

**AEDC-TR-94-15**



# Computational Time-Accurate Body Movement: Methodology, Validation, and Application

J. K. Jordan, N. E. Suhs, R. D. Thoms,  
R. W. Tramel, J. H. Fox, and J. C. Erickson, Jr.  
Micro Craft Technology/AEDC Operations

October 1995

Final Report for Period October 1993 — September 1994

Approved for public release; distribution is unlimited.

**ARNOLD ENGINEERING DEVELOPMENT CENTER  
ARNOLD AIR FORCE BASE, TENNESSEE  
AIR FORCE MATERIEL COMMAND  
UNITED STATES AIR FORCE**



## NOTICES

When U. S. Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise, or in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

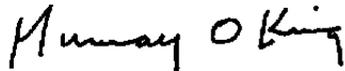
Qualified users may obtain copies of this report from the Defense Technical Information Center.

References to named commercial products in this report are not to be considered in any sense as an endorsement of the product by the United States Air Force or the Government.

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

## APPROVAL STATEMENT

This report has been reviewed and approved.



MURRAY O. KING  
Aircraft Systems Test Division  
Test Operations Directorate

Approved for publication:

FOR THE COMMANDER



EUGENE J. SANDERS  
Tech. Director, Aircraft Systems Test Division  
Test Operations Directorate

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302 and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503				
1 AGENCY USE ONLY (Leave blank)	2 REPORT DATE October 1995	3 REPORT TYPE AND DATES COVERED Final Report for Period October 1993 - September 1994		
4 TITLE AND SUBTITLE Computational Time-Accurate Body Movement: Methodology, Validation, and Application			5 FUNDING NUMBERS PE 27590F AF PN 0724	
6 AUTHOR(S) J. K. Jordan, N. E. Suhs, R. D. Thoms, R. W. Tramel, J. H. Fox, and J. C. Erickson, Jr., Micro Craft Technology/AEDC Operations				
7 PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Arnold Engineering Development Center/DOF Air Force Materiel Command Arnold Air Force Base, TN 37389-6000			8 PERFORMING ORGANIZATION (REPORT NUMBER) AEDC-TR-94-15	
9 SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force SEEK EAGLE Office (AFSEO/SKM) Eglin Air Force Base, FL 32542			10 SPONSORING/MONITORING AGENCY REPORT NUMBER	
11 SUPPLEMENTARY NOTES Available in Defense Technical Information Center (DTIC).				
12A DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12B DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) A computational fluid dynamics (CFD) procedure for computing time-accurate body motion has been developed. The procedure can be used for bodies using a predefined motion, or for body motion resulting from loads. The chimera moving overset-mesh methodology is used in either case. The procedure is validated by comparing predictions of a pitching airfoil and a generic store trajectory to measured data. The procedure is then applied to a multiple body launch from a simulated Triple Ejection Rack (TER) and used to estimate the effect of mutual interference on a trajectory.				
14 SUBJECT TERMS computational fluid dynamics, overset mesh, pressure distribution, Euler equations, generic store trajectory, aerodynamic forces, aerodynamic loads, mutual interference, time-accurate CFD, unsteady CFD			15 NUMBER OF PAGES 130	
			16 PRICE CODE	
17 SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18 SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19 SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20 LIMITATION OF ABSTRACT SAME AS REPORT	

## **PREFACE**

The work reported herein was performed at the Arnold Engineering Development Center (AEDC), Air Force Materiel Command (AFMC), under Program Element 27590F, Control Number 4037, at the request of the Air Force SEEK EAGLE Office (AFSEO/SKM), Eglin Air Force Base, Florida. The AFSEO Program Manager was Maj. Bob Cooper. The AEDC/DOFA Project Manager was Mr. Murray King, and the AEDC Project Engineer was Mr. Dave Carlson. The work was performed by Micro Craft Technology/AEDC Operations, support and technical service contractor of flight dynamics test facilities, Air Force Materiel Command, Arnold Engineering Development Center, Arnold AFB, Tennessee. The work was performed in the Technology and Development Facility (TDF) under AEDC Project Number 0724 during the period October 1, 1993 through September 30, 1994. The manuscript was submitted for publication on August 7, 1995.

## CONTENTS

1.0	INTRODUCTION .....	5
2.0	METHODOLOGY .....	6
2.1	Chimera Overset Grid Methodology .....	6
2.2	Overall Procedure .....	8
3.0	RESULTS AND DISCUSSION .....	9
3.1	Baseline Validation – Two-Dimensional Pitching Airfoil .....	10
3.2	Validation – Single Store Trajectory .....	12
3.3	Application – Multiple Body Mutual Interference .....	18
4.0	CONCLUSIONS .....	21
4.1	Pitching Airfoil .....	21
4.2	Single-Store Trajectory .....	21
4.3	Multiple-Body Trajectory .....	21
	REFERENCES .....	22

## ILLUSTRATIONS

<u>Figure</u>		<u>Page</u>
1.	Time-Accurate Store Trajectory Prediction Process .....	25
2.	Information Flow for the Time-Accurate Store Trajectory Prediction Process .....	26
3.	Pitch Rates for NACA-0012 Airfoil .....	27
4.	Comparisons of Computations and Measured Data for Pitching Airfoil, Cases A and B .....	28
5.	Time History of Pressure Coefficients on the Pitching Airfoil for Case B .....	29
6.	Comparisons of $C_N$ Results for Different DT's, Case B .....	45
7.	WL/AD Test Article for Single-Store Trajectory .....	46
8.	WL/AD Single-Store Configuration Geometry .....	47
9.	Comparisons of Computed Store Trajectory Translational Time-History with CTS Test Data .....	48
10.	Comparisons of Computed Store Trajectory Rotational Time-History with CTS Test Data .....	52
11.	Comparisons of Computed Store Trajectory Translational Time-History with CTS Test Data and WL/AD Computations .....	56
12.	Comparisons of Computed Store Trajectory Rotational Time-History with CTS Test Data and WL/AD Computations .....	59
13.	TER Solution Carriage Position .....	63
14.	Translational and Rotational Time History for TER Configuration, Bottom Store .....	63

<u>Figure</u>	<u>Page</u>
15. Translational and Rotational Time History for TER Configuration, Outboard Store . . . .	64
16. Translational and Rotational Time History for TER Configuration, Inboard Store . . . . .	65
17. Comparisons of Translational and Rotational Time History for the Single-Store Configuration (Inboard Store) and the TER Configuration (Inboard Store) . . . . .	66

**TABLES**

1. Physical Parameters for the Single Body Launch . . . . .	67
2. Physical Parameters for the Multiple Body Launch . . . . .	67
3. TER Configuration: Carriage Loads . . . . .	68
4. Inboard Store Carriage Loads – Single to TER Configuration Comparisons . . . . .	68

**APPENDICES**

A. Integration Procedure for Six-Degree-of-Freedom Equations of Motion . . . . .	69
B. Validation of SIXDOF Code . . . . .	76
C. Trajectory Sensitivity Study . . . . .	86
D. SIXDOF Code User’s Manual . . . . .	98
E. Ejector Models . . . . .	113
F. Relation of XAIR Time Increment and Real Time Increment . . . . .	117
G. Axis Systems . . . . .	118
 NOMENCLATURE . . . . .	 125

## 1.0 INTRODUCTION

Integrated Testing and Evaluation at the Arnold Engineering Development Center (AEDC) includes large-scale computations (Refs. 1–6) in support of users of the AEDC ground simulation facilities. One important new area in which the AEDC computational capability shows great promise, but must be demonstrated further, is the application of CFD to various aspects of bodies in motion. An important application of the capability is for the calculation of store separation trajectories with time-accurate CFD aerodynamic-load predictions. This computational capability complements Captive Trajectory Support system (CTS) testing (Refs. 7-8), dynamic drop testing (Refs. 7 and 9), and the application of engineering methods (Ref. 10). There are many potential CFD applications to store separation configurations that are very difficult or impossible to test with conventional ground simulation techniques. Examples include separation of multiple stores, e.g., ripple launches, launch from inside a weapons bay, and separation from maneuvering aircraft possibly at high angles of attack.

Several investigators (Refs. 11–13) have coupled moving-mesh flow solvers with six-degree-of-freedom (6DOF) solvers. To date, the most advanced of these investigations was that of Lijewski and Suhs (Ref. 12). This study was the first demonstration, by favorable comparisons with CTS test loads and trajectory data, of time-accurate CFD prediction of a body after it separated from the aircraft. The configuration investigated was the generic wing/pylon/finned store. A CTS test of this configuration was performed at the AEDC with Air Force Wright Laboratory/Armament Directorate (WL/AD) funding. This very extensive test included free-stream load measurements, simulated trajectories, pressure data, and flow visualization data. The success of the joint AEDC and WL/AD demonstration led the AEDC to develop the same time-accurate predictive capability using its own CFD codes (Refs. 1–6).

The validation of the procedures and codes for time-accurate body motion by comparison of predictions to measurements, and the application of the procedure to a multi-body launch from a simulated Triple Ejection Rack (TER) is reported in this document. The overall methodology that combines the chimera overset-mesh methodology, the flow solver, and the integration of the 6DOF equations of motion is outlined in Sec. 2.0. Results from the application of the procedures are discussed in Sec. 3.0, and the two-dimensional airfoil study used as the baseline validation for the time-accurate flow solver is described in Sec. 3.1. Presented in this section are comparisons of the computational results to data for a NACA-0012 airfoil following a predefined pitching motion. Validation of the procedure for load-induced motion using the WL/AD wing/pylon/finned store configuration is discussed in Sec. 3.2. Validation is provided by comparing the computational results to the CTS test data and the CFD results of Lijewski and Suhs (Ref. 12). Application of the load-induced motion procedure to the generic TER configuration is discussed in Sec. 3.3. The effect of mutual interference on the trajectory of the inboard shoulder store is found by comparison

of the three-body launch to a single-body launch from the inboard shoulder position. A discussion of the conclusions resulting from this work is presented in Sec. 4.0. The SIXDOF code for integrating the 6DOF equations of motion for a moving rigid body is described in detail in Appendix A. A discussion of the validation cases used for the SIXDOF code is included in Appendix B. A study of the sensitivity of trajectories to aerodynamic loads and the ejector model, angular-rate damping, and errors in the aerodynamic loads is presented in Appendix C. A user's guide for the SIXDOF code is contained in Appendix D. The store ejector models used in SIXDOF are described in Appendix E. The procedure necessary to convert the dimensionless time step used in the XAIR code to the dimensional time step required by SIXDOF is documented in Appendix F. Finally, definitions of the axis system that are used throughout this report are presented in Appendix G.

## **2.0 METHODOLOGY**

### **2.1 CHIMERA OVERSET MESH METHODOLOGY**

A fundamental part of the CFD capability for computing unsteady aerodynamic forces and moments at the AEDC is the moving-mesh capability of the chimera overset-mesh procedure. This capability has been developed at the AEDC jointly with the NASA/Ames Research Center and WL/AD (Refs. 12–16). The basic chimera overset-mesh procedure (Refs. 17–20) allows the modeling of a complex body using relatively simple overlapping meshes, where each describes a component of the body. The solution on the composite mesh is made continuous by the intercommunication among the individual meshes. The moving-mesh chimera procedure allows a time-accurate solution when one or more of the meshes in the system moves in relation to the others. This section will discuss the computational procedure required for time-accurate body motion. Before presenting the procedure, however, the five basic codes used are described. The first code, SIXDOF, integrates the 6DOF equations of motion. The next two codes, PEGSUS and ROTRANS, are used to process the mesh system. The fourth code, XAIR, solves for the flow field, and the fifth code, TESS, calculates the aerodynamic loads on the body.

#### **2.1.1 SIXDOF**

For the time-accurate body motion capability, a well-designed code which integrates the 6DOF equations of motion was required. The existing 6DOF integration code used at the AEDC for CTS testing (Refs. 7-8) and in an engineering analysis code (Ref. 10), as well as the related code used at the WL/AD (Ref. 12), contained many features that were not required for the present applications. Moreover, these codes were not developed using modern software engineering practices, and are difficult to understand, use, and modify for use with the codes required for time-accurate CFD solutions. The SIXDOF code was therefore developed and is documented in this

report. This code has been designed to have flexibility for future extensions. For ease of use and modification, the SIXDOF logic and coding have been kept as simple and straightforward as possible. However, there are two restrictions in the present version that may have to be eliminated, as necessary, for future applications. These restrictions are:

1. Steady-state flight of the aircraft is assumed. The aircraft cannot be accelerating with respect to an inertial frame of reference.
2. The code models rigid-body dynamics only, i.e., the moments of inertia must be invariant in time (the mass and mass distribution of the body cannot change).

A note should be made that for some applications, the use of aerodynamic loads to determine the motion of the body is not desirable, as is the case for the 2-D airfoil validation case documented in this report. To permit such applications, the SIXDOF code is written to allow prescribed motion. For prescribed motion, an external file containing the rotational and translational motion of the body at each time increment replaces the integration of the equations of motion.

### **2.1.2 PEGSUS and ROTRANS**

The PEGSUS code positions the moving meshes in the proper location with respect to the non-moving meshes and defines the communication and interpolation data among the interacting meshes. Version 4.0 of PEGSUS (Ref. 21) includes enhanced capabilities that are useful for application to moving meshes, namely the ADD feature. This feature eliminates the necessity for modification of the communication links for all meshes when only a few meshes (and those meshes with which these few meshes interact) are changed. However, even with the ADD feature, the PEGSUS code is the most time-intensive code in the set. Fortunately, it was found by Ref. 12 that the motion of bodies over a single time increment was so small that updating the interpolation information was not necessary every time increment. Therefore, the PEGSUS code is not executed every time step. When PEGSUS is not executed, the ROTRANS code, which is approximately 250 times faster than PEGSUS, is used to place the moving meshes at the proper positions without updating any communication information.

### **2.1.3 XAIR**

The XAIR code (Ref. 19) is a three-dimensional implicit Euler and/or Navier-Stokes solver that is first-order accurate in time and second-order accurate in space using finite-difference approximations. XAIR is based on the implicit, approximate factorization scheme of Beam and Warming (Ref. 22). Inputs to XAIR include the composite mesh and the interpolation data created by PEGSUS. For all calculations documented in this work, the Euler equations were solved.

### 2.1.4 TESS

TESS was developed to overcome a potential source of error in the calculation of forces on a body in a flow field. The source of error is intrinsic to the chimera overset-mesh methodology. Bodies in conjunction, whether attached or in very close proximity to one another, e.g., a wing/fuselage combination or a store mounted on a pylon, require that the mesh of one body open a "hole" in the mesh of the second body with a concomitant overlapping of the surface meshes. Accurate definition of the hole and the multiply defined area caused by the overlapped meshes at the hole perimeter is required to determine the correct forces on the body through integration of the surface pressure distribution. TESS was developed to accomplish this task by using a single unstructured mesh to replace the multiple structured meshes describing the body surface.

As stated earlier, some applications will not require calculation of the aerodynamic loads to determine body motion. Obviously, for these cases execution of the TESS code is not required.

## 2.2 OVERALL PROCEDURE

For a body following a prescribed motion, the procedure for time-accurate calculations is straightforward. The translational and rotational displacements, which must be supplied by some arbitrary source, are used by the flow solver to predict the unsteady flow field. The overall procedure for predicting body motion with time-accurate computation of aerodynamic loads is more complex and is illustrated schematically in Fig. 1. Steps 1 to 3 are performed at carriage and are typical of steady-state solution procedures using the chimera overset-mesh method. The only difference is related to the distinction between fixed meshes, i.e., those associated with the aircraft in steady translation, and the store meshes that can move with respect to the fixed meshes. In Step 1, PEGSUS is applied to determine the lines of communication and to assemble the composite mesh for all of the fixed aircraft meshes. Then, using the ADD feature of PEGSUS 4.0, the moving meshes are added in Step 2 to complete the communication links and the composite mesh at carriage. The steady-state solution for the carriage configuration is obtained in Step 3 using XAIR followed by the corresponding aerodynamic load calculation by TESS. Local time steps can be used to accelerate the approach to the steady-state solution at carriage. Experience has shown that a high degree of convergence to the steady-state solution is critical to avoid spurious transients in the unsteady motion immediately after store release. Generally, convergence of the force and moment coefficients to four significant digits has been found to be satisfactory to avoid transients. The flow in a weapons bay, on the other hand, is inherently unsteady (see Refs. 23–24). Therefore, the entire solution both before and after store release must be time accurate.

Once the steady-state solution at carriage is obtained, the store is released at time  $t = t_o$  and Steps 4 to 6 are performed repeatedly until some trajectory completion criterion is met. In Step

4, SIXDOF integrates the 6DOF equations of motion using aerodynamic loads calculated by TESS from the most recent XAIR solution. SIXDOF yields the new location and orientation of the store. Then, in Step 5, the ADD feature of PEGSUS 4.0 is used in conjunction with the fixed aircraft composite mesh to relocate and reorient the store and assemble the new total composite mesh and corresponding interpolation data. If only relocation and reorientation are needed, ROTRANS is used in place of PEGSUS 4.0. Finally, XAIR is used in Step 6 to advance the flow-field solution one time step in a time-accurate manner.

The PEGSUS code requires considerable computational resources and it is not feasible to make a complete PEGSUS run at each time step. Fortunately, the physical distance moved at each time step is so small that it is not necessary to run PEGSUS entirely at each time step. Instead, the mesh is moved at each step according to the SIXDOF output, but the interpolation data are not updated until a prescribed number of time steps has been accomplished. At this point a full PEGSUS run is made to move the mesh and to update the interpolation data. The PEGSUS code and the ROTRANS code are used to implement this simplification. The ROTRANS code simply places the moving meshes in the proper location, while PEGSUS places the moving meshes and updates the interpolation data. No systematic investigation of criteria for the frequency of the interpolation update has been performed. For the example discussed in Sec. 3.2, the interpolation update was made every 0.002 sec (every 20 steps) to be consistent with the trajectory predictions of Lijewski and Suhs (Ref. 12). For the calculations discussed in Sec. 3.3, the interpolation update was made every 0.005 sec (every 25 steps).

The flow of information among the five codes is shown graphically in Fig. 2, which complements Fig. 1 and the discussion above. The information in the left-hand column is user-supplied data based on the geometry of the configuration, the individual computational meshes, and the required flow-solver inputs. The information in the right-hand column is generated by the five codes. Figure 2 is applicable to Steps 1 to 3 as well as to Steps 4 to 6. It must be emphasized that each of the four codes must be run in sequence for each time step.

### 3.0 RESULTS AND DISCUSSION

The discussion of results is presented in three sections. Section 3.1 presents results of the baseline validation case of the pitching airfoil and compares them to data. In Sec. 3.2, the results of the validation case of a single store trajectory prediction are presented and compared to data and the WL/AD CFD predictions. Section 3.3 shows the results of the application of the trajectory prediction procedure to a multiple and single body trajectory, which are then used to determine the effect of mutual interference on the trajectory of the single store.

### 3.1 BASELINE VALIDATION - TWO-DIMENSIONAL PITCHING AIRFOIL

Validation of the time-accurate body movement capability was accomplished using computations of a two-dimensional pitching airfoil. An investigation into the effect of the size of the time step was also performed. The particular case that was used for comparison was that of an airfoil which was pitched from a stationary position of  $\alpha = 0$  to  $\alpha = 16$  deg. To begin, the test article and experimental data are described. Then, the computational meshes and procedure are defined, followed by analysis and discussion of the results.

#### 3.1.1 Test Article and Experimental Data

The experimental data that were used for validation were taken from results that were reviewed and compiled particularly for validation of CFD methodologies (Ref. 25). The particular set of data used for validations was taken at the Aircraft Research Association (ARA) two-dimensional tunnel using the pitching and heaving rig. The wind tunnel model was a NACA 0012 airfoil with a chord of 4 in. and a span of 8 in. which was tested at Mach numbers ranging from 0.30 to 0.75 for quasi-steady, oscillating, and ramped (pitching) airfoil test cases. The pitching airfoil case was the only case considered for the validation since it best models a store separation event. The pitching case starts with the airfoil at or near  $\alpha = 0$  and then is allowed to pitch about the quarter chord to angles greater than the dynamic stall for the airfoil at free-stream conditions.

Thirty Kulite<sup>®</sup> pressure transducers, 18 on the upper surface and 12 on the lower surface, were used to record the time-varying pressure during the motion. Loads on the airfoil were determined after every time step by integrating the pressures over the surface of the airfoil.

#### 3.1.2 Computational Meshes

The domain was modeled with three computational meshes: an airfoil C-mesh, an inner Cartesian mesh, and an outer Cartesian mesh. The airfoil mesh contained 301 points in the streamwise direction, with 201 points on the airfoil surface and 31 points normal to the airfoil surface. The total mesh system contained 86,709 points. The motion of the airfoil was simulated by pitching the inner airfoil mesh using a predefined motion while the outer Cartesian meshes remained stationary

#### 3.1.3 Implementation

To begin the time-accurate computation, a steady-state solution was obtained for all cases with the airfoil at  $\alpha = 0$ ,  $M_\infty = 0.57$ . The final 1,000 time steps of the steady-state solutions were executed time-accurately (i.e., using a global time stepping) and used the same time step,  $DT$ , that

was used for the time-accurate pitching calculations. This procedure was done to ensure that changes in the solution occurred from the movement of the airfoil and not from a change in DT. The time-accurate solution was obtained while moving the airfoil using the prescribed motion described in Sec. 3.1.4. The PEGSUS code was executed every time step. Pressures along the airfoil and total loads were saved for every time step.

### 3.1.4 Results

Computations were performed to demonstrate the prediction of aerodynamic time history effects and to investigate the time step restrictions necessary to obtain valid results. Two experimental data cases, which will be referred to as Case A and Case B, were used for comparison to the computed results. The free-stream Mach number for both validation cases presented use the same Mach number of 0.57, but different pitch rates. Cases A and B had approximate pitch rates of 425 deg/sec and 1,380 deg/sec, respectively. During neither of the tests was a constant pitch rate achievable. Therefore, a variable pitch rate for each case was used and is plotted in Fig. 3. The equations used to model the motions were:

$$\text{Case A : } \alpha(\tau) = 0.0000023 \tau^2 (378 - \tau) \quad 0 \leq \tau \leq 133.3$$

$$\text{Case B : } \alpha(\tau) = 0.0000519 \tau^2 (150 - \tau) \quad 0 \leq \tau \leq 42.3 \quad (\text{Ref. 26})$$

where  $\tau = 2 * V_{\infty} * t / c$ ,  $V_{\infty}$  is the free-stream velocity,  $t$  is the time, and  $c$  is the airfoil chord length.

Shown in Fig. 4 is a comparison of computed  $C_N$  to the experimental data for Case A and Case B. Both cases were computed with a DT of 0.05. In addition, steady-state  $C_N$  predictions for this airfoil at several angles of attack are shown in Fig. 4. A comparison of the steady-state force coefficients to the time-accurate force coefficients demonstrates that an aerodynamic time history effect is computed. As shown by the figure, both cases compare favorably to the experimental data up to  $\alpha = 8.0$  deg. After this point, dynamic stall of the airfoil is apparent in the experimental data, but not in the inviscid calculations. Surprisingly, even though separation is a viscous phenomenon, the inviscid airfoil calculations also show a stall, although at a higher angle of attack.

Case B, which has the higher pitching rate, will now be examined in greater detail. Shown in Fig. 5 is  $C_p$  versus  $x/c$  for the airfoil on the upper and lower surfaces. Each plot depicts the  $C_p$  distribution at a different time during the pitching motion. Comparisons of the computed results to the measured data are in agreement up to  $\alpha = 5.67$  deg (Fig. 5g). For higher angles of attack, the  $C_p$  on the upper surface continues to decrease. A shock is also forming on the upper surface (Fig. 5h). The shock predicted by the computations is forward of the shock location in the measured data. As the pitching motion of the airfoil continues, the agreement between the computation and

the measured data worsens on the upper surface (Figs. 5h–m). Also, the lower surface agreement (Figs. 5m–p) degrades. Obviously, a separated region is forming on the upper surface of the airfoil which is not predicted by the inviscid computation. One other point of interest is that the agreement for  $C_p$  degrades much more noticeably at a lower  $\alpha$  than did the  $C_N$  (Fig. 4).

To better understand the effect of the time step on the solution, additional cases were computed using different time increments. Shown in Fig. 6 are the results for Case B using a DT of 0.005, 0.05, 0.1, and 0.2. As DT increases, so does the disparity between the computations and the data. This result is particularly evident with a DT of 0.2, where oscillations in the computational results can be observed. With a DT of 0.005, 0.05, and 0.1, the steady-state solution continued to converge, but with a DT of 0.2 the solution was slowly diverging. This trend suggests that the calculation at DT = 0.2 was probably unstable at the initiation of the movement and throughout the movement of the airfoil.

### 3.2 VALIDATION — SINGLE STORE TRAJECTORY

The data from the WL/AD generic wing/pylon/finned store CTS test discussed below will be used in this section for validation of the overall AEDC computational procedure described in Sec. 2.0. In addition to the wind tunnel data, the WL/AD time-accurate computations will be compared to the AEDC results. Before presentation of the results, a description of the test model and test procedures will be given.

#### 3.2.1 Test Article and Experimental Data

The full-span wing/pylon/finned store test article is shown in the photograph in Fig. 7 and the half-span schematic three-view drawing in Fig. 8. The coordinate system in Fig. 8 is the Global Grid Axis System chosen for this case. This axis system and the angles are defined in Appendix G. Pertinent dimensions of the test article and the ejector characteristics which were used in the CTS test are given in Table 1. Although this model is not representative of any actual store, the CTS test software modeled the trajectory of a store 20 times larger than the model. A clipped, delta-planform wing was used that had a leading-edge sweep angle of 45 deg and a NACA 64A010 symmetrical airfoil section. A cross section of the pylon is a symmetrical section with sharp tangent-ogive leading and trailing edges with no change in the section thickness along the pylon. The store model is a tangent-ogive/cylinder/tangent-ogive with four fins and an afterbody that is truncated by an attached cylindrical sting. All wind tunnel data used for comparisons include the sting. As shown in Fig. 7, a dummy store without a sting is situated symmetrically on the test article at carriage and remains fixed as the instrumented store moves along its trajectory. The CTS trajectory at a free-stream Mach number of 0.95 and a wing angle of attack of 0 deg (Ref. 12) was chosen as the check case.

In brief, the CTS wind tunnel method used in this case for determining the store trajectories involves physically moving a subscale model of the store in a quasi-steady fashion to each position and attitude of the trajectory as predicted by the on-line integration of the six-degree-of-freedom equations of motion. Other methods of trajectory prediction using the CTS system are often used, but are not relevant to this study (see Ref. 8). At the beginning of each integration step, the forces and moments acting on the store are those measured by balances within the CTS rig. The quasi-steady approximation (Ref. 27) considers that the loads at each step in the trajectory are those that would exist on the store if the flow were steady at the instantaneous three-dimensional translational and rotational velocities. The missing features of the quasi-steady approximation are the time-history effects, such as the effect on the body of the shedding of vortices into the wake; however, corrections are made for certain dynamic effects.

Induced changes to the angle of attack and sideslip due to the dynamic translational motion of the cg are accounted for with corrections to the oncoming wind vector by additional physical movement of the store. Note that this correction has the effect of changing the attitude of the store with respect to the inertial coordinate system from that which it would have in free drop. Also, the forces and moments are adjusted on-line to account for the damping effects (fluid resistance to motion) a dynamic store would encounter (Ref. 8) by assigning constant values for the pitch-, yaw, and roll-damping coefficients (given in Table 1 in this case). And, in the quasi-steady mode, the effects of the instantaneous rotational rates are accounted for in the CTS procedure by multiplying each damping coefficient by its respective angular velocity.

### **3.2.2 Computational Meshes**

The test articles were modeled with 12 meshes. The wing, the pylon, the store body with the attached sting, and each fin were modeled with an individual mesh. The remaining five meshes were used to improve boundary communications between the other meshes. The surface mesh of the combined store/sting was constructed with 15,251 points; each fin surface was modeled with 2,255 points; the surface mesh of the wing was composed of 10,767 points; and the pylon surface was constructed using 3,477 points. The entire mesh system contained approximately 1.6 million points.

### **3.2.3 Implementation**

The fully time-accurate store separation prediction procedure of Fig. 2 has been assessed by computing the trajectory for the WL/AD generic wing/pylon/finned store configuration discussed in Sec. 3.2.1. The computational meshes and time step used (0.0001 sec) are identical to those used by Lijewski and Suhs (Ref.12); however, the present calculations use a different 6DOF code, a different force integration method, and a different flow solver, although both solvers were used to

solve the Euler equations. The WL/AD flow solver is a finite-volume, implicit Euler code that uses a flux-difference splitting scheme. The angular rate-damping coefficients were inadvertently included in the computations of Ref. 12, but were not included in the present computations. As shown in Appendix C, however, this difference should have little effect on the trajectory. The store is assumed to have been released at  $t_o = 0.0$  sec. The time of interest for the trajectory in the present study was 0.36 sec.

As already noted, Ref. 12 used a different flow solver that was based on a finite-volume formulation. The present solver, XAIR, is based on a finite-difference formulation. This distinction is practically negligible, except for the manner in which mesh singularities such as those that occur at the leading edges of control surfaces or polar axes are handled. As Vinokur points out (Ref. 28), finite-volume schemes are generally less accurate in regions surrounding mesh singularities. Since both solutions used the same set of meshes which contain singularities, the differences in the computed CFD force and moment coefficients to be discussed below may be attributable, in part, to this fundamental difference in the flow solvers.

### 3.2.4 Results

The comparison between the computed results of the AEDC procedure and the CTS data of Ref. 12 will be presented separately for the translational (Fig. 9) and the rotational (Fig. 10) constituents, for convenience. Note that translation and rotation are coupled aerodynamically through the time-accurate response of the store to its motion relative to an inertial frame.

#### 3.2.4.1 Comparisons of Time-Accurate Predictions with CTS Test Data

The aerodynamic force-coefficient components are  $C_A$ , the axial-force coefficient in the negative  $x_B$  direction;  $C_y$ , the side-force coefficient in the positive  $y_B$  direction; and  $C_N$ , the normal-force coefficient in the negative  $z_B$  direction of the Body Axis System. The time-accurate predictions and the CTS data force coefficients are compared in Fig. 9a. The uncertainties of each component of the force-coefficient test data are shown in Fig. 9a. At carriage, the predicted and measured values of both the axial- and normal-force coefficients are in good agreement, but the predicted side-force coefficient is more inward than the measured value and lies outside the uncertainty band of the data. As the store motion progresses, the predicted force coefficients deviate from the data. These deviations arise, in part, from the different trajectories that the predicted CFD and measured CTS forces dictate, but also may be due to the aerodynamic time-history effects, the lack of viscous effects in the present Euler calculations, or errors in the wind tunnel data.

The translation of the store center of gravity is given by the components  $x_F$ ,  $y_F$ , and  $z_F$  in the Flight Axis System. The time-accurate predictions and the CTS data are compared in Fig. 9b. Good agreement is observed for all components; however, the  $z_F$  component is dominated by the gravitational and ejector forces; the aerodynamic force coefficient,  $C_N$ , is of much less importance.

The velocity components of the store center of gravity are  $u_I$ ,  $v_I$ , and  $w_I$  relative to the origin of the Inertial Axis System. The time-accurate predictions and the CTS data are compared in Fig. 9c. The corresponding acceleration components are  $\dot{u}_I$ ,  $\dot{v}_I$ , and  $\dot{w}_I$  in the Inertial Axis System (see Fig. 9d). The comparisons in both Figs. 9c and 9d are reasonable, with differences consistent with those shown in Fig. 9b. The discontinuity in the  $\dot{w}_I$  component and the discontinuity in the slope of the  $\dot{u}_I$  component just beyond  $t = 0.05$  sec in Fig. 9d result from cut-off of the aft ejector-piston force before cutoff of the forward ejector-piston force. In the ejector model used in SIXDOF and the CTS software, the ejectors act perpendicularly to the  $x_B$  axis at all times. Therefore, as the store pitches up, an  $x_B$  component of force is generated by the ejectors, thereby affecting  $\dot{u}_I$ .

The aerodynamic moment-coefficient components are  $C_l$ , the rolling-moment coefficient about the  $x_B$  axis;  $C_m$ , the pitching-moment coefficient about the  $y_B$  axis; and  $C_n$ , the yawing-moment coefficient about the  $z_B$  axis. All of these coefficients are in the positive sense about the respective axes in the Body Axis System. The time-accurate predictions and the CTS data moment coefficients are compared in Fig. 10a. The uncertainties of each component of the moment-coefficient test data are shown in Fig. 10a. At carriage, the predicted and measured values of the rolling-moment coefficient are in good agreement, but the predicted pitching-moment coefficient is more nose-down (negative) by about 0.2, and the predicted yawing-moment coefficient is more nose-out (positive) by about 0.3. As the store motion progresses, the predicted pitching- and yawing-moment coefficients approach the data and eventually cross them and deviate in the opposite direction. As in the case of the forces, these deviations arise in part from the different trajectories and possibly from the aerodynamic time-history effects and the lack of viscous effects in the present Euler calculations.

The rotation of the store about its center of gravity is given by the Euler-angle components  $\psi_F$ ,  $\theta_F$ , and  $\phi_F$  in the Flight Axis System. The time-accurate predictions and the CTS data are compared in Fig. 10b with general agreement in trends. The differences are accounted for by the sensitivity to pitching moment discussed more fully in Appendix C, and a similar sensitivity to the rolling and yawing moments. The predicted time-accurate aerodynamic loads are used in store trajectory calculations in the absolute sense; i.e., they must predict reality with great fidelity.

The rotational velocity components of the store about its center of gravity are  $p_B$ ,  $q_B$ , and  $r_B$  in the Body Axis System. The time-accurate predictions and the CTS test data are compared in

Fig. 10c. The pitching velocity comparison confirms that differences in the pitching moment can lead to significant differences in the trajectory results. The corresponding rotational acceleration components are  $\dot{p}_B$ ,  $\dot{q}_B$ , and  $\dot{r}_B$  in the Body Axis System (see Fig. 10d). The comparisons in Figs. 10c–10d are consistent with Fig. 10b. The discontinuities in the  $\dot{q}_B$  component just beyond  $t = 0.05$  sec in Fig. 10d result from moment differences arising from cutoff of the aft ejector-piston force before cutoff of the forward ejector piston force.

### 3.2.4.2 Comparisons of AEDC and WL/AD CFD Time-Accurate Predictions with CTS Test Data

The comparisons between the AEDC predictions and the CTS test data have been presented in Sec. 3.2.4.1. Presented in this section is a comparison of the AEDC predictions with the WL/AD predictions. The translational and rotational results will be discussed separately again, for convenience. It must be reemphasized, however, that there is significant aerodynamic coupling between the translational and rotational degrees of freedom.

The translational results will be compared first in Fig. 11. The force coefficient components are shown in Fig. 11a, which is identical to Fig. 9a except for the addition of the Ref. 12 data. The AEDC results for the time history of the axial-force coefficient,  $C_A$ , are superior to the WL/AD results, especially at carriage and during the first 0.2 sec of store motion. This follows because the loads calculation in the TESS code includes a base-drag correction that is found by averaging the computed pressures around the periphery of the base. This base-drag correction reflects the test article quite accurately because a small gap is present between the store and the sting at the base of the store. This gap leads to a pressure within the model that is nearly equal to the peripheral pressure, so that the base pressure acts on an area approximately the size of the base. Both CFD predictions of the normal-force coefficient,  $C_N$ , are comparable in magnitude. They are in good agreement with data at carriage, but fall below the data as time elapses. Both CFD predictions of the side-force coefficient,  $C_y$ , are also comparable in magnitude. They are more inward (negative) than the CTS data at carriage, but cross over and are increasingly more nose-outward (positive) as time elapses. It should be emphasized that the forces are closely related to the details of the store trajectory, namely the position, the attitude, and the translational and rotational velocities and accelerations. Since the predicted and computed trajectories differ as time elapses, so will the forces.

The center-of-gravity locations in the Flight Axis System,  $x_F$ ,  $y_F$ , and  $z_F$ , are presented in Fig. 11b. The AEDC and WL/AD predictions show reasonable agreement with each other and the CTS data. The largest deviation from the data occurs for the  $z_F$  component, which differs from the data by less than 0.5 ft in either calculation. The AEDC results show superior agreement with the CTS  $x_F$  component of the data because the axial-force coefficient,  $C_A$ , is predicted more

accurately. The velocity components of the center of gravity,  $u_1$ ,  $v_1$ , and  $w_1$ , are presented in Fig. 11c. Both CFD predictions are consistent with the force and cg location results, but the velocity is a higher-order derivative, and the effects of the differences between details of the cg location time histories in the two CFD prediction models are amplified.

The rotational results are compared in Fig. 12. The moment coefficient components are shown in Fig. 12a, which is identical to Fig. 10a except for the addition of the Ref. 12 data. Both sets of CFD predictions for the time history of the rolling-moment coefficient,  $C_l$ , are essentially in agreement with the CTS data, although the magnitudes are small. However, the WL/AD  $C_l$  predictions do increase more than the AEDC predictions and the CTS data beyond about 0.2 sec. Both CFD predictions of the yawing-moment coefficient,  $C_n$ , agree until about 0.25 sec have elapsed, after which they diverge. The overall agreement of the predicted  $C_n$  with the CTS data is not good, with overprediction at carriage and underprediction beyond about 0.2 sec. The largest differences between the CFD predictions and the CTS data are for the pitching-moment coefficient,  $C_m$ , where there is underprediction at carriage and overprediction beyond about 0.25 sec.

The rotational history of the store about its center of gravity in the Flight Axis System,  $\psi_F$ ,  $\theta_F$ , and  $\phi_F$ , is presented in Fig. 12b. Both CFD predictions of the time history of the yaw angle agree, but lie above the CTS data. There is a significant difference in the predicted pitch angles, however, with the WL/AD results lying closer to the CTS data than the AEDC results throughout the entire time history. The reason for this is the extreme sensitivity of pitch angle to both pitching moment and ejector model (see Appendix C). The AEDC- and the WL/AD-predicted pitching-moment coefficients and ejector forces differed, thus producing different pitch results. The ejectors for the AEDC calculation were modeled after what was specified for the wind tunnel test. The ejectors modeled for the WL/AD calculation were adjusted to give the best match of the wind tunnel trajectories when using the wind tunnel loads in the WL/AD 6DOF code. Both CFD predictions of the roll angle agree until about 0.25 sec, after which they diverge, with the AEDC results remaining closer to the CTS data. This is consistent with the WL/AD rolling-moment coefficient increase mentioned earlier. The rotational velocity history of the store,  $p_B$ ,  $q_B$ , and  $r_B$ , is presented in Fig. 12c. The CFD predictions are consistent with the rotational history results.

The results of the two CFD prediction methods differ somewhat for reasons that are not clearly understood, except for the reasons already given at the beginning of Sec. 3.2.3, i.e., the difference between finite-volume and finite-difference discretization at mesh singularities. It would be very difficult to pinpoint the precise reasons, given the complexity of the respective 6DOF and time-accurate flow solvers and the large computational expense of a systematic investigation. Nevertheless, both procedures produce compatible and satisfactory results for this case.

### 3.3 APPLICATION - MULTIPLE-BODY MUTUAL INTERFERENCE

To demonstrate and test the multiple-body trajectory capability, a trajectory prediction for a hypothetical multiple-body launch was obtained. Three stores were placed near the WL/AD wing/pylon previously described in a simulated TER configuration. No attachment hardware was modeled and the store fins were arranged in an 'x' pattern. The positions of the stores at carriage are illustrated in Fig. 13. The computation was performed at a Mach number of 0.95 with the wing at 0-deg angle of attack. The stores were launched from the pylon in a bottom-outboard-inboard sequence with 40 msec between each launch. Computational results (carriage loads and trajectories) for the multiple-body case and the case of the single store launched from the inboard shoulder position were compared to evaluate the effect of mutual interference on the trajectory. No experimental loads or trajectory data were available for this configuration.

