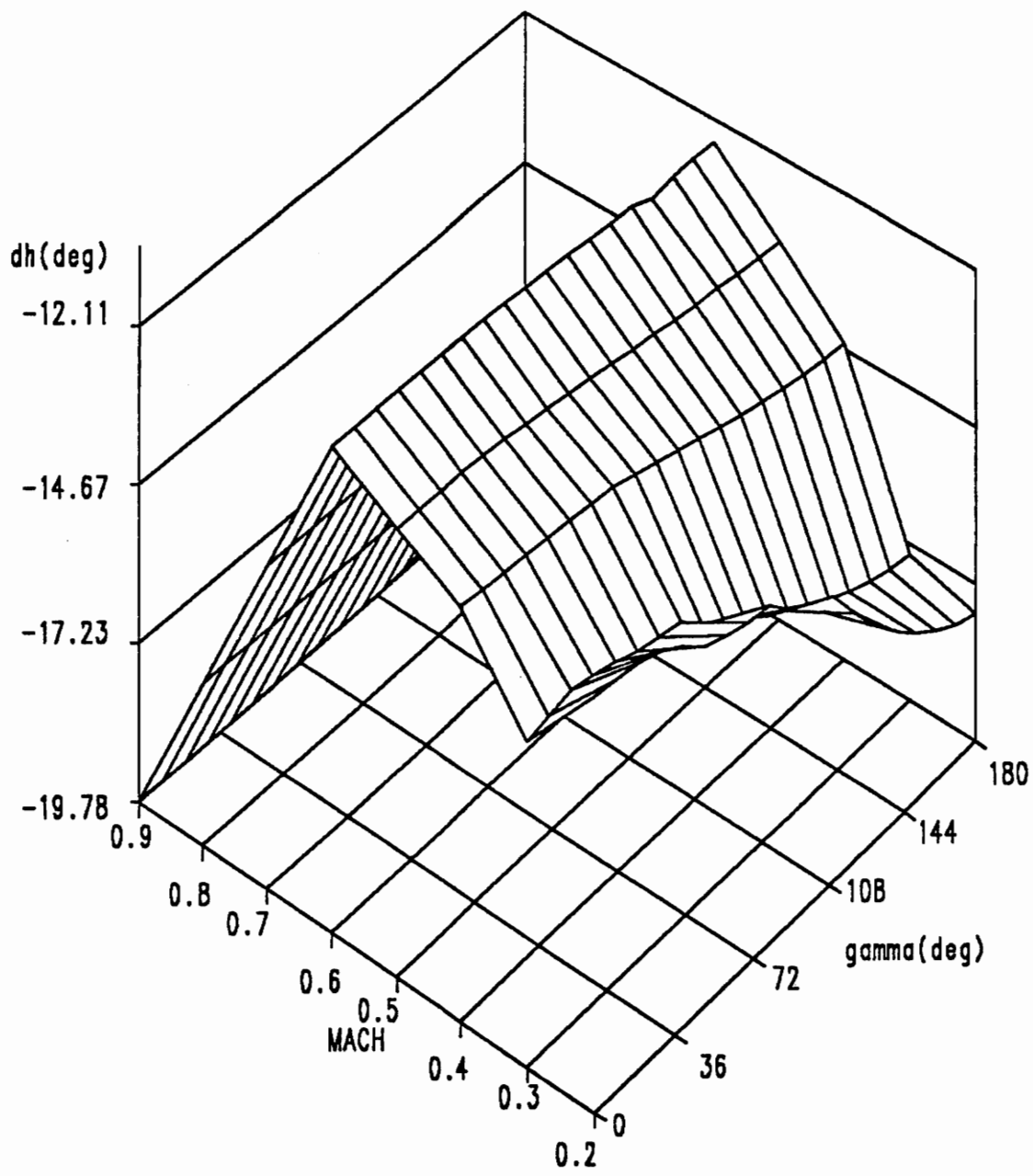


Figure 23. Outer Layer Stabilator Deflection (h = 5000 ft)

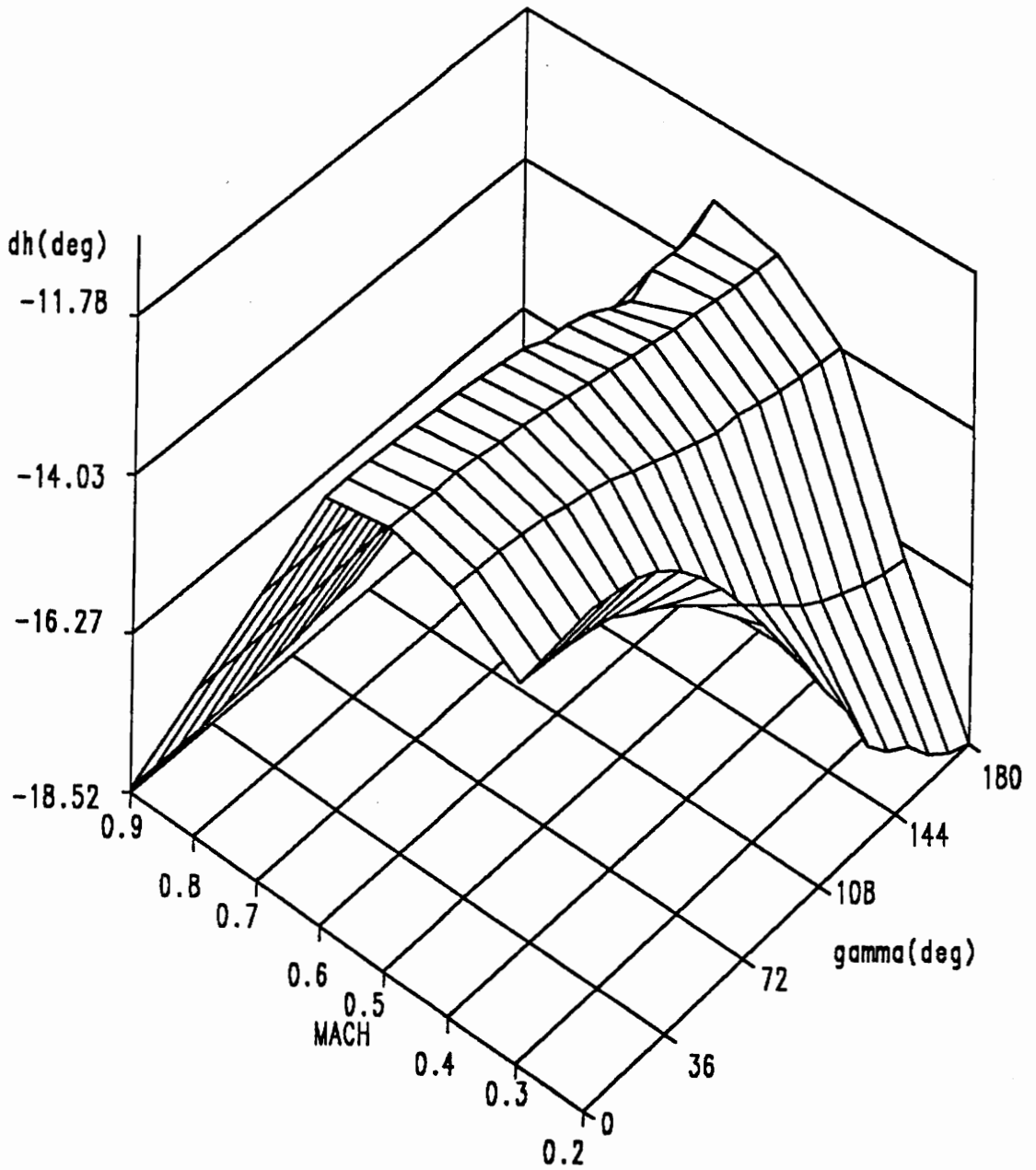


Figure 24. Outer Layer Stabilator Deflection ($h = 15000$ ft)

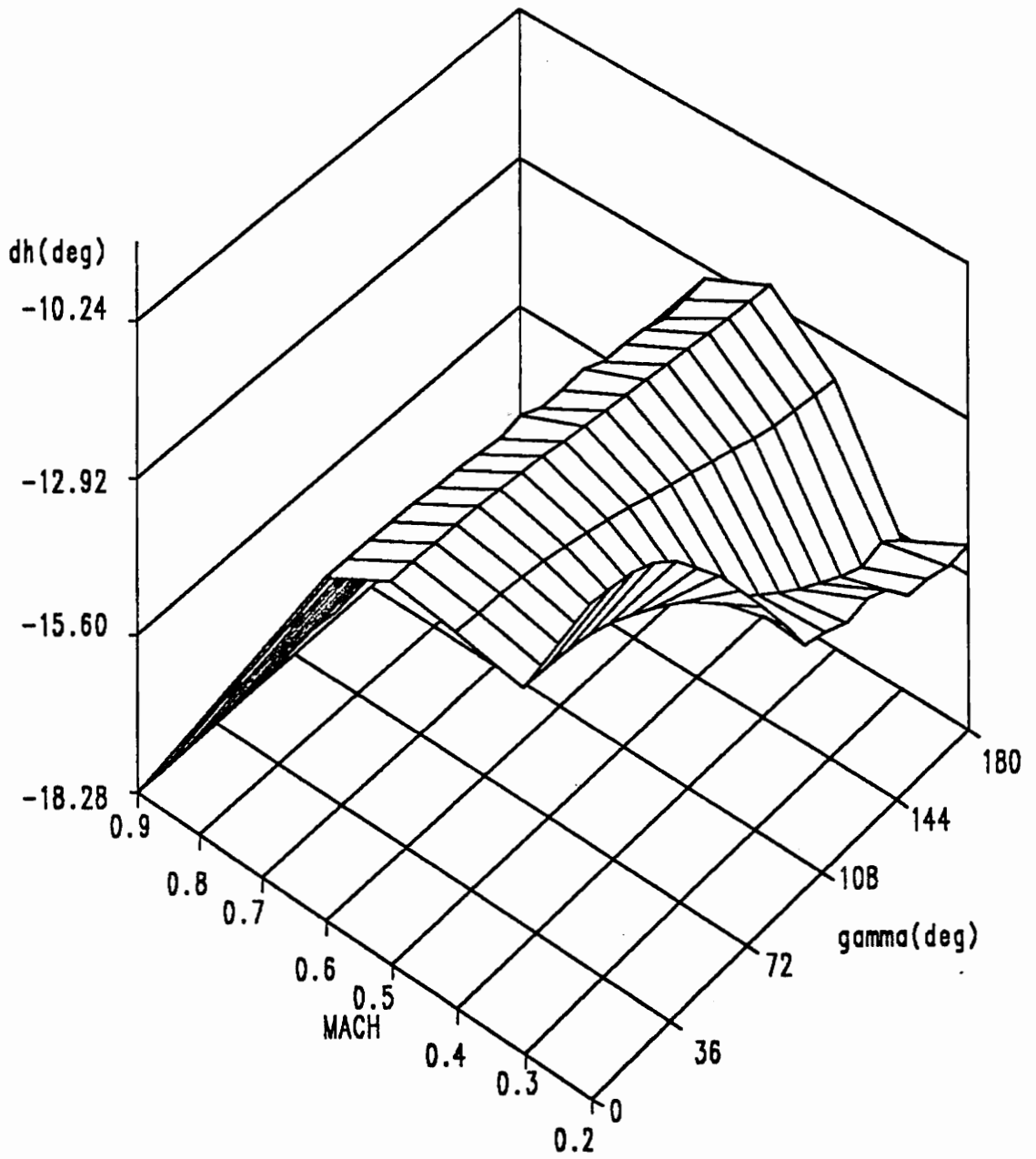


Figure 25. Outer Layer Stabilator Deflection ($h = 25000$ ft)

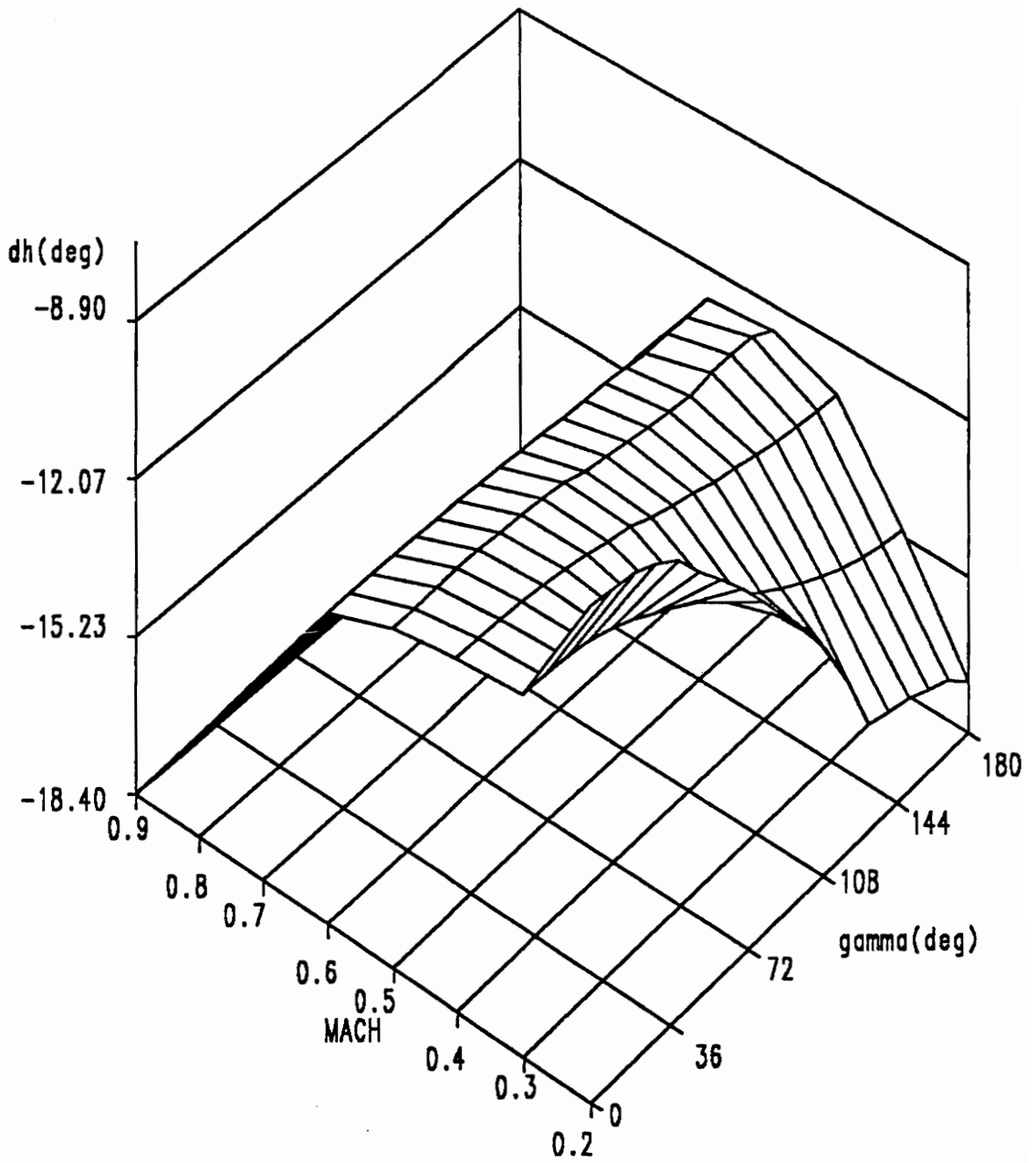


Figure 26. Outer Layer Stabilator Deflection ($h = 35000$ ft)

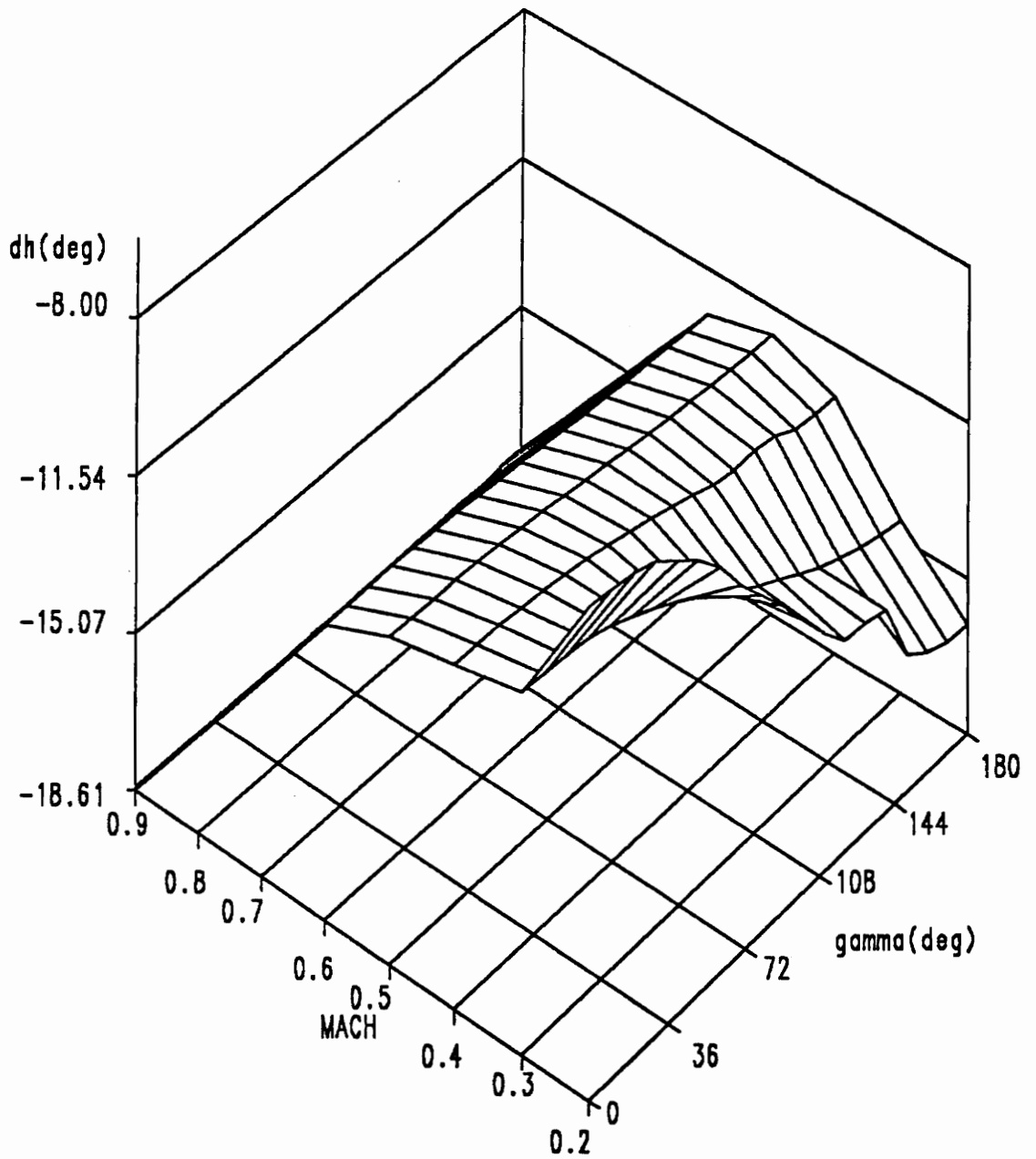


Figure 27. Outer Layer Stabilator Deflection ($h = 45000$ ft)

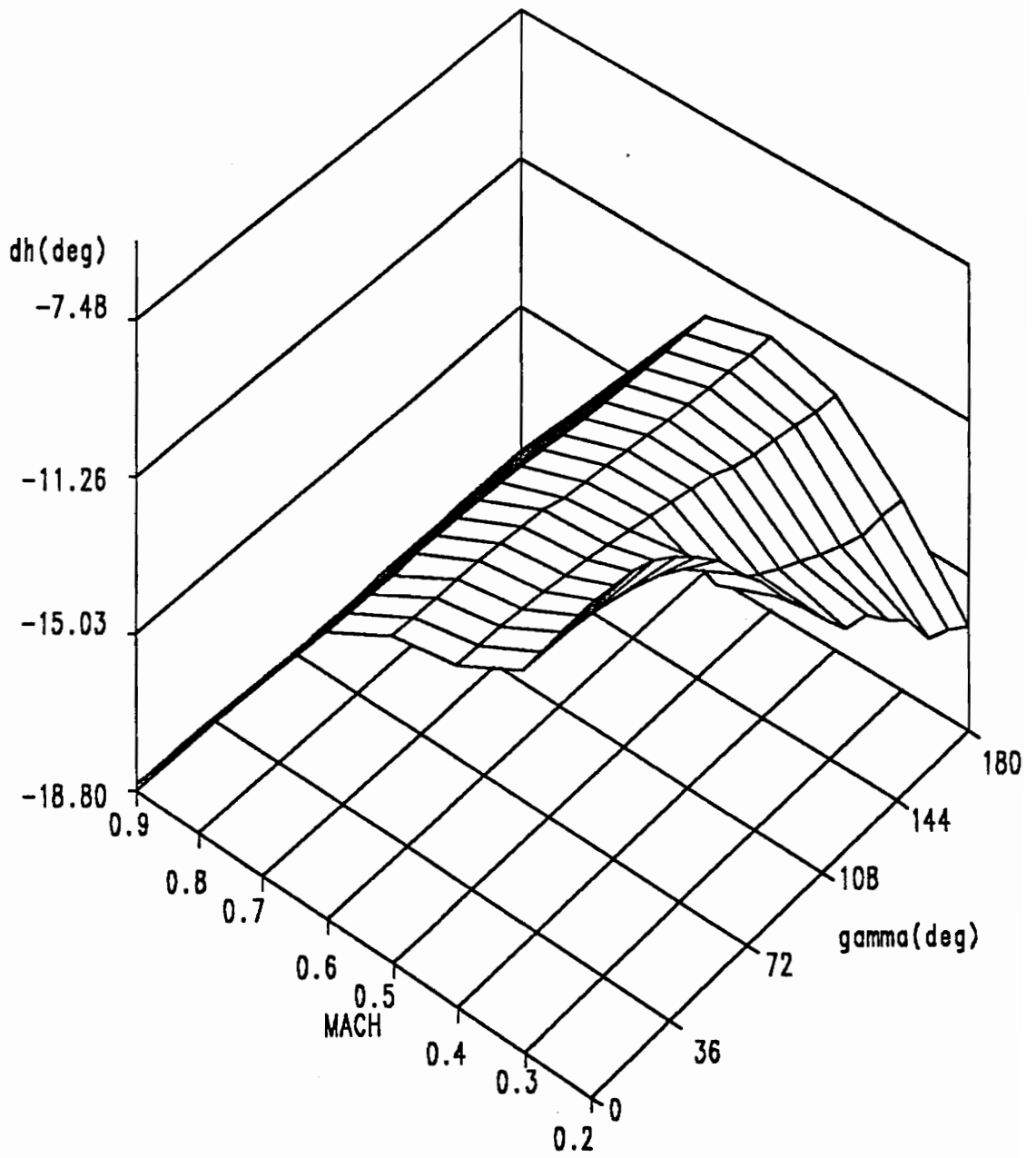


Figure 28. Outer Layer Stabilator Deflection ($h = 55000$ ft)

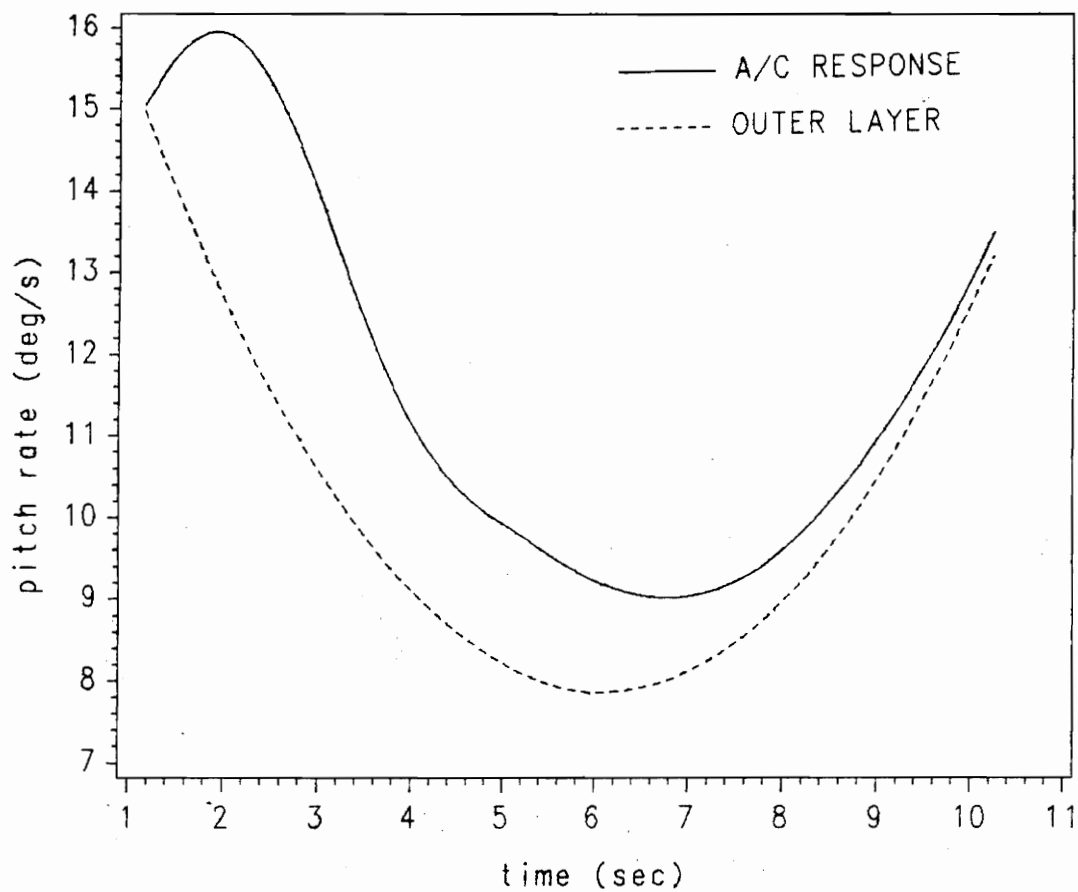


Figure 29. Simulation of Outer Layer ($M = 0.6$, $h = 15000$ ft)

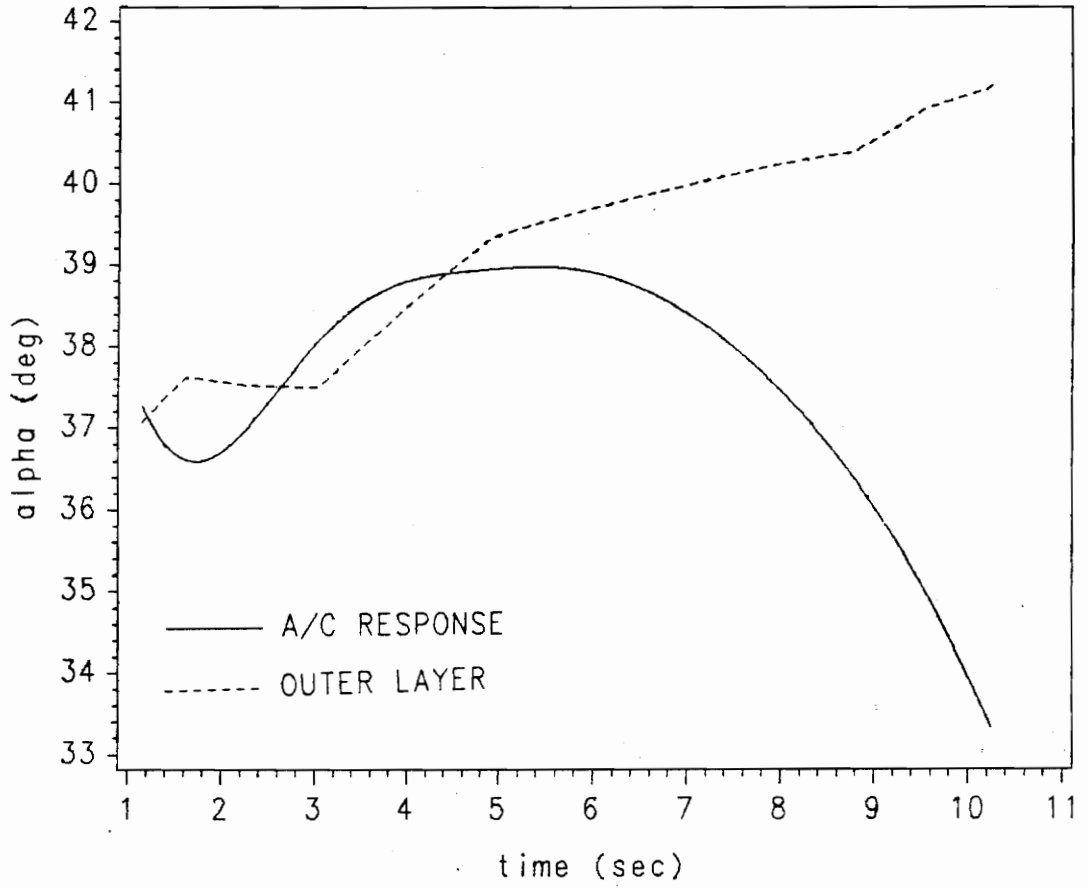


Figure 30. Simulation of Outer Layer ($M = 0.6$, $h = 15000$ ft)

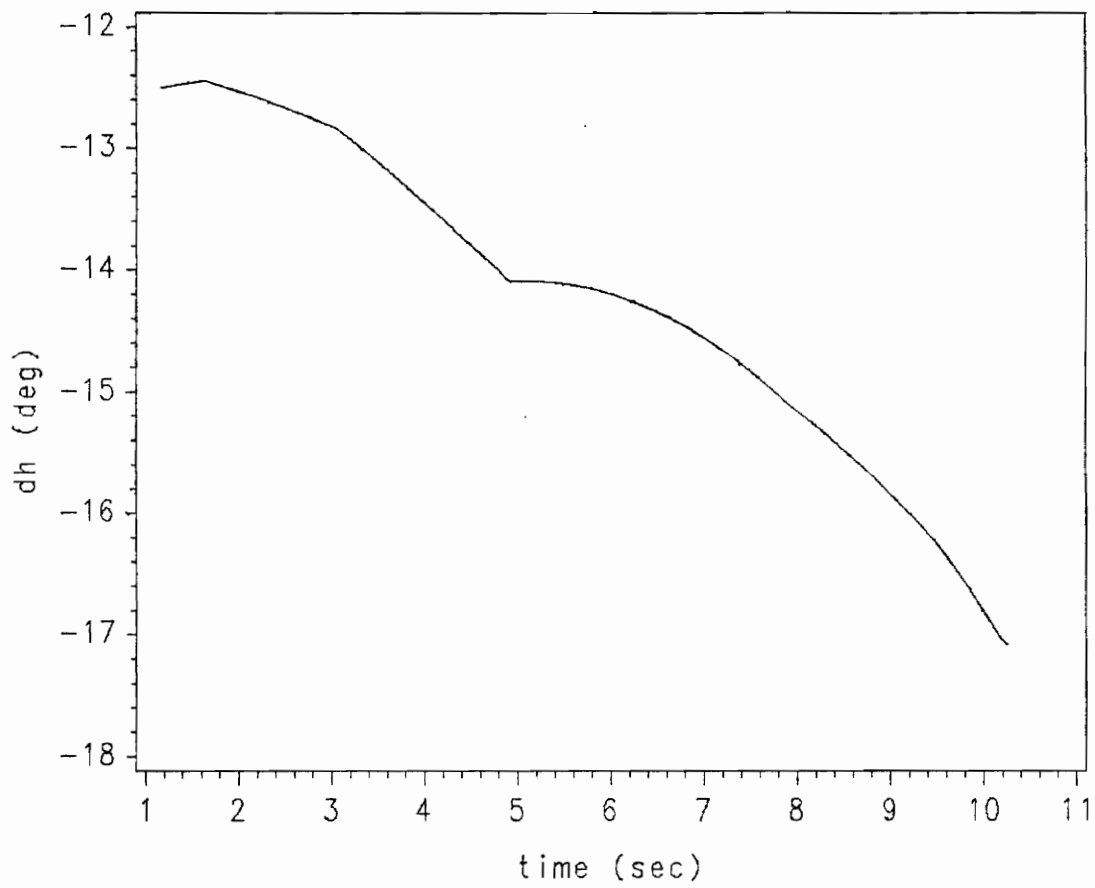


Figure 31. Simulation of Outer Layer ($M = 0.6$, $h = 15000$ ft)

5.3 Pole Placement

We observe, from Figures 29 and 30, that the actual states in the simulation of the feedback law deviate from the outer layer values of angle of attack and pitch rate. We are not sure why this is happening. It may be due to the relaxed stability of the F/A-18 aircraft and/or to the time varying Mach number which was assumed constant in deriving the outer layer solution. Consequently, we attempt the use of pole placement in order to bring the actual response closer to the outer layer solution. By placing the poles of the fast variable system at more negative values, the stability of the outer layer solution is improved. This pole placement is achieved by placing another loop around the system during the outer layer portion of the maneuver. The effect is to alter the stabilator feedback by an amount based on how far the system is from the desired outer layer angle of attack and pitch rate (fast variables). This decreases the time required to execute the outer layer portion of the trajectory. The system used in the pole placement is a linearized model of the outer layer fast variables. Pole placement is done using the Bass-Gura pole placement formula. The gains calculated using the Bass-Gura formula are multiplied by the corresponding divergences of the angle of attack and the pitch rate from the outer layer solution and these values are added to the outer layer stabilator feedback.

The linearized equations of motion for the fast variables are defined by :

$$\begin{bmatrix} \Delta \dot{\alpha} \\ \Delta \dot{q} \end{bmatrix} = \begin{bmatrix} \frac{df_{\alpha}}{d\alpha} & \frac{df_{\alpha}}{dq} \\ \frac{df_q}{d\alpha} & \frac{df_q}{dq} \end{bmatrix} \begin{bmatrix} \Delta \alpha \\ \Delta q \end{bmatrix} + \begin{bmatrix} \frac{df_{\alpha}}{d\delta_h} \\ \frac{df_q}{d\delta_h} \end{bmatrix} \delta_h \quad (5.32)$$

where:

$$f_\alpha = q + \frac{g}{V} \cos \gamma - \frac{\bar{q}S}{mV} C_L - \frac{T_x}{mV} \sin \alpha \quad (5.33)$$

$$f_q = \frac{1}{I_y} [l_{z_0} t_x + \bar{q}S(\bar{c}C_m + d_L C_L + d_D C_D)] \quad (5.34)$$

$$\frac{df_\alpha}{d\alpha} = - \frac{\bar{q}S}{mV} \frac{dC_L}{d\alpha} - \frac{T_x}{mV} \cos \alpha \quad (5.35)$$

$$\frac{df_\alpha}{dq} = 1 - \frac{\bar{q}S}{mV} C_{L_q} \quad (5.36)$$

$$\frac{df_q}{d\alpha} = \frac{\bar{q}S}{I_y} \left[\bar{c} \frac{dC_m}{d\alpha} + d_L \left(\frac{dC_L}{d\alpha} + C_D \right) + d_D \left(\frac{dC_D}{d\alpha} - C_L \right) \right] \quad (5.37)$$

$$\frac{df_q}{dq} = \frac{\bar{q}S}{I_y} \left[\bar{c} \frac{dC_m}{dq} + d_L \frac{dC_L}{dq} \right] \quad (5.38)$$

$$\frac{df_\alpha}{d\delta_h} = - \frac{\bar{q}S}{mV} \frac{dC_L}{d\delta_h} \quad (5.39)$$

$$\frac{df_q}{d\delta_h} = \frac{\bar{q}S}{I_y} \left[\bar{c} \frac{dC_m}{d\delta_h} + d_L \frac{dC_L}{d\delta_h} + d_D \frac{dC_D}{d\delta_h} \right] \quad (5.40)$$

Letting

$$A = \begin{bmatrix} \frac{df_\alpha}{d\alpha} & \frac{df_\alpha}{dq} \\ \frac{df_q}{d\alpha} & \frac{df_q}{dq} \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} \frac{df_\alpha}{d\delta_h} \\ \frac{df_q}{d\delta_h} \end{bmatrix} \quad (5.41)$$

The open-loop characteristic equation is

$$p_{OL}(\bar{\lambda}) = \det(\bar{\lambda}I - A) = \bar{\lambda}^2 + a_1\bar{\lambda} + a_2 \quad (5.42)$$

where $\bar{\lambda}$ = open-looped eigenvalues

$$a_1 = -\frac{df_\alpha}{d\alpha} - \frac{df_q}{dq} \quad (5.43)$$

$$a_2 = \frac{df_\alpha}{d\alpha} \frac{df_q}{dq} - \frac{df_\alpha}{dq} \frac{df_q}{d\alpha} \quad (5.44)$$

And the controllability matrix U is given as:

$$U = [B, AB] \\ = \begin{bmatrix} \frac{df_\alpha}{d\delta_h} & \frac{df_\alpha}{d\alpha} \frac{df_\alpha}{d\delta_h} & + & \frac{df_\alpha}{dq} \frac{df_q}{d\delta_h} \\ \frac{df_\alpha}{d\delta_h} & \frac{df_q}{d\alpha} \frac{df_\alpha}{d\delta_h} & + & \frac{df_q}{dq} \frac{df_q}{d\delta_h} \end{bmatrix} \quad (5.45)$$

It is desired to have Butterworth Poles. Therefore the desired closed-loop characteristic equation is:

$$p_{CL}(\tilde{\lambda}) = \tilde{\lambda}^2 + \bar{a}_1\tilde{\lambda} + \bar{a}_2 \quad (5.46)$$

where $\tilde{\lambda}$ = closed-loop poles

$$\bar{a}_1 = \sqrt{2\omega} \quad (5.47)$$

$$\bar{a}_2 = \omega^2 \quad (5.48)$$

$$\omega = 1.5 \quad (5.49)$$

The Bass-Gura Formula determines the feedback gains as follows:

$$K = (\bar{a}_1 - a_1, \bar{a}_2 - a_2)W^{-T}U^{-1} \quad (5.50)$$

Where the matrix W is a lower Toeplitz matrix defined by

$$W = \begin{bmatrix} 1 & 0 \\ a_1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{df_\alpha}{d\alpha} - \frac{df_q}{dq} & 1 \end{bmatrix} \quad (5.51)$$

The gains K are used in the outer layer feedback law to help stabilize the aircraft onto the outer layer. This is done in the following manner.

Define

$$\Delta\alpha = \alpha - \alpha^* \quad (5.52)$$

$$\Delta q = q - q^* \quad (5.53)$$

The total outer layer feedback law is then defined as:

$$\begin{aligned} \delta_h &= \delta_h^* - K \begin{bmatrix} \Delta\alpha \\ \Delta q \end{bmatrix} & -24^\circ \leq \delta_h \leq 10.5^\circ \\ &= \delta_h^* - K_1\Delta\alpha - K_2\Delta q \end{aligned} \quad (5.54)$$

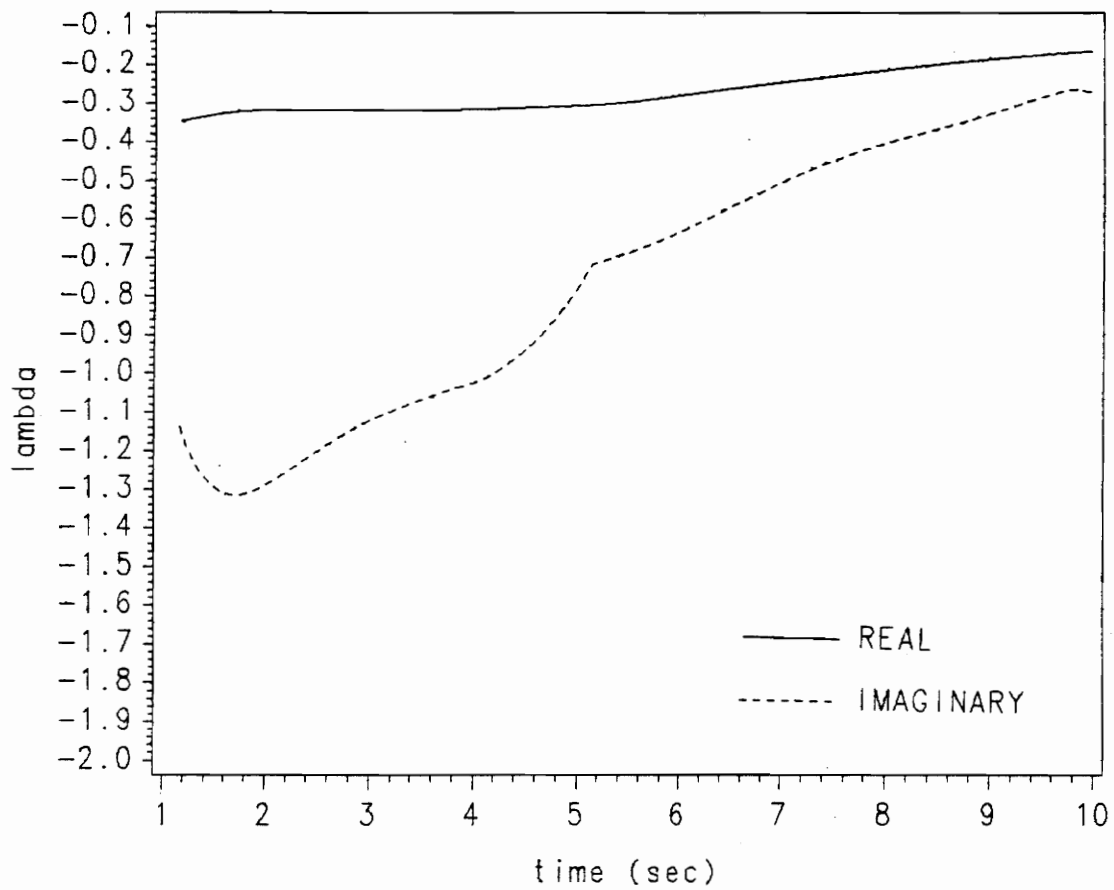


Figure 32. Eigenvalues of Fast System for Outer Layer Simulation

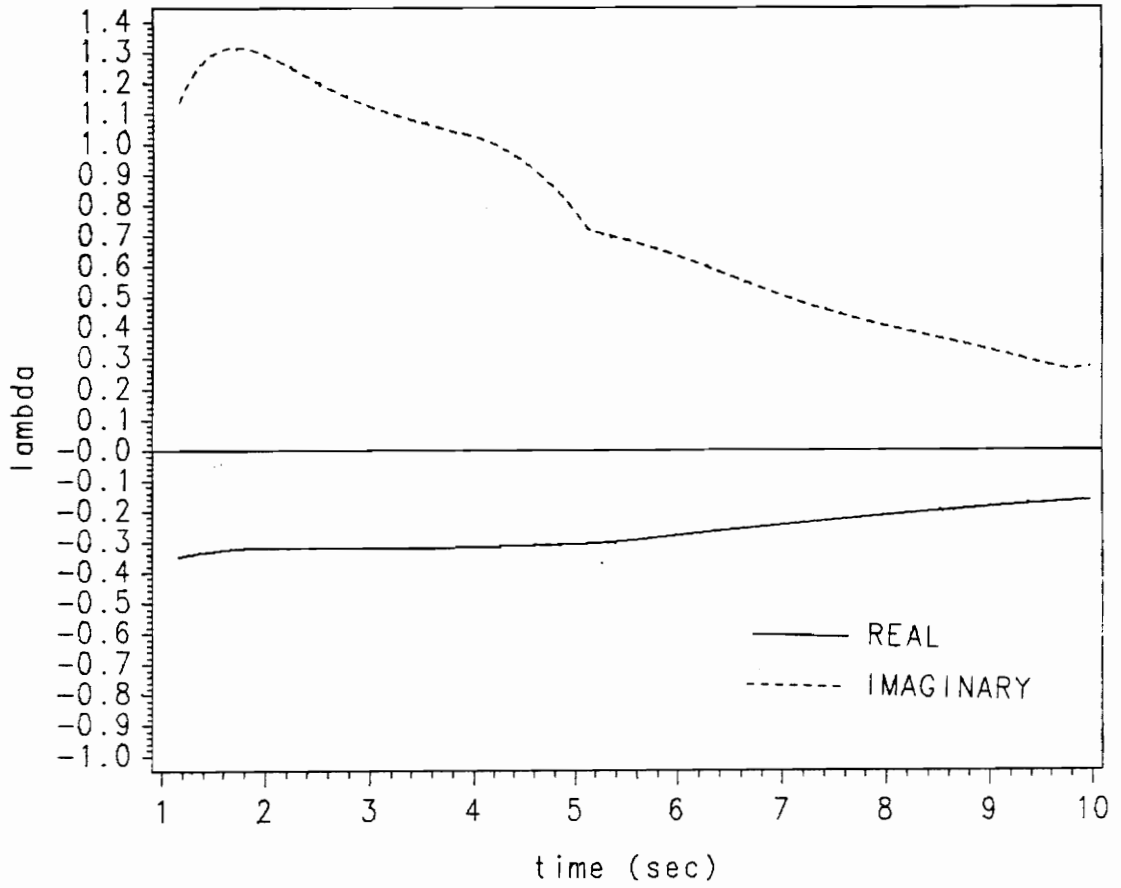


Figure 33. Eigenvalues of Fast System for Outer Layer Simulation

6.0 Transition Region Analysis

The transition region analysis involves the transition from the initial states to their outer layer values at the beginning of the maneuver and the transition from the outer layer to some final state at the end of the maneuver. The pitch rate and the angle of attack are assumed to be fast variables compared to Mach number, flight path angle and altitude. The analysis can be broken up into two portions. First, we consider the transition of the angle of attack from its initial state to its outer layer value. The pitch rate is assumed to be the fast variable here. Next, we consider the transition from the initial pitch rate to the outer layer pitch rate. We emphasize again that we are not obtaining a pure singular perturbation solution. Our singular perturbation treatment of these transition regions is separate from the singular perturbation treatment of the outer layer. We apply singular perturbation analysis to a new problem. This is the problem of bringing the states from a set of initial conditions to the outer layer values in minimum time, ignoring the fact that the outer layer values were obtained using singular perturbation analysis. That is, we use singular perturbation analysis to solve a new two-point boundary-value problem.

6.1 First Part of the Transition Region Analysis

The performance index ΔJ_α is the time required to transfer the angle of attack from its initial state to its outer layer value.

$$\Delta J_\alpha = \int_{t_0}^{t_f^\alpha} dt \quad (6.1)$$

The pitch rate is assumed to be fast compared to the angle of attack. Therefore, for a given flight path angle, Mach number and altitude, the equations of motion are:

$$\dot{\alpha} = q + \frac{g}{V} \cos \gamma - \frac{\bar{q}S}{mV} C_L - \frac{T_x}{mV} \sin \alpha \quad (6.2)$$

$$\varepsilon \dot{q} = \frac{1}{I_y} [l_z T_x + \bar{q}S(\bar{c}C_m + d_L C_L + d_D C_D)] \quad (6.3)$$

From equation (6.1)

$$\Delta J_\alpha = \int_{t_0}^{t_f^\alpha} dt = \int_{\alpha(t_0)}^{\alpha^\cdot(\gamma, M, h)} \frac{d\alpha}{\dot{\alpha}} \quad (6.4)$$

Equation (6.4) indicates that the integrand of the performance index is minimized when the rate of change of angle of attack is maximized. As ε approaches zero the problem can be considered in the following manner.

For a given γ , M , h and α ,

$$\max_{\delta_h, q} \dot{\alpha}(\alpha, \delta_h, \gamma, M, h, q)$$

Subject to equation (6.3) with $\varepsilon \rightarrow 0$

The values of the pitch rate and the stabilator deflection which satisfy these conditions are designated $\hat{\delta}_h(\alpha, \gamma, M, h)$ and $\hat{q}(\alpha, \gamma, M, h)$.

Let

$$\hat{\alpha}(\alpha, \gamma, M, h) = \max_{\delta_h, q} \dot{\alpha}(\alpha, \delta_h, \gamma, M, h, q) \quad (6.5)$$

Subject to

$$\begin{aligned} 0 &= \frac{1}{I_y} [l_z T_x + \bar{q}s(\bar{c}C_m + d_L C_L + d_D C_D)] \\ &= g(\alpha, \delta_h, \gamma, M, h, q) \end{aligned} \quad (6.6)$$

Define the Hamiltonian as:

$$\mathcal{H}(\alpha, \delta_h, \gamma, M, h, q) = -\dot{\alpha}(\alpha, \delta_h, \gamma, M, h, q) + \lambda g(\alpha, \delta_h, \gamma, M, h, q) \quad (6.7)$$

The necessary conditions for a stationary point on $-\dot{\alpha}$ are:

$$g(\alpha, \hat{\delta}_h, \gamma, M, h, \hat{q}) = 0 \quad (6.8)$$

$$\frac{d\mathcal{H}}{dq}(\alpha, \hat{\delta}_h, \gamma, M, h, \hat{q}) = 0 \quad (6.9)$$

$$\frac{d\mathcal{H}}{d\delta_h}(\alpha, \hat{\delta}_h, \gamma, M, h, \hat{q}) = 0 \quad (6.10)$$

Substituting equation (6.7) into equations (6.8) - (6.10)

$$g(\alpha, \hat{\delta}_h, \gamma, M, h, \hat{q}) = 0 \quad (6.11)$$

$$-\frac{d\dot{\alpha}}{dq}(\alpha, \hat{\delta}_h, \gamma, M, h, \hat{q}) + \lambda \frac{dg}{dq}(\alpha, \hat{\delta}_h, \gamma, M, h, \hat{q}) = 0 \quad (6.12)$$

$$-\frac{d\dot{\alpha}}{d\delta_h}(\alpha, \hat{\delta}_h, \gamma, M, h, \hat{q}) + \lambda \frac{dg}{d\delta_h}(\alpha, \hat{\delta}_h, \gamma, M, h, \hat{q}) = 0 \quad (6.13)$$

If

$$C_L(\alpha, \delta_h, M) = C_{L_0}(\alpha, \delta_h, M) + C_{L_q}(\alpha, \delta_h, M)q + C_{L_{\dot{\alpha}}}(\alpha, \delta_h, M)\dot{\alpha} \quad (6.14)$$

and

$$C_m(\alpha, \delta_h, M) = C_{m_0}(\alpha, \delta_h, M) + C_{m_q}(\alpha, \delta_h, M)q + C_{m_{\dot{\alpha}}}(\alpha, \delta_h, M)\dot{\alpha} \quad (6.15)$$

Then

$$\dot{\alpha} = \frac{q + \frac{g}{V} \cos \gamma - \frac{\bar{q}S}{mV} (C_{L_0} + C_{L_q}q + C_{L_{\dot{\alpha}}}\dot{\alpha}) - \frac{T_x}{mV} \sin \alpha}{1 + \frac{\bar{q}S}{mV} C_{L_{\dot{\alpha}}}} \quad (6.16)$$

and

$$k = \frac{1}{I_y} \left[l_{z^*} T_x + \bar{q}S [\bar{c} (C_{m_0} + C_{m_q}q + C_{m_{\dot{\alpha}}}\dot{\alpha}) + d_D C_D + d_L (C_{L_0} + C_{L_q}q + C_{L_{\dot{\alpha}}}\dot{\alpha})] \right] \quad (6.17)$$

Therefore,

$$\frac{d\dot{\alpha}}{dq} = \frac{1 - \frac{\bar{q}S}{mV} C_{L_q}}{1 + \frac{\bar{q}S}{mV} C_{L_{\dot{\alpha}}}} \quad (6.18)$$

$$\frac{d\dot{\alpha}}{d\delta_h} = \frac{-\frac{\bar{q}S}{mV} \frac{dC_{L_0}}{d\delta_h}}{1 + \frac{\bar{q}S}{mV} C_{L_{\dot{\alpha}}}} \quad (6.19)$$

$$\frac{dk}{dq} = \frac{\bar{q}S}{I_y} [\bar{c} C_{m_q} + d_L C_{L_q}] \quad (6.20)$$

$$\frac{dk}{d\delta_h} = \frac{\bar{q}S}{I_y} \left[\bar{c} \frac{dC_{m_0}}{d\delta_h} + d_D \frac{dC_D}{d\delta_h} + d_L \frac{dC_{L_0}}{d\delta_h} \right] \quad (6.21)$$

Equations (6.18)-(6.21) are used in equations (6.12)-(6.13) to satisfy the necessary conditions for a stationary point on $-\dot{\alpha}$

6.2 Results: First Part of the Transition Region Analysis

The problem is solved numerically using "SLLSQP" (Ref. 10). The feedback control law involves saturation of the control in order to minimize ΔJ_* . Consequently, this control law is not a function of flight path angle, Mach number or altitude. It is therefore only a function of angle of attack. This result is due the physical constraints of the stabilator angle deflection and indicates that the unconstrained minimum time control setting is outside the physical limits of the aircraft.

