JOURNAL OF AIRCRAFT
Vol. 31, No. 4, July–Aug. 1994

Time-Accurate Computational Fluid Dynamics Approach to Transonic Store Separation Trajectory Prediction

Lawrence E. Lijewski*
Wright Laboratory, Eglin Air Force Base, Florida 32542
and
Norman E. Suhs†
Calspan Corporation, Arnold Air Force Base, Tennessee 37389

A computational fluid dynamic technique is demonstrated for a time-accurate store separation event at transonic speeds. An overlapping grid approach is used coupled with an implicit Euler flow solver with a flux-difference split scheme based on Roe's approximate Riemann solver and a six-degree-of-freedom trajectory code. All major trends of the trajectory are captured and surface pressure distributions are predicted. Three sources of error are identified and discussed.

Nomenclature

CLL	=	rolling moment coefficient
CLN	=	yawing moment coefficient
CM	=	pitching moment coefficient
CN	=	normal force coefficient
Cp	=	pressure coefficient
CX	=	axial force coefficient
CY	=	yaw force coefficient
DT	=	real time step
DTMIN	=	nondimensional time step
p, q, r	=	roll, pitch, yaw angular rates
t	=	time
u, v, w	=	linear velocities
X/L	=	body axial location/body length
x, y, z	=	spacial directions
theta	=	body axis pitch angle
phi	=	body axis roll angle
PHI	=	store circumferential angle
psi	=	body axis yaw angle

Subscripts

B	=	body axis system
F	=	flight axis system

Introduction

THE carriage and release of stores from fighter aircraft are of primary importance to the Air Force. Accurate prediction of the trajectory of a store released from an aircraft is critical in assessing whether the store can be released safely as well as if it will accurately reach its target. The trajectory of stores released in aircraft flowfields has always been difficult to predict. Traditionally, the task has been left to wind-tunnel testing to obtain the necessary data. Typically, numerous wind-tunnel and flight tests are performed to obtain sufficient carriage and trajectory data for a store to be certified for Air Force use. This process can take up to several years and is required for each loading configuration of a store on a particular aircraft. However, with the advancement of computational fluid dynamic (CFD) techniques, the prediction of carriage and trajectory data is possible. These techniques can now overcome the long lead times of the wind tunnel and provide comparable results useful in aircraft/store analyses.

Efforts to help reduce the time and cost required to certify a store for use are now beginning to be impacted by the use of computational fluid dynamics. Steady-state CFD methods[1–3] have been used in conjunction with the semi-empirical methods[4] to predict safe release. Critical to an accurate trajectory are the surface pressure distributions and the resulting forces and moments acting on the store when in the captive position. Various researchers have shown encouraging predictions of steady-state surface pressure distributions and store loads in interference flowfields. These range from carriage or near-carriage predictions[5–7] to mutually interfering multiple stores.[8] Still, with the advancements made so far, these capabilities may not be sufficient when trying to predict trajectories for highly dynamic store separation cases. Particularly, the store released from within weapons bays, multiple store releases, fuel tank releases, and releases during maneuvers are store separation cases that are very difficult or currently impossible to simulate within the wind tunnel. A time-accurate computation is therefore required to sufficiently predict the trajectory of the store. The purpose of this article is to demonstrate the accuracy and technique of a time-accurate CFD approach to predict the trajectory of a finned body released from a generic wing-pylon configuration at transonic speeds.

Approach

The pacing item in conducting a time-accurate computational simulation of a store separation event is the generation and dynamic movement of the system of grids. The overall approach used to simulate this time-accurate movement is the chimera embedded grid methodology.[9,10] The chimera methodology is a domain decomposition technique that allows the computational domain to be divided into simpler overlapping or embedded regions for which grids are more easily constructed. The chimera methodology is particularly suited for movement in that grids for different parts of the configuration can be moved relative to one another to simulate the unsteady body dynamics.[11–13] Although this requires no regeneration of the grids, the communication among the grids must be recomputed for each movement or time step.

Basically, there are three processes that must be performed for each time step. First, the loads on the moving body must

Presented as Paper 92-4569 at the AIAA Atmospheric Flight Mechanics Conference, Hilton Head, SC, Aug. 10–12, 1992; revision received Sept. 16, 1992; accepted for publication Sept. 16, 1992.

*Senior Scientist, Armament Directorate, Weapon Airframe Research Section, Flight Vehicles Branch, Weapon Flight Mechanics Division. Senior Member AIAA.

†Senior Research Engineer, AEDC Operations, Computational Fluid Dynamics Section.

be calculated and then used to determine the new location and orientation of the body. This process requires a trajectory computer code that integrates the six-degree-of-freedom equations of motion. The second process begins by actually moving the grids to their new locations and computing the communications among all of the grids. The chimera domain decomposition code or domain connectivity algorithm is used to move the grids and determine the grid communications. In the third process, the flow solver is used to calculate one time step of the unsteady flow. How these processes are integrated into one system will be discussed after each process is described.