#### 3.3.1 Geometry/Computational Meshes

The wing and pylon mesh systems were the same as described for the single-body validation case. The store body and fin meshes were modified to remove the attached sting of the single-body validation case. The bodies of the stores used in the multiple-body launch tapered to a point at the rear (see Fig. 13). The surface mesh of each store and fin contained 10,087 points and 2,020 points, respectively. The entire solution set for the TER case consists of 20 meshes and approximately 2.1 million points. The meshes for the single store prediction were identical to those described above; however, since two of the stores have been removed, this solution set consists of 12 meshes and approximately 1.0 million points.

#### 3.3.2 Implementation

For the multiple-body trajectory prediction, the three stores were launched in a bottom-outboard-inboard sequence with 40 msec between launches. A full-scale time increment of 0.0002 sec, which is twice that of the single-store trajectory validation case was used in the time marching procedure. The PEGSUS code was used to update the interpolation coefficients every twenty-fifth time step. The prediction was terminated at  $t = 0.32$  sec, requiring 1,600 time steps. The ejector model used for this case models constant force ejectors which shut off at a predetermined ejector length as described earlier and in Appendix E. The bottom store is ejected straight down, while each of the shoulder stores are ejected down and to the side of the pylon at a 45-deg angle. Physical parameters for each store are given in Table 2. Note that the weight, moments of inertia, and ejector forces have been reduced from those used in the single-body validation case. This was done to allow the aerodynamic forces to make more of a contribution to the trajectory than was seen for the validation case.

The implementation of the trajectory prediction procedure for the single inboard store was identical to that of the multiple-body launch, except that the trajectory started at  $t = 0.08$  sec, terminated at  $t = 0.365$  sec, and required 1,425 time steps.

### 3.3.3 Results

#### 3.3.3.1 Multiple-Body Trajectory

The computed carriage loads on the bodies used in the simulated TER configuration are listed in Table 3. All three stores have a nose-out yawing moment and a nose-down pitching moment. From a downstream vantage point, the shoulder stores have a clockwise rolling moment while the bottom store has a small counterclockwise rolling moment. An inward side force is exerted on all three stores, and the drag of each is comparable. The bottom and inboard stores have an upward force, while the outboard store has a downward force.

The cg location and rotation angle of the bottom store plotted as a function of time are shown in Fig. 14. For this store at the instant of release, the ejectors push downward with 9,000 lbf and impart a 6,714 ft-lbf nose-up pitching moment. The aerodynamic forces cause an upward force of 492 lbf and a nose-down pitching moment of 1,373 ft-lbf. Obviously, for some period of time the ejector loads override the aerodynamic loads, causing the store to move downward with a nose-up pitch. The store initially pitches nose-up due to the positive ejector moment, reaching a maximum of approximately 7.5 deg, but eventually reverses direction due to the negative aerodynamic pitching moment. However, the ejectors exert no axial or sideward component; therefore, the aerodynamic loads cause the downstream translation and nose-out yaw.

The outboard store was launched at  $t = 0.040$  sec. The trajectory of this store is illustrated in Fig. 15. In this case, the ejector pistons exert the same force as for the bottom store, but are angled at 45 deg outward. Therefore, the downward and outward forces exerted by the pistons are 6,364 lbf, while the nose-up pitching moment and nose-in yawing moments are 4,748 ft-lbf. In comparison, the normal and side carriage forces due to the aerodynamics are 33 lbf, downward, and 499 lbf, inward, respectively. The nose-down pitching and nose-out yawing carriage moments were 827 ft-lbf and 1,487 ft-lbf, respectively. For the outboard store, the ejectors initially override the aerodynamic loads, except for the normal force, which assists the ejectors. The effect of the ejectors working at a 45-deg angle is well illustrated in Fig. 15a. In comparison to the bottom store, the outboard store has a larger outward translation and a smaller downward translation. As can be seen from Fig. 15b, the initial nose-in yawing and nose-up pitching motions imparted by the ejectors are eventually overcome by the nose-out and nose-down moments caused by the aerodynamics.

At  $t = 0.080$  sec the inboard store was launched. Illustrated in Fig. 16 is the trajectory of the inboard store. Again, the effect of the 45-deg ejector loads can be seen from the inward and downward motion of the cg and the initial nose-up pitching and nose-out yawing motions. The magnitudes of the ejector loads match those of the outboard store; however, the direction of the yawing moment is changed to a nose-out direction. For comparison, the aerodynamics caused an upward force of 622 lbf, an inward force of 499 lbf, a nose-down pitching moment of 2,499 ft-lbf, and a nose-out yawing moment of 1,002 ft-lbf. As was the case for the other stores, the initial nose-up pitching motion is eventually reversed because of the aerodynamic loads. However, unlike the outboard store, the nose-out yawing motion does not reverse. As can be seen in Fig. 16b, the store yaws 15 deg before the simulation terminates. For the inboard store, the ejector forces and the aerodynamic forces combine to produce a large nose-out yawing moment.

### 3.3.3.2 Effect of Mutual Interference on Inboard Store Trajectory

A comparison of the isolated inboard store trajectory and carriage loads to the analogous values in the TER configuration will serve to quantify the effects of mutual interference. Differences will be due to interference effects from the other stores at carriage and throughout the trajectory.

The carriage loads of the steady-state solution are listed in Table 4. The absence of the other two stores causes a significant change in the carriage loads. With the removal of the other two stores, the drag decreases slightly, the side force and normal force increase, and the moments about all three axes increase.

Depicted in Fig. 17 is the inboard store translation and rotation for both the TER configuration and the single-store configuration. Even with the changes in carriage loads, the location of the cg is almost the same for both cases. The rotation angles are very similar, with the largest discrepancy appearing in pitch, which differs by about 4 deg. As was the case for the multiple-body launch, the ejectors override most of the aerodynamic forces in determining the motion of the store, particularly in translation. In this case, the forces exerted by the ejectors total 9,000 lbf, while the forces corresponding to a  $C_N$  of 0.990 total approximately 1,026 lbf. The pitching moment exerted by the ejectors totals 4,748 ft-lbf, while the pitching moment corresponding to a  $C_m$  of 2.077 totals 3,587 ft-lbf. The disparity between the ejector and aerodynamic pitching moments is smaller than the differences seen for the stores in the multiple-body launch, with a noticeable effect on pitch. Thus, for this configuration, the mutual interference between the inboard store and the bottom and outboard stores has little effect on the translational path, but affects the rotation of the store.

## **4.0 CONCLUSIONS**

### **4.1 PITCHING AIRFOIL**

A time-accurate inviscid CFD capability for bodies in motion has been developed at AEDC. The time-accurate solution capability is validated with a pitching airfoil by comparison with data. The pressure distribution trends for the inviscid pitching airfoil calculations agree with the data until the wing stalls. The mismatch at stall is to be expected when using an inviscid flow assumption.

### **4.2 SINGLE-STORE TRAJECTORY**

The trajectory prediction procedure is validated by comparisons of the single WL/AD store trajectory prediction with a CTS trajectory and a time-accurate trajectory prediction performed by WL/AD personnel. The comparisons were good overall. Comparisons with CTS-produced cg locations were excellent, and the comparisons of store attitude were acceptable, especially considering the predictions were made using an inviscid flow-field model.

The less accurate prediction of store attitude was expected since the incorrect shock position is predicted by the inviscid assumption and, therefore, affects moments most adversely; however, the more pronounced inaccuracy in pitching led to a study of the sensitivity of store attitude to various factors. The store's pitch attitude was severely affected by seemingly small differences in the ejector forces and timings. These rather small differences at release served to affect the attitude markedly as the trajectory progressed.

In comparison to the WL/AD prediction, the AEDC trajectory displayed acceptable agreement both in cg location and in attitude, with some difference in pitch. The differences in the trajectories can be attributed to the differences in the ejector models and differences in the computed carriage moments. The differences in carriage moments (the pitching moments in particular) could be attributed to the differences in the flow solvers.

### **4.3 MULTIPLE-BODY TRAJECTORY**

A time-accurate multiple-body trajectory prediction capability has been developed and applied to a mutual interference problem. For the duration of the calculated trajectory, the mutual interference between the inboard store and the bottom and outboard stores had little effect on the store translation, since the ejector forces were almost nine times larger than the aerodynamic forces. However, the mutual interference may affect the rotation of the store, since the ejector moments were typically only 1.3 times larger than the aerodynamic moments.

**REFERENCES**

1. Fox, J. H. and Allee, E. G. "Experimental/Computational Study of a Transonic Aircraft with Stores." AIAA Paper 89-1832, June 1989.
2. Fox, J. H., Donegan, T. L., Jacocks, J. L., and Nichols, R. H. "Computed Euler Flowfield for a Transonic Aircraft with Stores." *Journal of Aircraft*, Vol. 28, No. 6, June 1991, pp. 389–396; also AIAA Paper 89-2219, July/August 1989.
3. Donegan, T. L. and Fox, J. H. "Analysis of Store Trajectories from Tactical Fighter Aircraft." *Proceedings of the 8<sup>th</sup> JOCG Aircraft/Stores Compatibility Symposium*. Fort Walton Beach, FL, 23–25 October 1990.
4. Donegan, T. L. and Fox, J. H. "Analysis of Store Trajectories from Tactical Fighter Aircraft." AIAA Paper 91-0183, January 1991.
5. Nichols, R. H., Jacocks, J. L., and Rist, M. J. "Calculation of the Carriage Loads of Tandem Stores on a Fighter Aircraft." AIAA Paper 92-0283, January 1992.
6. Jordan, J. K. "Computational Investigation of Predicted Store Loads in Mutual Interference Flow Fields." AIAA Paper 92-4570-CP, August 1992.
7. Carman, J. B., Jr. "Store Separation Testing Techniques at the Arnold Engineering Development Center - Volume I – An Overview." AEDC-TR-79-1, Vol. 1 (AD-A087583), August 1980.
8. Carman, J. B., Jr., Hill, D. W., Jr., and Christopher, J. P. "Store Separation Testing Techniques at the Arnold Engineering Development Center — Volume II — Description of Captive Trajectory Store Separation Testing in the Aerodynamic Wind Tunnel (4T)." AEDC-TR-79-1, Vol. II (AD-A087561), June 1980.
9. Allee, E. G., Jr. "Store Separation Testing Techniques at the Arnold Engineering Development Center — Volume IV — Description of Dynamic Drop Store Separation Testing." AEDC-TR-79-1, Vol. IV (AD-A087490), June 1980.
10. Keen, K. S. "New Approaches to Computational Aircraft/Store Weapons Integrations." AIAA Paper 90-0274, January 1990.

11. Lohner, R. and Baum, J. D. "Three Dimensional Store Separation Using a Finite Element Solver and Adaptive Remeshing." AIAA Paper 91-0602, January 1991.
12. Lijewski, L. E. and Suhs, N. E. "Chimera-Eagle Store Separation." AIAA Paper 92-4569, August 1992.
13. Meakin, R. L. "Computations of the Unsteady Flow About a Generic Wing/Pylon/Store Configuration." AIAA Paper 92-4568-CP, August 1992.
14. Dougherty, F. C., Benek, J. A., and Steger, J. L. "On Applications of Chimera Grid Scheme to Store Separation." NASA-TM-88193, October 1985.
15. Dougherty, F. C. and Kuan, J. "Transonic Store Separation Using a Three-Dimensional Chimera Grid Scheme." AIAA Paper 89-0637, January 1989.
16. Meakin, R. L. and Suhs, N. E. "Unsteady Aerodynamic Simulation of Multiple Bodies in Relative Motion." AIAA Paper 89-1996-CP, June 1989.
17. Benek, J. A., Steger, J. L., and Dougherty, F. C. "A Flexible Grid Embedding Technique with Applications to the Euler Equations." AIAA Paper 83-1944, July 1983.
18. Benek, J. A., Buning, P. G., and Steger, J. L. "A 3-D Chimera Grid Embedding Technique." AIAA Paper 85-1523, July 1985.
19. Benek, J. A., Dougherty, F. C., and Buning, P. G. "Chimera: A Grid Embedding Technique." AEDC-TR-85-64 (AD-A167466), December 1985.
20. Benek, J. A., Donegan, T. L., and Suhs, N. E. "Extended Chimera Grid Embedding Scheme with Applications to Viscous Flows." AIAA Paper 87-1126, June 1987.
21. Suhs, N. E. and Tramel, R. W. "PEGSUS 4.0 User's Manual." AEDC-TR-91-8 (AD-A280608), November 1991.
22. Beam, R. and Warming, R. F. "An Implicit Factored Scheme for Compressible Navier Stokes Equations." AIAA Paper 78-10, January 1978.
23. Suhs, N. E. "Computations of Three-Dimensional Cavity Flow at Subsonic and Supersonic Mach Numbers." AIAA Paper 87-1208, June 1987.

24. Suhs, N. E. "Transonic Flow Calculations for a Cavity with and without a Store." AEDC-TR-92-4 (AD-A255954), September 1992.
25. "Compendium of Unsteady Aerodynamic Measurements." AGARD Report 702.
26. "AGARD Two-Dimensional Aeroelastic Configurations." AGARD Advisory Report 156.
27. Fung, Y. C. *An Introduction to the Theory of Aeroelasticity*. John Wiley & Sons, Inc., New York, 1955.
28. Vinokur, M. "An Analysis of Finite-Difference and Finite-Volume Formulations of Conservation Laws." *Journal of Computational Physics*, Vol. 81, 1989.
29. Goldstein, H. *Classical Mechanics*. Addison-Wesley Publishing Company, Inc., Reading, Massachusetts, 1980 (Second Edition).
30. Sanz Serna, J. M. "Symplectic Runge-Kutta and Related Methods: Recent Results." *Physica D – Nonlinear Phenomena*, Vol. 60, Nos. 1–4, 1 November 1992, pp. 293–302.
31. Zill, D. G. *A First Course in Differential Equations with Applications*. Prindle, Weber and Schmidt, Boston, Massachusetts, 1982 (Second Edition).
32. Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. *Numerical Recipes — The Art of Scientific Computing*. Cambridge University Press, Cambridge, England, 1986.

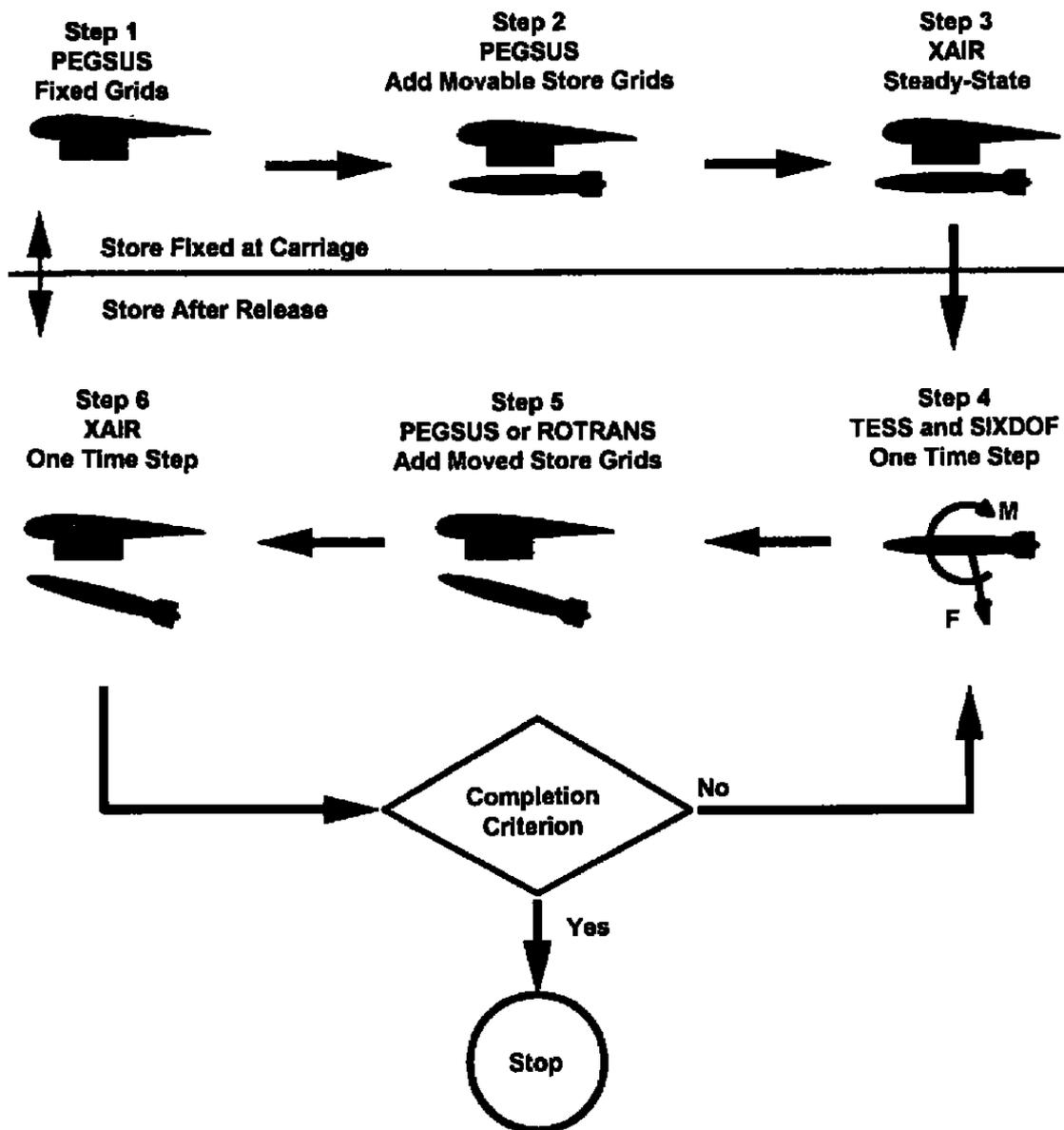


Figure 1. Time-accurate store trajectory prediction process.

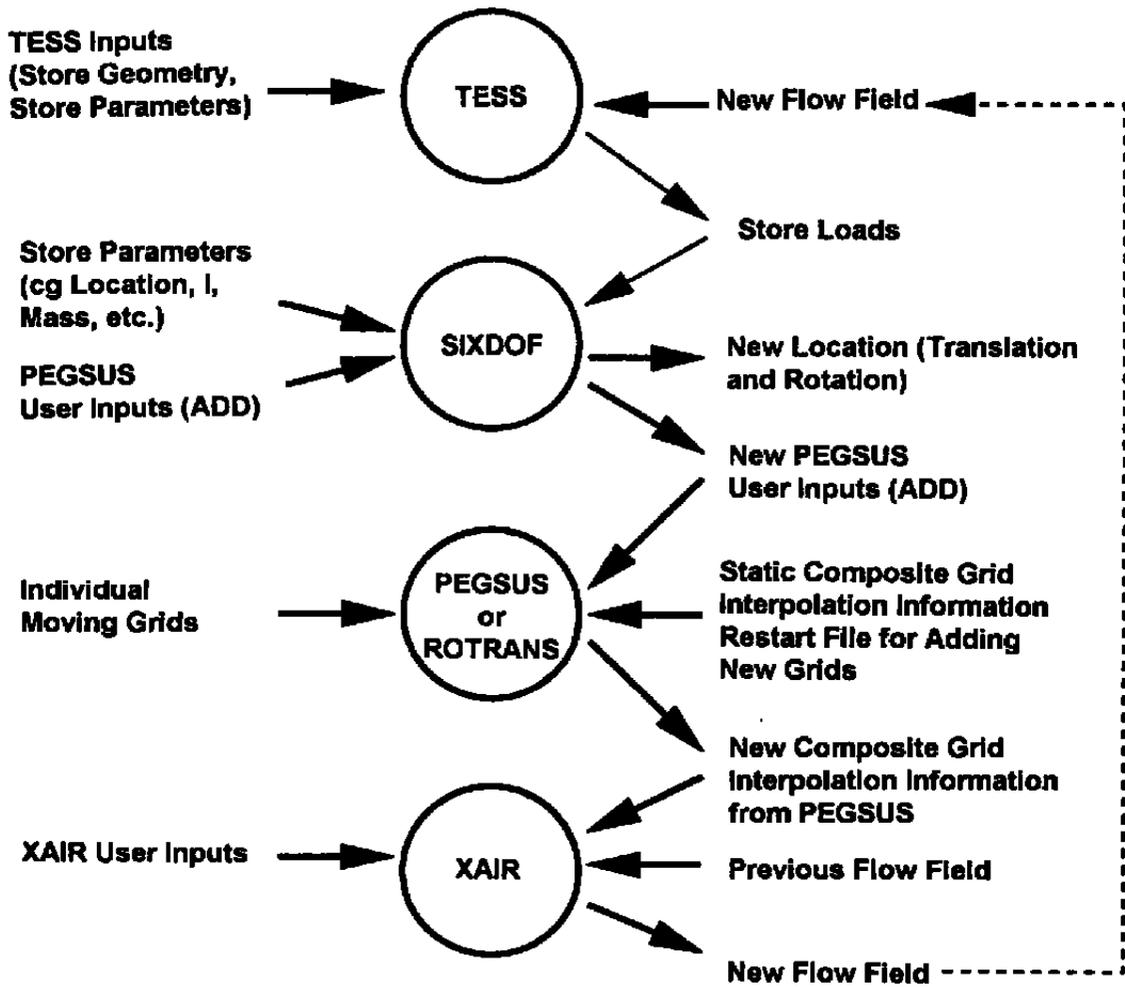


Figure 2. Information flow for the time-accurate store trajectory prediction process.

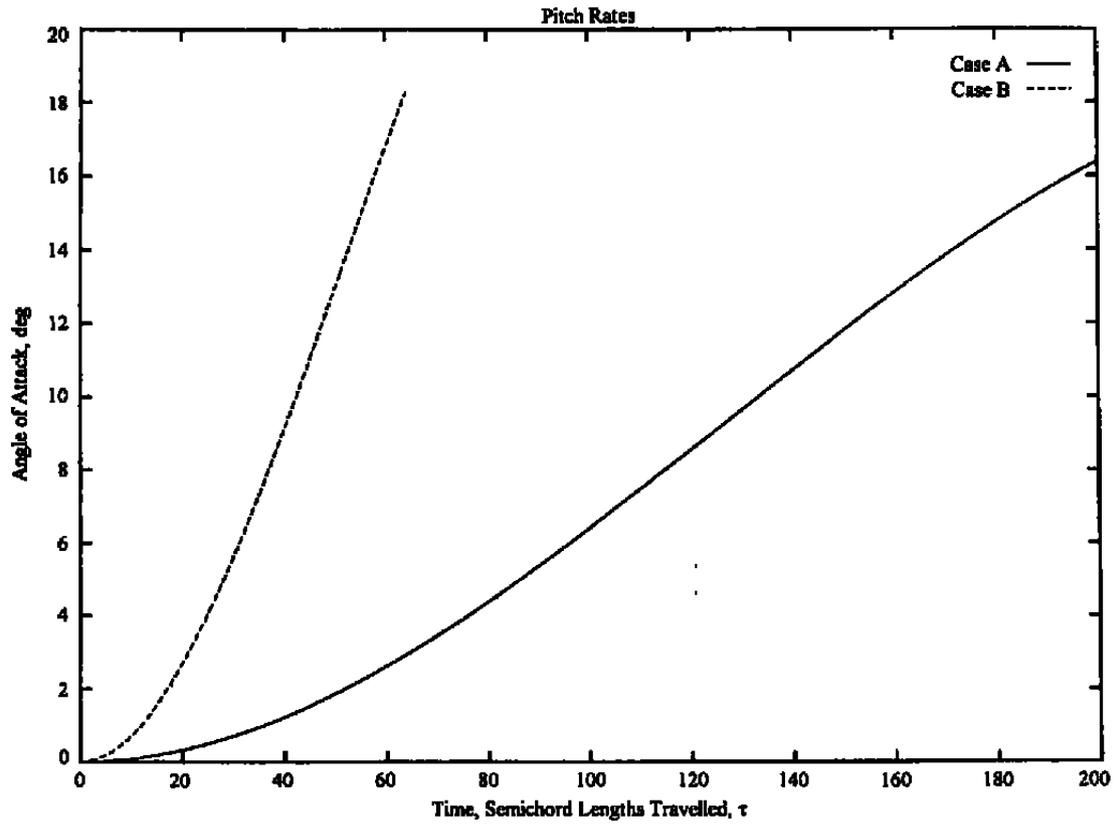
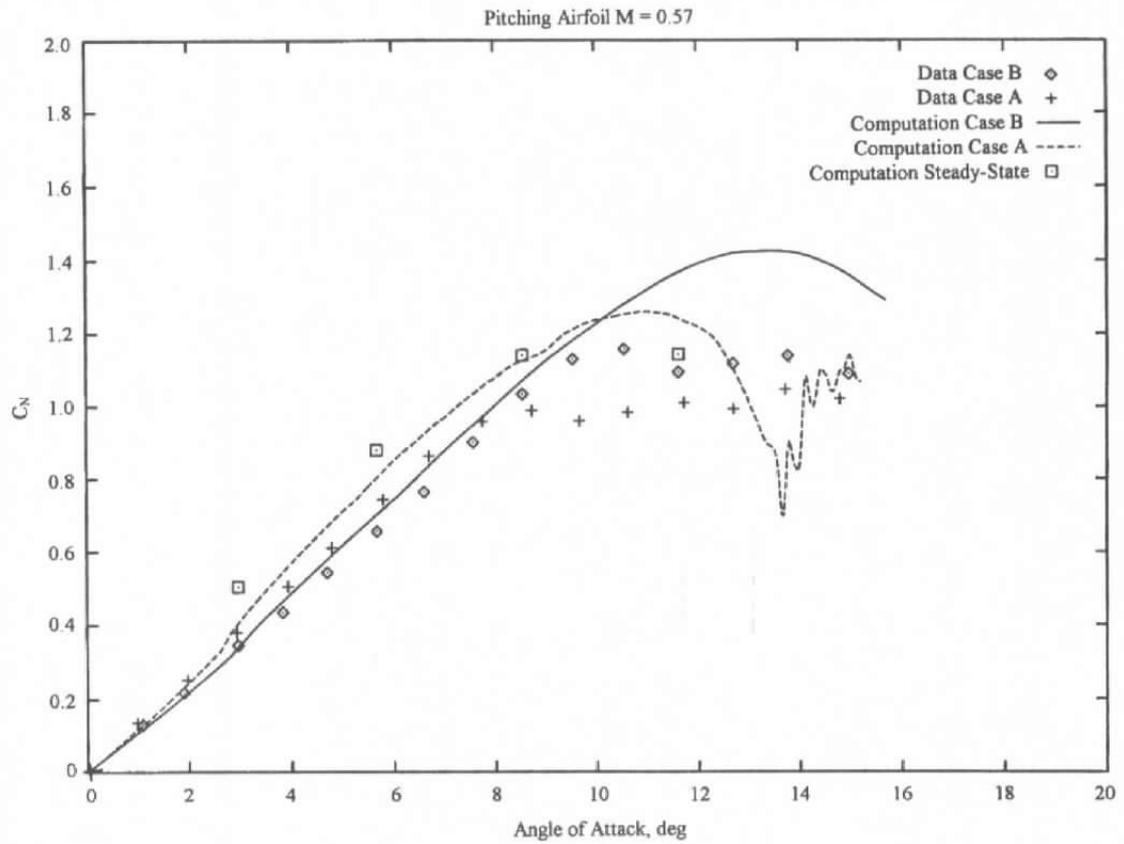


Figure 3. Pitch rates for NACA-0012 airfoil.



**Figure 4. Comparisons of computations and measured data for pitching airfoil, Cases A and B.**

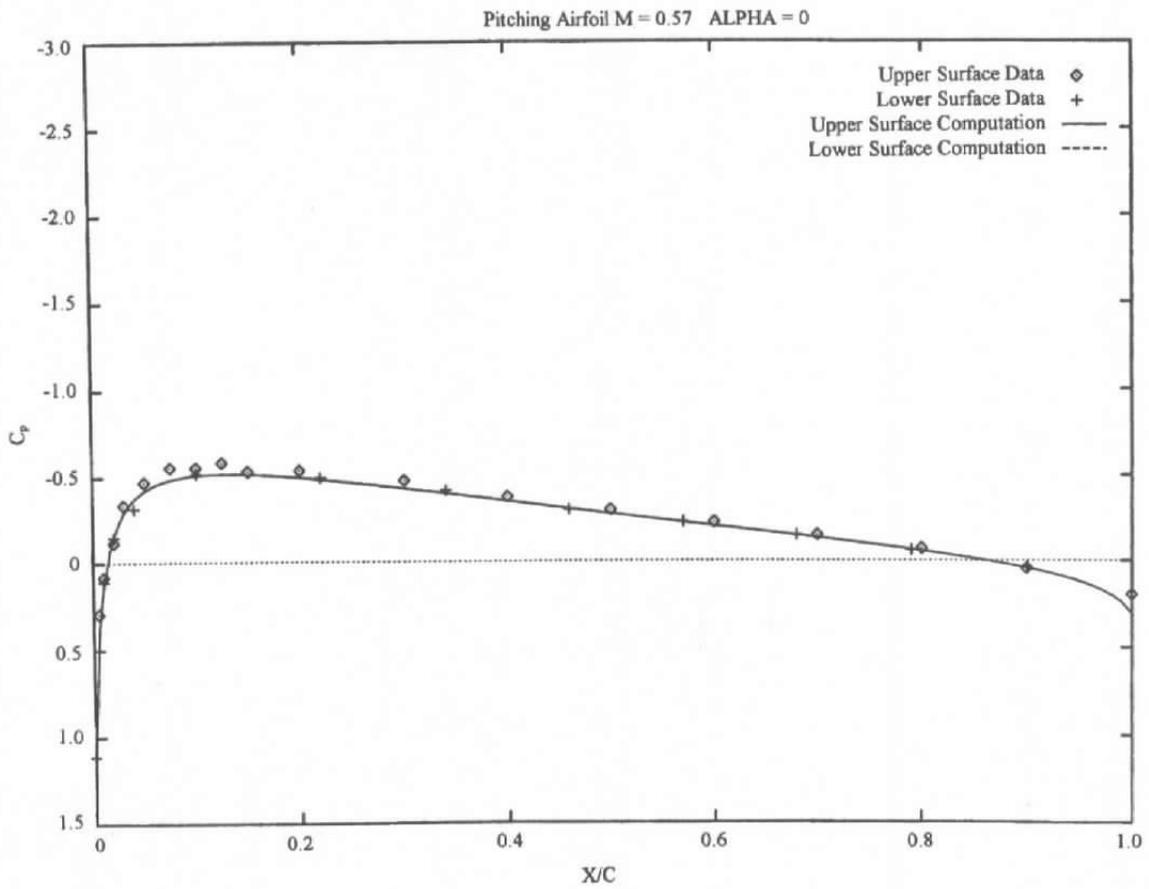
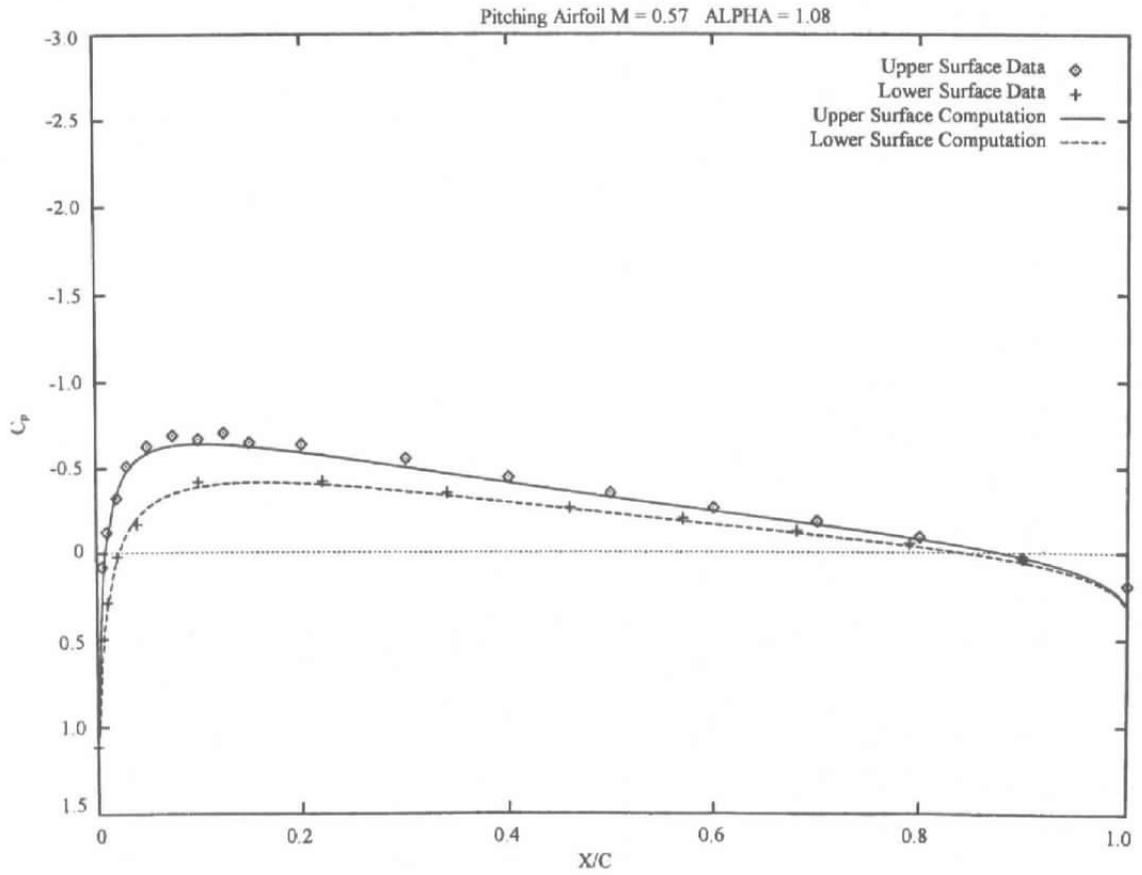
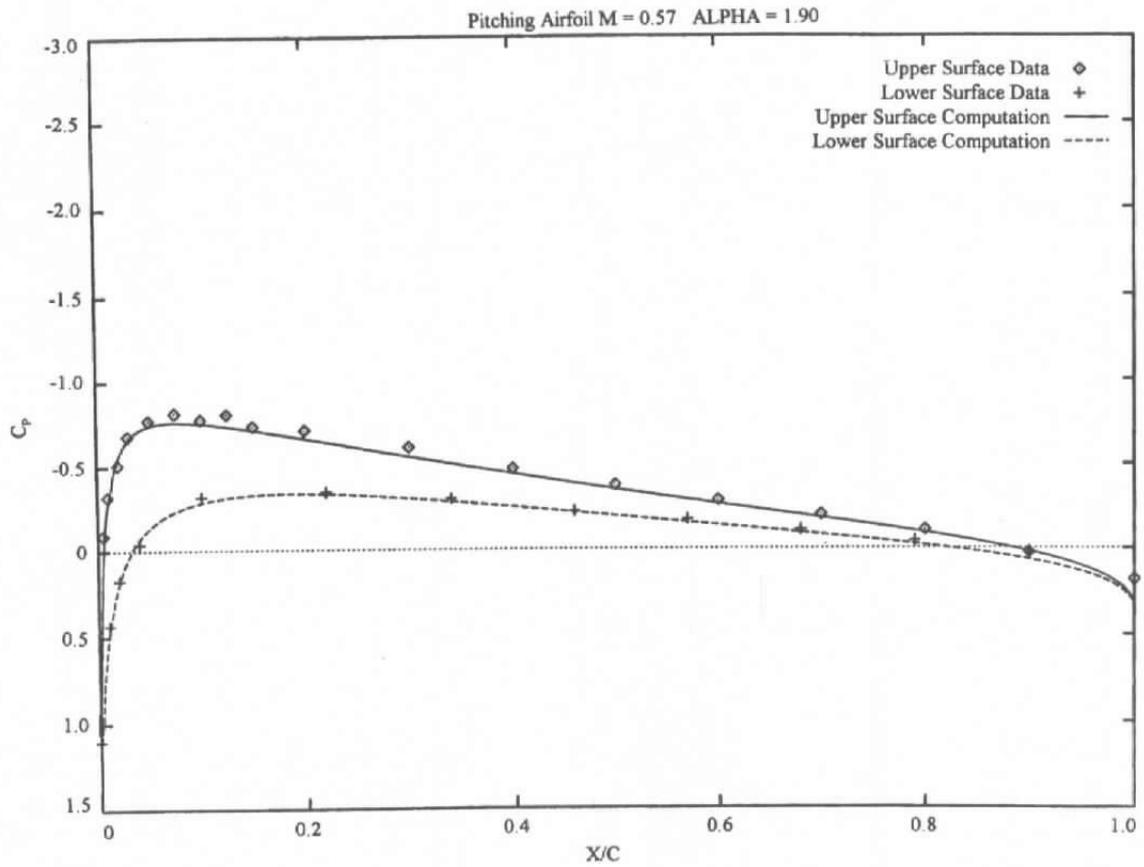


Figure 5. Time history of pressure coefficients on the pitching airfoil for Case B.