For $\alpha < \alpha^*(\gamma, M, h)$

$$\hat{\delta}_h(\alpha) = \begin{cases} -24^\circ & \alpha < 45^\circ \\ 10.5^\circ & 45^\circ \leq \alpha \leq 55^\circ \\ -24^\circ & \alpha > 55^\circ \end{cases} \quad (6.22)$$

For $\alpha > \alpha^*(\gamma, M, h)$

$$\hat{\delta}_h(\alpha) = \begin{cases} 10.5^\circ & \alpha < 45^\circ \\ -24^\circ & 45^\circ \leq \alpha \leq 55^\circ \\ 10.5^\circ & \alpha > 55^\circ \end{cases} \quad (6.23)$$

Equation (6.23) results from analyzing the inverse problem where the initial angle of attack is greater than the outer layer angle of attack.

Given the control which maximizes the rate of change of angle of attack for all pitch rates, we now must calculate the pitch rate which gives us $\dot{q} = 0$. This is done by solving equation (6.6) for q and using the control defined by equations (6.22) and (6.23).

$$\hat{q}(\alpha, M, h) = \frac{-l_z T_x - \bar{q} S (\bar{c} C_{m_0} + d_L C_{L_0} + d_D C_{D_0})}{\bar{q} S (\bar{c} C_{m_q} + d_L C_{L_q})} \quad (6.24)$$

Therefore equations (6.22)-(6.24) define the solution to the first part of the transition region analysis. In order to minimize ΔJ_s , we use the control law $\hat{\delta}_h(\alpha)$ with the pitch rate equal to \hat{q} .

6.3 Second Part of the Transition Region Analysis

For a given angle of attack, the transition from the initial pitch rate to $\hat{q}(M, h)$ is considered. The equations of motion are

$$\dot{q} = \frac{1}{I_y} [l_z T_x + \bar{q} S (\bar{c} C_m + d_L C_L + d_D C_D)] \quad (6.25)$$

$$\dot{V} = -g \sin \gamma - \frac{\bar{q} S}{m} C_D + \frac{T_x}{m} \cos \alpha \quad (6.26)$$

$$\dot{h} = V \sin \gamma \quad (6.27)$$

The performance index to be minimized is

$$\Delta J_q = \int_{t_0}^{t^*} dt \quad (6.28)$$

$$\dot{q} = \frac{dq}{dt} \quad (6.29)$$

Therefore,

$$\Delta J_q = \int_{q(t_0)}^{q^*(\gamma, M, h)} \frac{dq}{\dot{q}} \quad (6.30)$$

In order to minimize the integrand of the performance index, we desire to maximize the rate of change of pitch rate for all q on the interval from the initial pitch rate to \hat{q} . From equation (6.25)

$$\dot{q} = \frac{1}{I_y} [l_z T_x + \bar{q} S (\bar{c}(C_{m_0}(\alpha, \delta_h) + C_{m_q}(\alpha)q) + d_L(C_{L_0}(\alpha, \delta_h) + C_{L_q}(\alpha)q) + d_D C_D(\alpha, \delta_h))] \quad (6.31)$$

we can see that we choose δ_h such that the quantity

$$\bar{c}C_{m_0}(\alpha, \delta_h) + d_L C_{L_0}(\alpha, \delta_h) + d_D C_D(\alpha, \delta_h)$$

is maximized. Numerical results indicate that this quantity is maximized by the following control law.

$$\tilde{\delta}_h = \begin{cases} -24^\circ & q < \hat{q}(\alpha, M, h) \\ 10.5^\circ & q > \hat{q}(\alpha, M, h) \end{cases} \quad (6.32)$$

The stabilator angle which minimizes the integrand of the performance index is outside the allowable region for the F/A-18. Therefore, the control law which satisfies the aircraft constraints calls for the stabilator to be saturated.

7.0 Transition Regions

The entire maneuver is considered using three distinct portions. These portions include two transition regions and the outer layer region. The transition regions are necessary because the fast variables cannot be instantaneously changed from the initial conditions to the outer layer and from the outer layer to the final conditions.

We desire to execute the transition regions in minimum time. We treat the transition regions as a two-point boundary-value problem. Therefore, switching times are calculated by considering approximate solutions to the nonlinear differential equations of motion. The approximate solutions are used to find the switching times necessary to bring the aircraft state to the outer layer or the final state of the half-loop maneuver.

7.1 First Transition Region

The switching time for the first transition region is defined as the time at which a switch in control will bring the angle of attack and the pitch rate to their outer layer values simultaneously.

The time rate of change of the pitch rate is given by

$$\dot{q} = \frac{1}{I_y} \left[l_{z_s} T_x + \bar{q} S [\bar{c} C_{m_0} + d_L C_{L_0} + d_D C_D + (\bar{c} C_{m_{\dot{\alpha}}} + d_L C_{L_{\dot{\alpha}}}) \dot{\alpha} + (\bar{c} C_{m_q} + d_L C_{L_q}) q] \right] \quad (7.1)$$

$$\dot{q} = [f_2(\delta_h) + q] f_1 \quad (7.2)$$

Where

$$f_1 = \frac{\bar{q} S}{I_y} (\bar{c} C_{m_q} + d_L C_{L_q}) \quad (7.3)$$

$$f_2(\delta_h) = \frac{l_{z_s} T_x + \bar{q} S [\bar{c} C_{m_0} + d_L C_{L_0} + d_D C_D + (\bar{c} C_{m_{\dot{\alpha}}} + d_L C_{L_{\dot{\alpha}}}) \dot{\alpha}]}{\bar{q} S (\bar{c} C_{m_q} + d_L C_{L_q})} \quad (7.4)$$

The solution to equation (7.2) is

$$q(t) = [f_2(\delta_h) + q(t_0)] e^{f_1(t-t_0)} - f_2(\delta_h) \quad (7.5)$$

Using the control law

$$\delta_h^{(1)} = \begin{cases} -24^\circ & q < q^*(\gamma, M, h) \\ 10.5^\circ & q > q^*(\gamma, M, h) \end{cases} \quad (7.6)$$

in equation (7.5), it follows that

$$q^* = [f_2(\delta_h^{(1)}) + q(t_0)]e^{f_1\Delta t} - f_2(\delta_h^{(1)}) \quad (7.7)$$

Where Δt is the time required to change the pitch rate from $q(t_0)$ to q^* using $\delta_h^{(1)}$.

$$\Delta t = \frac{1}{f_1} \ln \left[\frac{q^* + f_2(\delta_h^{(1)})}{q(t_0) + f_2(\delta_h^{(1)})} \right] \quad (7.8)$$

The time rate of change of the angle of attack is given by

$$\dot{\alpha} = q + \frac{g}{V} \cos \gamma - \frac{\bar{q}S}{mV} (C_{L_0} + C_{L_q}q + c_{L_{\dot{\alpha}}}\dot{\alpha}) - \frac{T_x}{mV} \sin \alpha \quad (7.9)$$

$$\dot{\alpha} = f_3q + f_4(\delta_h) \quad (7.10)$$

where

$$f_3 = \frac{1 - \frac{\bar{q}S}{mV} C_{L_q}}{1 + \frac{\bar{q}S}{mV} C_{L_{\dot{\alpha}}}} \quad (7.11)$$

$$f_4(\delta_h) = \frac{\frac{g}{V} \cos \gamma - \frac{\bar{q}S}{mV} C_{L_0} - \frac{T_x}{mV} \sin \alpha}{1 + \frac{\bar{q}S}{mV} C_{L_{\dot{\alpha}}}} \quad (7.12)$$

The solution to equation (7.10) is

$$\alpha(t) = [f_4(\delta_h^{(1)})_h + f_3q] \Delta t \quad (7.13)$$

We calculate the change in angle of attack for a given Δt and $\delta_h^{(1)}$:

$$\Delta\alpha = [f_4(\delta_h^{(1)}) - f_3f_2(\delta_h^{(1)})]\Delta t + \frac{f_3}{f_1} [f_2(\delta_h^{(1)}) + q(t_0)](e^{f_1\Delta t} - 1) \quad (7.14)$$

Therefore, to bring the angle of attack and the pitch rate to the outer layer values at the same time, we apply $\delta_h^{(1)}$ when $|\alpha^*(\gamma, M, h) - \alpha| \leq \Delta\alpha$.

7.2 Second Transition Region

In order to change the angle of attack and the pitch rate from the outer layer to the final conditions of zero pitch rate and a specified flight path angle, the switching time for the second transition region must be determined. The switching time is defined as the time in which the pitch rate is changed from its outer layer value to zero at the same time that the flight path angle reaches the desired value ($\gamma_f = 180^\circ$ for the half-loop maneuver).

Assuming that the flight path angle is near the desired value and nearly constant aerodynamic coefficients, the time rate of change of flight path angle is given by

$$\dot{\gamma} = -\frac{g}{V} \cos \gamma + \frac{\bar{q}S}{mV} (C_{L_0} + C_{L_q}q + C_{L_\alpha}\dot{\alpha}) + \frac{T_x}{mV} \sin \alpha \quad (7.15)$$

$$\dot{\gamma} = f_5q + f_6(\delta_h) \quad (7.16)$$

where

$$f_5 = \frac{\bar{q}S}{mV} C_{L_q} \quad (7.17)$$

$$f_6(\delta_h) = -\frac{g}{V} \cos(\theta - \alpha) + \frac{\bar{q}S}{mV} (C_{L_0} + C_{L_\alpha} \dot{\alpha}) + \frac{T_x}{mV} \sin \alpha \quad (7.18)$$

Using $\delta_h^{(2)} = 10.5^\circ$ in equation (7.5) with final pitch rate equal to zero

$$0 = [f_2(\delta_h^{(2)}) + q^*] e^{f_1 \Delta t} - f_2(\delta_h^{(2)}) \quad (7.19)$$

The time required to reach this condition is

$$\Delta t = \frac{1}{f_1} \ln \left[\frac{f_2(\delta_h^{(2)})}{q^* + f_2(\delta_h^{(2)})} \right] \quad (7.20)$$

Equation (7.16) is solved to find the change in flight path angle for Δt and $\delta_h^{(2)}$

$$\Delta \gamma = [f_6(\delta_h^{(2)}) - f_5 f_2(\delta_h^{(2)})] \Delta t + \frac{f_5}{f_1} [f_2(\delta_h^{(2)}) + q^*] (e^{f_1 \Delta t} - 1) \quad (7.21)$$

Therefore, to bring the pitch rate to zero and the flight path angle to a specified value simultaneously, we apply $\delta_h^{(2)}$ when $\gamma_f - \gamma \leq \Delta \gamma$.

8.0 Nonlinear Feedback Law

The feedback laws for each portion of the maneuver are combined to define the overall feedback law. A computer code in the form of a subroutine provides the feedback control law. It generates a value of the stabilator control setting for any given value of the F/A-18 aircraft state vector. The feedback law is divided into three regions corresponding to the first transition region, the outer layer and the second transition region.

8.1 First Transition Region

In this region, we are concerned with bringing the aircraft states from their initial values to the outer layer values simultaneously. This process involves at least two and possibly three different combinations of states which require a particular control law strategy.

8.1.1 Part I

In this part of the first transition region, we wish to bring the angle of attack up to a prescribed neighborhood of the outer layer angle of attack. The pitch rate is not equal to the value which gives us a constant pitch rate as prescribed in the analysis of the first inner layer.

$$q \neq \hat{q}(\alpha, M, h)$$

and the angle of attack is not within the acceptable neighborhood of the outer layer angle of attack required for the first transition region.

$$|\alpha^*(\gamma, M, h) - \alpha| > \Delta\alpha$$

The control law for this is given by:

$$\text{For } q < \hat{q}(\alpha, M, h)$$

$$\delta_h^I = \begin{cases} -24^\circ & \alpha \leq 45^\circ \text{ or } \alpha \geq 55^\circ \\ 10.5^\circ & 45^\circ < \alpha < 55^\circ \end{cases} \quad (8.1)$$

$$\text{For } q > \hat{q}(\alpha, M, h)$$

$$\delta_h^I = \begin{cases} 10.5^\circ & \alpha \leq 45^\circ \text{ or } \alpha \geq 55^\circ \\ -24^\circ & 45^\circ < \alpha < 55^\circ \end{cases} \quad (8.2)$$

We apply δ_h^I until $q = \hat{q}(\alpha, M, h)$ or $|\alpha^*(\gamma, M, h) - \alpha| \leq \Delta\alpha$.

8.1.2 Part II

This part of the first transition region is used to bring the angle of attack up to the prescribed neighborhood for the case where the pitch rate is equal to the value required for a constant pitch rate in the first part of the transition region analysis

$$q = \hat{q}(\alpha, M, h)$$

and the angle of attack is not within the required neighborhood of the outer layer value desired.

$$|\alpha^*(\theta, M, h) - \alpha| > \Delta\alpha$$

The feedback law for this part of the transition is

For $\alpha < \alpha^*(\gamma, M, h)$

$$\delta_h^{II} = \begin{cases} -24^\circ & \alpha \leq 45^\circ \text{ or } \alpha \geq 55^\circ \\ 10.5^\circ & 45^\circ < \alpha < 55^\circ \end{cases} \quad (8.3)$$

For $\alpha > \alpha^*(\gamma, M, h)$

$$\delta_h^{II} = \begin{cases} 10.5^\circ & \alpha \leq 45^\circ \text{ or } \alpha \geq 55^\circ \\ -24^\circ & 45^\circ < \alpha < 55^\circ \end{cases} \quad (8.4)$$

We apply δ_h^{II} until $|\alpha^*(\gamma, M, h) - \alpha| \leq \Delta\alpha$.

Often, this portion of the transition region is not used because the angle of attack is within the required neighborhood before the pitch rate reaches \hat{q} . If this occurs it is no longer desirable to have the pitch rate equal to \hat{q}

8.1.3 Part III

In this portion of the transition region, we are concerned with applying a control which will bring the pitch rate to the required outer layer pitch rate. The pitch rate is less than or equal to the \hat{q}

$$q \leq \hat{q}(\alpha, M, h)$$

and the angle of attack is within the necessary neighborhood to bring it to the outer layer angle of attack

$$|\alpha^*(\gamma, M, h) - \alpha| \leq \Delta\alpha$$

The control which accomplishes the objective of this portion of the transition is given as

$$\delta_h^{III} = \begin{cases} -24^\circ & q < q^*(\gamma, M, h) \\ 10.5^\circ & q > q^*(\gamma, M, h) \end{cases} \quad (8.5)$$

We apply δ_h^{III} until the angle of attack and the pitch rate are equal to the desired outer layer values: $\alpha = \alpha^*(\gamma, M, h)$ and $q = q^*(\gamma, M, h)$.

8.2 Outer Layer

In the outer layer region, the velocity vector is rotated through a flight path angle of one hundred and eighty degrees. The values of the angle of attack, pitch rate and stabilator deflection are determined by the surfaces of the outer layer solution. Therefore, the values of angle of attack and pitch rate are given by

$$q = q^*(\gamma, M, h)$$

$$\alpha = \alpha^*(\gamma, M, h)$$

and the flight path angle is not within the neighborhood of the desired final flight path angle.

$$\gamma < \gamma_f - \Delta\gamma$$

The control law is defined by the surface represented by the outer layer solution for stabilator deflection

$$\delta_h^{OL} = \delta_h^*(\gamma, M, h) \tag{8.6}$$

We apply δ_h^{OL} until $\gamma \geq \gamma_f - \Delta\gamma$.

The size of the neighborhood defined by $\Delta\gamma$ is calculated throughout the outer layer portion of the maneuver in order to determine when to switch to the control law of the second transition region. This time is defined as when the flight path angle equals the

value at which a switch in the control will bring the flight path angle to its final value at the same time that the pitch rate is zero.

8.3 *Second Transition Region*

This final region of the control law is used to bring the flight path angle to the desired final value and the pitch rate to zero simultaneously. The angle of attack and pitch rate are equal to their outer layer values at the beginning of the region

$$\left. \begin{aligned} q &= q^*(\gamma, M, h) \\ \alpha &= \alpha^*(\gamma, M, h) \end{aligned} \right\} \text{initially}$$

and the flight path angle is within the neighborhood required for the second transition region.

$$\gamma \geq \gamma_f - \Delta\gamma$$

The control used to bring the pitch rate down to zero is

$$\delta_h^F = 10.5^\circ \tag{8.7}$$

and is applied until $q = 0$ and $\gamma = \gamma_f$

Equations (8.1)-(8.7) define the entire feedback control law necessary to bring the aircraft from an initial state through the half-loop maneuver to the final flight path angle with a zero pitch rate.

9.0 Results

The following figures show the results of simulations of the half-loop maneuver using the feedback control law developed in this study. The initial altitude in all cases was 15,000 ft. These simulations were generated using a fourth-order Runge-Kutta integrator on the VAX 8800 computer at Virginia Polytechnic Institute and State University. Figures 34 - 46 show the simulation results for an initial Mach number of 0.9. The simulation results for an initial Mach number of 0.6 are shown in figures 47 - 59. In figures 60 - 72, the simulation results are given for an initial Mach number of 0.3. For each simulation, the circular frequency of the Butterworth Poles is 1.5 rad/sec.

When the initial Mach number is 0.9, the maneuver takes approximately nine (9) seconds. In figures 34 and 35, it is possible to see the large change in the angle of attack and pitch rate associated with the first transition region. This region of the maneuver takes about one (1) second to complete. The angle of attack is increased from approximately six degrees to the outer layer value of about thirty-four (34) degrees. In the same time period, the pitch rate is increased from zero up to approximately one hundred and ten (110) degrees per second and back down to the outer layer pitch rate which is about

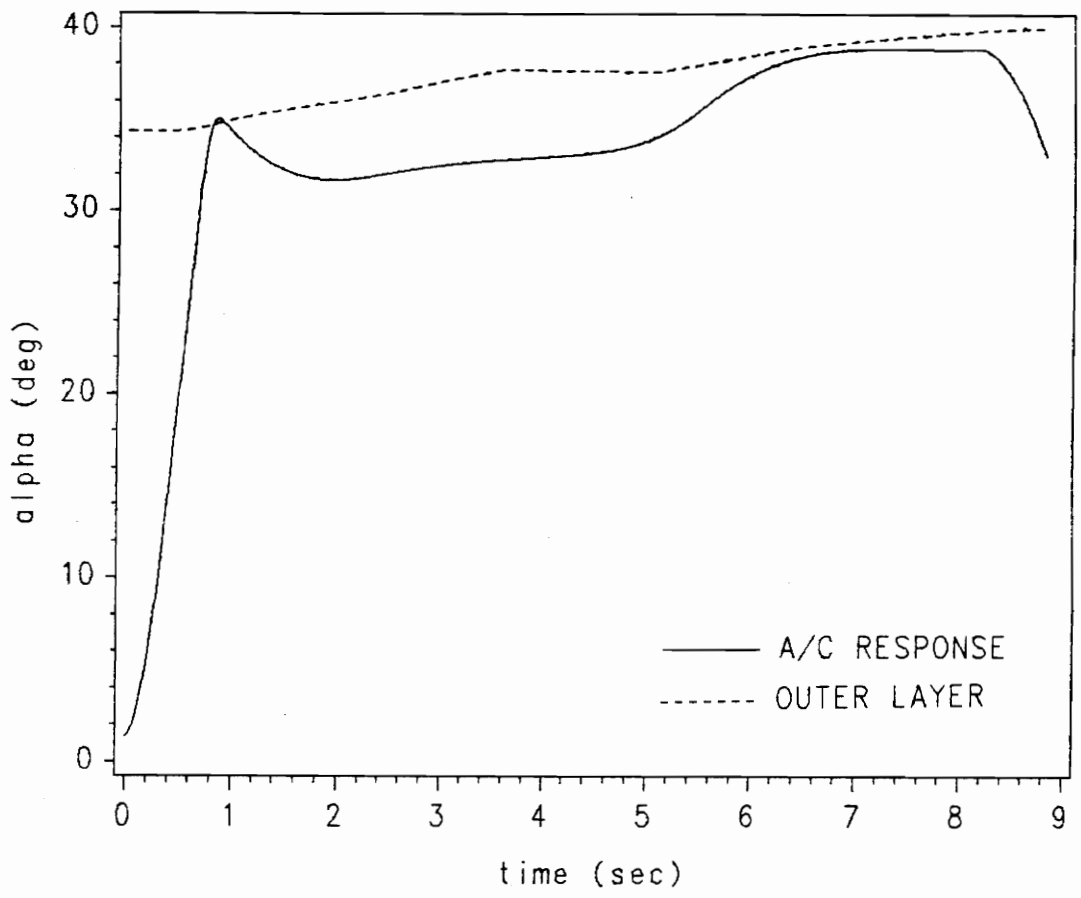


Figure 34. Half-Loop Simulation: Angle of Attack ($M(t=0)=0.9$, $h=15000$ ft)

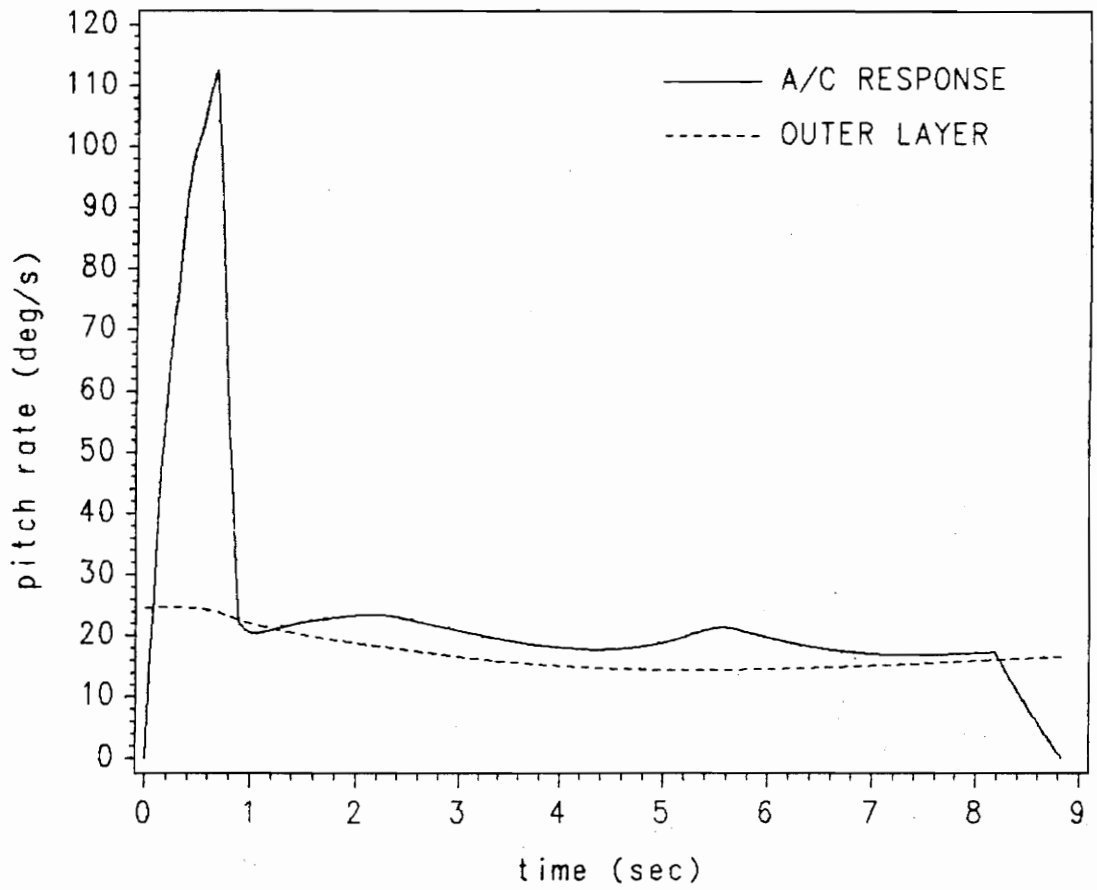


Figure 35. Half-Loop Simulation: Pitch Rate ($M(t=0) = 0.9$, $h = 15000$ ft)

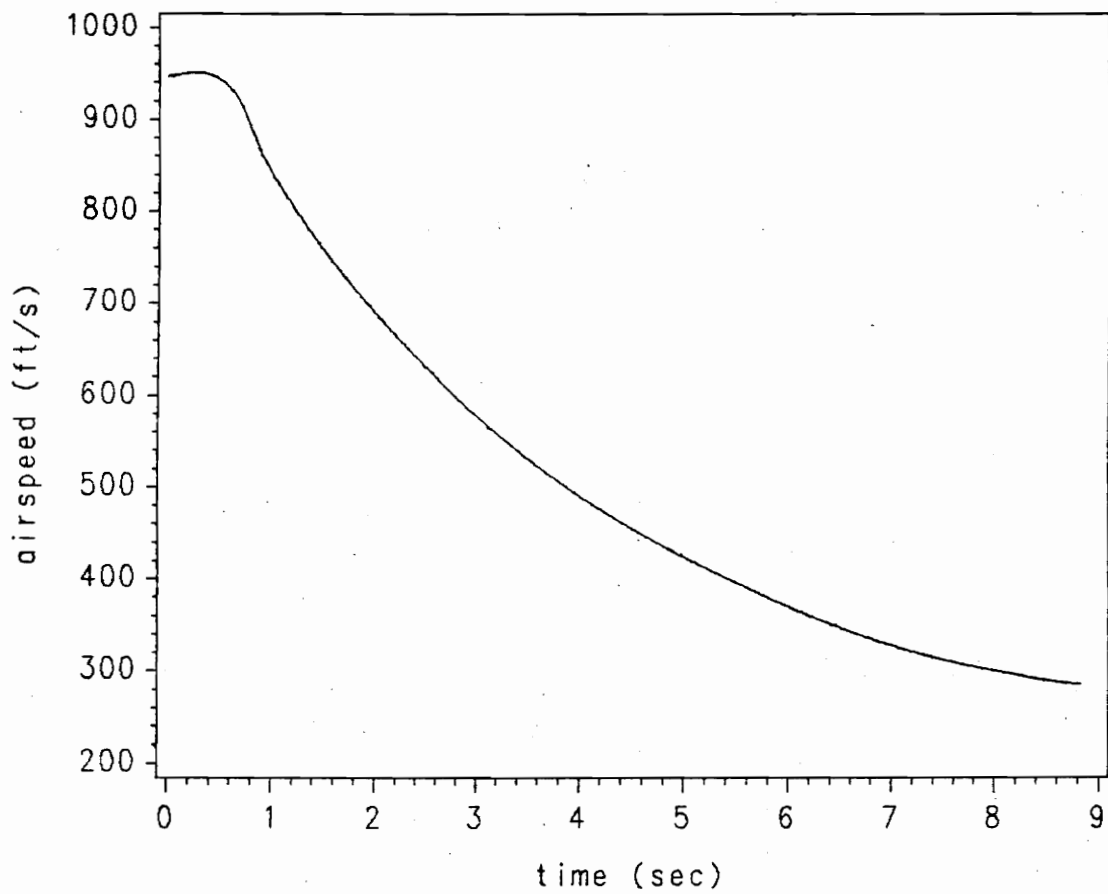


Figure 36. Half-Loop Simulation: Airspeed ($M(t=0)=0.9$, $h=15000$ ft)

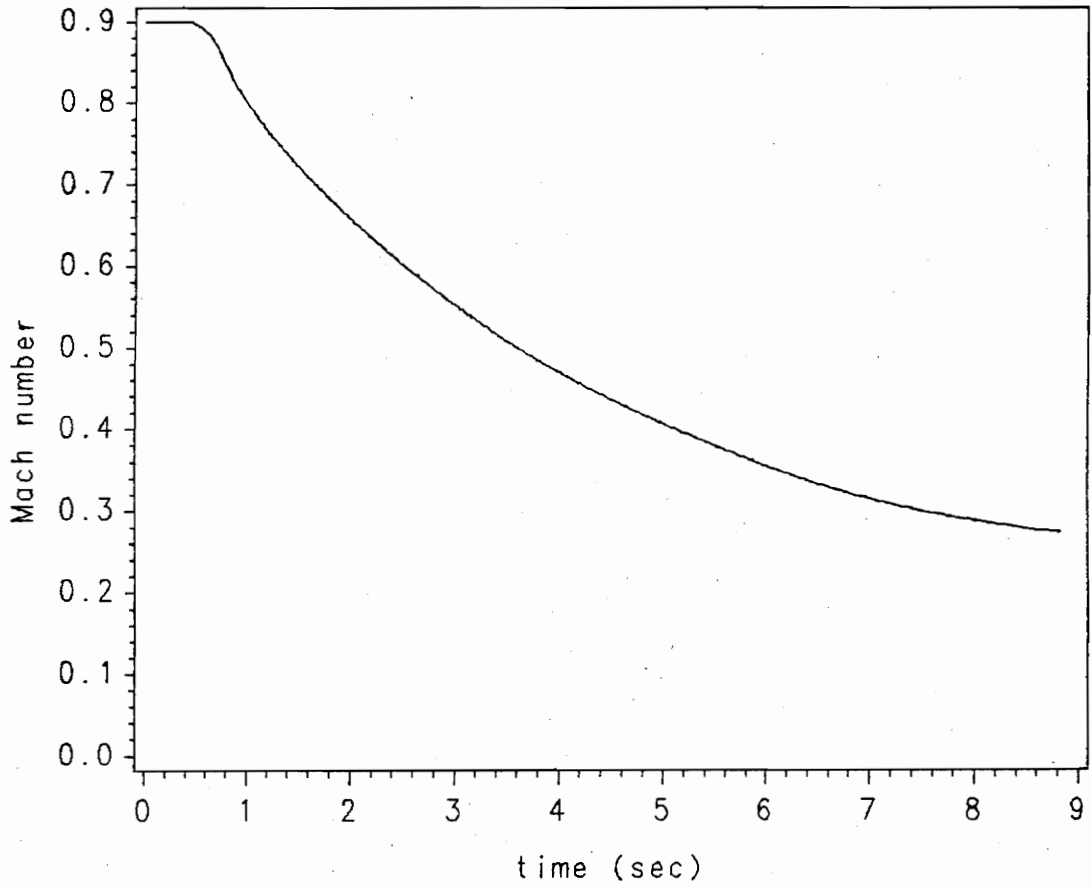


Figure 37. Half-Loop Simulation: Mach Number ($M(t=0)=0.9$, $h=15000$ ft)

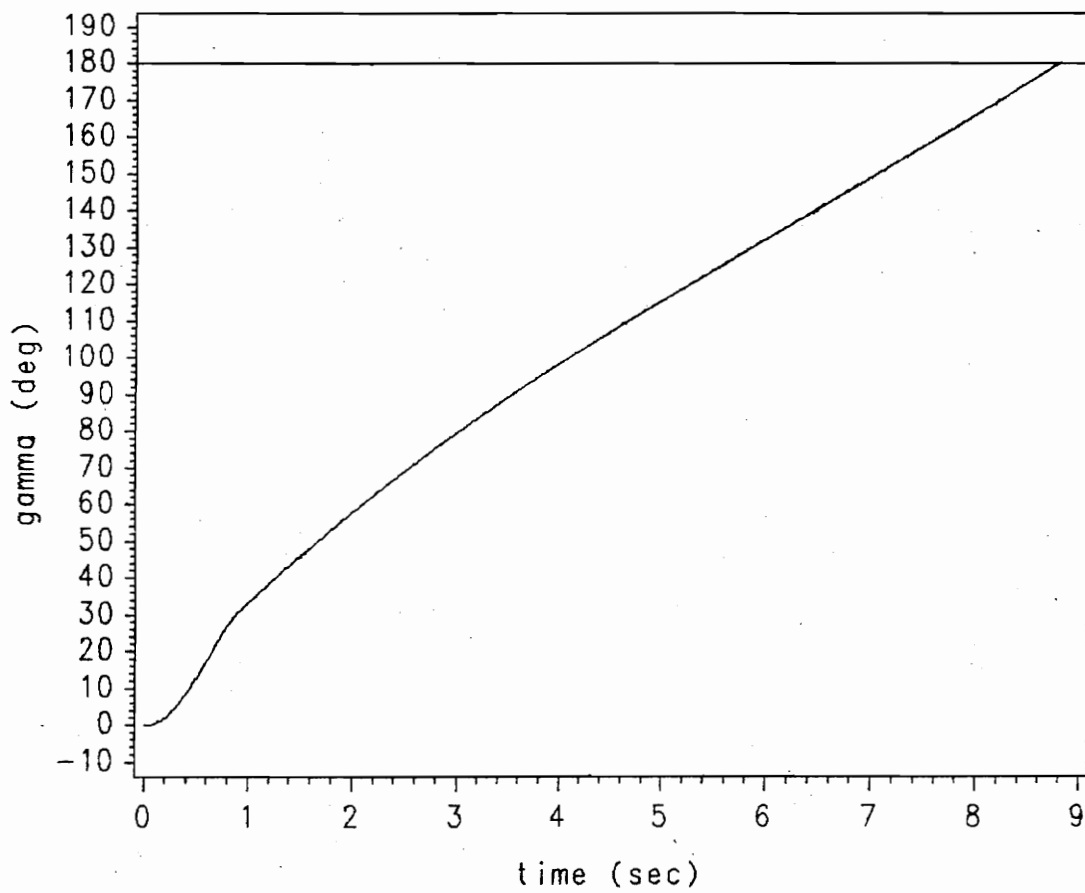


Figure 38. Half-Loop Simulation: Flight Path Angle ($M(t=0)=0.9$, $h=15000$ ft)

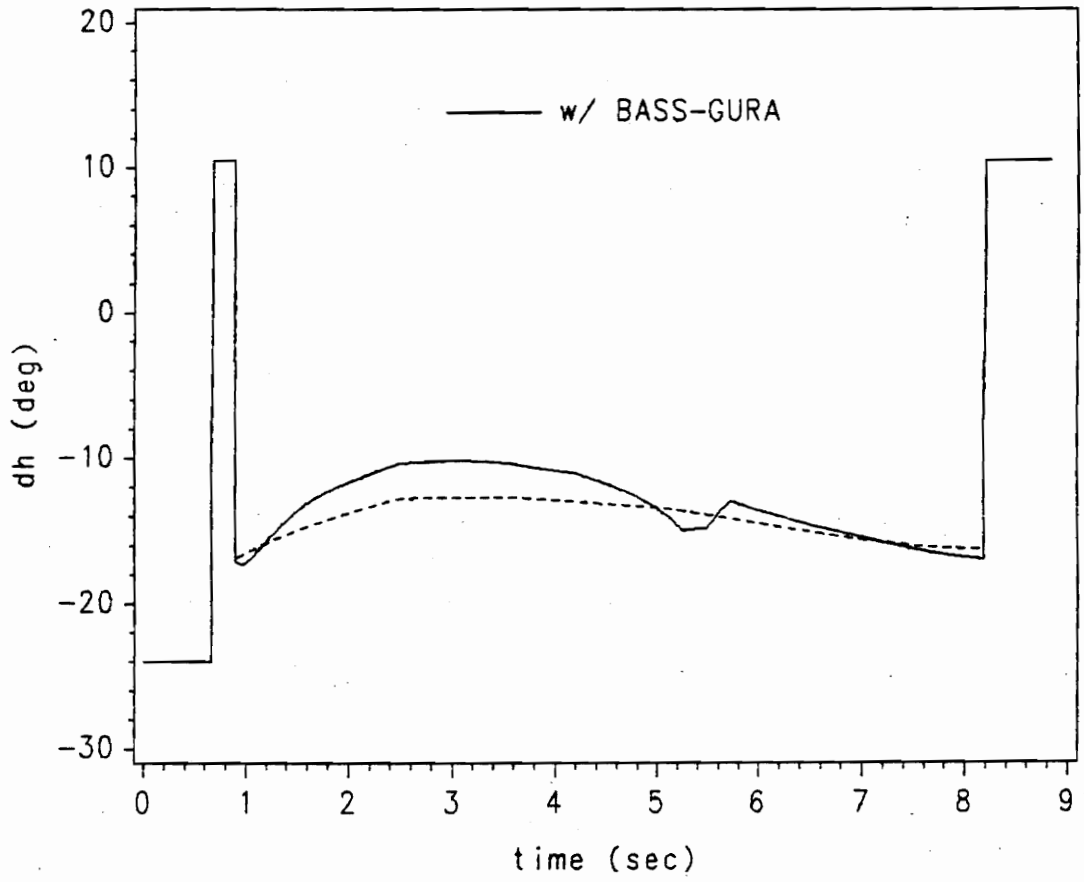


Figure 39. Half-Loop Simulation: Stabilator Angle ($M(t=0)=0.9$, $h=15000$ ft)

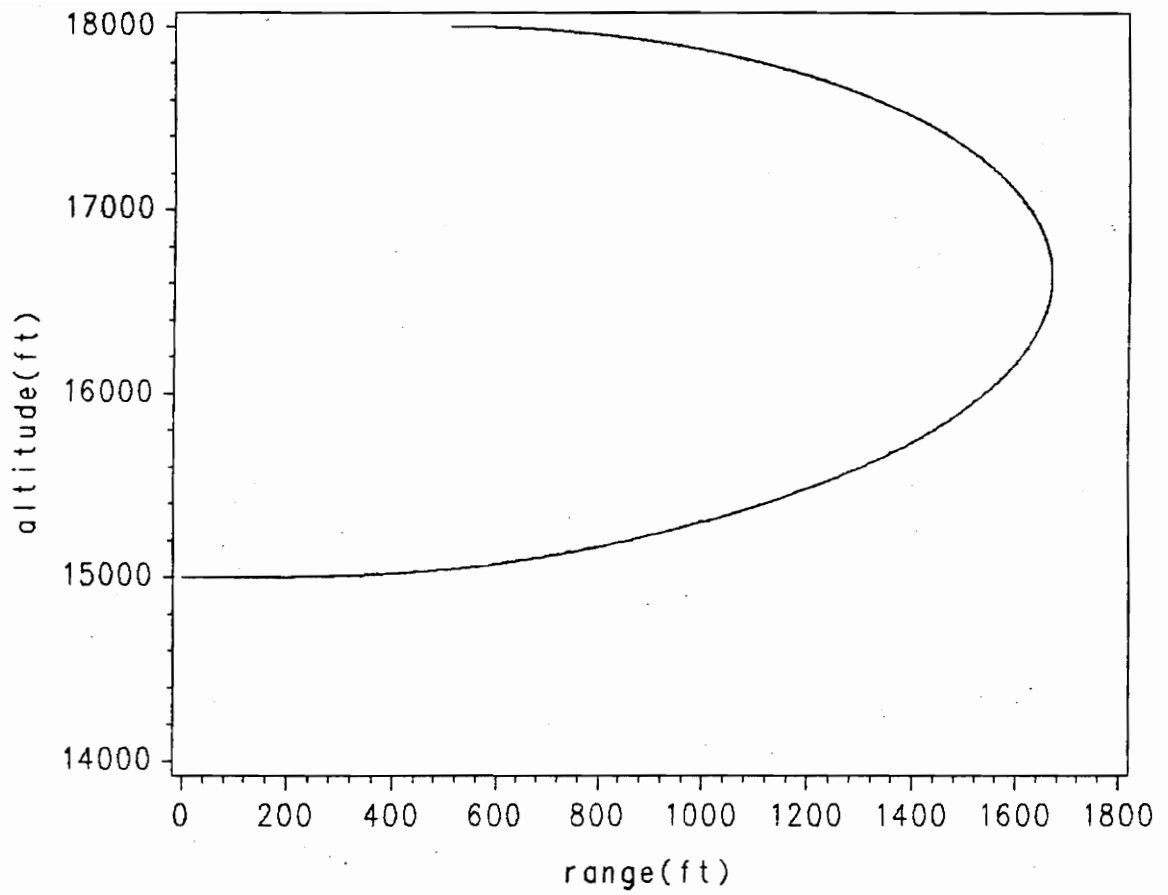


Figure 40. Half-Loop Simulation: Altitude vs Range ($M(t=0)=0.9$, $h=15000$ ft)

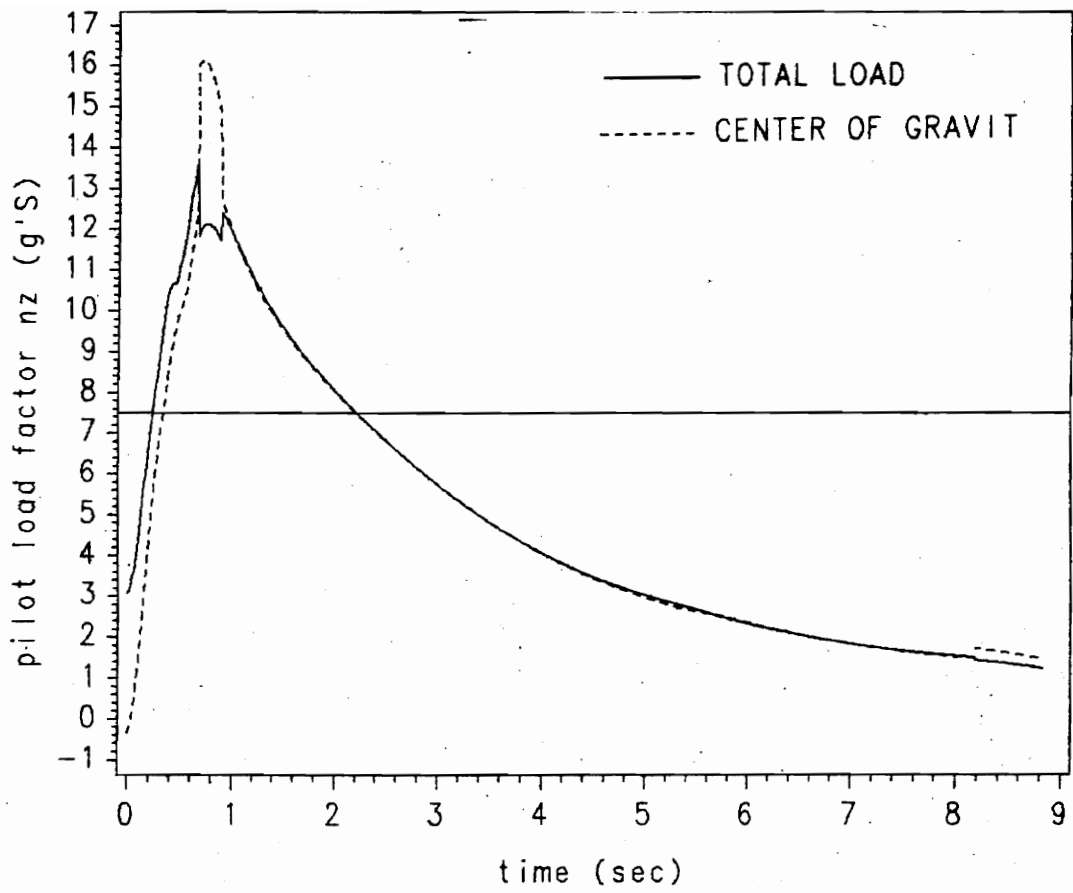


Figure 41. Half-Loop Simulation: Load Factor ($M(t=0)=0.9$, $h=15000$ ft)

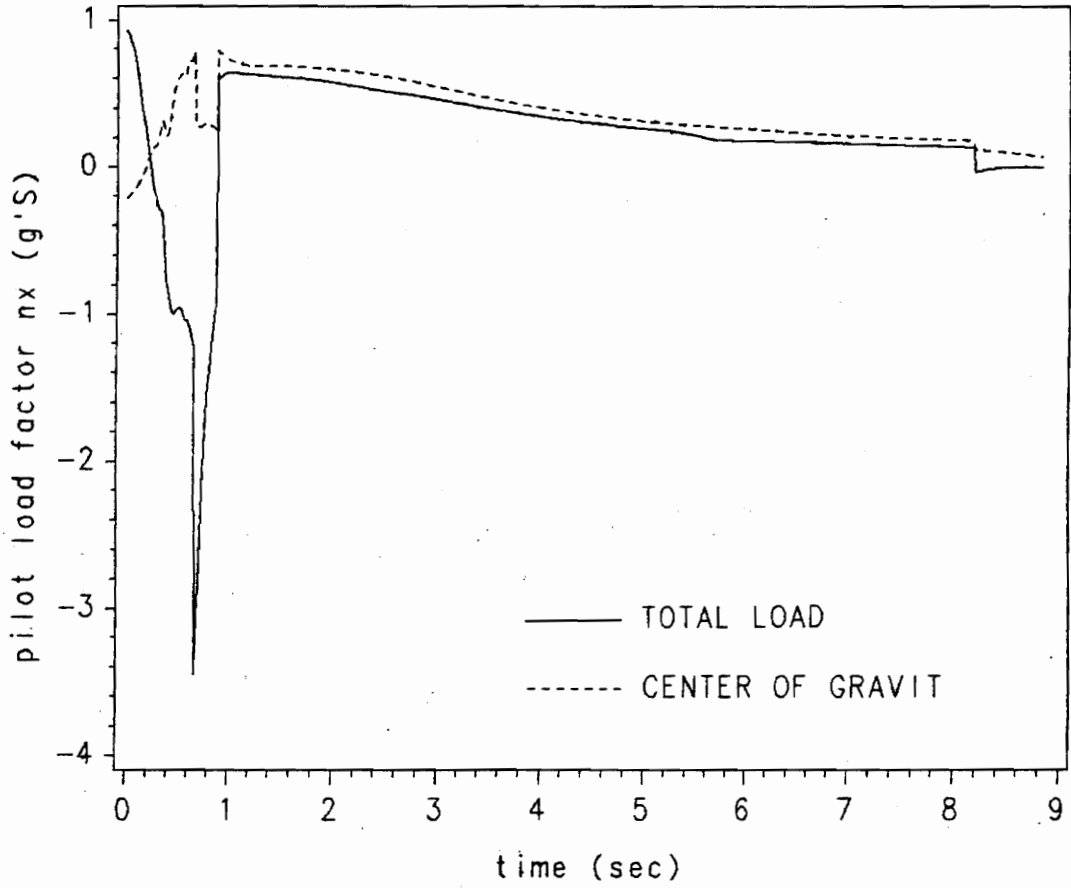


Figure 42. Half-Loop Simulation: Load Factor ($M(t=0)=0.9$, $h=15000$ ft)

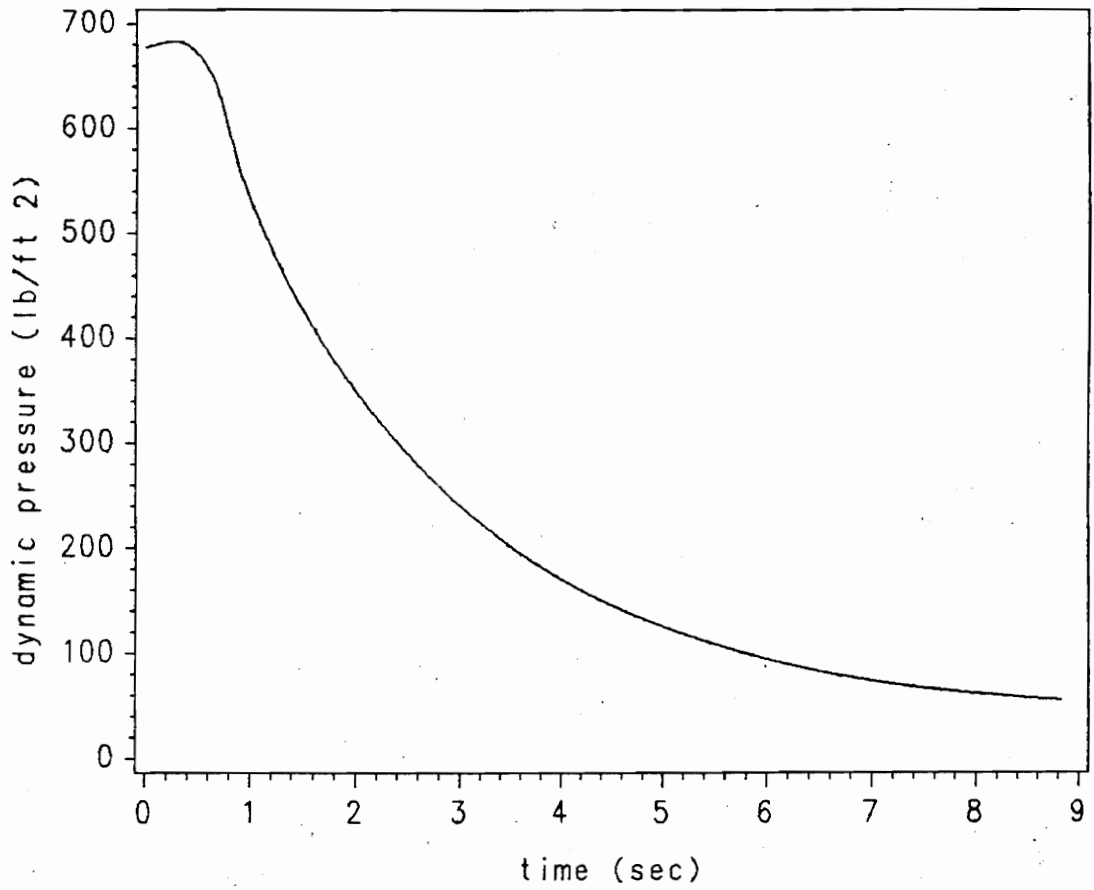


Figure 43. Half-Loop Simulation: Dynamic Pressure ($M(t=0)=0.9$, $h=15000$ ft)

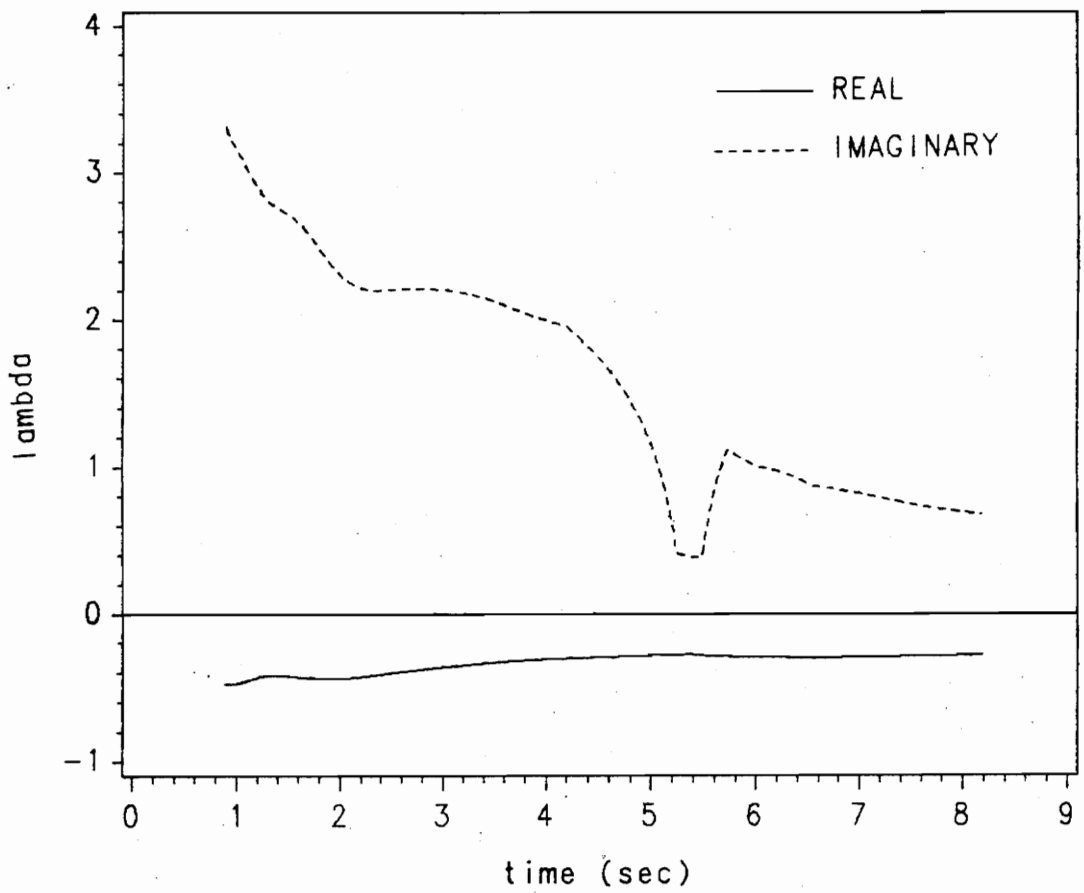


Figure 44. Half-Loop Simulation: Fast System Eigenvalues ($M(t=0) = 0.9$, $h = 15000$ ft)