Trajectory Code

The trajectory code used in the three-process store separation system is a six-degree-of-freedom code used by the Air Force for many years and is similar to others[14] in the field. This code integrates the equations of motion using a fourth-order Runge-Kutta method and provides results in up to eight different axis systems. Major inputs include aircraft and store physical characteristics, ejector characteristics, and initial flight conditions. These inputs are fixed at time zero with the only time-varying input being the six store force and moment coefficients. The code outputs the store c.g. location and angular orientation as input to the domain decomposition code. Execution time for the trajectory code is less than 1 s.

Chimera Domain Decomposition Code

The domain decomposition code takes individual meshes along with appropriate user inputs and creates the communication or connectivity among the meshes. Trilinear interpolation accounts for the transfer of information from one mesh to another. Output includes a standardized file that contains the interpolation data as well as a file with the meshes. These files are transferred to the flow solver to perform the computation.

The current version of the domain decomposition code[15] allows for a restart capability that greatly enhances its efficiency for a moving mesh problem. For previous versions, the communication among the meshes was completely recomputed for each time step. Now, only the communications changed by the moving grids must be recomputed.

Flow Solution Code

The flow solver is an implicit Euler algorithm described by Belk and Whitfield.[16] The algorithm is a flux difference split scheme based on Roe's approximate Riemann solver. The version of this code has the appropriate logic to utilize the chimera embedded grid methodology,[6] and includes the time metric terms[17] to account for the movement of the mesh. The boundary conditions are set for either steady-state[18] or dynamic[19] conditions. The code obtains the flow solution for one time step based on the newly moved grids received from the domain connectivity model, and outputs the new force and moment coefficients acting on the store to the six-degree-of-freedom model. The flow solver runs at a speed of 4.78×10^{-5} s/point/iteration on a Cray Y-MP supercomputer.

Computational Process

The overall computational process (Fig. 1) is illustrated for a typical store separation event. The process is divided into two parts: 1) a steady-state part and 2) a dynamic part. The steady-state part has three steps. The first step determines the communication among all of the nonmoving meshes using the domain decomposition code. In step 2 the meshes that will move are added to the nonmoving meshes of step 1. A steady-state flow solution, step 3, is computed for the entire configuration from step 2. This steady-state flow solution gives the initial conditions for the dynamic computations.

The dynamic part starts with step 4 where the new location for the moving body is determined using the trajectory code with the steady-state carriage forces and moments as input.

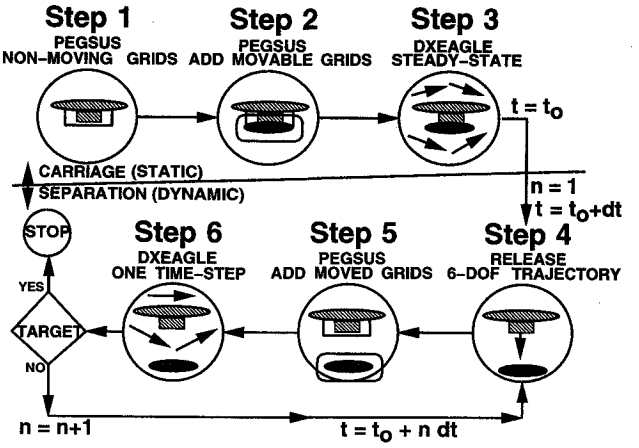

Fig. 1 Store separation computation process.

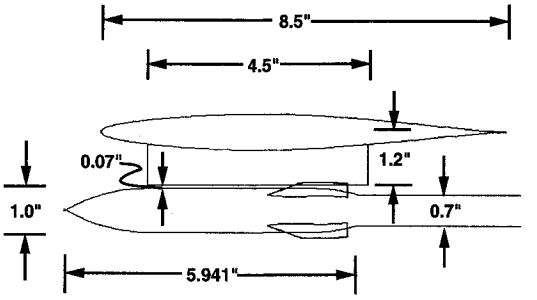

Fig. 2 Configuration geometry.

Once the new location of the moving body or bodies is determined, the domain decomposition code is executed, step 5, by adding the moving body at its new location to the nonmoving meshes of step 1. Next, the flow solver is executed for one time step on each of the meshes, step 6. The time-accurate trajectory prediction is completed by repeating steps 4, 5, and 6 until the target is reached. This target may be a specified location or time.

A modification to the dynamic portion of the above approach was implemented in the interest of reducing the overall computational time. The domain decomposition code was not executed each time step, but periodically. This means that during the intermediate time steps, although the grids were moved, the communication among the grids remained the same. Since the movement was small, the resulting error introduced into the flow solution would be small.