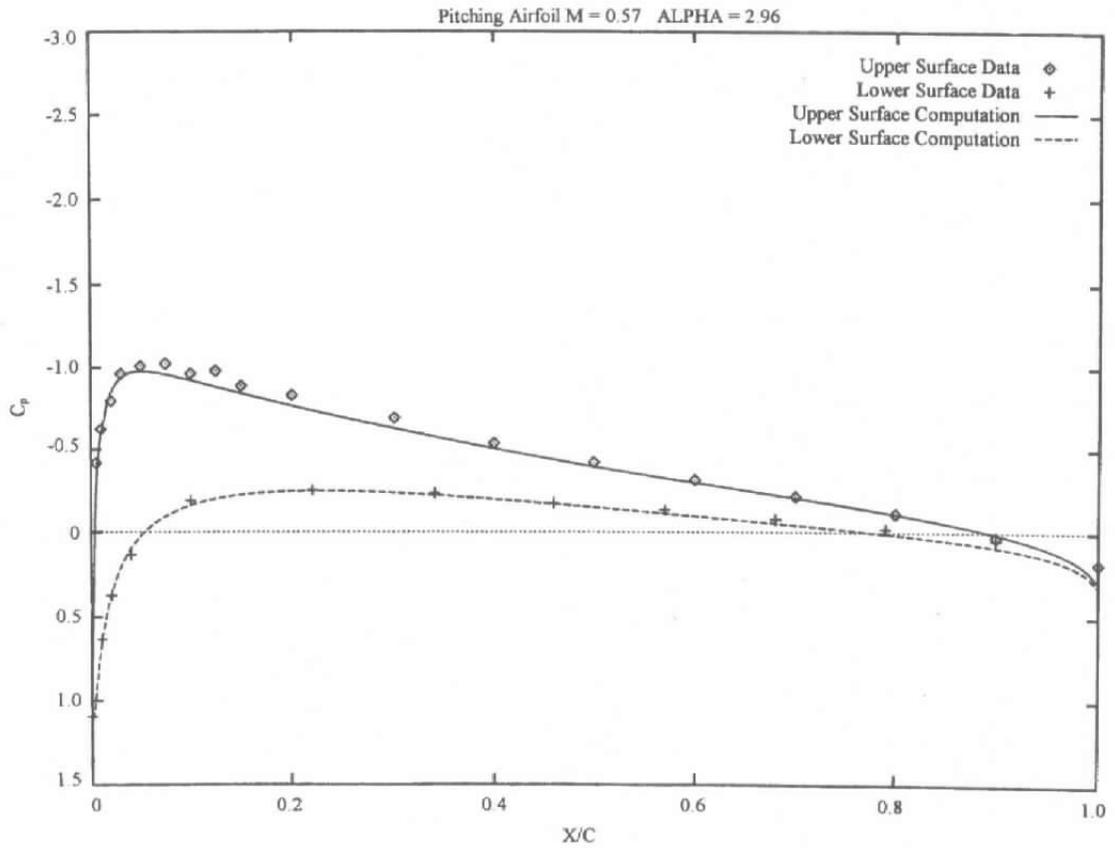


b.  $\alpha = 1.08$   
**Figure 5. Continued.**

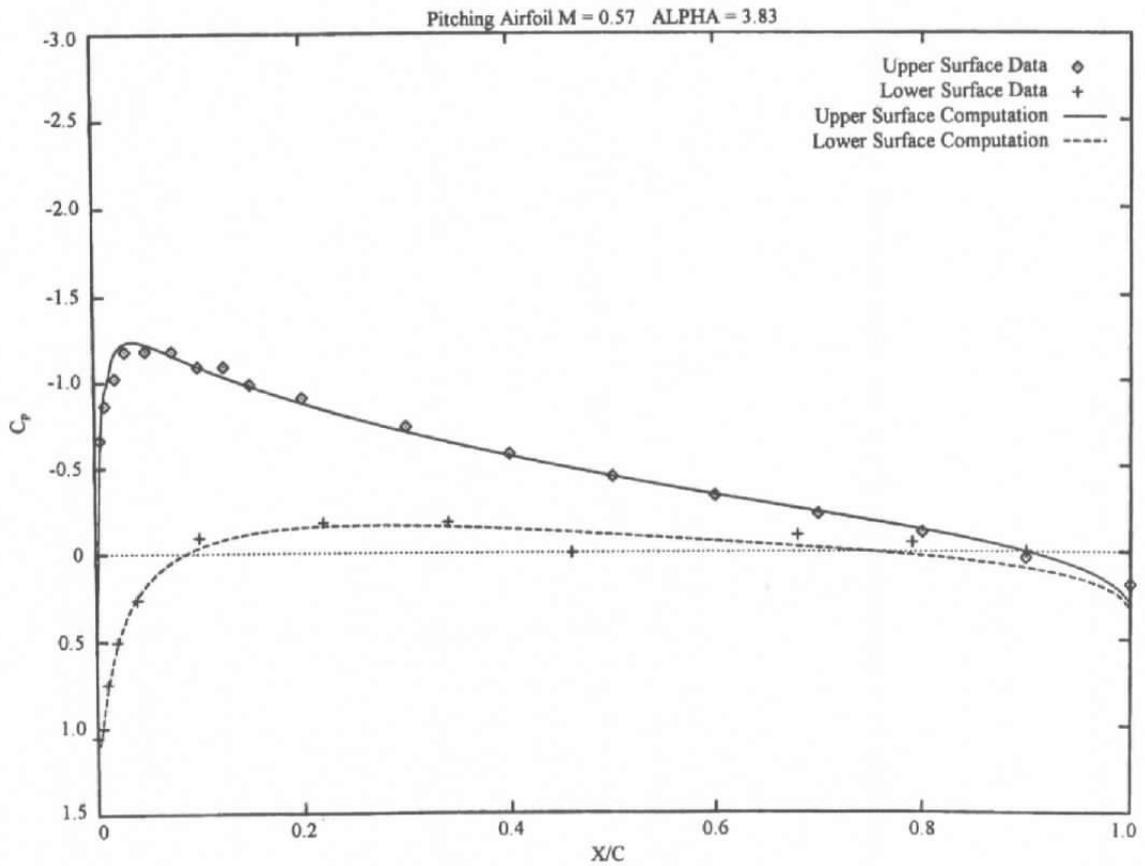


c.  $\alpha = 1.90$

Figure 5. Continued.

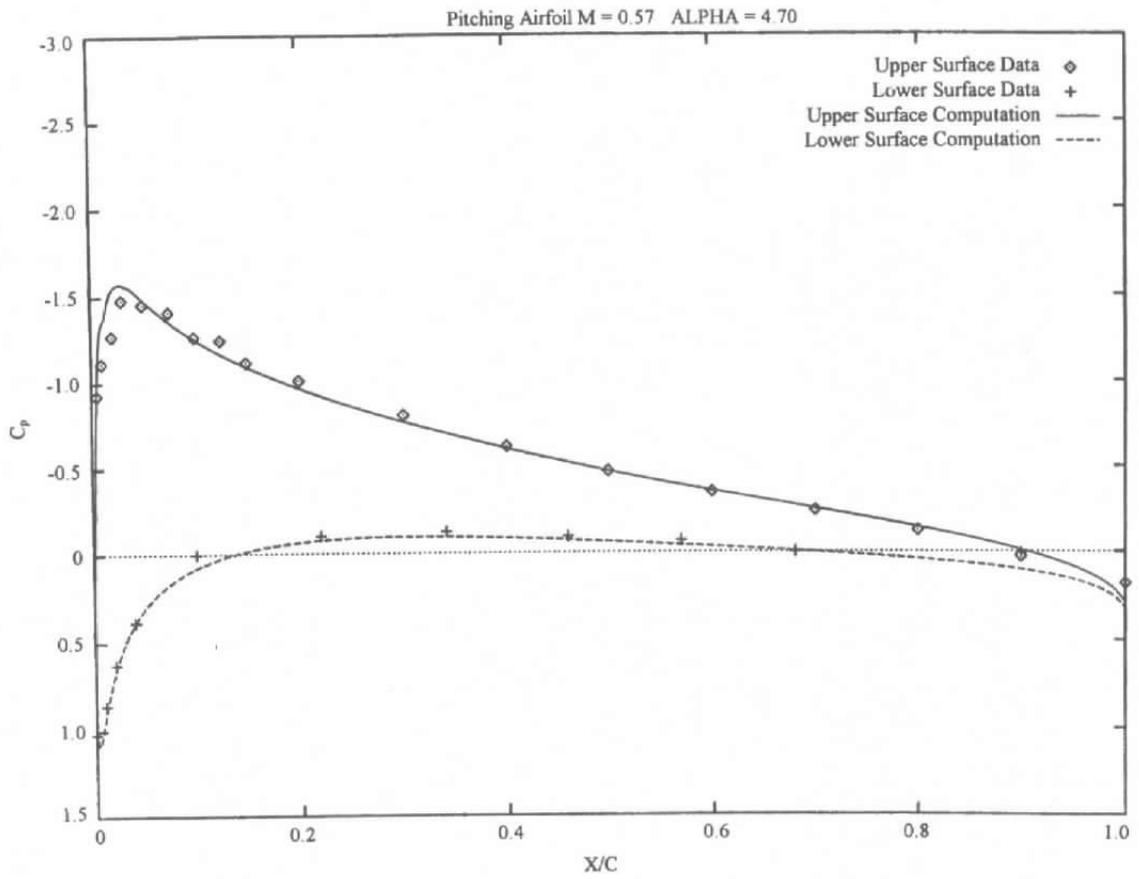


d.  $\alpha = 2.96$   
**Figure 5. Continued.**

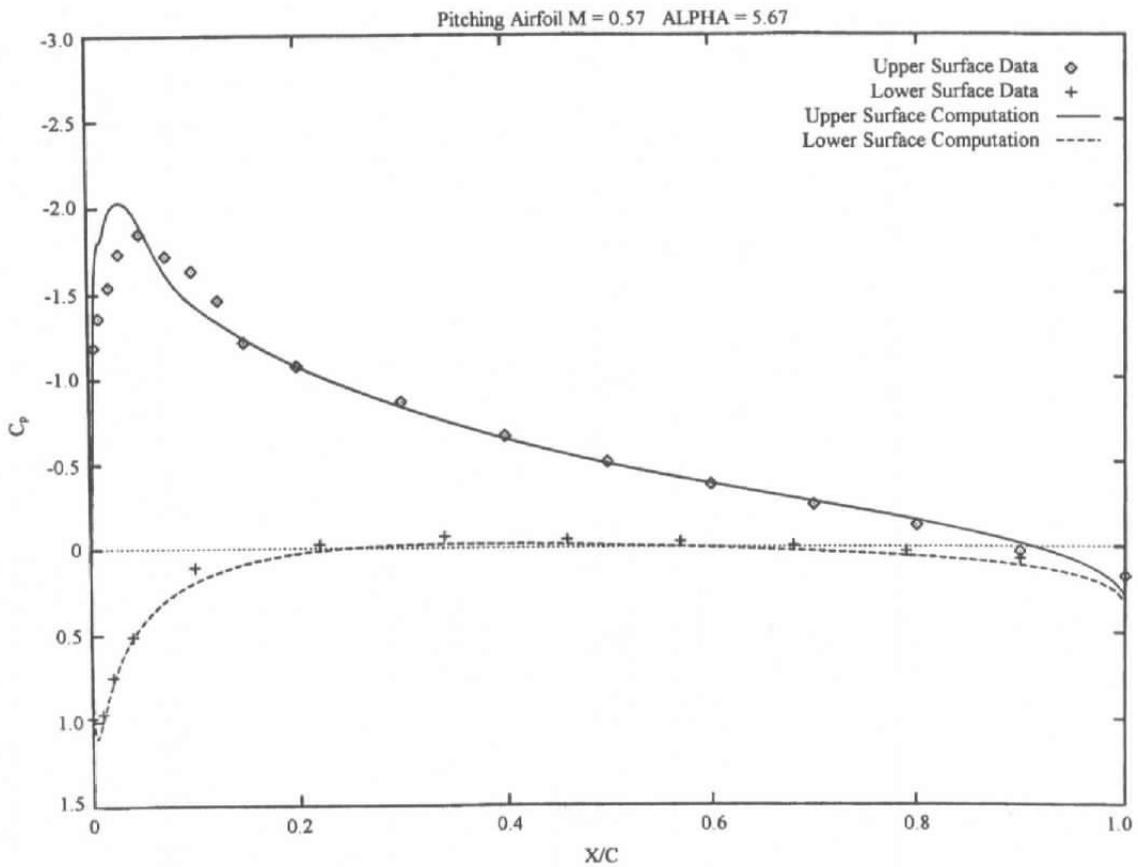


e.  $\alpha = 3.83$

Figure 5. Continued.

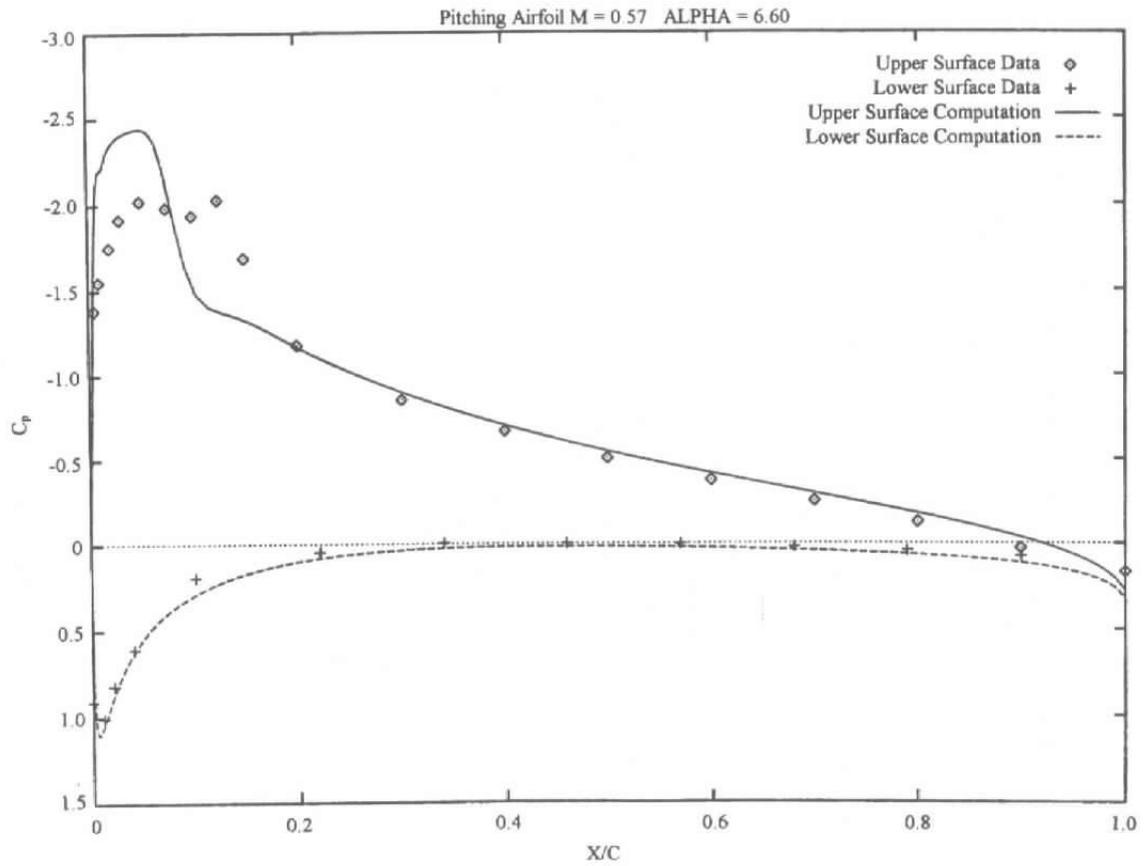


f.  $\alpha = 4.70$   
**Figure 5. Continued.**

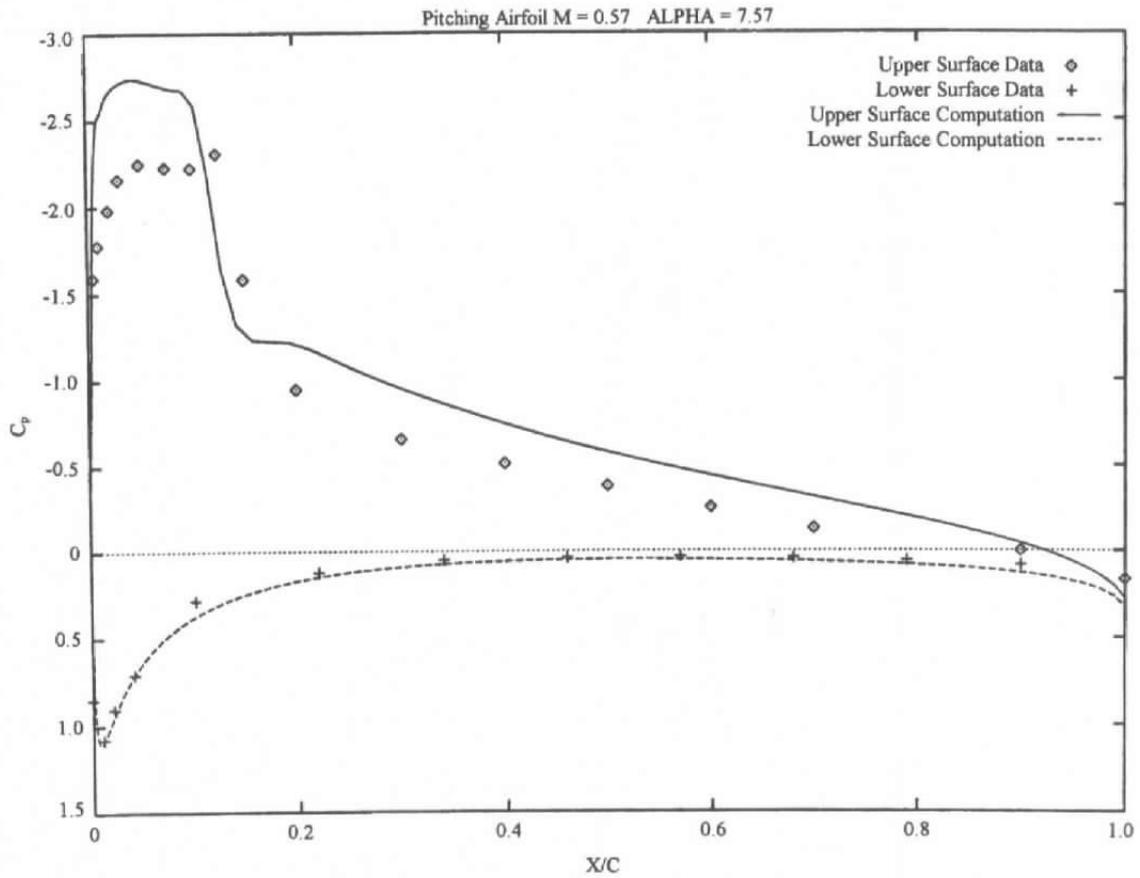


g.  $\alpha = 5.67$

Figure 5. Continued.

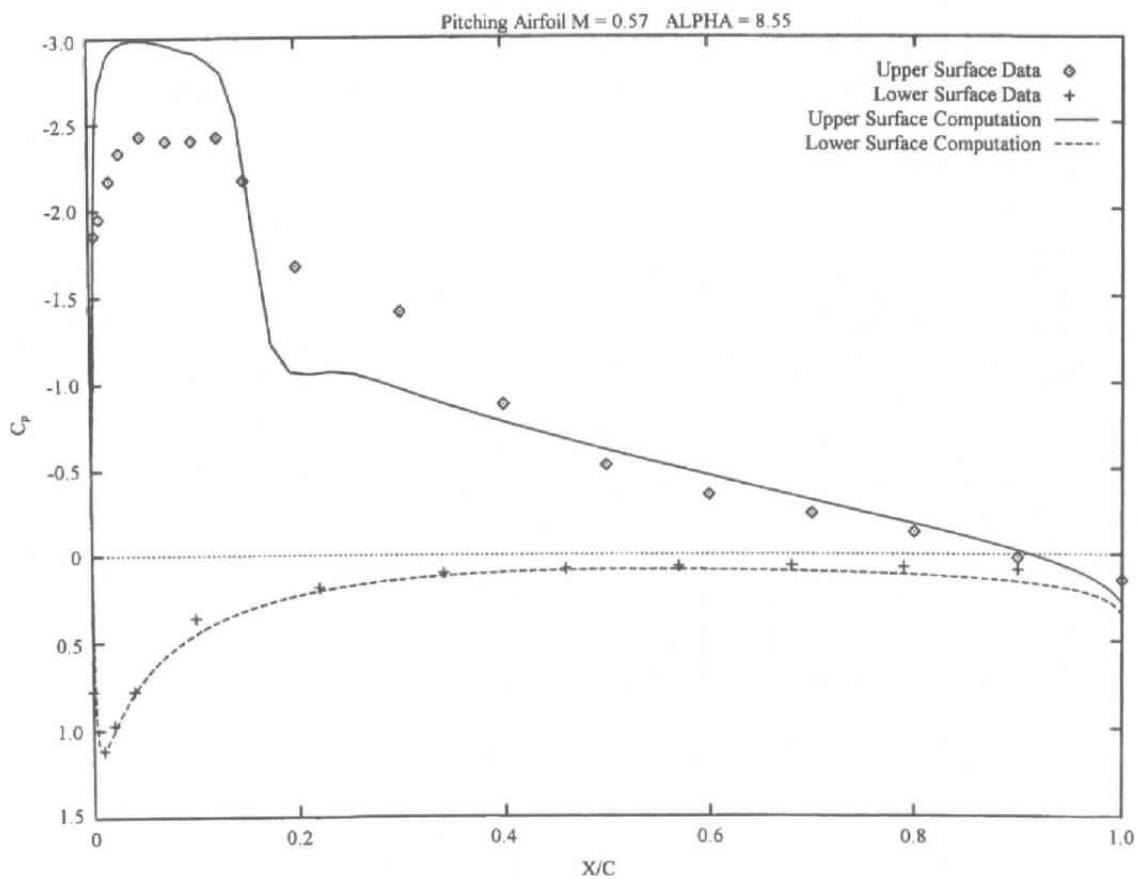


h.  $\alpha = 6.60$   
**Figure 5. Continued.**

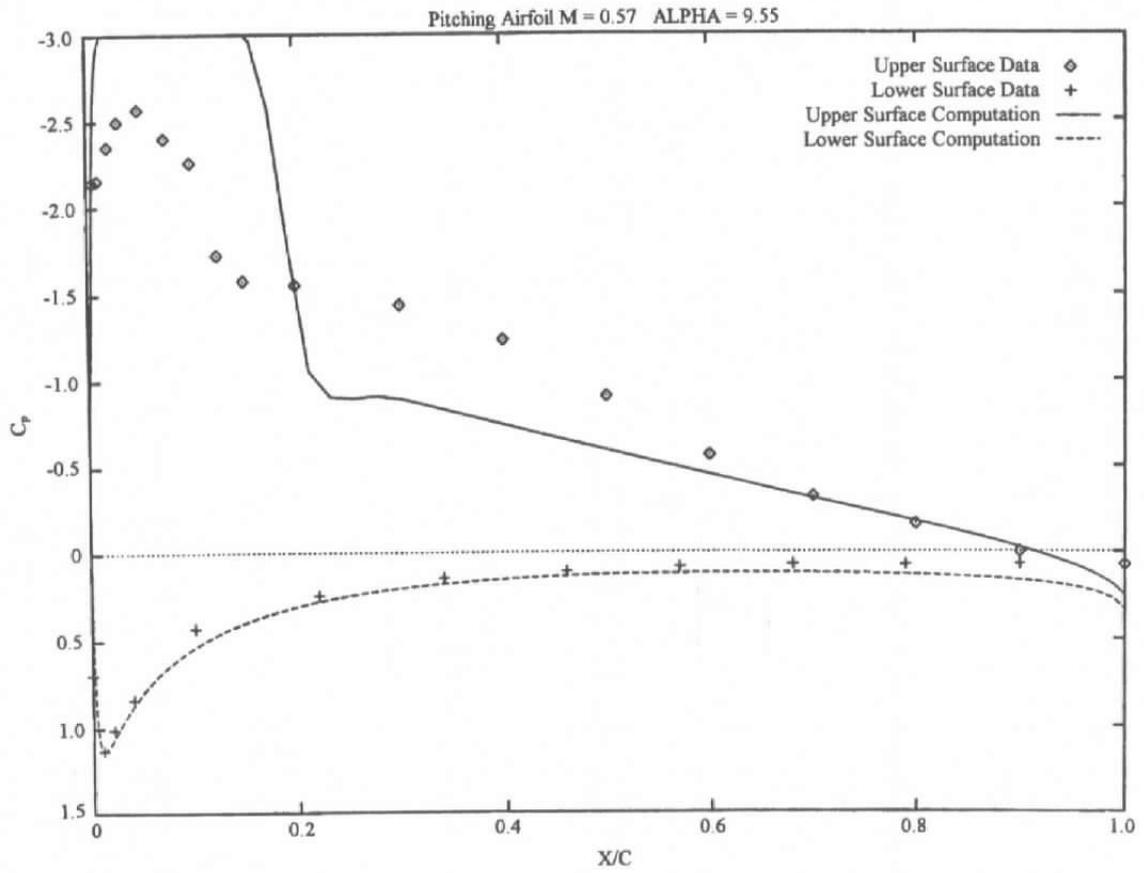


i.  $\alpha = 7.57$

Figure 5. Continued.

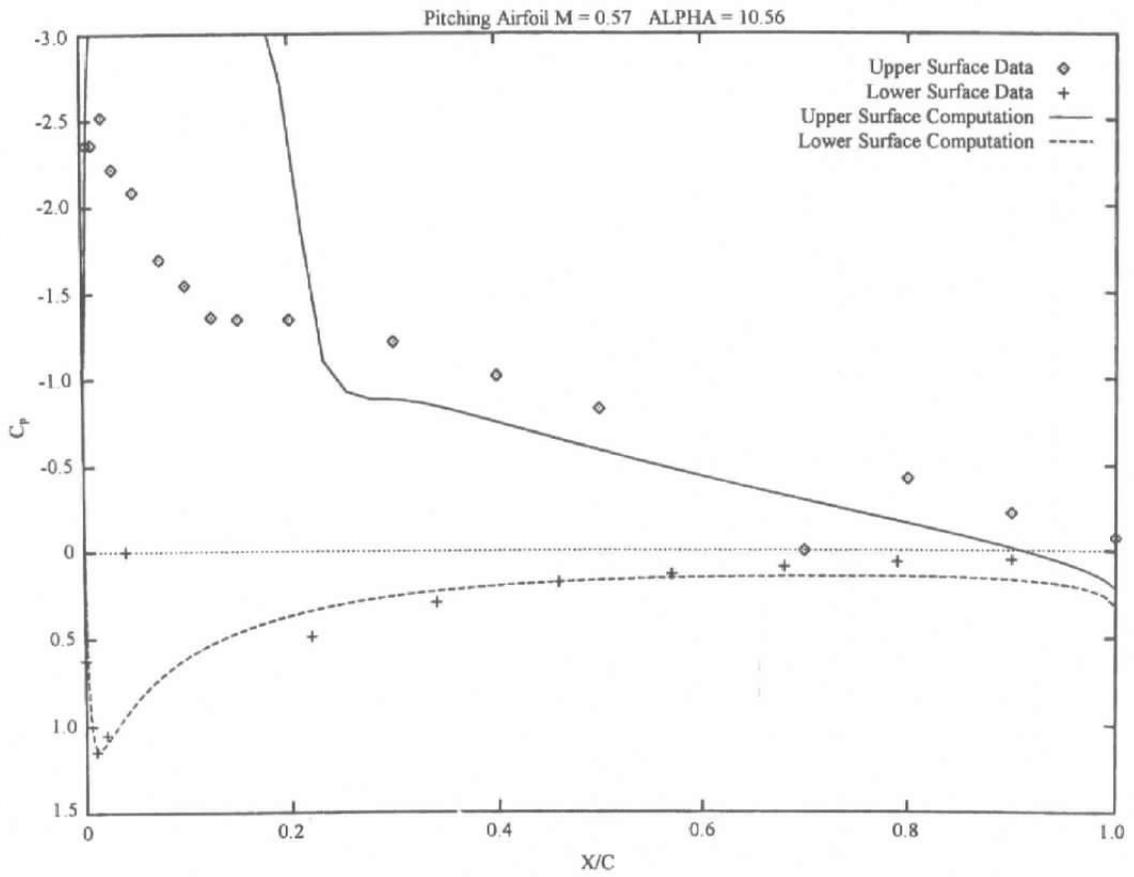


j.  $\alpha = 8.55$   
**Figure 5. Continued.**

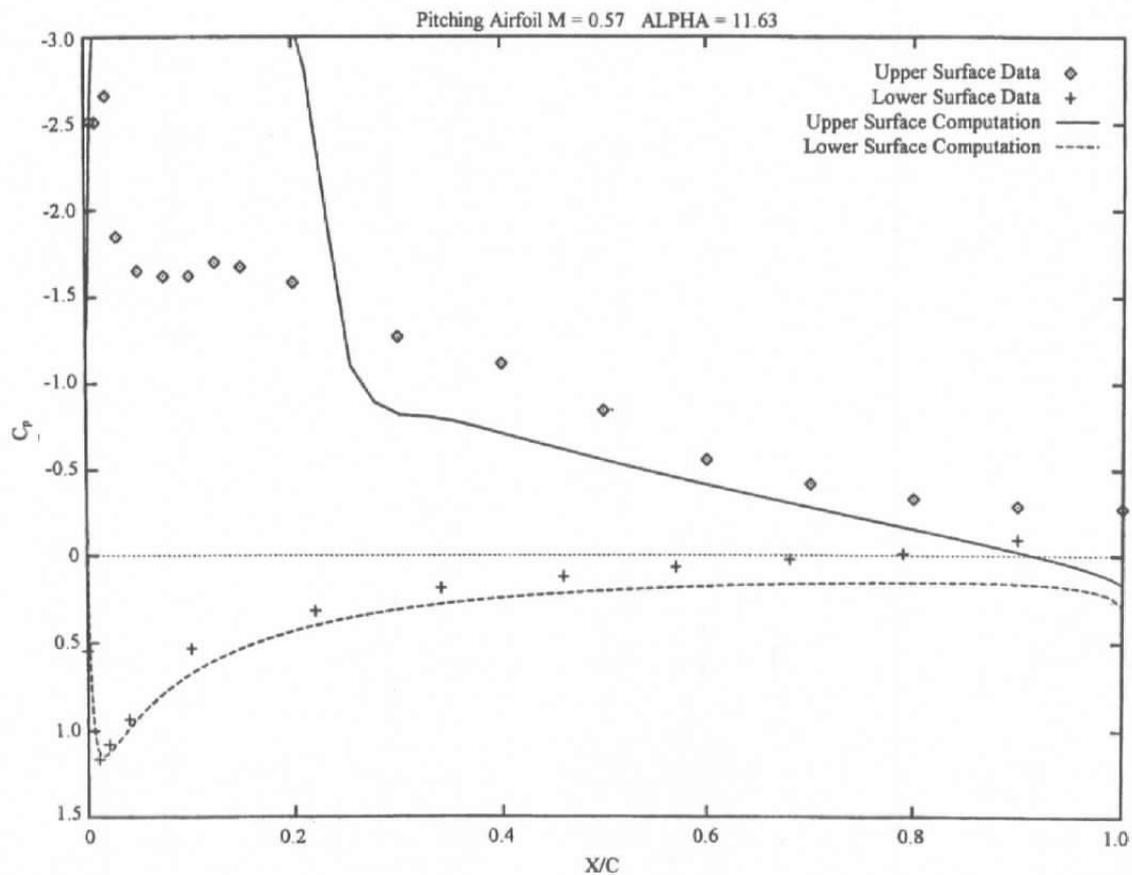


k.  $\alpha = 9.55$

Figure 5. Continued.

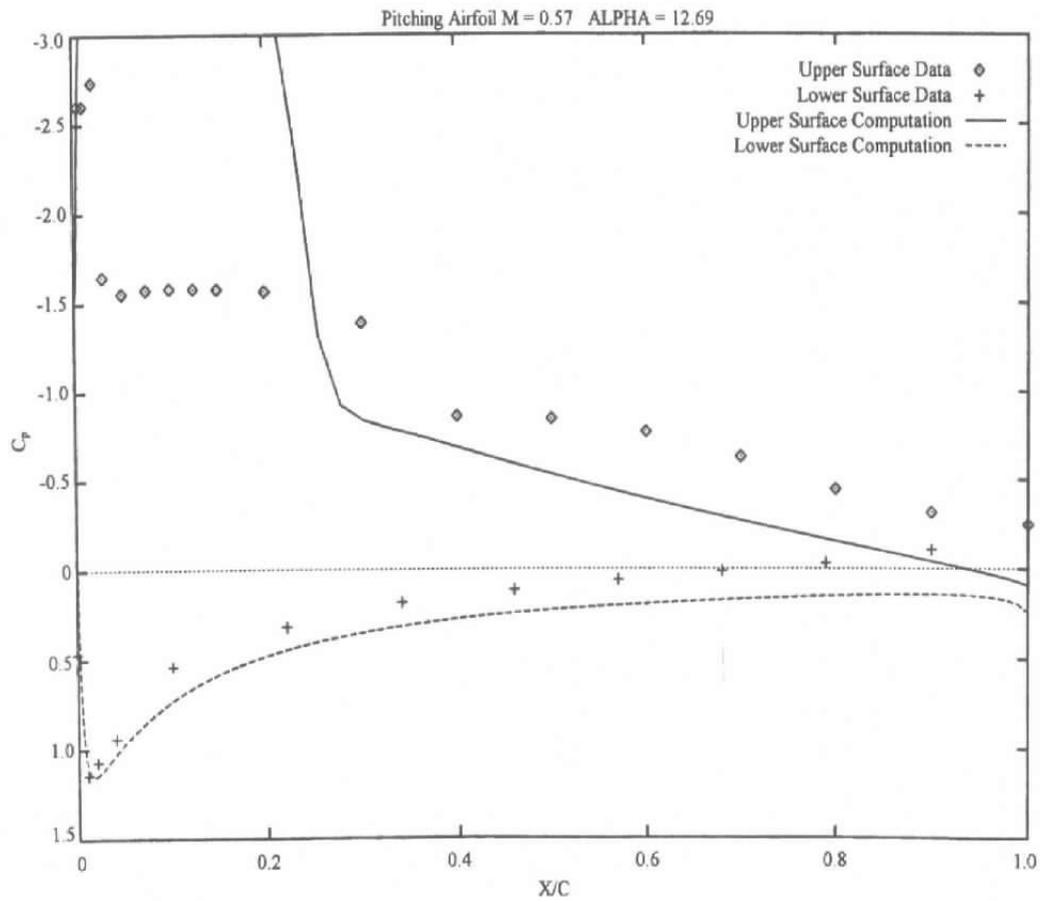


1.  $\alpha = 10.56$   
**Figure 5. Continued.**

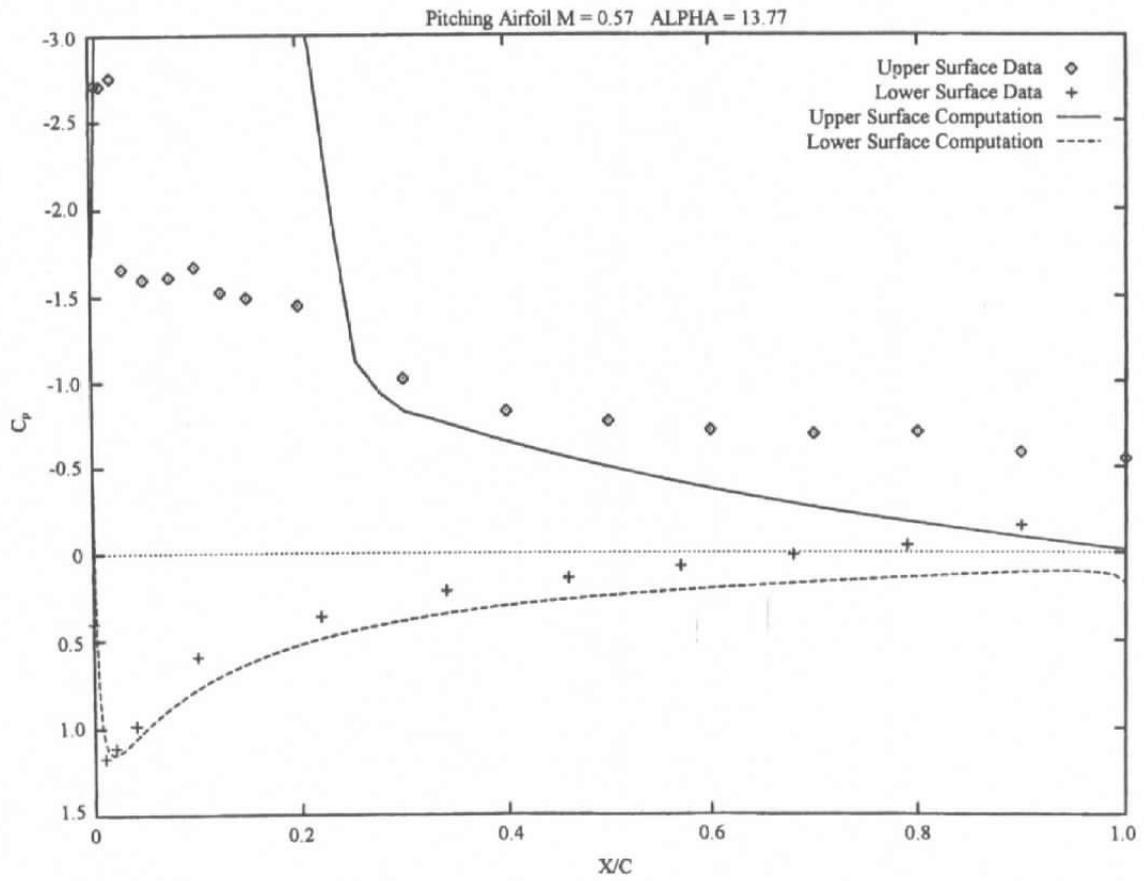


m.  $\alpha = 11.63$

Figure 5. Continued.

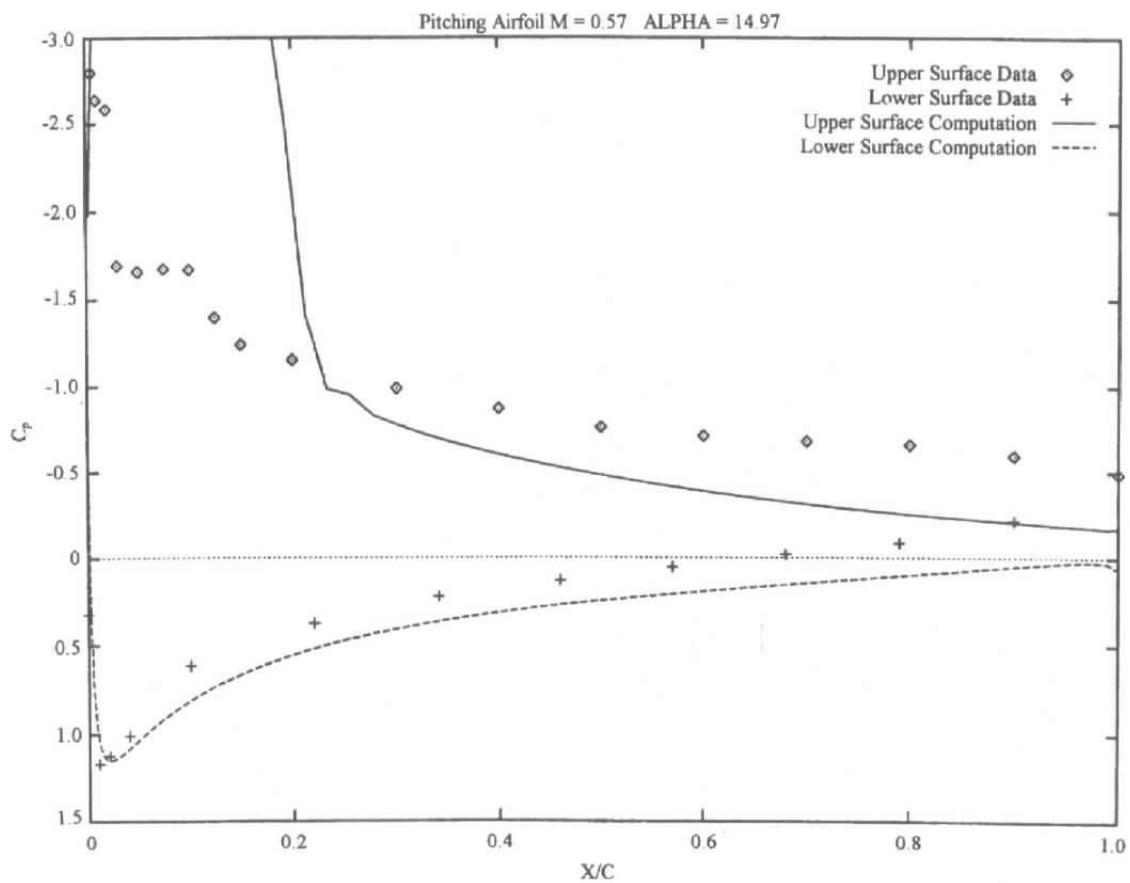


n.  $\alpha = 12.69$   
**Figure 5. Continued.**



$\alpha = 13.77$

Figure 5. Continued.



p.  $\alpha = 14.97$   
**Figure 5. Concluded.**

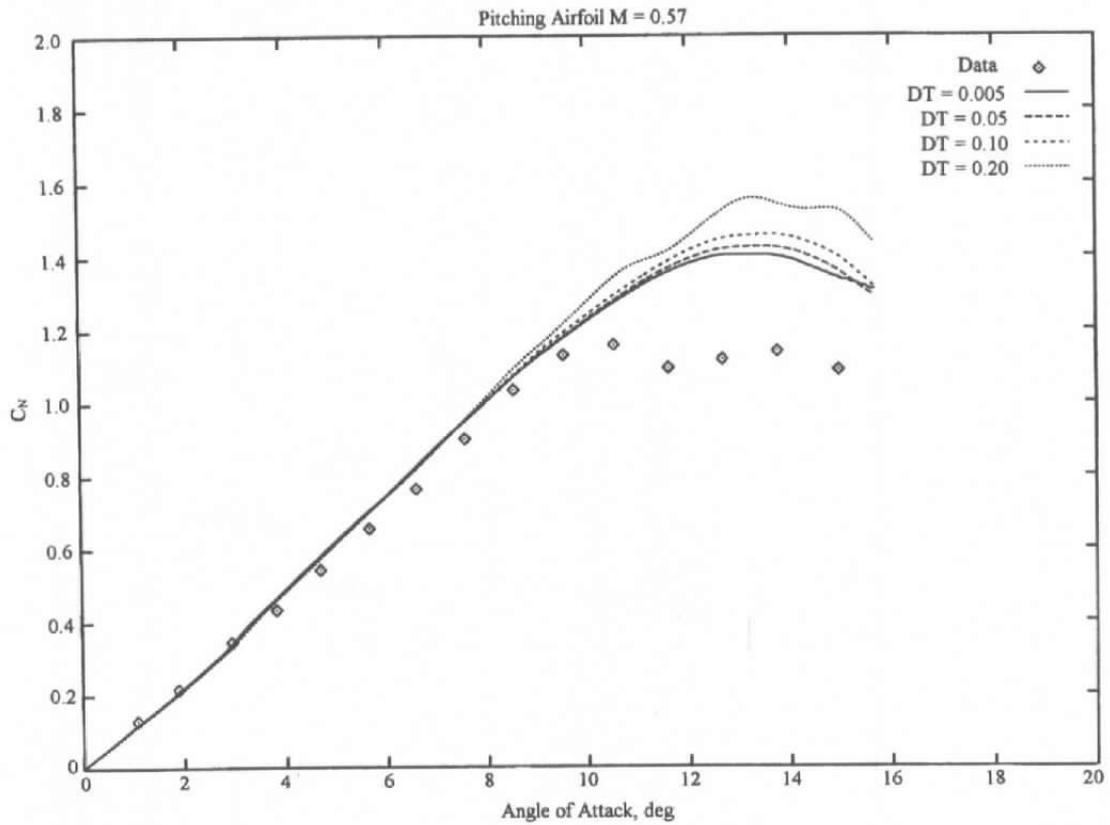


Figure 6. Comparison of  $C_N$  results for different DT's, Case B.