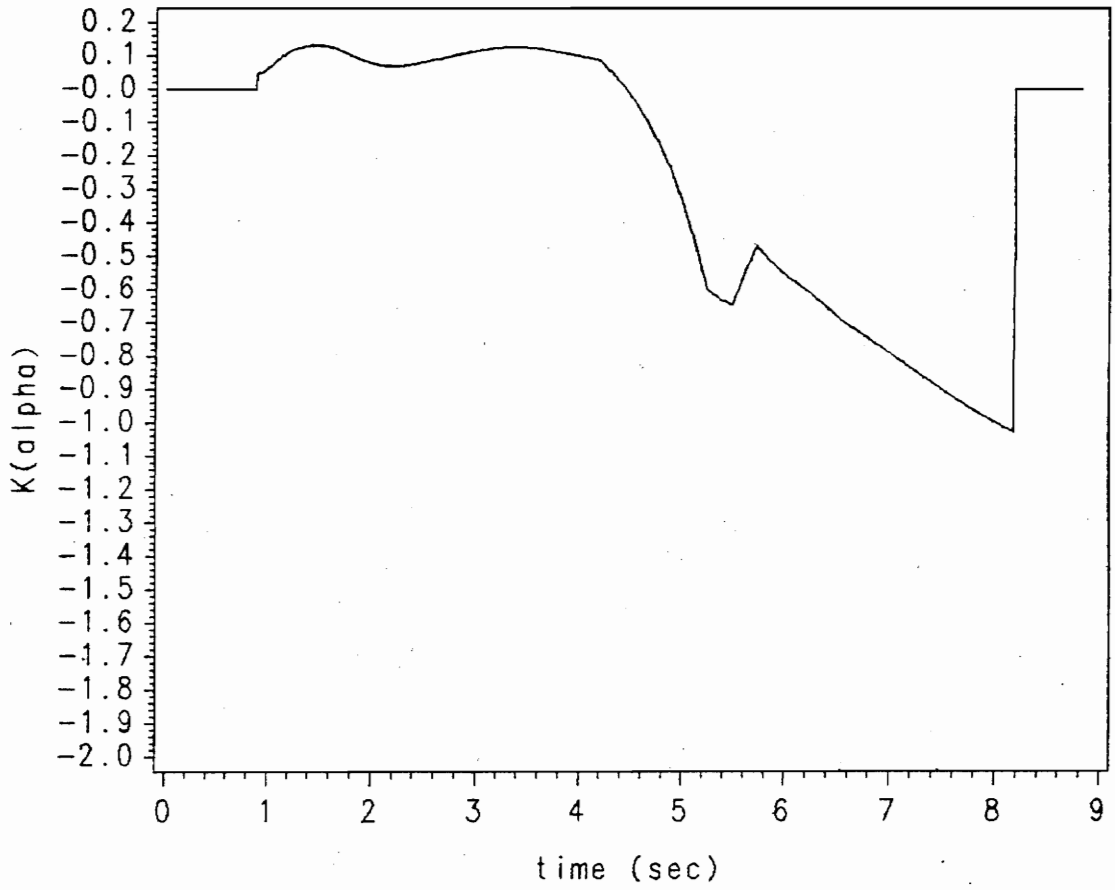


Figure 45. Half-Loop Simulation: Pole Placement Gain ($M(t=0)=0.9$, $h=15000$ ft)

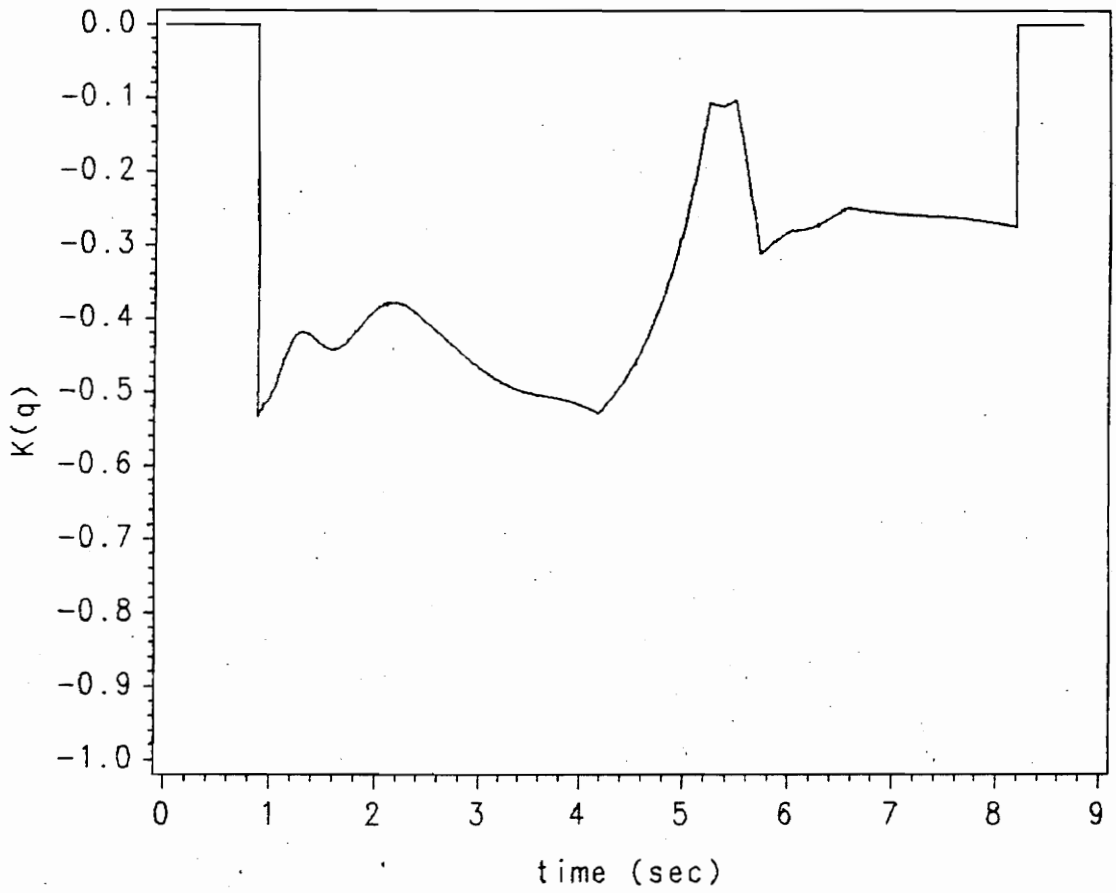


Figure 46. Half-Loop Simulation: Pole Placement Gain ($M(t=0)=0.9$, $h=15000$ ft)

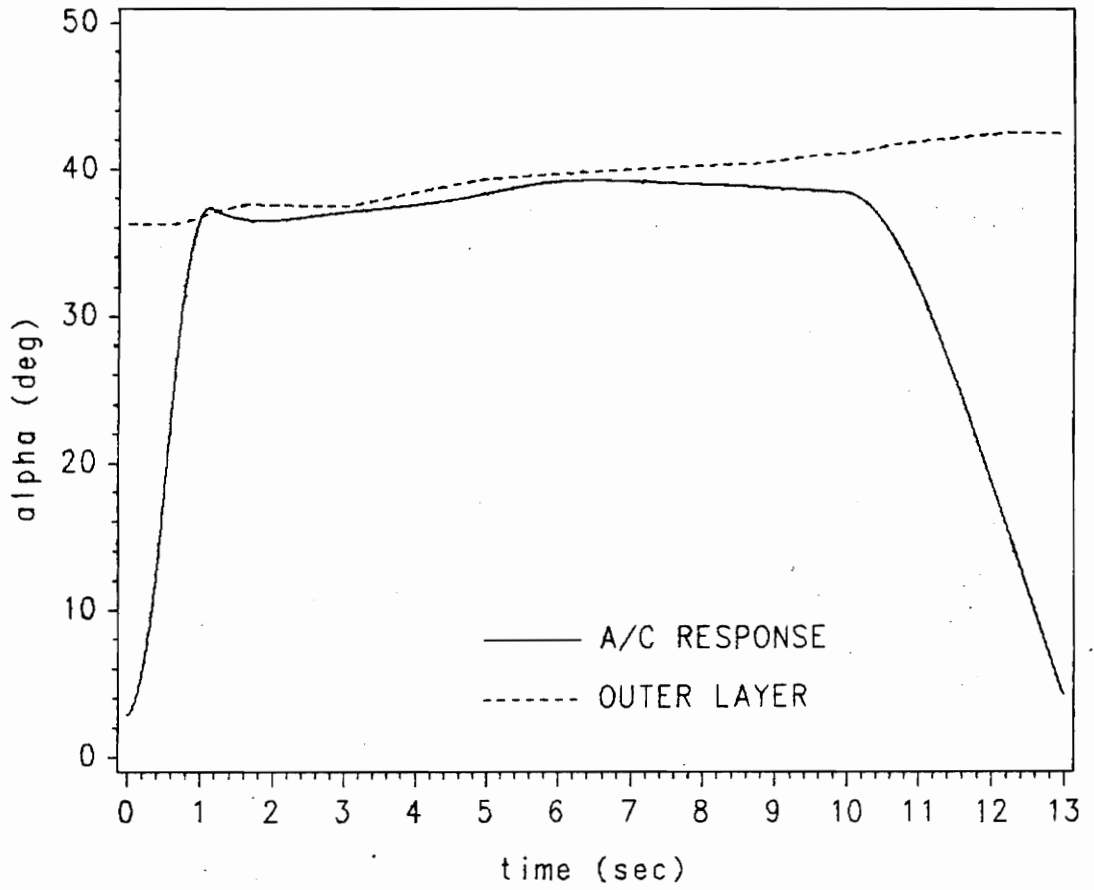


Figure 47. Half-Loop Simulation: Angle of Attack ($M(t=0)=0.6$, $h=15000$ ft)

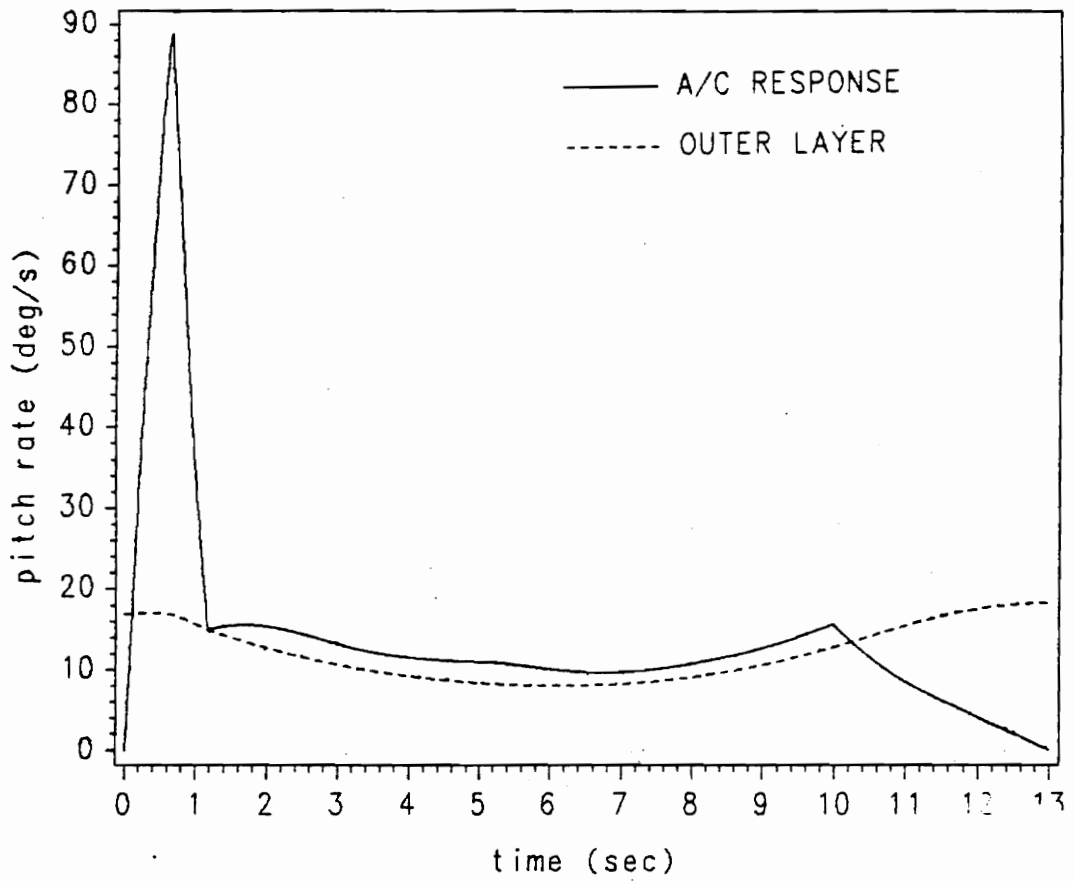


Figure 48. Half-Loop Simulation: Pitch Rate ($M(t=0)=0.6$, $h=15000$ ft)

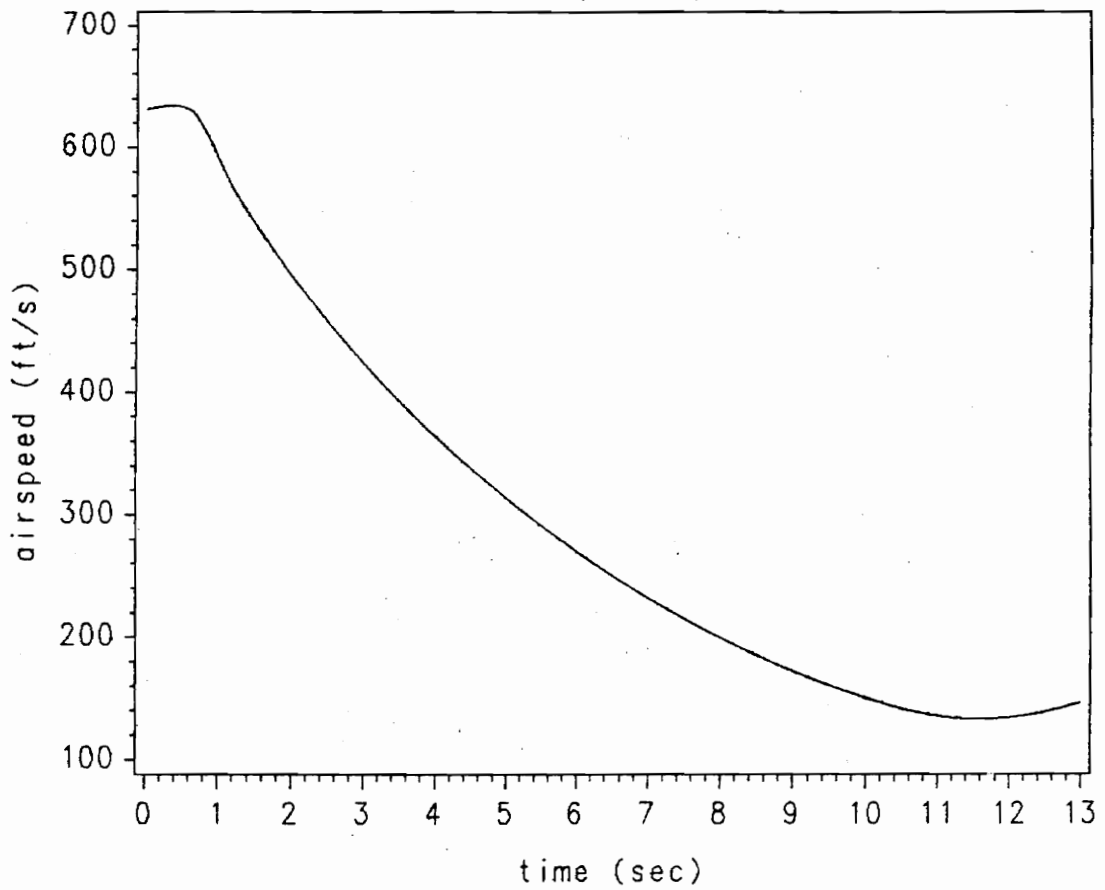


Figure 49. Half-Loop Simulation: Airspeed ($M(t=0)=0.6$, $h=15000$ ft)

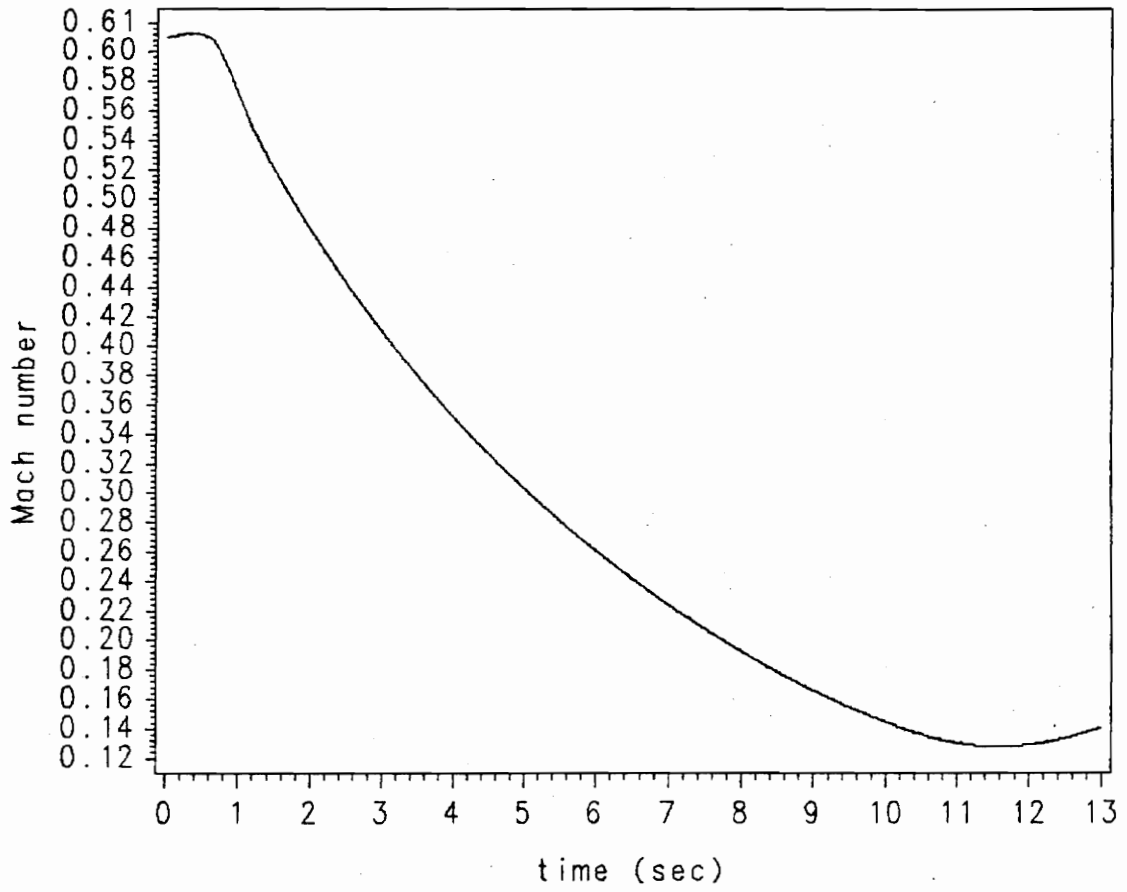


Figure 50. Half-Loop Simulation: Mach Number ($M(t=0) = 0.6$, $h = 15000$ ft)

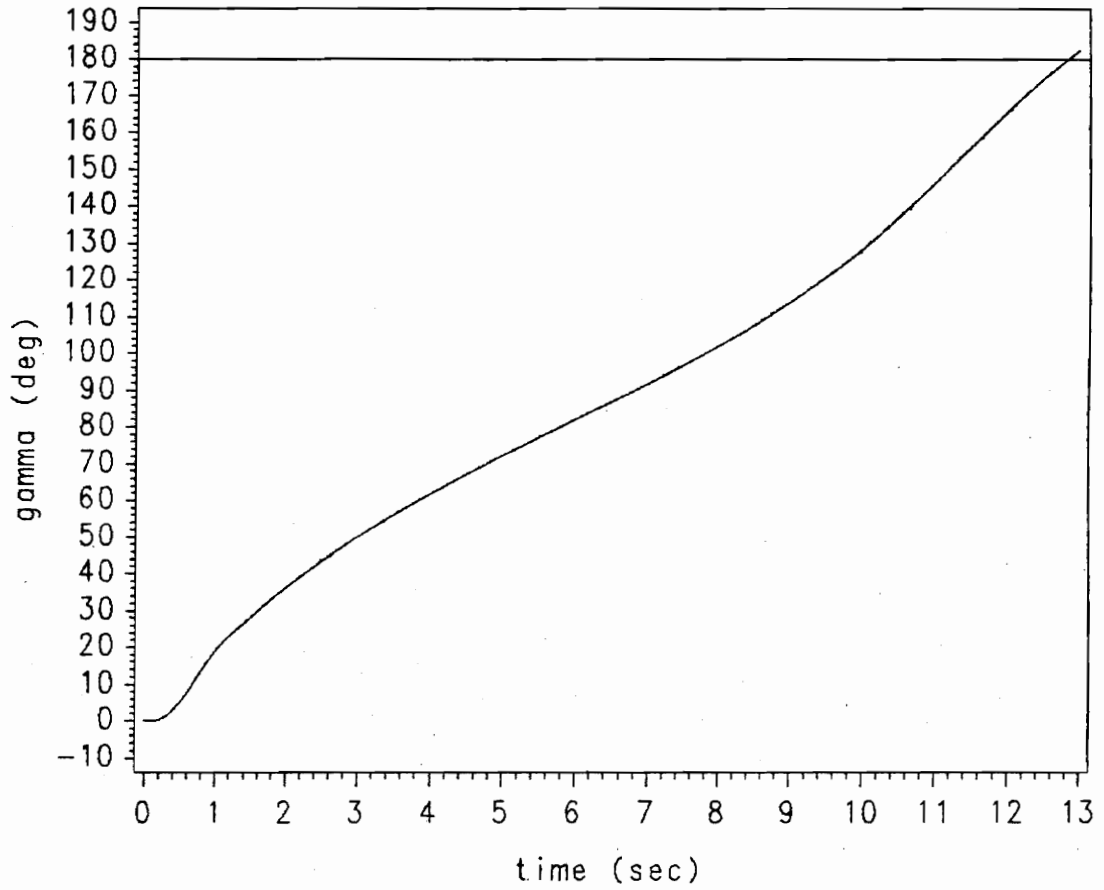


Figure 51. Half-Loop Simulation: Flight Path Angle ($M(t=0)=0.6$, $h=15000$ ft)

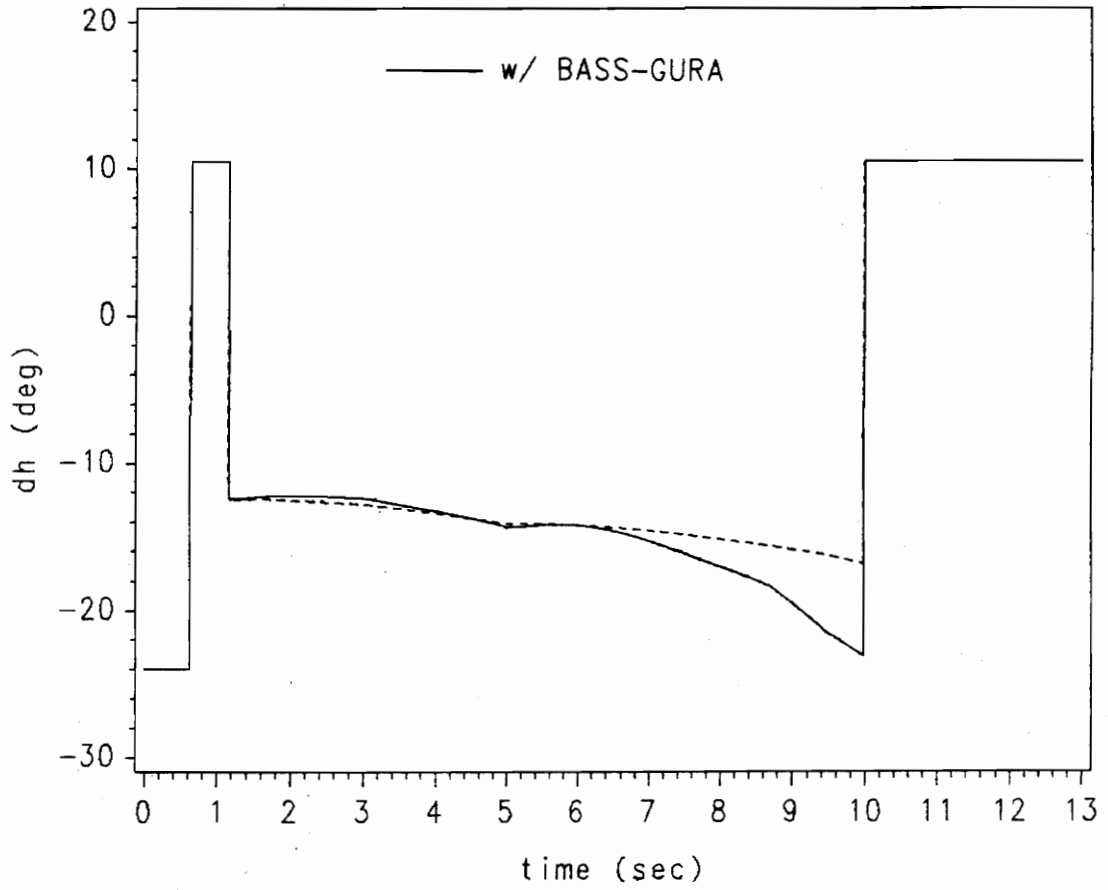


Figure 52. Half-Loop Simulation: Stabilator Angle ($M(t=0)=0.6$, $h=15000$ ft)

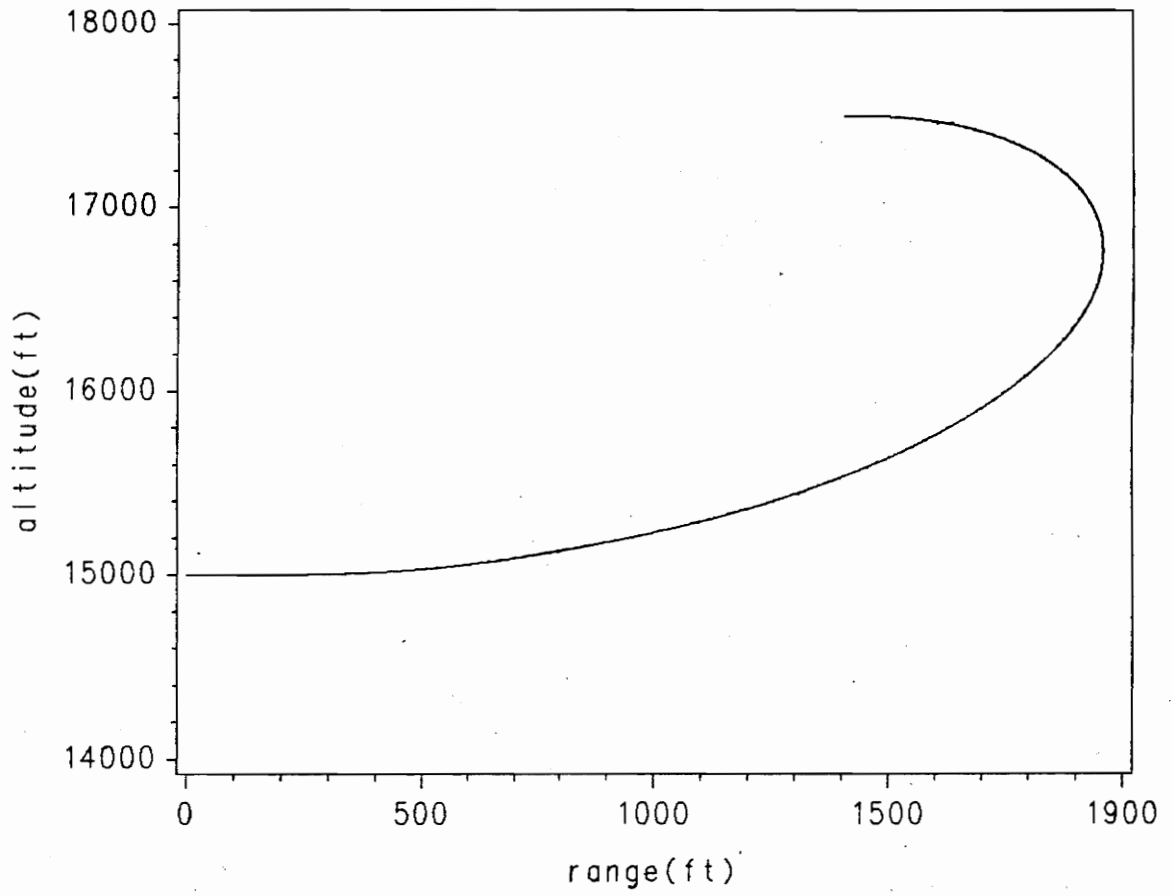


Figure 53. Half-Loop Simulation: Altitude vs Range ($M(t=0)=0.6$, $h=15000$ ft)

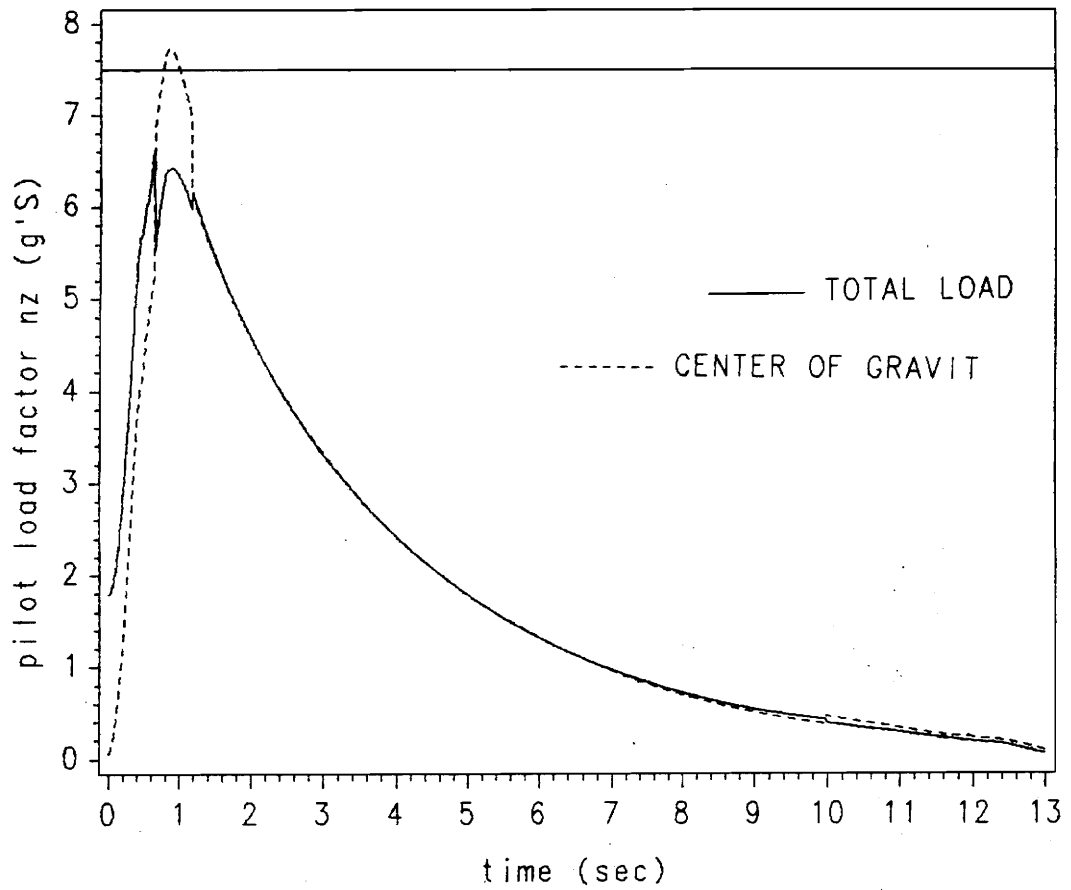


Figure 54. Half-Loop Simulation: Load Factor ($M(t=0)=0.6$, $h=15000$ ft)

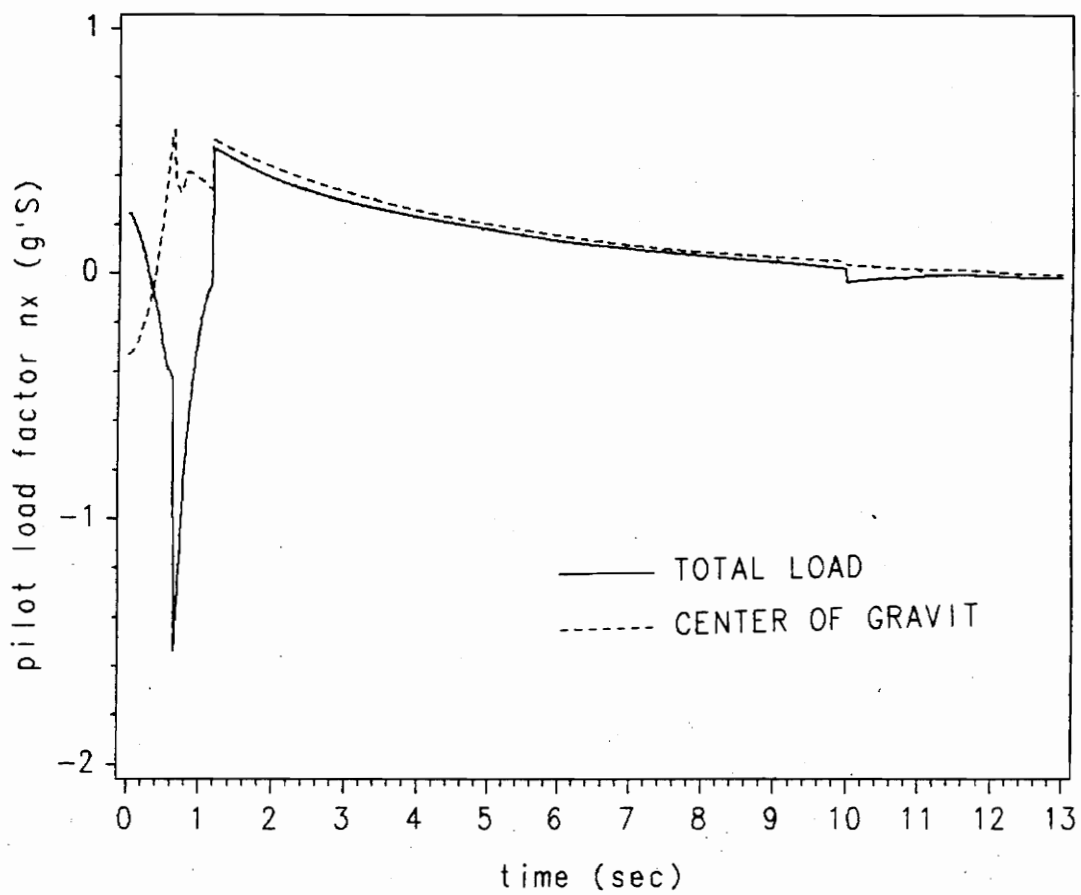


Figure 55. Half-Loop Simulation: Load Factor ($M(t=0)=0.6$, $h=15000$ ft)

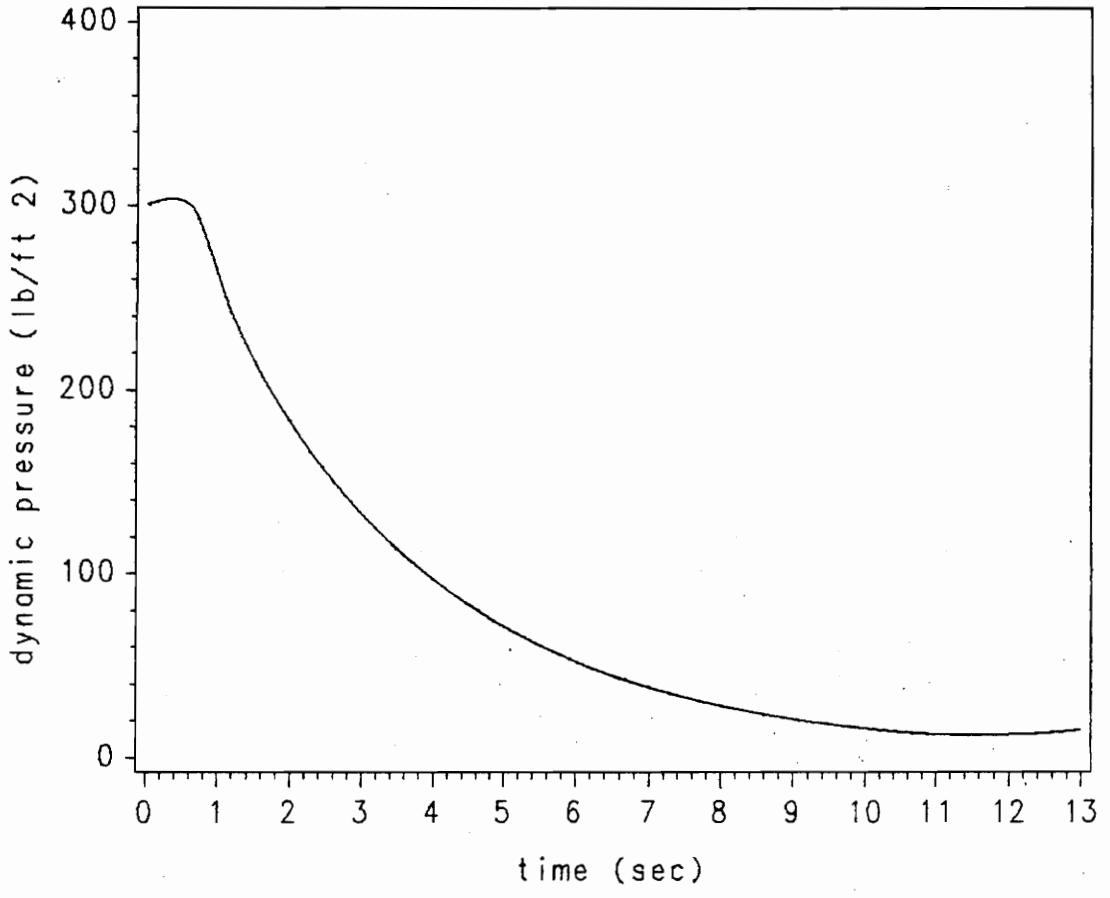


Figure 56. Half-Loop Simulation: Dynamic Pressure ($M(t=0)=0.6$, $h=15000$ ft)

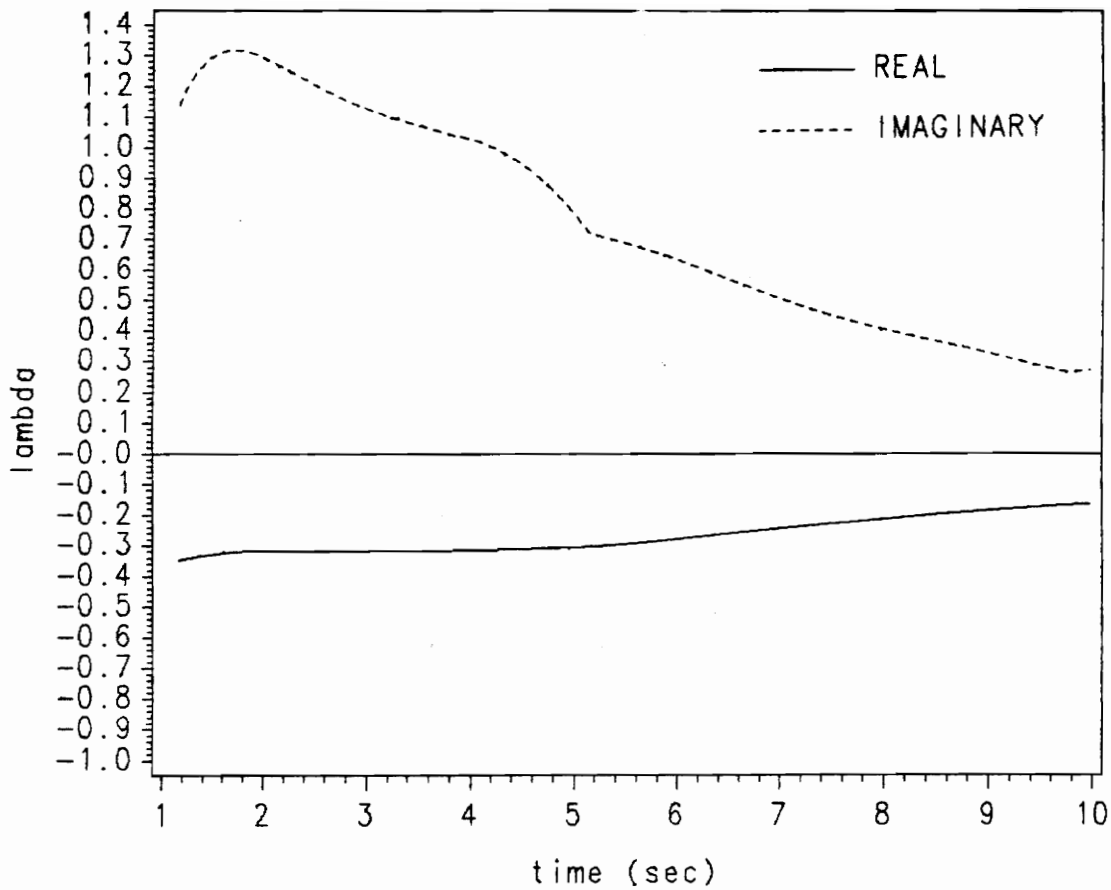


Figure 57. Half-Loop Simulation: Fast System Eigenvalues ($M(t=0)=0.6$, $h=15000$ ft)

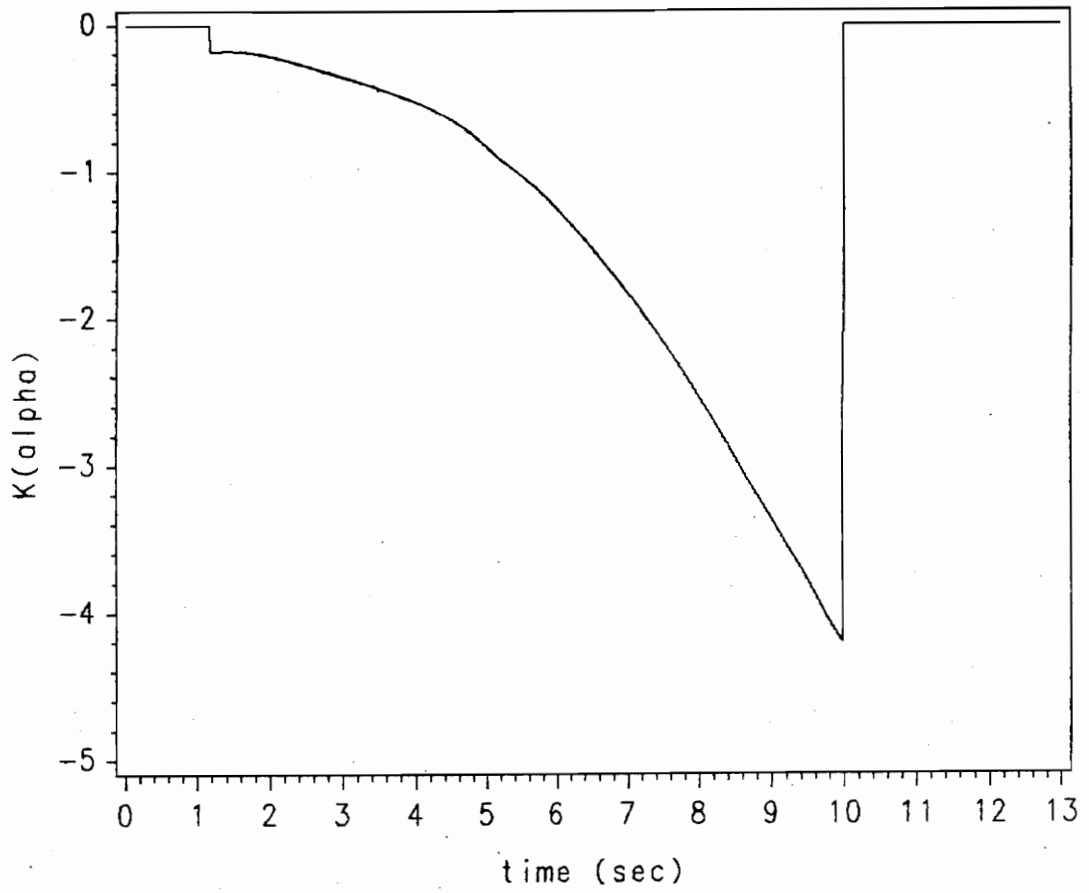


Figure 58. Half-Loop Simulation: Pole Placement Gain ($M(t=0) = 0.6$, $h = 15000$ ft)

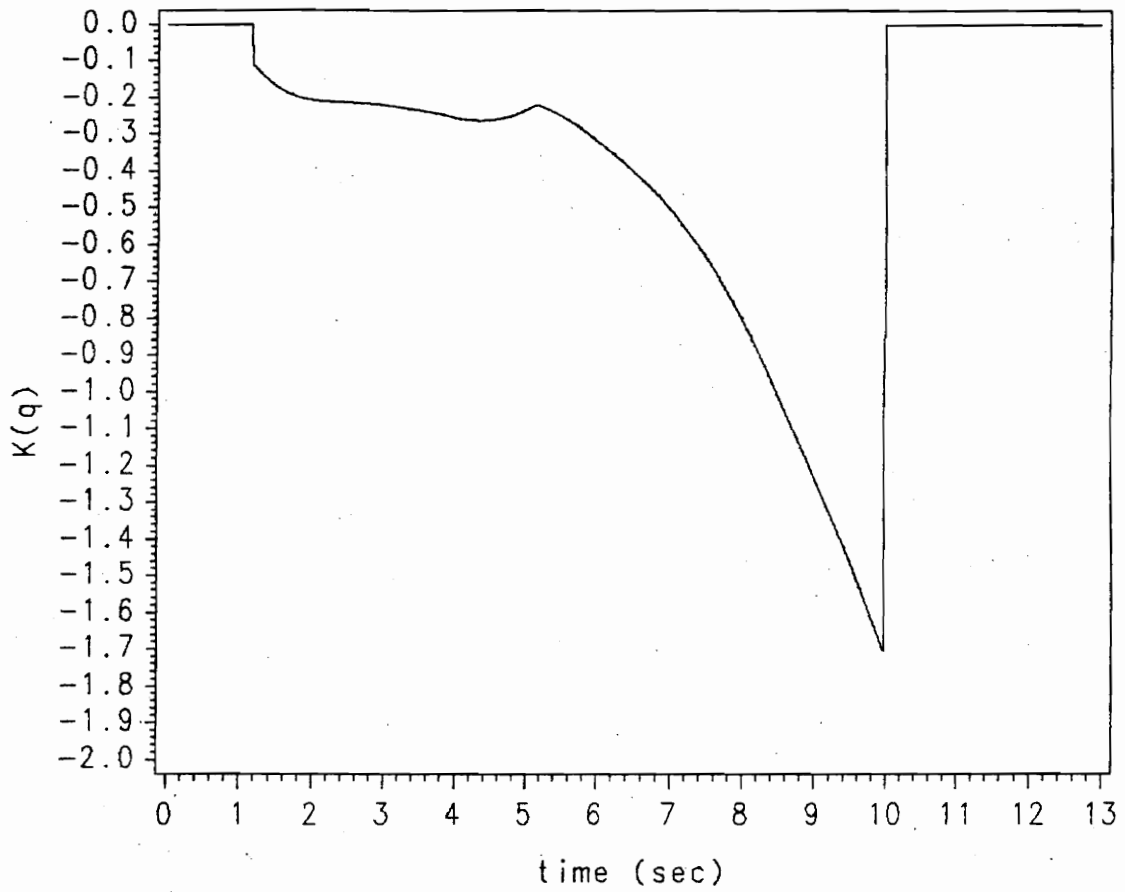


Figure 59. Half-Loop Simulation: Pole Placement Gain ($M(t=0)=0.6$, $h=15000$ ft)

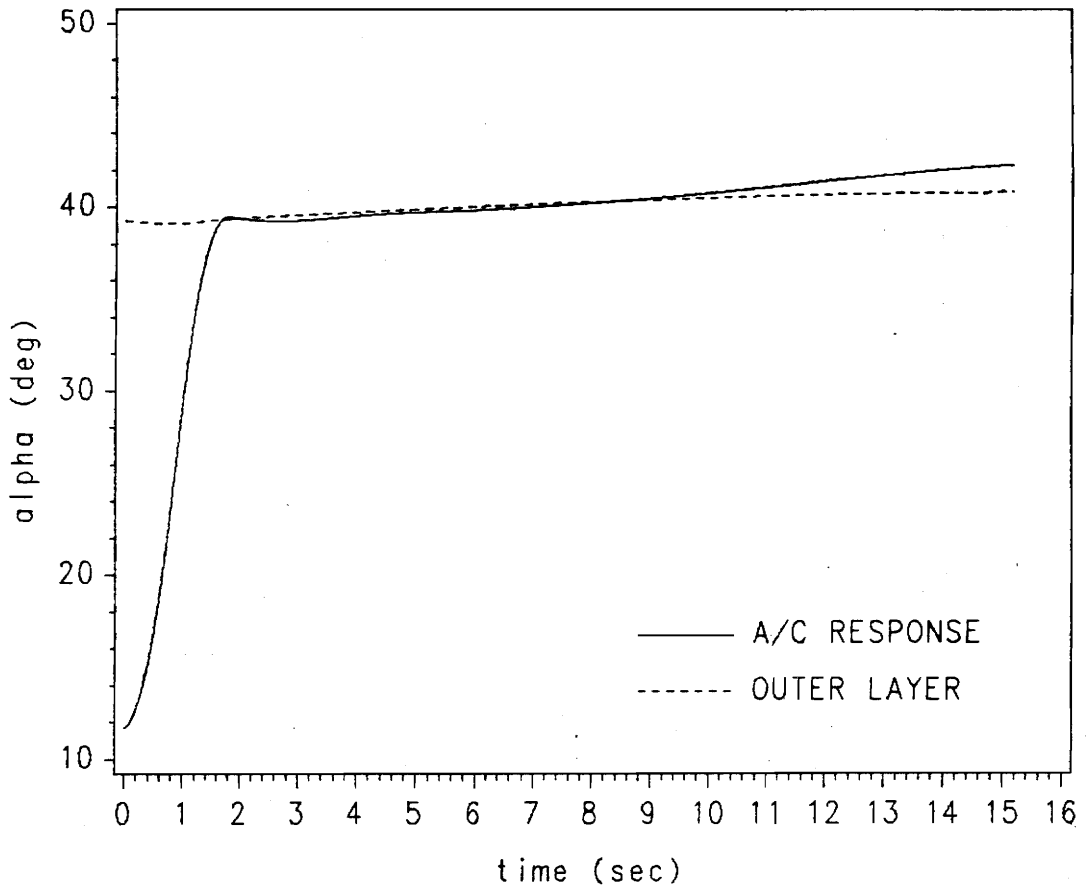


Figure 60. Half-Loop Simulation: Angle of Attack ($M(t=0)=0.3$, $h=15000$ ft)