Configuration Geometry and Grid Generation

The wing-pylon-finned body configuration consisted of a clipped delta wing with a 45-deg sweep and a constant NACA 64A010 airfoil section. The pylon had an ogive-flat plate-ogive cross section shape, whereas the store body was an ogive-cylinder-ogive with an aft cylindrical sting. All wind-tunnel data used for comparison included the cylindrical sting.

The fins consisted of a clipped delta wing with a 45-deg sweep and a constant NACA 0008 airfoil section. Figure 2 presents a side view of the configuration with important dimensions illustrated. The wing chord dimension shown is at the centerline of the store when in the carriage position. To accurately model test conditions, a 0.07-in. gap was left between the pylon and finned body.

The basic generation of the individual grids for the wing-pylon-finned body configuration was done with the Eglin Arbitrary Geometry Implicit Euler (EAGLE)[20,21] grid generation code. This code is a general three-dimensional algebraic/elliptic grid generation system that generates both the boundary surfaces and the interior grid. For overlapping grid applications, the grids for each component are built separately and then overlaid using the domain connectivity algorithm to

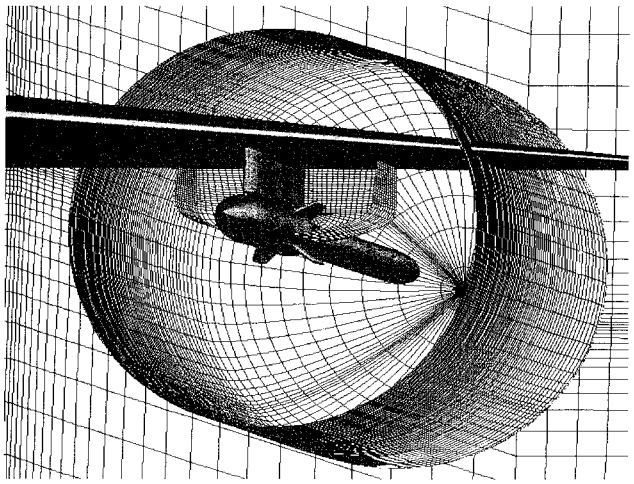

Fig. 3 Wing-pylon-finned store overlap grids.

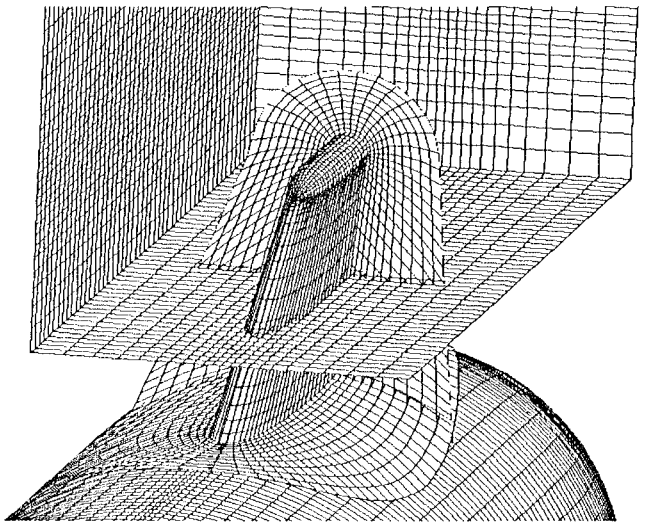

Fig. 4 Overlap grids fin topology.

obtain the final configuration. Sixteen separate grids were built with a total of approximately 1.5 million points. Although this represents an increase in grid points over a comparable block-grid scheme, the grid generation setup time is greatly decreased due to the simplicity of building separate grids. The wing was immersed in a C-O grid with the pylon, fins, and body, each immersed in O-grids. A perspective cutaway view of the overlapped grid system is shown in Fig. 3. Only portions of the wing, pylon, and store grids are presented for clarity.

Grid details of one typical fin are illustrated in Fig. 4. The fin grid is an O-type grid conforming to the store body shape. Although the fin tips are flat in the wind-tunnel model, the gridded fin tips are rounded to permit easier grid generation and to reduce the chances of the flow solution diverging at the fin tips due to flow separation and overexpansion. With this difference in fin tip geometry, to more closely simulate the flat fin tip wind-tunnel model, the grid cells on the rounded portion of the fin tips were not included in the force calculations. To provide additional overlapping cells in the fin tip region, a rectangular box grid was built. Three of the boundaries of this grid are shown here. This grid acted as an interface between the relatively coarse wing and store meshes and the fine fin mesh.[6,15]

Experimental Data

The wind-tunnel data selected as a benchmark to evaluate the accuracy of the computational process was obtained in 1990 at the Arnold Engineering Development Center in Aerodynamic Wind Tunnel(4T). This data set consists of trajectory data, force and moment results, and surface pressure distributions on the model described in the previous section at selected trajectory locations. This article will address all three types of data for the Mach number 0.95 case.