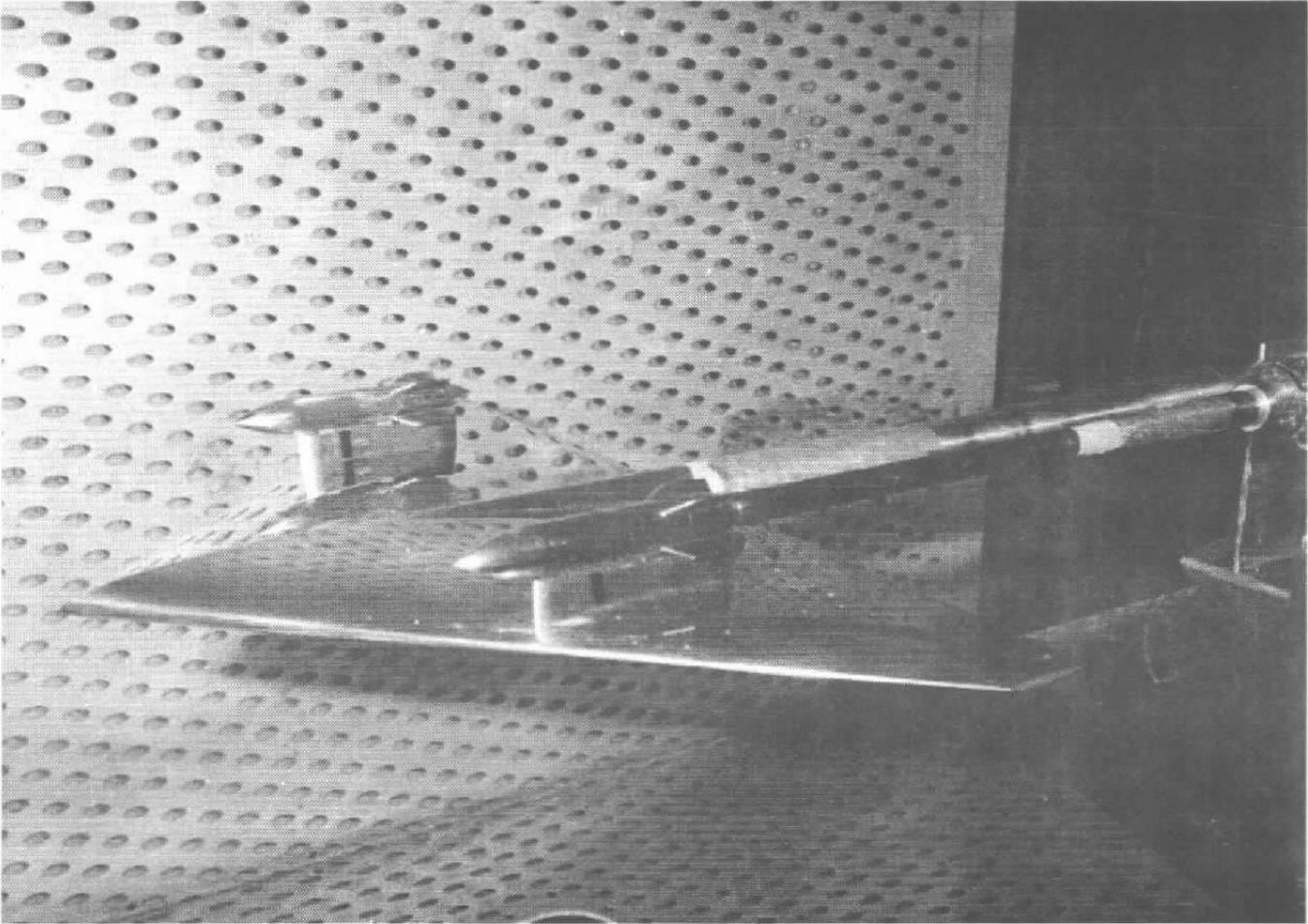


Figure 7. WL/AD test article for single-store trajectory.

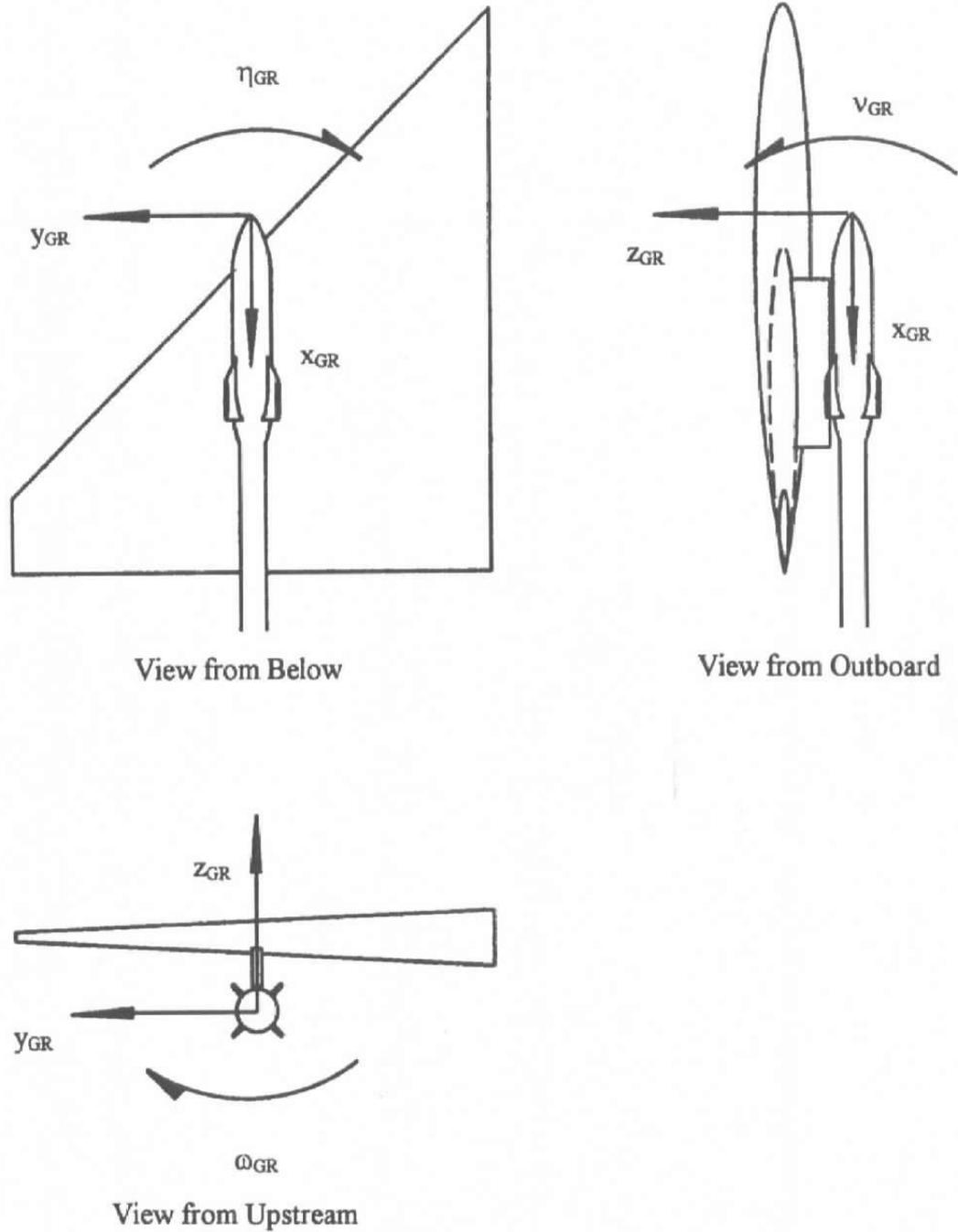
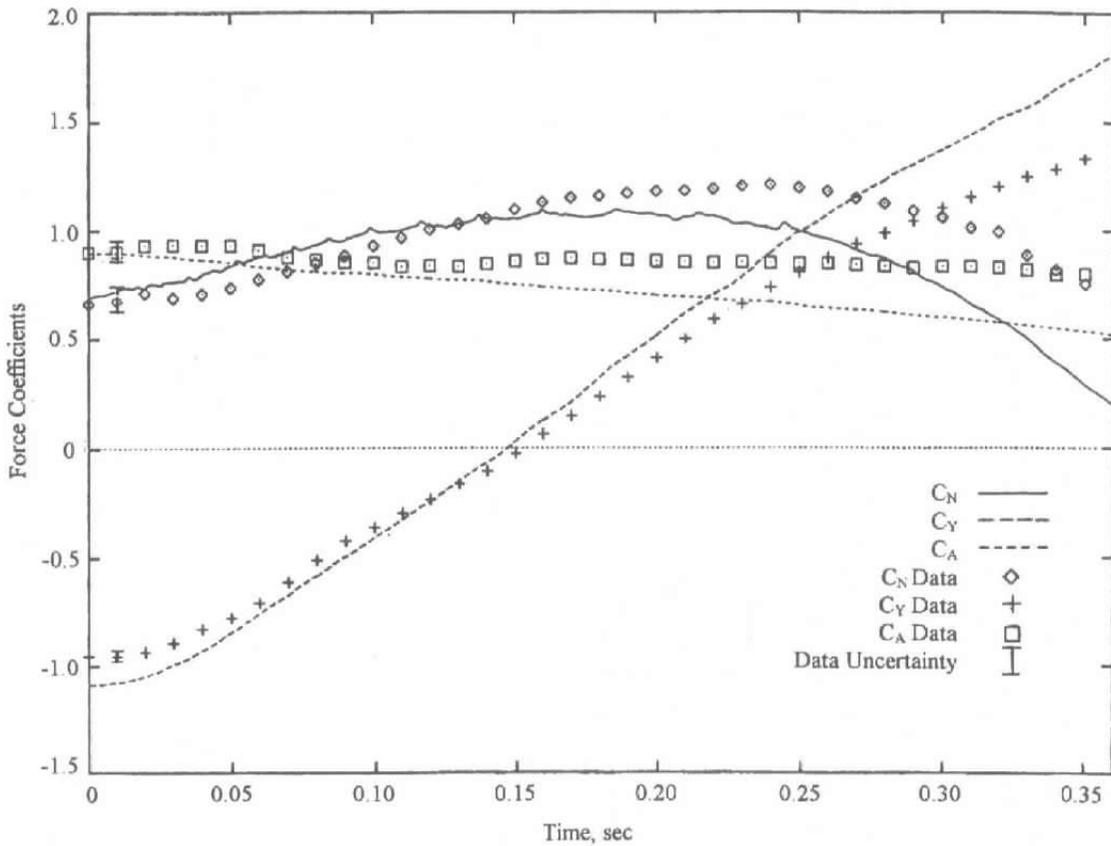
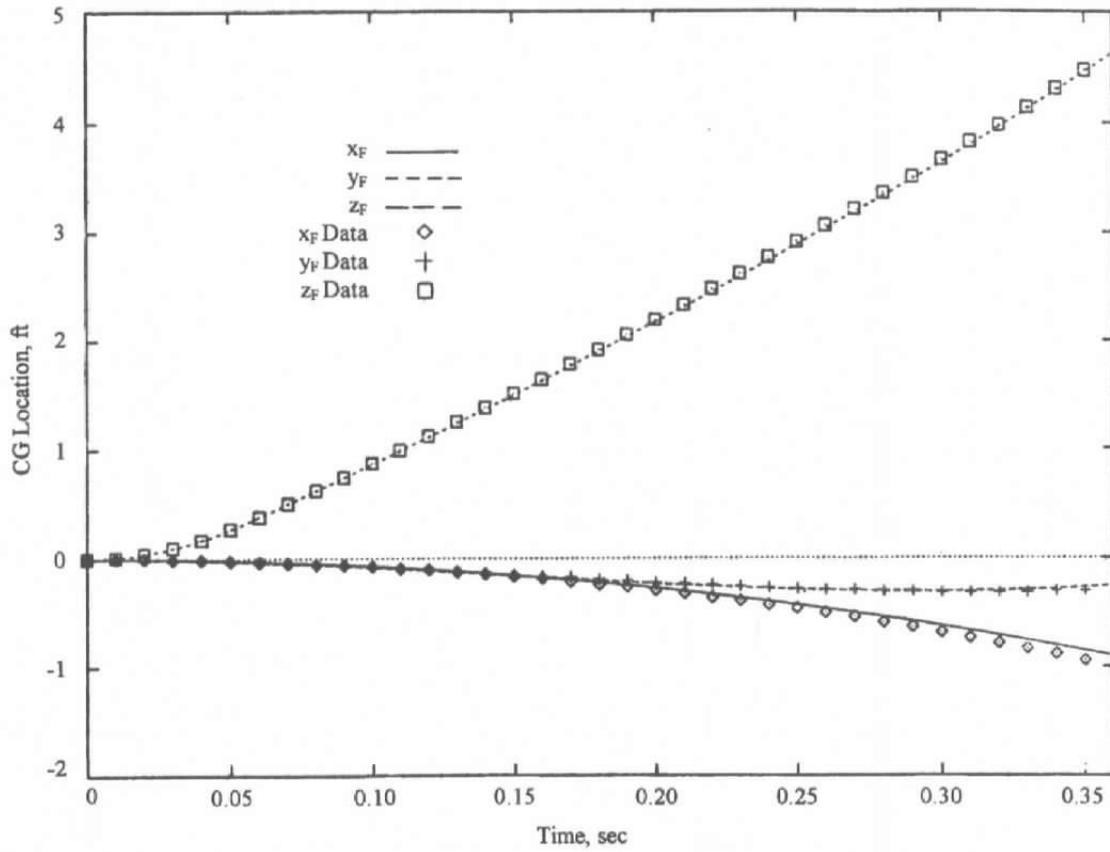


Figure 8. WL/AD single-store configuration geometry.

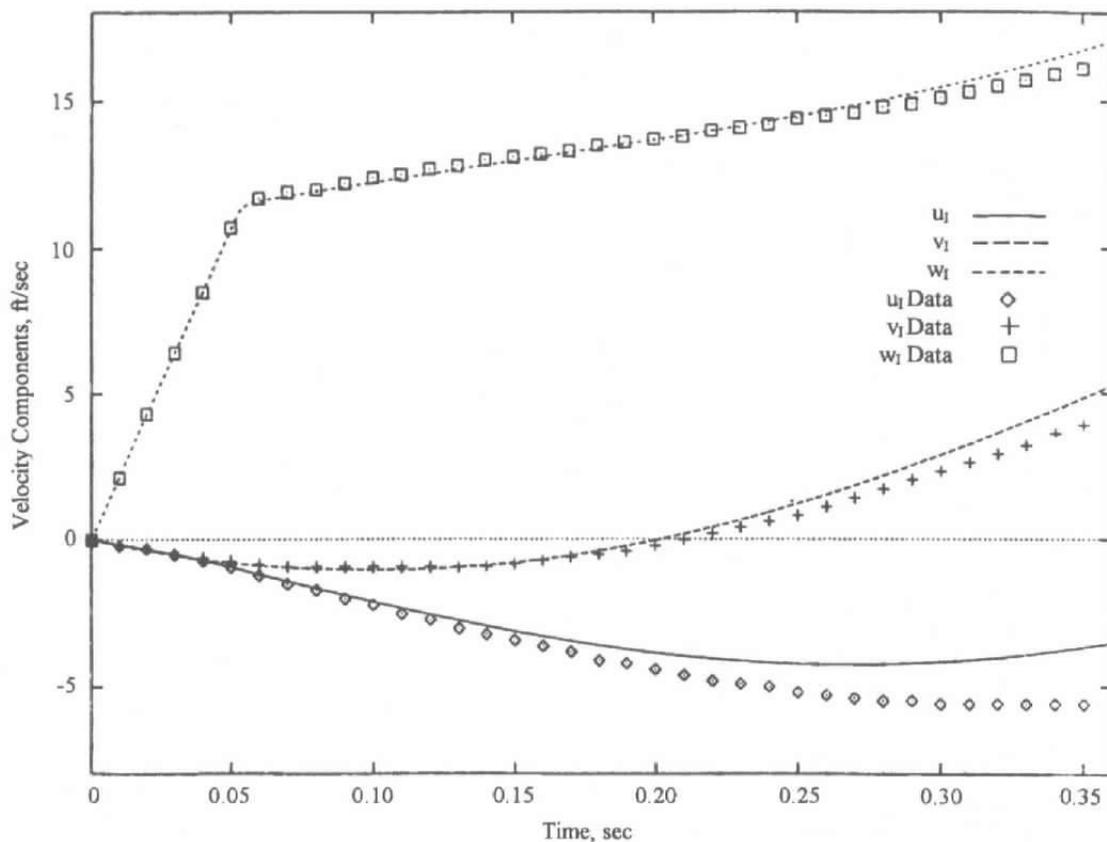


a. Force coefficients in body axis system

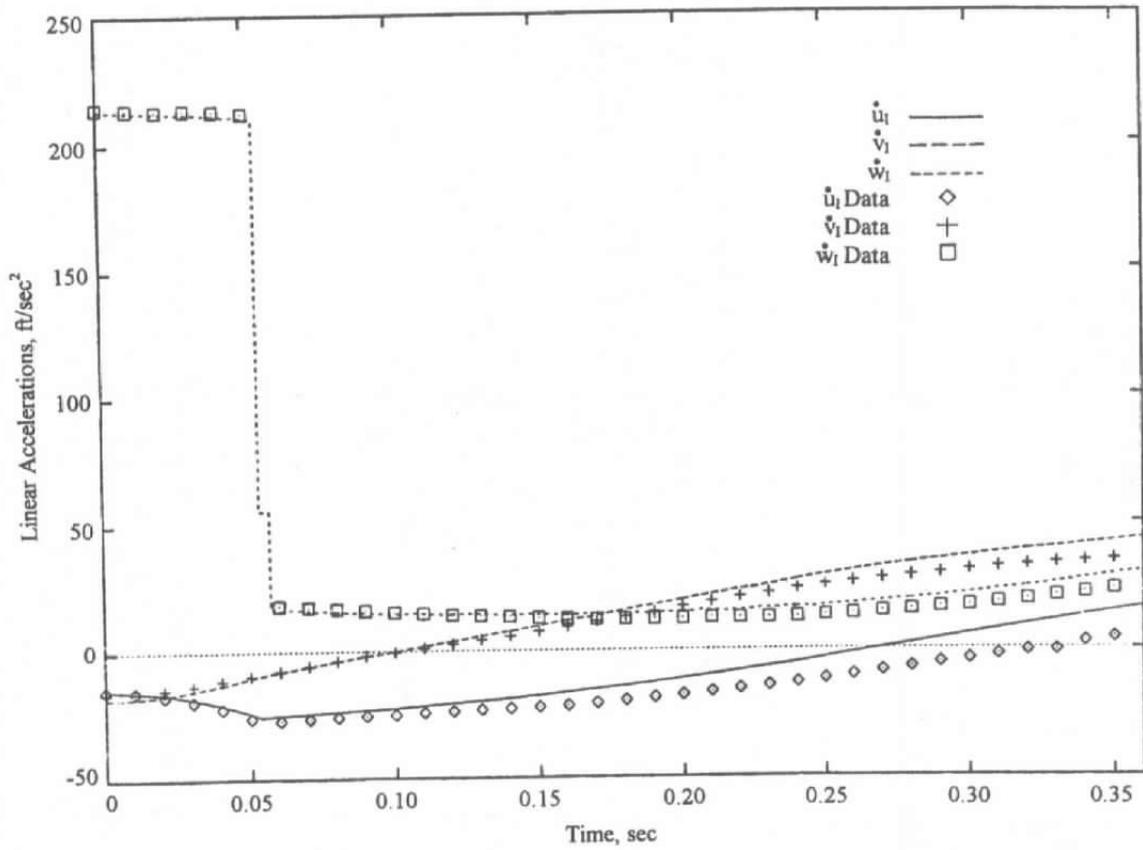
Figure 9. Comparisons of computed store trajectory translational time-history with CTS test data.



**b. Translation of center of gravity in flight axis system**  
**Figure 9. Continued.**

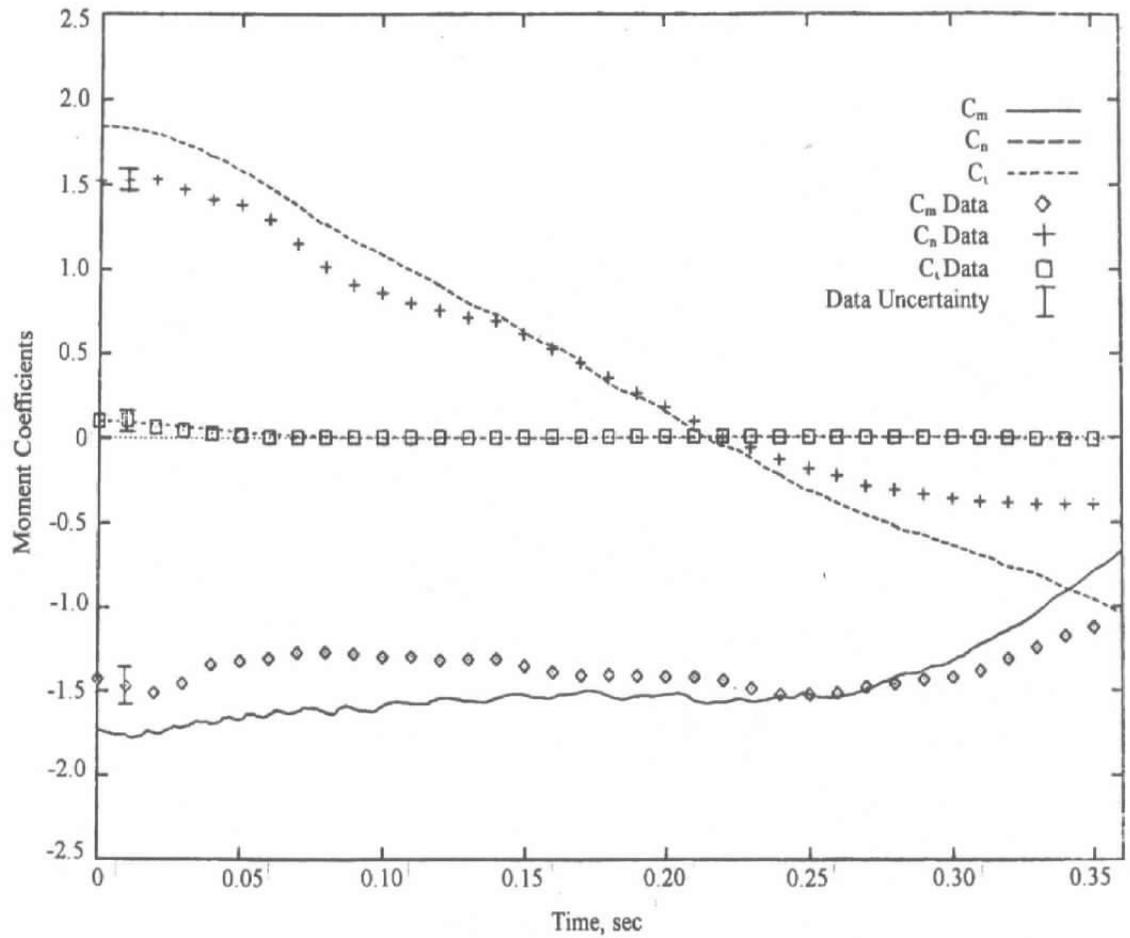


c. Translational velocity in inertial axis system  
 Figure 9. Continued.



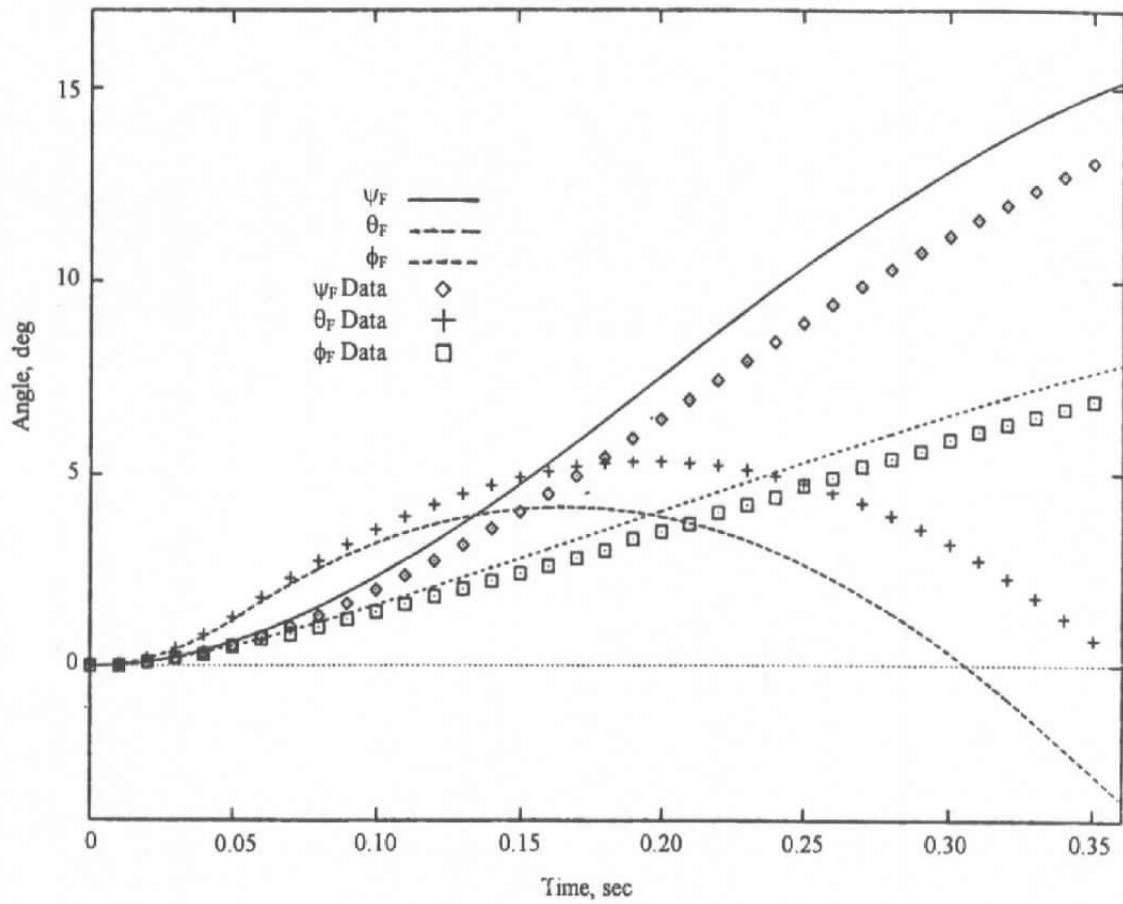
d. Translational acceleration in inertial axis system

Figure 9. Concluded.

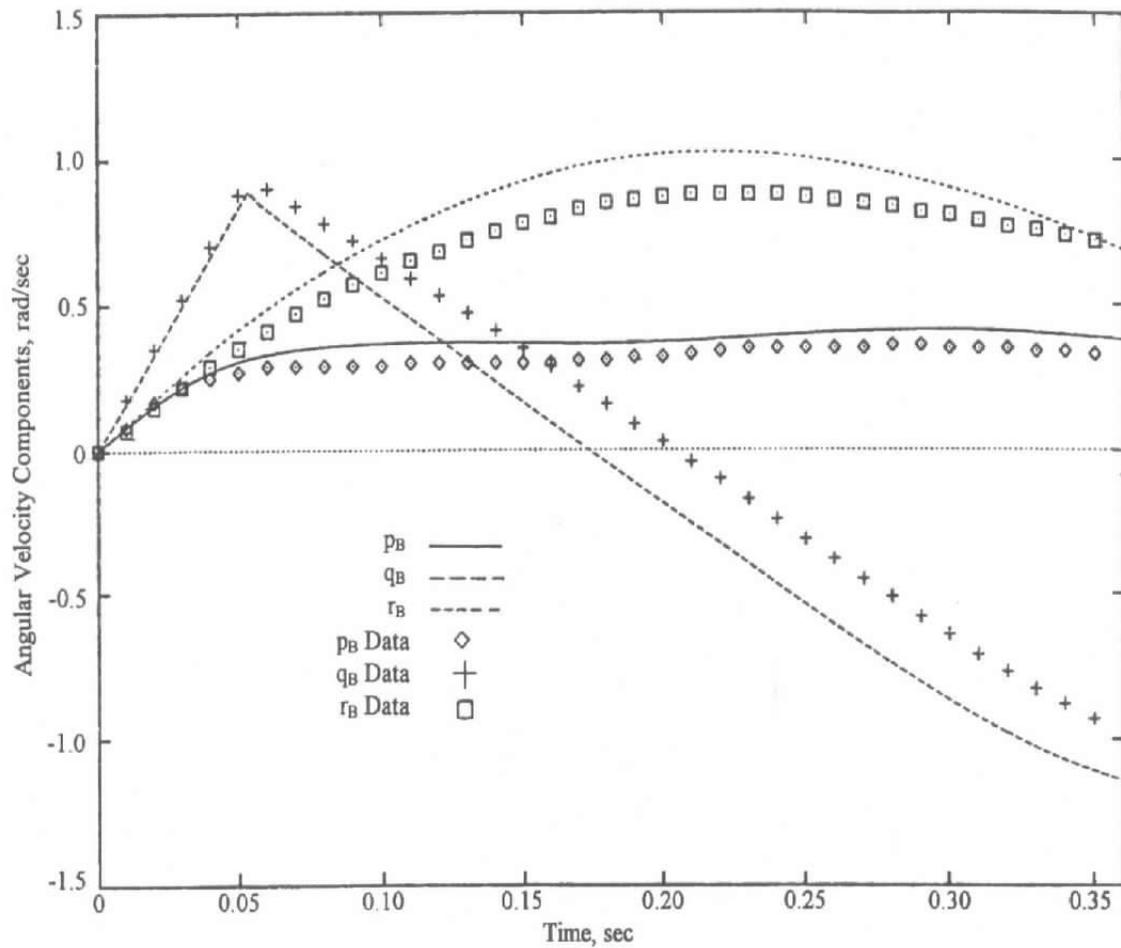


a. Moment coefficients

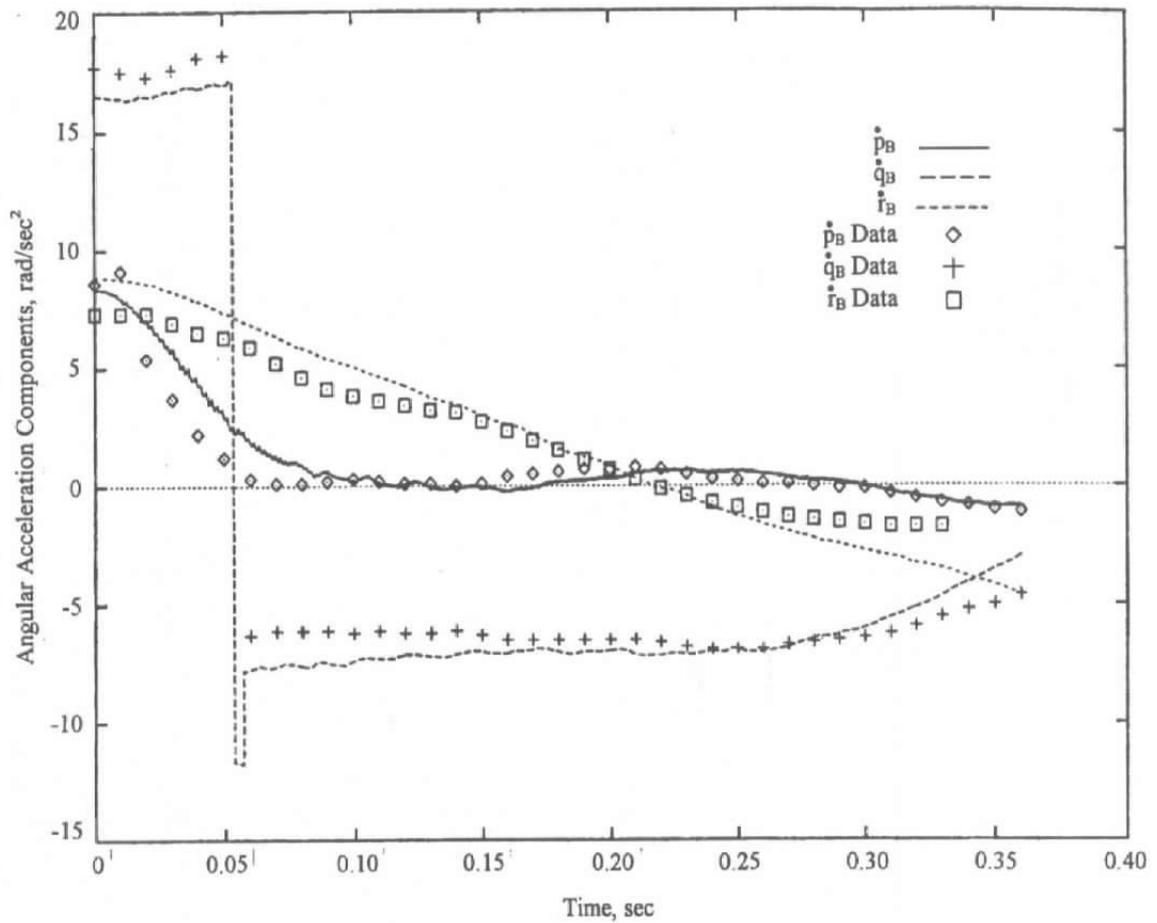
Figure 10. Comparisons of computed store trajectory rotational time-history with CTS test data.



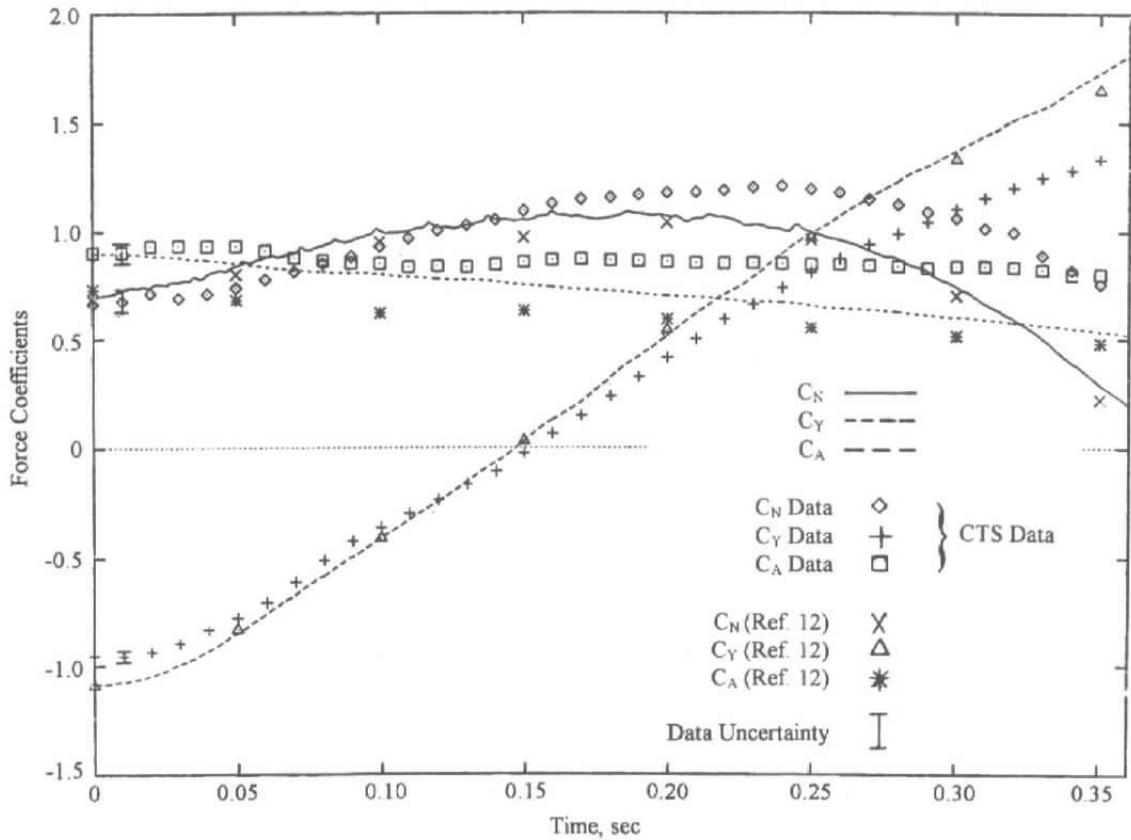
b. Rotational angles with respect to flight axis system  
Figure 10. Continued.



c. Rotational velocity in body axis system  
 Figure 10. Continued.

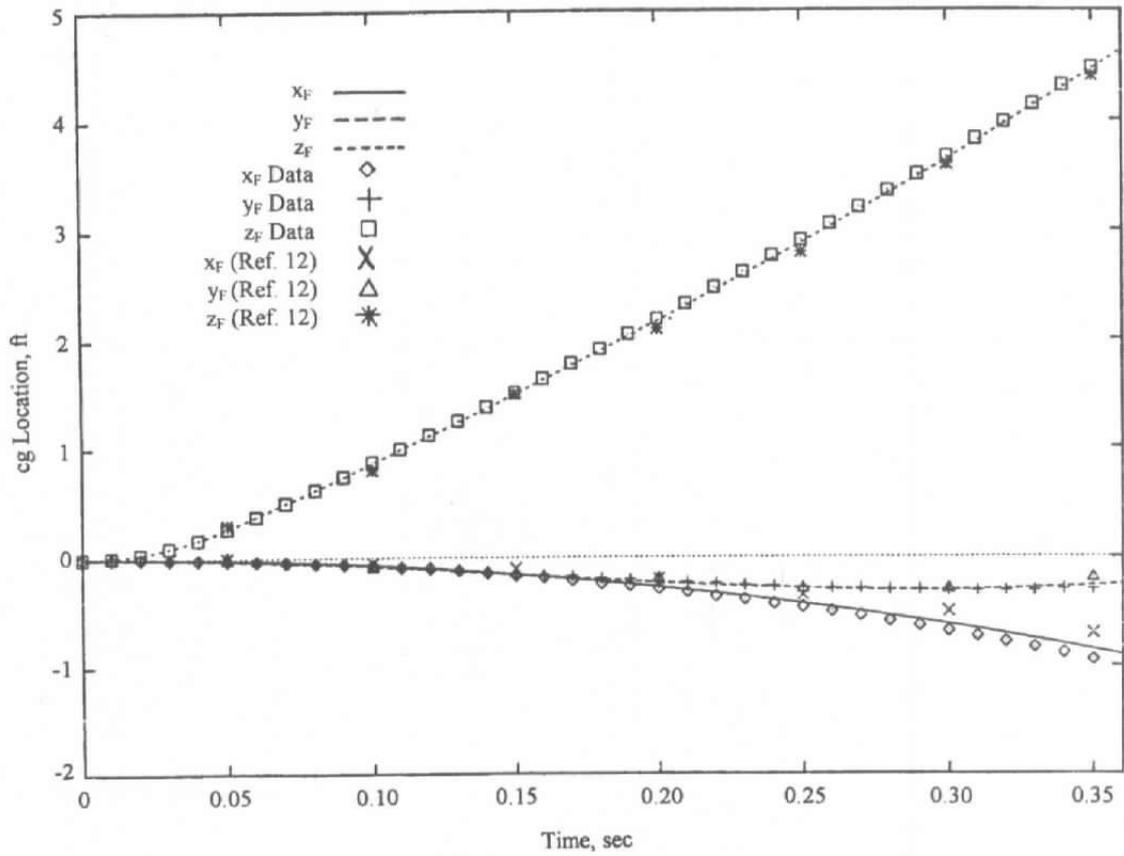


d. Rotational acceleration in body axis system  
Figure 10. Concluded.