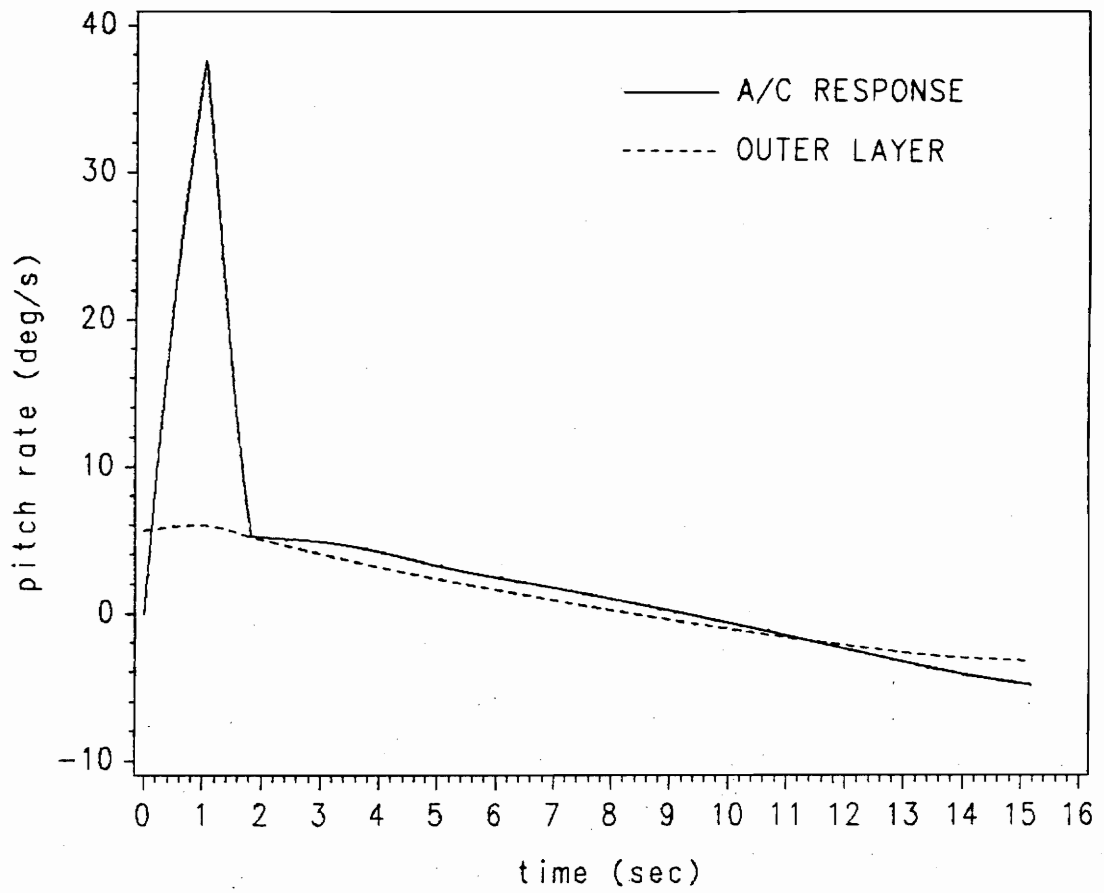


Figure 61. Half-Loop Simulation: Pitch Rate ($M(t=0)=0.3$, $h=15000$ ft)

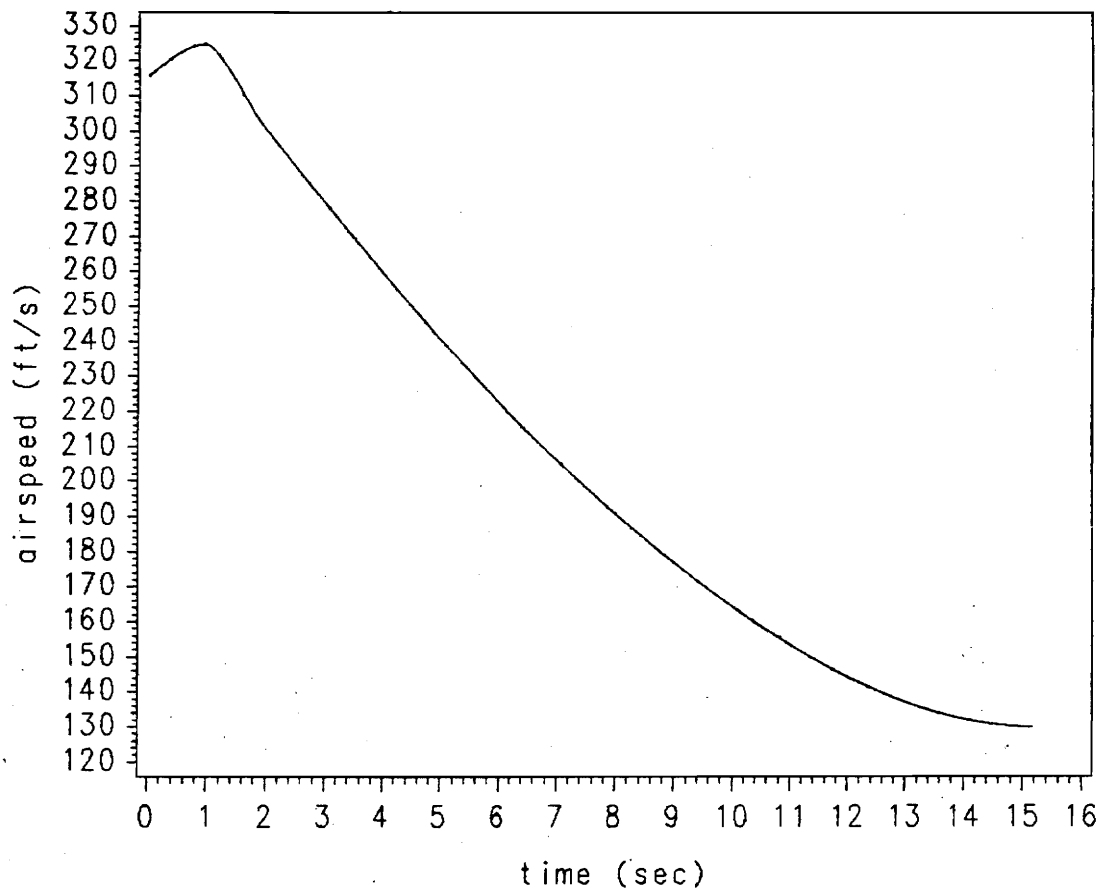


Figure 62. Half-Loop Simulation: Airspeed ($M(t=0)=0.3$, $h=15000$ ft)

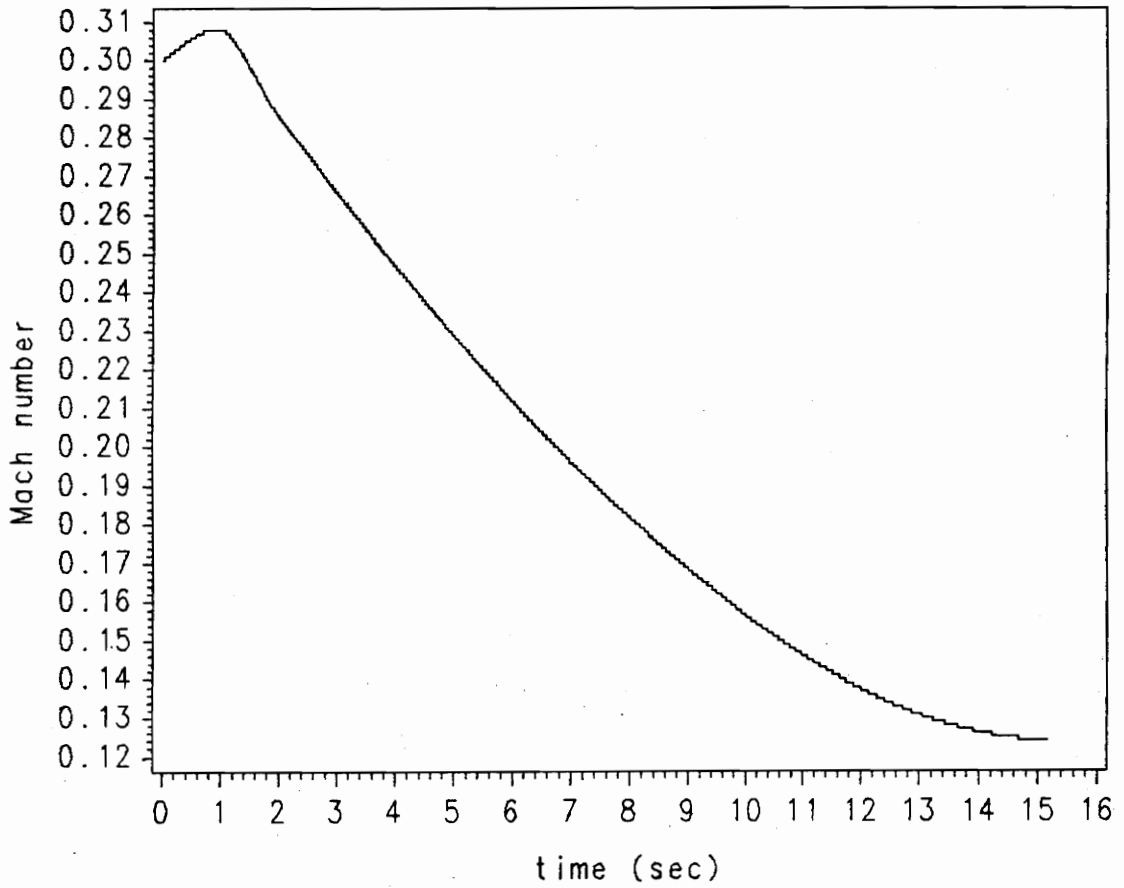


Figure 63. Half-Loop Simulation: Mach Number ($M(t=0)=0.3$, $h=15000$ ft)

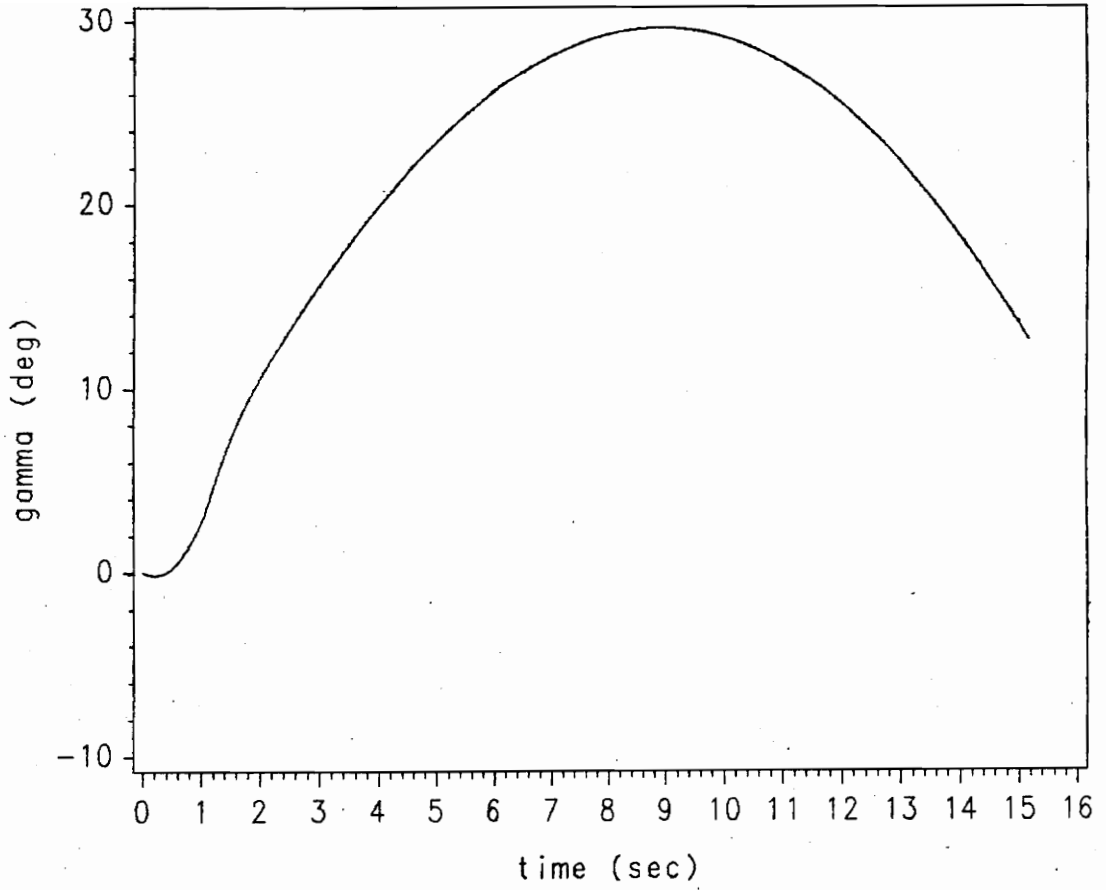


Figure 64. Half-Loop Simulation: Flight Path Angle ($M(t=0)=0.3$, $h=15000$ ft)

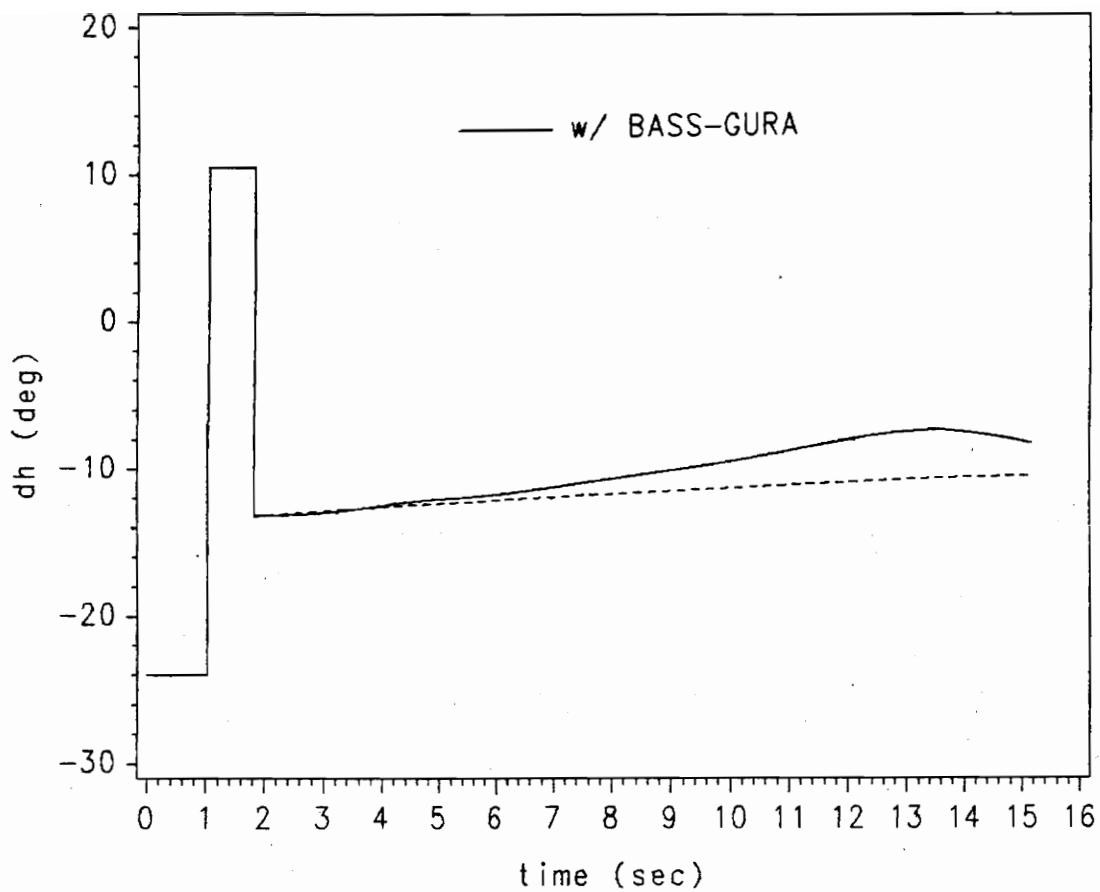


Figure 65. Half-Loop Simulation: Stabilator Angle ($M(t=0)=0.3$, $h=15000$ ft)

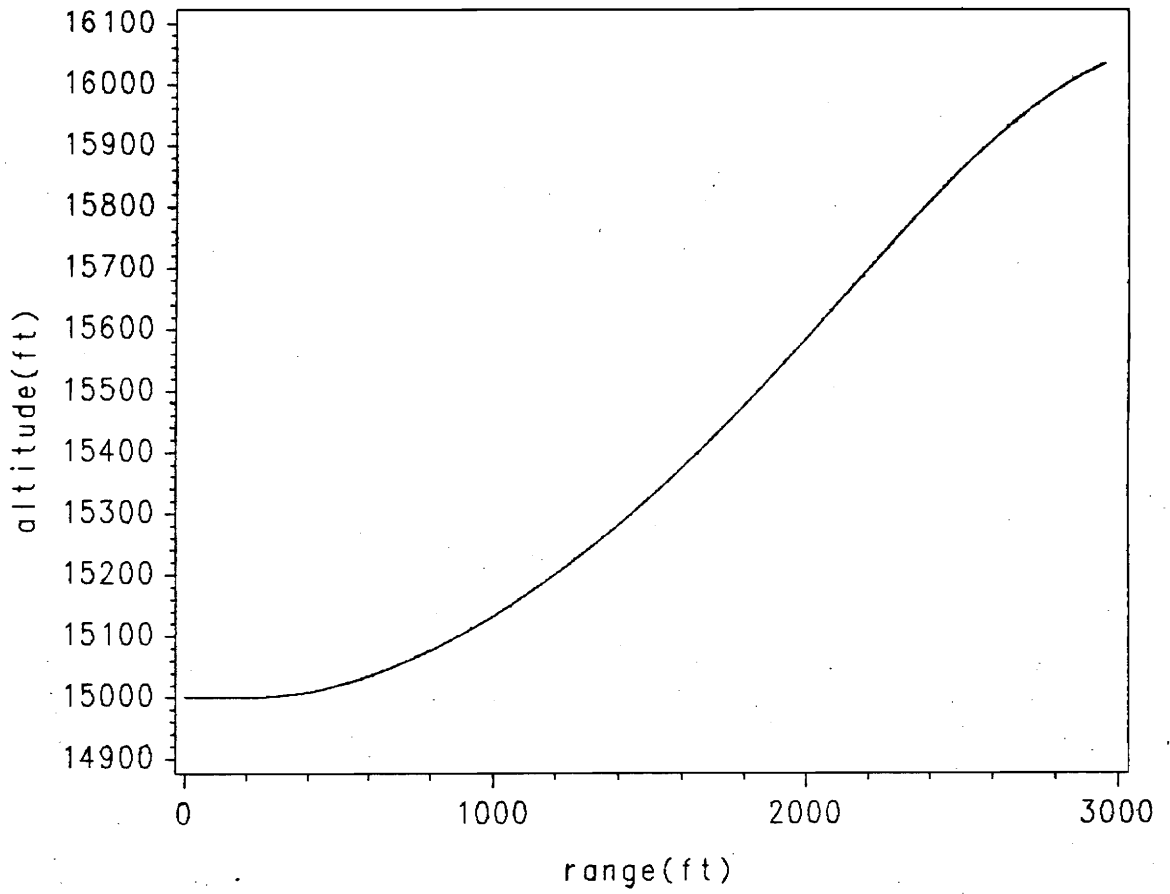


Figure 66. Half-Loop Simulation: Altitude vs Range ($M(t=0)=0.3$, $h=15000$ ft)

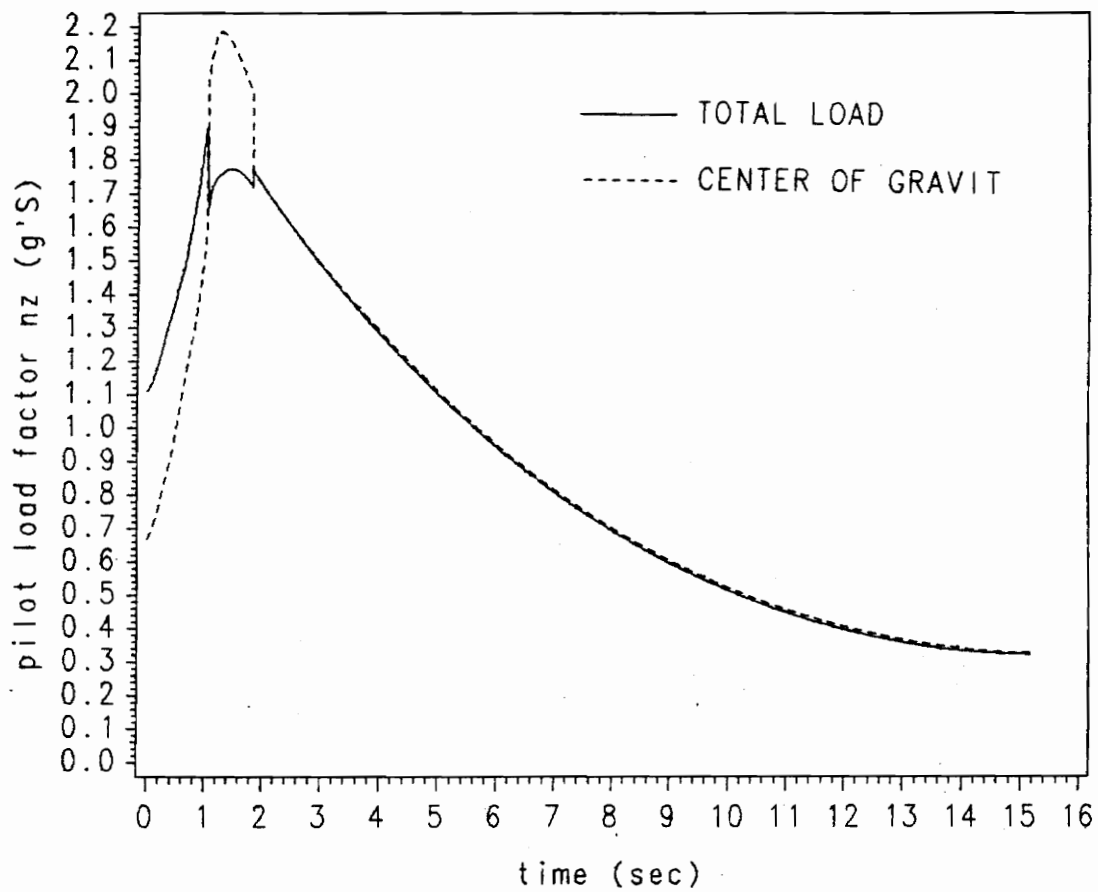


Figure 67. Half-Loop Simulation: Load Factor ($M(t=0)=0.3$, $h=15000$ ft)

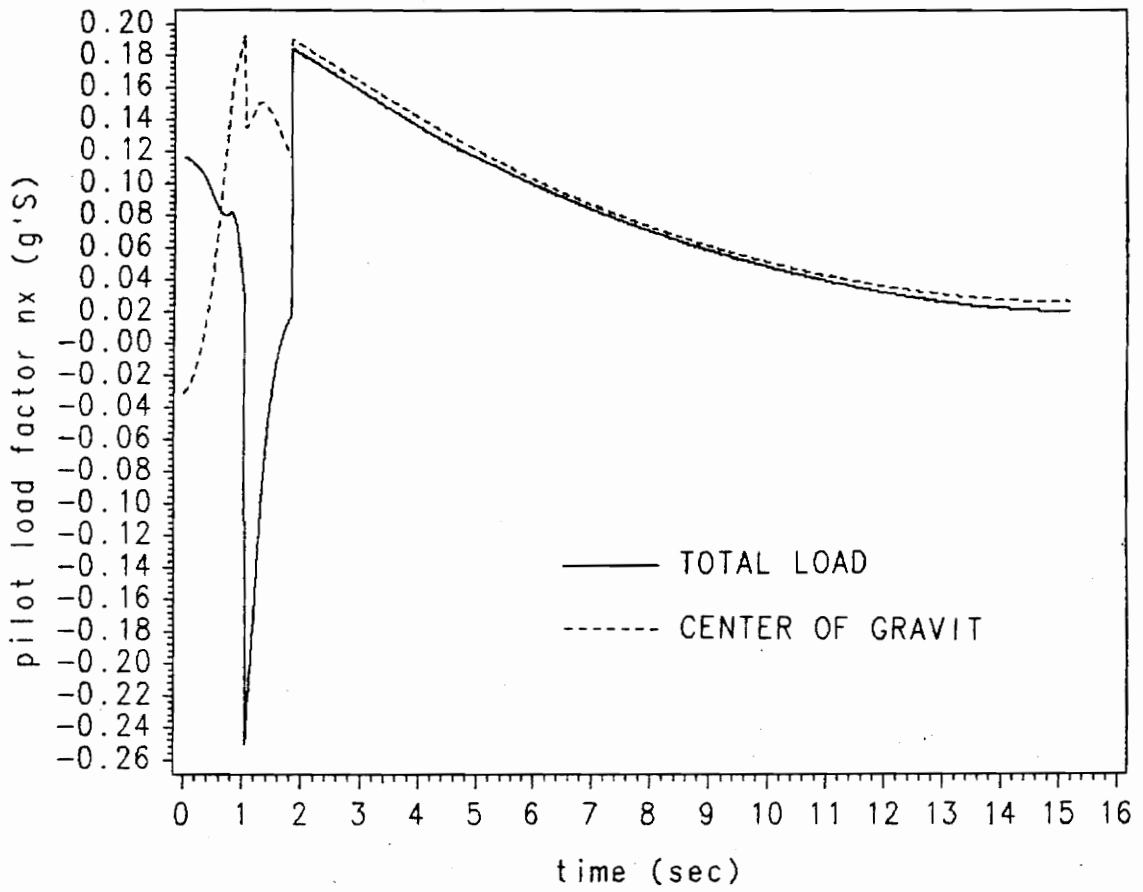


Figure 68. Half-Loop Simulation: Load Factor ($M(t=0)=0.3$, $h=15000$ ft)

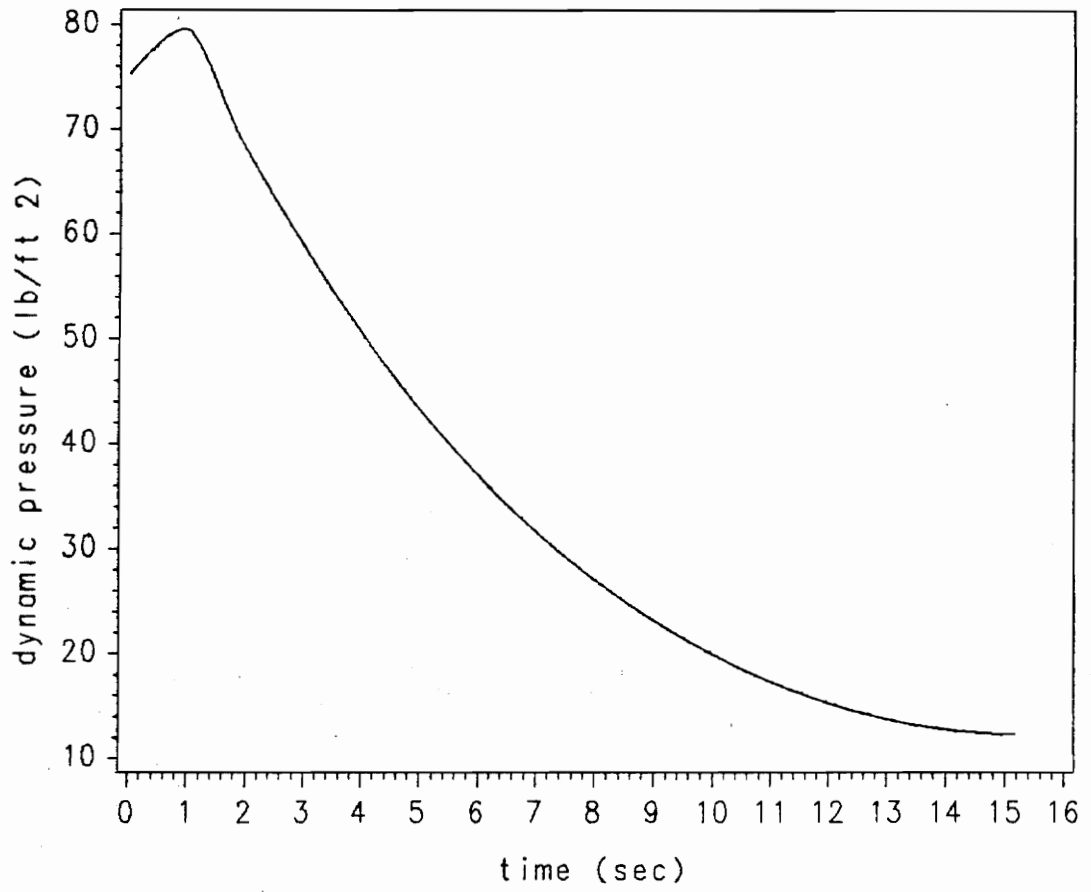


Figure 69. Half-Loop Simulation: Dynamic Pressure ($M(t=0)=0.3$, $h=15000$ ft)

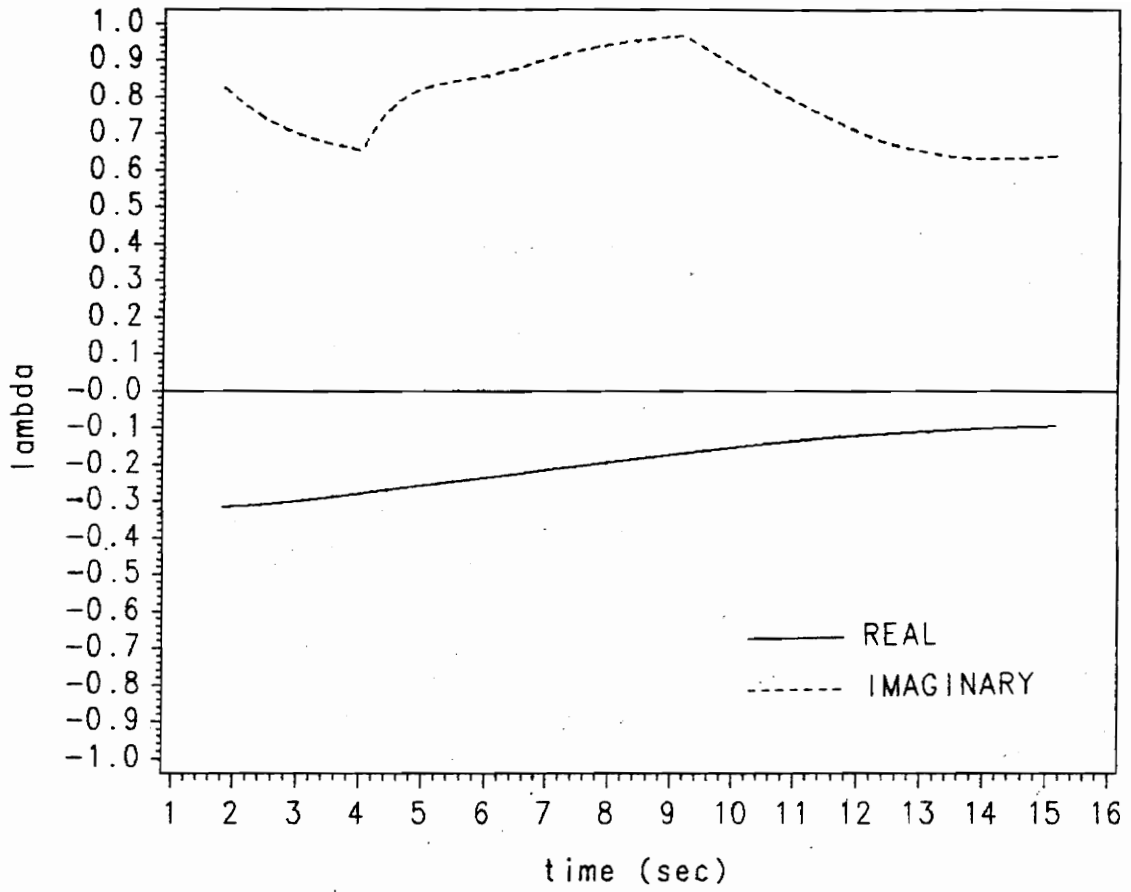


Figure 70. Half-Loop Simulation: Fast System Eigenvalues ($M(t=0)=0.3$, $h=15000$ ft)

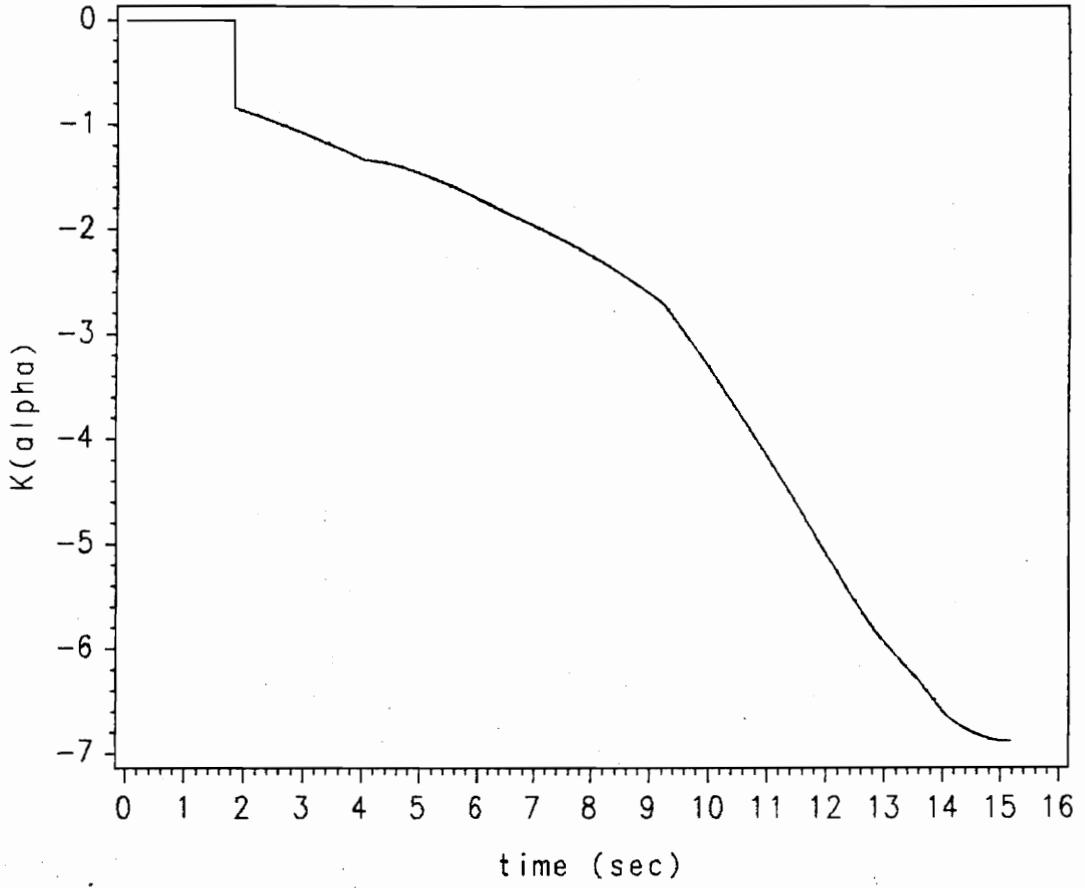


Figure 71. Half-Loop Simulation: Pole Placement Gain ($M(t=0)=0.3$, $h=15000$ ft)

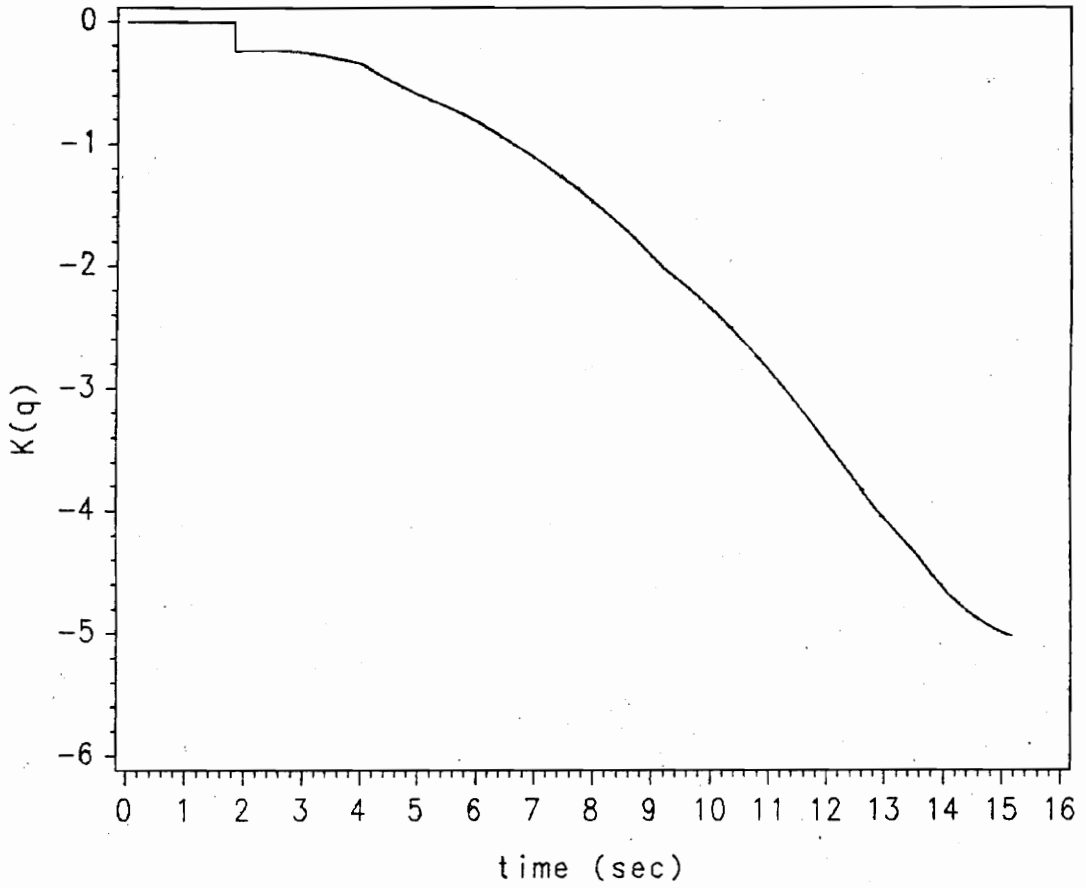


Figure 72. Half-Loop Simulation: Pole Placement Gain ($M(t=0)=0.3$, $h=15000$ ft)

twenty-five (25) degrees per second. The outer layer portion of the solution requires around seven and one half (7.5) seconds. We observe that the simulation does not follow exactly the outer layer values for the angle of attack and the pitch rate. Figure 39 gives the time history of the stabilator deflection. The bang-bang nature of the first transition region is evident. The difference between the solid and dotted lines show the effect of the pole placement on the stabilator feedback. The end of the maneuver consists of a final bang control which can be seen here. The second transition region only takes about one half (0.5) second. Figure 40 gives a representation of the path of the maneuver in x-y space. This figure is not to the same scale in both the x and y directions and therefore the relative altitude and range changes are skewed. However, it is possible to see that the conclusion of the maneuver is downrange from the beginning. The evolution of the flight path angle from zero (0) degrees to one hundred and eighty (180) degrees is given in figure 38.

Two significant effects involved in this maneuver can be seen in figures 36 - 37 and 41 - 42. Figures 36 and 37 show the large drain in energy involved in this maneuver. The airspeed and Mach numbers are significantly decreased over the course of the maneuver. The dynamic pressure in figure 43 also shows this effect. This condition is one reason why pilots will often avoid the half-loop maneuver in a combat situation. The small increase in Mach number and airspeed at the beginning of the maneuver is due to the fact that the thrust is not at an equilibrium value at the beginning of the maneuver. The second effect of interest is the large load factors encountered at the beginning of the maneuver. As can be seen in figure 41, the load factor in the z-body direction is over twice the allowable limit of seven and one half (7.5) g's. Another effect of interest is the large onset of the load factor in in the first transition region. The load factor does fall below the allowable limit for most of the maneuver primarily because of the large

amount of energy being lost. The pilot load factor, after the switch in the first transition region, is not as large as that of the center of gravity because the pitch rate is dropping off rapidly at that time. From this figure, we can see that the effect of constraining the maneuver below a prescribed load factor would primarily change the feedback law in the first transition region and at the beginning of the outer layer region. The load factor in the x-body direction, figure 42, does not show large effects which would be dangerous for the aircraft or the pilot.

The eigenvalues of the fast system are plotted in figure 44 in order to see how the pole placement has effected them. Figures 45 and 46 give the pole placement gains corresponding to the fast variables of angle of attack (α) and pitch rate (q). These gains where multiplied by the amount of divergence of the corresponding states from the outer layer values desired. They are equal to zero during the transition regions because pole placement is not used during these regions of the maneuver.

For the initial Mach number of 0.6, the maneuver requires approximately thirteen (13) seconds. In figures 47 and 48, we see that the angle of attack and the pitch rate follow the outer layer values better than for the Mach 0.9 case. The outer layer angle of attack is slightly larger as is the outer layer pitch rate. However, the pitch rate only reaches around ninety (90) degrees per second at its maximum point. The time required to execute the first transition region is about the same as in the previous case but the second transition region takes considerably longer. In figures 49 and 50, the same loss of energy associated with this maneuver is visible with the final Mach number going as low as 0.13.

The flight path angle in figure 51 exhibits results very similar to that of the Mach 0.9 case. This is true of the stabilator deflection in figure 52 and the flight trajectory plot in figure 53. The load factor in the z-body direction does not violate the pilot load factor

limitation but the rate at which the load factor increases during the first transition region is still a point of concern. Again, the load factor in the x-body direction is not of any great concern. Figures 56 - 59 indicate trends very similar to those discussed earlier for the initial Mach number of 0.9.

As can be seen in figures 60 - 72, the aircraft was unable to execute the half-loop maneuver for an initial Mach number of 0.3. This is not unexpected because the thrust-to-weight ratio (approximately 0.6) is much less than unity for this low Mach number. At Mach 0.3, the aircraft lacks the energy required to execute the half-loop maneuver. The NASA simulations that will be discussed later also indicate that the maneuver is not possible at this low initial Mach number. For this simulation, the results are plotted out to the point where it is obvious that the aircraft cannot execute the maneuver. The angle of attack and the pitch rate states both follow the outer layer values to a good extent. This indicates a trend that the fast variable system is more stable as initial Mach number decreases. The first transition region takes slightly longer to reach the outer layer at this Mach number.

The remaining figures indicate results very similar to those expected for a failed maneuver attempt. The flight path angle is only able to reach around thirty (30) degrees. The corresponding altitude change is about one thousand (1000) feet over a range of three thousand (3000) feet. The z-body direction load factor is not large and as expected the x-body direction load factor is very small. In general, the result for the initial Mach number of 0.3 are of little importance because the maneuver cannot be executed at this slow airspeed.

The next set of figures are a comparison with half-loop maneuver simulations done at NASA Langley Research Center. Figures 73 - 80 are for an initial Mach number of 0.9.

The simulation comparisons for an initial Mach number of 0.6 are given in figures 81 - 88. A comparison of the Mach 0.3 case is also included in figures 89 - 96.

In general, the aircraft response to our feedback control law is similar to the response in the NASA simulations. The major differences are that our control law uses the capabilities of the aircraft more efficiently. This efficiency produces a half-loop maneuver which requires about half the time as the NASA maneuver. The tradeoff comes in the large load factors encountered and the high rate at which these load factors increase. The energy lost in the maneuver is about the same for our control law as with the NASA maneuver. This is seen in the fact that the final Mach numbers and airspeeds are approximately equivalent. However, our maneuvers do not gain as much altitude as those of NASA indicating that the amount of energy lost is a little larger for our feedback control law. However, our half-loop maneuver is tighter. That is, it is performed with less altitude change.

It is important to note that the NASA maneuvers were not simulated in an attempt to do a half-loop in a fast time. For this reason, it is a bit unfair to make too many conclusions based on this comparison. However, these comparisons do indicate that our feedback control law does produce a fast half-loop maneuver which is similar to those done at NASA. Another important factor to keep in mind when making this comparison is that the roll at the end of the maneuver is included in the NASA simulation but not in ours. This roll can be executed in very little time but it is important to note that the final times of the NASA simulations and our simulations do not correspond to the exact same final aircraft state.

Figures 97 - 99 show the time histories for the thrust-to-weight ratios for each initial Mach number. The simulation does not include the change in weight of the aircraft due

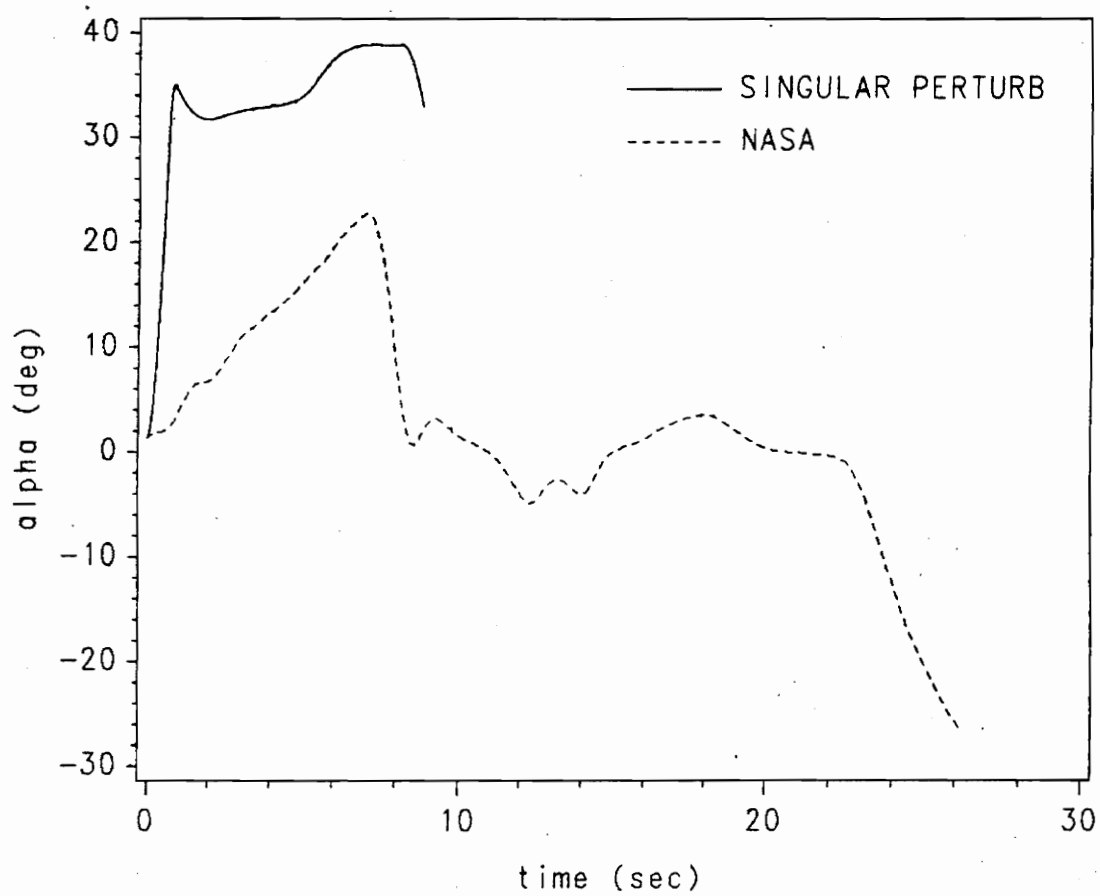


Figure 73. Comparison with NASA: Angle of Attack ($M(t=0)=0.9$, $h=15000$ ft)

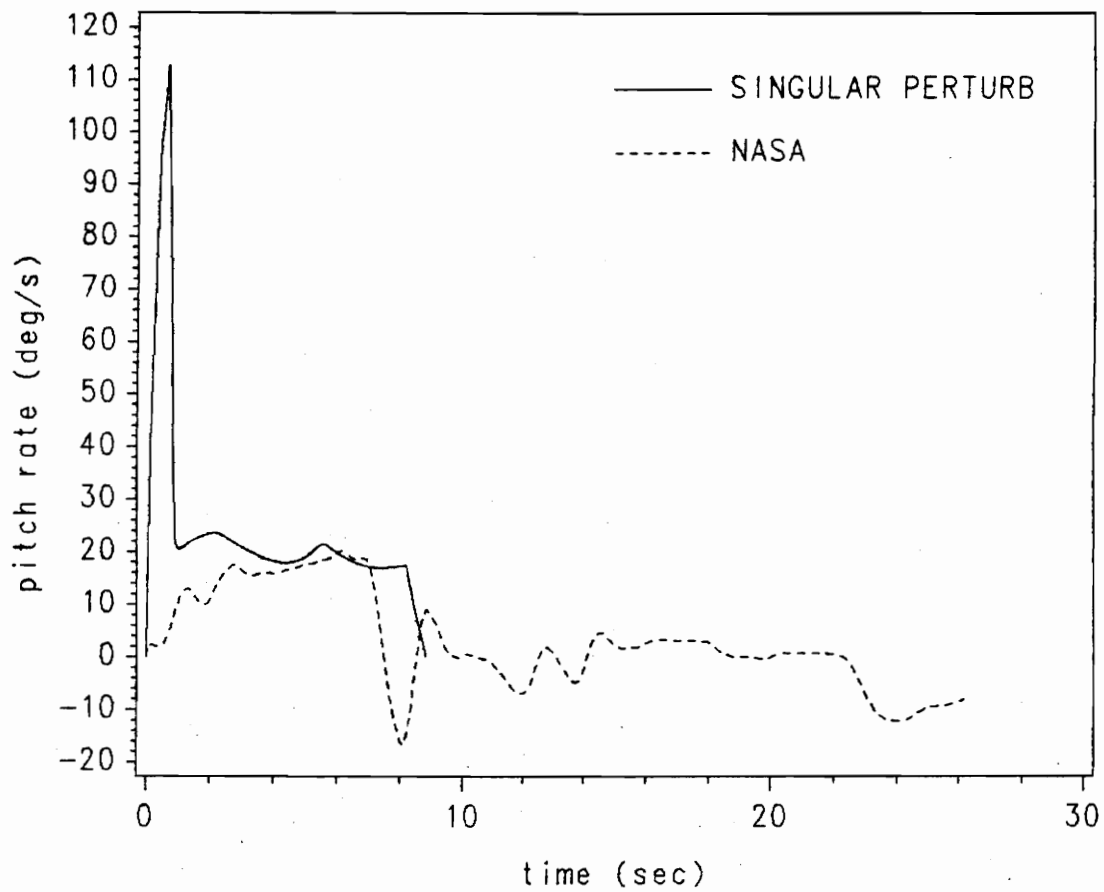


Figure 74. Comparison with NASA: Pitch Rate ($M(t=0)=0.9$, $h=15000$ ft)