The previous studies[6,7] and the data confirmed that a severe nose-down pitching moment exists on the store while in carriage. First observed during the wind-tunnel test, this moment had to be counteracted to ensure a safe, smooth separation. Consequently, the store was ejected downward from the pylon with a nose upward rotation. This was accomplished by setting the aft ejector force to four times that of the forward ejector. The result was a smooth, controlled store separation event. The ejector force and moments were included in the computational store separation process. The ejector characteristics, as well as the flight conditions and other full-scale store parameters, are listed in Table 1 (the wind-tunnel model was assumed to be 5% scale).

Computational Results

This section examines the results obtained from the computational process and compares them to the wind-tunnel data previously discussed. The axis systems used in this discussion are illustrated in Fig. 5 and correspond to the wind-tunnel experiment. The body axis system has its origin at the store c.g. and is fixed with respect to the store. The positive axis directions are shown as well as the positive force coefficient directions. The flight axis system is fixed with respect to the wing-pylon with the origin located at the store c.g. when in carriage. Location and velocity of the store during the trajectory are relative to this axis system. The Euler angles required as input to the domain decomposition code are found from the orientation of the body axis system relative to the flight axis system.

Table 1 Store and ejector parameters

Weight	2,000 lb
Center of gravity	4.65 ft aft of store nose
Forward ejector location	4.06 ft aft of store nose
Aft ejector location	5.73 ft aft of store nose
Forward ejector force	2,400 lb
Aft ejector force	9,600 lb
Ejector stroke length	0.33 ft
Roll moment of inertia, *Ixx*	20 slug-ft^2
Pitch moment of inertia, *Iyy*	360 slug-ft^2
Yaw moment of inertia, *Izz*	360 slug-ft^2
Freestream Mach number	0.95
Aircraft angle of attack	0.0 deg
Altitude	26,000 ft

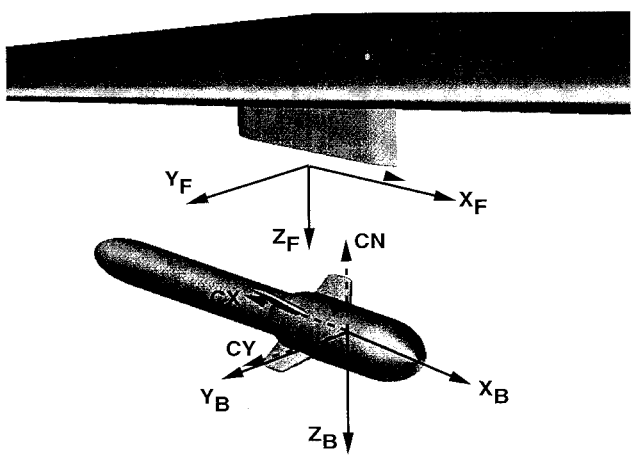

Fig. 5 Axis systems.

Trajectory Characteristics

The initial condition for the trajectory was a fully converged, flow solution obtained by running the flow solver for 3800 iterations in a steady-state mode. The first 2500 iterations were done with local time-stepping, and the final 1300 iterations time-accurately with a nondimensional $DTMIN = 0.06$. This value was used subsequently in the flow solver throughout the trajectory and corresponds to the real time $DT = 0.0001$ used in the trajectory code. The selection of $DTMIN$ was dependent on the smallest grid cell size and the stability of the solution within the pylon-store wake region. The resulting maximum Courant (CFL) number was approximately 2330, located in the wing grid just aft of the wingtip.

The first trajectory parameter examined was the c.g. history (Fig. 6). In all three coordinate directions the computations closely match the wind-tunnel data. The store is moving downward, to the rear and slightly inboard with time. The corresponding angular orientation of the store is shown in Fig. 7. Here, the difference between computations and data is more significant. Major trends are captured, however, and fall within 20% of the wind-tunnel data. Initially, the store is pitched upward and yawed outboard until $t = 0.20$, where the pitch angle begins to decrease. This behavior is expected since the ejectors give the store an initial pitch-up rotation. The store's inherent nose-down pitching moment then takes over and begins to rotate the store downward. This is clearly evident in a plot of the angular rate histories (Fig. 8). The pitch rate q is driven by the ejectors until the end of the stroke, near $t = 0.05$, where the slope of the pitch rate reverses. Trends of all three angular rates are captured well.

Figure 9 compares data to the velocity calculations in the flight axis system. Close agreement is obtained for v and w with the u velocity mismatch partially attributed to the inviscid nature of the flow solver. The initial downward velocity w is dominated by the ejector stroke. Near $t = 0.05$ the store begins the free flight trajectory. The side velocity v is initially inboard, but gradually changes to outboard after the ejector stroke, tending to follow the normal wing outwash.