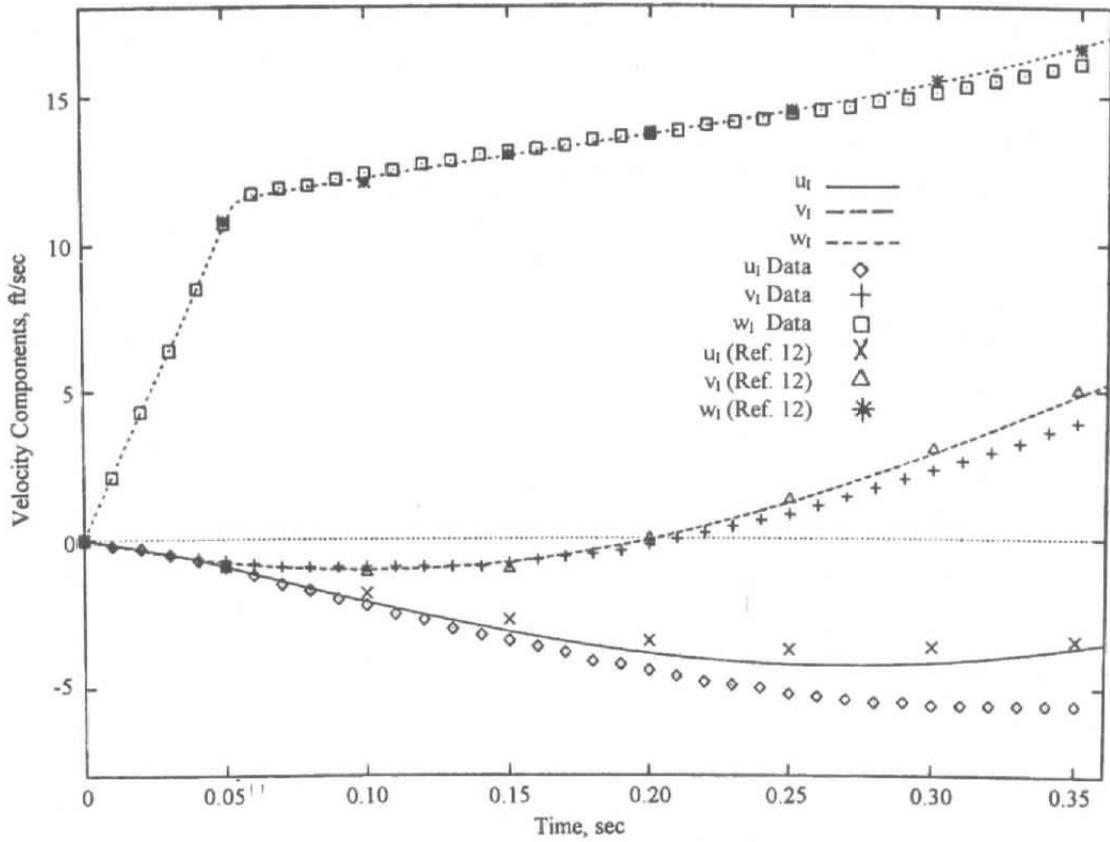


a. Force coefficients in body axis system

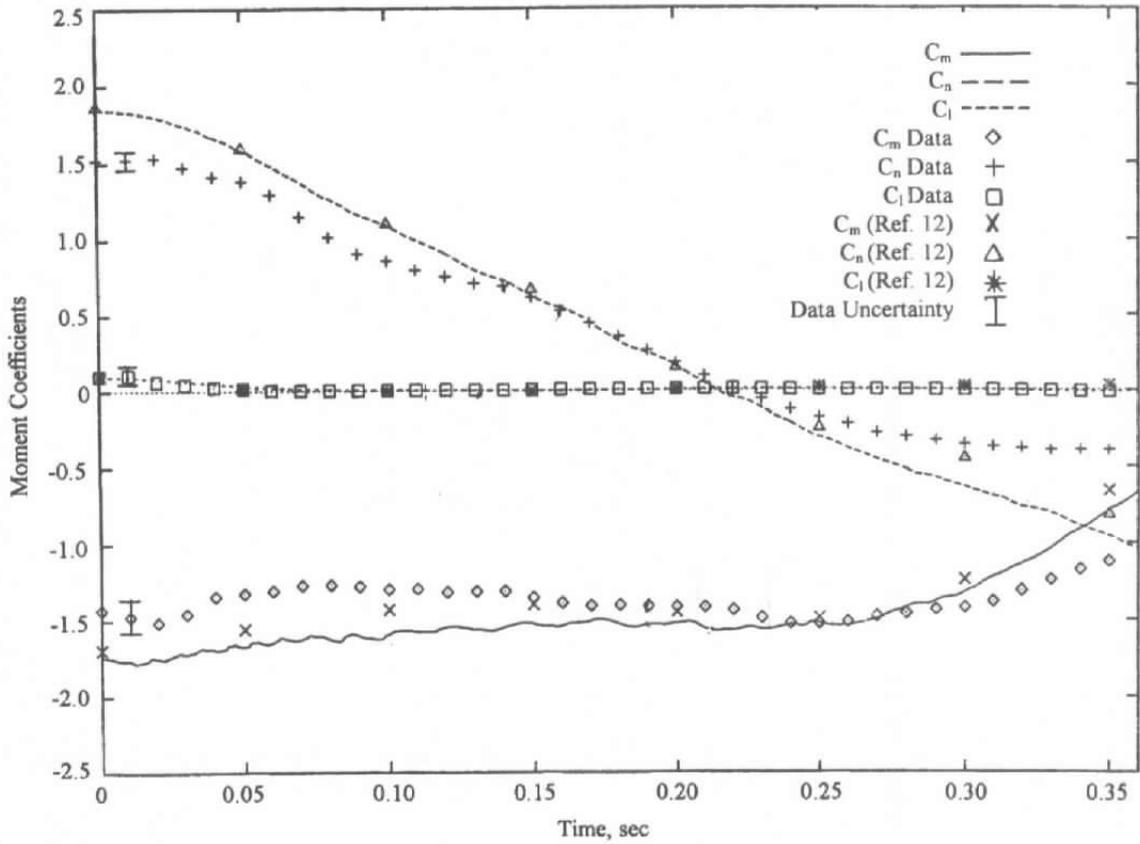
Figure 11. Comparisons of computed store trajectory translational time-history with CTS test data and WL/AD computations.



**b. Translation of center of gravity in flight axis system**  
**Figure 11. Continued.**

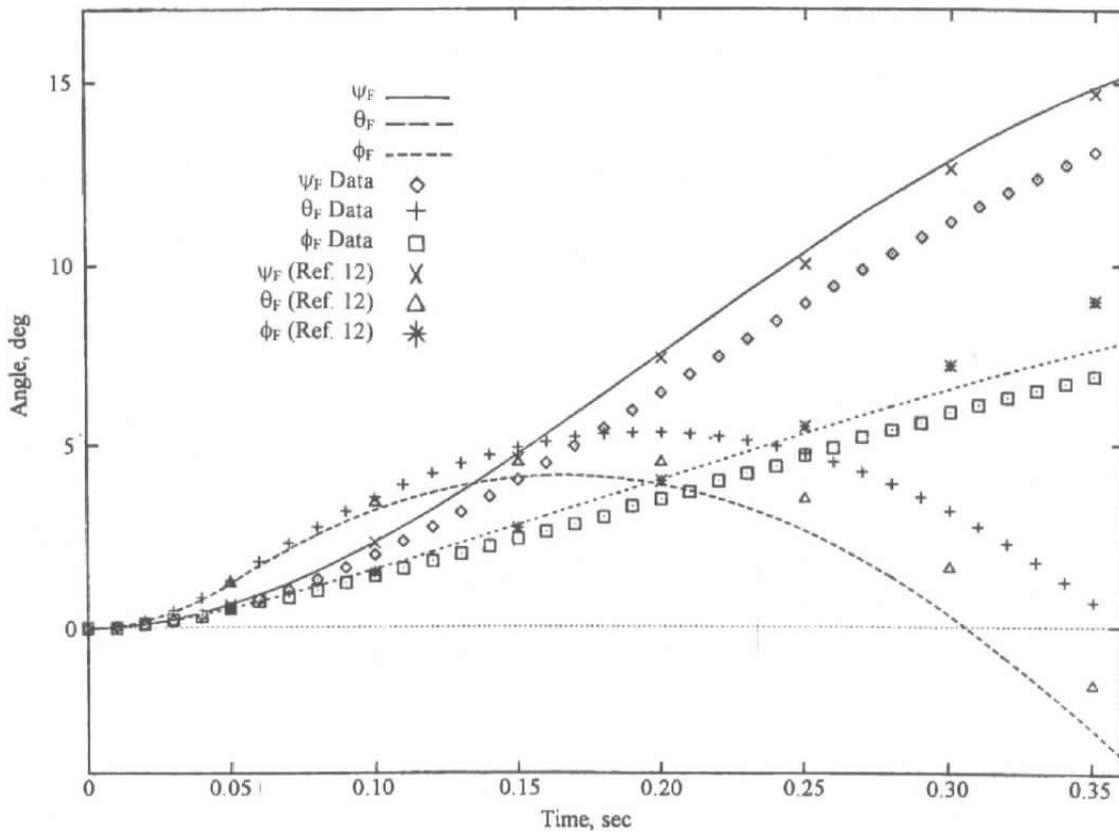


c. Translational velocity in inertial axis system  
 Figure 11. Concluded.

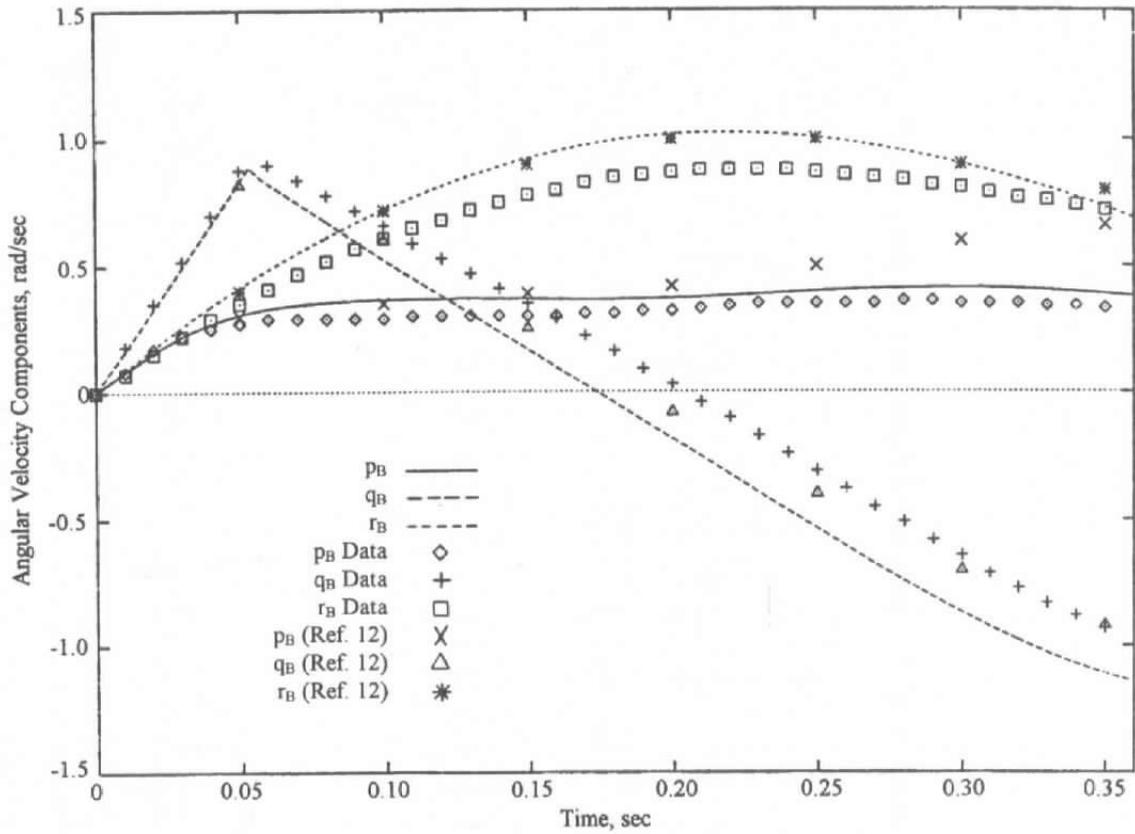


a. Moment coefficients in body axis system

Figure 12. Comparisons of computed store trajectory rotational



**b. Rotational angles with respect to flight axis system**  
**Figure 12. Continued.**



c. Rotational velocity in body axis system  
 Figure 12. Concluded.

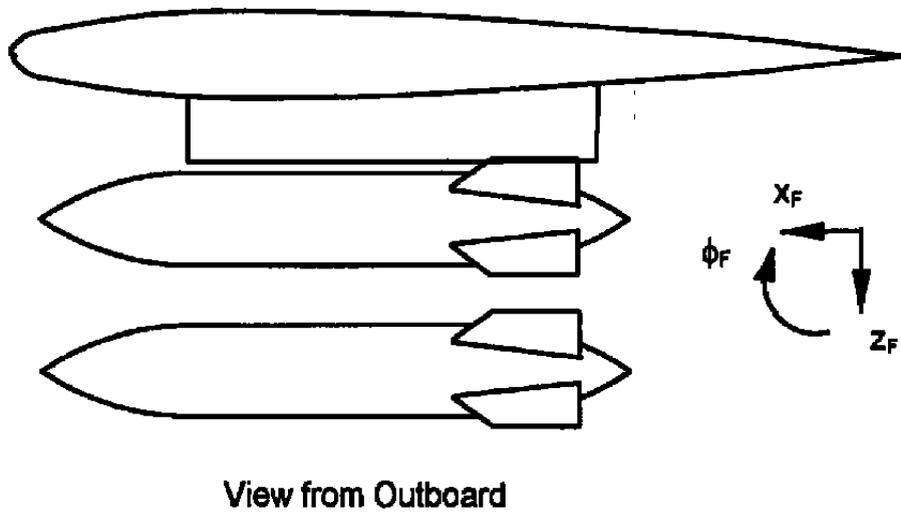
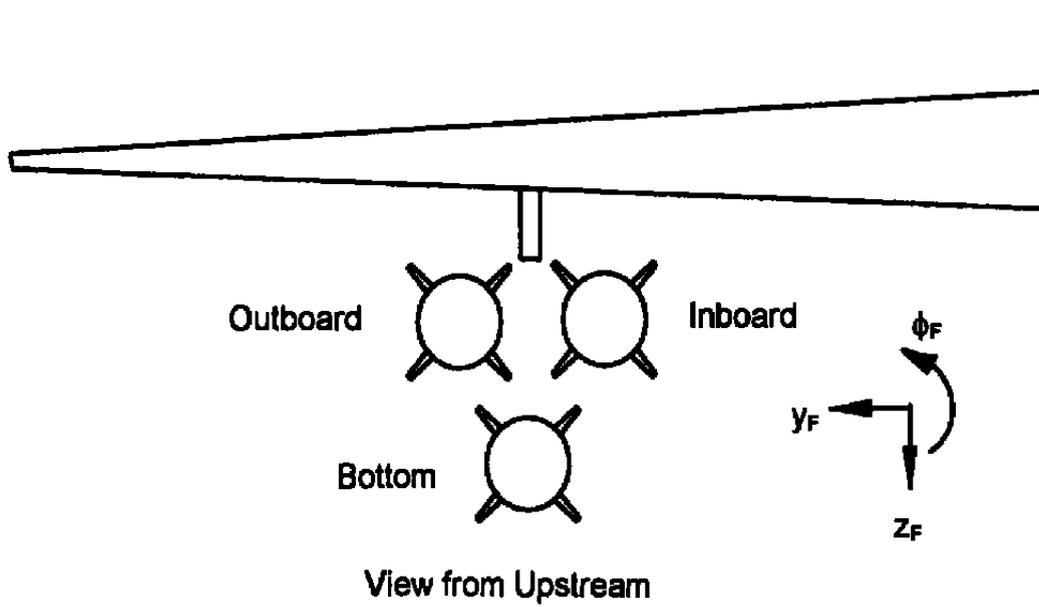
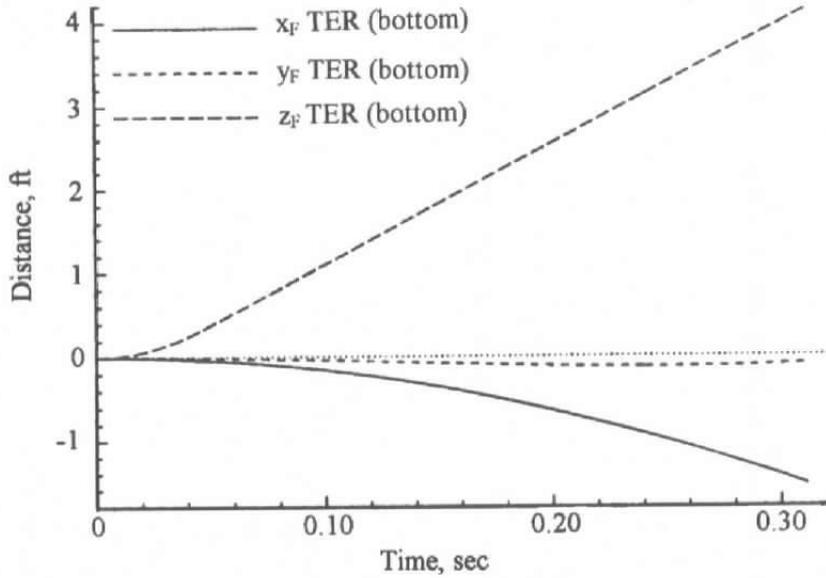
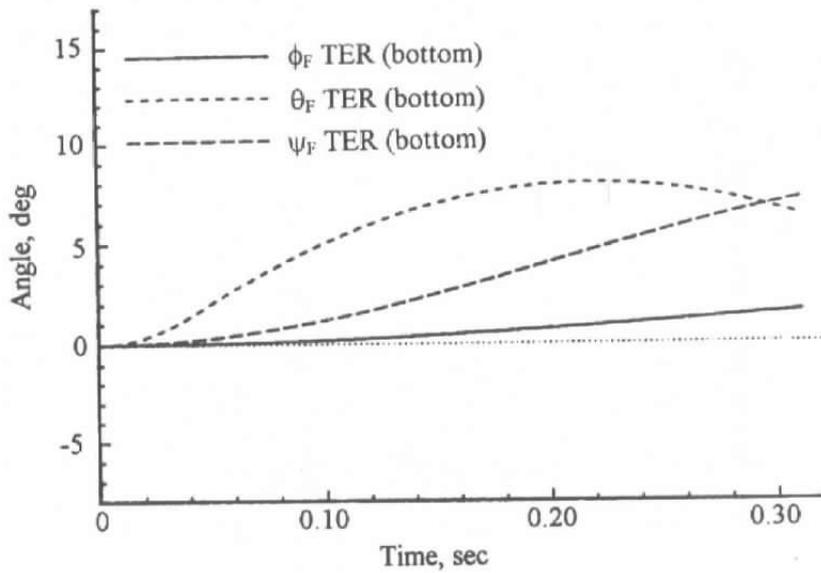


Figure 13. TER solution carriage position.

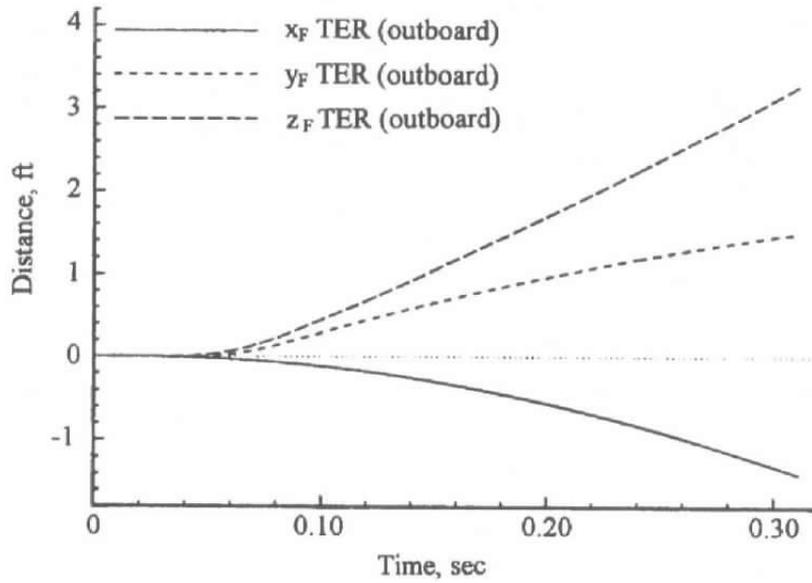


a. Translation distance

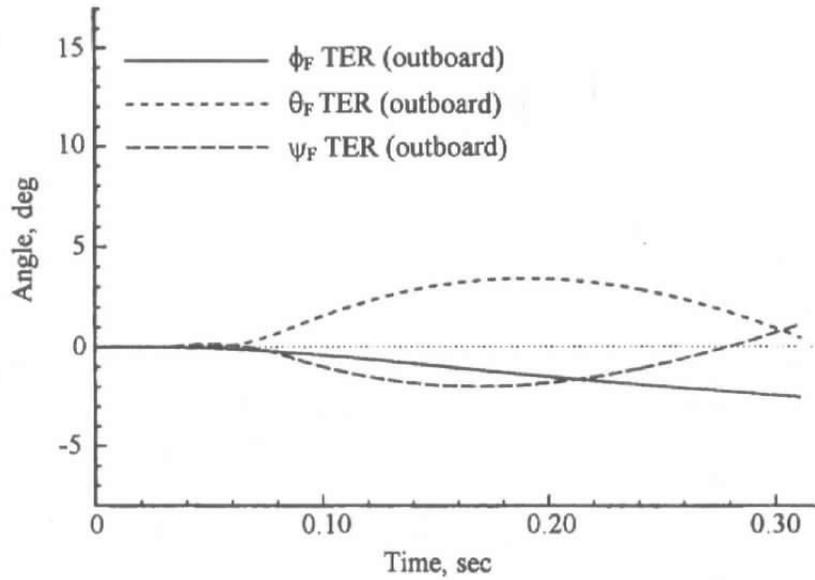


b. Rotation angle

Figure 14. Translational and rotational time history for TER configuration bottom store.

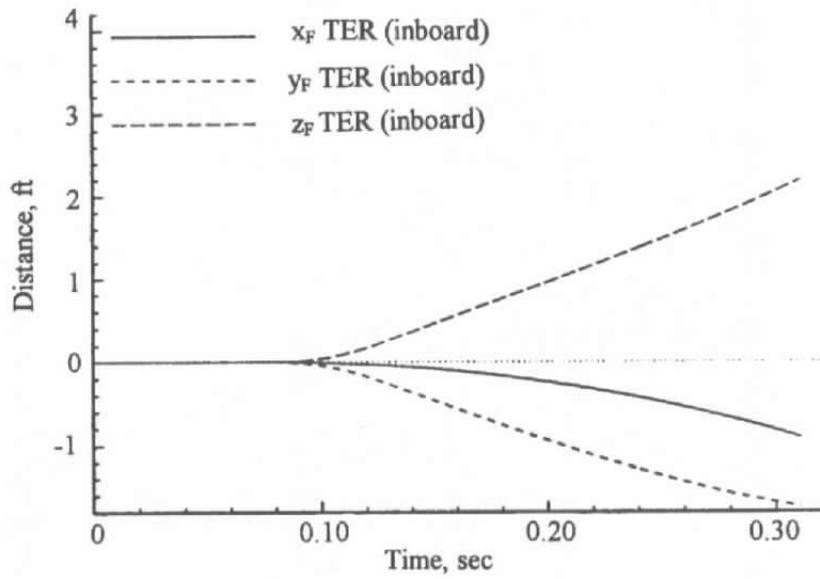


a. Translational distance

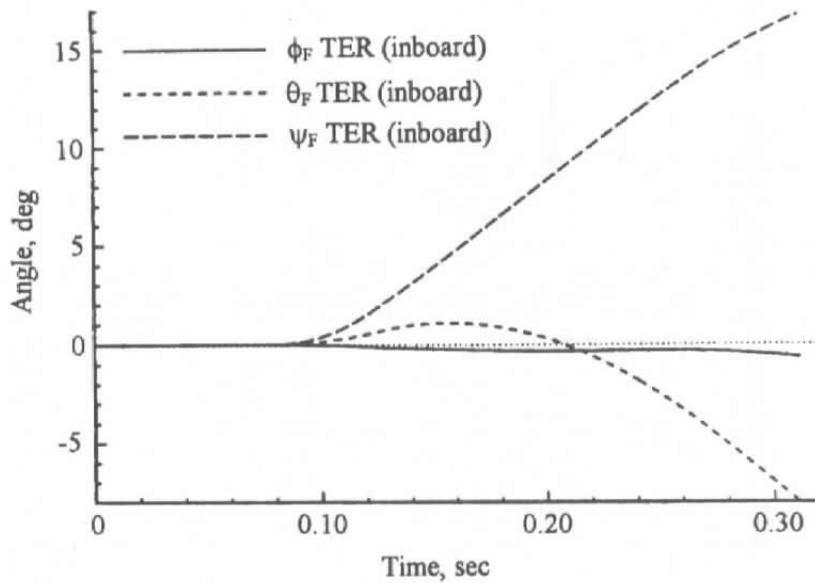


b. Rotation angle

Figure 15. Translational and rotational time history for TER configuration, outboard store.

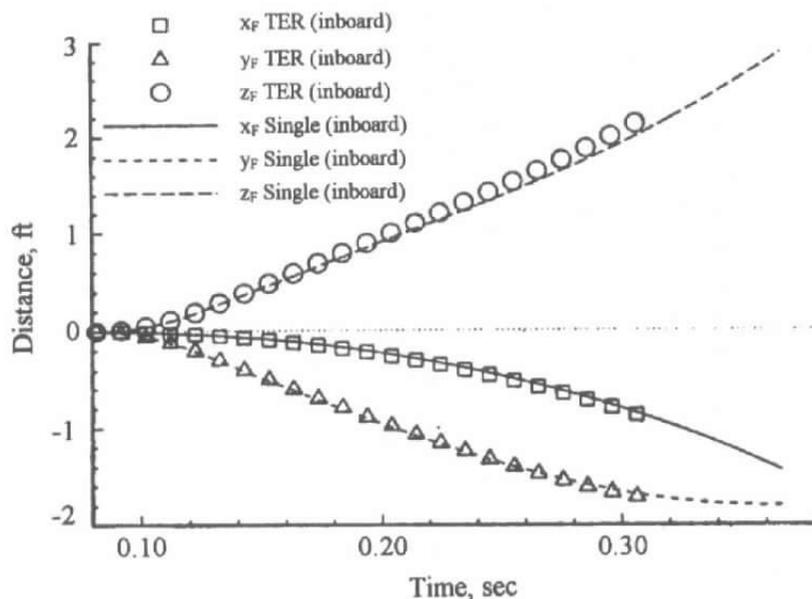


a. Translation distance

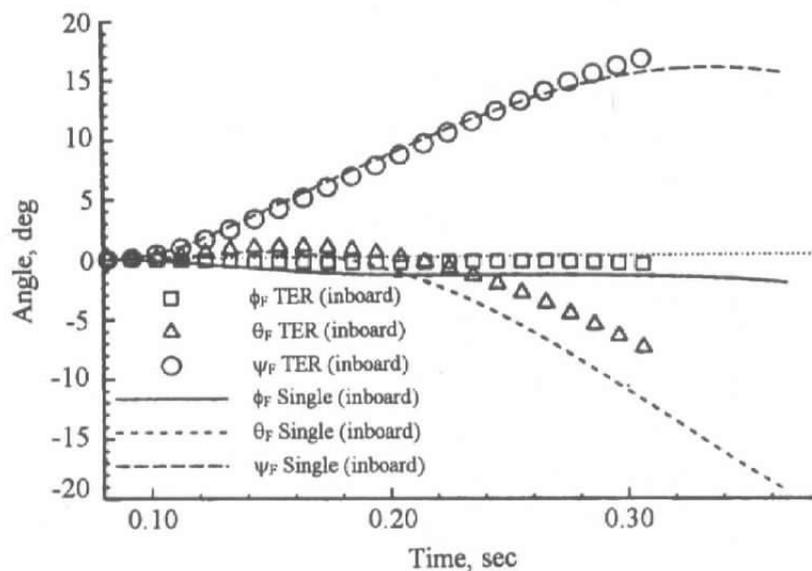


b. Rotation angle

Figure 16. Translational and rotational time history for TER configuration, inboard store.



a. Translation distance



b. Rotation angle

Figure 17. Comparisons of translational and rotational time history for the single-store configuration (inboard store) and the TER configuration (inboard store).

**Table 1. Physical Parameters for the Single Body Launch**

$I_{xx}$	20 slug-ft <sup>2</sup>
$I_{yy}$	360 slug-ft <sup>2</sup>
$I_{zz}$	360 slug-ft <sup>2</sup>
Weight	2,000 lbf
Location of cg	4.65 ft aft of store nose
Forward Ejector	
Location	4.06 ft aft of store nose
Force	2,400 lbf
Aft Ejector	
Location	5.73 ft aft of store nose
Force	9,600 lbf
Ejector Stroke Length	0.33 ft
Store Reference Area	2.18166 ft <sup>2</sup>
Store Reference Length	1.6667 ft

**Table 2. Physical Parameters for the Multiple Body Launch**

$I_{xx}$	10 slug-ft <sup>2</sup>
$I_{yy}$	180 slug-ft <sup>2</sup>
$I_{zz}$	180 slug-ft <sup>2</sup>
Weight	1,000 lbf
Location of cg	4.65 ft aft of store nose
Forward Ejector	
Location	4.06 ft aft of store nose
Force	1,800 lbf
Aft Ejector	
Location	5.73 ft aft of store nose
Force	7,200 lbf
Ejector Stroke Length	0.33 ft
Store Reference Area	2.18166 ft <sup>2</sup>
Store Reference Length	1.6667 ft

**Table 3. TER Configuration: Carriage Loads**

	Bottom	Outboard	Inboard
$C_A$	1.032	1.093	1.176
$C_Y$	-0.262	-0.482	-0.482
$C_N$	0.285	-0.032	0.600
$C_l$	0.002	-0.016	-0.017
$C_m$	-0.795	-0.479	-1.447
$C_n$	0.432	0.861	0.580

**Table 4. Inboard Store Carriage Loads - Single to TER Comparisons**

	TER (Inboard)	Single Store (Inboard)
$C_A$	1.176	1.004
$C_Y$	-0.482	-0.696
$C_n$	0.600	0.990
$C_l$	-0.017	-0.059
$C_m$	-1.447	-2.077
$C_n$	0.580	1.036

## APPENDIX A

### INTEGRATION PROCEDURE FOR SIX-DEGREE-OF-FREEDOM EQUATIONS OF MOTION

Before any discussion of the integration procedure commences, the axis systems required are briefly discussed (complete definitions for all axis systems used in this work are included in Appendix G). Discussion of the equations of motion and the procedure used to integrate the equations are then presented.

#### A.1.0 COORDINATE-AXIS SYSTEMS

Coordinate-axis systems must be defined very carefully in store-separation testing and analysis. The definitions require special care when they must be coupled with the coordinate-axis system definitions used in grid generation and flow solutions with the PEGSUS and XAIR codes, the TESS code, and CTS tests at the AEDC.

Seven coordinate-axis systems are used in the present analysis. They are defined in Appendix G, and three of the systems are shown in Fig. A-1. Two of the systems are used specifically for integration of the six-degree-of-freedom equations of motion. These are the inertial Global Grid Axis System, which is used for integrating the translational equations of motion, and the noninertial CFD Body Axis System, which is used for integrating the rotational equations of motion.

An arbitrary point in space is located by the position vector  $\vec{r}_{GR}$  with respect to the Global Grid Axis System and by the vector  $\vec{r}_{CB}$  with respect to the CFD Body Axis System (see Fig. A-1). The origin of the Global Grid Axis System is at a convenient, fixed point in space, while the origin of the CFD Body Axis System is attached to the center of gravity of the store. The origin of the Global Grid Axis System translates in the initial flight path direction at  $V_\infty$ . A third coordinate-axis system used is the Spatial Axis System. An arbitrary point in space is located by the position vector  $\vec{r}_{SP}$  with respect to the Spatial Axis System. Like the CFD Body Axis System, the origin of the Spatial Axis System is located at the center of gravity of the store. However, unlike the CFD Body Axis System, which translates and rotates with the store, the Spatial Axis System translates with the center of gravity but does not rotate (i.e., its axes remain parallel to the Global Grid Axis System). The Spatial Axis System is useful for relating variables between the Global Grid Axis System and the CFD Body Axis System.

Figure A-1 shows that

$$\vec{r}_{GR} = \vec{CG}_{GR} + \vec{r}_{SP} \quad (\text{A-1})$$

where  $\overrightarrow{CG}_{GR}$  is the position vector of the store center of gravity in the Global Grid Axis System. The components of the vectors  $\vec{r}$  in the Spatial and CFD Body Axis Systems are related by an orthogonal matrix  $G2B$ , namely

$$\vec{r}_{CB} = [G2B]\vec{r}_{SP} = [G2B](\vec{r}_{GR} - \overrightarrow{CG}_{GR}) \quad (A-2)$$

The matrix  $G2B$  is defined by the yaw-pitch-roll, Euler-angle sequence that begins from the Spatial Axis System. The Euler-angle vector is

$$\vec{E} = \begin{bmatrix} \eta_{GR} \\ \nu_{GR} \\ \omega_{GR} \end{bmatrix} \quad (A-3)$$

where  $\eta_{GR}$  is the yaw angle,  $\nu_{GR}$  is the pitch angle, and  $\omega_{GR}$  is the roll angle. In this notation,  $G2B$  has the matrix representation

$$G2B = \begin{bmatrix} \cos(\nu_{GR})\cos(\eta_{GR}) & \cos(\nu_{GR})\sin(\eta_{GR}) \\ \sin(\omega_{GR})\sin(\nu_{GR})\cos(\eta_{GR}) - \cos(\omega_{GR})\sin(\eta_{GR}) & \sin(\omega_{GR})\sin(\nu_{GR})\sin(\eta_{GR}) + \cos(\omega_{GR})\cos(\eta_{GR}) \\ \cos(\omega_{GR})\sin(\nu_{GR})\cos(\eta_{GR}) + \sin(\omega_{GR})\sin(\eta_{GR}) & \cos(\omega_{GR})\sin(\nu_{GR})\sin(\eta_{GR}) - \sin(\omega_{GR})\cos(\eta_{GR}) \\ -\sin(\nu_{GR}) & \\ \sin(\omega_{GR})\cos(\nu_{GR}) & \\ \cos(\omega_{GR})\cos(\nu_{GR}) & \end{bmatrix} \quad (A-4)$$

The inverse of  $G2B$  is required to transform from the CFD Body Axis System to the Global Grid Axis System (or to the Spatial Axis System). Since the  $G2B$  transformation is orthogonal (see pp. 132–148 of Ref. 29) the inverse of  $G2B$  is its transpose, i.e.,  $[G2B]^{-1} = [G2B]^T$ . Therefore, Eq. (A-2) can be written as

$$\vec{r}_{GR} = [G2B]^T \vec{r}_{CB} + \overrightarrow{CG}_{GR} \quad (A-5)$$

The other four axis systems are conventional systems used in CTS testing and are used in SIXDOF only for specific purposes as described briefly here. The Pylon Axis System is used to calculate the displacement of the store from the pylon since ejector models used in the CTS tests and SIXDOF cut off when certain displacement criteria are met in the Pylon Axis System. The Body Axis System is not used in the integration of the equations of motion, but is used to make direct comparisons with tabulated force- and moment-coefficient, rotational velocity, and

rotational acceleration data from the CTS test. The Body Axis System is obtained from the CFD Body Axis System by a rotation of 180 deg about the y-axis, which is common to both systems. The Inertial Axis System is used for direct comparisons with tabulated translational velocity and translational acceleration data from the CTS test. Finally, the Flight Axis System is used for direct comparisons with tabulated translational coordinate data of the store center of gravity and the Euler-angle data of the store from the CTS test. For the case of a non-maneuvering aircraft, the transformations to the Inertial Axis System and the Flight Axis System are straightforward. These transformations have been coded directly in SIXDOF immediately prior to the print statements.

## A.2.0 EQUATIONS OF MOTION

Goldstein, in Secs. 5.1 to 5.5 of Ref. 29, describes how the general translational and rotational motion of any rigid body can be represented by a translation of the center of gravity plus a rotation about the center of gravity.

The six degrees of freedom in the equations of motion implemented in SIXDOF have been chosen to be the three Cartesian coordinates of the store center of gravity to represent the translation of the store with respect to the aircraft, and the three Euler angles to represent the rotation of the store about the center of gravity.

### A.2.1 TRANSLATION

The translational equations of motion of the store center of gravity are integrated in the Global Grid Axis System. This axis system is fixed with respect to the steady translation of the aircraft and is an inertial system. The aerodynamic, gravity, and ejector forces must be defined with respect to the Global Grid Axis System. The aerodynamic and ejector forces are calculated in the CFD Body Axis System, and must therefore be transformed to the Global Grid Axis System by multiplication by the transpose of the  $G2B$  transformation matrix,  $[G2B]^T$ .

The translational equations of motion to be integrated are simply those of Newton's second law, namely,

$$\dot{\vec{v}} = \vec{F}/m \quad (\text{A-6})$$

where  $\vec{v}$  is the translational velocity vector of the store,  $\vec{F}$  is the total-force vector (comprised of aerodynamic, ejector, and gravity contributions) acting on the store,  $m$  is the store mass, and  $(\dot{\quad})$  denotes a time derivative. The velocity vector,  $\vec{v}$ , is related to the time rate of change of CG by

$$\frac{d}{dt} \vec{CG} = \vec{v} \quad (\text{A-7})$$

## A.2.2 ROTATION

The rotational equations of motion about the store center of gravity are integrated in the CFD Body Axis System. This axis system is fixed in the store and so is noninertial; however, the inertia tensor of the store is invariant in this system. The aerodynamic and ejector moments are evaluated in the CFD Body Axis System, so transformation is not required. Since all equations in this subsection are in the CFD Body Axis System, the subscripts CB will be omitted for clarity.