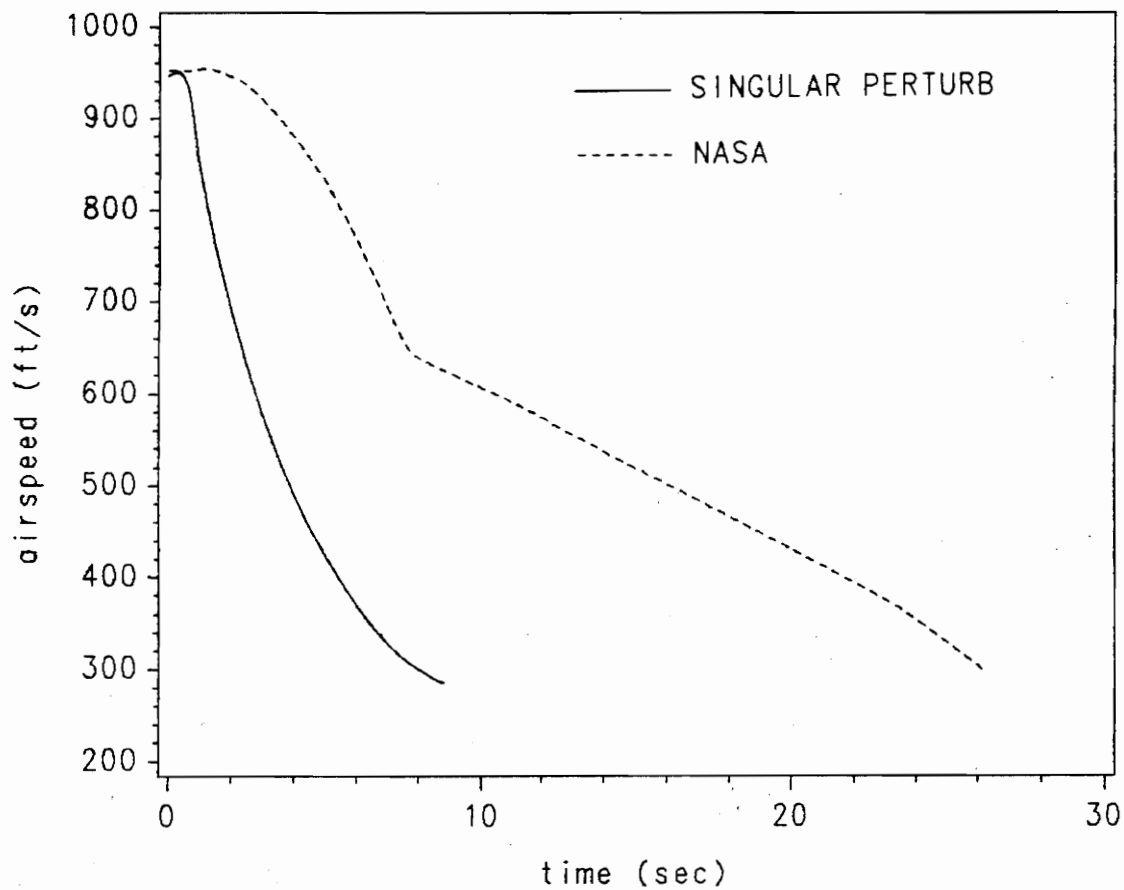


Figure 75. Comparison with NASA: Airspeed ($M(t=0)=0.9$, $h=15000$ ft)

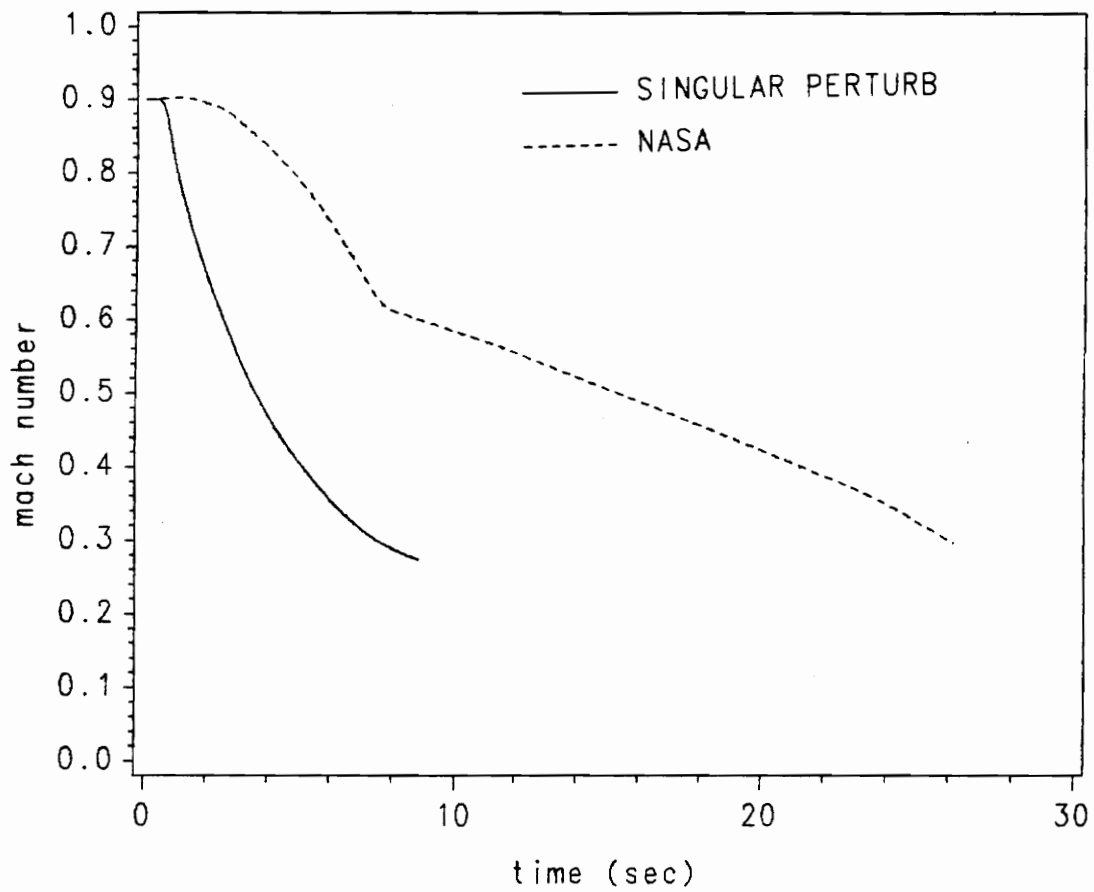


Figure 76. Comparison with NASA: Mach Number ($M(t=0)=0.9$, $h=15000$ ft)

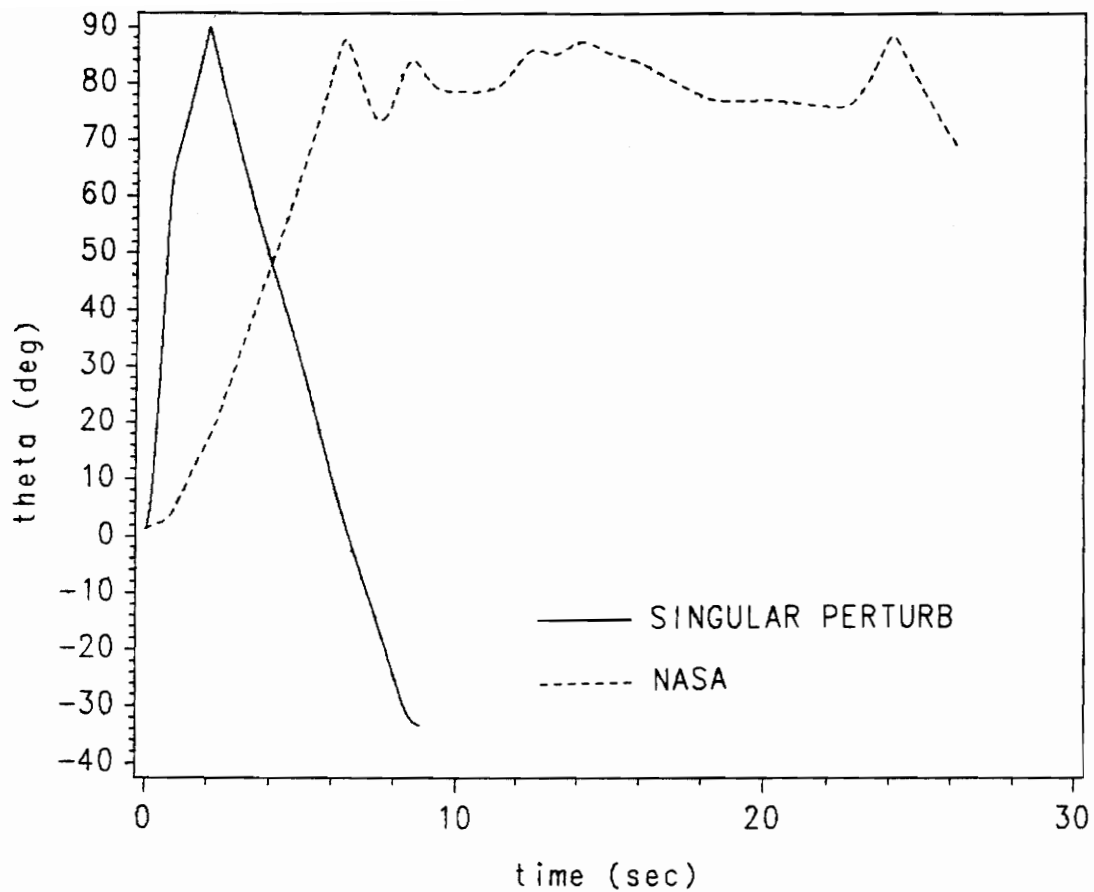


Figure 77. Comparison with NASA: Pitch Angle ($M(t=0)=0.9$, $h=15000$ ft)

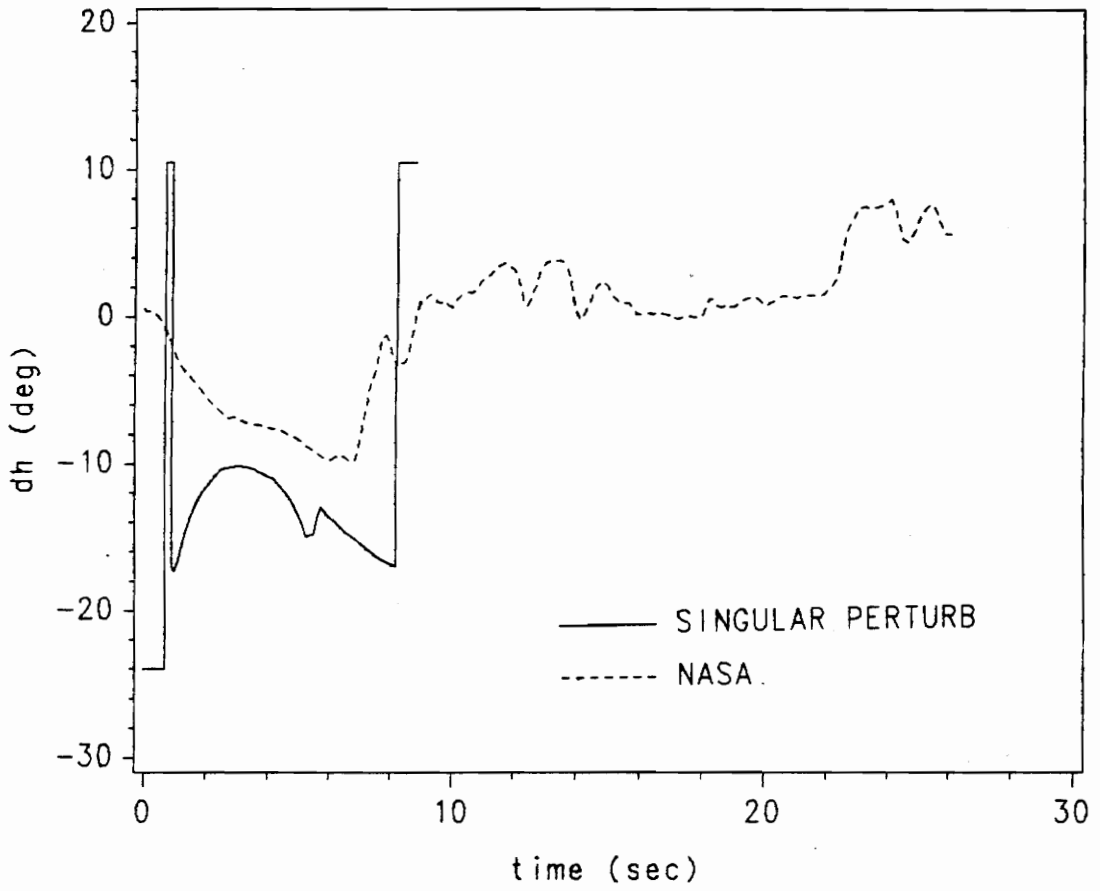


Figure 78. Comparison with NASA: Stabilator Angle ($M(t=0)=0.9$, $h=15000$ ft)

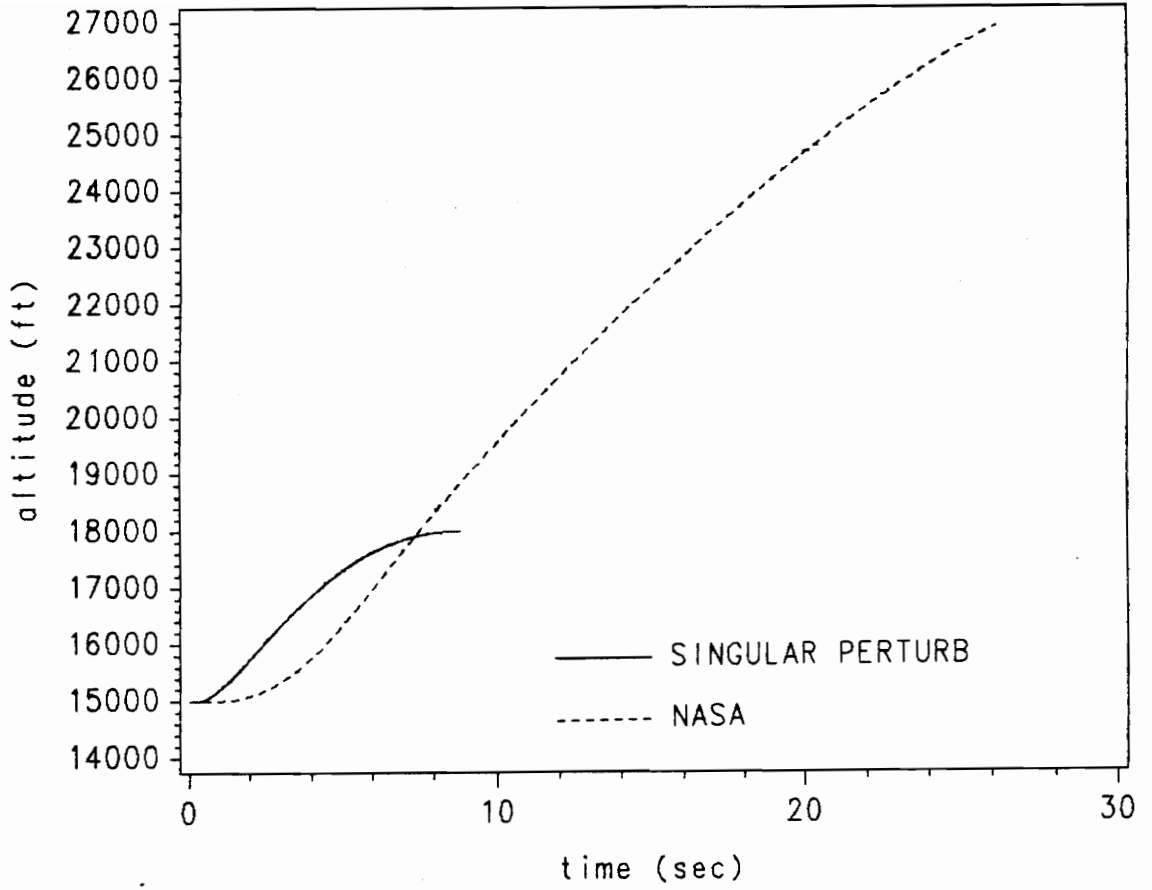


Figure 79. Comparison with NASA: Altitude ($M(t=0)=0.9$, $h=15000$ ft)

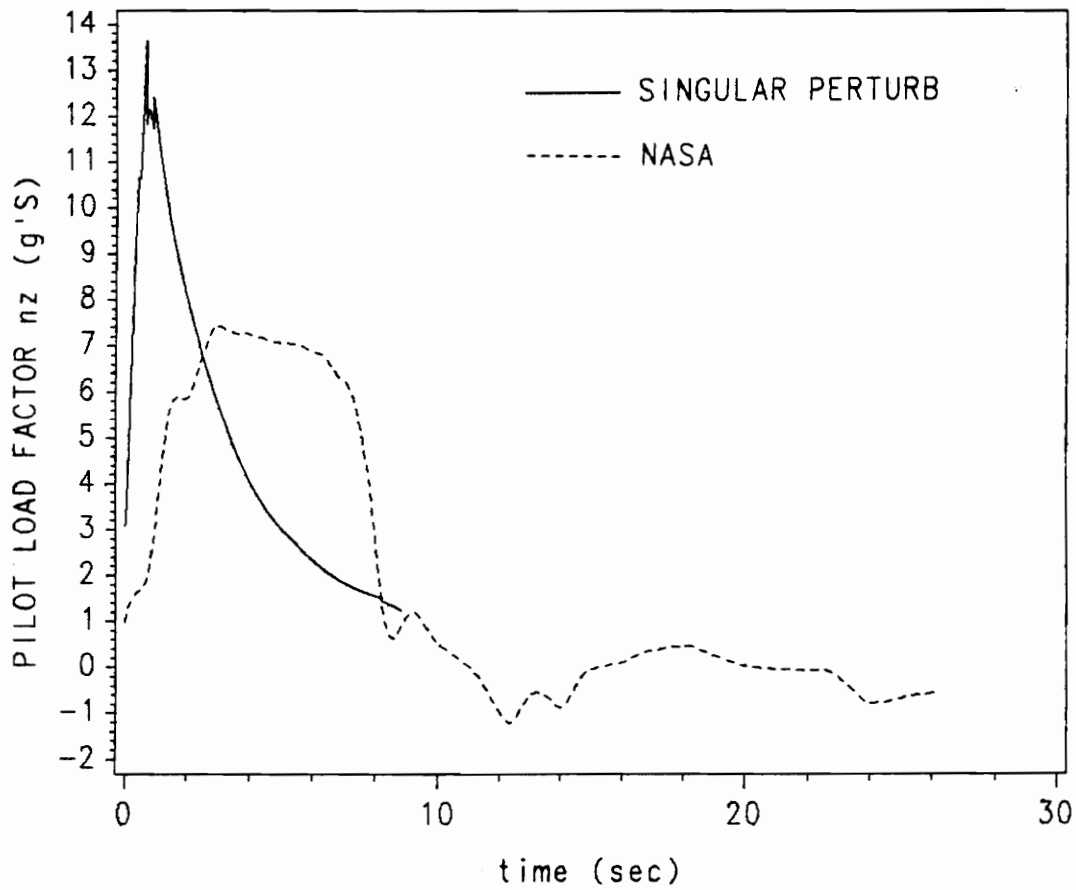


Figure 80. Comparison with NASA: Load Factor ($M(t=0)=0.9$, $h=15000$ ft)

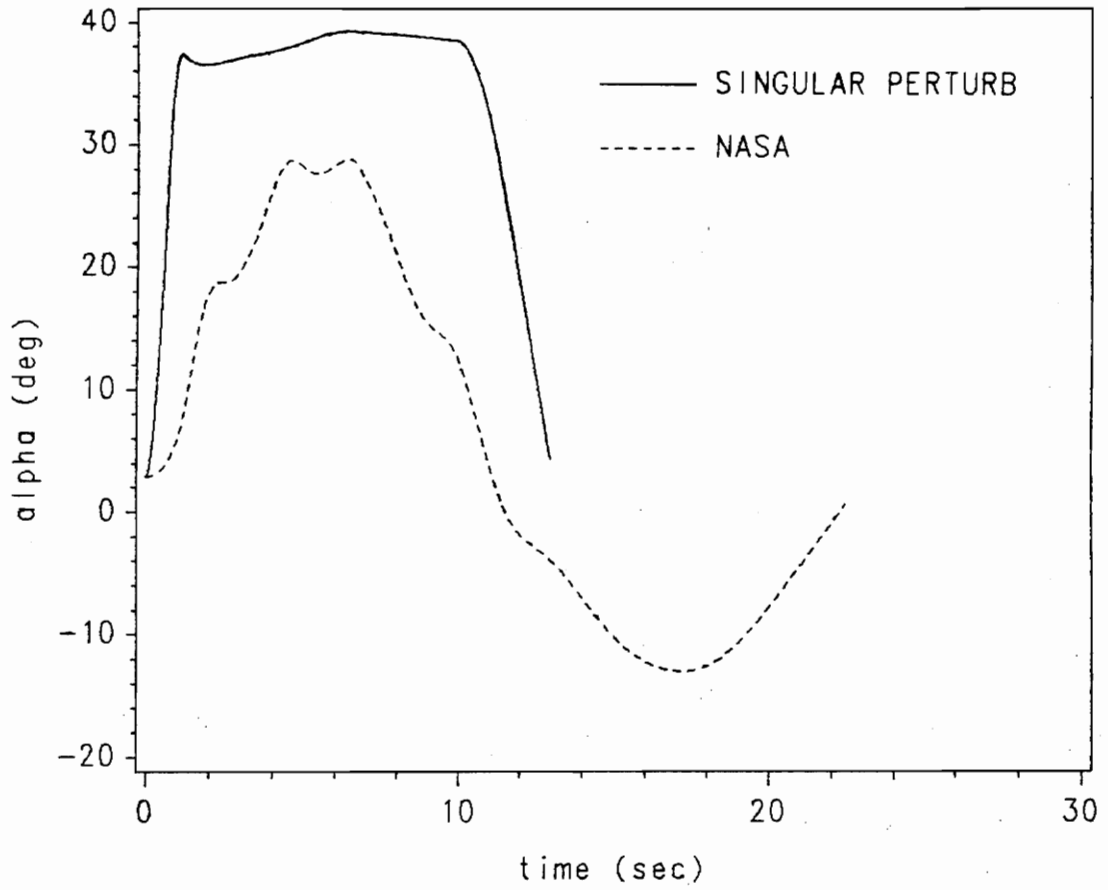


Figure 81. Comparison with NASA: Angle of Attack ($M(t=0)=0.6$, $h=15000$ ft)

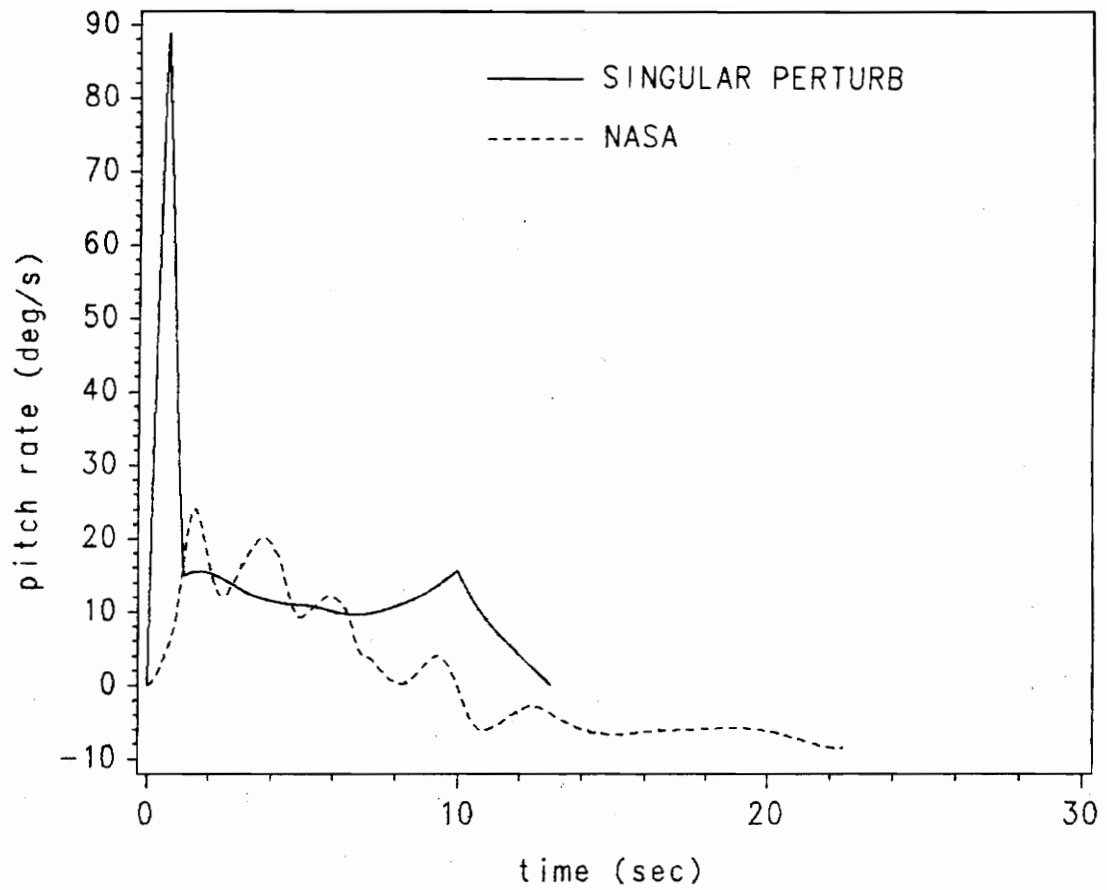


Figure 82. Comparison with NASA: Pitch Rate ($M(t=0)=0.6$, $h=15000$ ft)

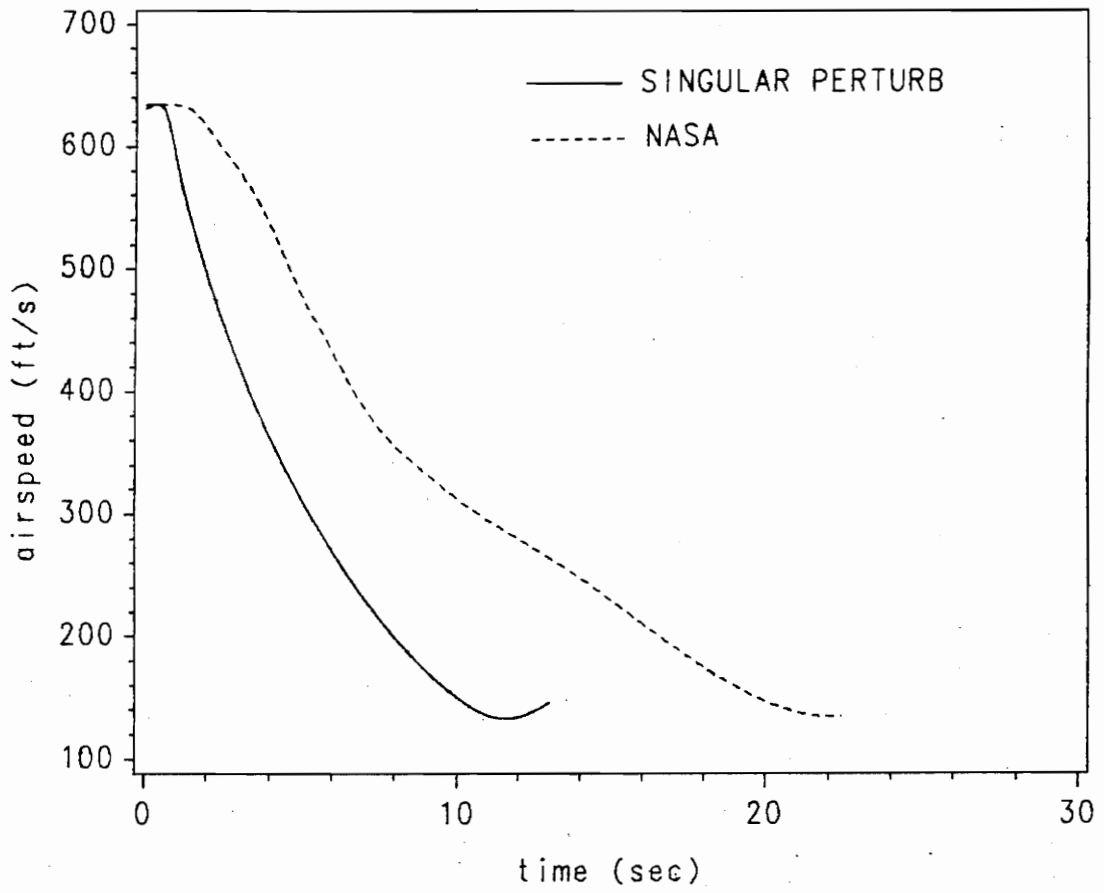


Figure 83. Comparison with NASA: Airspeed ($M(t=0)=0.6$, $h=15000$ ft)

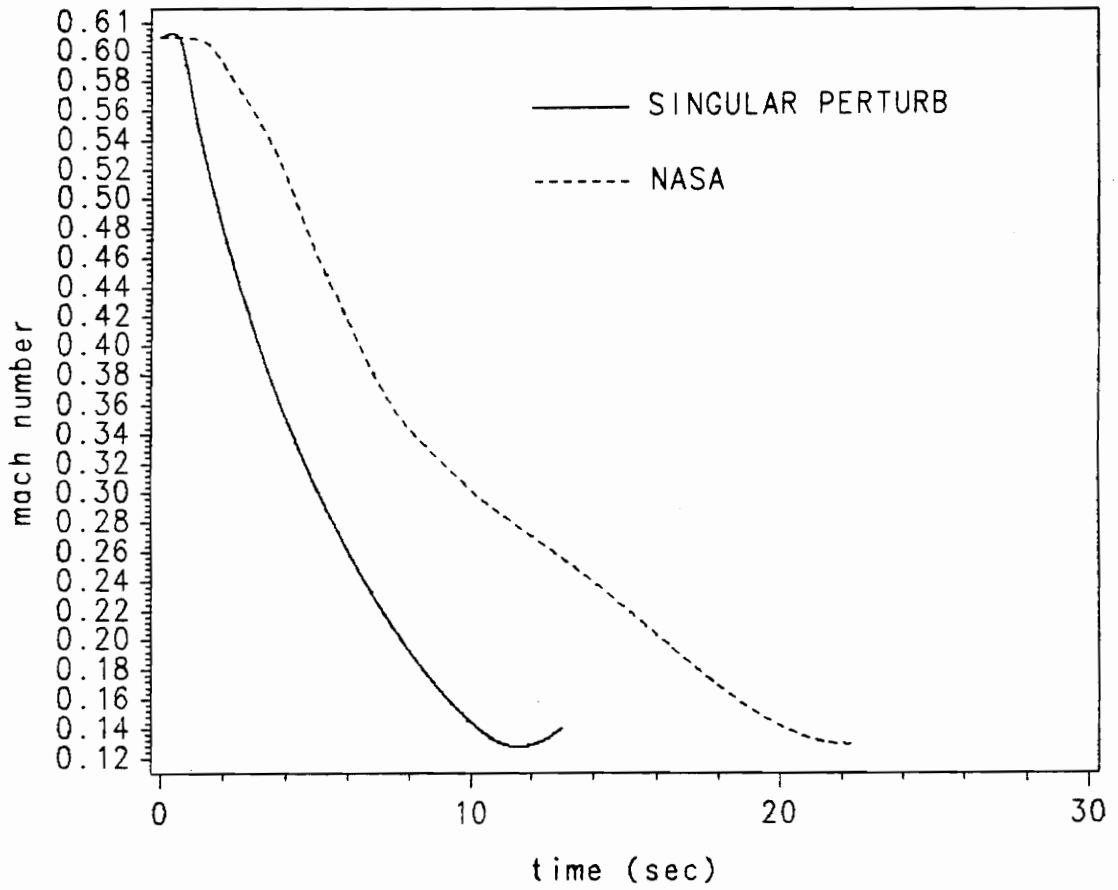


Figure 84. Comparison with NASA: Mach Number ($M(t=0) = 0.6$, $h = 15000$ ft)

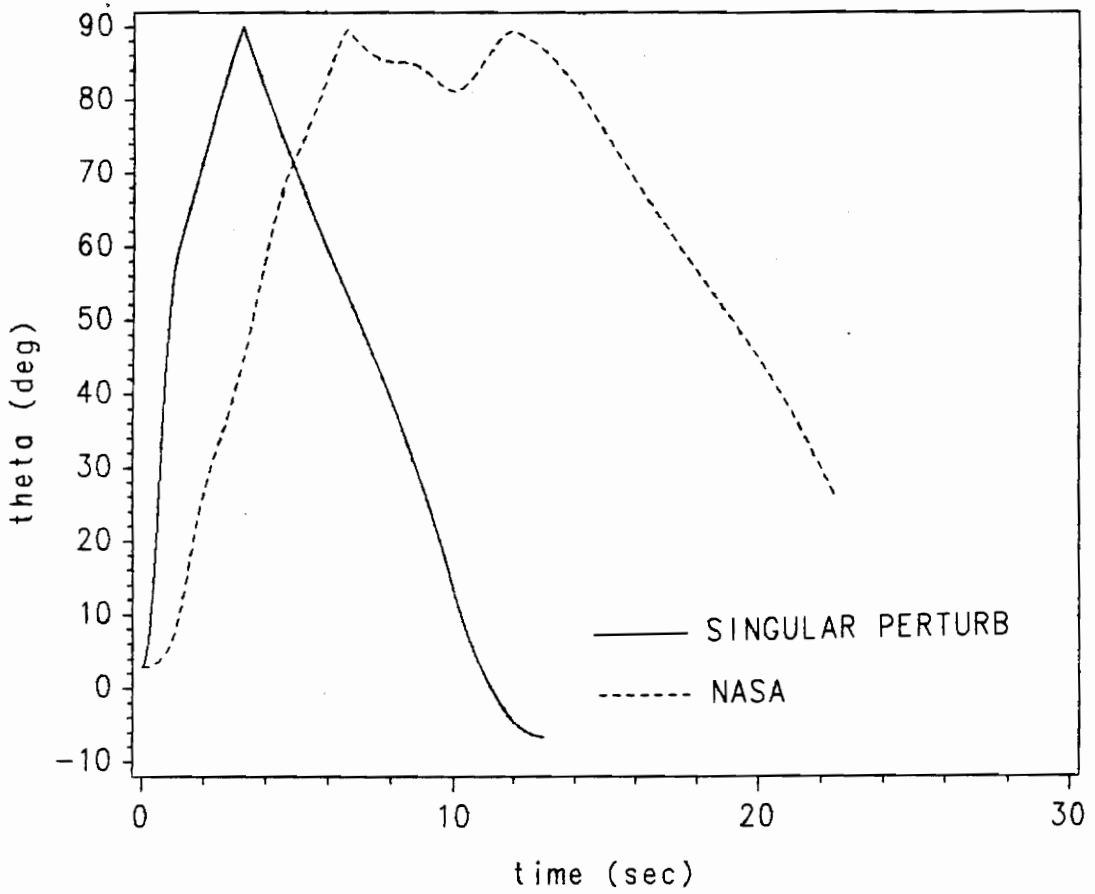


Figure 85. Comparison with NASA: Pitch Angle ($M(t=0)=0.6$, $h=15000$ ft)

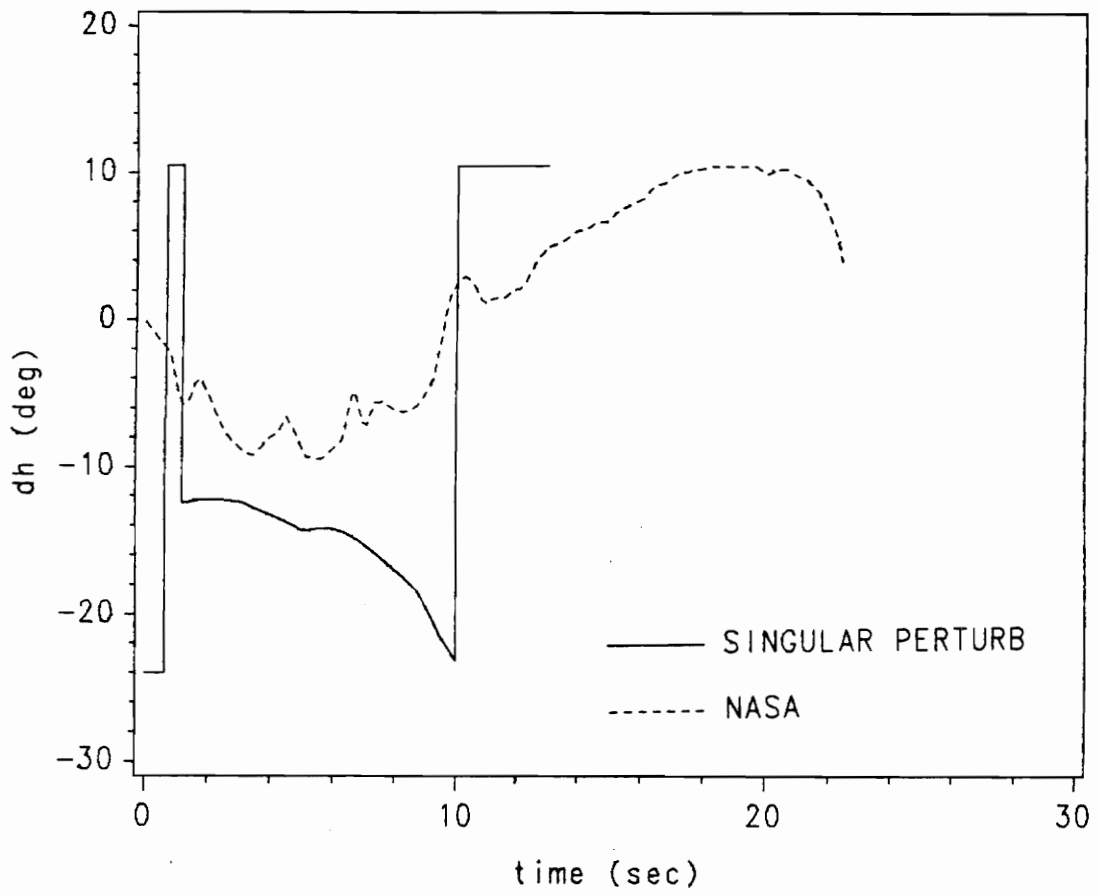


Figure 86. Comparison with NASA: Stabilator Angle ($M(t=0)=0.6$, $h=15000$ ft)

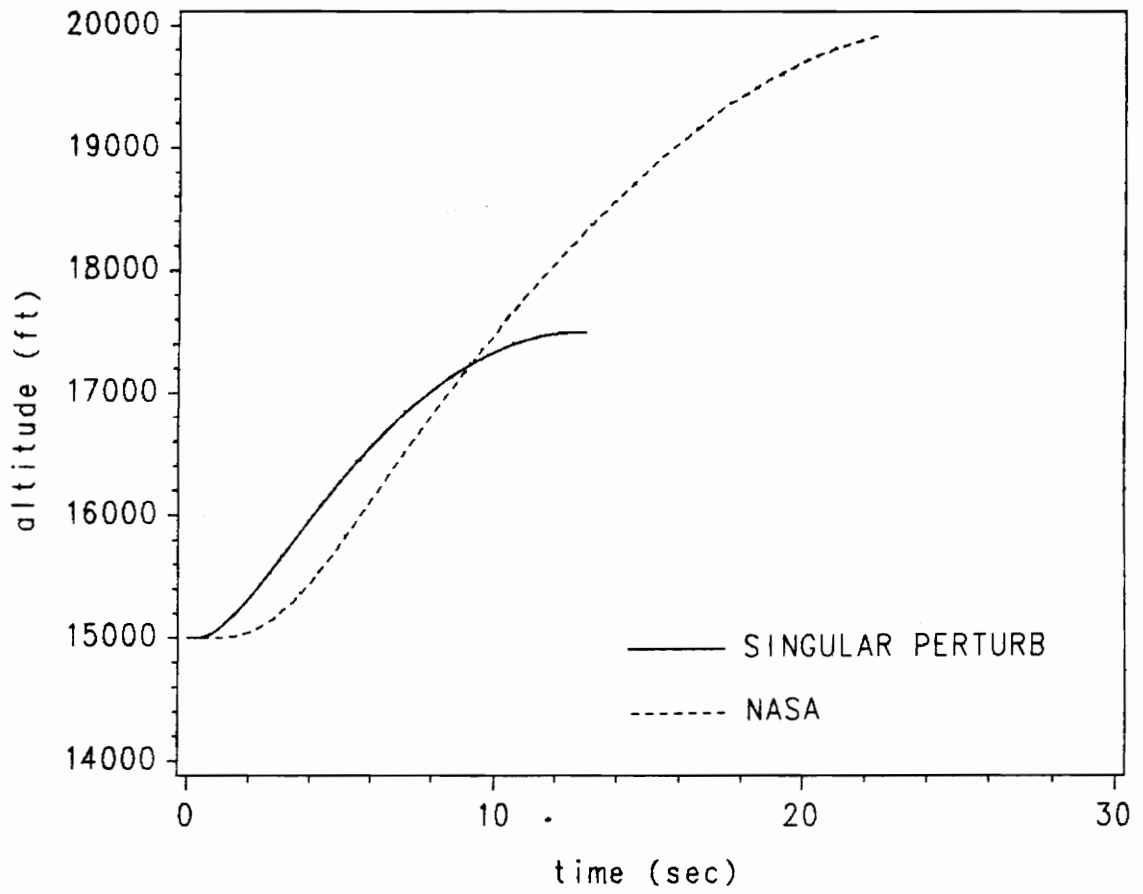


Figure 87. Comparison with NASA: Altitude ($M(t=0)=0.6$, $h=15000$ ft)

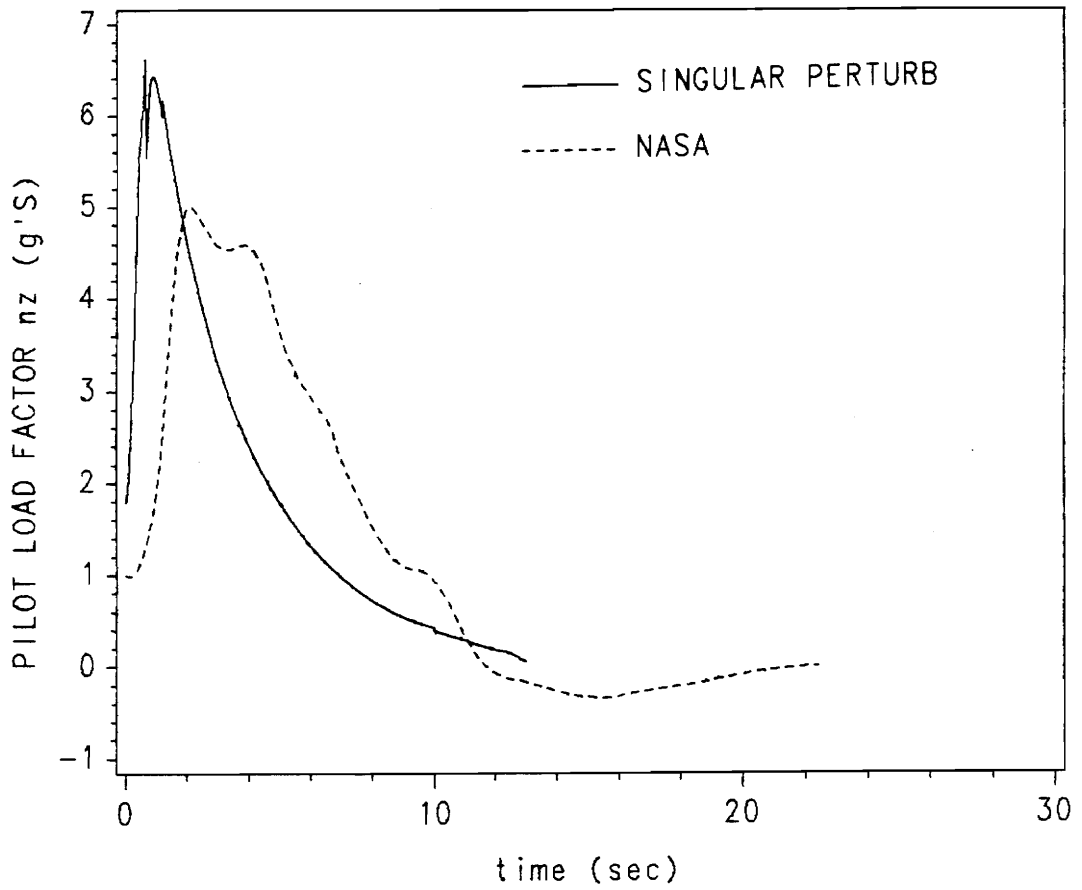


Figure 88. Comparison with NASA: Load Factor ($M(t=0)=0.6$, $h=15000$ ft)

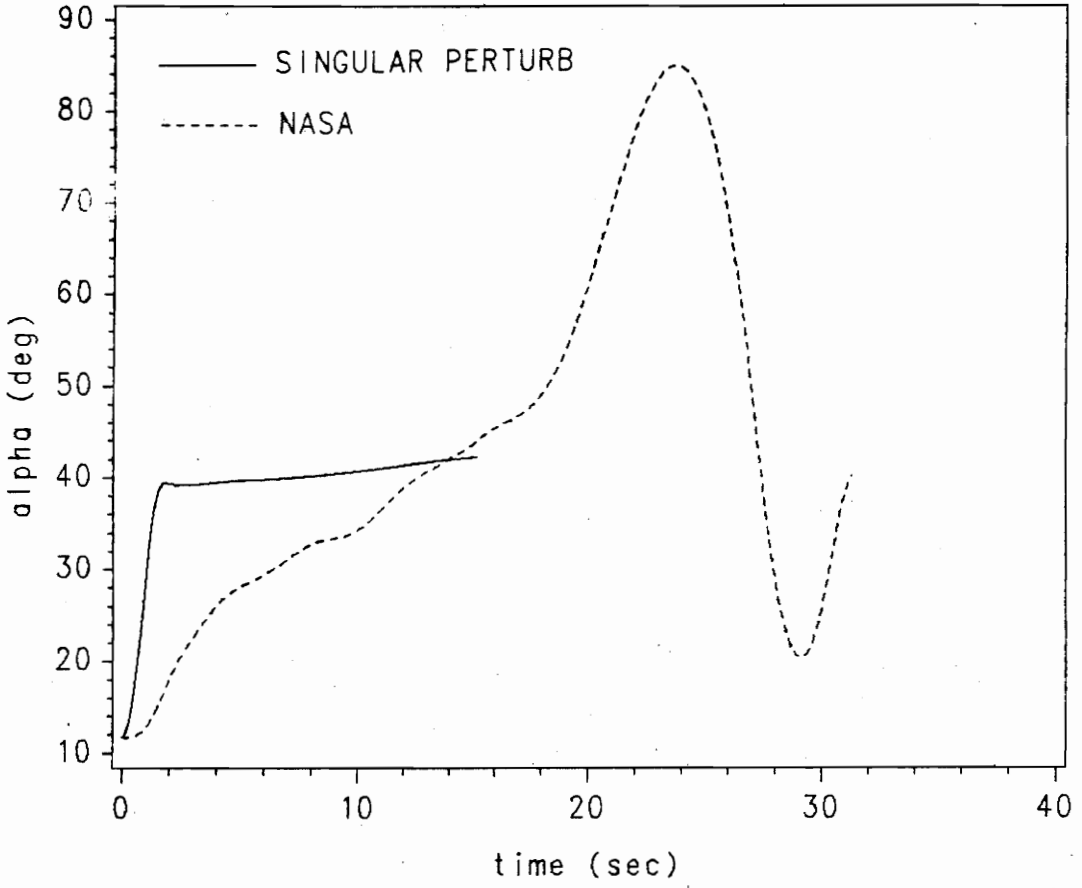


Figure 89. Comparison with NASA: Angle of Attack ($M(t=0)=0.3$, $h=15000$ ft)

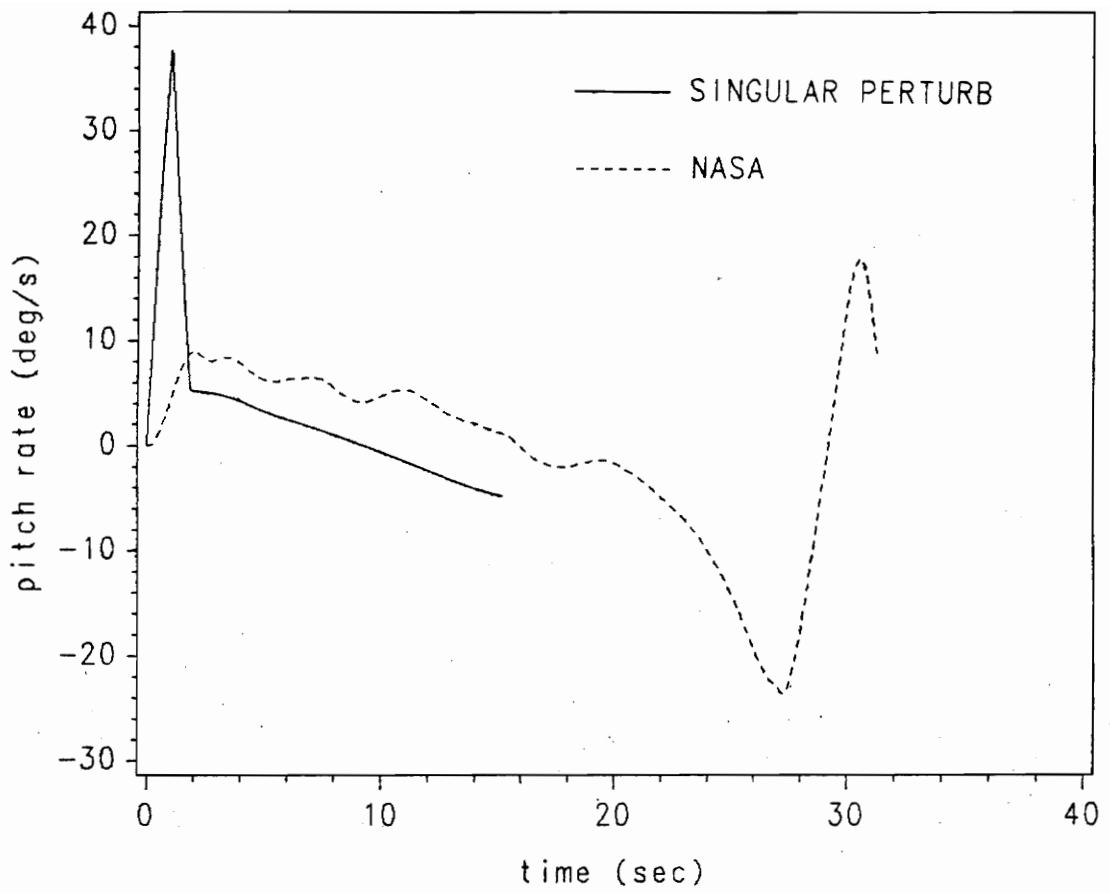


Figure 90. Comparison with NASA: Pitch Rate ($M(t=0)=0.3$, $h=15000$ ft)

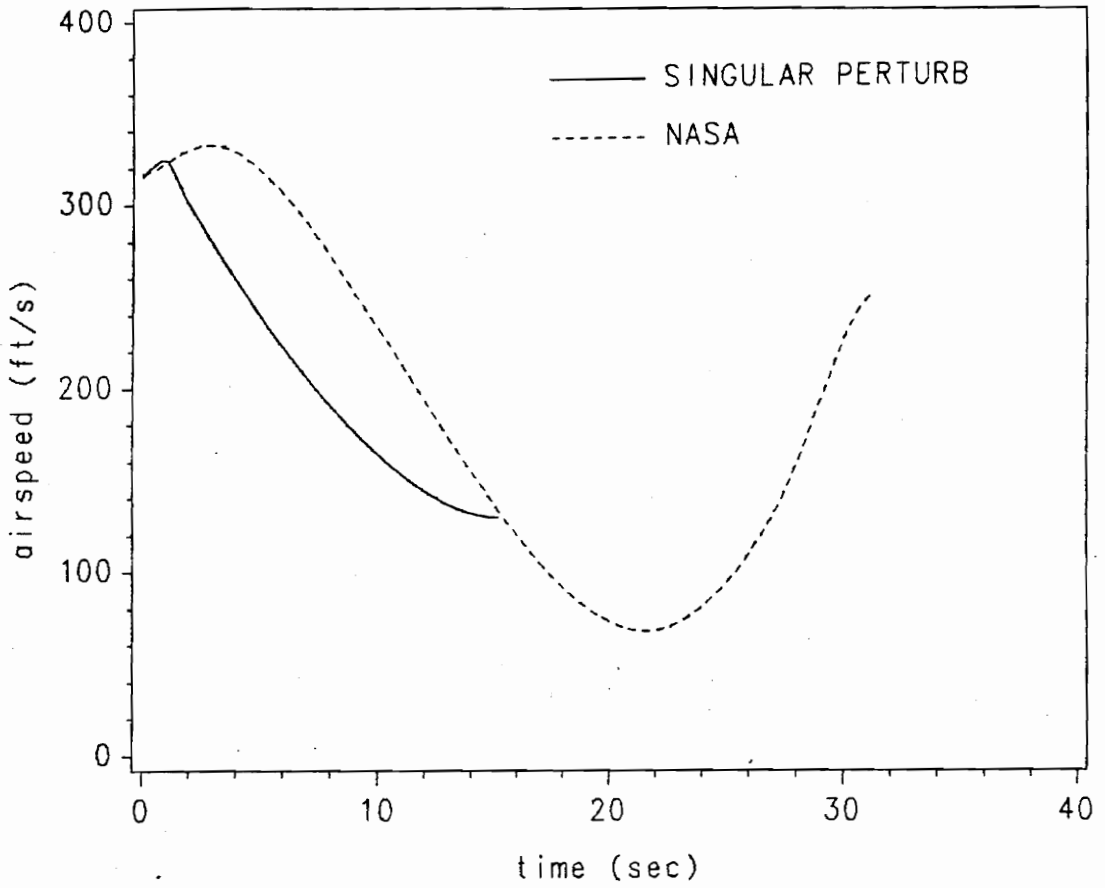


Figure 91. Comparison with NASA: Airspeed ($M(t=0)=0.3$, $h=15000$ ft)