The force coefficient histories are illustrated in Fig. 10. The axial force is underpredicted due to the lack of viscous forces in the flow solver. Initially, the side and normal forces match the data quite well, but they begin to diverge from the data around $t = 0.15$. One possible explanation is the increasing difference in the angular orientation of the store with time, between the computations and the wind-tunnel data (Fig. 7). This difference in orientation results in the discrepancy between the computational and experimentally measured side and normal forces (Fig. 10). In addition, the store total angle of attack increases during the trajectory up to 18 deg at $t = 0.46$. Typically, the Euler code does not predict forces as well at higher angles of attack.

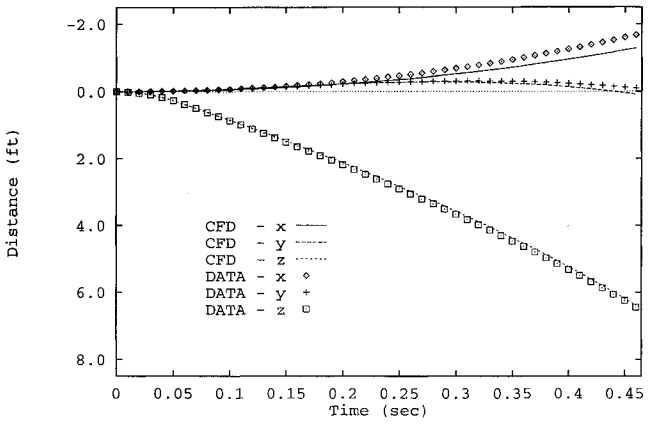

Fig. 6 Trajectory c.g. locations.

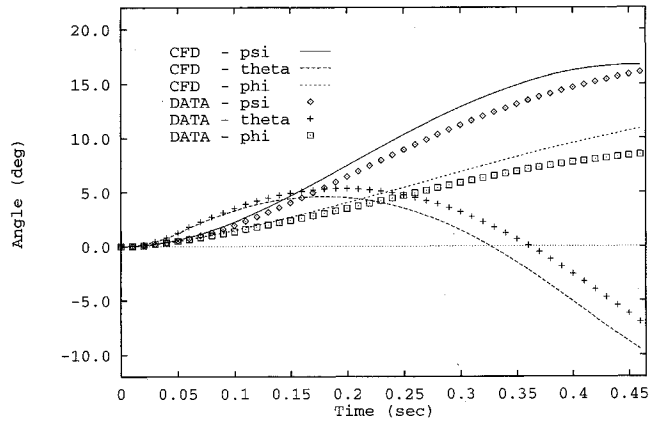

Fig. 7 Trajectory angular orientations.

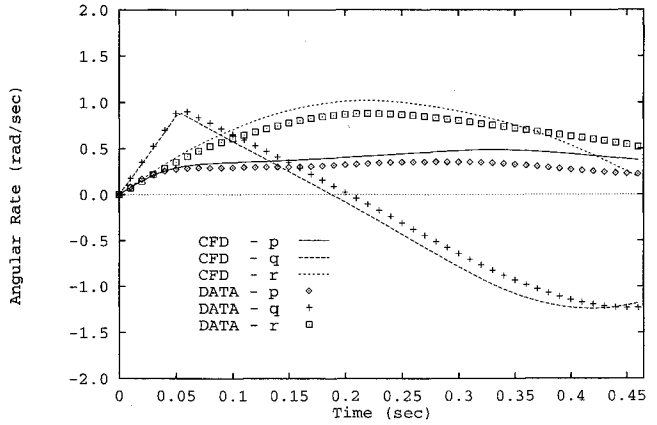

Fig. 8 Trajectory angular rates.

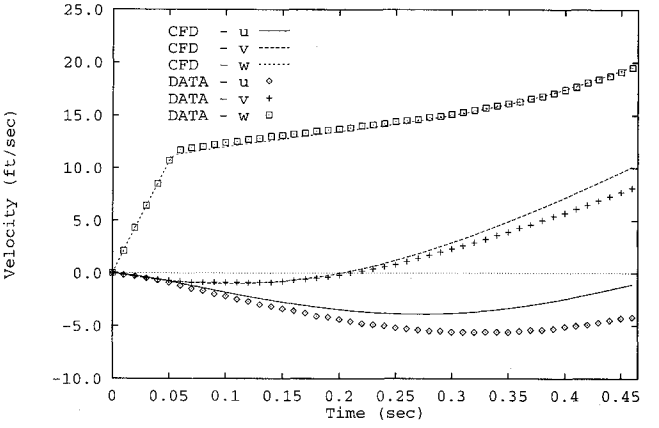

Fig. 9 Trajectory linear velocities.