The rotational equations of motion to be integrated are discussed in Chapter 5 of Ref. 29. Specifically, Euler's equations of rotational motion about the center of gravity [Eqs. (5-37) of Ref. 29] are

$$\dot{\vec{L}} = \vec{N} - \vec{\omega} \times \vec{L} \quad (\text{A-8})$$

where  $\vec{N}$  is the external moment vector (comprised of aerodynamic and ejector contributions) about the center of gravity,  $\vec{\omega}$  is the angular velocity vector, and  $\vec{L}$  is the total angular momentum vector [Eqs. (5-5) and (5-9) of Ref. 29], namely,

$$\vec{L} = I\vec{\omega} \quad (\text{A-9})$$

The symbol  $I$  is used for the inertia tensor (using the sign convention of Appendix A.3 of Ref. 8) as (in the Body Axis System):

$$I = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \quad (\text{A-10})$$

The diagonal terms are the moments of inertia and the off-diagonal terms are the products of inertia. The inertia tensor is invariant with respect to time in the Body Axis System. Note that Eq. (A-10) is defined in the Body Axis System. In the current methodology, the CFD Body Axis System is used for integrating the rotational equations of motion. Therefore, to be used in the current implementation of the 6DOF integration code, Eq. (A-10) must be expressed in the CFD Body Axis System. This conversion is accomplished by simply changing the signs on the  $I_{xy}$  and  $I_{yz}$  terms appearing in Eq. (A-10).

Equations (A-8) and (A-9) can be put into a form suitable for numerical integration as follows. First, since  $I$  is invariant with time, Eq. (A-9) can be differentiated with respect to time to give

$$\dot{\vec{L}} = I\dot{\vec{\omega}} \quad (\text{A-11})$$

so that, with the inverse of the inertia tensor

$$\dot{\vec{\omega}} = I^{-1} \dot{\vec{L}} \quad (\text{A-12})$$

The relationship between  $\dot{\vec{\omega}}$  and the time derivatives,  $\dot{\vec{E}}$ , of the Euler angles is given [Eqs. (B-14xyz) of Appendix B in Ref. 29] by the transformation matrix  $T$ , namely,

$$\dot{\vec{\omega}} = T \dot{\vec{E}}, \quad (\text{A-13})$$

where

$$T = \begin{bmatrix} -\sin(v_{GR}) & 0 & 1 \\ \cos(v_{GR})\sin(\omega_{GR}) & \cos(\omega_{GR}) & 0 \\ \cos(v_{GR})\cos(\omega_{GR}) & (-\sin(\omega_{GR})) & 0 \end{bmatrix} \quad (\text{A-14})$$

The inverse of  $T$  is

$$T^{-1} = \begin{bmatrix} 0 & \sin(\omega_{GR})/\cos(v_{GR}) & \cos(\omega_{GR})/\cos(v_{GR}) \\ 0 & \cos(\omega_{GR}) & -\sin(\omega_{GR}) \\ 1 & (-\sin(\omega_{GR})\tan(v_{GR})) & \cos(\omega_{GR})\tan(v_{GR}) \end{bmatrix} \quad (\text{A-15})$$

so that

$$\dot{\vec{E}} = T^{-1} \dot{\vec{\omega}} \quad (\text{A-16})$$

### A.3.0 NUMERICAL INTEGRATION OF THE EQUATIONS OF MOTION

Numerical integration of the translational equations of motion is performed by a symplectic second-order Runge-Kutta scheme, which has been shown by Sanz-Serna (Ref. 30) to perform better than its nonsymplectic counterparts for integration over long time periods. The values of  $\vec{F}$ ,  $\vec{v}$ , and  $\vec{CG}$  from the previous time step are used in the integration of the next time step. For Eqs. (A-6) and (A-7), the symplectic integration scheme corresponds to a basic Euler (first-order Runge-Kutta) integration (Ref. 31) of Eq. (A-6), for one time step, to obtain the updated vector  $\vec{v}$ , followed by a second-order Runge-Kutta integration of Eq. (A-5), for the same time step, using the previous and updated  $\vec{v}$ , to obtain  $\vec{CG}$ . The algorithm is:

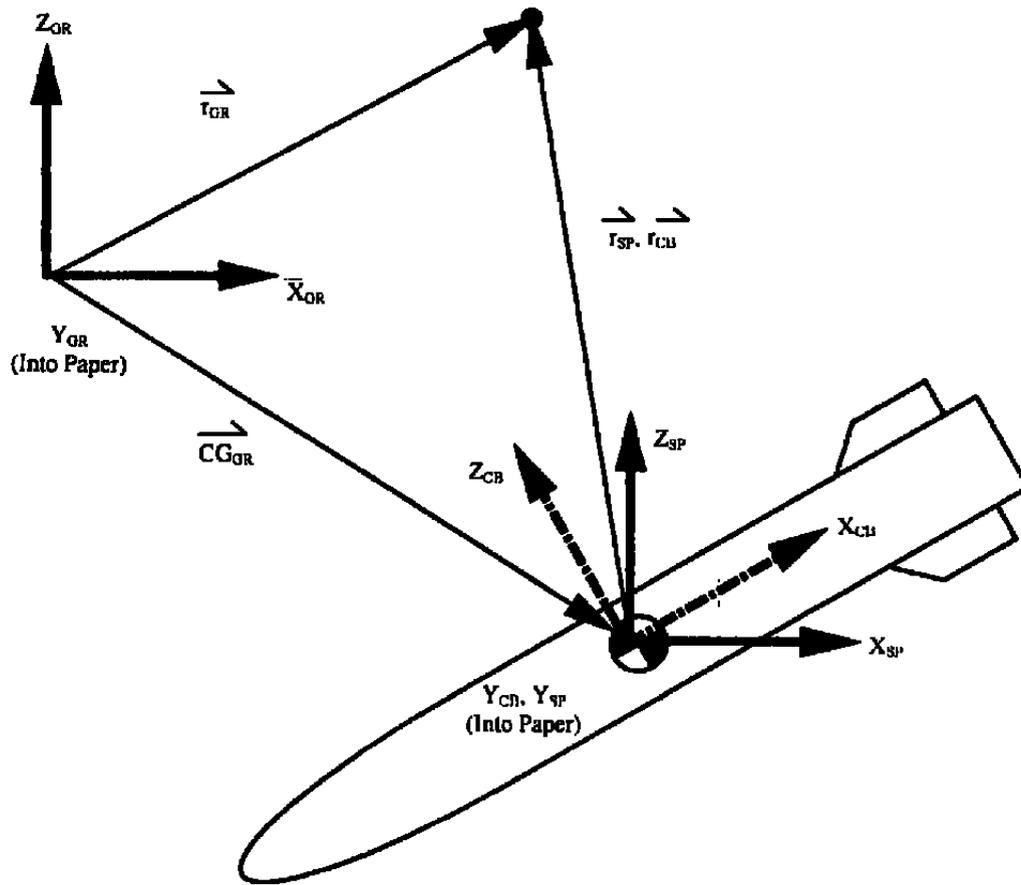
$$(\dot{\mathbf{v}}^{n+1} - \dot{\mathbf{v}}^n)/\Delta t = \dot{\mathbf{F}}^n/m, \quad (\text{A-17})$$

then

$$(\overrightarrow{CG}^{n+1} - \overrightarrow{CG}^n)/\Delta t = (\dot{\mathbf{v}}^{n+1} + \dot{\mathbf{v}}^n)/2, \quad (\text{A-18})$$

where the superscripts  $n$  and  $n+1$  denote the previous and present time steps, respectively.

Numerical integration of the rotational equations of motion is performed by a fourth-order Runge-Kutta scheme (Ref. 31). The first step is to invert the inertia tensor,  $I$ . The values of the vectors  $\vec{\omega}$ ,  $\vec{N}$ , and  $\vec{E}$ , from the previous time step, are used in the integration of the next time step. Thus, for each of the four evaluations that comprise the next time step in the fourth-order Runge-Kutta scheme, the integration proceeds as follows. Equation (A-9) is evaluated to find  $\vec{L}$ , and  $\vec{\omega} \times \vec{L}$  is subtracted from  $\vec{N}$  to determine  $\vec{L}$  by Eq. (A-8). The resulting  $\vec{L}$  is multiplied by  $I'$  to find  $\vec{\omega}$  from Eq. (A-12). Next,  $\vec{\omega}$  is used to determine  $\vec{E}$  from Eq. (A-16). The four evaluations of the six equations for the components of  $\vec{\omega}$  and  $\vec{E}$  are integrated by the Runge-Kutta procedure to determine the updated vectors  $\vec{\omega}$  and  $\vec{E}$ .



**Figure A-1. Computational axis system.**

## APPENDIX B VALIDATION OF SIXDOF

Six preliminary check cases were performed as initial tests of the SIXDOF code. In each case, the basic validity of the method of integrating the 6DOF equations implemented in the SIXDOF code was confirmed.

### B.1.0 TRAJECTORY IN A CONSTANT GRAVITATIONAL FIELD WITH NO LOADS

In the first case, the store was dropped in a uniform, gravitational field with all ejector forces and aerodynamic force and moment coefficients set to zero. Using SIXDOF, the exact solution to the equations of motion was reproduced, independent of the time step used.

### B.2.0 TRAJECTORY IN A CONSTANT GRAVITATIONAL FIELD WITH CONSTANT TORQUES

For the next three cases, the store was again dropped in a constant gravitational field. However, in these three cases, all force coefficients were set to zero but a constant torque was applied about each of the three axes, in turn. The SIXDOF code produced the exact solution for the translational and rotational motion independent of the time step in these cases as well.

### B.3.0 ASYMMETRICAL BODY WITH INITIAL ANGULAR VELOCITIES

The fifth and sixth validation cases were more rigorous and require more discussion. Computations were performed on a completely asymmetrical body undergoing free motion with no torques or constraints. In this case, in an inertial axis system the angular momentum of the body is constant. Using the Spatial Axis System,

$$\frac{d\vec{L}_S}{dt} = 0$$

where  $\vec{L}_S = I_S \vec{\omega}_S$ ,  $I_S$  is the inertia tensor and  $\vec{\omega}_S$  is the angular velocity in the Spatial Axis System. Note that in this case, constant  $\vec{L}_S$  implies that each component of  $\vec{L}_S$  is constant. The equation above is defined in the Spatial Axis System; however, the rotational equations of motion in SIXDOF are integrated in the CFD Body Axis System. In the CFD Body Axis System, the rotational equations of motion for free rotation become

$$\frac{d\vec{L}_{CB}}{dt} + \vec{\omega} \times \vec{L}_{CB} = 0 \quad (\text{B-1})$$

which implies that components of  $L_{CB}$  evolve in time. The magnitude of  $L_{CB}$ , however, is constant. This can be seen by forming the dot product of  $\dot{L}_{CB}$  and Eq. (B-1). Since  $\dot{L}_{CB} \bullet (\vec{\omega}_{CB} \times \dot{L}_{CB}) = 0$ ,

$$\dot{L}_{CB} \bullet \frac{d\dot{L}_{CB}}{dt} = \frac{1}{2} \frac{d}{dt} (\dot{L}_{CB} \bullet \dot{L}_{CB}) = 0.$$

This implies that  $\dot{L}_{CB} \bullet \dot{L}_{CB} = \text{constant}$ , i.e., the magnitude of  $\dot{L}_{CB}$  is constant.

The vectors  $\dot{L}_{CB}$  and  $\dot{L}_S$  are related to each other by an orthogonal transformation; namely,  $\dot{L}_S = G2B^T \dot{L}_{CB}$ , where  $G2B$  is defined in Eq. (A-4) and is a function of the Euler angles defined in Eq. (A-3). Therefore, Eq. (A-16) must be integrated simultaneously with Eq. (B-1) in order to determine  $\dot{L}_S$  from  $\dot{L}_{CB}$ . This provides a demanding test of the coding and integration routines in SIXDOF. If coding errors were present or a time step were used which caused excessively large truncation errors, the magnitude of the  $\dot{L}_S$  and  $\dot{L}_{CB}$  vectors would not remain constant throughout the course of the integration.

Computations were performed for an object whose inertia tensor in the CFD Body Axis System is given as follows:

$$I_{CB} = \begin{bmatrix} 20 & -36 & -2 \\ -36 & 360 & -36 \\ -2 & -36 & 360 \end{bmatrix} \text{slug-ft}^2$$

At  $t = 0$ , the CFD Body Axis and Spatial Axis systems coincide. The initial values of  $\vec{\omega}_{CB}$  and  $\vec{\omega}_S$  were taken to be

$$\vec{\omega}_{CB}(t = 0) = \vec{\omega}_S(t = 0) = \begin{bmatrix} 20 \\ 30 \\ 40 \end{bmatrix} \text{rad/sec}$$

The motion of the body can be understood by examining Fig. B-1a. The body experiences oscillations in the pitch angle while experiencing a nearly constant increase in the yaw and roll angles. Although the Euler angles and the components of  $\dot{L}_{CB}$  undergo a complicated time evolution, the magnitude of  $\dot{L}_S$  and  $\dot{L}_{CB}$  and the individual components of  $\dot{L}_S$  must remain constant.

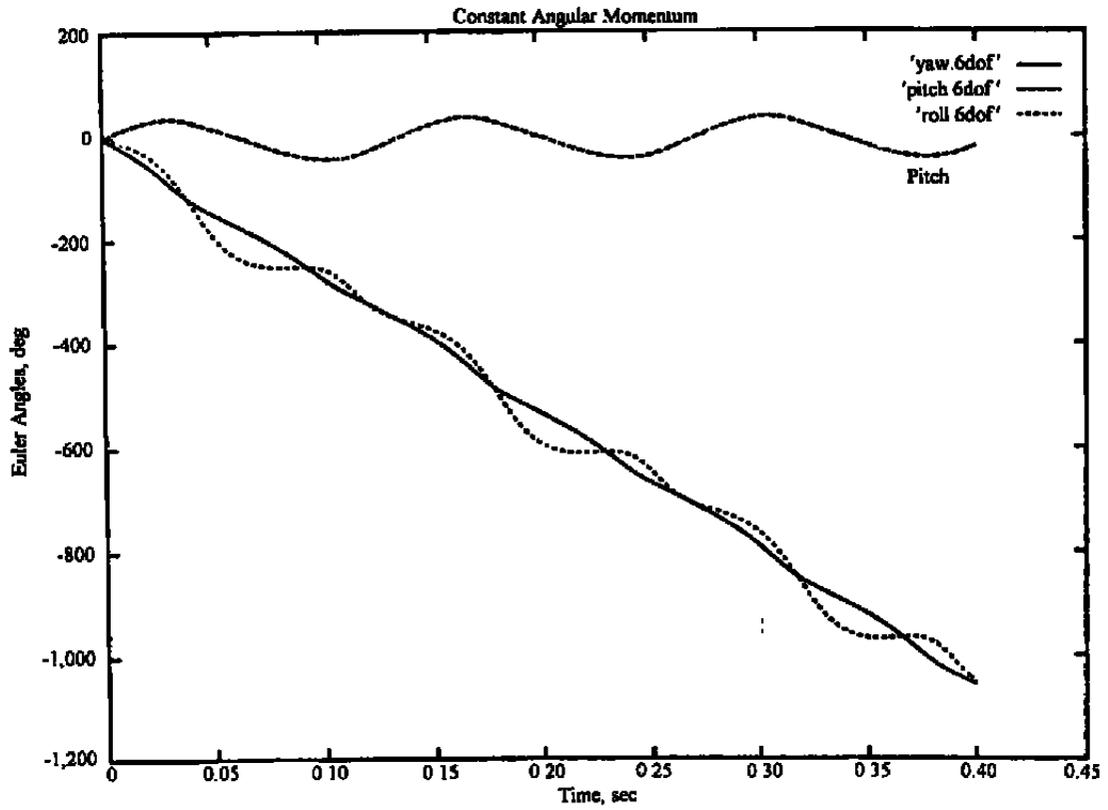
Computations were performed using time steps of  $\Delta t = 0.001$  and  $\Delta t = 0.01$  sec. In both cases, the numerical integrations proceeded until  $t = 0.4$  sec was reached. The resulting magnitudes of  $\vec{L}_{CB}$  for both cases are presented in Fig. B-1b. The magnitudes of  $\vec{L}_S$  are not included in the plot, but in both cases were equal to the analogous magnitudes of  $\vec{L}_{CB}$ . Examination of Fig. B-1b shows that with the 0.001-sec time step, the magnitudes of  $\vec{L}_{CB}$  and  $\vec{L}_S$  were held constant. The individual components of  $\vec{L}_{CB}$  are not plotted, but were also held constant. Therefore, the integration routines used in the SIXDOF code were validated. However, with the 0.01-sec time step, the magnitude of  $\vec{L}_{CB}$  decreased approximately 8 percent. This result indicates that, even though the SIXDOF routines are valid, there is a limit to the motion that can accurately be predicted over a given time increment due to truncation errors. As a rule of thumb, the product of the magnitude of  $\vec{\omega}_{CB}$  and  $\Delta t$  should be approximately 0.05 rad or less to avoid the dissipative effects of the Runge-Kutta algorithm which occur when large truncation errors are present.

#### B.4.0 WL/AD STORE TRAJECTORY USING CTS TEST MEASURED LOADS

Another basic check of the validity of the SIXDOF code was made by changing the procedure outlined in Fig. 2. The time-accurate prediction of the aerodynamic loads from the XAIR flow solver and TESS was replaced with the measured values from the WL/AD CTS test. Therefore, all of the CFD-related codes in Fig. 2, namely PEGSUS, ROTRANS, XAIR, and TESS, were eliminated from the data flow. The ejector model remained the same. The measured aerodynamic force and moment coefficients listed at time intervals of 0.01 sec were used to check SIXDOF. These tabulated data were at time intervals 100 times greater than the 0.0001-sec time step used in SIXDOF for the single-store validation case. Linear interpolation was used to obtain the force and moment coefficients at each SIXDOF time step that did not appear in the data tabulation. The aerodynamic angular-rate damping coefficients from Table 1 were used in the calculation, as they had been in the CTS test.

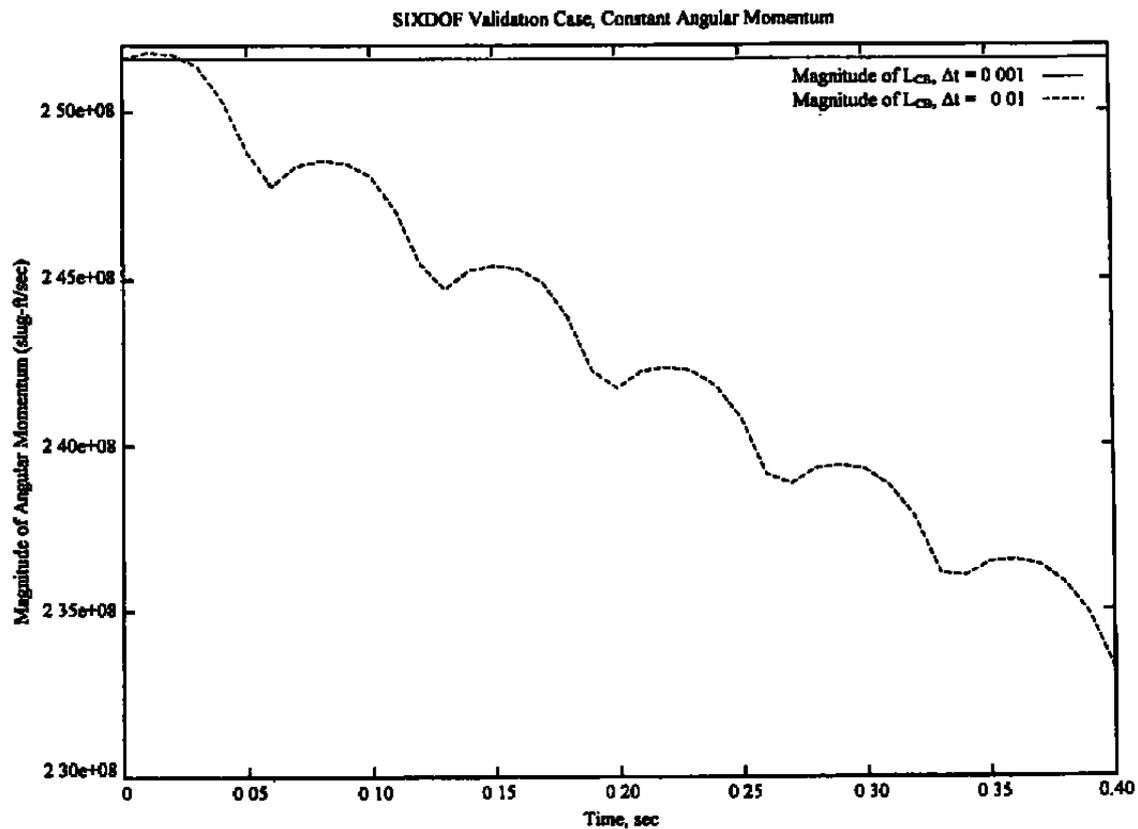
The translational and rotational constituents of the store trajectory and velocity components calculated by the SIXDOF code using the measured aerodynamic loads are shown in Fig. B-3. Translation of the store center of gravity is given by the components  $x_F$ ,  $y_F$ , and  $z_F$  in the Flight Axis System. The time history of the SIXDOF translational results in Fig. B-3a is in excellent agreement with the data from the CTS test, deviating by less than 0.1 ft in all cases. The translational velocity components of the store center of gravity in Fig. B-3b are  $u_I$ ,  $v_I$ , and  $w_I$  relative to the origin of the Inertial Axis System but in the directions of the Body Axis System. The SIXDOF translational velocity results also are in excellent agreement with the CTS data. The rotational angles (here the Euler angles) of the store are  $\psi_F$ ,  $\theta_F$ , and  $\phi_F$  defined in the Flight Axis System and shown in Fig. B-2c. The time history of the SIXDOF Euler angles is in reasonable agreement with the CTS data, deviating by less than 0.5 deg. The store rotational velocity

components are  $p_B$ ,  $q_B$ , and  $r_B$  defined in the Body Axis System and shown in Fig. B-2d. There are slight differences in the SIXDOF rotational velocity components of the store; these differences are consistent with the differences observed in Fig. B-2c for the store rotational angles. Overall, the agreement is acceptable and once again confirms the basic validity of the SIXDOF code.

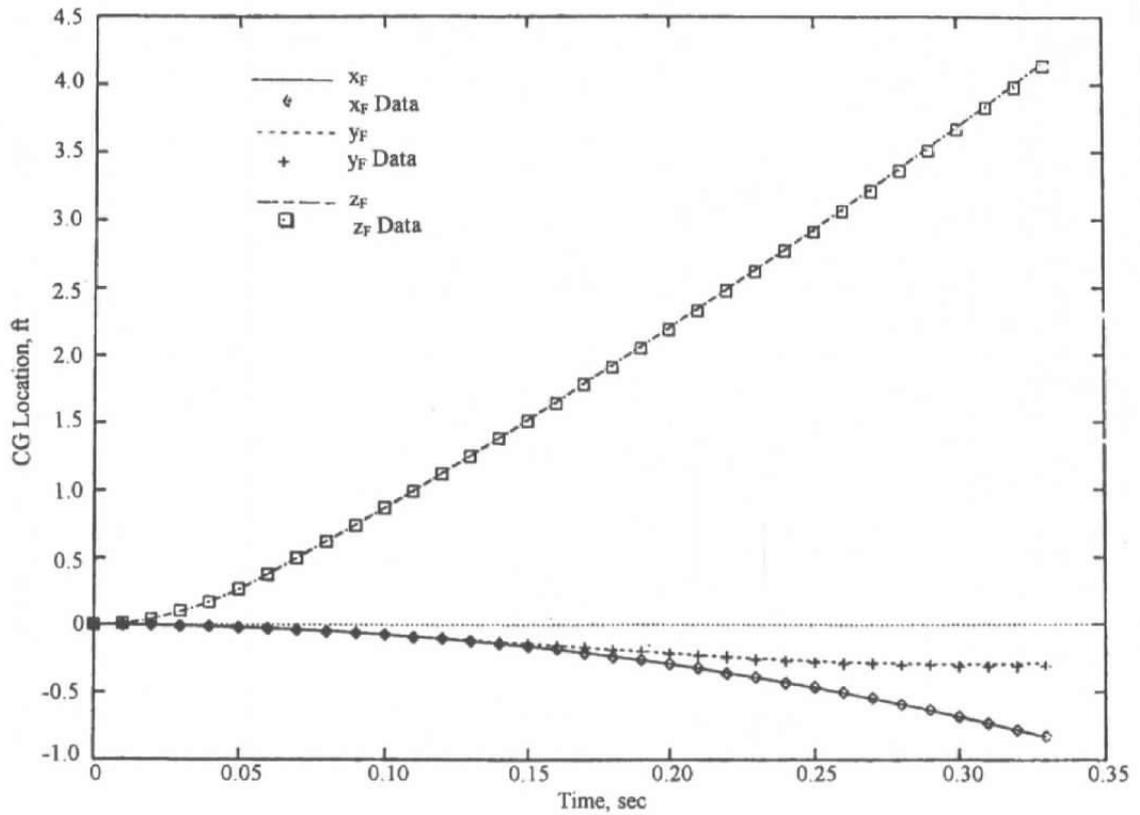


a. Angular motion

Figure B-1. SIXDOF validation: asymmetrical body with initial angular velocities.

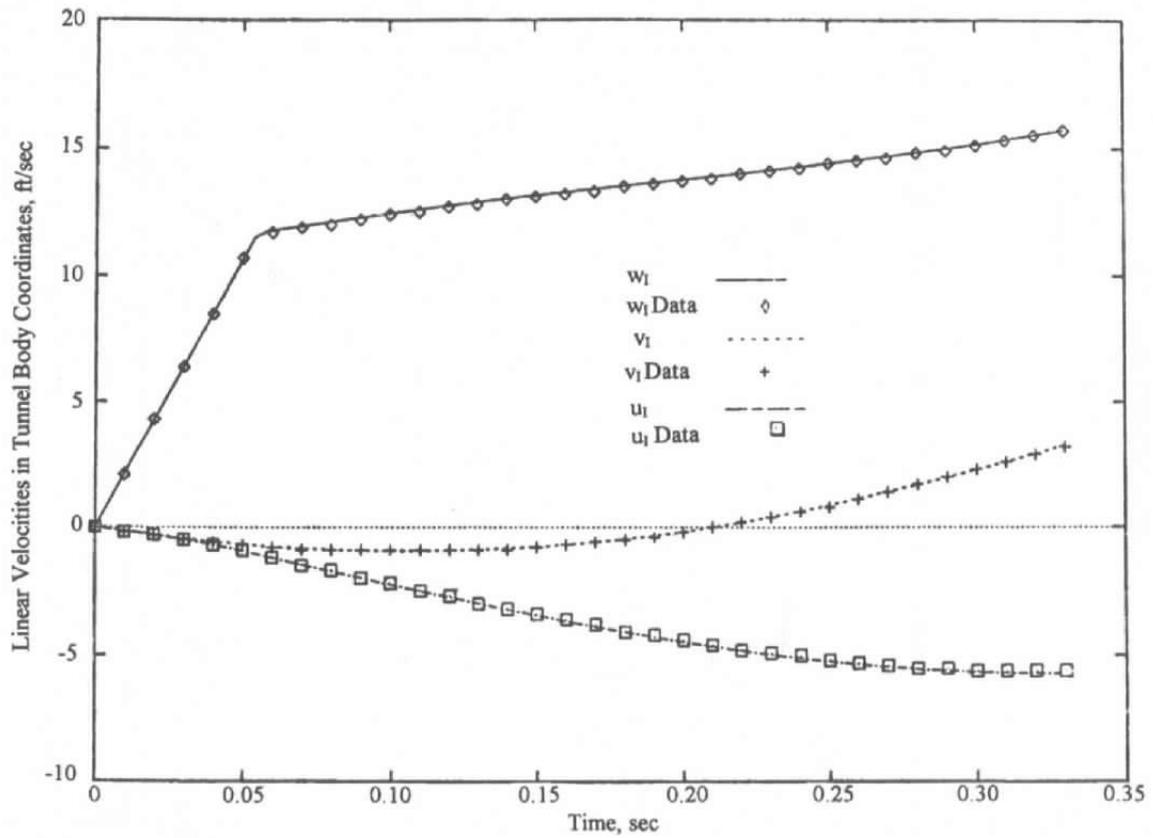


b. Angular momentum magnitudes  
Figure B-1. Concluded.

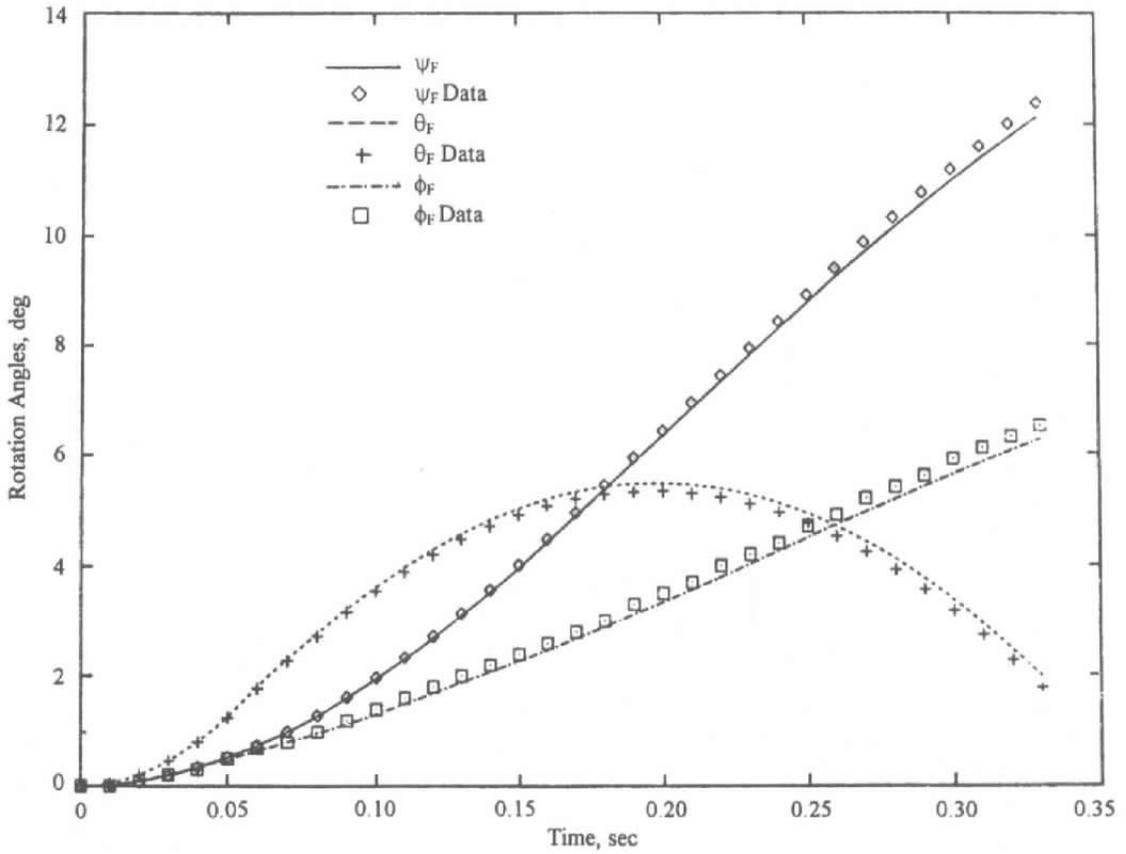


a. Translation of center of gravity in flight axis system

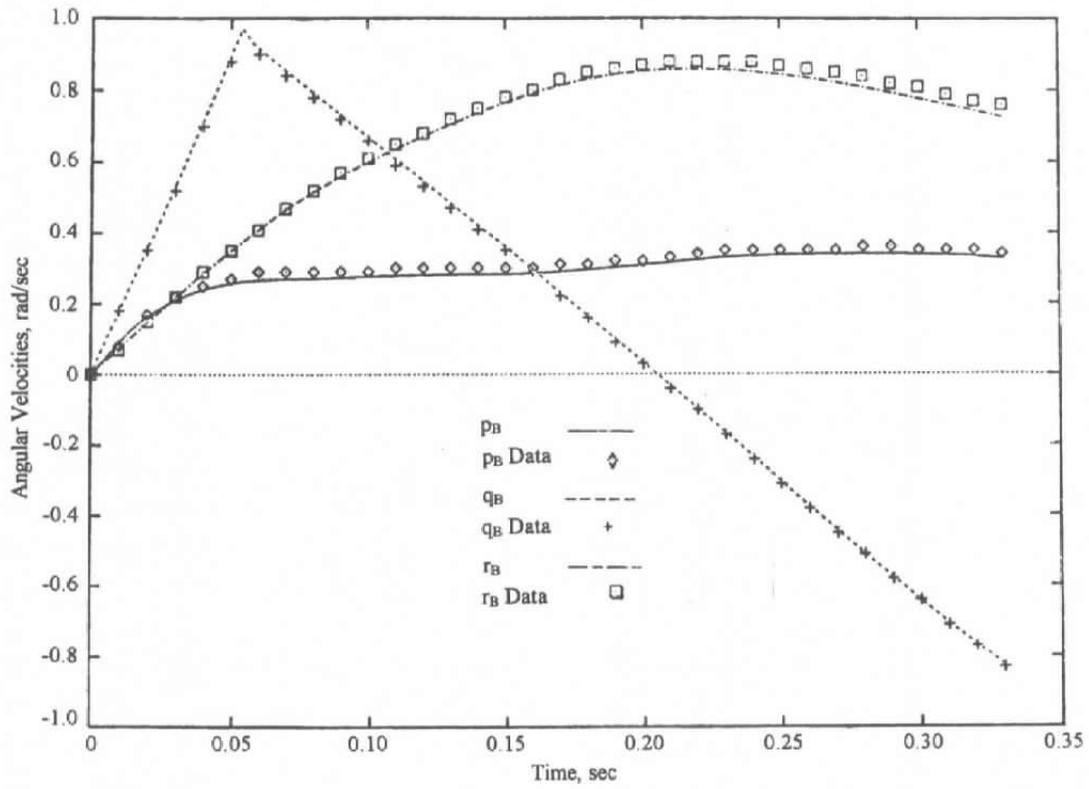
Figure B-2. SIXDOF validation: WL/AD store trajectory using CTS test measured loads.



b. Translational velocity in inertial axis system  
 Figure B-2. Continued.



c. Rotational angles with respect to flight axis system  
 Figure B-2. Continued.



**d. Rotational velocity in body axis system**  
**Figure B-2. Concluded.**

## APPENDIX C TRAJECTORY SENSITIVITY STUDY

Results of the single-store validation case led to sensitivity studies with the rapid trajectory computational capability described in Appendix B. In each of the sensitivity studies to follow, the procedure for calculating a trajectory is that described in Sec. B4.0. In all cases, the load coefficients used are obtained from the CTS test data of the WL/AD store.

### C1.0 TRAJECTORY SENSITIVITY TO AERODYNAMIC LOADS AND EJECTOR MODEL

Two parameters that have a large influence on the trajectory, specifically on the pitch angle history, are the aerodynamic pitching-moment coefficient and the stroke length of the ejector pistons. The effect of reducing the nose-up pitching-moment coefficients uniformly by 0.2 (approximately the difference between measured and predicted levels, see Fig. 11a) is shown in Fig. C-1a. The stroke length of the ejector piston remains at 0.33 ft (see Table 1), and leads to cut-off of the aft piston at  $t = 0.053$  sec and cutoff of the forward piston at  $t = 0.056$  sec. The yaw and roll angles are changed by less than 1 deg from those of Fig. B-2c, but the pitch angle differs by approximately 3 deg. If the pitching-moment coefficient remains changed uniformly by 0.2, but the stroke length of the ejector piston is increased from 0.33 ft to 0.40 ft, the rotational portion of the trajectory is changed as shown in Fig. C-1b. In this example, the rear piston cuts off at  $t = 0.058$  sec and the front piston cuts off at  $t = 0.062$  sec. The computed  $\theta_F$  angles in Fig. C-1b are closer to the CTS data than the computed angles in Fig. C-1a, but still are not back to the levels of agreement with CTS data in Fig. B-2c. Although not shown here, the translational portion of these trajectories is not changed significantly, even when the ejector stroke is increased.

A final result that further illuminates the sensitivity of the results to the initial conditions imparted by the ejectors was computed as follows. The aerodynamic force and moment coefficients calculated by the AEDC time-accurate solution procedure, and shown in Figs. 10a and 11a, were used as input to the trajectory calculation procedure of Appendix B. The only difference in the calculations from the time-accurate results in Sec. 3.2.4.1 was to increase the stroke length of the ejector pistons from 0.33 ft to 0.42 ft. The resulting rotational angles of the store,  $\psi_F$ ,  $\theta_F$ , and  $\phi_F$ , are presented in Fig. C-1c. Comparison of Fig. C-1c with Fig. 12b shows that the AEDC predictions of the yaw and roll angles are virtually unchanged, but the prediction of the pitch angle,  $\theta_F$ , now differs from the CTS data by less than 1 deg.

These calculations clearly demonstrate the extreme sensitivity of the trajectory to moments exerted on the store and especially the initial conditions imparted by the ejectors.

### C. 2.0 TRAJECTORY SENSITIVITY TO ANGULAR-RATE DAMPING

In CTS testing, angular-rate damping (damping coefficients in yaw, pitch, and roll) is not always known accurately. Note that this difficulty does not arise in the time-accurate CFD computation of the aerodynamic force and moment coefficients because the angular-rate damping effects are part of the solution. Nevertheless, the sensitivity of the trajectory to the damping coefficients is of interest.

The check-case trajectory in Sec. B-4.0 was repeated with the angular-rate damping coefficients set to zero so that the sensitivity to the rates could be determined. Only the rotational results are presented in Fig. C-2a, since the translational results are independent of the damping rates in the approximations of this sensitivity investigation. Comparison of Figs. C-2b and B-2c shows that the computed yaw and roll are changed so that the agreement with CTS data is somewhat better, while the agreement in pitch,  $\theta_F$ , is somewhat degraded. These trajectory differences are reflected in the rotational velocity comparisons of Figs. C-2c and B-2d. For the WL/AD store, then, the angular-rate damping does not have a major influence on the rotational portion of the store trajectory, in contrast to the significant effects of pitching-moment coefficient.