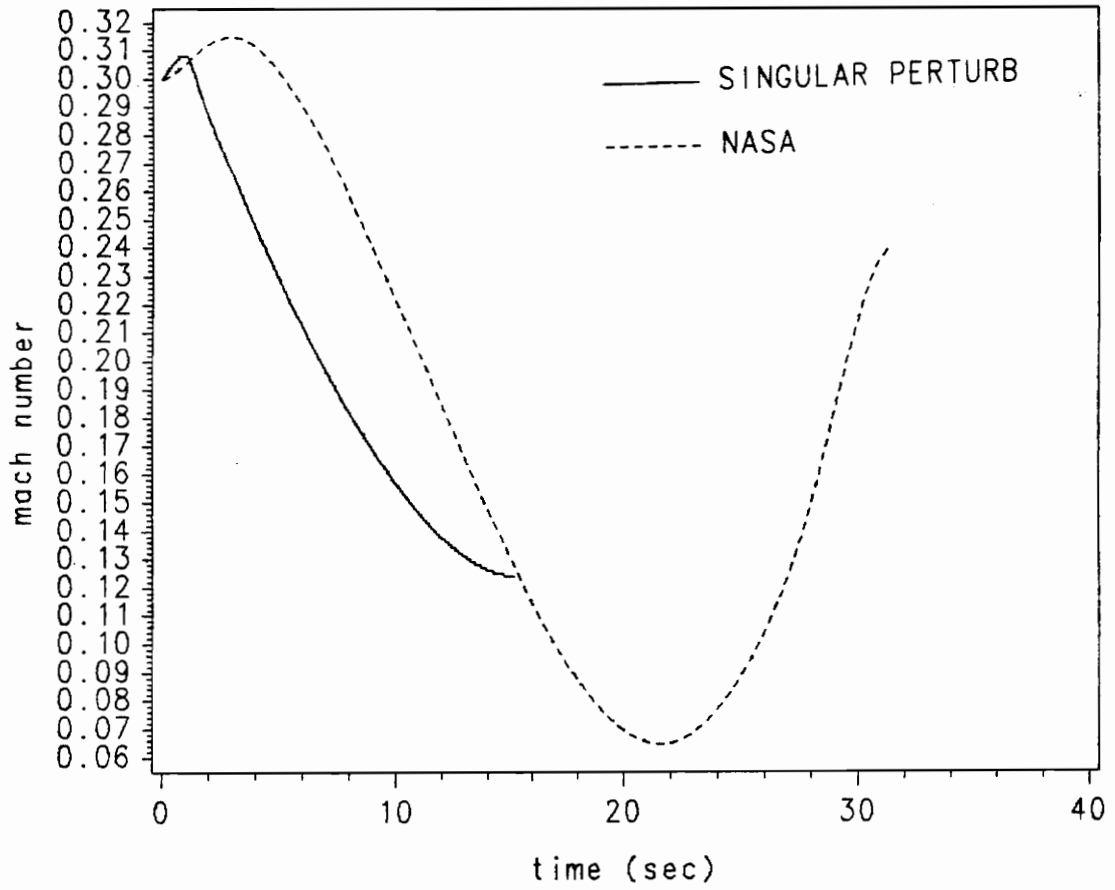


Figure 92. Comparison with NASA: Mach Number ($M(t=0)=0.3$, $h=15000$ ft)

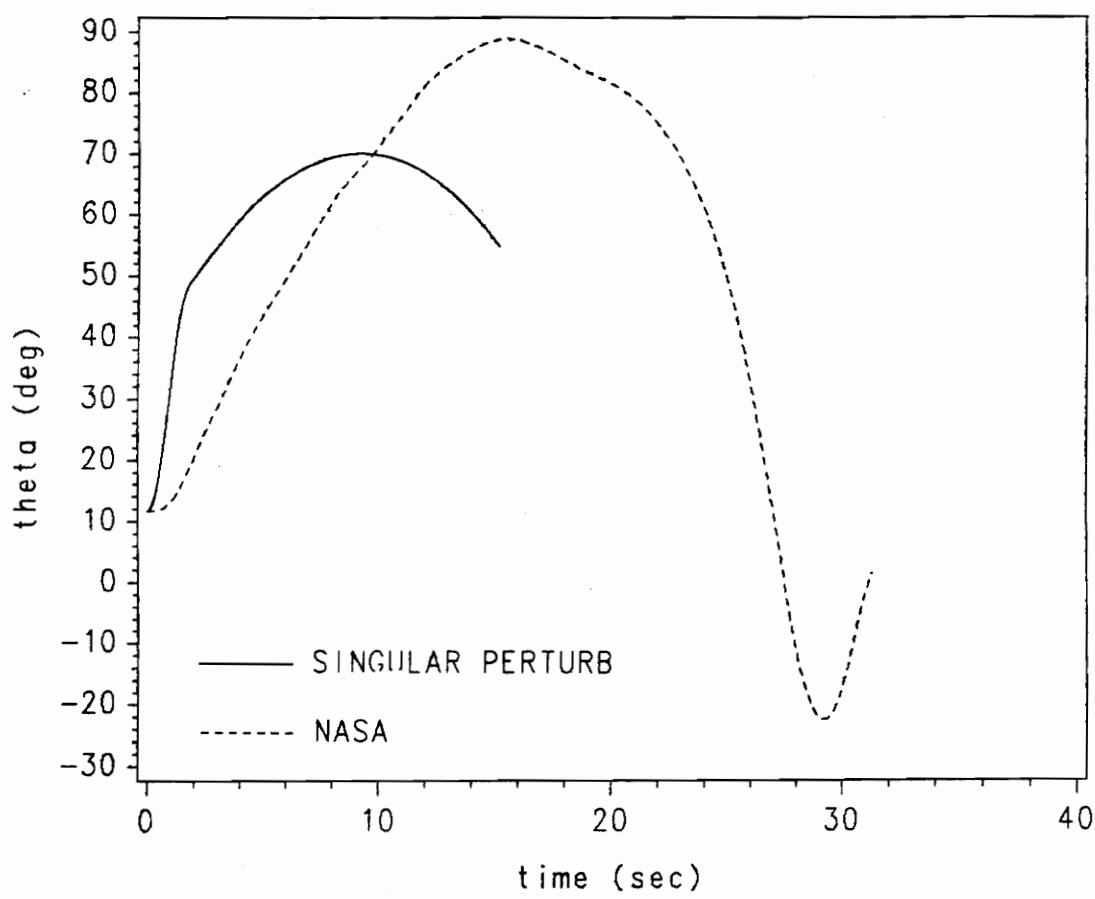


Figure 93. Comparison with NASA: Pitch Angle ($M(t=0) = 0.3$, $h = 15000$ ft)

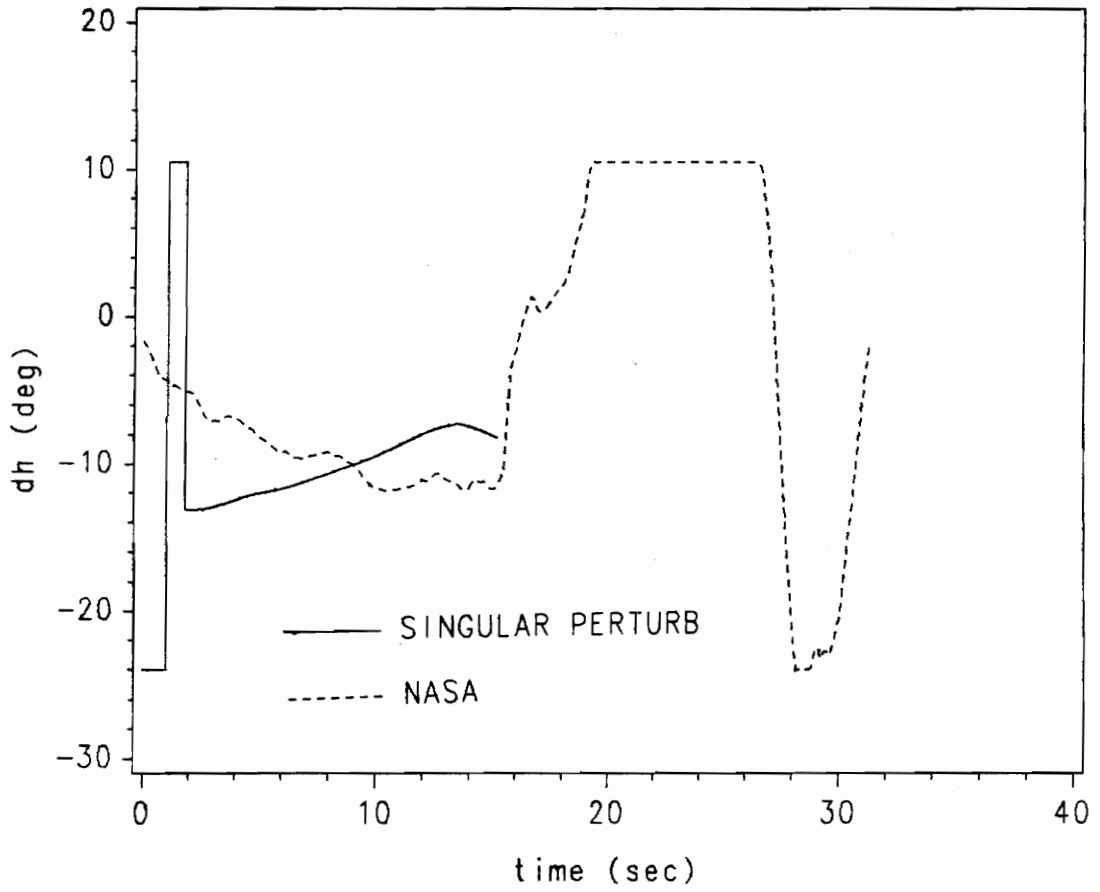


Figure 94. Comparison with NASA: Stabilator Angle ($M(t=0)=0.3$, $h=15000$ ft)

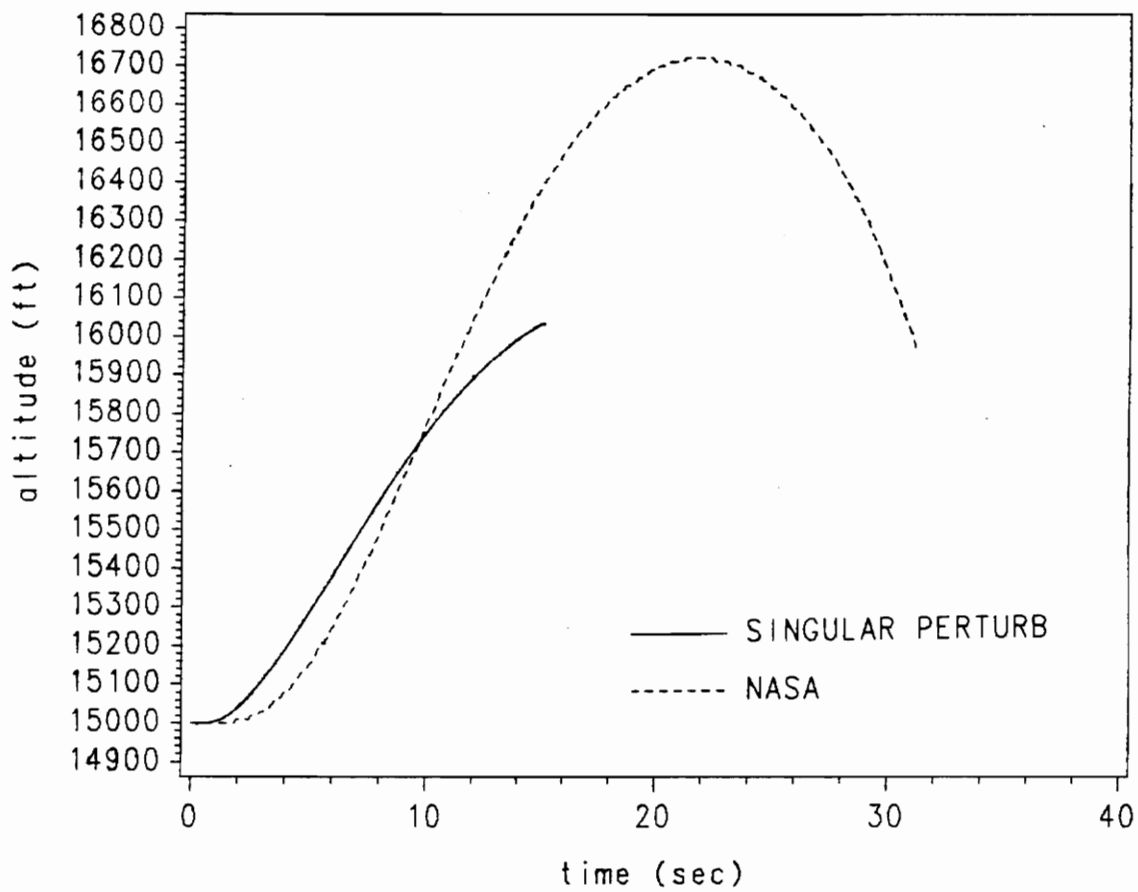


Figure 95. Comparison with NASA: Altitude ($M(t=0)=0.3$, $h=15000$ ft)

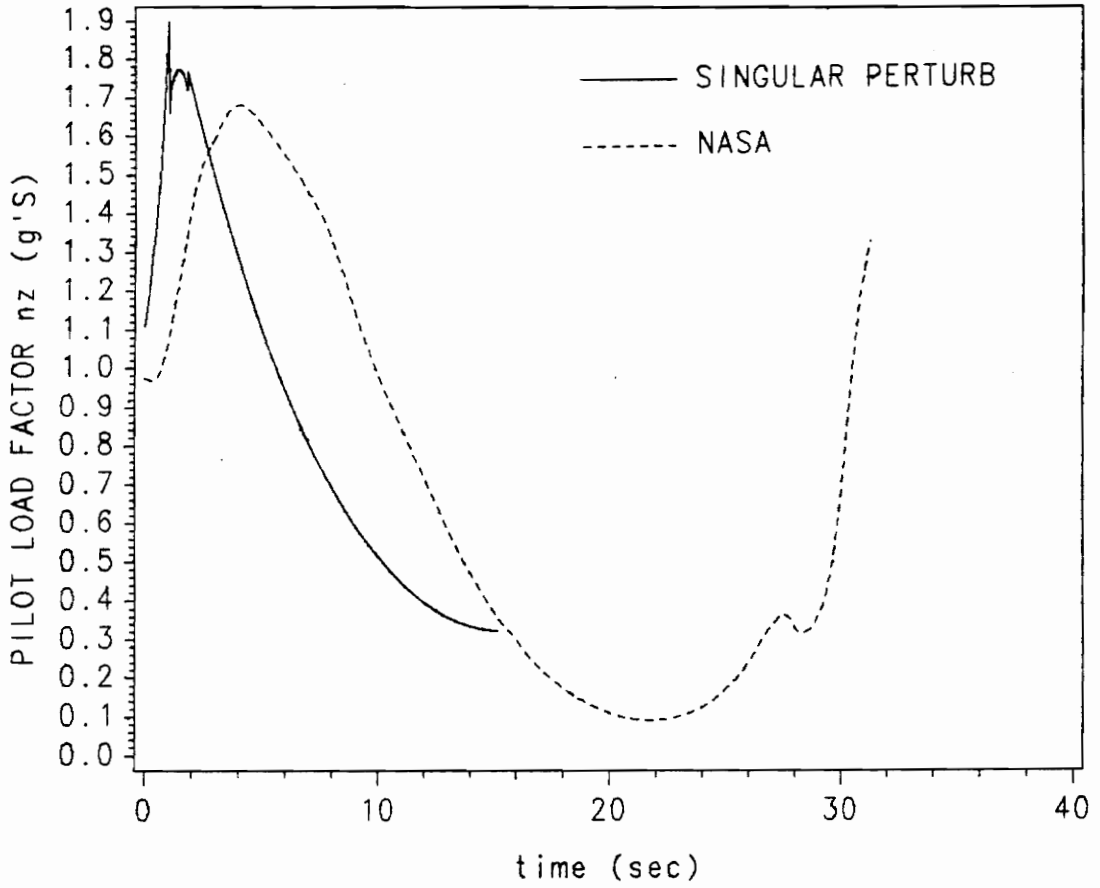


Figure 96. Comparison with NASA: Load Factor ($M(t=0)=0.3$, $h=15000$ ft)

to fuel usage. Thus these thrust-to-weight time histories only show the effect of airspeed and altitude changes in the course of the maneuver. Mach number appears to be the predominant effect on these values. These plots also indicate a reason why pilots dread the high load factors of this maneuver.

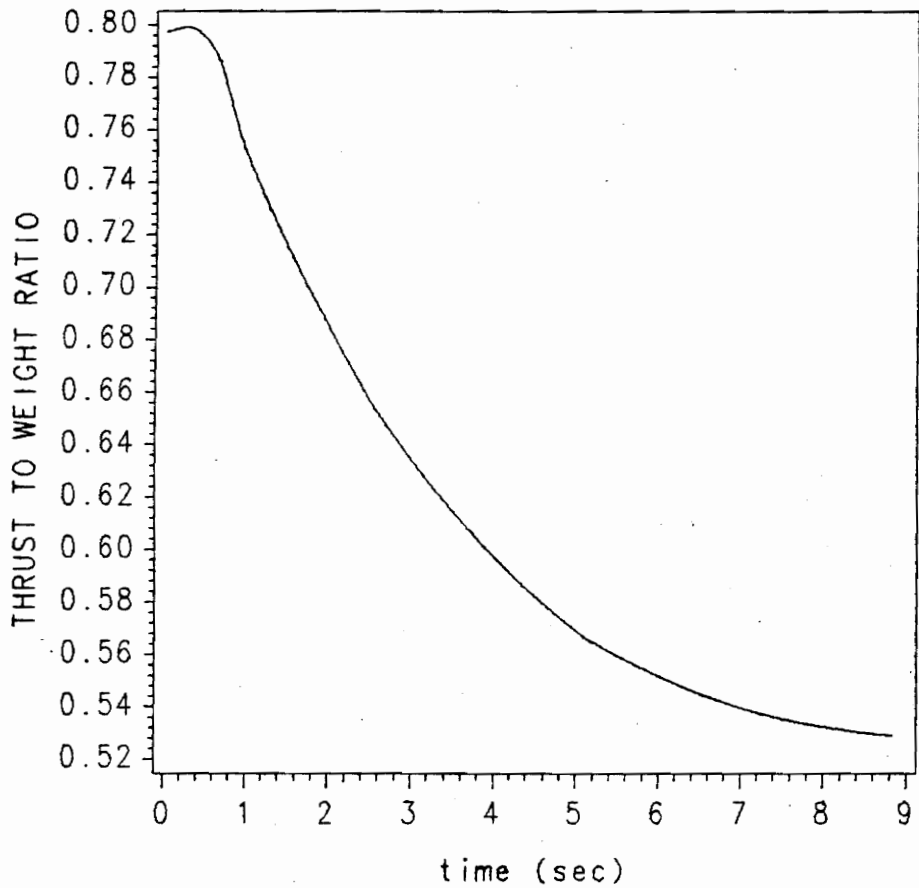


Figure 97. Thrust-to-Weight Ratio ($M(t=0)=0.9$, $h = 15000$ ft)

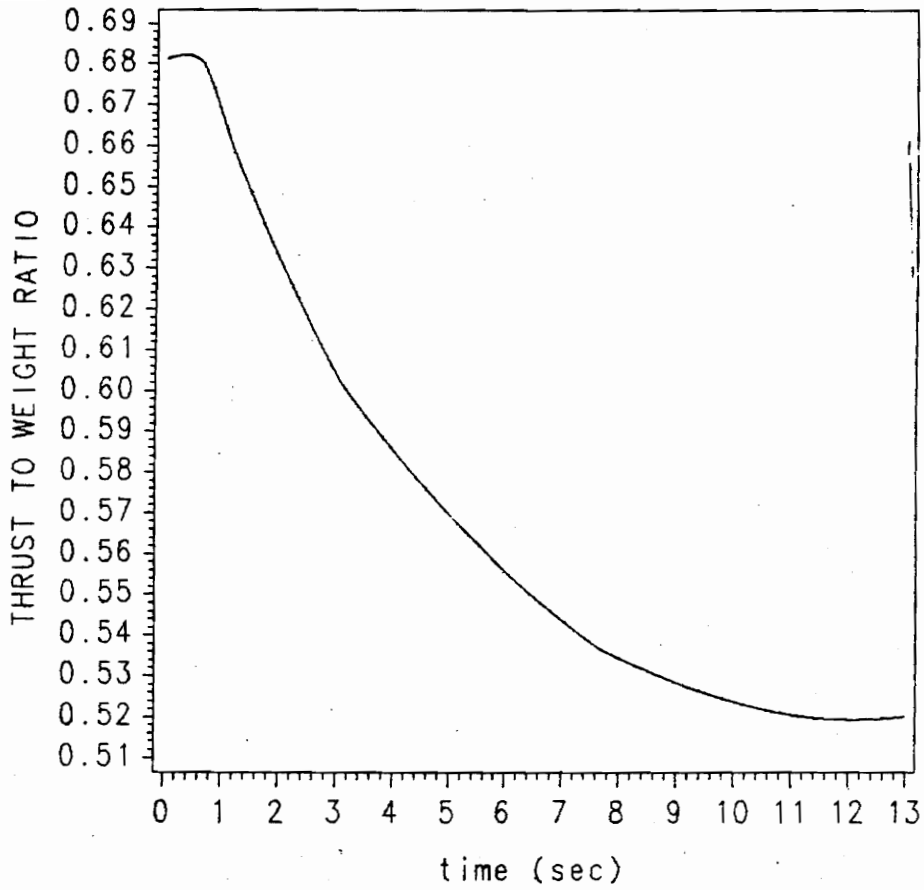


Figure 98. Thrust-to-Weight Ratio ($M(t=0)=0.6$, $h=15000$ ft)

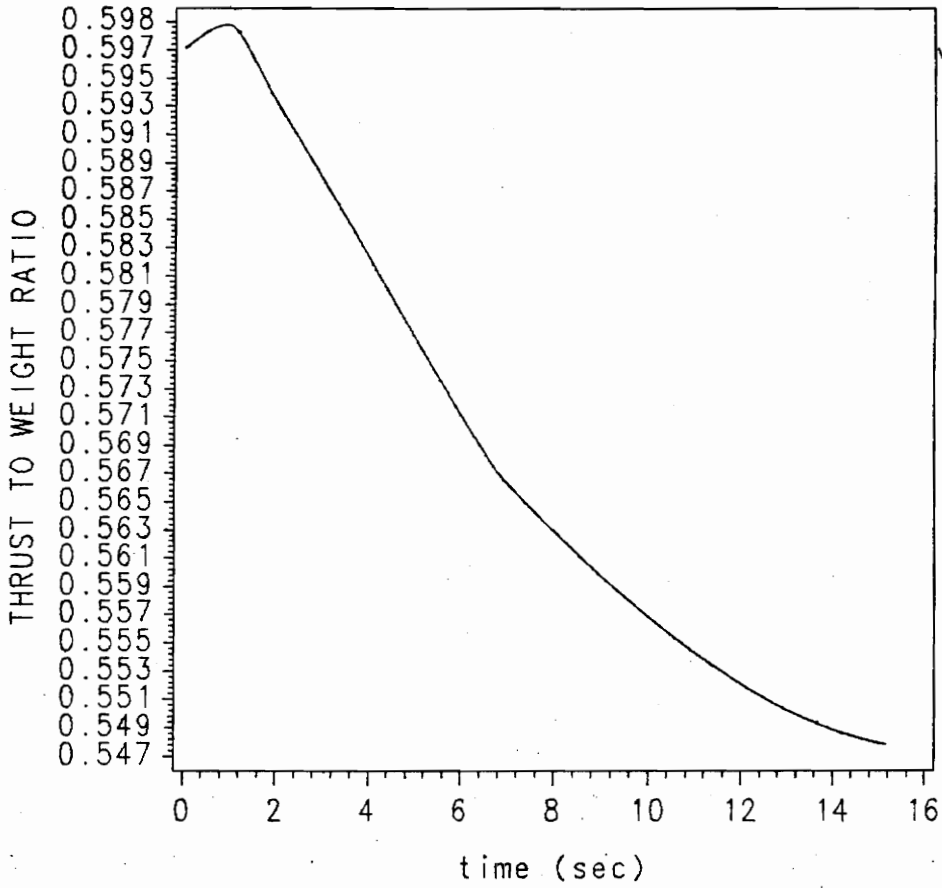


Figure 99. Thrust-to-Weight Ratio ($M(t=0)=0.3$, $h=15000$ ft)

10.0 Summary

The nonlinear feedback control law constructed in this study has been shown to produce a fast half-loop maneuver for the F/A-18 aircraft for any given initial state. The control law is relatively easy to generate using the singular perturbation approach. The feedback law approach indicates that the aircraft's capabilities can be more efficiently utilized than with a pilot supplying the input. This is visible in the comparisons with the NASA maneuvers which were generated by input from the stick by a pilot. However, the high load factors and the rate of onset of the load factor are factors to be examined in the use of the feedback control law at high Mach numbers. One way to soften the load factors at high Mach numbers is to generate a constrained feedback control law which would take the load factor limitation into account. Results from such a study would only differ in the first transition region at high initial Mach numbers. Another important point is that the aircraft is shown to be unable to execute the half-loop maneuver at an initial Mach number of 0.3. This is true for both our simulations and the NASA simulation. The low thrust-to-weight ratio of the F/A-18 (approx. 0.6) makes it impossible to perform the half-loop at low Mach numbers.

The small amount of computer code required for the feedback control law to be implemented is a benefit for this approach. The entire feedback control law is set up in about one hundred lines of FORTRAN code. The outer layer surfaces required about nine hundred (900) lines of data each in separate input files which are read at initialization. This small amount of code is able to handle all aircraft initial states possible in the region defined by altitudes up to sixty thousand (60,000) feet and Mach numbers up to 0.9. The feedback control subroutine returns the stabilator deflection, outer layer angle of attack, outer layer pitch rate and the region of the maneuver for any given state vector of the aircraft within the allowable region of Mach number and altitude. This makes this method very useful in practical implementation of the feedback control law.

The efficient use of the aircraft capabilities is a good selling point for the singular perturbations approach to this problem. The feedback control law for the outer layer is generated so that the majority of the energy used in the maneuver is used in accomplishing the rotation of the velocity vector through the required flight path angle. This is done by holding the states, which are not required to rotate the velocity vector, constant. The result is an efficient use of the aircraft state to produce the desired response. The transition regions required to get the aircraft in a position to use the outer layer control law are fast and efficient. Therefore, the singular perturbation approach is a good way of developing an efficient nonlinear feedback control law for the half-loop maneuver.

In comparison with the NASA simulations, our simulations show that our feedback control law executes the half-loop maneuver relatively fast. We notice that, in some cases, the NASA simulations appear to follow our trajectories or oscillate about them. This is an indication that our feedback control law is following a more time efficient

trajectory than that of the NASA simulation. However, it is important to remember that the NASA simulations include the roll at the end of the maneuver, which is ignored in our study, and that the terminal states of the aircraft are slightly different from those in our simulations. Also, it is important to note that the pilot in the NASA simulations was not necessarily trying to execute the maneuver in the fastest possible time. Therefore, the main reason for showing these comparisons is to show that our feedback control law produces a response which is fast in comparison with a similar maneuver. Also, these comparisons indicate that the tremendous loss in energy we experienced with our simulations are inherent in the maneuver because the NASA simulations show similar effects. Finally, the trajectories produced by our feedback control law are more efficient versions of those in the NASA simulations.

Major limitations are that the load factors produced in the first second of the maneuver for high Mach numbers are large, the onset of these large load factors is significant and the maneuver cannot be completed at low Mach numbers. A solution to the load factor problem is to apply a load factor constraint on the analysis. The first transition region would then take a longer time to complete. After the first second of the maneuver (i.e. when the load factor has fallen below 7.5 g's), the solution would be identical to that of this study. A similar approach could be used to limit the onset of the load factor in the first transition region. The inability to complete the maneuver at low Mach numbers is a problem associated with the engines of the F/A-18. If the thrust-to-weight ratio were improved, our feedback control law would be able to execute the maneuver just as well as with higher initial Mach numbers. Therefore, there is no modification to our feedback law which would improve this problem.

In conclusion, the singular perturbation approach used in this study does give us a relatively fast nonlinear feedback control law for the half-loop maneuver. A similar approach could be used to develop a nonlinear feedback control law for the very similar split-s maneuver. In each case, our feedback control law could be implemented in a practical setting onboard the F/A-18 aircraft. Also, the next step would be to use these sub-optimal control laws as a comparison or baseline for the development of both open-loop and closed-loop time optimal control laws for these maneuvers. A study of each of these aspects of the problem is planned for the future.

11.0 References

1. Arbuckle, P.D., Buttrill, C.S. and Zeiler, T.A., "A New Simulation Model Building Process for use in Dynamic Systems Integration Research ", Presented at the AIAA Flight Simulation Technologies Conference, Monterey, California, August 17-19, 1987, AIAA Paper 87-2498-CP.
2. Arbuckle, P.D. and Buttrill, C.S., "NASA F/A-18 Fighter Escort Model " computer code
3. Ardema, M.D., "Singular Perturbations in Flight Mechanics ", NASA Technical Memorandum X-62,380, Ames Research Center, Moffett Field, California, August 1974, Revised July 1977.
4. Bryson, A.E. and Ho, Y.C., *Applied Optimal Control*, Hemisphere, 1975.
5. Calise, A.J., "Singular Perturbation Methods for Variational Problems in Aircraft Flight", IEEE Transactions on Automatic Control, Volume AC-21, No. 3, 1976, pp 345-353.
6. *International Mathematical and Statistical Language*, Version 10.0, IMSL, INC., Houston, 1987.
7. *Jane's All the World's Aircraft 1986-87*, Jane's Publishing Company Limited, London, 1987, pp 251-252.
8. Jordan, D.W. and Smith, P., *Nonlinear Ordinary Differential Equations*, Clarendon Press, Oxford, 1987, pp 146-168.
9. Kelley, H.J., "Aircraft Maneuver Optimization by Reduced Order Approximation", *Control and Dynamic Systems*, Volume 10, C.T. Leondes, ed., Academic Press, 1973.

10. Kraft, D., "A Software Package for Sequential Quadratic Programming", to be published in August 1988.
11. Shankar, U.J., "Singular-Perturbation Analysis of Climb-Cruise-Dash Optimization", M.S. Thesis, Virginia Polytechnic Institute and State University, May 1985.
12. Stalford, H.L., and Hoffman E., "On Minimum Time to Point, Maneuver, and Shoot: Singular Perturbation Feedback Law in Head-on-pass Engagement", Presented to the 16th Congress of the International Council of the Aeronautical Sciences (ICAS), August 28-September 2, 1988, Jerusalem, Israel.
13. Stalford, H.L. and Garrett , F.E., Jr., "Robust Nonlinear Control for High Angle of Attack", Presented at the AIAA 25th Aerospace Sciences Meeting, Reno, Nevada, January 12-15, 1987, AIAA Paper 87-0346.
14. Stalford, H.L., "High-Alpha Aerodynamic Model Identification of T-2C Aircraft Using the EBM Method", Journal of Aircraft, Volume 18, No. 10, October 1981, AIAA 80-0172R, pp 801-809.
15. Stalford, H.L., and Hoffman E., "On Minimum Time to Point, Maneuver, and Shoot: Maximum Principle Solution in Head-on-pass Engagement", Presented to the 16th Congress of the International Council of the Aeronautical Sciences (ICAS), August 28-September 2, 1988, Jerusalem, Israel.
16. Visser, H.G., "An Introduction to the Technique of Singular Perturbations Applied to Performance Optimization Problems in Atmospheric Flight Mechanics" , Report LR-374, Delft University of Technology, Department of Aerospace Engineering, Delft-The Netherlands, December, 1982.

Appendix A. F/A - 18 Fighter Escort Model

The NASA F/A - 18 Fighter Escort Model consists of an aerodynamic model and a thrust model. Both models use linear interpolation based on nodes contained in a set of look-up tables. This is accomplished using two subroutines for each model. One subroutine sets up the tables in memory while the other does the interpolation.

A.1 Aerodynamic Model

Nondimensional force and moment coefficients are calculated in the aerodynamic model. The model was developed from two wind tunnel tests. Results from these were then combined to create the model. This approach resulted in a discontinuity in the model at the angle of attack where the two sets of test data were joined ($\alpha = 40^\circ$). A flap schedule is used to compensate for this discontinuity. However, for Mach numbers in the region (0.6 to 0.9) and low altitudes a discontinuity still exists. The remedy for this problem is discussed in Appendix B.

The present study deals with flight in the plane of symmetry. Therefore, the coefficients used in longitudinal flight equations of motions are discussed here.

A.1.1 Lift Coefficients

$$\begin{aligned}
 C_L &= f(\alpha, \beta, q, M, h, \dot{\alpha}, \delta_h, \delta_a, \delta_r, \delta_{lef}, \delta_{tef}, \delta_{SB}, \Delta lg) \\
 &= C_{L0}(\alpha, M) + (\Delta C_{L0})_f(\alpha, M, h) + \Delta C_{L_{\delta_h}}(\alpha, \delta_h, M) \\
 &\quad + [C_{L_{\delta_{lef}}}(\alpha, M) + (\Delta C_{L_{\delta_{lef}}})_f(\alpha, M, h)]\delta_{lef} \\
 &\quad + [C_{L_{\delta_{tef}}}(\alpha, M) + (\Delta C_{L_{\delta_{tef}}})_f(\alpha, M, h)]\delta_{tef} \\
 &\quad + \Delta C_{L_{\delta_a}}(\alpha, \delta_a, M) + \Delta C_{L_{\delta_r}}(\alpha, \delta_r, M) + \Delta C_{L_{\delta_r}}(\alpha, \delta_r, M) \\
 &\quad + \Delta C_{L_{ES}} + C_{L_{\delta_{SB}}}(\alpha, M)\delta_{SB} + \Delta C_{L_{ig}}(\alpha)\Delta lg \\
 &\quad + \frac{\bar{c}}{2V} [C_{L_q}(\alpha, M) + (\Delta C_{L_q})_f(M, h)]q \\
 &\quad + \frac{\bar{c}}{2V} [C_{L_{\dot{\alpha}}}(\alpha, M) + (\Delta C_{L_{\dot{\alpha}}})_f(M, h)]\dot{\alpha}
 \end{aligned} \tag{A.1}$$

For flight in the plane of symmetry:

$$\beta = \delta_a = \delta_r = 0 \tag{A.2}$$

And assuming landing gear up ($\Delta lg = 0$), no speed brake deflection ($\delta_{SB} = 0$) and no extra external stores ($\Delta C_{LES} = 0$):

$$\begin{aligned}
 C_L &= C_{L0}(\alpha, M) + (\Delta C_{L0})_f(\alpha, M, h) + \Delta C_{L_{\delta_h}}(\alpha, \delta_h, M) \\
 &\quad + [C_{L_{\delta_{lef}}}(\alpha, M) + (\Delta C_{L_{\delta_{lef}}})_f(\alpha, M, h)]\delta_{lef} \\
 &\quad + [C_{L_{\delta_{tef}}}(\alpha, M) + (\Delta C_{L_{\delta_{tef}}})_f(\alpha, M, h)]\delta_{tef} \\
 &\quad + \frac{\bar{c}}{2V} [C_{L_q}(\alpha, M) + (\Delta C_{L_q})_f(M, h)]q \\
 &\quad + \frac{\bar{c}}{2V} [C_{L_{\dot{\alpha}}}(\alpha, M) + (\Delta C_{L_{\dot{\alpha}}})_f(M, h)]\dot{\alpha}
 \end{aligned} \tag{A.3}$$

A.1.2 Drag Coefficient

$$\begin{aligned}
 C_D &= f(C_L, \alpha, M, \delta_h, \delta_r, \delta_{lef}, \delta_{tef}, \delta_{SB}, \Delta l_g) \\
 &= C_{D_0}(\alpha, M) + (\Delta C_D)_{C_L}(C_L, M) + \Delta C_{D_{\delta_h}}(\alpha, \delta_h, M) \\
 &\quad + C_{D_{\delta_{lef}}}(\alpha, M)\delta_{lef} + C_{D_{\delta_{tef}}}(\alpha, M)\delta_{tef} \\
 &\quad + \Delta C_{D_{\delta_r}}(\alpha, \delta_r, M) + C_{D_{\delta_{SB}}}(\alpha, M)\delta_{SB} \\
 &\quad + \Delta C_{D_{l_g}}(\alpha)\Delta l_g
 \end{aligned} \tag{A.4}$$

For flight in the plane of symmetry:

$$\delta_r = 0 \tag{A.5}$$

And assuming landing gear up ($\Delta l_g = 0$), no speed brake deflection ($\delta_{SB} = 0$) and no extra external stores ($\Delta C_{D_{ES}} = 0$):

$$\begin{aligned}
 C_D &= C_{D_0}(\alpha, M) + (\Delta C_D)_{C_L}(C_L, M) + \Delta C_{D_{\delta_h}}(\alpha, \delta_h, M) \\
 &\quad + C_{D_{\delta_{lef}}}(\alpha, M)\delta_{lef} + C_{D_{\delta_{tef}}}(\alpha, M)\delta_{tef}
 \end{aligned} \tag{A.6}$$

A.1.3 Pitching Moment Coefficient

$$\begin{aligned}
 C_m &= f(\alpha, \beta, q, M, h, \dot{\alpha}, \delta_h, \delta_a, \delta_r, \delta_{lef}, \delta_{tef}, \delta_{SB}, \Delta lg) \\
 &= C_{m_0}(\alpha, M) + (\Delta C_{m_0})_f(\alpha, M, h) + \Delta C_{m_{lex}}(\alpha, M) \\
 &\quad + \Delta C_{m_{\delta_h}}(\alpha, \delta_h, M) + [C_{m_{\delta_{lef}}}(\alpha, M) + (\Delta C_{m_{\delta_{lef}}})_f(\alpha, M, h)]\delta_{lef} \\
 &\quad + [C_{m_{\delta_{tef}}}(\alpha, M) + (\Delta C_{m_{\delta_{tef}}})_f(\alpha, M, h)]\delta_{tef} + \delta C_{m_\beta}(\alpha, \beta, M) \\
 &\quad + \Delta C_{m_{\delta_a}}(\alpha, \delta_a, M) + \Delta C_{m_{\delta_r}}(\alpha, \delta_r, M) + \Delta C_{m_{ES}} \\
 &\quad + C_{m_{\delta_{SB}}}(\alpha, M)\delta_{SB} + \Delta C_{m_g}(\alpha)\Delta lg \\
 &\quad + \frac{\bar{c}}{2V} [C_{m_q}(\alpha, M) + (\Delta C_{m_q})_f(M, h)]q \\
 &\quad + \frac{\bar{c}}{2V} [C_{m_{\dot{\alpha}}}(\alpha, M) + (\Delta C_{m_{\dot{\alpha}}})_f(M, h)]\dot{\alpha}
 \end{aligned} \tag{A.7}$$

For flight in the plane of symmetry:

$$\beta = \delta_a = \delta_r = 0 \tag{A.8}$$

And assuming landing gear up ($\Delta lg = 0$), no speed brake deflection ($\delta_{SB} = 0$) and no extra external stores ($\Delta C_{m_{ES}} = 0$):

$$\begin{aligned}
 C_m &= C_{m_0}(\alpha, M) + (\Delta C_{m_0})_f(\alpha, M, h) + \Delta C_{m_{lex}}(\alpha, M) \\
 &\quad + \Delta C_{m_{\delta_h}}(\alpha, \delta_h, M) + [C_{m_{\delta_{lef}}}(\alpha, M) + (\Delta C_{m_{\delta_{lef}}})_f(\alpha, M, h)]\delta_{lef} \\
 &\quad + [C_{m_{\delta_{tef}}}(\alpha, M) + (\Delta C_{m_{\delta_{tef}}})_f(\alpha, M, h)]\delta_{tef} \\
 &\quad + \frac{\bar{c}}{2V} [C_{m_q}(\alpha, M) + (\Delta C_{m_q})_f(M, h)]q \\
 &\quad + \frac{\bar{c}}{2V} [C_{m_{\dot{\alpha}}}(\alpha, M) + (\Delta C_{m_{\dot{\alpha}}})_f(M, h)]\dot{\alpha}
 \end{aligned} \tag{A.9}$$

A.1.4 Flap Schedule

The deflection of the leading edge flaps (δ_{lef}) and the trailing edge flaps (δ_{tef}) is determined by a set of subroutines based on Mach number, altitude and angle of attack.

$$\delta_{lef} = f(\alpha, M, h) \quad (A.10)$$

$$\delta_{tef} = f(\alpha, M, h) \quad (A.11)$$

The flap schedule is used to alleviate a discontinuity in the aerodynamic model.

A.2 Thrust Model

The F/A - 18 engine model calculates the thrust produced by each engine as a function of altitude, Mach number and throttle setting. Fuel flow and other dynamic engine parameters are calculated as well, but are not used in the present study.

$$T = f(h, M, \delta_T) \quad (A.12)$$

The engines on the F/A - 18 are positioned at an angle of 1.98 degrees from the center line of the aircraft. Therefore it is necessary to consider the reduction of thrust due to this situation. Of interest in the present study is the total thrust in the x - direction T_x .

$$T_x = 2T \cos(1.98^\circ) \quad (A.13)$$

A.3 F/A - 18 Constants

The F/A - 18 fighter escort configuration is characterized by:

60% Fuel

Wing tip missiles

Total weight $W = 33,310$ lb.

Mass $m = 1034.47$ slugs

Wing area $S = 400$ ft.²

Wing span $b = 37.42$ ft.

Mean aerodynamic chord $\bar{c} = 11.52$ ft.

Aerodynamic Center:

x-location wrt cg $l_x = -0.297$ ft.

z-location wrt cg $l_z = 0.23$ ft.

Thrust Centerline:

z-location wrt cg $l_{z_t} = 0.23$ ft.

Control surface limits are:

Stabilator $-24^\circ \leq \delta_h \leq 10.5^\circ$ (+ for trailing edge down)

Trailing edge flap $-8^\circ \leq \delta_{trf} \leq 45^\circ$ (+ for trailing edge down)

Leading edge flap $-3^\circ \leq \delta_{lef} \leq 34^\circ$ (+ for trailing edge down)

Throttle limits are:

$$30^\circ \leq \delta_T \leq 106.5^\circ$$

Idle Maximum Military Thrust

$$106.5^\circ \leq \delta_T \leq 131^\circ$$

Minimum Maximum Afterburner

Appendix B. Modified F/A - 18 Model

The NASA Aerodynamic Model is not well suited for use with a nonlinear programming routine. This is primarily due to the linear interpolation used in the model. Therefore, a B-spline interpolation is created using the NASA model as a basis. Several modifications to the NASA model are necessary to make the B-spline useful and to correct the discontinuity in the NASA model at transonic Mach numbers and low altitudes.

B.1 Modifications

Assuming:

$$\begin{aligned} C_L &= (C_L)_{q=0} + \frac{\bar{c}}{2V} [C_{L_q} + (\Delta C_{Lq})_f] q + \frac{\bar{c}}{2V} [C_{L_{\dot{\alpha}}} + (\Delta C_{L_{\dot{\alpha}}})_f] \dot{\alpha} \\ &= (C_L)_{q=0} + (C_{L_q})_m q + (C_{L_{\dot{\alpha}}})_M \dot{\alpha} \end{aligned} \tag{B.1}$$

$$\begin{aligned}
C_m &= (C_m)_{q=0} + \frac{\bar{c}}{2V} [C_{m_q} + (\Delta C_{m_q})_f]_q + \frac{\bar{C}}{2V} [C_{m_i} + (\Delta C_{m_i})_f]_{\dot{\alpha}} \\
&= (C_m)_{q=0} + (C_{m_q})_{m_q} + (C_{m_i})_{M\dot{\alpha}}
\end{aligned} \tag{B.2}$$

where:

$$\begin{aligned}
(C_L)_{q=0} &= f(\alpha, \delta_h, M, \delta_{lef}, \delta_{tef}, h) \\
&= C_{L_0}(\alpha, M) + (\Delta C_{L_0})_f(\alpha, M, h) + \Delta C_{L_{\delta_h}}(\alpha, \delta_h, M) \\
&\quad + [C_{L_{\delta_{lef}}}(\alpha, M) + (\Delta C_{L_{\delta_{lef}}})_f(\alpha, M, h)]\delta_{lef} \\
&\quad + [C_{L_{\delta_{tef}}}(\alpha, M) + (\Delta C_{L_{\delta_{tef}}})_f(\alpha, M, h)]\delta_{tef}
\end{aligned} \tag{B.3}$$

$$\begin{aligned}
(C_m)_{q=0} &= f(\alpha, \delta_h, M, \delta_{lef}, \delta_{tef}, h) \\
&= C_{m_0}(\alpha, M) + (\Delta C_{m_0})_f(\alpha, M, h) + \Delta C_{m_{\text{tex}}}(\alpha, M) \\
&\quad + \Delta C_{m_{\delta_h}}(\alpha, \delta_h, M) \\
&\quad + [C_{m_{\delta_{lef}}}(\alpha, M) + (\Delta C_{m_{\delta_{lef}}})_f(\alpha, M, h)]\delta_{lef} \\
&\quad + [C_{m_{\delta_{tef}}}(\alpha, M) + (\Delta C_{m_{\delta_{tef}}})_f(\alpha, M, h)]\delta_{tef}
\end{aligned} \tag{B.4}$$

$$\begin{aligned}
(C_{L_q})_M &= f(\alpha, M, h) \\
&= \frac{\bar{c}}{2V} [C_{L_q}(\alpha, M) + (\Delta C_{L_q})_f(M, h)]
\end{aligned} \tag{B.5}$$

$$\begin{aligned}
(C_{m_q})_M &= f(\alpha, M, h) \\
&= \frac{\bar{c}}{2V} [C_{m_q}(\alpha, M) + (\Delta C_{m_q})_f(M, h)]
\end{aligned} \tag{B.6}$$

$$\begin{aligned}
(C_{L_i})_M &= f(\alpha, M, h) \\
&= \frac{\bar{C}}{2V} [C_{L_i}(\alpha, M) + (\Delta C_{L_i})_f(M, h)]
\end{aligned} \tag{B.7}$$

$$\begin{aligned}
(C_{m_z})_M &= f(\alpha, M, h) \\
&= \frac{\bar{c}}{2V} [C_{m_z}(\alpha, M) + (\Delta C_{m_z})_f(M, h)]
\end{aligned}
\tag{B.8}$$

The utility of having the lift and pitching moment coefficient in this form will be apparent in the description of the B-spline.

In order to obtain the data in this form, it is necessary to make following modifications to the outputs of NASA subroutine "REFAERO" :

1) The subroutine is called with pitch rate equal to zero. The lift coefficient (C_L) returned is equal to $(C_L)_{q=0}$ and the pitching moment coefficient (C_m) returned is equal to $(C_m)_{q=0}$.

2) Coefficients C_{L_q} , C_{m_q} , $(\Delta C_{L_q})_f$ and $(\Delta C_{m_q})_f$ are added as outputs.

3) Coefficients $(C_L)_M$ and $(C_m)_M$ are calculated as defined above.

4) Coefficients $(C_{L_z})_M$ and $(C_{m_z})_M$ are used as they are output by "REFAERO".

These coefficients are calculated, as defined above, inside the subroutine "REFAERO".

5) A correction factor is added to the lift, drag, and pitching moment coefficients for Mach numbers above 0.6 and angles of attack below forty degrees (40°). This is necessary to correct a discontinuity which occurs in this region.

$$(C_L)_{q=0} = (C_L)_{q=0}(\alpha, \delta_h, M, \delta_{lef}, \delta_{ief}, h) + \Delta C_{L_{CORR}}(\delta_h, M, \delta_{lef}, \delta_{ief}, h)
\tag{B.9}$$

$$C_D = C_D(\alpha, \delta_h, M, C_L, \delta_{lef}, \delta_{ief}) + \Delta C_{D_{CORR}}(\delta_h, M, C_L, \delta_{lef}, \delta_{ief})
\tag{B.10}$$

$$(C_m)_{q=0} = (C_m)_{q=0}(\alpha, \delta_h, M, \delta_{lef}, \delta_{ief}, h) + \Delta C_{m_{CORR}}(\delta_h, M, \delta_{lef}, \delta_{ief}, h)
\tag{B.11}$$

where:

$$\Delta C_{L_{CORR}} = \begin{cases} 0 & M \leq 0.5 \\ (C_L)_{q=0}(\alpha = 40^\circ) - (C_L)_{q=0}(\alpha = 39^\circ) & M > 0.5, \alpha < 40^\circ \\ 0 & M > 0.5, \alpha \geq 40^\circ \end{cases} \quad (B.12)$$

$$\Delta C_{D_{CORR}} = \begin{cases} 0 & M \leq 0.5 \\ C_D(\alpha = 40^\circ) - C_D(\alpha = 39^\circ) & M > 0.5, \alpha > 40^\circ \\ 0 & M > 0.5, \alpha \leq 40^\circ \end{cases} \quad (B.13)$$

$$\Delta C_{m_{CORR}} = \begin{cases} 0 & M \leq 0.5 \\ (C_m)_{q=0}(\alpha = 40^\circ) - (C_m)_{q=0}(\alpha = 39^\circ) & M > 0.5, \alpha < 40^\circ \\ 0 & M > 0.5, \alpha \leq 40^\circ \end{cases} \quad (B.14)$$

The correction factors are calculated in a separate subroutine.

B.2 B - Spline Interpolation

The B-spline interpolation routines used were developed by the International Mathematical and Statistical Library (IMSL, Ref. 6). This process consists of the use of four subroutines to create and evaluate the B-spline. The algorithm is based on one by De Boor (Ref. 6). The subroutines listed in their order of use are:

Subroutine	Purpose
1. "BSNAK"	Computes the required knot sequence
2. "BS3IN"	Computes 3 dimensional spline coefficients
or "BS2IN"	Computes 2 dimensional spline coefficients

3. "BS3VL" Evaluates spline interpolant (3 dimensional)
- or "BS2VL" Evaluates spline interpolant (2 dimensional)
4. "BS3DR" Evaluates the derivative of the spline interpolant (3 dimensional)
- or "BS2DR" Evaluates the derivative of the spline interpolant (2 dimensional)

The first two subroutines are used in subroutine "BCSMOD" to set up the spline coefficients which are passed to the subroutine "COEFF" for evaluation. The coefficients are passed using common block "SPLINE". "COEFF" is the subroutine which is called by all programs to get the aerodynamic coefficients.

The subroutine "BCSMOD" is used to implement the modifications discussed in part A) and is only called once in order to set up the spline. Subroutine "COEFF" can then be used to evaluate the spline using the last two subroutines above.

The following is a discussion of the details concerning the choice of knots and the limits imposed on the modified model.

B.2.1 Lift Coefficients

$$\begin{aligned}
 C_L &= (C_L)_{q=0} + (C_L)_M q + (C_L)_M \dot{\alpha} \\
 &= f(\alpha, \delta_h, M, \delta_{lef}, \delta_{ref}, h)
 \end{aligned}
 \tag{B.15}$$

The leading edge and trailing edge flaps are scheduled as a function of angle of attack, Mach number, and altitude. Therefore $(C_L)_{q=0}$ is actually a function of four variables: angle of attack, stabilator deflection, Mach number, and altitude. A three dimensional B-spline is used to model $(C_L)_{q=0}$ for a given altitude.