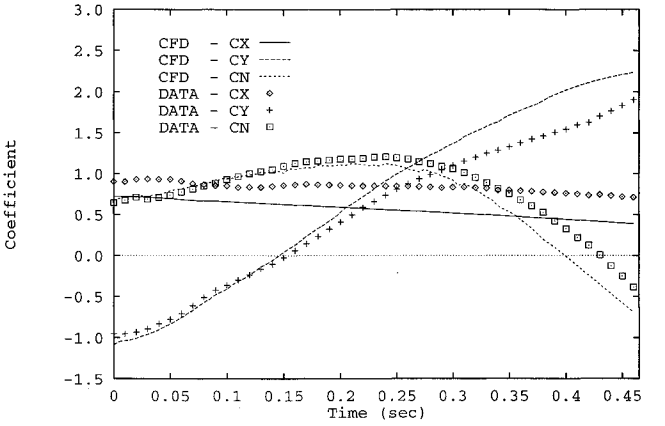

Fig. 10 Force coefficient history.

The moment coefficient history is shown in Fig. 11. Interestingly, the moment coefficients agree better with the data than the forces. Typically, the reverse is true. However, the effect of the over/underpredicted forces causes the moments to deviate from the data near $t = 0.3$. The actual trajectory (Fig. 6), however, illustrates the relative insensitivity of the store to noticeable differences in the force coefficients. This large store is more inertia-driven than aerodynamically responsive. A smaller, low-drag store such as a missile might not be so forgiving. Clearly, the viscous effects are important in the overall trajectory because they affect the total movement of the store.

Besides viscous effects, two other sources of error are evident. One is the modeling of the ejector. The exact time history of the ejector stroke was not recorded, and so the exact time of the end-of-stroke is unknown. This can have a significant effect on the flight conditions the store encounters as it enters free flight. This could partially account for the differences seen in the angular rate comparisons (Fig. 8), orientations (Fig. 7), and the subsequent force coefficients (Fig. 10). The second source of error is the experimental data itself. Although the computational process is time-accurate, the wind-tunnel data is not. The data was taken with the captive trajectory system (CTS) by the pitch-pause method, which is not time-accurate but quasisteady. The motion of the store is accounted for by an induced angle that is added when positioning the store with the CTS. The induced angle represents the instantaneous velocity of the store. This problem was known before the wind-tunnel test was designed, but facilities did not exist to take true unsteady, time-accurate data on a body as well-instrumented as the one tested here.

Surface Pressure Distributions

At five locations in the trajectory, surface pressure data was obtained on the store. This section examines the comparison between the computations at three of the trajectory locations with data and discusses the transient aerodynamic phenomena. The three locations correspond with the carriage position and time steps $t = 0.0$, 0.17, and 0.32. Figures 12–15 are similar in nature in that a pressure coefficient time history is plotted for a fixed location on the store. This fixed location is an axial row of points on the store at a given circumferential angle. The angle is measured with respect to a vertical plane passing through the pylon and store centerlines when the store is in the captive position. This angle is graphically represented by an arrow on the store cross section in the upper left corner of each plot.

Figure 12 plots the pressure coefficient history at a 5-deg circumferential angle. This location is on top of the store, under the pylon while in carriage, but 5 deg outboard. At time zero the pylon has a significant effect on the store pressure distribution. The agreement with data is noteworthy since

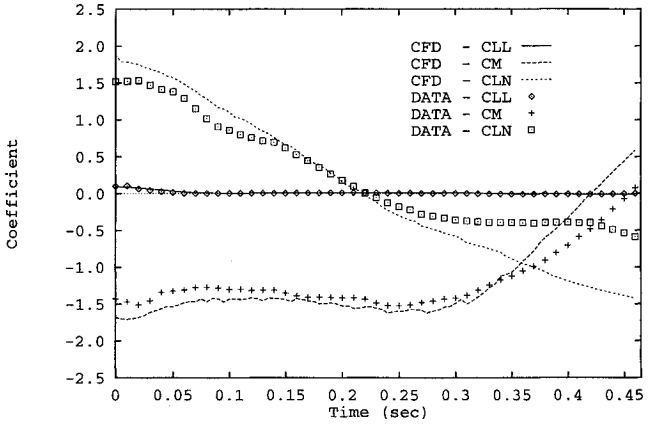

Fig. 11 Moment coefficient history.

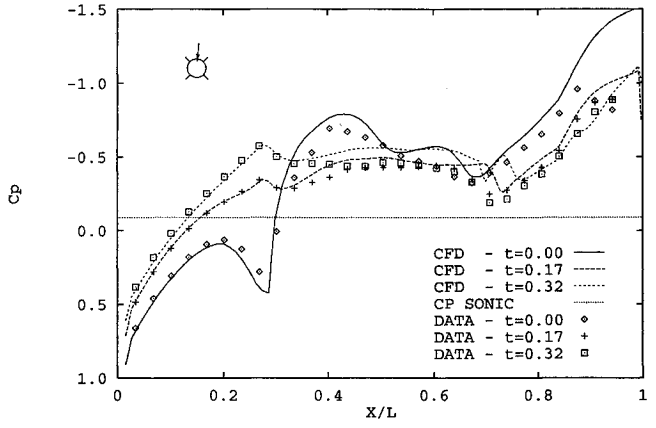

Fig. 12 Pressure coefficient history, PHI = 5 deg.