### C.3.0 TRAJECTORY SENSITIVITY TO RANDOM ERRORS IN AERODYNAMIC LOADS

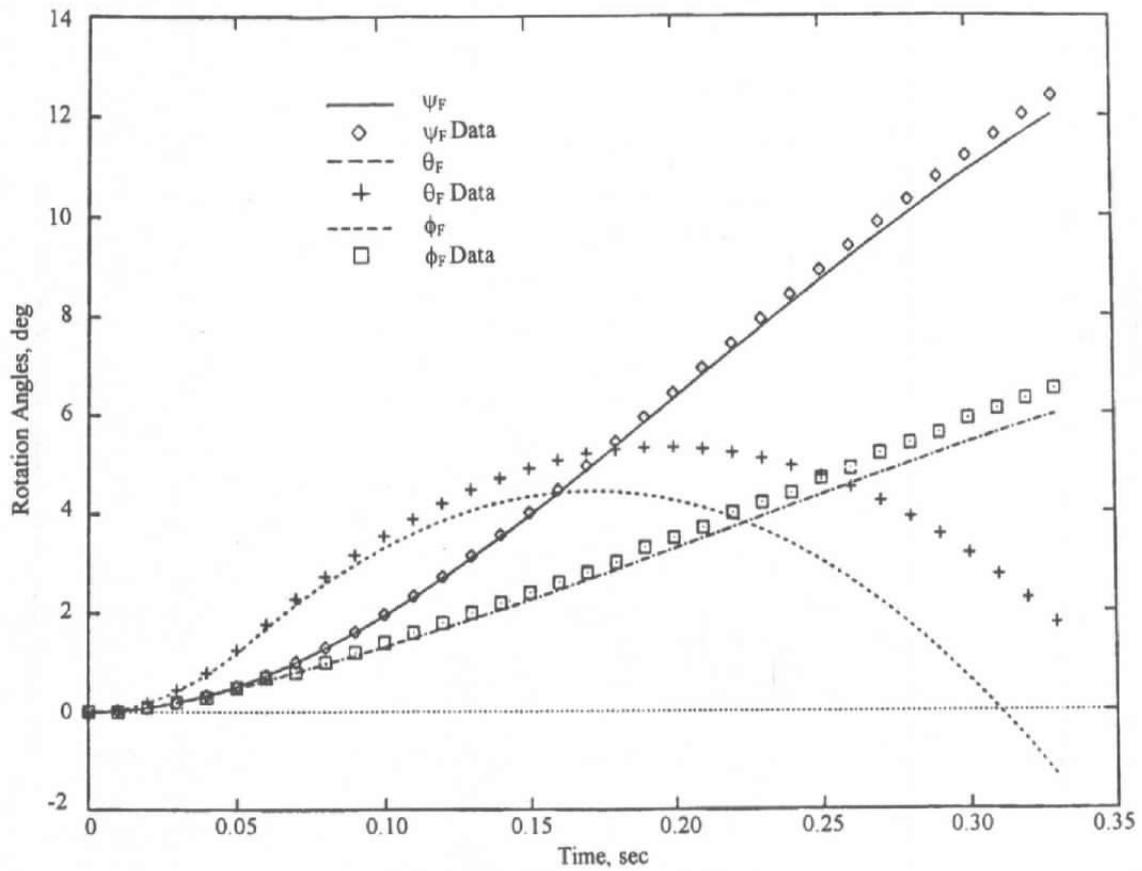
Bias effects of differences in the pitching-moment coefficient are shown in Sec. C.1.0 to be significant in the accuracy of the store trajectories. These results are considered to be representative, generally, of constant bias effects from all force and moment coefficients. In this section, the effects of random errors in the aerodynamic loads are investigated.

Investigation of these effects were achieved by using a random noise term taken from Program GASDEV in Ref. 32 to perturb all of the force and moment coefficients that were obtained from the CTS test of the WL/AD store. The random numbers produced by GASDEV have a Gaussian distribution centered at 0.0 with a standard deviation of 1.0, which is large compared to the expected random noise in the data. These numbers are scaled by 0.5 and used to compute the force and moment coefficients randomly by the equation

$$C_{XCalculation} = [1.0 + 0.5 \cdot GASDEV(ISEED)]C_{XData}$$

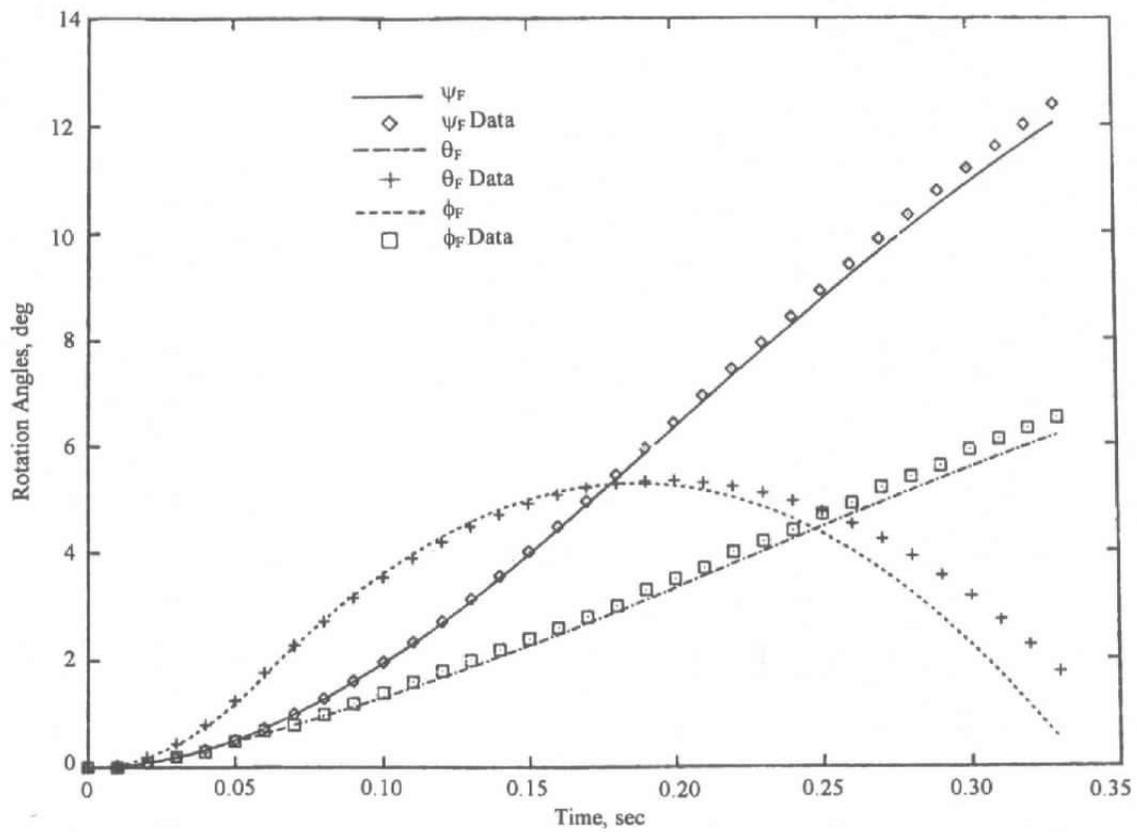
where  $C_x$  is any of the coefficients and ISEED is an arbitrary initializing negative integer number used in generating the random sequence. The value of ISEED has no effect on the standard deviation of the Gaussian distribution. Computations were run with ISEED values of  $-1$  and  $-987654321$ . The translational and rotational results for the ISEED =  $-1$  computation are presented in Fig. C-3a, and C-3b, respectively and for the ISEED =  $-987654321$  computation in Fig. C-3c and C-3d,

respectively. Comparisons of the translational results in Figs. C-3a and C-3c with the baseline check case results of Fig. B-2a show an insignificant effect. Comparisons of the rotational results in Figs. C-3b and C-3d with the baseline check case results of Fig. B-2c show a larger effect. Here, the computed results fall on opposite sides of the data due to the random nature of the force- and moment-coefficient inputs to the calculations. However, the rotational results are much closer to the CTS data than are the computed results in Fig. C-1, where realistic differences in predicted and measured pitching-moment coefficients and ejector model uncertainty have been presented. The conclusion is that realistic values of random noise in the measurements do not result in significant errors in the trajectories.

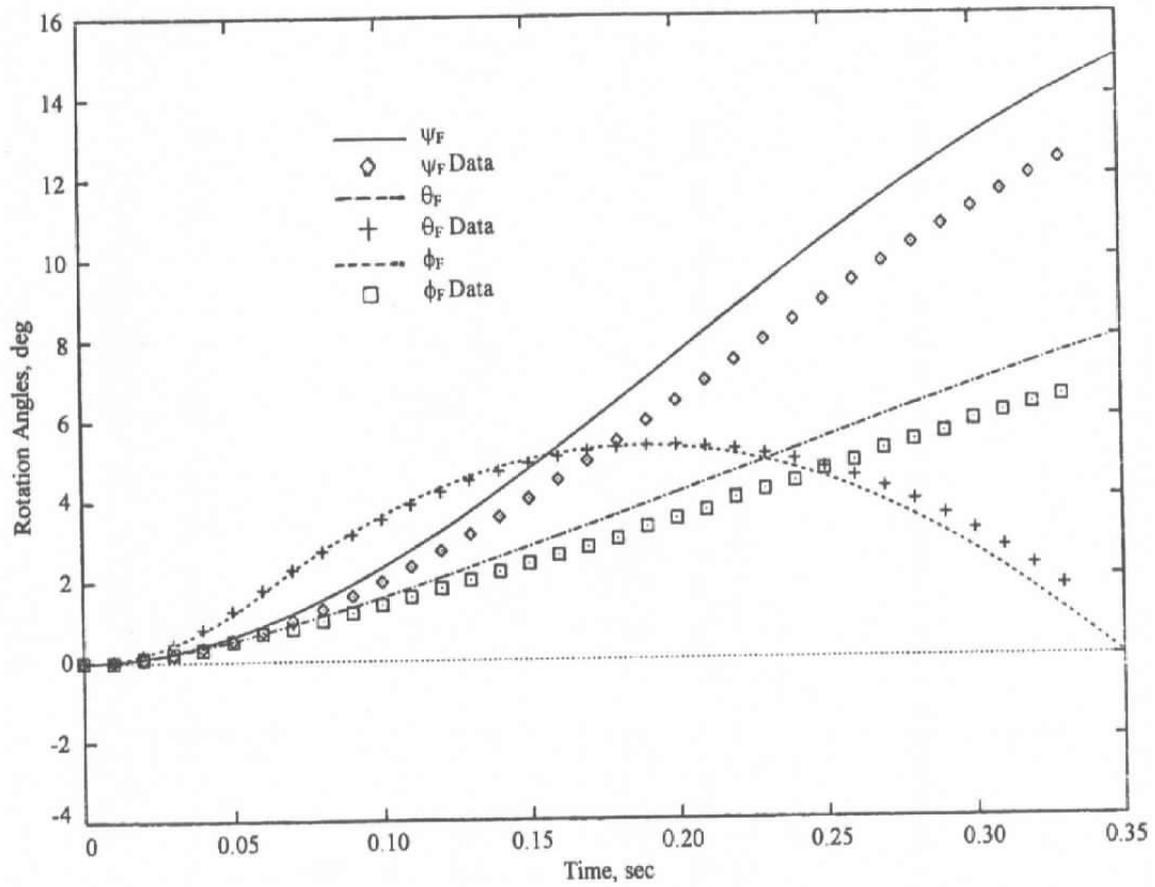


a. Perturbed  $C_m$

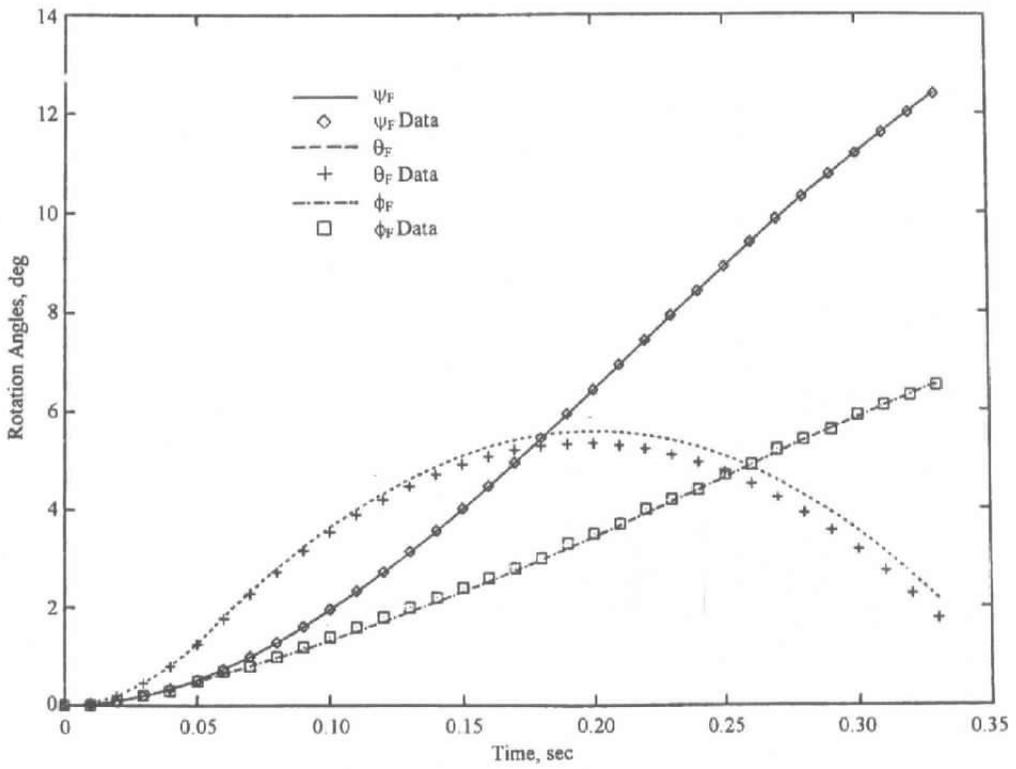
Figure C-1. Trajectory sensitivity to aerodynamic loads and ejector model.



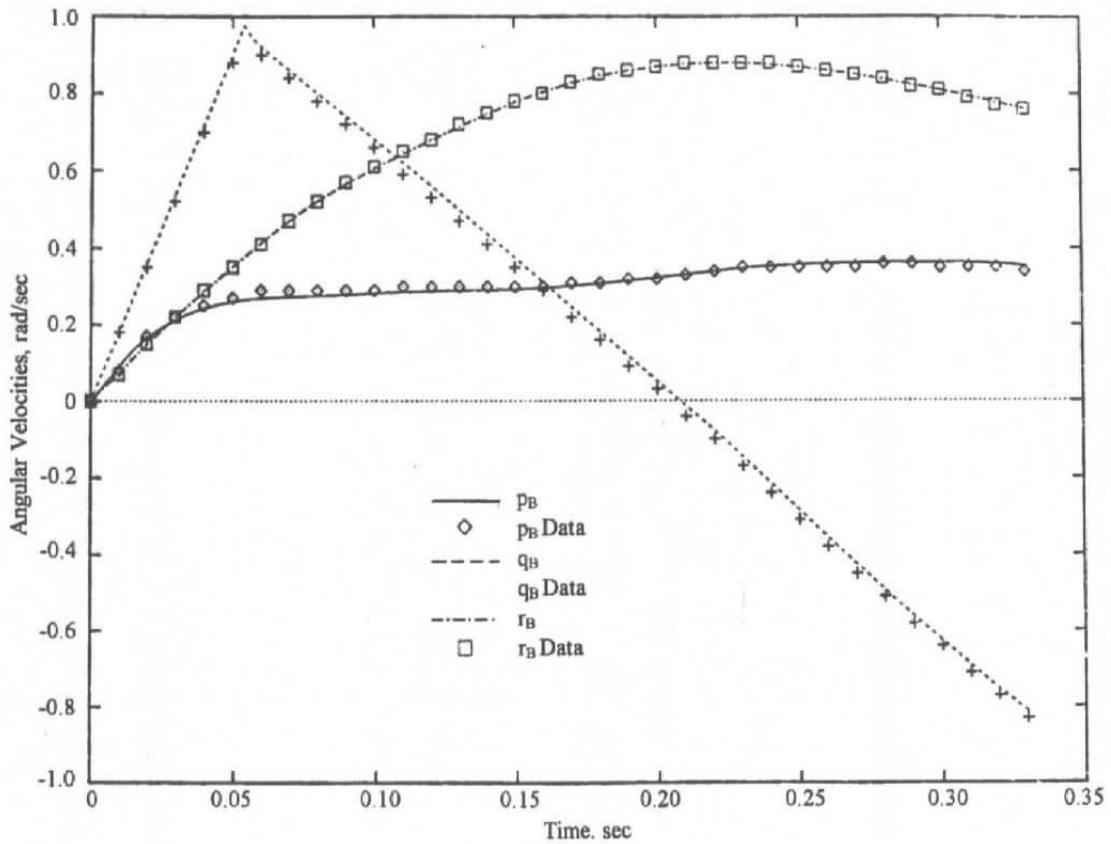
**b. Perturbed  $C_m$  and ZEJCT**  
**Figure C-1. Continued.**



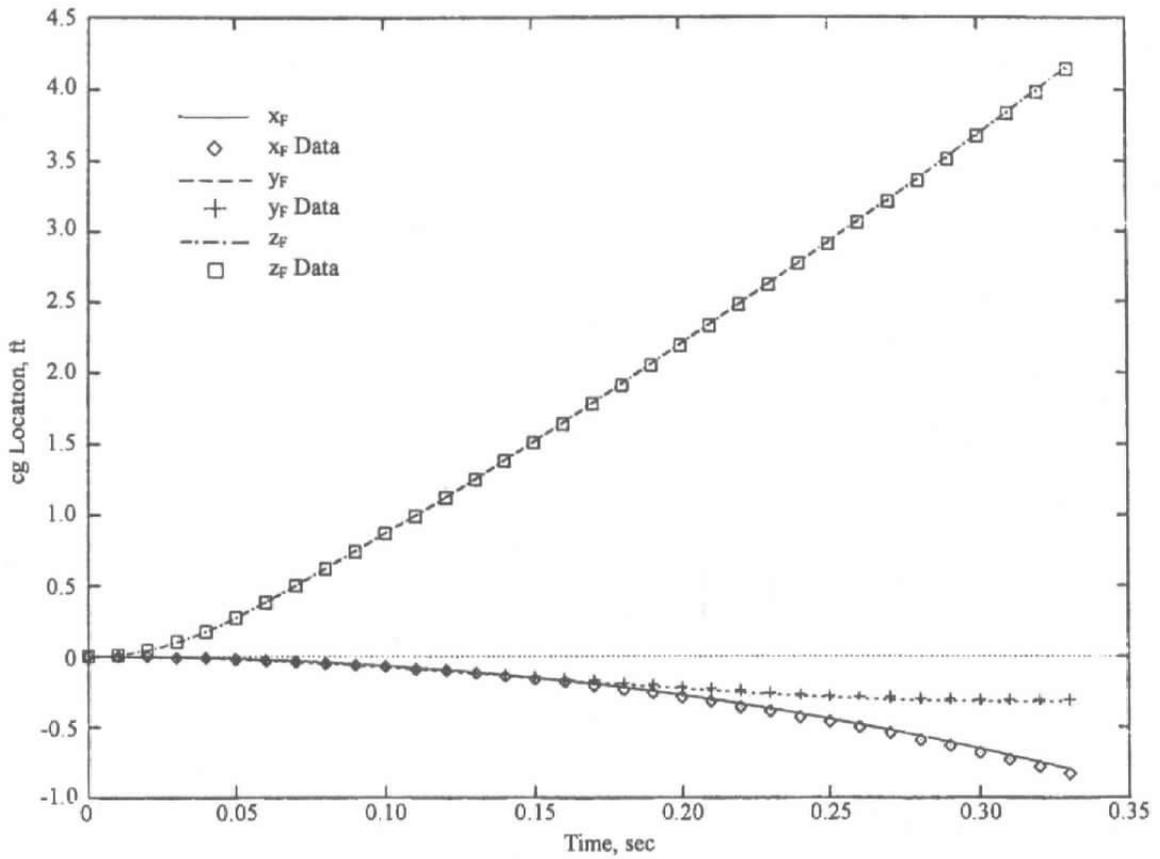
c. Perturbed ZEJCT  
Figure C-1. Concluded.



**a. Angles with respect to flight axis system**  
**Figure C-2. Trajectory sensitivity to angular rate damping.**

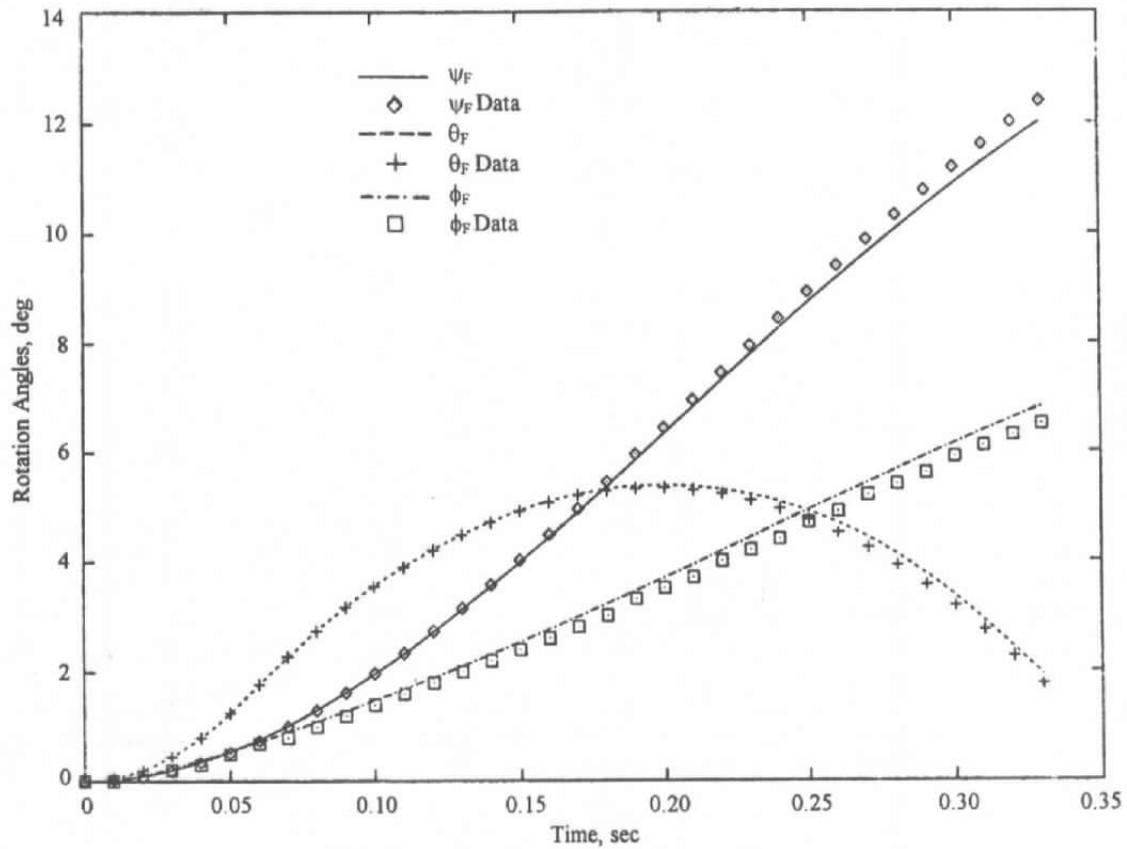


b. Rotational velocity in body axis system  
Figure C-2. Concluded.

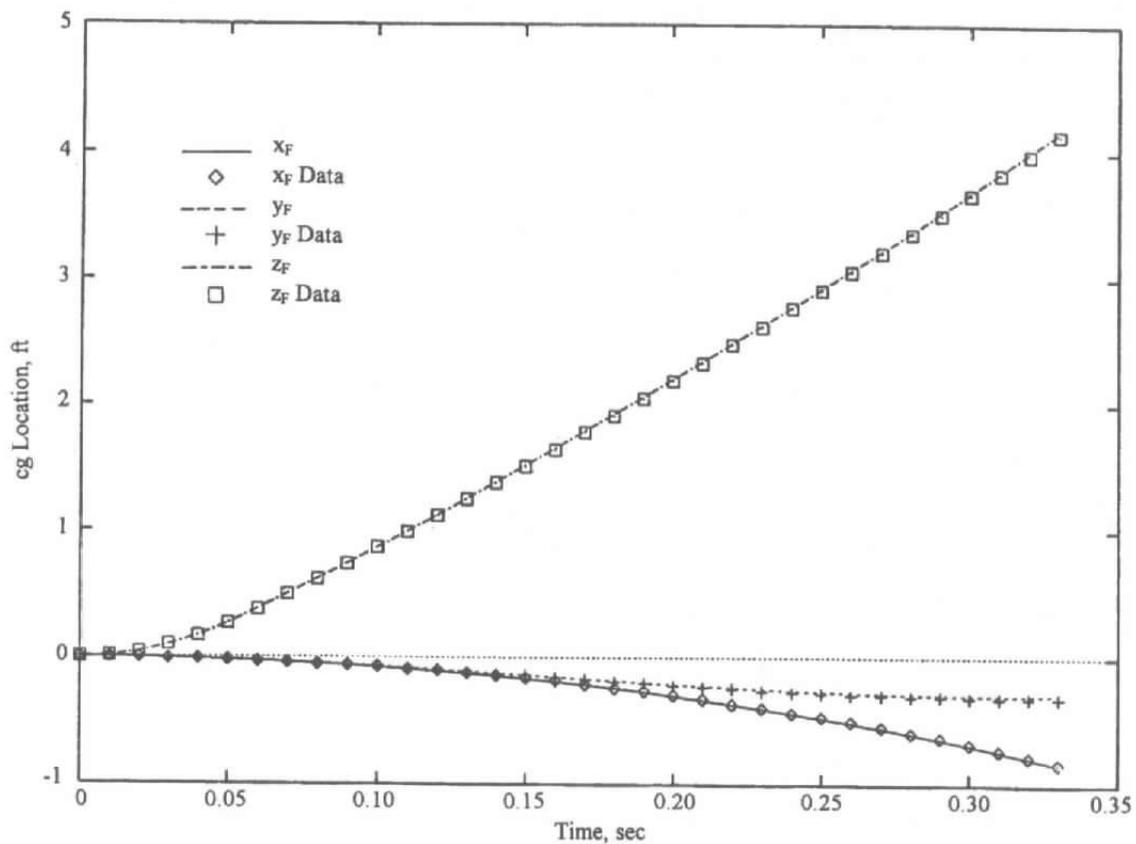


a. Translation (ISEED = -1)

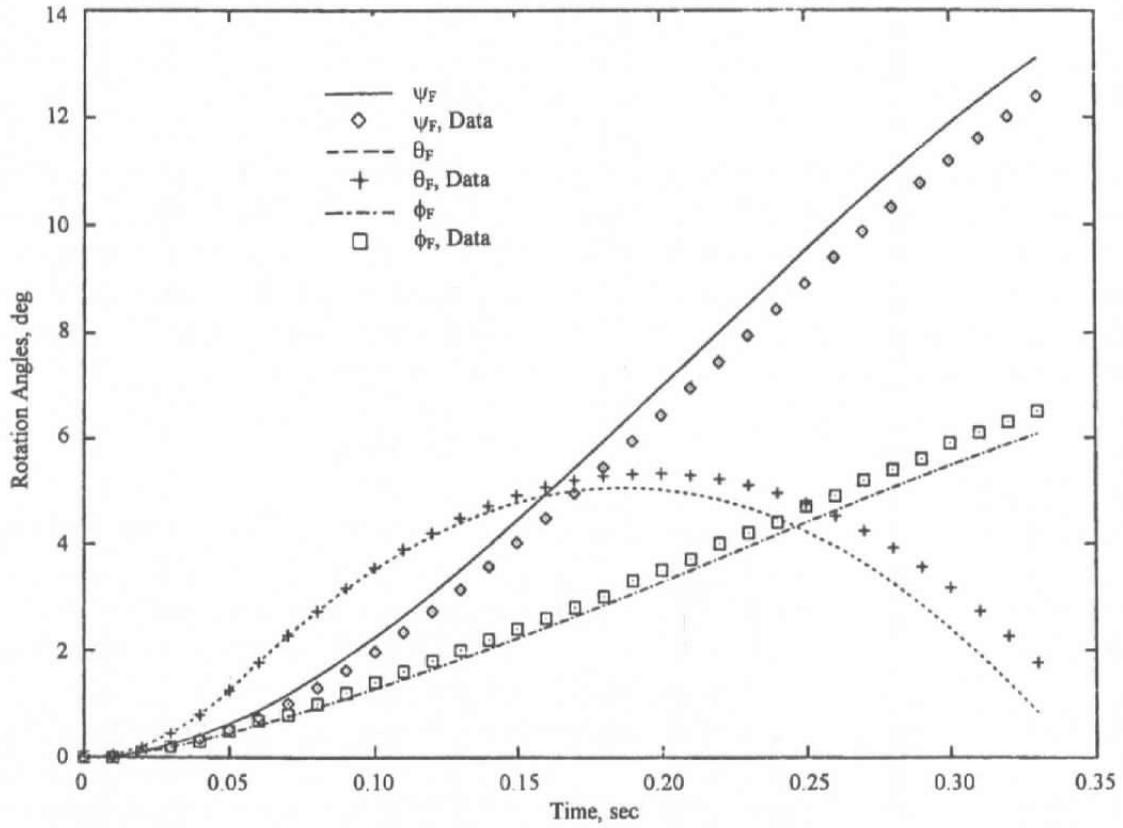
Figure C-3. Trajectory sensitivity to random errors in aerodynamic loads.



**b. Rotation (ISEED = -1)**  
**Figure C-3. Continued.**



c. Translation (ISEED = -987654321)  
 Figure C-3. Continued.



d. Rotation (ISEED = -987654321)  
 Figure C-3. Concluded.

## APPENDIX D SIXDOF USER'S MANUAL

This appendix includes a more complete description of the input and output files required by the SIXDOF code, a flowchart, a brief description of each subroutine used, and a list of inputs used for the present study. Section A.1.0 and the Nomenclature should be referred to for axis system definitions.

### D.1.0 INPUT FILES

#### **sixdof.in (unit 5)**

This file contains the basic formatted inputs for the SIXDOF code in namelist format. The file contains two namelist files (FILES and INPUT) which are read in Subroutine RSTIN. The FILES namelist gives the names of the remaining SIXDOF input files (each of which is described below), and the INPUT namelist contains physical information required for execution of the program. All inputs are full-scale, dimensional, and must have consistent units which assume  $g_c = 1$ . Once created, this file is not modified in the overall trajectory calculation process.

<b>Name</b>	<b>Description</b>
<b><u>NAMELIST \$FILES</u></b>	
<b>SXDFRST</b>	Name of the file containing restart information from the last execution of SIXDOF.
<b>COEFWRK</b>	Name of the file containing the load coefficients output by TESS.
<b>PEGORIG</b>	Name of the file containing the PEGSUS input deck that was used in the construction of the grids with the store at carriage.
<b>PEGNEW</b>	Name of the modified PEGSUS input deck which the SIXDOF code will create.
<b>LOCHST</b>	Name of the file to which SIXDOF will write the translational and rotational values at each time level.
<b>COEFHST</b>	Name of the file to which SIXDOF will write the store loads (in coefficient form) at each time level.
<b>VELHST</b>	Name of the file to which SIXDOF will write the translational and rotational velocities at each time level.
<b>PRSLOC</b>	Name of the file which SIXDOF will read for translational and rotational positions when using prescribed motion. This file is used only if IEJCT=4 in the SXDFRST file.

**NAMelist \$INPUT**

<b>Name</b>	<b>Description</b>
<b>KEYWORD</b>	Character string appearing in the PEGSUS input which marks the lines used to translate and rotate the store in the original PEGSUS input deck.
<b>DT</b>	Full-scale, dimensional time step. Note that this variable is not the same as the non-dimensional time used in XAIR, but must be consistent with the value of DT used in XAIR (see Appendix F).
<b>T0</b>	Time at release for the store.
<b>XCG0, YCG0, ZCG0</b>	Translation vector required to change from the Global Grid Axis System to the Spatial Axis System.
<b>GAMMA</b>	Gas constant.
<b>FSMACH</b>	Aircraft flight Mach number.
<b>RHO</b>	Far-field density used to calculate the dimensional loads from the dimensionless coefficients.
<b>UINF,VINF,WINF</b>	Aircraft flight velocity components used to calculate $q_{\infty}$ .
<b>IXXB,IXYB,IXZB IYYB,IYZB,IZZB</b>	Inertia tensor (which is symmetric about the diagonal, see Appendix A) in the CFD Body Axis System.
<b>WTX,WTY,WTZ</b>	Store weight vector in the Global Grid Axis System.
<b>G</b>	Gravitational acceleration.
<b>XE1B,YE1B,ZE1B</b>	Forward ejector position in the CFD Body Axis System.
<b>XE2B,YE2B,ZE2B</b>	Aft ejector position in the CFD Body Axis System.
<b>EFX1B,EFY1B, EFZ1B</b>	Force vector of the forward ejector in the CFD Body Axis System.

<b>Name</b>	<b>Description</b>
EFX2B,EFY2B, EFZ2B	Force vector of the aft ejector in the CFD Body Axis System.
ZEJCT	Piston stroke length in the Pylon Axis System.
G2PXX,G2PXY,G2PXZ, G2PYX,G2PYY,G2PYZ, G2PZX,G2PZY,G2PZZ	Rotation matrix used in conjunction with the appropriate translation appropriate translation to transform from the Global Grid Axis System to the Pylon Axis System. These variables are used only if the spatial shut-off ejector model is used, i.e., if the value of IEJCT is 1 in the SXDFRST file.
REFARE	Reference area used to define force and moment coefficients.
REFLEN	Reference length used to define moment coefficients.
GRD2RL	Scale factor to convert from the grid scale to full scale.

**COEFWRK (unit 9)**

This is a formatted file written by the TESS code at each step in the trajectory calculation process. It is read in Subroutine FORMNT and used by SIXDOF to calculate the full-scale, dimensional forces and moments on the full-scale store.

<b>Name</b>	<b>Description</b>
CZ	Force coefficient in the $z_{CB}$ direction, $CZ=(\text{force in } z_{CB})/(q_{\infty} * \text{REFARE})$ .
CMY	Moment coefficient about the $y_{CB}$ axis, $CMY=(\text{moment about } y_{CB})/(q_{\infty} * \text{REFARE} * \text{REFLEN})$ .
CY	Force coefficient in the $y_{CB}$ direction, $CY=(\text{force in } y_{CB})/(q_{\infty} * \text{REFARE})$ .
CMZ	Moment coefficient about the $z_{CB}$ axis, $CMZ=(\text{moment about } z_{CB})/(q_{\infty} * \text{REFARE} * \text{REFLEN})$ .
CMX	Moment coefficient about the $x_{CB}$ axis, $CMX=(\text{moment about } x_{CB})/(q_{\infty} * \text{REFARE} * \text{REFLEN})$ .

<b>Name</b>	<b>Description</b>
<b>CX</b>	Force coefficient in the $x_{CB}$ direction, $CX=(\text{force in } x_{CB})/(q_{\infty} * \text{REFARE})$ .
<b>FORMAT:</b>	<b>READ (9,*) CZ, CMY, CY, CMZ, CMX, CX.</b>

**SXDFRST (unit 8)**

This file contains formatted inputs for updating the SIXDOF code and is read in Subroutine RSTIN. Information from the previous time step is read from this file before the code executes and current information is written to this file once the code is finished. For the first time step, only the variables marked with a (\*) are required to be in the file.

<b>Name</b>	<b>Description</b>
<b>T*</b>	Full-scale time elapsed since release.
<b>DTP*</b>	Full-scale time step for the previous SIXDOF run. For the first time step, this variable should be set to the value used in the namelist input.
<b>YAW*</b>	Corresponds to $\eta_{GR}$ as defined in Nomenclature.
<b>PITCH*</b>	Corresponds to $v_{GR}$ as defined in Nomenclature.
<b>ROLL*</b>	Corresponds to $\omega_{GR}$ as defined in Nomenclature.
<b>IEJCT*</b>	Flag that determines the ejector model to be used. Possible values for IEJCT are: <ul style="list-style-type: none"> <li>0- No ejectors used.</li> <li>1- Ejectors which shut off after the pistons have extended a specified distance.</li> <li>4- Prescribed motion specified in the PRSLOC file.</li> </ul>
<b>XCG,YCG,ZCG</b>	The location of the store cg in the Global Grid Axis System.
<b>INTP array</b>	The value of the inertia tensor during the previous time step. In this version of the code, the inertia tensor is constant.

Name	Description
DINTP array	The time derivatives of the inertia tensor during the previous time step. The values are all zero in the current version of the code.
OMEGXP,OMEGYP, OMEGZP	Correspond with $\omega$ as defined in Nomenclature. Units in rad/(unit time).
UP,VP,WP	The velocity vector of the cg during the previous step in the Global Grid Axis System.
XE1G0, YE1G0, ZE1G0	The forward ejector position in the Global Grid Axis System prior to store release. These values must be read, but they are not used unless IEJCT=1.
XE2G0, YE2G0, ZE2G0	The aft ejector position in the Global Grid Axis System prior to store release. These values must be read, but they are not used unless IEJCT=1.
FORMAT:	<pre> READ(8,*) T, DTP READ(8,*) YAW, PITCH, ROLL READ(8,*) IEJCT IF( MOVING ) THEN   READ(8,*) XCG, YCG, ZCG   READ(8,*) INTP, DINTP   READ(8,*) OMEGXP, OMEGYP, OMEGZP, UP, VP,   WP   READ(8,*) XE1G0, YE1G0, ZE1G0, XE2G0, YE2G0,   ZE2G0 ENDIF </pre>

**LOCHST (unit 12)**

This file is a formatted file containing full-scale location history information intended for comparison to tabulated tunnel data. This file is never actually read in, but new information is appended to the end of the file in Subroutine HIST (see Sec. D.2.0). For the first time step, this file should be empty or not exist at all. The file contains the following variables (in order):

Name	Description
T	Full-scale time elapsed since store release.
X,Y,Z	Current $x_F$ , $y_F$ , $z_F$ values.
PSI	Corresponds to $\psi_F$ as defined in Nomenclature.
THETA	Corresponds to $\theta_F$ as defined in Nomenclature.
PHI	Corresponds to $\phi_F$ as defined in Nomenclature.

### COEFHST (unit 16)

This file is a formatted file containing load history information intended for comparison to tabulated tunnel data. This file is never actually read in, but new information is appended to the end of the file in Subroutine HIST (see Sec. D.2.0). For the first time step, this file should be empty or not exist at all. The file contains the following variables (in order):

Name	Description
T	Time since store release.
CN	Force coefficient in the negative $z_B$ direction. $CN = (\text{force in } -z_B) / (q_\infty * \text{REFARE})$ .
CLM	Moment coefficient about the $y_B$ axis, $CLM = (\text{moment about } y_B) / (q_\infty * \text{REFARE} * \text{REFLEN})$ .
CY	Force coefficient in the $y_B$ direction. $CY = (\text{force in } y_B) / (q_\infty * \text{REFARE})$ .
CLN	Moment coefficient about the $z_B$ axis, $CLN = (\text{moment about } z_B) / (q_\infty * \text{REFARE} * \text{REFLEN})$ .
CLL	Moment coefficient about the $x_B$ axis, $CLL = (\text{moment about } x_B) / (q_\infty * \text{REFARE} * \text{REFLEN})$ .
CA	Force coefficient in the negative $x_B$ direction, $CA = (\text{force in } -x_B) / (q_\infty * \text{REFARE})$ .

If the spatial shut-off ejectors are used (IEJCT=1) and are still functioning, this file also has an additional line at each time step with the following values (in order):

<b>Name</b>	<b>Description</b>
EJCTFZ	Full-scale ejector force in the $z_{CB}$ direction.
EJCTMY	Full-scale ejector moment about the $y_{CB}$ axis.
EJCTFY	Full-scale ejector force in the $y_{CB}$ direction.
EJCTMZ	Full-scale ejector moment about the $z_{CB}$ axis.
EJCTMX	Full-scale ejector moment about the $x_{CB}$ axis.
EJCTFX	Full-scale ejector force in the $x_{CB}$ direction.

#### **VELHST (unit 18)**

This file is a formatted file containing full-scale velocity and acceleration information intended for comparison to tabulated tunnel data. This file is never actually read in, but new information is appended to the end of the file in Subroutine HIST (see Sec. D.2.0). For the first time step, this file should be empty or not exist at all. The file contains two lines for each time step. The file contains the following variables (in order):

#### First Line:

<b>Name</b>	<b>Description</b>
T	Full-scale elapsed time since store release.
U,V,W	Velocity vector of the store relative to the origin of the Inertial Axis System in the positive $x_B$ , $y_B$ , and $z_B$ directions. These correspond to $u_I$ , $v_I$ , $w_I$ as defined in Nomenclature.
P,Q,R	Instantaneous angular velocities of the store about the positive $x_B$ , $y_B$ , and $z_B$ axes, rad/(unit time). These correspond to $p_B$ , $q_B$ , $r_B$ as defined in Nomenclature.

Second Line:

<b>Name</b>	<b>Description</b>
UDOT,VDOT,WDOT	Accelerations of the store relative to the origin of the Inertial Axis System in the positive $x_B$ , $y_B$ , and $z_B$ directions.
PDOT,QDOT,RDOT	Angular accelerations of the store about the positive $x_B$ , $y_B$ , and $z_B$ directions, $\text{rad}/(\text{unit time})^2$ .

