

AIAA 2000-4321  
Steady-State Computation of  
Constant Rotational Rate  
Dynamic Stability Derivatives

Michael A. Park  
George Washington University  
Joint Institute for the Advancement of Flight Sciences (JIAFS)

Lawrence L. Green  
NASA Langley Research Center, Hampton, Virginia

**18<sup>th</sup> Applied Aerodynamics Conference**  
14–17 August 2000  
Denver, Colorado



## STEADY-STATE COMPUTATION OF CONSTANT ROTATIONAL RATE DYNAMIC STABILITY DERIVATIVES

Michael A. Park\*

George Washington University Joint Institute for the Advancement of Flight Sciences (JIAFS)  
mikepark@tabdemo.larc.nasa.gov

Lawrence L. Green†

NASA Langley Research Center, Hampton, Virginia  
l.l.green@larc.nasa.gov

### **Abstract**

Dynamic stability derivatives are essential to predicting the open and closed loop performance, stability, and controllability of aircraft. Computational determination of constant-rate dynamic stability derivatives (derivatives of aircraft forces and moments with respect to constant rotational rates) is currently performed indirectly with finite differencing of multiple time-accurate computational fluid dynamics solutions. Typical time-accurate solutions require excessive amounts of computational time to complete. Formulating Navier-Stokes (N-S) equations in a rotating, noninertial reference frame and applying an automatic differentiation tool to the modified code has the potential for directly computing these derivatives with a single, much faster steady-state calculation. The ability to rapidly determine static and dynamic stability derivatives by computational methods can benefit multidisciplinary design methodologies and reduce dependency on wind tunnel measurements. The CFL3D thin-layer N-S computational fluid dynamics code was modified for this study to allow calculations on complex three-dimensional configurations with constant rotation rate components in all three axes. These CFL3D modifications also have direct application to rotorcraft and turbomachinery analyses. The modified CFL3D steady-state calculation is a new capability that showed excellent agreement with results calculated by a similar formulation. The application of automatic differentiation to CFL3D allows the static stability and body-axis rate derivatives to be calculated quickly and exactly.

\*Graduate Student, NASA Langley Research Center, Multidisciplinary Optimization Branch, MS 159, Hampton, Virginia, Member AIAA

†Research Scientist, Multidisciplinary Optimization Branch, MS 159, Senior Member AIAA

Copyright © 2000 by the American Institute of Aeronautics, Inc. No copyright is asserted in the United States under Title 17, U.S. Code. The Government has royalty-free license to exercise all rights under the copyright claimed herein for government purposes. All other rights reserved by the copyright owner.

### **Introduction**

Dynamic derivatives quantify the aerodynamic damping of aircraft motions and are used to predict the longitudinal short period, lateral pure roll, and lateral Dutch roll behavior of the configuration. Analytical, empirical, and vortex lattice methods of estimating these derivative values are not suited to unconventional configurations or high-speed, compressible flows dominated by viscous effects. Evaluating unconventional configurations is of growing interest due to the design and analysis of next generation attack, transport, and reusable launch vehicles. Examples of these new, unconventional designs are the blended wing body and the X-33 configurations. A methodology of using high fidelity, noninertial Euler and Navier-Stokes (N-S) calculations gives improved capability in predicting these dynamic stability derivative values in compressible flow on conventional or unconventional designs.

Due to cost and time limitations, it is impractical to construct and test numerous wind tunnel models during initial prototyping. Therefore, measurement of the effects of aircraft dynamics on preliminary configuration aerodynamic forces and moments is limited. The application of automatic differentiation to a noninertial reference frame Euler and N-S code has potential for providing designers with insight, gained from higher fidelity codes, into aircraft dynamics at the preliminary design stage. This design stage is when control surface size and preliminary control laws are being evaluated. Computational determination of these derivatives is cheaper and faster than performing wind tunnel measurements and will aid rapid prototyping and multidisciplinary design.

The modification of the CFL3D<sup>1</sup> (Computational Fluids Laboratory Three-Dimensional) computational fluid dynamics (CFD) code to perform calculations in a noninertial, rotating reference frame has the potential to reduce the reliance on forced-motion wind tunnel and free-flight wind tunnel tests. Considerable previous work performed on turbomachinery has demonstrated noninertial, rotating reference frame

fluid mechanics as a means to greatly reduce computational time (for an example see Ref. 2). Kandil and Chuang<sup>3,4</sup> have demonstrated noninertial reference frame calculations for general motions on rolling aircraft stability problems. Limache and Cliff<sup>3</sup> devised an efficient scheme for the special case of steady-rate motion and applied this technique to stability and control work with a two-dimensional (2-D), unstructured grid code and the sensitivity equation method.