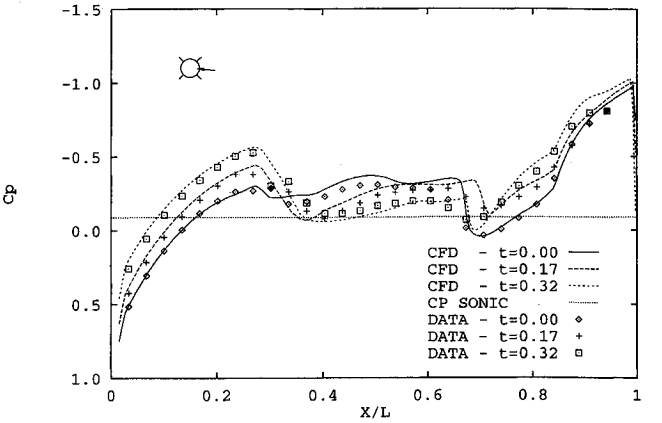

Fig. 13 Pressure coefficient history, PHI = 95 deg.

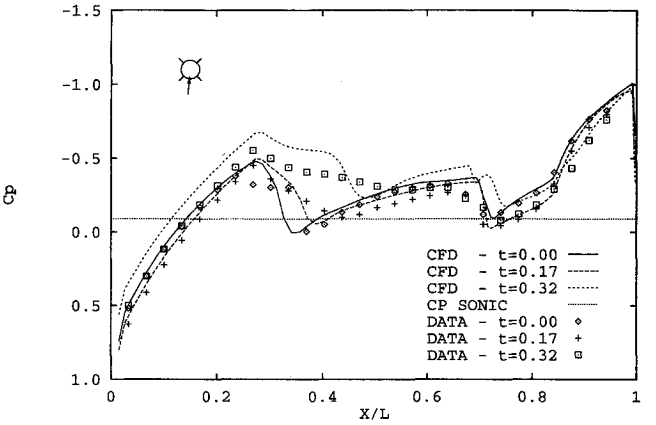

Fig. 14 Pressure coefficient history, PHI = 185 deg.

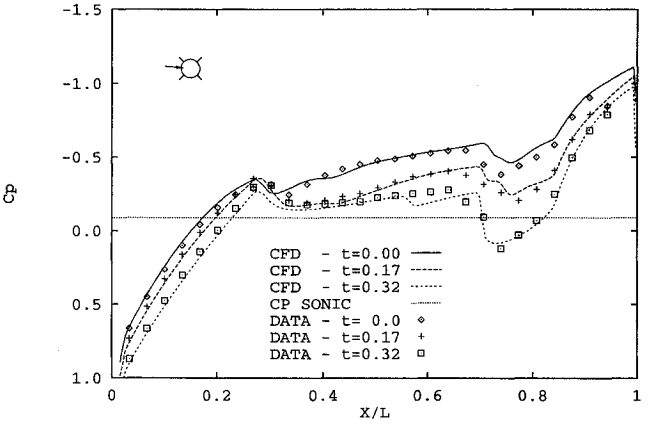

Fig. 15 Pressure coefficient history, PHI = 275 deg.

the data location is in the 0.07-in. gap between the pylon and store. As the store falls away from the pylon, the pronounced compression at approximately $X/L = 0.25$ diminishes with the more normal expansion on the store nose reappearing. Similarly, the large expansion region in the tail area at time zero recedes as the store begins to move out of the influence of the pylon.

At a circumferential angle of 95 deg on the outboard side of the store, the pressure coefficient calculations match the data quite well (Fig. 13). The pylon is not as big an influence here as in the 5-deg position. However, as the store moves down, the flow in the nose region expands and the shock at the leading edge of the fins weakens, near $X/L = 0.7$.

On the bottom of the store at the 185-deg circumferential angle location (Fig. 14), the effects of the interference flowfield are the weakest. Both the store nose and tail section pressure distributions begin to resemble freestream conditions. Only the centerbody expansion region and the weak shocks at $X/L = 0.4$ and 0.7 give evidence of the influence of the wing-pylon. At $t = 0.32$, the pressure prediction is significantly different from the data. With the store yaw angle near 14 deg, the predicted overexpansion of the flow may be indicative of the onset of a local flow separation phenomenon on store. This may be the result of the store bottom experiencing a significant crossflow at 14-deg yaw angle.

At the 275-deg circumferential angle on the inboard side of the store, the agreement with data is extremely good (Fig. 15). Here the effect of the interference flowfield can be seen in the centerbody and tail regions. As the store falls in time, the expansion in these two regions diminishes tending toward the freestream condition.