The coefficients $(C_{L_q})_M$ and $(C_{L_{\dot{\alpha}}})_M$ are modelled using a two dimensional B-spline for each altitude. This is possible because these coefficients are not a function of stabilator deflection. The nodes chosen for each coefficient are given in section B.2.4.

B.2.2 Drag Coefficient

The drag coefficient is modelled using a three dimensional B-spline for a given altitude. This coefficient is not a function of pitch rate or angle of attack derivative and therefore it is unnecessary to model the corresponding coefficients. The node sequence is given in section B.2.4.

B.2.3 Pitching Moment Coefficients

The pitching moment coefficient is modelled using the same scheme as with the lift coefficient.

$$\begin{aligned} C_m &= (C_m)_{q=0} + (C_{m_q})_M q + (C_{L_{\dot{\alpha}}})_M \dot{\alpha} \\ &= f(\alpha, \delta_h, M, \delta_{lef}, \delta_{lef}, h) \end{aligned} \tag{B.16}$$

As with the lift coefficient, $(C_m)_{q=0}$ is modelled using a three dimensional B-spline for each altitude. The coefficients $(C_{m_q})_M$ and $(C_{L_{\dot{\alpha}}})_M$ are modelled using two dimensional B-spline. See section B.2.4 for the node sequence used.

B.2.4 Node Sequences

The following tables contain the node sequences used to create the modified F/A - 18 Model.

Table 2. Stabilator Nodes (deg)

$(C_L)_{q=0}$	C_D	$(C_m)_{q=0}$
-24	-24	-24
-12	-12	-12
-6	-6	-11
-3	-3	-10
0	0	-9
3	3	-8
6	6	-7
10.5	10.5	-6
		-3
		0
		3
		6
		10.5

Table 3. Mach Number Nodes

$(C_L)_{q=0}$	C_D	$(C_m)_{q=0}$	$(C_{Lq})_M$	$(C_{mq})_M$	$(C_{Lq})_M$	$(C_{mq})_M$
			0.05			
			0.10			
			0.20			
			0.40			
			0.60			
			0.80			
			0.85			
			0.90			

Table 4. Angle of Attack Nodes (deg)

$(C_L)_{q=0}$	C_D	$(C_m)_{q=0}$	$(C_{Lq})_M$	$(C_{mq})_M$	$(C_{Lz})_M$	$(C_{mz})_M$
-10	-10	-10	-10	-10	-10	-10
-8	-8	-8	-5	-5	-8	-8
-6	-6	-6	2	2	-6	-6
-4	-4	-4	4	4	-4	-4
-2	-2	-2	6	6	-2	-2
0	0	0	8	8	0	0
2	2	2	11	12	2	2
4	4	4	13	14	4	4
6	6	6	16	16	6	6
8	8	8	18	18	8	8
10	10	10	19	20	10	10
12	12	12	21	22	12	12
14	14	14	22	24	14	14
16	16	16	24	26	16	16
18	18	18	28	28	18	18
20	20	20	32	30	20	20
22	22	22	36	32	22	22
24	24	24	38	34	24	24
26	26	26	44	36	26	26
28	28	28	46	38	28	28
30	30	30	48	39	30	30
32	32	32	50	41	32	32
34	34	34	55	42	34	34
36	36	35	60	45	36	35
38	38	36	65	47	38	36
50	50	37	70	48	50	37
55	55	38	75	49	55	38
60	60	39	80	51	60	39
65	65	40	85	55	65	40
70	70	41	90	60	70	41
75	75	42		64	75	42
80	80	43		66	80	43
85	85	44		70	85	44
90	90	45		75	90	45
		50		80		50
		55		90		55
		60				60
		65				65
		70				70
		75				75
		80				80
		85				85
		90				90

Table 5. Model Limits

$-10^\circ \leq \alpha \leq 90^\circ$
$-24^\circ \leq \delta_h \leq 10.5^\circ$
$0.05 \leq M \leq 0.90$

Appendix C. Equations of Motion

The equations of motion for the longitudinal flight of the aircraft are developed using Stalford (Ref. 14). Assuming flight in the symmetric plane and that the center of gravity is the origin of the body axes, the equations of motion are given as:

$$\dot{u} = -qw - g \sin \theta + \frac{T_x}{m} + \frac{\rho V^2 S}{2m} C_x \quad (\text{C.1})$$

$$\dot{w} = qu + g \cos \theta + \frac{\rho V^2 S}{2m} C_z \quad (\text{C.2})$$

$$\dot{q} = \frac{1}{I_y} [l_{ze} T_x + \frac{\rho V^2 S}{2} (\bar{c} C_m + l_z C_x - l_x c_z)] \quad (\text{C.3})$$

$$\dot{\theta} = q \quad (\text{C.4})$$

$$\dot{h} = u \sin \theta - w \cos \theta \quad (\text{C.5})$$

The transformation to wind axes is used as follows:

$$\alpha = \tan^{-1} \frac{w}{u} \quad (C.6)$$

$$V^2 = u^2 + w^e \quad (C.7)$$

$$\bar{q} = \frac{1}{2} \rho V^2 \quad (C.8)$$

$$C_x = C_L \sin \alpha - C_d \cos \alpha \quad (C.9)$$

$$C_z = -C_L \cos \alpha - C_D \sin \alpha \quad (C.10)$$

Using this transformation, the equations of motion become:

$$mV\dot{\alpha} = mVq + mg \cos(\theta - \alpha) - \bar{q}SC_L - T_x \sin \alpha \quad (C.11)$$

$$m\dot{V} = -mg \sin(\theta - \alpha) - \bar{q}SC_D + T_x \cos \alpha \quad (C.12)$$

$$I_y \dot{q} = l_z T_x + \bar{q}S[\bar{c}C_m + (l_x \cos \alpha + l_z \sin \alpha)C_L + (l_x \sin \alpha - l_z \cos \alpha)C_D] \quad (C.13)$$

$$\dot{\theta} = q \quad (C.14)$$

$$\dot{h} = V \sin(\theta - \alpha) \quad (C.15)$$

letting

$$d_L(\alpha) = l_x \cos \alpha + l_z \sin \alpha \quad (C.16)$$

$$d_D(\alpha) = l_x \sin \alpha - l_z \cos \alpha \quad (C.17)$$

$$\gamma = \theta - \alpha \quad (C.18)$$

it follows that:

$$\dot{\alpha} = q + \frac{g}{V} \cos \gamma - \frac{\bar{q}S}{mV} C_L - \frac{T_x}{mV} \sin \alpha \quad (\text{C.19})$$

$$\dot{V} = -g \sin \gamma - \frac{\bar{q}S}{m} C_D + \frac{T_x}{m} \cos \alpha \quad (\text{C.20})$$

$$\dot{q} = \frac{1}{I_y} [l_{ze}T_x + \bar{q}s(\bar{c}C_m + d_L C_L + d_D C_D)] \quad (\text{C.21})$$

$$\dot{\theta} = q \quad (\text{C.22})$$

$$\dot{\gamma} = q - \dot{\alpha} \quad (\text{C.23})$$

$$\dot{h} = V \sin \gamma \quad (\text{C.24})$$

We remark that the F/A-18 aerodynamic coefficients C_m , C_L and C_D are defined with respect to the aerodynamic center. Consequently, we need the last two terms in equation (C.21) to account for the moment about the center of gravity rather than the aerodynamic center.

In the half-loop maneuver, a special definition for the pitch angle (θ) is used. It is allowed to have values from 0° to 360° . Therefore, for values greater than 90° , it no longer is defined in the conventional manner. This definition makes it necessary to define flight path angle (γ) in the same manner.

Appendix D. Computer Programs

```
C=====
C=====
C===== F18_FEEDBACK.FOR F/A-18 FEEDBACK SUBROUTINES =====
C=====
C===== THIS FILE CONTAINS FEEDBACK SUBROUTINES =====
C===== FOR THE F/A-18 AIRCRAFT =====
C=====
C-----
C
C F/A-18 FEEDBACK SUBROUTINE
C
C THIS SUBROUTINE RETURNS THE FEEDBACK CONTROL FOR
C HALF-LOOP AND SPLIT-S MANEUVER.
C
C INPUTS - ALPDEG ANGLE OF ATTACK (deg)
C GAMDEG FLIGHT PATH ANGLE (deg)
C GAMMAF FINAL FLIGHT PATH ANGLE (deg)
C MACH MACH NUMBER
C Q PITCH RATE (rad/s)
C IPORT FEEDBACK INDICATOR
C = 0 INITIALIZATION
C = 1 FIRST TRANSITION : PART I
C = 2 FIRST TRANSITION : PART II
C = 3 FIRST TRANSITION : PART III
C = 4 OUTER LAYER
C = 5 FINAL TRANSITION
C = 0 END OF MANEUVER
C ALT ALTITUDE (ft)
C AKM MANEUVER INDICATOR
C = 1. FOR HALF-LOOP
```

```

C                                     = - 1. FOR SPLIT-S
C
C   OUTPUTS -   DHT           STABILATOR ANGLE           (deg)
C               ASTAR        OUTER LAYER ANGLE OF ATTACK (deg)
C               QDSTAR       OUTER LAYER PITCH RATE      (deg/s)
C               IPORT        FEEDBACK INDICATOR
C                               (AS ABOVE)
C
C   AUTHOR  -   FREDERICK E. GARRETT, JR.
C               GRADUATE RESEARCH ASSISTANT
C               AEROSPACE ENGINEERING
C               VPI & SU, BLACKSBURG, VA.
C
C   DATE    -   APRIL 1988
C
C-----
C   SUBROUTINE FEDBCK(DHT      ,ASTAR  ,QDSTAR  ,
C                   =f(
C   1               ALPDEG ,GAMDEG ,GAMMAF  ,MACH  ,
C   2               Q      ,ALT    ,IPOINT  ,AKM   )
C   REAL MACH
C   DTR = ACOS(-1.) / 180.
C-----
C   DETERMINE PORTION OF MANEUVER
C-----
C   IF(IPOINT .EQ. 0) THEN
C       IERF = 0
C       IPOINT = 1
C   END IF
C   GAMREF = GAMDEG * AKM
C   CALL OLAYER(DHSTAR ,ASTAR  ,QDSTAR ,GAMREF ,MACH  ,ALT  ,
C   1           IERF   )
C   CALL FINLAY(QHAT   ,ALPDEG ,GAMDEG ,MACH  ,IPOINT )
C   QSTAR = QDSTAR * DTR
C   GOTO(100,200,300,400,500),IPOINT
C-----
C   FIRST TRANSITION : PART I
C-----
C   100 CONTINUE
C   CALL FTRANS(DELALP ,
C   1           ALPDEG ,GAMDEG ,Q      ,QSTAR  ,MACH  )
C   QCHK = ABS(Q - QHAT)
C   ACHK = ABS(ASTAR - ALPDEG)
C   IF(QCHK .LE. 1.E-4 .OR. ACHK .LE. DELALP) THEN
C       IPOINT = 2
C       GOTO 200
C   ELSE
C       IF(Q .LT. QHAT) THEN
C           IF(ALPDEG .LT. 45.0 .OR. ALPDEG .GT. 55.0) THEN
C               DHT = -24.0

```

```

        ELSE
            DHT = 10.5
        END IF
    ELSE
        IF(ALPDEG .LT. 45.0 .OR. ALPDEG .GT. 55.0) THEN
            DHT = 10.5
        ELSE
            DHT = -24.0
        END IF
    END IF
END IF
RETURN
C -----
C FIRST TRANSITION : PART II
C -----
200 CONTINUE
CALL FTRANS(DELALP ,
1          ALPDEG ,GAMDEG ,Q          ,QSTAR          ,MACH )
ACHK = ABS(ASTAR - ALPDEG)
IF(ACHK .LT. DELALP) THEN
    IPORT = 3
    GOTO 300
ELSE
    IF(ALPDEG .LT. ASTAR) THEN
        IF(ALPDEG .LT. 45.0 .OR. ALPDEG .GT. 55.0) THEN
            DHT = -24.0
        ELSE
            DHT = 10.5
        END IF
    ELSE
        IF(ALPDEG .LT. 45.0 .OR. ALPDEG .GT. 55.0) THEN
            DHT = 10.5
        ELSE
            DHT = -24.0
        END IF
    END IF
END IF
RETURN
C -----
C FIRST TRANSITION: PART III
C -----
300 CONTINUE
QCHK = ABS(Q - QSTAR)
ACHK = ABS(ALPDEG - ASTAR)
IF(QCHK .LE. 1.E-3 .AND. ACHK .LE. 3.E-1) THEN
    IPORT = 4
    GOTO 400
ELSE
    IF(Q .LT. QSTAR) THEN
        DHT = -24.0
    
```

```

        ELSE
            DHT = 10.5
        END IF
    END IF
    RETURN
C -----
C   OUTER LAYER
C -----
400 CONTINUE
    CALL STRANS(DELGAM ,
1          ALPDEG ,GAMDEG ,Q          ,MACH )
    GAMCHK = GAMMAF - DELGAM
    IF(GAMREF .GE. GAMCHK) THEN
        IPORT = 5
        GOTO 500
    ELSE
        DHT = DHSTAR
    END IF
    RETURN
C -----
C   FINAL TRANSITION
C -----
500 CONTINUE
    IF(ABS(Q) .LT. 1.E-3) THEN
        IPORT = 0
    ELSE
        DHT = 10.5
    END IF
    RETURN
    END
C -----
C
C   F/A-18 FIRST INNER LAYER SUBROUTINE
C
C   THIS SUBROUTINE CALCULATES THE PITCH RATE FOR
C   MAXIMUM d(alpha)/dt IN THE FIRST INNER LAYER.
C
C   INPUTS - ALPDEG    ANGLE OF ATTACK        (deg)
C            GAMDEG    FLIGHT PATH ANGLE      (deg)
C            MACH       MACH NUMBER
C            IPORT      FEEDBACK INDICATOR
C
C   OUTPUTS - QHAT     PITCH RATE              (rad/s)
C                  FOR MAXIMUM d(alpha)/dt
C
C   AUTHOR - FREDERICK E. GARRETT, JR.
C            GRADUATE RESEARCH ASSISTANT
C            AEROSPACE ENGINEERING
C            VPI & SU, BLACKSBURG, VA
C

```

```

C   DATE       - APRIL 1988
C
C-----
C   SUBROUTINE FINLAY(QHAT ,
C                   =f(
1       ALPDEG , GAMDEG , MACH , IPORT )
REAL MACH
REAL MASS
COMMON/FCONT /
1     BETDEG , ALT     , SPSD   , RHO   , P     , R     ,
2     HGCL   , DAR    , DAL    , DRR   , DRL   , DSBK  ,
3     DLG    , DEBUG  , QSE
COMMON/CONST /
1     G      , DUMMY  , DTR
COMMON/F18FC /
1     CBAR   , B      , S      , EIY   , ELZ   , ALZ   ,
2     ALX    , MASS
COMMON/AMISC /
1     AKM    , TX     , QBAR   , KDER
C-----
C   DETERMINE CONTROL
C-----
C   IF(IPORT .GT. 3) THEN
C       DHT = 10.5
C   ELSE
C       DHT = -24.0
C   END IF
C-----
C   CALCULATE MODEL
C-----
C   CALL COEFF(CL    , CD    , CM    , CLQ   , CMQ   ,
1       CLO    , CDO    , CMO    , CLAD  , CMAD  ,
2       CLA    , CDA    , CMA    , CLDH  , CDDH  ,
3       CMDH   , CLOA   , CMOA   , CLODH , CMODH ,
4       CLQA   , CMQA   ,
5       ALPDEG , DHT    , MACH   , 0.    )
C-----
C   CALCULATE QHAT
C-----
ALP = ALPDEG * DTR
GAMMA= GAMDEG * DTR
DL = ALX * COS(ALP) + ALZ * SIN(ALP)
DD = ALX * SIN(ALP) - ALZ * COS(ALP)
VEL = MACH * SPSD
ADOT = (1. - QBAR * S * CLQ / (MASS * VEL)) * Q
1     + G / VEL * COS(GAMMA)
2     - QBAR * S * CLO / (MASS * VEL)
3     - TX * SIN(ALP) / (MASS * VEL)
ADOT = ADOT / (1. + QBAR * S * CLAD / (MASS * VEL))
QHAT = - (ELZ * TX + QBAR * S

```



```

IF(Q .LT. QSTAR) THEN
  DHT = -24.0
ELSE
  DHT = 10.5
END IF

C -----
C CALCULATE MODEL
C -----
CALL COEFF(CL      ,CD      ,CM      ,CLQ      ,CMQ      ,
1          CLO      ,CDO      ,CMO      ,CLAD      ,CMAD      ,
2          CLA      ,CDA      ,CMA      ,CLDH      ,CDDH      ,
3          CMDH      ,CLOA      ,CMOA      ,CLODH      ,CMODH      ,
4          CLQA      ,CMQA      ,
5          ALPDEG ,DHT      ,MACH      ,0.      )

C -----
C CALCULATE F1 AND F2
C -----
ALP = ALPDEG * DTR
GAMMA= GAMDEG * DTR
DL = ALX * COS(ALP) + ALZ * SIN(ALP)
DD = ALX * SIN(ALP) - ALZ * COS(ALP)
VEL = MACH * SPSD
ADOT = (1. - QBAR * S * CLQ / (MASS * VEL)) * Q
1      + G / VEL * COS(GAMMA)
2      - QBAR * S * CLO / (MASS * VEL)
3      - TX * SIN(ALP) / (MASS * VEL)
ADOT = ADOT / (1. + QBAR * S * CLAD / (MASS * VEL))
F1 = QBAR * S * (CBAR * CMQ + DL * CLQ) / EIY
F2 = (ELZ * TX + QBAR * S
1      * (CBAR * CMO + DL * CLO + DD * CDO
2      + (CBAR * CMAD + DL * CLAD) * ADOT))
3      / (F1 * EIY)

C -----
C DELT
C -----
DELT = LOG(ABS((QSTAR + F2) / (Q + F2))) / F1

C -----
C DELALP
C -----
F3 = 1. - QBAR * S * CLQ / (MASS * VEL)
F3 = F3 / (1. + QBAR * S * CLAD / (MASS * VEL))
F4 = G / VEL * COS(GAMMA)
1      - QBAR * S * CLO / (MASS * VEL)
2      - TX * SIN(ALP) / (MASS * VEL)
F4 = F4 / (1. + QBAR * S * CLAD / (MASS * VEL))
DELALP = (F4 - F3 * F2) * DELT
1      + F3 * (F2 + Q) * (EXP(F1 * DELT) - 1.) / F1
DELALP = DELALP / DTR
RETURN

```

```

END
-----
C
C
C   F/A-18 SECOND TRANSITION SUBROUTINE
C
C   THIS SUBROUTINE CALCULATES THE VALUES USED TO
C   DETERMINE WHEN THE END OF THE SECOND TRANSITION
C   HAS BEEN REACHED.
C
C   INPUTS  -  ALPDEG    ANGLE OF ATTACK          (deg)
C              GAMDEG    FLIGHT PATH ANGLE        (deg)
C              Q          PITCH RATE              (rad/s)
C              MACH      MACH NUMBER
C
C   OUTPUTS -  DELGAM    DELTA GAMDEG             (deg)
C
C   AUTHOR  -  FREDERICK E. GARRETT, JR.
C              GRADUATE RESEARCH ASSISTANT
C              AEROSPACE ENGINEERING
C              VPI & SU, BLACKSBURG, VA
C
C   DATE    -  APRIL 1988
C
-----
C   SUBROUTINE STRANS(DELGAM ,
C                   =f(
1     ALPDEG ,GAMDEG ,Q      ,MACH )
REAL MACH
REAL MASS
COMMON/FCONT /
1     BETDEG ,ALT      ,SPSD  ,RHO   ,P     ,R     ,
2     HGCL   ,DAR     ,DAL   ,DRR   ,DRL   ,DSBK  ,
3     DLG    ,DEBUG   ,QSE
COMMON/CONST /
1     G      ,DUMMY   ,DTR
COMMON/F18FC /
1     CBAR   ,B       ,S     ,EIY   ,ELZ   ,ALZ   ,
2     ALX    ,MASS
COMMON/AMISC /
1     AKM    ,TX      ,QBAR  ,KDER
C
-----
C   DETERMINE CONTROL SETTING
C
-----
C   DHT = 10.5
C
-----
C   CALCULATE MODEL
C
-----
CALL COEFF(CL      ,CD      ,CM      ,CLQ     ,CMQ     ,
1     CLO      ,CDO      ,CMO      ,CLAD    ,CMAD    ,
2     CLA      ,CDA      ,CMA      ,CLDH    ,CDDH    ,

```

```

3          CMDH  ,CLOA  ,CMOA  ,CLODH  ,CMODH  ,
4          CLQA  ,CMQA  ,
5          ALPDEG ,DHT   ,MACH  ,0.      )
C -----
C  CALCULATE F1 AND F2
C -----
ALP = ALPDEG * DTR
GAMMA= GAMDEG * DTR
DL = ALX * COS(ALP) + ALZ * SIN(ALP)
DD = ALX * SIN(ALP) - ALZ * COS(ALP)
VEL = MACH * SPSD
ADOT = (1. - QBAR * S * CLQ / (MASS * VEL)) * Q
1      + G / VEL * COS(GAMMA)
2      - QBAR * S * CLO / (MASS * VEL)
3      - TX * SIN(ALP) / (MASS * VEL)
ADOT = ADOT / (1. + QBAR * S * CLAD / (MASS * VEL))
F1 = QBAR * S * (CBAR * CMQ + DL * CLQ) / EIY
F2 = (ELZ * TX + QBAR * S
1      * (CBAR * CMO + DL * CLO + DD * CDO
2      + (CBAR * CMAD + DL * CLAD) * ADOT))
3      / (F1 * EIY)
C -----
C  DELT
C -----
DELT = LOG( F2 / (Q + F2)) / F1
C -----
C  DELGAM
C -----
F5 = QBAR * S * CLQ / (MASS * VEL)
F6 = - G * COS(GAMMA) / VEL
1      + QBAR * S * (CLO + CLAD * ADOT) / (MASS * VEL)
2      + TX * SIN(ALP) / (MASS * VEL)
DELGAM = (F6 - F5 * F2) * DELT
1      = + F5 * (F2 + Q) * (EXP(F1 * DELT) - 1) / F1
DELGAM = DELGAM / DTR
RETURN
END
C -----
C  F/A-18 OUTER LAYER SUBROUTINE
C
C  THIS SUBROUTINE CALCULATES THE FEEDBACK CONTROL
C  FOR THE OUTER LAYER
C
C  INPUTS - GAMDEG  FLIGHT PATH ANGLE      (deg)
C          MACH     MACH NUMBER
C          ALT      ALTITUDE                (ft)
C          IERF     = 0 READ INTERPOLATION DATA
C                  = 1 DO INTERPOLATION
C

```

```

C   OUTPUTS - DHSTAR   STABILATOR ANGLE           (deg)
C               ASTAR   ANGLE OF ATTACK           (deg)
C               QSTAR   PITCH RATE                (deg/s)
C
C   AUTHOR -  FREDERICK E. GARRETT, JR.
C             GRADUATE RESEARCH ASSISTANT
C             AEROSPACE ENGINEERING
C             VPI & SU, BLACKSBURG,VA
C
C   DATE    -  APRIL 1988
C
C-----
C   SUBROUTINE OPLAYER(DHSTAR ,ASTAR ,QSTAR ,
C                   = f(
1     GAMDEG ,MACH ,ALT ,IERF )
C   COMMON/OLTBL /
1     ALPHA ,Q ,DHT
C   REAL MACH
C   REAL ALPHA ( 19 , 7 , 6 )
C   REAL Q ( 19 , 7 , 6 )
C   REAL DHT ( 19 , 7 , 6 )
C   REAL AS ( 2 )
C   REAL QS ( 2 )
C   REAL DHS ( 2 )
C   CHARACTER*80 HEADER
C   IF(IERF .EQ. 1) GOTO 100
C-----
C   READ INTERPOLATION DATA
C-----
C   DO 50 K = 1,6
C     DO 40 J = 1,7
C       READ(30,5) HEADER
5     FORMAT(80A)
C     DO 30 I = 1,19
C       READ(30,10) ALPHA(I,J,K),Q(I,J,K),
1     DHT(I,J,K)
C     FORMAT(3F14.7)
C   30 CONTINUE
C   40 CONTINUE
C   50 CONTINUE
C     IERF = 1
C-----
C   DETERMINE INTERPOLATION NODES
C-----
100 I = INT(GAMDEG / 10.) + 1
C   IF(I .LT. 1) I = 1
C   IF(I .GE. 19) I = 18
C   GA = FLOAT(I-1) * 10.0
C   GA1 = GA + 10.0
C   J = INT(MACH * 10.) - 2

```

```

AMACH = MACH
IF(J .LT. 1) THEN
  J = 1
  AMACH = 0.3
END IF
IF(J .GE. 7) THEN
  J = 6
  AMACH = 0.9
END IF
AM = FLOAT(J+2) / 10.
AM1 = AM + 0.1
K = INT((ALT / 1000. - 5.) / 10.) + 1
IF(K .LT. 1) K = 1
IF(K .GE. 6) K = 5
AL = (FLOAT(K-1) * 10. + 5.) * 1000.
AL1 = AL + 10000.
DO 200 L = 1,2
  M = K + L - 1
C-----
C INTERPOLATE FOR OUTER LAYER ANGLE OF ATTACK (ASTAR)
C-----
      ALJ = ATERP(GA ,GA1
1          ALPHA(I ,J ,M ) ,ALPHA(I+1,J ,M ) ,
2          GAMDEG )
      ALJ1 = ATERP(GA ,GA1
1          ALPHA(I ,J+1,M ) ,ALPHA(I+1,J+1,M ) ,
2          GAMDEG )
      AS(L) = ATERP(AM ,AM1
1          ALJ ,ALJ1 ,
2          AMACH )
C-----
C INTERPOLATE FOR OUTER LAYER Q (QSTAR)
C-----
      QJ = ATERP(GA ,GA1
1          Q(I ,J ,M ) ,Q(I+1,J ,M ) ,
2          GAMDEG )
      QJ1 = ATERP(GA ,GA1
1          Q(I ,J+1,M ) ,Q(I+1,J+1,M ) ,
2          GAMDEG )
      QS(L) = ATERP(AM ,AM1
1          QJ ,QJ1 ,
2          AMACH )
C-----
C INTERPOLATE FOR OUTER LAYER DHT (DHSTAR)
C-----
      DHJ = ATERP(GA ,GA1
1          DHT(I ,J ,M ) ,DHT(I+1,J ,M ) ,
2          GAMDEG )
      DHJ1 = ATERP(GA ,GA1
1          DHT(I ,J+1,M ) ,DHT(I+1,J+1,M ) ,

```

```

2          GAMDEG          )
DHS(L) = ATERP(AM          ,AM1          ,
1          DHJ            ,DHJ1          ,
2          AMACH          )
200 CONTINUE
  ASTAR = ATERP(AL        ,AL1        ,
1          AS(1)        ,AS(2)        ,
2          ALT          )
  QSTAR = ATERP(AL        ,AL1        ,
1          QS(1)        ,QS(2)        ,
2          ALT          )
  DHSTAR = ATERP(AL        ,AL1        ,
1          DHS(1)       ,DHS(2)       ,
2          ALT          )
  RETURN
  END
-----
C
C   LINEAR INTERPOLATION FUNCTION <FEG>
C
-----
FUNCTION ATERP(X1 ,X2 ,Y1 ,Y2 ,X )
  ATERP = Y2 + (X - X2) * (Y1 - Y2) / (X1 - X2)
  RETURN
  END
-----
C
C
C   F/A-18 BASS-GURA FORMULA POLE PLACEMENT SUBROUTINE
C
C   THIS SUBROUTINE CALCULATES THE FEEDBACK GAINS
C   NECESSARY TO PLACE THE POLES OF THE SYSTEM EQUAL
C   TO THE SECOND ORDER BUTTERWORTH POLES.
C
C   ie. CHARACTERISTIC EQUATION GIVEN BY:
C
C    $2 + \sqrt{2} \lambda \omega + \omega^2 = 0$ 
C
C   SUBROUTINE COEFF MUST BE CALLED BEFORE CALLING
C   THIS SUBROUTINE
C
C   INPUTS - ALPDEG      ANGLE OF ATTACK      (deg)
C           DHT         STABILATOR DEFLECTION (deg)
C           MACH        MACH NUMBER
C           OMEGA       SPEED OF THE BUTTERWORTH POLES
C                   thalf ]
C                   ] = ln(2) / [sqrt(2) * omega]
C                   tdouble ]
C                   period = 1 / [sqrt(2) * omega]
C
C   OUTPUTS - K         GAIN ROW VECTOR OF DIMENSION 2
C                   K( 1 , 1 ) => GAIN FOR delta ALPHA
C                   K( 1 , 2 ) => GAIN FOR delta Q

```

```

C          IERR          ERROR INDICATOR
C          IERR = 0      NO ERROR
C          IERR = 1      MATRIX IS INVERTABLE IN
C                          SUBROUTINE MATIVX
C          IERR = 100    SYSTEM UNCONTROLABLE
C                          (U IS NOT FULL RANK)

```

```

C  AUTHOR -  FREDERICK E. GARRETT, JR.
C            GRADUATE RESEARCH ASSISTANT
C            AEROSPACE ENGINEERING
C            VPI & SU, BLACKSBURG, VA

```

```

C  DATE    -  DECEMBER 1987

```

```

-----
C  SUBROUTINE BASGURA(K          , IERR          ,
C                        =f(
1      ALPDEG , MACH          , OMEGA          )
COMMON/FCONT /
1      BETDEG , ALT          , SPSD          , RHO          , P          , R          ,
2      HGCL   , DAR          , DAL          , DRR          , DRL          , DSBK          ,
3      DLG    , DEBUG        , QSE
COMMON/CONST /
1      G      , DUMMY        , DTR
COMMON/F18FC /
1      CBAR   , B            , S            , EIY          , ELZ          , ALZ          ,
2      ALX    , MASS
COMMON/ACOFF /
1      CL     , CD           , CM           , CLQ          , CMQ          , CLO          ,
2      CDO    , CMO          , CLAD         , CMAD         , CLA          , CDA          ,
3      CMA    , CLDH         , CDDH         , CMDH         , CLOA         , CMOA         ,
4      CLODH  , CMODH        , CLQA         , CMQA
COMMON/AMISC /
1      AKM    , TX           , QBAR         , KDER
REAL MACH
REAL MASS
REAL A ( 2 , 2 )
REAL AB ( 2 , 1 )
REAL BB ( 2 , 1 )
REAL KD ( 1 , 2 )
REAL O ( 1 , 2 )
REAL U ( 2 , 2 )
REAL UW ( 2 , 2 )
REAL W ( 2 , 2 )
REAL CA ( 2 )
REAL OA ( 2 )
REAL K ( 1 , 2 )
LOGICAL QSE
LOGICAL DEBUG
-----

```

```

C      INITIALIZATION
C-----
      V      = MACH * SPSD
      ALP    = ALPDEG * DTR
      PL1    = ALX * COS(ALP) + ALZ * SIN(ALP)
      PL2    = ALX * SIN(ALP) - ALZ * COS(ALP)
      IERR   = 0
      AMV    = MASS * V
C-----
C      CALCULATE SYSTEM MATRIX  A
C-----
C      d( d(alpha)/dt )/d( alpha )
C-----
      A( 1 , 1 ) = - QBAR * S * CLA / AMV
      2          - TX * COS(ALP) / AMV
C-----
C      d( d(alpha)/dt )/d( q )
C-----
      A( 1 , 2 ) = 1 - QBAR * S * CLQ / AMV
C-----
C      d( d(q)/dt )/d( alpha )
C-----
      A( 2 , 1 ) = QBAR * S / EIY
      1          * (CBAR * CMA
      2            + PL1 * CLA
      3            + PL2 * CDA
      4            - (ALX * CL - ALZ * CD) * SIN(ALP)
      5            + (ALX * CD + ALZ * CL) * COS(ALP))
C-----
C      d( d(q)/dt )/d( q )
C-----
      A( 2 , 2 ) = QBAR * S / EIY
      1          * (CBAR * CMQ
      2            + PL1 * CLQ
      3            + PL2 * CDQ)
C-----
C      CALCULATE CONTROL MATRIX  BB
C-----
C      d( d(alpha)/dt )/d( dht )
C-----
      BB( 1 , 1 ) = - QBAR * S * CLDH / AMV
C-----
C      d( d(q)/dt )/d( dht )
C-----
      BB( 2 , 1 ) = QBAR * S / EIY
      1          * (CBAR * CMDH
      2            + PL1 * CLDH

```

```

3          + PL2 * CDDH)
C-----
C  CALCULATE COEFFICIENT VECTORS
C-----
C
C  OPEN LOOP COEFFICIENT VECTOR  OA
C-----
C  OA( 1 ) = -(A( 1 , 1 ) + A( 2 , 2 ))
C  OA( 2 ) = A( 1 , 1 ) * A( 2 , 2 )
1          - A( 1 , 2 ) * A( 2 , 1 )
C-----
C  CLOSED LOOP COEFFICIENT VECTOR  CA
C-----
C  CA( 1 ) = SQRT( 2. ) * OMEGA
C  CA( 2 ) = OMEGA * OMEGA
C-----
C  CALCULATE THE TOTAL COEFFICIENT MATRIX O
C-----
C  O ( 1 , 1 ) = CA( 1 ) - OA( 1 )
C  O ( 1 , 2 ) = CA( 2 ) - OA( 2 )
C-----
C  TEST FOR CONTROLLABILITY
C-----
C
C  CALCULATE CONTROLLABILITY MATRIX U
C-----
C  U( 1 , 1 ) = BB( 1 , 1 )
C  U( 2 , 1 ) = BB( 2 , 1 )
C  CALL MULT(AB ,2 ,2 ,1 ,A ,BB )
C  U( 1 , 2 ) = AB( 1 , 1 )
C  U( 2 , 2 ) = AB( 2 , 1 )
C-----
C  DETERMINE RANK OF U
C-----
C  DETU = U( 1 , 1 ) * U( 2 , 2 )
1          - U( 1 , 2 ) * U( 2 , 1 )
C  IF( DETU .EQ. 0.) THEN
C      IERR = 100
C      RETURN
C  END IF
C-----
C  INVERT CONTROLLABILITY MATRIX U
C-----
C  U11 =  U( 2 , 2 ) / DETU
C  U12 = - U( 2 , 1 ) / DETU
C  U21 = - U( 1 , 2 ) / DETU
C  U22 =  U( 1 , 1 ) / DETU
C  U( 1 , 1 ) = U11
C  U( 1 , 2 ) = U12
C  U( 2 , 1 ) = U21

```

```

      U( 2 , 2 ) = U22
C -----
C   U IS NOW THE INVERSE OF THE
C   CONTROLLABILITY MATRIX
C -----
C -----
C   CREATE BASS-GURA MATRIX W
C   NOTE:
C     THIS MATRIX IS ACTUALLY THE TRANSPOSE
C     OF THE MATRIX W IN THE ALGORITHM
C -----
      W( 1 , 1 ) = 1.
      W( 1 , 2 ) = OA( 1 )
      W( 2 , 1 ) = 0.
      W( 2 , 2 ) = 1.
C -----
C   INVERT W
C -----
      DETW = W( 1 , 1 ) * W( 2 , 2 )
1      - W( 1 , 2 ) * W( 2 , 1 )
      W11 = W( 2 , 2 ) / DETW
      W12 = - W( 2 , 1 ) / DETW
      W21 = - W( 1 , 2 ) / DETW
      W22 = W( 1 , 1 ) / DETW
      W( 1 , 1 ) = W11
      W( 1 , 2 ) = W12
      W( 2 , 1 ) = W21
      W( 2 , 2 ) = W22
C -----
C   W IS NOW EQUAL TO THE INVERSE OF THE
C   TRANSPOSE OF W IN THE ALGORITHM
C -----
C -----
C   CALCULATE THE GAIN MATRIX K
C -----
      CALL MULT(WU ,2 ,2 ,2 ,W ,U )
      CALL MULT(K ,1 ,2 ,2 ,0 ,WU )
10 CONTINUE
      RETURN
      END
C -----
C
C   MATRIX MULTIPLICATION SUBROUTINE <FEG>
C
C   INPUTS:
C
C     N      ROW DIMENSION OF MATRIX A
C     K      COLUMN DIMENSION OF MATRIX A
C            AND ROW DIMENSION OF MATRIX B
C     M      COLUMN DIMENSION OF MATRIX B

```

```

C      A      n x k MATRIX
C      B      k x m MATRIX
C
C      OUTPUTS:
C
C      AB      AB(n x m) = A(n x k) * B(k x m)
C-----
C      SUBROUTINE MULT(AB ,
C                      =f(
1      N ,K ,M ,A ,B )
C      REAL A ( N , K )
C      REAL B ( K , M )
C      REAL AB ( N , M )
C      DO 30 I = 1 , N
C          DO 20 J = 1 , M
C              AB( I , J ) = 0.
C              DO 10 L = 1 , K
C                  AB( I , J ) = AB( I , J )
1      + A( I , L ) * B( L , J )
C      10 CONTINUE
C      20 CONTINUE
C      30 CONTINUE
C      RETURN
C      END
C-----
C
C      F/A-18 OUTER LAYER STABILITY SUBROUTINE
C
C      THIS SUBROUTINE CALCULATES THE EIGENVALUES OF
C      LINEARIZED EQUATIONS FOR THE FAST VARIABLES
C      IN THE OUTER LAYER
C
C      INPUTS - ALPDEG      ANGLE OF ATTACK (deg)
C              MACH        MACH NUMBER
C
C      OUTPUTS - A          SYSTEM MATRIX
C                      A( 1 , 1 ) => d( d(alpha)/dt ) / d(alpha)
C                      A( 1 , 2 ) => d( d(alpha)/dt ) / d(q)
C                      A( 2 , 1 ) => d( d(q)/dt ) / d(alpha)
C                      A( 2 , 2 ) => d( d(q)/dt ) / d(q)
C
C      ALAMR1      REAL PART OF FIRST EIGENVALUE
C      ALAMI1      IMAGINARY PART OF FIRST EIGENVALUE
C      ALAMR2      REAL PART OF SECOND EIGENVALUE
C      ALAMI2      IMAGINARY PART OF SECOND EIGENVALUE
C
C      AUTHOR - FREDERICK E. GARRET, JR.
C              GRADUATE RESEARCH ASSISTANT
C              AEROSPACE ENGINEERING
C              VPI & SU, BLACKSBURG,VA

```

```

C
C   DATE       -   MARCH 1988
C
C-----
C   SUBROUTINE STBLTY(A           ,ALAMR1 ,ALAMI1 ,ALAMR2 ,ALAMI2 ,
C                               =f(
C   1                               ALPDEG ,MACH   )
C   COMMON/FCONT /
C   1   BETDEG ,ALT   ,SPSD   ,RHO   ,P     ,R     ,
C   2   HGCL   ,DAR   ,DAL    ,DRR   ,DRL   ,DSBK  ,
C   3   DLG    ,DEBUG ,QSE
C   COMMON/CONST /
C   1   G      ,DUMMY ,DTR
C   COMMON/F18FC /
C   1   CBAR   ,B     ,S      ,EIY   ,ELZ   ,ALZ   ,
C   2   ALX    ,MASS
C   COMMON/ACOFF /
C   1   CL     ,CD    ,CM     ,CLQ   ,CMQ   ,CLO   ,
C   2   CDO   ,CMO   ,CLAD   ,CMAD   ,CLA   ,CDA   ,
C   3   CMA   ,CLDH  ,CDDH   ,CMDH   ,CLOA  ,CMOA  ,
C   4   CLODH ,CMODH ,CLQA   ,CMQA
C   COMMON/AMISC /
C   1   AKM    ,TX    ,QBAR   ,KDER
C   REAL MACH
C   REAL MASS
C   REAL A( 2 , 2 )
C-----
C   INITIALIZATION
C-----
C   V      = MACH * SPSD
C   ALP    = ALPDEG * DTR
C-----
C   CALCULATE SYSTEM MATRIX
C-----
C-----
C   d( d(alpha)/dt ) / d(alpha)
C-----
C   A(1,1) = - QBAR * S * CLA / (MASS * V)
C   2      - TX   * COS(ALP) / (MASS * V)
C-----
C-----
C   d( d(alpha)/dt ) / d(q)
C-----
C   A(1,2) = 1 - QBAR * S * CLQ / (MASS * V)
C-----
C-----
C   d( d(q)/dt ) / d(alpha)
C-----
C   A(2,1) = QBAR * S / EIY
C   1      * (CBAR * CMA
C   2      + (ALX * COS(ALP) + ALZ * SIN(ALP)) * CLA
C   3      + (ALX * SIN(ALP) - ALZ * COS(ALP)) * CDA

```

```

4          - (ALX * CL - ALZ * CD) * SIN(ALP)
5          + (ALX * CD + ALZ * CL) * COS(ALP))
C -----
C      d( d(q)/dt ) / d(q)
C -----
      A(2,2) = QBAR * S / EIY
1          * (CBAR * CMQ
2          + (ALX * COS(ALP) + ALZ * SIN(ALP)) * CLQ)
C -----
C      CALCULATE EIGENVALUES
C
C      det(LAMDA * I - A) = 0.
C          or
C      2 LAMQADB * LAMDA + QUADC = 0.
C -----
      QUADA = 1.
      QUADB = - (A(1,1) + A(2,2))
      QUADC = A(1,1) * A(2,2) - A(1,2) * A(2,1)
      RAD   = QUADB * QUADB - 4 * QUADA * QUADC
      IF(RAD .LT. 0.) THEN
          ALAMR1 = - QUADB / (2 * QUADA)
          ALAMR2 = ALAMR1
          ALAMI1 = SQRT(-RAD) / (2 * QUADA)
          ALAMI2 = - ALAMI1
      ELSE
          ALAMR1 = (- QUADB + SQRT(RAD)) / (2 * QUADA)
          ALAMR2 = (- QUADB - SQRT(RAD)) / (2 * QUADA)
          ALAMI1 = 0.
          ALAMI2 = 0.
      END IF
      RETURN
      END
C+++++
C+++
C+++      F18_SIMULATION.FOR      F/A-18 SIMULATION PROGRAM      +++
C+++
C+++      PURPOSE      -      TO SIMULATE THE LONGITUDINAL      +++
C+++                        FLIGHT OF THE F/A-18 FIGHTER      +++
C+++                        ESCORT CONFIGURATION      +++
C+++
C+++      PRECISION -      SINGLE      +++
C+++
C+++      REQUIRED      -      F18_SUB.FOR      +++
C+++                        F18_AERO_TABLES.DAT      +++
C+++                        F18_ENG_TABLES.DAT      +++
C+++                        IMSL MATH/LIBRARY      +++
C+++
C+++      AUTHOR      -      FREDERICK E. GARRETT, JR.      +++
C+++                        GRADUATE RESEARCH ASSISTANT      +++
C+++                        AEROSPACE ENGINEERING DEPT.      +++

```

```

C+++          VPI & SU, BLACKSBURG,VA          +++
C+++          DATE          -    DECEMBER 1987    +++
C+++          C+++          +++

```

```

C-----
COMMON/FCONT /
1      BETDEG ,ALT      ,SPSD      ,RHO      ,P      ,R      ,
2      HGCL   ,DAR     ,DAL      ,DRR     ,DRL     ,DSBK   ,
3      DLG    ,DEBUG   ,QSE
COMMON/CONST /
1      G      ,DUMMY   ,DTR
COMMON/F18FC /
1      CBAR   ,B      ,S      ,EIY     ,ELZ     ,ALZ     ,
2      ALX    ,MASS
COMMON/ACOFF /
1      CL     ,CD      ,CM      ,CLQ     ,CMQ     ,CLO     ,
2      CDO    ,CMO     ,CLAD    ,CMAD    ,CLA     ,CDA     ,
3      CMA    ,CLDH   ,CDDH   ,CMDH   ,CLOA    ,CMOA    ,
4      CLODH  ,CMODH  ,CLQA   ,CMQA
COMMON/AMISC /
1      AKM    ,TX      ,QBAR   ,KDER
COMMON/WORKSP/
1      RWKSP
REAL MACH
REAL MASS
REAL A      ( 2 , 2 )
REAL K      ( 1 , 2 )
REAL RWKSP ( 5137 )
REAL YPRIME ( 6 )
REAL Y      ( 6 )
LOGICAL DEBUG
LOGICAL QSE
CHARACTER*80 HEADER
CHARACTER*10 MANVER
CALL IWKIN ( 5137 )

```

```

C-----
C      INITIALIZATION
C-----
C
C-----
C      GENERAL CONSTANTS
C-----
G      = 32.174
DUMMY  = 1.
DTR    = ACOS(-1.) / 180.
KINV   = 0
KDER   = 1
C-----
C      FLIGHT CONDITIONS
C-----
BETDEG = 0.