The noninertial modifications to CFL3D were initially validated in this study for a 2-D NACA0012 airfoil case with comparisons to previously published results by Limache and Cliff.<sup>5</sup> The modified CFL3D was then applied to the full three-dimensional (3-D) Lockheed Martin Tactical Aircraft Systems—Innovative Control Effectors™ (ICE)<sup>‡</sup> configuration<sup>6</sup> (Fig. 1) with a turbulent Navier-Stokes calculation.

### **Technical Approach**

This study adopted the Limache and Cliff<sup>6</sup> approach. There were two major aspects to this project. The first was modifying CFL3D to perform calculations in a rotating, noninertial reference frame. These CFL3D modifications included adding a source term to the residual calculation and modifying the boundary and initial conditions. The second aspect was the application of ADIFOR<sup>7,8</sup> (Automatic Differentiation in FORTRAN) to the latest parallel version of CFL3D. This code was used to compute derivatives of aircraft forces and moments with respect to the flow angles and constant rotational rates in the roll, pitch, and yaw axes. The application of ADIFOR to the unmodified version of CFL3D has been performed successfully to calculate static stability derivatives<sup>9</sup> (derivatives of aircraft forces and moments with respect to angle of attack and angle of sideslip).

### **CFL3D Introduction**

The CFL3D code is a FORTRAN 77 (F77) Reynolds-averaged thin-layer N-S flow solver for structured-volume grids. CFL3D was written primarily at NASA Langley Research Center and is undergoing continuous development and improvement. The code has the ability to compute inviscid Euler, laminar N-S, and turbulent N-S calculations. The code employs parallelization by decomposing the computational domain into many separate blocks. These individual blocks are analyzed in separate processes that communicate with each other by means of the Message Passing Interface (MPI) standard. Analysis for this

study has been performed in an inviscid, Euler mode and a viscous mode with the N-S equations coupled to the Spalart-Allmaras (S-A) turbulence model.<sup>1</sup>

### **CFL3D Noninertial Reference Frame Modifications**

There are two reference frames depicted in Fig. 2: the inertial reference frame (denoted with upper-case symbols) and the noninertial frame (denoted with lower-case symbols). The CFD grid (depicted as a cube) is embedded in the noninertial reference frame. Positions relative to each of these two reference frames are quantified by three scalar quantities ( $X, Y, Z$  and  $x, y, z$ ) that describe location along three orthonormal unit vectors ( $\mathbf{I}, \mathbf{J}, \mathbf{K}$  and  $\mathbf{i}, \mathbf{j}, \mathbf{k}$ ). Note that bold type face indicates vector symbols. The inertial frame is fixed in space and the noninertial frame can translate and rotate with the rotation described with three scalar components ( $\omega_x, \omega_y, \omega_z$ ) of the rotation vector  $\boldsymbol{\omega}$ . The noninertial frame and CFD grid follow a curved path (denoted as the curved, dashed arrow) as they simultaneously translate and rotate.

Each of these coordinate systems has advantages and disadvantages. An advantage of the inertial frame is that it is not moving (the existing stationary grid N-S equations in CFL3D are only formulated for nonmoving frames). An advantage of the noninertial reference frame is that it moves with the CFD grid; therefore, the stationary grid formulation of the CFL3D N-S equations is already coded in this frame of reference with local (lower-case) variables. A disadvantage of the noninertial reference frame is the current, stationary grid N-S equations are not formulated correctly because the CFD grid and its associated reference frame are rotating (e. g., accelerating) in this study to simulate aircraft constant-rate motion.

In order to correctly modify the stationary grid N-S equations to calculate valid solutions with a translating and rotating CFD grid, the relation between the descriptions of the same point in both reference frames ( $\mathbf{b}$  and  $\mathbf{B}$ ) must be sought. Note that all of these derivations are performed at the instant in time when the unit normal vectors of both systems are parallel, which removes the necessity of a rotational coordinate transform. Also, the noninertial reference frame is translating and rotating at a constant rate; therefore angular acceleration,  $\dot{\boldsymbol{\omega}}$ , is zero. A nonzero value of  $\dot{\boldsymbol{\omega}}$  can be included to model more general motions,<sup>3,4</sup> but such was not the intention of this study.