**PEGORIG (unit 10)**

The PEGORIG file is a formatted file containing the original PEGSUS input which adds the moving grids to the non-moving grids (see Sec. 2.1.2 and Ref. 21). The file is read in Subroutine PEGOUT. The lines defining the location of the moving grids must be preceded by a line with the character 'C' followed by the KEYWORD defined in the INPUT namelist (unit 5). The purpose of this line is explained in the PEGOUT subroutine description in Sec. D.4.0. SIXDOF takes this PEGSUS input and writes a modified version which PEGSUS or ROTRANS will use to add the moving grids (at their new positions) to the non-moving grids.

**PRSLOC (unit 12)**

The PRSLOC is a formatted file containing full-scale location history information. The values from this file will be used in lieu of the translations and rotations that would arise from the loads. The file is used only if IEJCT=4 in the 'sixdof.in' file. The file contains the following variables (in order) for each time level at which the trajectory should be prescribed:

<b>Name</b>	<b>Description</b>
T	Full-scale time elapsed since store release.
X,Y,Z	Values for $x_F$ , $y_F$ , $z_F$ .
PSI	Corresponds to $\psi_F$ as defined in Nomenclature.
THETA	Corresponds to $\theta_F$ as defined in Nomenclature.
PHI	Corresponds to $\phi_F$ as defined in Nomenclature.

## D.2.0 OUTPUT FILES

### sixdof.out (unit 7)

The sixdof.out file is a formatted file used primarily to write variables to aid in debugging the code.

### PEGNEW (unit 11)

The PEGNEW file is a formatted file containing the modified PEGSUS input which will be used to add the moving grids to the non-moving grids (see Sec. 2.0 and Ref. 21). This file is written from subroutine PEGOUT.

### SXDFRST (unit 8)

This file has been described in Sec. D.1.0, and is written from subroutine RSTOUT.

#### FORMAT:

```
WRITE(8,*) T, DT, MOVING
WRITE(8,*) YAW, PITCH, ROLL
WRITE(8,*) XCG, YCG, ZCG
WRITE(8,*) INT, DINT
WRITE(8,*) OMEGX, OMEGY, OMEGZ, U, V, W
WRITE(8,*) XE1G0, YE1G0, ZE1G0, XE2G0, YE2G0, ZE2G0
```

### LOCHST (unit 12)

This file has been described in Sec. D.1.0. It is written from Subroutine HIST.

#### FORMAT:

```
WRITE(12,102) T, X, Y, Z, PSI, THETA, PHI
102 FORMAT(1X,F7.5,6F11.5)
```

**COEFHST (unit 16)**

This file has been described in Sec. D.1.0. It is written from Subroutine HIST.

**FORMAT:**

```
WRITE(16,102) T, CN, CLM, CY, CLN, CLL, CA
102 FORMAT(1X,F7.5,6F11.5)
```

If the ejector force is nonzero, then the following line is also written:

```
WRITE(16,104) EJCTFZ, EJCTMY, EJCTFY, EJCTMZ,
EJCTMX, EJCTFX
104 FORMAT(8X,6F11.0)
```

**VELHST (unit 18)**

This file has been described in Sec. D.1.0. It is written from Subroutine HIST.

**FORMAT:**

```
WRITE(18,102) T, U, V, W, P, Q, R
WRITE(18,103) UDOT, VDOT, WDOT, PDOT,
QDOT, RDOT
102 FORMAT(1X,F7.5,6F11.5)
103 FORMAT(8X,6F11.2)
```

**D.3.0 CODE STRUCTURE**

Figure D-1 depicts the hierarchical structure of the SIXDOF code. The topmost box represents the main program and each box below that represents a subroutine whose function will be described in Sec. D.4.0.

**D.4.0 SUBROUTINE DESCRIPTIONS****EJSPAC**

This subroutine models two ejector pistons pushing with a constant force that shut off after extending a specified distance. The subroutine calculates the ejector forces and moments for each ejector piston independently. The ejector models are described more fully in Appendix E.

## **EJSPRSC**

This subroutine uses an external file (PRSLOC) to specify the translational and rotational positions of the store. The subroutine overrides the solutions of the equations of motion. Refer to Appendix E for more detail.

## **FORMNT**

This subroutine calculates the full-scale forces and moments given the appropriate coefficients in the CFD Body Axis System, the weight, and the ejector forces and moments. The forces are calculated in the Global Grid Axis System while the moments are calculated in the CFD Body Axis System (see Appendix A).

## **HIST**

This subroutine performs the necessary calculations to transform code variables to the axis system required by the output and appends new information to the LOCHST (unit 12), COEFHST (unit 16), and VELHST (unit 18) files.

## **INITIA**

This subroutine is used to set variables necessary to begin code execution. Most of the variables are set using calls to RSTIN, ROTMAT, and TRAJ0 (which is called only for the first execution of SIXDOF). However, calculations for mass and conversion of angles from degrees to radians are performed directly inside this routine.

## **INV**

This subroutine calculates the inverse of the inertia tensor (see Appendix A).

## **IXYZ**

This subroutine evaluates INT, the current value of INTP (see Sec. D.1.0), by setting the diagonal elements of the inertia tensor to the input values and converting to the CFD Body Axis System. The time rate of change of INT, namely DINT, is set to zero (see Sec. D.1.0).

## OMGDOT

This subroutine computes the time rate of change of the angular velocities and the time rate of change of the Euler angles using the method outlined Appendix A. This routine is called by ROTATE to provide information required to solve the differential equations for angular velocities and Euler angles. In subroutine HIST, this routine is called to provide values for the angular accelerations to write to the file VELHST (unit 18)

## PEGOUT

This subroutine reads file PEGORIG (unit 10), then writes the PEGNEW input file (unit 11), which will be used to add the moving grids (at their new locations) to the stationary grids. One statement appears in the file that is not standard PEGSUS input. It is a line at the end of each moving grid MESH namelist (see Ref. 21) with the character 'C' followed by the KEYWORD defined in the sixdof.in file. Since 'C' appears in the first column, PEGSUS treats the line as a comment and ignores it. However, in PEGOUT, the characters are used to identify the lines that will tell PEGSUS where to place each moving grid. If the keyword does not appear, the line is read from the PEGORIG file and echoed directly to the PEGNEW file. When a line is found that does contain the keyword, the current location information that will be required by PEGSUS is written out instead of the line from the PEGORIG file.

## ROTATE

This subroutine solves the differential equations for the incremental angular velocities and the Euler angles of the store at the current time step using a fourth-order Runge-Kutta method (see Appendix A). If the prescribed motion ejector model is being used (IEJCT=4), this subroutine is not called.

## ROTMAT

This subroutine calculates a rotation matrix given the  $\eta_{GR}$ ,  $\nu_{GR}$ ,  $\omega_{GR}$  Euler angles. The rotations are performed in the  $\eta_{GR} - \nu_{GR} - \omega_{GR}$  sequence (analogous to the yaw-pitch-roll sequence).

## RSTIN

This subroutine reads the formatted input files sixdof.in (unit 5) and SXDFRST (unit 8).

**RSTOUT**

This subroutine changes yaw, pitch, and roll from radians back to degrees and writes information necessary for the next run of SIXDOF to the SXDFRST file (unit 8).

**TRAJ0**

This subroutine is called by INITIA only if the code is executing for the first time. It sets the initial cg location equal to the XCG0, YCG0, ZCG0 location read in from sixdof.in (unit 5), sets the translational and rotational velocities for the previous step to zero, sets the inertia tensor to the moments of inertia read in from sixdof.in, converts the inertia tensor to the CFD Body Axis System, and sets the time rate of change of the inertia tensor to zero.

**TRANS**

This subroutine calculates the incremental translational motion of the store based on the loads calculated by FORMNT (see Appendix A). If the prescribed motion ejector model is being used (IEJCT=4), this subroutine is not called.

**D.5.0 SINGLE STORE VALIDATION INPUTS**

The following is a list of the inputs from the sixdof.in (unit 5) and SXDFRST (unit 8) files for the single-store validation case. The basic units used are feet, seconds, and pounds force. The variable GRD2RL is used to convert from 1/20th-scale inches to full-scale feet, and RHO has units of slugs/ft<sup>3</sup>.

**UNIT 5****NAMELIST FILES**

```

SXDFRST = "sixdof.rst",
COEFWRK = "coeff.wrk",
PEGORIG = "peg.in.orig",
PEGNEW = "peg.in"
LOCHST = "loc.hist",
COEFHST = "coeff.hist",
VELHST = "vel.hist",
PRSLOC = "prs.loc",

```

## NAMELIST INPUT

DT= 0.0001,	T0 = 0.0,	KEYWORD = 'MOVE',
XCG0= 4.650000,	YCG0= 0.000000,	ZCG0= 0.000000,
GAMMA= 1.4,	FSMACH= 0.95,	RHO= 0.0010292,
UINF= 961.3,	VINF= 0.0,	WINF= 0.0,
IXXB = 20.00,	IXYB = 0.00,	IXZB = 0.00,
IYYB = 360.00,	IYZB = 0.00,	
IZZB = 360.00,		
WTX= 0.0,	WTY= 0.0,	WTZ= -2000.0,
G = 32.174,		
ZEJCT= 0.33,		
XE1B= -0.59,	YE1B= 0.0,	ZE1B= 0.8333333,
XE2B= 1.08,	YE2B= 0.0,	ZE2B= 0.8333333,
EFX1B= 0.0,	EFY1B= 0.0,	EFZ1B= -2400.00,
EFX2B= 0.0,	EFY2B= 0.0,	EFZ2B= -9600.00,
G2PXX=-1.0,	G2PXY= 0.0,	G2PXZ= 0.0,
G2PYX= 0.0,	G2PYY= 1.0,	G2PYZ= 0.0,
G2PZX= 0.0,	G2PZY= 0.0,	G2PZZ=-1.0,
REFARE= 2.18166,	REFLEN= 1.6667,	GRD2RL= 1.66666667,

UNIT 8

T=0.0, DTP=0.0001,  
 YAW=0.0, PITCH=0.0, ROLL=0.0,  
 IEJCT=1

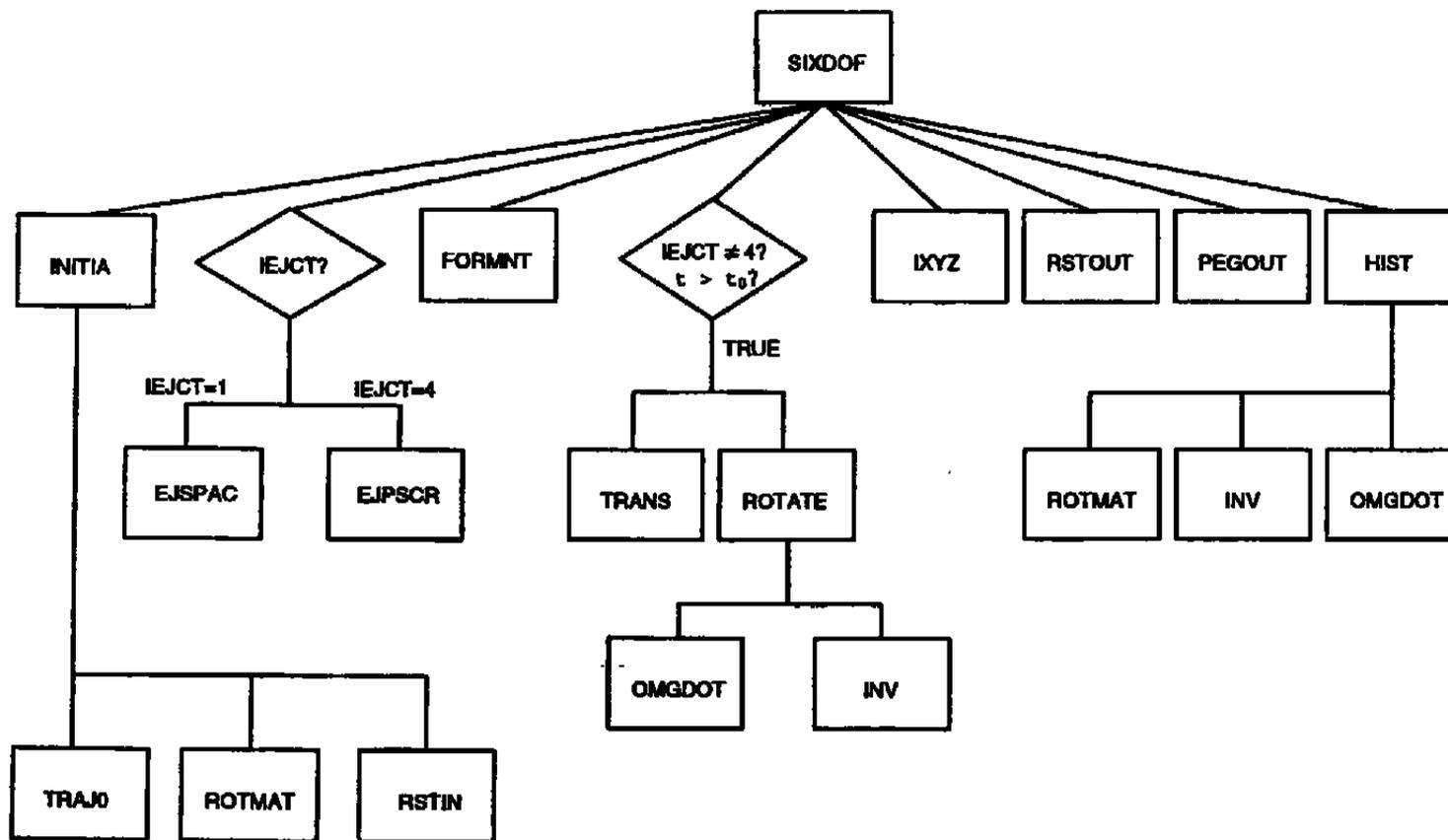


Figure D-1. SIXDOF program structure.

## APPENDIX E EJECTOR MODELS

Experience has shown that use of the correct ejector model is crucial to the accurate reproduction of CTS test data by the CFD-based trajectories. Currently, two ejector models have been implemented in program SIXDOF. The first models two ejector pistons that shut off when the pistons extend a specified distance. The second model uses an external file to specify the position of the body at the desired time levels.

### E.1.0 EJECTOR PISTONS WITH SPATIAL SHUT-OFF

The objective of this model was to reproduce faithfully the procedure followed in the CTS test. The ejector model is based upon the AEDC CTS procedure described in Appendix G of Ref. 8. However, the subroutine- EJSPEC- has been written independently of the CTS code so that it is compatible with the overall CFD-based procedure.

The following three assumptions are made in the present form of subroutine EJSPEC:

1. The model assumes that there are two ejector pistons, each of which has a force vector that is constant in the Body Axis System. Each piston acts until its stroke length is reached, at which time its force is set to zero.
2. The piston-force magnitudes and points of application are input to SIXDOF in the CFD Body Axis System at carriage and remain invariant with time in that axis system. Each piston can have arbitrary  $y_{CB}$  - and  $z_{CB}$  -components of force, but the  $x_{CB}$  -component must be zero. The points of application can have nonzero  $x_{CB}$  - and  $z_{CB}$  -components only.
3. The stroke length in the Pylon Axis System is the same for both pistons and is part of the SIXDOF input. In the present version of SIXDOF, only the  $Z_p$  -component of the piston stroke is checked in the Pylon Axis System to determine cutoff.

For the remainder of this Appendix, it is convenient to consider the discussion as part of the User's Guide in Appendix D. Therefore, the forces, moments, and coordinates are expressed as the Fortran variables used in Subroutine EJSPEC and EJPRSC.

The components of the forward ejector piston force in the CFD Body Axis System are EFX1B, EFY1B, and EFZ1B in the  $x_{CB}$  -,  $y_{CB}$  -, and  $z_{CB}$  -coordinate directions, respectively. The

coordinates of the point of application are XE1B, YE1B, and ZE1B. The corresponding quantities for the aft ejector are EFX2B, EFX2B, EFZ2B, XE2B, YE2B, and ZE2B. As mentioned above, EFX1B, EFX2B, YE1B, and YE2B must be zero for the present version of Subroutine EJSPAC.

Other quantities provided to Subroutine EJSPAC are the components of the initial location of the store center of gravity in the Global Grid Axis System, namely XCG0, YCG0, and ZCG0, respectively. At the beginning of the first time step, the components of the instantaneous location of the center of gravity, namely XCG, YCG, and ZCG, are set equal to the initial values above. At each subsequent time step, XCG, YCG, and ZCG are updated by the integration procedure. The stroke length of the piston, ZEJCT, is also given. The final data provided are the transformation matrices G2P (from the Global Grid Axis System to the Pylon Axis System), and the current G2B (from the Global Grid Axis System to the CFD Body Axis System), see Appendix A.

The first step in Subroutine EJSPAC is to calculate the ejector moments EMX1B, EMY1B, EMZ1B, EMX2B, EMY2B, and EMZ2B arising from the ejector forces. The moments are calculated directly in the CFD Body Axis System from the ejector forces and points of application.

The next step is to calculate the current location of the points of application of the pistons in the Global Grid Axis System. The components XE1G, YE1G, ZE1G, XE2G, YE2G, and ZE2G are evaluated from XE1B, etc. and XCG, etc. using Eq. (A-2).

If the integration is at the beginning of the first time step, the initial values XE1G0, YE1G0, ZE1G0, XE2G0, YE2G0, and ZE2G0 of XE1G, etc. are evaluated by equating XE1G0 to XE1G, etc. Then, for all time steps including the first, the increments DXE1G, DYE1G, DZE1G, DXE2G, DYE2G, and DZE2G are evaluated in the Global Grid Axis System by  $DXE1G = XE1G - XE1G0$ , etc. These increments are the total distances that each of the pistons has moved from  $t=t_0$  until this time step. The increments are transformed to DXE1P, DYE1P, DZE1P, DXE2P, DYE2P, and DZE2P in the Pylon Axis System according to the equation

$$d\vec{r}_p = [G2P]d\vec{r}_{GR}$$

where the components of  $d\vec{r}_p$  are DXE1P, etc., and of  $d\vec{r}_{GR}$  are DXE1G, etc.

Next, DZE1P and DZE2P are checked to see if they are greater than ZEJCT. If DZE1P is greater than ZEJCT, the forward ejector forces (EFX1G, EFX1G, and EFZ1G) and moments (EMX1B, EMY1B, and EMZ1B) are set to zero. If DZE1P is less than or equal to ZEJCT, the force components in the Global Grid Axis System are found by the transformation

$$\vec{F}_{GR} = [G2B]^T \vec{F}_{CB}$$

where the components of  $F_{GR}$  are EFX1G, etc., and of  $F_{CB}$  are EFX1B, etc. The moment components are unchanged because they are already in the CFD Body Axis System. The identical procedure is used for the aft ejector forces and moments. Since the store generally rotates after release, it is likely that each ejector will shut off at a different time. If this is the case, the forces and moments of the appropriate ejector piston are set to zero, but the forces and moments of the other piston are applied until its stroke is completed.

The force and moment components of each piston are then summed to get the total ejector-force components (EJCTFX, EJCTFY, and EJCTFZ) in the Global Grid Axis System and the total ejector-moment components (EJCTMX, EJCTMY, and EJCTMZ) in the CFD Body Axis System. This completes both a call to Subroutine EJSPAC and the description of the model for the ejector forces and moments. In SIXDOF, the ejector-force components are added to the aerodynamic- and gravity-force components to determine the total force in Eq. (A-6). Similarly, the ejector-moment components are added to the aerodynamic-moment components to determine the total external moment in Eq. (A-8).

When both ejectors have extended the specified distance, the ejectors are shut off (the IEJCT variable is set to 0 in the SXDFRST file, see Appendix D), and the motion is thereafter determined by the aerodynamic loads.

## E.2.0 PRESCRIBED MOTION

The objective of this model was to enable the user to specify the motion of the body if data were available and more desirable to use. This model could be used for situations other than store separation. For example, the model could be used to simulate the motion of a pitching airfoil or maneuvering aircraft.

The following assumptions are made in the present formulation of subroutine EJPRSC:

1. The trajectory will be supplied via an external file which will contain the  $x_F$ ,  $y_F$ ,  $z_F$ ,  $\psi_F$ ,  $\theta_F$ , and  $\phi_F$  values at each time level for which the motion is to be specified. Note that the position is defined as full-scale values in the Flight Axis System and that the entire timespan of interest (including  $t=0$ ) should be included.
2. The time increment must be constant throughout the specified motion and must match the time increment specified in the SIXDOF input.

Inside subroutine EJPRSC, the first step is to read the position for the current time level from the PRSLOC file described in Appendix D. Using the time level and the time increment, the proper line of the PRSLOC file is identified and read. The  $x_F$ ,  $y_F$ ,  $z_F$ ,  $\psi_F$ ,  $\theta_F$ , and  $\phi_F$  values are then converted to the Global Grid System. Derivatives of  $\vec{E}$  are calculated with a one-sided difference using the current and previous values of  $\vec{E}$ . The rotational velocity components in the CFD Body Axis System are then determined using Eq. (13), i.e.,

$$\vec{\omega} = T\dot{\vec{E}}.$$

The translational velocities are approximated by a one-sided difference of the current and previous values of  $x_{GR}$ ,  $y_{GR}$ , and  $z_{GR}$ . This completes a call to EJPRSC. This ejector model is used until the end of the PRSLOC file is detected in subroutine EJPRSC. At that point, the ejector model is shut off (the value for IEJCT is reset to 0, see Appendix D), and thereafter the aerodynamic loads determine the motion of the body.

**APPENDIX F**  
**RELATIONSHIP OF XAIR TIME INCREMENT**  
**AND REAL-TIME INCREMENT**

A time step value is set in both the SIXDOF code and the XAIR code. Although these values are different, they are related because the value for SIXDOF is a dimensional number and the value for XAIR is nondimensional. The time step used in XAIR is chosen based on stability of the code; therefore, the time step used in SIXDOF must be calculated from the time step used in XAIR. The procedure for the determination of the time step used in SIXDOF is herein documented for future reference.

The equations in SIXDOF are dimensional because the mass and moments of inertia of the store (which are used to determine the accelerations) are dependent on the geometry of the store and are difficult to nondimensionalize. On the other hand, the conservation equations of XAIR describe a field surrounding an object; the mass and moments of inertia are of no consequence. In this form, density, velocity, and energy are field properties; consequently, the equations can be rendered dimensionless, thereby having a one-to-one correspondence with an infinity of solutions. A particular dimensional solution can be defined from the dimensionless flow variables by using the dimensional reference variables. The value of the time step in XAIR is nondimensionalized by the free-stream speed of sound,  $a_\infty$ , and a reference length,  $L$ . The time step is made dimensionless using the formula

$$DT = \Delta t a_\infty / L, \quad (F-1)$$

where  $DT$  is the nondimensional time step used in XAIR and  $\Delta t$  is the dimensional time step used in SIXDOF.

For the single-store case presented in Sec. 3.2, the computational meshes were constructed to model a 1/20-scale store which used inches as the length scale. To obtain the full-scale, dimensional time step for the nondimensional  $DT$  of 0.06, values of  $L = 20.0$  in. and  $a_\infty = 12,143$  in./sec. are used to compute a value of  $\Delta t = 0.0001$  sec. The value for  $L$  relates the length unit in the computational mesh to the equivalent dimensional length that will be used in the SIXDOF code.

## APPENDIX G AXIS SYSTEMS

Many coordinate systems are used in store separation testing and analysis. This appendix defines all the systems which are referenced in this report. A sketch of each of the systems defined below is included in Fig. G-1.

### Global Grid Axis System

Coordinate Directions:

- $x_{GR}$  Parallel to  $V_{\infty}$ , positive rearward.
- $y_{GR}$  Perpendicular to  $x_{GR}$  and  $z_{GR}$ , positive to the right as viewed by the pilot.
- $z_{GR}$  Perpendicular to the  $x_{GR}$  and  $y_{GR}$  axes, positive upward as viewed by the pilot.

Origin:

The origin is placed arbitrarily as defined by the composite grid output by PEGSUS. The Global Grid Axis System origin translates along the initial flight path direction at  $V_{\infty}$ .

Attitudes:

Given a basis with axes  $x$ ,  $y$ , and  $z$  that is initially co-located with the basis of the Global Grid Axis System,

- $\eta_{GR}$  The rotation angle about the  $z$  axis, positive in the right-handed sense (yaw).
- $\nu_{GR}$  The rotation angle about the new  $y$  axis of the basis that has been rotated through  $\eta_{GR}$ , positive in the right-handed sense (pitch).
- $\omega_{GR}$  The rotation angle about the new  $x$  axis of the basis that has been rotated through  $\eta_{GR}$ , then  $\nu_{GR}$ , positive in the right-handed sense (roll).

Alternatively, the angles can be defined as:

- $\eta_{GR}$  Angle between the projection of the store longitudinal ( $x_{CB}$ ) axis in the  $x_{GR}-y_{GR}$  plane and the  $x_{GR}$  axis, positive in the right-handed sense.
- $\nu_{GR}$  Angle between the store longitudinal ( $x_{CB}$ ) axis and its projection in the  $x_{GR}-y_{GR}$  plane, positive when the store nose is raised as viewed by the pilot.
- $\omega_{GR}$  Angle between the store lateral ( $y_{CB}$ ) axis and the intersection of the  $y_{CB}-z_{CB}$  and  $x_{GR}-y_{GR}$  planes, positive for counterclockwise rotation as seen by the pilot.

### Spatial Axis System

Coordinate Directions:

$x_{SP}, y_{SP}, z_{SP}$  The Spatial Axes are parallel to the respective Global Grid Axes.

Origin:

The origin of the system is at the cg of the store and translates with the store, but the coordinate axes do not rotate with the store.

Attitudes:

As per the Global Grid Axis System.

### CFD Body Axis System

Coordinate Directions:

The axes are parallel to a basis originally co-located with the Spatial Axis System that has been rotated through  $\eta_{GR}$ , then  $\nu_{GR}$ , then  $\omega_{GR}$ . At carriage, the axes directions can be defined as:

$x_{CB}$  Parallel to the store longitudinal axis positive rearward from the pilot's view.

$y_{CB}$  Perpendicular to  $x_{CB}$  and  $z_{CB}$ , positive to the right from the pilot's point of view.

$z_{CB}$  Perpendicular to the  $x_{CB}$  and  $y_{CB}$  axes, positive upward from the pilot's point of view.

Origin:

The CFD Body Axis System origin is coincident with the store cg at all times. The coordinate axes rotate with the store so that (in this system) the inertia tensor does not vary in time.

### Body Axis System

Coordinate Directions:

The axes are parallel to a basis originally co-located with the Flight Axis System that has been rotated through  $\psi_F$ , then  $\theta_F$ , then  $\phi_F$ . At carriage, the axes directions can be defined as:

$x_B$  Parallel to the store longitudinal axis, the positive direction is forward when the store is at carriage.

$y_B$  Perpendicular to the  $x_B$  and  $z_B$  axes, positive to the right as viewed by the pilot when the store is at zero  $\psi_F$  and  $\phi_F$ .

$z_B$  Orthogonal to the  $x_B$  and  $y_B$  directions and positive downward (as viewed by the pilot) when the store is at zero  $\psi_F$  and  $\phi_F$ .

Origin:

The Body Axis System origin is coincident with the store cg at all times. The coordinate axes rotate with the store so that (in this system) the inertia tensor does not vary in time.

Instantaneous Rotational Velocity:

$p_B$  Rotational velocity component in the positive sense about the  $x_B$  axis.

$q_B$  Rotational velocity component in the positive sense about the  $y_B$  axis.

$r_B$  Rotational velocity component in the positive sense about the  $z_B$  axis.

**Instantaneous Force and Moment Coefficients:**

- $C_A$  Axial-force coefficient, positive in the negative  $x_B$  direction.
- $C_Y$  Side-force coefficient, positive in the positive  $y_B$  direction.
- $C_N$  Normal-force coefficient, positive in the negative  $z_B$  direction.
- $C_l$  Rolling-moment coefficient, positive about the  $x_B$  axis.
- $C_m$  Pitching-moment coefficient, positive about the  $y_B$  axis.
- $C_n$  Yawing-moment coefficient, positive about the  $z_B$  axis.

**Pylon Axis System****Coordinate Directions:**

- $x_p$  Parallel to the store longitudinal axis at carriage and at constant angular orientation with respect to the current aircraft flight path direction, positive forward.
- $y_p$  Perpendicular to the  $x_p$  and  $z_p$  axes, positive to the right as viewed by the pilot.
- $z_p$  Perpendicular to the  $x_p$  and  $y_p$  directions, positive downward as viewed by the pilot.

**Origin:**

The Pylon Axis System origin is coincident with the store cg at carriage. It is fixed with respect to the aircraft and thus translates along the current aircraft flight path at the flight velocity.

**Flight Axis System****Coordinate Directions:**

- $x_F$  Parallel to the current flight path direction, positive forward, as viewed by the pilot.

- $y_F$  Perpendicular to the  $x_F$  and  $z_F$  axes, positive to the right as viewed by the pilot.
- $z_F$  Parallel to the aircraft plane of symmetry and perpendicular to the current aircraft flight path direction, positive downward as viewed by the pilot.

**Origin:**

The Flight Axis System origin is coincident with the store cg at carriage. It is fixed with respect to the aircraft and thus translates along the current aircraft flight path at the aircraft flight velocity. The coordinate axes rotate to maintain alignment with the aircraft flight direction.

**Attitudes:**

Given a basis with axes  $x$ ,  $y$ , and  $z$  that is initially co-located with the basis of the Flight Axis System,

- $\psi_F$  The rotation angle about the  $z$  axis, positive in the right-handed sense.
- $\theta_F$  The rotation angle about the new  $y$  axis of the basis that has been rotated through  $\psi_F$ , positive in the right-handed sense.
- $\phi_F$  The rotation angle about the new  $x$  axis of the basis that has been rotated through  $\psi_F$ , then  $\theta_F$ , positive in the right-handed sense.

Alternatively, the angles can be defined as:

- $\psi_F$  Angle between the projection of the store longitudinal ( $x_B$ ) axis in the  $x_F-y_F$  plane and the  $x_F$  axis, positive in the right-handed sense.
- $\theta_F$  Angle between the store longitudinal ( $x_B$ ) axis and its projection in the  $x_F-y_F$  plane, positive when the store nose is raised as viewed by pilot.
- $\phi_F$  Angle between the store lateral ( $y_B$ ) axis and the intersection of the  $y_B-z_B$  and  $x_F-y_F$  planes, positive for clockwise rotation as seen by the pilot.

## **Inertial Axis System**

### **Coordinate Directions:**

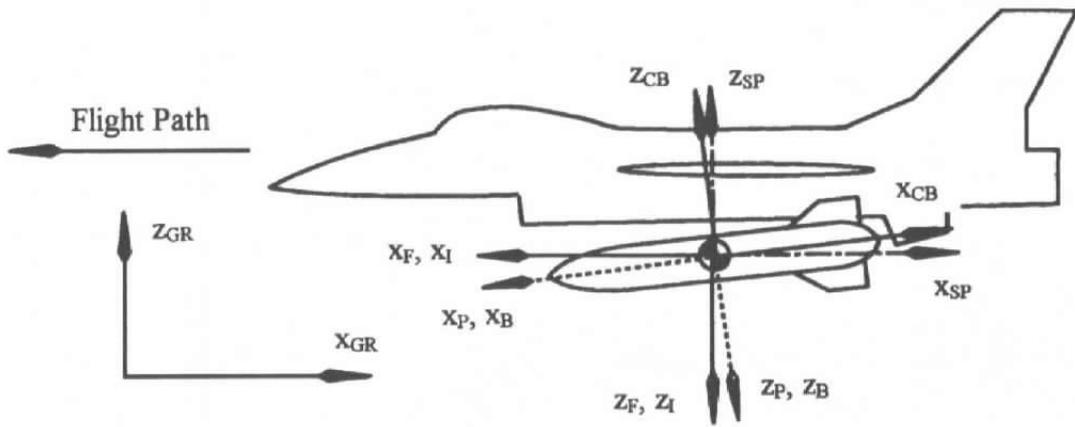
- $x_I$  Parallel to the flight path direction at store release, positive forward.
- $y_I$  Perpendicular to the  $x_I$  and  $z_I$  axes, positive to the right when looking down the positive  $x_I$  axis.
- $z_I$  Parallel to the aircraft plane of symmetry and perpendicular to the aircraft flight path direction at store release, positive downward as viewed by the pilot.

### **Origin:**

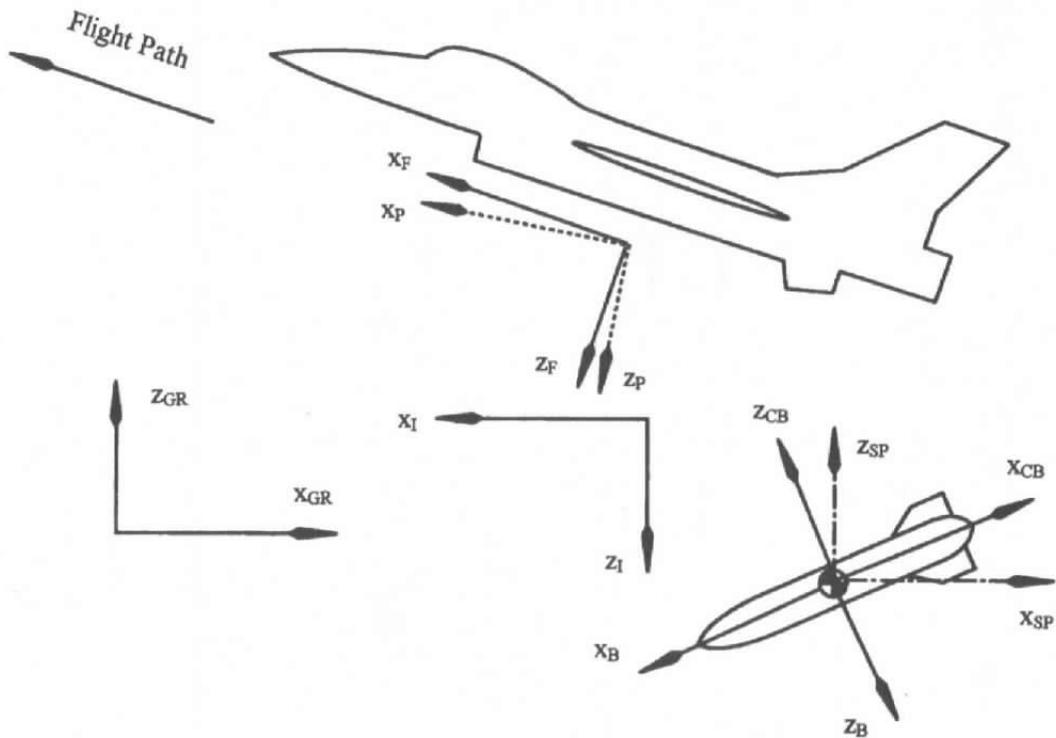
The Inertial Axis System origin is coincident with the store cg at release and translates along the initial flight path direction at the aircraft flight velocity. The coordinate axes do not rotate to maintain alignment with the aircraft flight direction.

### **Instantaneous Translational Velocity:**

- $u_I$  Instantaneous translational velocity component relative to the origin of the Inertial Axis System, but in the instantaneous  $x_B$  direction.
- $v_I$  Instantaneous translational velocity component relative to the origin of the Inertial Axis System, but in the instantaneous  $y_B$  direction.
- $w_I$  Instantaneous translational velocity component relative to the origin of the Inertial Axis System, but in the instantaneous  $z_B$  direction.



a. Systems at carriage



b. Systems after store release and aircraft maneuver

Figure G-1. Axis systems.

## NOMENCLATURE

$A_{ref}$	Reference area
$c$	Chord length of an airfoil
$C_A, C_Y, C_N$	Force coefficients in the $-x_B, y_B, -z_B$ directions, respectively (see Appendix G) $C_A = (\text{Force in } -x_B) / (q_\infty A_{ref})$ $C_Y = (\text{Force in } y_B) / (q_\infty A_{ref})$ $C_N = (\text{Force in } -z_B) / (q_\infty A_{ref})$
$C_l, C_m, C_n$	Moment coefficients about the $x_B, y_B, z_B$ axes, respectively (see Appendix G) $C_l = (\text{Moment about } x_B) / (q_\infty A_{ref} L_{ref})$ $C_m = (\text{Moment about } y_B) / (q_\infty A_{ref} L_{ref})$ $C_n = (\text{Moment about } z_B) / (q_\infty A_{ref} L_{ref})$
$cg$	Center of gravity
$\vec{CG}_{GR}$	Position vector of the store $cg$ referenced from the origin of the Global Grid Axis System (see Fig. A-1a)
$DT$	Non-dimensional time step used in the flow solver, XAIR
$\vec{E}$	Euler angle vector
$\vec{F}$	Total force vector acting on the store in the Global Grid Axis System
$g_c$	Proportionality factor, the value of which depends on the units being used
$G2B$	Transformation matrix to transform a vector quantity from the Global Grid Axis System to the CFD Body Axis System
$G2P$	Transformation matrix to transform a vector quantity from the Global Grid Axis System to the Pylon Axis System
$I$	Inertia tensor computed in the Body Axis System, Eq. (A-10)

$\bar{L}$	Angular momentum vector of the store in the CFD Body Axis System, Eq. (A-9)
$L_{ref}$	Reference length
$M_\infty$	Free-stream Mach number
$m$	Store mass
$\vec{N}$	External moment vector about the store cg in the CFD Body Axis System
$p_B, q_B, r_B$	Rotational velocity components about the $x_B, y_B,$ and $z_B$ axis, respectively (see Appendix G)
$q_\infty$	Dynamic pressure, $q_\infty = \frac{1}{2}\rho_\infty V_\infty^2$
$\vec{r}$	Position vector, Fig. A-1a
$T$	Transformation matrix used to relate $\vec{\omega}$ to Eqs. (A-13) and (A-14)
$t$	Dimensional time
$t_0$	Dimensional time at the instant of store release
$u_1, v_1, w_1$	Velocity components in the $x_1, y_1,$ and $z_1$ directions, respectively (see Appendix G)
$\vec{v}$	Translational velocity vector of the store in the Global Grid Axis System
$V_\infty$	Aircraft flight velocity
$x, y, z$	Coordinate system axes (see Appendix G)
$\alpha$	Angle of attack
$\tau$	The nondimensional time required for a given flow to pass half the chord of an airfoil, $\tau = 2 * V_\infty * t / c$
$\omega_{GR}, \nu_{GR}, \eta_{GR}$	Euler angles about the Global Grid and Spatial X, Y, and Z axes, respectively. Rotations are performed in $\omega_{GR}, \nu_{GR}, \eta_{GR}$ order (i.e., yaw, pitch, roll). See the Global Grid Axis System description in Appendix G.

$\rho_\infty$	Far-field density
$\vec{\omega}$	Angular velocity vector of the store in the CFD Body Axis System
$\psi, \theta, \phi$	Euler angles defined in the Flight Axis System. Rotations are performed in order. See the Flight Axis description in Appendix G.
$d(\ )$	Difference of a quantity evaluated at time level $n+1$ minus the evaluation at $t_0$
$\Delta t$	Dimensional time step
$\Delta(\ )$	Difference of a quantity evaluated at time level $n+1$ minus the evaluation at time level $n$
$(\dot{\ })$	Denotes a time derivative
<b>Superscripts</b>	
$(\ )^r$	Evaluated at the previous time level
$(\ )^{n+1}$	Evaluated at the current time level
$\{ \}^T$	Transpose of the matrix
$[ \ ]^{-1}$	Inverse of the matrix
<b>Subscripts</b>	
$(\ )_B$	Evaluated in the Body Axis System
$(\ )_{CB}$	Evaluated in the CFD Body Axis System
$(\ )_F$	Evaluated in the Flight Axis System
$(\ )_{GR}$	Evaluated in the Global Grid Axis System
$(\ )_I$	Evaluated in the Inertial Axis System
$(\ )_P$	Evaluated in the Pylon Axis System
$(\ )_{SP}$	Evaluated in the Spatial Axis System