```

```

ALT      = 15000.
CALL ATMOS(RHO ,SPSD ,ALT )
P        = 0.
R        = 0.
HGCL     = ALT
C -----
C ENGINE PARAMETERS
C -----
WDTF     = DUMMY
GK1      = DUMMY
P1       = DUMMY
P2       = DUMMY
TAU      = DUMMY
TESTV    = DUMMY
VAE      = DUMMY
C -----
C CHOOSE MANEUVER
C       AKM = 1. => HALF-LOOP
C       AKM = -1. => SPLIT-S
C -----
AKM      = 1.
IF( AKM .EQ. 1. ) THEN
  MANVER = 'HALF-LOOP'
ELSE
  MANVER = 'SPLIT-S'
END IF
C -----
C F/A-18 CONSTANTS FIGHTER ESCORT CONFIGURATION
C -----
CBAR     =      11.52
XBAR     =      18.0
ZBAR     =       6.0
B        =     37.42
S        =     400.0
EIY      = 151293.0
ELZ      =       2.8 / 12.0
ALZ      =       2.8 / 12.0
ALX      =      - 3.56 / 12.0
MASS     =     1034.47
C -----
C IGNORABLE CONTROL SETTINGS
C -----
DAR      = 0.
DAL      = 0.
DRR      = 0.
DRL      = 0.
DSBK     = 0.
DLG      = 0.
DTH      = 130.
C -----

```

C LOGICAL DECLARATIONS FOR SUBROUTINE REFAERO

C

```
-----  
DEBUG = .FALSE.  
QSE   = .TRUE.  
CALL IAERO (HEADER)  
CALL IENG  (HEADER)
```

C

C INITIAL STATES AND CONTROLS

C

```
-----  
MACH = 0.6  
IF(MACH .EQ. 0.3) THEN  
  ALPDEG = 11.68  
  DHT    = - 1.675  
END IF  
IF(MACH .EQ. 0.6) THEN  
  ALPDEG = 2.884  
  DHT    = - 0.1561  
END IF  
IF(MACH .EQ. 0.9) THEN  
  ALPDEG = 1.314  
  DHT    = 0.5223  
END IF  
Y( 1 ) = ALPDEG * DTR  
Y( 2 ) = MACH * SPSD  
QD     = 0.0  
Y( 3 ) = QD * DTR  
THETAD = ALPDEG  
THOUT  = THETAD  
GAMDEG = 0.0  
Y( 4 ) = THETAD * DTR  
Y( 5 ) = ALT  
Y( 6 ) = 0.0  
T      = 0.0  
TTRIM = 0.0
```

C

C INITIALIZE INTEGRATION SUBROUTINE

C

```
-----  
DT      = 1.E-2  
KSTEP  = 4 * INT(1.E-2 / DT + 1.E-6)  
IPORT  = 0  
IERF   = 0  
NOPOLE = 1  
GAMMAF = 180.  
OMEGA  = 1.5  
N      = 6  
INTG   = 1  
DO 10 I = 1,N  
  YPRIME(I) = 0.  
10 CONTINUE
```

C

```

C   CALCULATE B-SPLINE COEFFICIENTS
C   USING SUBROUTINE BCSMOD
C   -----
C   CALL BCSMOD
C   -----
C   INITIALIZE OUTPUT FILES
C   -----
      WRITE(10,11) MANVER ,ALT
11  FORMAT(' F/A-18 SIMULATION (STATES)',/,
1    ' FOR THE ',A10,' MANEUVER',/,
2    ' AT ',F9.2,' ft',//,
3    '   t(sec)  alpha(deg)  v(ft/s)  q(deg/s) ',
4    '   theta(deg) MACH ',//)
C   WRITE(11,12) MANVER ,ALT
C   12  FORMAT(' F/A-18 SIMULATION (STATE DERIVATIVES)',/,
C   1    ' FOR THE ',A10,' MANEUVER',/,
C   2    ' AT ',F9.2,' ft',//,
C   3    '   t(sec)  d(alpha)/dt  d(v)/dt(ft/s/s) ',
C   4    '   d(q)/dt  d(theta)/dt  d(z)/dt  d(x)/dt',//)
      WRITE(12,13) MANVER ,ALT
13  FORMAT(' F/A-18 SIMULATION (FEEDBACK DATA)',/,
1    ' FOR THE ',A10,' MANEUVER',/,
2    ' AT ',F9.2,' ft',//,
3    '   t(sec)  dh*(deg)  alpha*(deg)  q*(deg/s) ' ,//)
      WRITE(13,14) MANVER ,ALT
14  FORMAT(' F/A-18 SIMULATION (LOAD FACTORS)',/,
1    ' FOR THE ',A10,' MANEUVER',/,
2    ' AT ',F9.2,' ft',//,
32)'   t(sec)  anz(g's)  anx(g's)  dynamic pressure(lb/ft
4    ' altitude(ft)  range(ft)  theta(deg)',//)
      WRITE(15,51) MANVER ,ALT
51  FORMAT(' F/A-18 SIMULATION (GAINS AND CONTROLS)',/,
1    ' FOR THE ',A10,' MANEUVER',/,
2    ' AT ',F9.2,' ft',//,
3    '   t(sec)  Kalpha  Kq  dh(deg)  IPORT'
4    ,//)
C   -----
C   SIMULATION TIME LOOP
C
C   Y( 1 ) => ANGLE OF ATTACK (rad)
C   Y( 2 ) => VELOCITY (ft/s)
C   Y( 3 ) => PITCH RATE (rad/s)
C   Y( 4 ) => PITCH ANGLE (rad)
C   Y( 5 ) => ALTITUDE (ft)
C   Y( 6 ) => RANGE (ft)
C   -----
      DO 20 I = 1,100000
C   -----
C   UPDATE VELOCITY TERMS
C   -----

```

```

      QBAR = 0.5 * RHO * Y( 2 ) * Y( 2 )
      CO2VT = CBAR / ( 2 * Y( 2 ) )
      BO2VT = B / ( 2 * Y( 2 ) )
C -----
C CALCULATE THRUST
C -----
      CALL ENGI (TH ,WDTF ,GK1 ,P1 ,P2 ,
1          TAU ,TESTV ,TMIL ,TAUGMN ,DTH ,
2          MACH ,ALT ,VAE )
      TX = 2 * TH * COS(1.98 * DTR)
C -----
C CALL FEEDBACK SUBROUTINE
C -----
      IF(T .GE. TTRIM) THEN
          QQF = Y(3)
          CALL FEDBCK(DHSTAR ,ASTAR ,QSTAR ,
1          ALPDEG ,GAMDEG ,GAMMAF ,MACH ,
2          QQF ,ALT ,IPORT ,AKM )
          IF(NOPOLE .EQ. 0 .AND. IPORT .EQ. 4) THEN
C -----
C POLE PLACEMENT
C -----
              DA = Y( 1 ) - ASTAR * DTR
              DQ = Y( 3 ) - QSTAR * DTR
              CALL BASGURA(K ,IERR ,ALPDEG ,
1          MACH ,OMEGA )
              DHT = DHSTAR - ( K( 1 , 1 ) * DA
1          + K( 1 , 2 ) * DQ ) / DTR
              IF( DHT .GT. 10.5 ) DHT = 10.5
              IF( DHT .LT. -24.0 ) DHT = -24.0
              ELSE
                  DHT = DHSTAR
              END IF
          END IF
C -----
C CALCULATE AERODYNAMIC COEFFICIENTS
C -----
          CALL COEFF (CL ,CD ,CM ,CLQ ,CMQ ,
1          CLO ,CDO ,CMO ,CLAD ,CMAD ,
2          CLA ,CDA ,CMA ,CLDH ,CDDH ,
3          CMDH ,CLOA ,CMOA ,CLODH ,CMODH ,
4          CLQA ,CMQA ,
5          ALPDEG ,DHT ,MACH ,0. )
C -----
C CALL DIFFERENTIAL EQUATIONS
C -----
          CALL RHS(N ,T ,Y ,YPRIME )
C -----
C CALCULATE FAST EQUATION EIGENVALUES
C -----

```

```

      IF(IPORT .EQ. 4) THEN
        CALL STBLTY(A      ,ALAMR1 ,ALAMI1 ,ALAMR2 ,ALAMI2 ,
1          ALPDEG ,MACH  )
      END IF
C -----
C CALCULATE LOAD FACTORS
C -----
      ANZG = QBAR * S * (CDO * SIN(Y(1)))
1          + (CLO + CLQ * Y(3) + CLAD * YPRIME(1)) * COS(Y(1))
2          / (MASS * G)
      ANZP = (XBAR * YPRIME(3) + ZBAR * Y(3) * Y(3)) / G
      ANZ = ANZG + ANZP
      ANXG = QBAR * S * (- CDO * COS(Y(1)))
1          + (CLO + CLQ * Y(3) + CLAD * YPRIME(1)) * SIN(Y(1))
2          / (MASS * G)
      ANXP = (- XBAR * Y(3) * Y(3) + ZBAR * YPRIME(3)) / G
      ANX = ANXG + ANXP
C -----
C OUTPUT EVERY 10th STEP
C -----
      IF( MOD((I-1),KSTEP) .EQ. 0 ) THEN
        WRITE(10,15) T ,ALPDEG  ,Y( 2 )  ,QD      ,THETAD  ,
1          MACH
15      FORMAT( F7.3 , 4F12.4 , F9.3 )
C      WRITE(11,16) T ,YPRIME(1),YPRIME(2),YPRIME(3),YPRIME(4),
C      YPRIME(5),YPRIME(6)
C 16      FORMAT( F7.3 , 6F14.7 )
        TTOW = TX / 33310.
        WRITE(11,16) T ,TX      ,TTOW
16      FORMAT( F7.3 , 2F14.7 )
        WRITE(12,17) T ,DHSTAR ,ASTAR ,QSTAR
17      FORMAT( F7.3 , F14.7 , F10.3 , F14.7 )
        WRITE(13,18) T ,ANZ    ,ANX    ,QBAR    ,ALT    ,
1          Y(6)      ,THOUT
18      FORMAT( F7.3 , 6F11.3 )
        WRITE(14,55) T ,ANZG    ,ANXG    ,ANZP    ,ANXP
55      FORMAT( F7.3 , 4F12.3 )
        WRITE(15,56) T ,K(1,1) ,K(1,2) ,DHT    ,GAMDEG ,
1          IPORT
56      FORMAT( F7.3 , 4F14.7 ,I3)
        IF(IPORT .EQ. 4) THEN
          WRITE(16,57) T ,ALAMR1 ,ALAMR2
57          FORMAT( F7.3 , 2F14.7)
          WRITE(17,58) T ,ALAMI1 ,ALAMI2
58          FORMAT( F7.3 , 2F14.7)
        END IF
      END IF
      IF(IPORT .EQ. 3) THEN
        DT = 1.E-3
        KSTEP = 4 * INT(1.E-2 / DT + 1.E-6)

```

```

      IF(T .GT. 3.0) STOP
      ELSE
        DT      = 1.E-2
        KSTEP   = 4 * INT(1.E-2 / DT + 1.E-6)
      END IF
      IF(IPORT .EQ. 5) THEN
        K(1,1) = 0.
        K(1,2) = 0.
      END IF
      IF(GAMDEG .GT. 200.0) STOP
      IF(IPORT .EQ. 0) STOP
C -----
C CALL INTEGRATION ROUTINE
C -----
      CALL XGRAT(N      ,Y      ,YPRIME ,INTG ,T      ,DT )
C -----
C TRANSITION FROM STATE MATRIX TO STATE VARIABLES
C -----
      ALT      = Y( 5 )
      CALL ATMOS(RHO ,SPSD ,ALT )
      ALPDEG   = Y( 1 ) / DTR
      MACH     = Y( 2 ) / SPSD
      QD      = Y( 3 ) / DTR
      THETAD   = Y( 4 ) / DTR
      GAMDEG   = THETAD - ALPDEG
      IF(THETAD .LE. 90.0) THEN
        THOUT   = THETAD
      ELSE
        THOUT   = 180 - THETAD
      END IF
20 CONTINUE
      STOP
      END
C -----
C DIFFERENTIAL EQUATION SUBROUTINE
C
C PURPOSE - TO COMPUTE THE NUMERICAL APPROXIMATION
C           FOR THE DIFFERENTIAL EQUATIONS FOR THE
C           F/A-18 LONGITUDINAL MODEL
C           REQUIRED FOR IMSL SUBROUTINE DVERK
C
C INPUTS - N      NUMBER OF EQUATIONS
C          T      TIME (sec)
C          Y( 1 ) ANGLE OF ATTACK (rad)
C          Y( 2 ) VELOCITY (ft/s)
C          Y( 3 ) PITCH RATE (rad/s)
C          Y( 4 ) PITCH ANGLE (rad)
C          Y( 5 ) ALTITUDE (ft)
C
C OUTPUTS - YPRIME( I ) d( Y( I ) ) / dt

```

```

C
C   AUTHOR   -   FREDERICK E. GARRETT, JR.
C               GRADUATE RESEARCH ASSISTANT
C               VPI & SU, BLACKSBURG, VA.
C
C   DATE     -   DECEMBER 1987
C

```

```

C-----
SUBROUTINE RHS(N      ,T      ,Y      ,YPRIME )

```

```

COMMON/CONST /

```

```

1      G      ,DUMMY  ,DTR

```

```

COMMON/F18FC /

```

```

1      CBAR   ,B      ,S      ,EIY   ,ELZ   ,ALZ   ,

```

```

2      ALX    ,MASS

```

```

COMMON/ACOFF /

```

```

1      CL     ,CD     ,CM     ,CLQ   ,CMQ   ,CLO   ,

```

```

2      CDO    ,CMO    ,CLAD   ,CMAD   ,CLA   ,CDA   ,

```

```

3      CMA    ,CLDH   ,CDDH   ,CMDH   ,CLOA  ,CMOA  ,

```

```

4      CLODH  ,CMODH  ,CLQA   ,CMQA

```

```

COMMON/AMISC /

```

```

1      AKM    ,TX     ,QBAR   ,KDER

```

```

REAL MASS

```

```

REAL MACH

```

```

REAL Y      ( 6 )

```

```

REAL YPRIME ( 6 )

```

```

C-----
d(alpha)/dt

```

```

YPRIME ( 1 ) = ( 1 - QBAR * S * CLQ / ( MASS * Y( 2 ) ) ) * Y( 3 )

```

```

1      + ( AKM * G * COS(Y( 4 ) - AKM * Y( 1 ) ) ) / Y( 2 )

```

```

2      - ( QBAR * S * CLO ) / ( MASS * Y( 2 ) )

```

```

3      - ( TX * SIN(Y( 1 ) ) ) / ( MASS * Y( 2 ) )

```

```

YPRIME ( 1 ) = YPRIME( 1 ) / ( 1 + QBAR * S * CLAD

```

```

1      / ( MASS * Y( 2 ) )

```

```

C-----
d(V)/dt

```

```

YPRIME ( 2 ) = - G * SIN(Y( 4 ) - AKM * Y( 1 ) )

```

```

1      - ( QBAR * S * CDO ) / MASS

```

```

2      + TX * COS(Y( 1 ) ) / MASS

```

```

C-----
d(q)/dt

```

```

YPRIME ( 3 ) = ( ELZ * TX

```

```

1      + QBAR * S

```

```

2      * ( CBAR * ( CMO + CMQ * Y(3)

```

```

3      + CMAD * YPRIME( 1 ) )

```

```

4      + ( ALX * COS(Y( 1 ) ) + ALZ * SIN(Y( 1 ) ) )

```

```

5      * ( CLO + CLQ * Y( 3 )

```

```

6      + CLAD * YPRIME( 1 ) )

```

```

7          + (ALX * SIN(Y( 1 )) - ALZ * COS(Y( 1 )))
8          * CDO
9          )
&        ) / EIY
C -----
C   d(theta)/dt
C -----
C   YPRIME ( 4 ) = AKM * Y( 3 )
C -----
C   d(h)/dt
C -----
C   YPRIME ( 5 ) = Y( 2 ) * SIN(Y( 4 ) - Y( 1 ))
C -----
C   d(x)/dt
C -----
C   YPRIME ( 6 ) = Y( 2 ) * COS(Y( 4 ) - Y( 1 ))
RETURN
END

```

```

C -----
C
C   INTEGRATION SUBROUTINE
C
C   FOURTH ORDER RUNGE-KUTTA INTEGRATOR
C
C   INPUTS  -  N      NUMBER OF DIFFERENTIAL EQUATIONS
C             Y(N)   INITIAL STATES
C             YDOT(N) DERIVATIVES OF STATES wrt TIME
C             INTG   INTEGRATION STEP ( =1 TO INITIALIZE)
C             T     TIME
C             H     TIME STEP
C
C   OUTPUT  -  Y(N)   UPDATED STATES
C             INTG   INTEGRATION STEP
C
C   AUTHOR  -  DR. HAROLD L. STALFORD
C             ASSOCIATE PROFESSOR
C             AEROSPACE ENGINEERING
C             VPI & SU, BLACKSBURG, VA.
C -----

```

```

SUBROUTINE XGRAT(N      ,Y      ,YDOT ,INTG ,T      ,H )
COMMON/XINTG /
1   XK      ,X      ,Q
REAL XK   ( 7 )
REAL Q    ( 7 )
REAL Y    ( 7 )
REAL YDOT ( 7 )
GOTO(10,20,30,40),INTG
10 CONTINUE
DO 15 J = 1,N

```

```

      XK(J) = H * YDOT(J)
      X     = 0.5 * (XK(J) - 2.0 * Q(J))
      Y(J)  = Y(J) + X
      Q(J)  = Q(J) + 3.0 * X - 0.5 * XK(J)
15 CONTINUE
      T     = T + 0.5 * H
      INTG = 2
      RETURN
20 CONTINUE
      DUM = 1.0 - SQRT(0.5)
      DO 25 J = 1,N
          XK(J) = H * YDOT(J)
          X     = DUM * (XK(J) - Q(J))
          Y(J)  = Y(J) + X
          Q(J)  = Q(J) + 3.0 * X - DUM * XK(J)
25 CONTINUE
      INTG = 3
      RETURN
30 CONTINUE
      DUM = 1.0 + SQRT(0.5)
      DO 35 J = 1,N
          XK(J) = H * YDOT(J)
          X     = DUM * (XK(J) - Q(J))
          Y(J)  = Y(J) + X
          Q(J)  = Q(J) + 3.0 * X - DUM * XK(J)
35 CONTINUE
      INTG = 4
      T     = T + 0.5 * H
      RETURN
40 CONTINUE
      DUM = 1.0 / 6.0
      DO 45 J = 1,N
          XK(J) = H * YDOT(J)
          X     = DUM * (XK(J) - 2.0 * Q(J))
          Y(J)  = Y(J) + X
          Q(J)  = Q(J) + 3.0 * X - 0.5 * XK(J)
45 CONTINUE
      INTG = 1
      RETURN
      END

```

```

=====
C=====
C=====
C===== F18_SUB.FOR F/A-18 MODEL SUBROUTINES =====
C=====
C===== THIS FILE CONTAINS THE MODEL SUBROUTINES =====
C===== FOR THE F/A-18 AIRCRAFT =====
C=====
C=====
C-----
C COEFFICIENT MODEL SUBROUTINE

```

```

C
C   THIS SUBROUTINE USES THE B-SPLINE COEFFICIENTS GENERATED
C   BY SUBROUTINE 'BCSMOD' IN THE IMSL FUNCTIONS BS3VL AND
C   BS3DR (STORED IN COMMON BLOCK 'SPLINE') TO DETERMINE THE
C   COEFFICIENTS CLO ,CDO ,CMO ,CLQ ,CMQ ,CLAD ,CMAD ,CLA ,
C   CDA ,CMA ,CLDH ,CDDH ,CMDH ,CL ,CD ,AND CM.
C
C   SUBROUTINE BCSMOD MUST BE CALLED PRIOR TO THIS SUBROUTINE
C
C   ACCEPTABLE RANGES ARE:
C       -10. <= ALPDEG <= 90.
C       -24. <= DHT <= 10.5
C       0.05 <= MACH <= 0.90
C
C   INPUTS - ALPDEG   ANGLE OF ATTACK      (deg)
C            DHT      STABILATOR DEFLECTION (deg)
C            MACH     MACH NUMBER
C            Q        PITCH RATE           (rad/s)
C            KDER     DERIVATIVE CONDITION
C                   KDER = 0 => CALCULATE alpha AND dht
C                               DERIVATIVES
C                   KDER = 1 => DO NOT CALCULATE alpha
C                               AND dht DERIVATIVES
C                   (SAVES TIME IF THESE DERIVATIVES ARE NOT
C                     DESIRED)
C
C   OUTPUTS - CL      LIFT COEFFICIENT
C            CD       DRAG COEFFICIENT
C            CM       PITCHING MOMENT COEFFICIENT
C            CLQ      d(CL)/d(q)           (s/rad)
C            CMQ      d(CM)/d(q)           (s/rad)
C            CLO      LIFT COEFFICIENT FOR q=0, d(alpha)/dt=0
C            CDO      DRAG COEFFICIENT FOR q=0, d(alpha)/dt=0
C            CMO      PITCHING MOMENT COEFFICIENT FOR
C                   q=0, d(alpha)/dt=0
C            CLAD     d(CL)/d( d(alpha)/dt ) (s/rad)
C            CMAD     d(CM)/d( d(alpha)/dt ) (s/rad)
C            CLA      d(CL)/d( alpha )     (/deg)
C            CDA      d(CD)/d( alpha )     (/deg)
C            CMA      d(CM)/d( alpha )     (/deg)
C            CLDH     d(CL)/d( dht )       (/deg)
C            CDDH     d(CD)/d( dht )       (/deg)
C            CMDH     d(CM)/d( dht )       (/deg)
C            CLOA     d(CLO)/d( alpha )    (/deg)
C            CMOA     d(CMO)/d( alpha )    (/deg)
C            CLODH    d(CLO)/d( dht )     (/deg)
C            CMODH    d(CMO)/d( dht )     (/deg)
C            2)      CLQA     d(CLQ)/d( alpha ) (/rad)
C            2)      CMQA     d(CMQ)/d( alpha ) (/rad)
C

```

C AUTHOR - FREDERICK E. GARRETT, JR.
 C GRADUATE RESEARCH ASSISTANT
 C AEROSPACE ENGINEERING
 C VPI & SU, BLACKSBURG, VA.

C DATE - DECEMBER 1987
 C

```

-----
SUBROUTINE COEFF(CL ,CD ,CM ,CLQ ,CMQ ,
5 CLO ,CDO ,CMO ,CLAD ,CMAD ,
2 CLA ,CDA ,CMA ,CLDH ,CDDH ,
3 CMDH ,CLOA ,CMOA ,CLODH ,CMODH ,
4 CLQA ,CMQA ,
C =f(
5 ALPDEG ,DHT ,MACH ,Q )
EXTERNAL BS3VL
EXTERNAL BS2VL
EXTERNAL BS3DR
EXTERNAL BS2DR
COMMON/FCONT /
1 BETDEG ,ALT ,SPSD ,RHO ,P ,R ,
2 HGCL ,DAR ,DAL ,DRR ,DRL ,DSBK ,
3 DLG ,DEBUG ,QSE
COMMON/F18FC /
1 CBAR ,B ,S ,EIY ,ELZ ,ALZ ,
2 ALX ,MASS
COMMON/AMISC /
1 AKM ,TX ,QBAR ,KDER
COMMON/SPLINE /
1 CCL ,CCD ,CCM ,CCLAD ,CCLQ ,CCMAD ,CCMQ ,
2 X ,XM ,XLQ ,XMQ ,Y ,YM ,Z ,
2 NX ,NXM ,NXLQ ,NXMQ ,NY ,NYM ,NZ ,
3 XKNOT ,XNOTM ,XNTLQ ,XNTMQ ,YKNOT ,YNOTM ,ZKNOT ,
4 KXORD ,KYORD ,KZORD ,KYORDM
REAL CCL ( 34 , 8 , 8 )
REAL CCD ( 34 , 8 , 8 )
REAL CCM ( 43 , 13 , 8 )
REAL CCLAD( 34 , 8 )
REAL CCLQ ( 30 , 8 )
REAL CCMAD( 43 , 8 )
REAL CCMQ ( 35 , 8 )
REAL X ( 34 )
REAL XM ( 43 )
REAL XLQ ( 30 )
REAL XMQ ( 35 )
REAL Y ( 8 )
REAL YM ( 13 )
REAL Z ( 8 )
REAL XKNOT( 37 )
REAL XNOTM( 46 )

```

```

REAL XNTLQ( 33 )
REAL XNTMQ( 38 )
REAL YKNOT( 11 )
REAL YNOTM( 16 )
REAL ZKNOT( 11 )
REAL MACH
DTR = ACOS(-1.) / 180.

```

```

C -----
C CALCULATE COEFFICIENTS FOR ALPDEG AND DHT
C (USING IMSL FUNCTION BS3VL)
C -----

```

```

IF(DHT .LT. -24. ) DHT = -24.
IF(DHT .GT. 10.5) DHT = 10.5
IF(ALPDEG .GT. 90. ) ALPDEG = 90.
IF(ALPDEG .LT. -10. ) ALPDEG = -10.
IF(MACH .LT. 0.05) MACH = 0.05
IF(MACH .GT. 0.90) MACH = 0.90
CLO = BS3VL(ALPDEG ,DHT ,MACH ,KXORD ,KYORD ,KZORD ,
1          XKNOT ,YKNOT ,ZKNOT ,NX ,NY ,NZ ,
2          CCL )
CDO = BS3VL(ALPDEG ,DHT ,MACH ,KXORD ,KYORD ,KZORD ,
1          XKNOT ,YKNOT ,ZKNOT ,NX ,NY ,NZ ,
2          CCD )
CMO = BS3VL(ALPDEG ,DHT ,MACH ,KXORD ,KYORDM ,KZORD ,
1          XNOTM ,YNOTM ,ZKNOT ,NXM ,NYM ,NZ ,
2          CCM )

```

```

C -----
C CALCULATE DERIVATIVE COEFFICIENTS
C (USING IMSL FUNCTION BS3VL)
C -----

```

```

CLAD = BS2VL(ALPDEG ,MACH ,KXORD ,KZORD ,
1          XKNOT ,ZKNOT ,NX ,NZ ,
2          CCLAD )
CMAD = BS2VL(ALPDEG ,MACH ,KXORD ,KZORD ,
1          XNOTM ,ZKNOT ,NXM ,NZ ,
2          CCMAD )
CLQ = BS2VL(ALPDEG ,MACH ,KXORD ,KZORD ,
1          XNTLQ ,ZKNOT ,NXLQ ,NZ ,
2          CCLQ )
CMQ = BS2VL(ALPDEG ,MACH ,KXORD ,KZORD ,
1          XNTMQ ,ZKNOT ,NXMQ ,NZ ,
2          CCMQ )

```

```

C -----
C CHECK DERIVATIVE CONDITION
C -----

```

```

IF(KDER .EQ. 1) THEN
  CLA = 0.
  CDA = 0.
  CMA = 0.
  CLDH = 0.

```

```

CDDH = 0.
CMDH = 0.
CLOA = 0.
CMOA = 0.
CLODH = 0.
CMODH = 0.
CLQA = 0.
CMQA = 0.
GOTO 10
END IF
C -----
C CALCULATE DERIVATIVE COEFFICIENTS
C (USING IMSL FUNCTIONS BS3DR)
C -----
CLOA = BS3DR(1 ,0 ,0 ,ALPDEG ,DHT ,MACH ,
1 KXORD ,KYORD ,KZORD ,XKNOT ,YKNOT ,ZKNOT ,
2 NX ,NY ,NZ ,CCL )
CDOA = BS3DR(1 ,0 ,0 ,ALPDEG ,DHT ,MACH ,
1 KXORD ,KYORD ,KZORD ,XKNOT ,YKNOT ,ZKNOT ,
2 NX ,NY ,NZ ,CCD )
CMOA = BS3DR(1 ,0 ,0 ,ALPDEG ,DHT ,MACH ,
1 KXORD ,KYORDM ,KZORD ,XNOTM ,YNOTM ,ZKNOT ,
2 NXM ,NYM ,NZ ,CCM )
CLODH = BS3DR(0 ,1 ,0 ,ALPDEG ,DHT ,MACH ,
1 KXORD ,KYORD ,KZORD ,XKNOT ,YKNOT ,ZKNOT ,
2 NX ,NY ,NZ ,CCL )
CDODH = BS3DR(0 ,1 ,0 ,ALPDEG ,DHT ,MACH ,
1 KXORD ,KYORD ,KZORD ,XKNOT ,YKNOT ,ZKNOT ,
2 NX ,NY ,NZ ,CCD )
CMODH = BS3DR(0 ,1 ,0 ,ALPDEG ,DHT ,MACH ,
1 KXORD ,KYORDM ,KZORD ,XNOTM ,YNOTM ,ZKNOT ,
2 NXM ,NYM ,NZ ,CCM )
CLQA = BS2DR(1 ,0 ,ALPDEG ,MACH ,
1 KXORD ,KZORD ,XNTLQ ,ZKNOT ,
2 NXLQ ,NZ ,CCLQ )
CMQA = BS2DR(1 ,0 ,ALPDEG ,MACH ,
1 KXORD ,KZORD ,XNTMQ ,ZKNOT ,
2 NXMQ ,NZ ,CCMQ )
C -----
C CALCULATE TOTAL COEFFICIENT DERIVATIVES
C -----
CLA = CLOA + CLQA * Q
CDA = CDOA
CMA = CMOA + CMQA * Q
CLDH = CLODH
CDDH = CDODH
CMDH = CMODH
C -----
C PUT DERIVATIVES IN CORRECT UNITS
C -----

```

```

CLQA = CLQA / DTR
CMQA = CMQA / DTR
C -----
C CALCULATE TOTAL COEFFICIENTS
C -----
10 CL = CLO + CLQ * Q
CD = CDO
CM = CMO + CMQ * Q
RETURN
END

C -----
C B-SPLINE MODEL SUBROUTINE <FEG>
C
C THIS SUBROUTINE USES IMSL SUBROUTINES BS3IN AND BSNK TO
C CALCULATE B-SPLINE COEFFICIENTS (STORED IN COMMON BLOCK
C SPLINE) FOR SUBROUTINE COEFF.
C
C NASA SUBROUTINE IAERO MUST BE CALLED PRIOR TO THIS SUBROUTINE
C
C OUTPUTS - COMMON BLOCK SPLINE
C
C AUTHOR - FREDERICK E. GARRETT, JR.
C GRADUATE RESEARCH ASSISTANT
C AEROSPACE ENGINEERING DEPT.
C VPI & SU, BLACKSBURG, VA;
C
C DATE - DECEMBER 1987
C -----
SUBROUTINE BCSMOD
EXTERNAL BS3IN
EXTERNAL BS2IN
EXTERNAL BSNK
COMMON/FCONT /
1 BETDEG ,ALT ,SPSD ,RHO ,P ,R ,
2 HGCL ,DAR ,DAL ,DRR ,DRL ,DSBK ,
3 DLG ,DEBUG ,QSE
COMMON/F18FC /
1 CBAR ,B ,S ,EIY ,ELZ ,ALZ ,
2 ALX ,MASS
COMMON/SPLINE /
1 CCL ,CCD ,CCM ,CCLAD ,CCLQ ,CCMAD ,CCMQ ,
2 X ,XM ,XLQ ,XMQ ,Y ,YM ,Z ,
2 NX ,NXM ,NXLQ ,NXMQ ,NY ,NYM ,NZ ,
3 XKNOT ,XNOTM ,XNTLQ ,XNTMQ ,YKNOT ,YNOTM ,ZKNOT ,
4 KXORD ,KYORD ,KZORD ,KYORDM
REAL CCL ( 34 , 8 , 8 )
REAL CCD ( 34 , 8 , 8 )
REAL CCM ( 43 , 13 , 8 )
REAL FCL ( 34 , 8 , 8 )

```

```

REAL FCD ( 34 , 8 , 8 )
REAL FCM ( 43 , 13 , 8 )
REAL CCLAD( 34 , 8 )
REAL CCLQ ( 30 , 8 )
REAL CCMAD( 43 , 8 )
REAL CCMQ ( 35 , 8 )
REAL FCLA ( 34 , 8 )
REAL FCLQ ( 30 , 8 )
REAL FCMA ( 43 , 8 )
REAL FCMQ ( 35 , 8 )
REAL X ( 34 )
REAL XM ( 43 )
REAL XLQ ( 30 )
REAL XMQ ( 35 )
REAL Y ( 8 )
REAL YM ( 13 )
REAL Z ( 8 )
REAL XKNOT( 37 )
REAL XNOTM( 46 )
REAL XNTLQ( 33 )
REAL XNTMQ( 38 )
REAL YKNOT( 11 )
REAL YNOTM( 16 )
REAL ZKNOT( 11 )
REAL MASS
LOGICAL DEBUG
LOGICAL QSE
DATA X/ -10., -8., -6., -4., -2., 0., 2., 4.,
1 6., 8., 10., 12., 14., 16., 18., 20.,
2 22., 24., 26., 28., 30., 32., 34., 36.,
3 38., 50., 55., 60., 65., 70., 75., 80.,
4 85., 90./,
5 XM/ -10., -8., -6., -4., -2., 0., 2., 4.,
6 6., 8., 10., 12., 14., 16., 18., 20.,
7 22., 24., 26., 28., 30., 32., 34., 35.,
8 36., 37., 38., 39., 40., 41., 42., 43.,
9 44., 45., 50., 55., 60., 65., 70., 75.,
& 80., 85., 90./,
1 XLQ/ -10., -5., 2., 4., 6., 8., 11., 13.,
2 16., 18., 19., 21., 22., 24., 28., 32.,
3 36., 38., 44., 46., 48., 50., 55., 60.,
4 65., 70., 75., 80., 85., 90./,
5 XMQ/ -10., -5., 2., 4., 6., 8., 12., 14.,
6 16., 18., 20., 22., 24., 26., 28., 30.,
7 32., 34., 36., 38., 39., 41., 42., 45.,
8 47., 48., 49., 51., 55., 60., 64., 66.,
9 70., 80., 90./,
& Y/ -24., -12., -6., -3., 0., 3., 6., 10.5/,
1 YM/ -24., -12., -11., -10., -9., -8., -7., -6.,
2 -3., 0., 3., 6., 10.5/,

```

```

3      Z/ 0.05, 0.10, 0.20, 0.40, 0.60, 0.80, 0.85, 0.90/,
4      NX/ 34 /,
5      NXM/ 43 /,
6      NXLQ/ 30 /,
7      NXMQ/ 35 /,
8      NY/ 8 /,
9      NYM/ 13 /,
&      NZ/ 8 /,
1      Q/ 0. /,
2      KXORD/ 3 /, KYORD/ 3 /, KZORD/ 3 /, KYORDM/ 3 /

```

C
C
C
C

```

-----
SET UP INTERPOLATION POINTS
(USING IMSL SUBROUTINE BSNAK)
-----

```

```

CALL BSNAK (NX ,X ,KXORD ,XKNOT )
CALL BSNAK (NXM ,XM ,KXORD ,XNOTM )
CALL BSNAK (NXLQ ,XLQ ,KXORD ,XNTLQ )
CALL BSNAK (NXMQ ,XMQ ,KXORD ,XNTMQ )
CALL BSNAK (NY ,Y ,KYORD ,YKNOT )
CALL BSNAK (NYM ,YM ,KYORDM ,YNOTM )
CALL BSNAK (NZ ,Z ,KZORD ,ZKNOT )

```

C
C
C

```

-----
GET COEFFICIENT ARRAYS
-----

```

```

DO 70 K = 1,NZ
  VEL = Z(K) * SPSD
  QBAR = 0.5 * RHO * VEL*VEL
  CO2VT = CBAR / (2 * VEL)
  BO2VT = B / (2 * VEL)
  DO 20 I = 1,NX
    CALL FLPSCH(Z(K) ,ALT ,X(I) ,DLFR ,DTFR )
    DLFL = DLFR
    DTFL = DTFR
    DO 10 J = 1,NY
      CALL REFAERO(FCD(I,J,K) ,CY ,FCL(I,J,K) ,C1 ,
1          CM ,CN ,CLAD ,CMAD ,
2          CLQ ,DCLQF ,CMQ ,DCMQF ,
3          X(I) ,BETDEG ,Z(K) ,P ,Q ,
4          R ,ALT ,HGCL ,QBAR ,CO2VT ,
5          BO2VT ,Y(J) ,Y(J) ,DAR ,DAL ,
6          DRR ,DRL ,DTFR ,DTFL ,DLFR ,
7          DLFL ,DSBK ,DLG ,DEBUG ,QSE )
      IF(Z(K) .GE. 0.6 .AND. X(I) .LT. 40.0) THEN
        CALL CORREC(CORL ,CORD ,CORM ,
1          Y(J) ,Z(K) ,QBAR ,CO2VT ,BO2VT )
        FCL(I,J,K) = FCL(I,J,K) + CORL
        FCD(I,J,K) = FCD(I,J,K) + CORD
      END IF
      IF(Y(J) .EQ. 0.) FCLA(I,K) = CO2VT * CLAD
10    CONTINUE

```

```

20  CONTINUE
    DO 40 I = 1,NXM
        CALL FLPSCH(Z(K) ,ALT ,XM(I) ,DLFR ,DTFR )
        DLFL = DLFR
        DTFI = DTFR
        DO 30 J = 1,NYM
            CALL REFAERO(CD ,CY ,CL ,C1 ,
1             FCM(I,J,K) ,CN,CLAD ,CMAD ,
2             CLQ ,DCLQF ,CMQ ,DCMQF ,
3             XM(I) ,BETDEG ,Z(K) ,P ,Q ,
4             R ,ALT ,HGCL ,QBAR ,CO2VT ,
5             BO2VT ,YM(J) ,YM(J) ,DAR ,DAL ,
6             DRR ,DRL ,DTFR ,DTFI ,DLFR ,
7             DLFL ,DSBK ,DLG ,DEBUG ,QSE )
            IF(Z(K) .GE. 0.6 .AND. XM(I) .LT. 40.0) THEN
                CALL CORREC(CORL ,CORD ,CORM ,
1                 YM(J) ,Z(K) ,QBAR ,CO2VT ,BO2VT )
                FCM(I,J,K) = FCM(I,J,K) + CORM
            END IF
            IF(YM(J) .EQ. 0.) FCMA(I,K) = CO2VT * CMAD
30  CONTINUE
40  CONTINUE
    DO 50 I = 1,NXLQ
        CALL FLPSCH(Z(K) ,ALT ,XLQ(I) ,DLFR ,DTFR )
        DLFL = DLFR
        DTFI = DTFR
        CALL REFAERO(CD ,CY ,CL ,C1 ,
1             CM ,CN ,CLAD ,CMAD ,
2             CLQ ,DCLQF ,CMQ ,DCMQF ,
3             XLQ(I) ,BETDEG ,Z(K) ,P ,Q ,
4             R ,ALT ,HGCL ,QBAR ,CO2VT ,
5             BO2VT ,0. ,0. ,DAR ,DAL ,
6             DRR ,DRL ,DTFR ,DTFI ,DLFR ,
7             DLFL ,DSBK ,DLG ,DEBUG ,QSE )
        FCLQ(I,K) = CO2VT * (CLQ + DCLQF)
50  CONTINUE
    DO 60 I = 1,NXMQ
        CALL FLPSCH(Z(K) ,ALT ,XMQ(I) ,DLFR ,DTFR )
        DLFL = DLFR
        DTFI = DTFR
        CALL REFAERO(CD ,CY ,CL ,C1 ,
1             CM ,CN ,CLAD ,CMAD ,
2             CLQ ,DCLQF ,CMQ ,DCMQF ,
3             XMQ(I) ,BETDEG ,Z(K) ,P ,Q ,
4             R ,ALT ,HGCL ,QBAR ,CO2VT ,
5             BO2VT ,0. ,0. ,DAR ,DAL ,
6             DRR ,DRL ,DTFR ,DTFI ,DLFR ,
7             DLFL ,DSBK ,DLG ,DEBUG ,QSE )
        FCMQ(I,K) = CO2VT * (CMQ + DCMQF)
60  CONTINUE

```

70 CONTINUE

```
C -----
C   CALCULATE B-SPLINE COEFFICIENTS
C   (USING IMSL SUBROUTINE BS3IN)
C -----
CALL BS3IN (NX      ,X      ,NY      ,Y      ,NZ      ,Z      ,
1          FCL     ,NX     ,NY     ,KXORD  ,KYORD  ,KZORD  ,
2          XKNOT   ,YKNOT  ,ZKNOT  ,CCL   )
CALL BS3IN (NX      ,X      ,NY      ,Y      ,NZ      ,Z      ,
1          FCD     ,NX     ,NY     ,KXORD  ,KYORD  ,KZORD  ,
2          XKNOT   ,YKNOT  ,ZKNOT  ,CCD   )
CALL BS3IN (NXM     ,XM     ,NYM    ,YM     ,NZ      ,Z      ,
1          FCM     ,NXM    ,NYM    ,KXORD  ,KYORDM ,KZORD  ,
2          XNOTM   ,YNOTM  ,ZKNOT  ,CCM   )
CALL BS2IN (NX      ,X      ,NZ      ,Z      ,
1          FCLA    ,NX     ,KXORD  ,KZORD  ,
2          XKNOT   ,ZKNOT  ,CCLAD  )
CALL BS2IN (NXM     ,XM     ,NZ      ,Z      ,
1          FCMA    ,NXM    ,KXORD  ,KZORD  ,
2          XNOTM   ,ZKNOT  ,CCMAD  )
CALL BS2IN (NXLQ    ,XLQ    ,NZ      ,Z      ,
1          FCLQ    ,NXLQ   ,KXORD  ,KZORD  ,
2          XNTLQ   ,ZKNOT  ,CCLQ   )
CALL BS2IN (NXMQ    ,XMQ    ,NZ      ,Z      ,
1          FCMQ    ,NXMQ   ,KXORD  ,KZORD  ,
2          XNTMQ   ,ZKNOT  ,CCMQ   )
RETURN
END
```

```
C -----
C   MODEL CORRECTION SUBROUTINE
C
C   THIS SUBROUTINE CORRECTS THE NASA MODEL
C   FOR HIGH ANGLES OF ATTACK
C
C   INPUTS - DH      STABILATOR ANGLE (deg)
C           MACH     MACH NUMBER
C           QBAR     DYNAMIC PRESSURE (lb/ft**2)
C           CO2VT    CBAR / (2 * V)
C           BO2VT    B    / (2 * V)
C
C   OUTPUTS - CORL   CORRECTION IN CL
C              FOR ANGLES OF ATTACK > 40 deg
C           CORD     CORRECTION IN CD
C              FOR ANGLES OF ATTACK > 40 deg
C           CORM     CORRECTION IN CM
C              FOR ANGLES OF ATTACK > 40 deg
C
C   AUTHOR - FREDERICK E. GARRETT, JR.
C           GRADUATE RESEARCH ASSISTANT
C           AEROSPACE ENGINEERING
```

```

C          VPI & SU, BLACKSBURG, VA
C
C      DATE   -   MARCH 1988
C
C-----
C      SUBROUTINE CORREC(CORL ,CORD ,CORM ,
C          =f(
1          DH , MACH , QBAR , CO2VT , BO2VT)
COMMON/FCONT /
1      BETDEG ,ALT ,SPSD ,RHO ,P ,R ,
2      HGCL ,DAR ,DAL ,DRR ,DRL ,DSBK ,
3      DLG ,DEBUG ,QSE
LOGICAL DEBUG
LOGICAL QSE
REAL MACH
Q = 0.
C-----
C      CALCULATE CL AT ALPDEG = 39.99999 deg
C-----
ALPDEG = 39.99999
CALL FLPSCH(MACH ,ALT ,ALPDEG,DLFR ,DTFR )
DLFL = DLFR
DTFL = DTFR
CALL REFAERO(CD39 ,CY ,CL39 ,C1 ,
1      CM39 ,CN ,CLAD ,CMAD ,
2      CLQ ,DCLQF ,CMQ ,DCMQF ,
3      ALPDEG,BETDEG ,MACH ,P ,Q ,
4      R ,ALT ,HGCL ,QBAR ,CO2VT ,
5      BO2VT ,DH ,DH ,DAR ,DAL ,
6      DRR ,DRL ,DTFR ,DTFL ,DLFR ,
7      DLFL ,DSBK ,DLG ,DEBUG ,QSE )
C-----
C      CALCULATE CL AT ALPDEG = 40.0000 deg
C-----
ALPDEG = 40.0
CALL FLPSCH(MACH ,ALT ,ALPDEG,DLFR ,DTFR )
DLFL = DLFR
DTFL = DTFR
CALL REFAERO(CD40 ,CY ,CL40 ,C1 ,
1      CM40 ,CN ,CLAD ,CMAD ,
2      CLQ ,DCLQF ,CMQ ,DCMQF ,
3      ALPDEG,BETDEG ,MACH ,P ,Q ,
4      R ,ALT ,HGCL ,QBAR ,CO2VT ,
5      BO2VT ,DH ,DH ,DAR ,DAL ,
6      DRR ,DRL ,DTFR ,DTFL ,DLFR ,
7      DLFL ,DSBK ,DLG ,DEBUG ,QSE )
C-----
C      CALCULATE CORRECTION
C-----
CORL = CL40 - CL39

```

```

CORD = CD40 - CD39
CORM = CM40 - CM39
RETURN
END

C-----
C   ATMOSPHERIC MODEL
C-----
C   SUBROUTINE ATMOS(RHO ,SPSD ,ALT )
C-----
C   SPEED OF SOUND
C-----
H   = ALT / 1000.
H   = H / 3.28084
TEMP = 3.72E-3 * H + 0.193315
TEMP = TEMP * H - 8.87743
TEMP = TEMP * H + 292.1
SPSD = 20.0468 * SQRT(TEMP)
SPSD = SPSD / 0.3048

C-----
C   DENSITY
C-----
ARG = 1.15219733E-6 * H - 8.3300053E-5
ARG = ARG * H + 3.50991865D-3
ARG = ARG * H - 3.48643241E-2
ARG = ARG * H
PHI = 1.0228055 * EXP(-ARG)
RHO = PHI - 1.0228055 - 0.12122693 * H
RHO = EXP(RHO)
RHO = RHO * 1.225
RHO = RHO * 0.02832
RHO = RHO * 0.06852
RETURN
END

C+++++
C+++
C+++   F18_OUTER_LAYER.FOR   F/A-18 OUTER LAYER PROGRAM   +++
C+++
C+++   PURPOSE   -   TO CALCULATE THE OUTER LAYER   +++
C+++                   SOLUTION USING PARAMETER   +++
C+++                   OPTIMIZATION   +++
C+++
C+++
C+++   REQUIRED   -   F18_SUB.FOR   +++
C+++                   F18_AERO_TABLES.DAT   +++
C+++                   F18_ENG_TABLES.DAT   +++
C+++                   IMSL MATH/LIBRARY   +++
C+++                   SLLSQ   +++
C+++                   DBLAS   +++
C+++
C+++   AUTHOR   -   FREDERICK E. GARRETT, JR.   +++

```

```

C+++          GRADUATE RESEARCH ASSISTANT          +++
C+++          AEROSPACE ENGINEERING DEPT.         +++
C+++          VPI & SU, BLACKSBURG, VA            +++
C+++          DATE          -   MARCH 1988         +++
C+++

```

```

C+++++-----+

```

```

COMMON/FCONT /
1      BETDEG ,ALT      ,SPSD      ,RHO      ,P      ,R      ,
2      HGCL   ,DAR      ,DAL      ,DRR      ,DRL      ,DSBK      ,
3      DLG    ,DEBUG    ,QSE
COMMON/CONST /
1      G      ,DUMMY    ,DTR
COMMON/F18FC /
1      CBAR   ,B        ,S        ,EIY      ,ELZ      ,ALZ      ,
2      ALX    ,MASS
COMMON/AMISC /
1      AKM    ,TX       ,QBAR    ,KDER
COMMON/WORKSP/
1      RWKSP
REAL*8 DG( 1 , 2 )
REAL*8 GG( 1 )
REAL*8 X ( 2 )
REAL*8 XL( 2 )
REAL*8 XU( 2 )
REAL*8 DF( 2 )
REAL*8 W ( 150 )
REAL*8 ACC
REAL*8 ALPHA
REAL*8 DHT
REAL*8 F
REAL*8 AMV
REAL*8 GV
REAL*8 QBSM
REAL*8 QBI
REAL*8 TXI
REAL*8 TMV
REAL*8 QT
REAL*8 QB
REAL*8 QQ
REAL*8 DL
REAL*8 DD
REAL*8 QTA
REAL*8 QBA
REAL*8 QA
REAL*8 QDH
REAL*8 GAMMA
REAL*8 DMASS
REAL*8 FCOST
REAL*8 GCOST

```

```

REAL*8 DBG
REAL*8 DDTR
REAL*8 DCBAR
REAL*8 DB
REAL*8 DS
REAL*8 DEIY
REAL*8 DELZ
REAL*8 DALZ
REAL*8 DALX
REAL*8 DTX
REAL*8 DQBAR
REAL*8 DV
REAL*8 DAKM
REAL MACH
REAL MASS
REAL RWKSP ( 5137 )
INTEGER INDEX ( 10 )
LOGICAL DEBUG
LOGICAL QSE
CHARACTER*80 HEADER
CHARACTER*10 MANVER
CALL IWKIN ( 5137 )

```

```

C-----
C  INITIALIZATION
C-----

```

```

C-----
C  GENERAL CONSTANTS
C-----

```

```

G      = 32.174
DUMMY  = 1.
DTR    = ACOS(-1.) / 180.
DBG    = DBLE ( G )
DDTR   = DBLE ( DTR )

```

```

C-----
C  FLIGHT CONDITIONS
C-----

```

```

BETDEG = 0.
P       = 0.
R       = 0.

```

```

C-----
C  CHOOSE MANEUVER
C      AKM = 1. => HALF-LOOP
C      AKM = -1. => SPLIT-S
C-----

```

```

AKM    = - 1.
DAKM   = DBLE ( AKM )
IF( AKM .EQ. 1. ) THEN
  MANVER = 'HALF-LOOP'
ELSE
  MANVER = 'SPLIT-S'

```

```

      END IF
C -----
C F/A-18 CONSTANTS FIGHTER ESCORT CONFIGURATION
C -----
      CBAR   = 11.52
      B      = 37.42
      S      = 400.0
      EIY    = 151293.
      ELZ    = 0.23
      ALZ    = 0.23
      ALX    = -0.297
      MASS   = 1034.47
      DCBAR  = DBLE ( CBAR )
      DB     = DBLE ( B   )
      DS     = DBLE ( S   )
      DEIY   = DBLE ( EIY )
      DELZ   = DBLE ( ELZ )
      DALZ   = DBLE ( ALZ )
      DALX   = DBLE ( ALX )
      DMASS  = DBLE ( MASS )
C -----
C IGNORABLE CONTROL SETTINGS
C -----
      DAR    = 0.
      DAL    = 0.
      DRR    = 0.
      DRL    = 0.
      DSBK   = 0.
      DLG    = 0.
C -----
C LOGICAL DECLARATIONS FOR SUBROUTINE REFAERO
C -----
      DEBUG  = .FALSE.
      QSE    = .TRUE.
      CALL IAERO (HEADER)
C -----
C ENGINE PARAMETERS
C -----
      WDTF   = DUMMY
      GK1    = DUMMY
      P1     = DUMMY
      P2     = DUMMY
      TAU    = DUMMY
      TESTV  = DUMMY
      VAE    = DUMMY
      CALL IENG (HEADER)
C -----
C CONSTANT STATES AND CONTROLS
C -----
      DTH    = 130.0

```

```

      KDER = 0
C -----
C   INITIALIZE SLLSQ
C -----
      ACC = 1.D-05
      MAXIT = 1 E+09
      N = 2
      M = 1
      ME = 1
      LDG = 1
      LW = 150
      KW = 10
      ALOW = 30.0
      DLOW = -24.0
      AUP = 50.0
      DUP = 10.5
      XL( 1 )= DBLE ( ALOW * DTR )
      XL( 2 )= DBLE ( DLOW * DTR )
      XU( 1 )= DBLE ( AUP * DTR )
      XU( 2 )= DBLE ( DUP * DTR )
C -----
C   ALTITUDE LOOP
C -----
      DO 200 IH = 1,6
      ALT = (FLOAT(IH - 1) * 10. + 5.) * 1000.
      CALL ATMOS(RHO ,SPSD ,ALT )
      HGCL = ALT
C -----
C   CALCULATE B-SPLINE COEFFICIENTS
C   USING SUBROUTINE BCSSMOD
C -----
      CALL BCSSMOD
C -----
C   MACH NUMBER LOOP
C -----
      DO 140 K = 3,9
      MACH = FLOAT(K) / 10.
      V = MACH * SPSD
      DV = DBLE ( V )
      QBAR = 0.5 * RHO * V*V
      DQBAR = DBLE ( QBAR )
      WRITE(10,20) ALT,MACH
20  FORMAT(' ALTITUDE = ',F14.2,' ft  MACH = ',F10.2)
C -----
C   CALCULATE THRUST
C -----
      CALL ENG1 (TH ,WDTF ,GK1 ,P1 ,P2 ,
1          TAU ,TESTV ,TMIL ,TAUGMN ,DTH ,
2          MACH ,ALT ,VAE )
      TX = 2.0 * TH * COS(1.98 * DTR)

```

```

DTX = DBLE ( TX )
AMV = DMASS * DV
GV = DBG / DV
QBSM = DQBAR * DS / AMV
QBI = DQBAR * DS / DEIY
TXI = DELZ * DTX / DEIY
TMV = DTX / AMV
C -----
C FLIGHT PATH ANGLE LOOP
C -----
DO 130 J = 1,19
GAMDEG = AKM * FLOAT(J-1) * 10.
GAMMA = DBLE(GAMDEG) * DDTR
KZERO = 0
FCOST = 1.DO
GCOST = 1.DO
ALPHA = 36.DO
X( 1 )= ALPHA * DDTR
DHT = -13.DO
X( 2 )= DHT * DDTR
C -----
C ITERATION LOOP
C -----
DO 110 I = 1,2
MODE = 0
DO 100 II = 1,250
C -----
C CALCULATE AERODYNAMIC COEFFICIENTS
C -----
SALPHA = SNGL ( ALPHA )
SDHT = SNGL ( DHT )
CALL COEFF(SCL ,SCD ,SCM ,SCLQ ,SCMQ ,
1 SCL0 ,SCD0 ,SCM0 ,SCLAD ,SCMAD ,
2 SCLA ,SCDA ,SCMA ,SCLDH ,SCDDH ,
3 SCMDH ,SCLOA ,SCMOA ,SCLODH ,SCMODH ,
4 SCLQA ,SCMQA ,
5 SALPHA ,SDHT ,MACH ,0. )
C -----
C CONVERT COEFFICIENTS TO DOUBLE PRECISION
C -----
CL = DBLE ( SCL )
CD = DBLE ( SCD )
CM = DBLE ( SCM )
CLQ = DBLE ( SCLQ )
CMQ = DBLE ( SCMQ )
CLO = DBLE ( SCL0 )
CDO = DBLE ( SCDO )
CMO = DBLE ( SCMO )
CLAD = DBLE ( SCLAD )
CMAD = DBLE ( SCMAD )

```

```

CLA  = DBLE ( SCLA  )
CDA  = DBLE ( SCDA  )
CMA  = DBLE ( SCMA  )
CLDH = DBLE ( SCLDH )
CDDH = DBLE ( SCDDH )
CMDH = DBLE ( SCMDH )
CLOA = DBLE ( SCLOA )
CMOA = DBLE ( SCMOA )
CLODH = DBLE ( SCLODH )
CMODH = DBLE ( SCMODH )
CLQA = DBLE ( SCLQA )
CMQA = DBLE ( SCMQA )

```

C
C
C

```

-----
CONVERT DERIVATIVES TO RADIANS
-----

```

```

CLA  = CLA  / DDTR
CDA  = CDA  / DDTR
CMA  = CMA  / DDTR
CLDH = CLDH / DDTR
CDDH = CDDH / DDTR
CMDH = CMDH / DDTR
CLOA = CLOA / DDTR
CMOA = CMOA / DDTR
CLODH = CLODH / DDTR
CMODH = CMODH / DDTR
DL    = DALX * DCOS(X(1)) + DALZ * DSIN(X(1))
DD    = DALX * DSIN(X(1)) - DALZ * DCOS(X(1))
IF(MODE .EQ. -1) GOTO 40

```

C
C
C

```

-----
OBJECTIVE FUNCTION d(alpha)/dt = 0
-----

```

```

QT   = - GV   * DCOS(GAMMA) * DAKM
1    + QBSM * CLO
2    + TMV  * DSIN(X(1))
QB   = 1     - QBSM * CLQ
QQ   = QT / QB
F    = - FCOST * QQ

```

C
C
C

```

-----
EQUALITY CONSTRAINT d(q)/dt = 0
-----

```

```

GG(1) = TXI
1      + QBI * (DCBAR * CMO + DL * CLO + DD * CDO
2      + (DCBAR * CMQ + DL * CLQ) * QQ)
GG(1) = GCOST * GG(1)
IF(MODE .EQ. 1) GOTO 50

```

C
C
C

```

-----
DERIVATIVES OF OBJECTIVE FUNCTION
-----

```

```

40    QTA  =  QBSM * CLOA
1     + TMV * DCOS(X(1))

```

```

QBA = - QBSM * CLQA
QA = (QTA * QB - QT * QBA) / (QB * QB)
DF(1) = - FCOST * QA
QDH = QBSM * CLODH / QB
DF(2) = - FCOST * QDH

```

```

C -----
C NORMALS TO THE CONSTRAINT
C -----
      DG(1,1) = QBI * (DCBAR * CMOA + DL * (CLOA + CDO)
1          + DD * (CDA - CLO)
2          + (DCBAR * CMQA + DL * CLQ - DD * CLQA)
3          * QQ
4          + (DCBAR * CMQ + DL * CLQ) * QA)
      DG(1,1) = GCOST * DG(1,1)
      DG(1,2) = QBI * (DCBAR * CMODH + DL * CLODH + DD * CDDH
1          + (DCBAR * CMQ + DL * CLQ) * QDH )
      DG(1,2) = GCOST * DG(1,2)
C -----
C CALL SLLSQP
C -----
50      IF(KZERO .EQ. 1) GOTO 120
      CALL SLLSQP(M ,ME ,LDG ,N ,X ,XL ,XU ,
1          F ,GG ,DF ,DG ,ACC ,MAXIT ,MODE ,
2          W ,LW ,INDEX)
      ALPHA = X(1) / DDTR
      DHT = X(2) / DDTR
      IF(MODE .EQ. 0) KZERO = 1
100     CONTINUE
      FCOST = FCOST * 1.D1
      GCOST = GCOST * 1.D-1
110     CONTINUE
      WRITE(10,116)
116     FORMAT(' SLLSQP DID NOT CONVERGE')
120     AMAX = SNGL ( ALPHA )
      DHMAX = SNGL ( DHT )
      QMAX = SNGL ( QQ )
      QDMAX = QMAX / DTR
      WRITE(10,125) AMAX,QDMAX,DHMAX
125     FORMAT(3F14.7)
130     CONTINUE
140     CONTINUE
200     CONTINUE
      STOP
      END

```

Vita

The author was born on February 5, 1961 in South Boston, VA. He graduated from Prince Edward Academy in Farmville, VA. in June, 1979. He received a B.S. in Aerospace and Ocean Engineering in 1986 from Virginia Polytechnic Institute and State University, Blacksburg, Va. and began graduate study there in August, 1986.

A handwritten signature in black ink, appearing to read 'F. M. E. S.', with a long, sweeping underline that extends to the right.