There are two points in space of interest for this derivation: The position of the noninertial frame origin,  $\mathbf{C}$ , expressed as a function of the inertial coordinates ( $X, Y, Z$ ) and a fluid particle,  $\mathbf{B}$  and  $\mathbf{b}$ , expressed in the inertial and noninertial coordinates ( $X, Y, Z$  and  $x, y, z$ ), respectively. At any instant in time, it is very easy to express the relation of the position of a

<sup>‡</sup> The use of trademarks or names of manufacturers in this report is for accurate reporting and does not constitute an official endorsement, either expressed or implied, of such products or manufacturers by the National Aeronautics and Space Administration.

point in both coordinate systems by addition of vectors.

$$\mathbf{B} = \mathbf{C} + \mathbf{b} \quad (1)$$

The next step is to find the relationship between the instantaneous velocity of a point expressed in both coordinate systems. The velocity is found by differentiating the expression for the vector relation of position, Eq. (1), with respect to time. Note that there will be an added complexity to computing the derivative of any vector quantity expressed in the noninertial frame, e. g.,  $\mathbf{b}$ , because the unit normal vectors ( $\mathbf{i}, \mathbf{j}, \mathbf{k}$ ) are changing as a function of time, due to rotation. To find the derivative of  $\mathbf{b}$ , the product rule is used on the multiplication of the scalar components ( $x, y, z$ ) and unit vectors ( $\mathbf{i}, \mathbf{j}, \mathbf{k}$ ). The relation between the instantaneous derivatives of the unit vectors and  $\boldsymbol{\omega} \times \mathbf{b}$  is found by taking the limit on the derivative as  $dt$  goes to zero.<sup>10</sup>

$$\frac{d\mathbf{B}}{dt} = \frac{d\mathbf{C}}{dt} + \frac{d\mathbf{b}}{dt} \quad (2)$$

$$\dot{\mathbf{B}} = \dot{\mathbf{C}} + \dot{\mathbf{b}} + \boldsymbol{\omega} \times \mathbf{b} \quad (3)$$

Now that the velocity relationship, Eq. (3), has been derived, the far-field boundary conditions of the noninertial CFD problem can be discussed. For free stream boundary conditions, the fluid particles are at rest in the inertial frame; therefore their velocity,  $\dot{\mathbf{B}}|_{\infty}$ , is zero. The expression for the fluid velocity at the CFD grid free stream boundary conditions is  $\dot{\mathbf{b}}|_{\infty}$ . The aircraft, the noninertial reference frame, and the CFD grid are all translating together at a velocity,  $\dot{\mathbf{C}}$ , which is negative the free stream velocity,  $\mathbf{u}_{\infty}$ . The boundary values for  $\dot{\mathbf{B}}|_{\infty}$  and  $\dot{\mathbf{C}}$  are substituted into Eq. (3), which is rearranged to form Eq. (4).

$$\dot{\mathbf{b}}|_{\infty} = \mathbf{u}_{\infty} - \boldsymbol{\omega} \times \mathbf{b}|_{\infty} \quad (4)$$

Therefore, the free stream boundary conditions can be described as the combination of a uniform flow component,  $\mathbf{u}_{\infty}$ , and a rigid body rotation component,  $\boldsymbol{\omega} \times \mathbf{b}|_{\infty}$ . The CFL3D boundary conditions at the near-field or solid surface boundary conditions are not affected in this noninertial formulation.