Conclusions

A CFD approach to store separation has been shown to be very effective in predicting trajectory characteristics at transonic speeds. The accuracy of the store c.g. location and transverse velocities was especially encouraging, although the angular orientations and angular rates were not as precise. However, all major trends in the trajectory were captured. The trajectory of the store was found to be relatively insensitive to significant excursions in normal and axial forces. The inertially driven characteristics of the store were the cause of this lack of response. Surface pressure distributions calculated on the store at three selected locations along the trajectory were found to be in good to excellent agreement with the wind-tunnel data.

Three sources of error were identified in the store separation process. The inviscid nature of the flow solver resulted in errors in the forces, moments, and velocities, especially axial force and axial velocity, which resulted in errors in the x location of the trajectory. Secondly, the approximate modeling of the ejector sequence contributed to the error in the Euler angles and angular rates. Finally, the quasisteady nature of the wind-tunnel data was expressly different than the time-accurate computations. To isolate and define these sources of error should be the subject of future work.

Still, the effectiveness of the three-process technique has been proven. Sound engineering decisions on flight clearances and store certification can be made based on the type and quality of the results shown. The fact that these results were obtained at transonic speeds makes the technique an attractive tool for Air Force fighter store separation.

References

[1]Donegan, T. L., and Fox, J. H., "Analysis of Store Trajectories from Tactical Fighter Aircraft," AIAA Paper 91-0183, Jan. 1991.

[2]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, Jan. 1992.

[3]Arabshahi, A., and Whitfield, D. L., "A Multi-Block Approach to Solving the Three-Dimensional Unsteady Euler Equations About a Wing-Pylon-Store Configuration," AIAA Paper 89-3401, Aug. 1989.

[4]Keen, K. S., "New Approaches to Computational Aircraft/Store Weapons Integration," AIAA Paper 90-0274, Jan. 1990.

[5]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, 1991, pp. 389–396.

[6]Lijewski, L. E., "Transonic Euler Solutions of a Wing-Pylon-Finned Body Configuration Using Blocked and Overlapping Grid Schemes," AIAA Paper 91-2854, Aug. 1991.

[7]Jordan, J. K., "Computational Investigation of Predicted Store Loads in Mutual Interference Flowfields," AIAA Paper 92-4570, Aug. 1992.

[8]Lijewski, L. E., "Transonic Euler Solutions on Mutually Interfering Finned Bodies," *AIAA Journal*, Vol. 28, No. 6, 1990, pp. 982–988.

[9]Steger, J. L., Dougherty, F. C., and Benek, J. A., "A Chimera Grid Scheme," *Advances of Grid Generation*, edited by K. N. Ghia and U. Ghia, American Society of Mechanical Engineers, ASME FED-5, New York, 1983, pp. 59–69.

[10]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, June 1983.

[11]Dougherty, F. C., Benek, J. A., and Steger, J. L., "On Applications of Chimera Grid Scheme to Store Separation," NASA TM-88193, Oct. 1985.

[12]Dougherty, F. C., and Kuan, J., "Transonic Store Separation Using a Three-Dimensional Chimera Grid Scheme," AIAA Paper 89-0637, Jan. 1989.

[13]Meakin, R. L., and Suhs, N. E., "Unsteady Aerodynamics Simulation of Multiple Bodies in Relative Motion," AIAA Paper 89-1966, June 1989.

[14]Carman, J. B., Hill, D. W., and Christopher, J. P., "Store Separation Testing Techniques at the Arnold Engineering Development Center," Arnold Engineering Development Center, AEDC TR-79-1, June 1980.

[15]Suhs, N. E., and Tramel, R. W., "PEGSUS 4.0 User's Manual," Arnold Engineering Development Center, AEDC TR-91-8, Oct. 1991.

[16]Belk, D. M., and Whitfield, D. L., "Three-Dimensional Euler Solutions on Blocked Grids Using an Implicit, Two-Pass Algorithm," AIAA Paper 87-0450, Jan. 1987.

[17]Belk, D. M., and Whitfield, D. L., "Time-Accurate Euler Equations Solutions on Dynamic Blocked Grids," *Proceedings of the 8th Computational Fluid Dynamic Conference* (Honolulu, HI), AIAA, Washington, DC, 1987, pp. 292–302 (AIAA Paper 87-1127).

[18]Whitfield, D. L., and Janus, J. M., "Three-Dimensional Unsteady Euler Equations Solutions Using Flux Vector Splitting," AIAA Paper 84-1552, June 1984.

[19]Belk, D. M., Janus, J. M., and Whitfield, D. L., "Three-Dimensional Unsteady Euler Equations Solutions on Dynamic Grids," *AIAA Journal*, Vol. 25, No. 9, 1987, pp. 1160, 1161.

[20]Thompson, J. F., "A Composite Grid Generation Code for General Three-Dimensional Regions—The EAGLE Code," *AIAA Journal*, Vol. 26, No. 3, 1988, pp. 271, 272.

[21]Thompson, J. F., and Gatlin, B., "Program EAGLE User's Manual, Volumes II and III, Surface and Grid Generation Codes," AFATL TR-88-117, Sept. 1988.