The expression for acceleration is computed by differentiating the velocity relation, Eq. (3).<sup>10</sup> The time derivatives of the  $\mathbf{b}$  and  $\dot{\mathbf{b}}$  terms are determined in the same fashion as the derivative of the  $\mathbf{b}$  term was derived in Eq. (3). This relation is valid for any vector quantity expressed in a noninertial frame. Note that

the  $\dot{\boldsymbol{\omega}}$  term is zero because the rotation rate is assumed to be constant in this steady-state formulation.

$$\frac{d\dot{\mathbf{B}}}{dt} = \frac{d\dot{\mathbf{C}}}{dt} + \frac{d\dot{\mathbf{b}}}{dt} + \frac{d(\boldsymbol{\omega} \times \mathbf{b})}{dt} \quad (5)$$

$$\ddot{\mathbf{B}} = \ddot{\mathbf{C}} + \ddot{\mathbf{b}} + \boldsymbol{\omega} \times \dot{\mathbf{b}} + \dot{\boldsymbol{\omega}} \times \mathbf{b} + \boldsymbol{\omega} \times \dot{\mathbf{b}} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{b}) \quad (6)$$

$$\ddot{\mathbf{B}} = \ddot{\mathbf{C}} + \ddot{\mathbf{b}} + 2\boldsymbol{\omega} \times \dot{\mathbf{b}} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{b}) \quad (7)$$

Now the acceleration of the origin of the noninertial, grid-fixed reference frame,  $\ddot{\mathbf{C}}$ , must be sought. In this formulation, the noninertial reference frame (Fig. 2) is following a curved path (the dashed arrow) through inertial space as it simultaneously translates ( $\dot{\mathbf{C}}$ ) and rotates ( $\boldsymbol{\omega}$ ). The origin of the noninertial reference frame must accelerate to follow this curved path. The expression for the acceleration of a grid that is moving in a curved path with constant rotation rate is

$$\ddot{\mathbf{C}} = \boldsymbol{\omega} \times \dot{\mathbf{C}}, \quad \ddot{\mathbf{C}} = \boldsymbol{\omega} \times -\mathbf{u}_{\infty} \quad (8)$$

Note that this reference frame origin acceleration is zero when  $\dot{\mathbf{C}}$  is parallel to  $\boldsymbol{\omega}$ .

Now that the  $\ddot{\mathbf{C}}$  term is known, the expressions for the difference between the accelerations computed in the inertial frame and the noninertial frame (CFD grid) can be completed. This difference in acceleration is computed by subtracting the acceleration of a fluid particle in the noninertial frame,  $\ddot{\mathbf{b}}$ , from the acceleration of the same particle expressed in the inertial frame,  $\ddot{\mathbf{B}}$ . By accounting for this difference in acceleration (pseudo-acceleration,  $\ddot{\mathbf{B}} - \ddot{\mathbf{b}}$ ), equations describing the motion of particles measured in a noninertial frame can correctly mimic the total acceleration of these fluid particles in the inertial frame.

$$\ddot{\mathbf{B}} - \ddot{\mathbf{b}} = \boldsymbol{\omega} \times -\mathbf{u}_{\infty} + 2\boldsymbol{\omega} \times \dot{\mathbf{b}} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{b}) \quad (9)$$

The goal is to model constant-rate CFD grid rotation and translation with a steady-state CFD calculation. The difference in acceleration between an inertial and a noninertial frame of reference is employed to form a source term correction to the N-S equations in CFL3D. To illustrate the formulation of the source term, the existing implementation of the stationary grid N-S equations in CFL3D must be examined. In the CFL3D code, the N-S equations are expressed in a regular-spaced, Cartesian grid coordinate system. The generalized grid coordinate system that defines the problem ( $x, y, z$ ) is internally mapped by CFL3D to this regular-spaced Cartesian

grid ( $\xi, \eta, \zeta$ ) with a coordinate transform. The N-S equations are written in the regular-spaced, Cartesian grid coordinate system as<sup>1</sup>

$$\frac{\partial \hat{Q}}{\partial t} + \frac{(\hat{F} - \hat{F}_v)}{\partial \xi} + \frac{(\hat{G} - \hat{G}_v)}{\partial \eta} + \frac{(\hat{H} - \hat{H}_v)}{\partial \zeta} = 0 \quad (10)$$

where  $\hat{Q}$  is a vector of the conserved variables. The conserved variables are a combination of density,  $\rho$ , velocity components ( $u, v, w$ ) and total energy per unit volume,  $e$ . The vector  $\hat{Q}$  is the conserved variables divided by  $J$ .

$$\hat{Q} = \frac{\mathbf{Q}}{J} = \frac{1}{J} [\rho \ \rho u \ \rho v \ \rho w \ e]^T \quad (11)$$

The Jacobean ( $J$ ) of the coordinate transformation from the Cartesian to the generalized coordinate system is

$$J = \frac{\partial(\xi, \eta, \zeta)}{\partial(x, y, z)} \quad (12)$$

The inviscid flux terms are  $\hat{F}$ ,  $\hat{G}$ , and  $\hat{H}$  and the viscous flux terms are  $\hat{F}_v$ ,  $\hat{G}_v$ , and  $\hat{H}_v$ . The  $\hat{F}$ ,  $\hat{G}$ ,  $\hat{H}$ ,  $\hat{F}_v$ ,  $\hat{G}_v$ , and  $\hat{H}_v$  flux terms are created by dividing by  $J$  in the same manner as  $\hat{Q}$  was. For a nondeforming mesh ( $J$  is constant with respect to time), the solution is advanced in time with the residual,  $R$ .

$$\frac{1}{J} \frac{\partial \hat{Q}}{\partial t} = R(\mathbf{Q}) \quad (13)$$

The residual is computed as

$$R(\mathbf{Q}) = - \left[ \frac{\partial(\hat{F} - \hat{F}_v)}{\partial \xi} + \frac{\partial(\hat{G} - \hat{G}_v)}{\partial \eta} + \frac{\partial(\hat{H} - \hat{H}_v)}{\partial \zeta} \right] \quad (14)$$

To permit noninertial calculations, a source term ( $S$ ) is added to the standard CFL3D residual calculation.

$$\frac{1}{J} \frac{\partial \hat{Q}}{\partial t} = R(\mathbf{Q}) + S \quad (15)$$

The source term is a vector with 4 nonzero components (the continuity equation is not affected).

$$S = \frac{\rho}{J} [0 \ S_x \ S_y \ S_z]^T \quad (16)$$

The momentum equation source terms ( $S_x$ ,  $S_y$ , and  $S_z$ ) are the three components of the pseudo-acceleration ( $\ddot{\mathbf{B}} - \dot{\mathbf{b}}$ ). The "work" done by these momentum source terms must be included in the energy equation. The energy equation source term ( $S_e$ ) is the dot product of the local velocity with the pseudo-acceleration.

The N-S equations in CFL3D are written in conservative form, so the vector source term for the four momentum and energy equations must include the volume of each computational cell ( $J^{-1}$ ) and the density,  $\rho$ , of the fluid. The energy equation source term is the dot product of the local flow velocity,  $\dot{\mathbf{b}}$ , with pseudo-acceleration,

$$S = \frac{\rho}{J} \begin{bmatrix} 0 \\ (\boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{b}) + 2\boldsymbol{\omega} \times \dot{\mathbf{b}} - \boldsymbol{\omega} \times \mathbf{u}_\infty)_x \\ (\boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{b}) + 2\boldsymbol{\omega} \times \dot{\mathbf{b}} - \boldsymbol{\omega} \times \mathbf{u}_\infty)_y \\ (\boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{b}) + 2\boldsymbol{\omega} \times \dot{\mathbf{b}} - \boldsymbol{\omega} \times \mathbf{u}_\infty)_z \\ \dot{\mathbf{b}} \bullet (\boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{b}) + 2\boldsymbol{\omega} \times \dot{\mathbf{b}} - \boldsymbol{\omega} \times \mathbf{u}_\infty) \end{bmatrix} \quad (17)$$

where  $\mathbf{b} = [x \ y \ z]$  and  $\dot{\mathbf{b}} = [u \ v \ w]$ . For additional papers on the source term equations and associated physics of similar applications of this theory, see Refs. 2-5. Refs. 4 and 5 also include reference frame angular and translation acceleration terms.

#### ADIFOR Automatic Differentiation

Automatic differentiation is a technique for augmenting computer programs with statements for the computation of derivatives. This technique relies on the fact that every function, no matter how complicated, is executed on a computer as a (potentially very long) sequence of elementary operations such as additions, multiplications, and elementary functions (e. g., sine and cosine). By repeatedly applying the chain rule of differential calculus to the composition of those elementary operations, derivative information can be computed exactly and in a completely automated fashion.

The ADIFOR process is a technique that applies the chain rule of differentiation to propagate, equation by equation, derivatives of intermediate variables with respect to the input variables. The ADIFOR tool has been developed jointly by the Center for Research on Parallel Computation at Rice University and the Mathematics and Computer Sciences Division at Argonne National Laboratory. In general, to apply ADIFOR to a given F77 code, the user is only required to specify those program variable names that correspond to the independent and dependent variables of the target differentiation. The ADIFOR tool then determines the variables that require associated derivative computations, formulates the appropriate derivative expressions, and generates new F77 code for the computation of both the original simulation and the specified derivatives.

The modified version of CFL3D is able to compute the aircraft forces and moments as a function of the three body-axis orthogonal components ( $p$ ,  $q$ , and  $r$ ) of the rotation vector. The application of ADIFOR to the modified version of CFL3D produced a code that computed the forward mode derivatives of the aircraft body-axis force and moment coefficients ( $C_N$ ,  $C_A$ ,  $C_S$ ,  $C_l$ ,  $C_m$ , and  $C_n$ ) with respect to the three body-axis rotation rates ( $p$ ,  $q$ , and  $r$ ) and the flow angles of attack ( $\alpha$ ) and sideslip ( $\beta$ ).

The latest beta version of CFL3D employs dynamic memory allocation and MPI libraries for ease of use and efficient, scalable parallelization. These implementations are not standard F77 features, and therefore previous releases of ADIFOR cannot handle the code without manual preprocessing and postprocessing. The latest release of ADIFOR 3.0<sup>11</sup> has reduced or eliminated much of the manual processing associated with the MPI libraries; techniques for handling the dynamic memory allocation libraries are being developed.

### **Examples and Results**

The two examples in this study were a 2-D Euler study of NACA0012 airfoil and a 3-D turbulent N-S calculation on the ICE configuration<sup>6</sup> (Fig. 1). The NACA0012 study will be detailed first, because it was used for initial validation by comparisons to existing methods.

#### **NACA0012 Dynamic Derivatives**

The NACA0012 study focused on the effect of pitch rate on the coefficients of normal force (Fig. 3) and pitching moment (Fig. 4) at zero deg angle of attack and zero pitch rate. The derivatives ( $C_{N_q}$ , Fig. 3, and  $C_{m_q}$ , Fig. 4) of these force and moment coefficients with respect to nondimensional pitch rate were computed by the ADIFOR-generated, noninertial CFL3D code (CFL3D.NI.AD). The pitch rate derivatives are nondimensionalized by dividing by the airfoil chord and multiplying by two times the free stream velocity. The NACA0012 pitching moment center is located at the leading edge of the airfoil. The convergence history of the derivative values is shown in Figs. 3 and 4. The discontinuities in the derivative convergence history are due to mesh sequencing from a coarser to a finer mesh every 500 iterations. A maximum of three levels of multigrid was employed on the finer meshes. The 2-D grid dimensions ( $49 \times 13$ ,  $97 \times 25$ ,  $193 \times 49$ , and  $385 \times 97$ ) are denoted for each mesh sequencing level. The derivative values are compared to results computed by a similar method published by Limache and Cliff (SFLOW),<sup>5</sup> a panel method (QUADPAN),<sup>12</sup> and a vortex lattice method (VORLAX).<sup>13</sup>

These 2-D NACA0012 cases shown in Figs. 3 and 4 were chosen for initial validation of CFL3D.NI.AD. To improve convergence, a blend of half standard CFL3D and half CFL3D low-Mach-number preconditioning was applied for the 0.1 Mach (Figs. 3a and 4a) case. This preconditioning option was not applied to the 0.5 Mach (Figs. 3b and 4b) or 0.8 Mach (Figs. 3c and 4c) cases. Note that the CFL3D.NI.AD derivative values are in excellent agreement with the SFLOW values. For the 0.1 and 0.5 Mach cases, the differences between CFL3D.NI.AD and SFLOW, although small, are most likely due to the formulation differences between the flow solvers in CFL3D.NI.AD and SFLOW. The SFLOW code employs the hand-coded sensitivity equation technique and an unstructured grid discretization, whereas CFL3D.NI.AD is an automatically differentiated structured grid formulation.

The 0.8 Mach case (Figs. 3c and 4c) shows poor convergence properties. The poor convergence of CFL3D.NI.AD at 0.8 Mach may be due to the interaction of a shock, the flux limiter implemented in CFL3D, and the automatic differentiation technique. The CFL3D smooth flux limiter<sup>1</sup> tuned to  $\kappa = 1/3$  was employed for the NACA0012 study. This poor convergence may be due to the automatic differentiation technique attempting to formulate the continuous derivative of a shock and flux limiter, which does not have a continuous derivative. The 0.8 Mach case is also the worst comparison to SFLOW. Only the final value for SFLOW is quoted in Ref. 5; therefore the SFLOW 0.8 Mach case may or may not be fully converged. The convergence was not improved by disabling multigrid calculations or performing additional iteration cycles. At 0.8 Mach, the final value of CFL3D.NI.AD and SFLOW differ in normal force pitch rate derivative by 4.4% (Fig. 3c) and in pitching moment pitch rate derivative by 8.9% (Fig. 4c).

#### **ICE Pathlines at Zero Rotational Rate**

After CFL3D.NI.AD was validated by comparison with SFLOW, the ICE configuration (Fig. 1) flow structure was examined with pathlines. These pathlines are shown to illustrate the changes in airflow structure with increasing angles of attack. Pathlines for the starboard half-span of the ICE configuration are shown in Fig. 5. The pathlines were seeded slightly ahead of the sharp leading edge (just outside the boundary layer). The pathlines were computed from a full-span N-S CFL3D solution on a grid with approximately 3 million cells. These symmetric solutions in Fig. 5 were calculated at 0.6 Mach, zero deg angle of sideslip, zero rotational rate, and various angles of attack. All ICE CFL3D solutions in this study were computed with the S-A turbulence model at a Reynolds number of

2,490,000 per foot or 71,760,000 per mean aerodynamic chord. Three levels of multigrid were used for the fine grid solution of the 0–15 deg angle of attack cases and multigrid was disabled for the fine grid solution of the 20–30 deg angle of attack cases.

The structure of the symmetric flow depicted in Fig. 5 aids interpretation of the subsequent figures, which depict force, moment, and stability derivative information at these flight conditions. Note the attached flow at 5 deg angle of attack (Fig. 5a). Weak leading edge vortical flow was present at 10 deg angle of attack (Fig. 5b). This initial leading edge vortex structure gained strength at 15 deg angle of attack (Fig. 5c). A vortex burst developed near the trailing edge at 20 deg angle of attack (Fig. 5d). This vortex burst structure is identified by an abrupt streamwise increase in vortex diameter. The initial vortex burst structure intensified and moved forward at 25 and 30 deg angles of attack (Fig. 5e and 5f).

#### ICE Forces, Moments, and Lateral Derivatives

A summary of the figures depicting ICE forces, moments, and stability derivatives in this study is given in Table 1. Note that the elements assumed zero would become significant at a nonzero angle of sideslip, roll rate, or yaw rate. The ICE angle-of-attack derivatives are not presented in this study, but are presented in Ref. 6 for 0–10 deg angles of attack. The orientation of the six forces and moments, two flow angles, and three body-axis rotational rates is shown in Fig. 1. Figure 6 shows a comparison of longitudinal forces and moment at 0.6 Mach, zero deg angle of sideslip, and zero rotational rate. The moment reference center of the ICE is located longitudinally at 39% of the mean aerodynamic chord (MAC) and a distance of 16% of the MAC below the body. The comparison is the wind tunnel data (solid line, WT) present in the ICE simulator database<sup>6</sup> and CFL3D (dashed line). Coefficients of normal force, axial force, and pitching moment are shown in Fig. 6a, 6b, and 6c, respectively.

Figure 6 shows good agreement between WT and CFL3D. The WT data has more detail because it was measured at approximately 1 deg increments, which were smaller than the 5 deg increments of the CFL3D calculations. There is no flow visualization information available for the WT data, but the CFL3D pathlines will be used to infer the effects of flow structure on the CFL3D calculations, which may also indicate the flow structure effects on WT measurements. Note that the initiation and strengthening of vortical flow between 5 and 15 deg angles of attack (Fig. 5a–5c) increased the normal force (Fig. 6a). The increasing strength of the vortex flow and the forward movement of the burst location over the wing between 20 and 30 deg angles of attack (Fig. 5d–5f), increased the pitching moment

(Fig. 6c) which resulted in static longitudinal instability above 15 deg angle of attack. Note that CFL3D captures the radical change in  $C_m$  measured by WT (Fig. 6c).

Figure 7 shows the comparison among the lateral angle of sideslip derivatives for three central-finite-difference estimates from wind tunnel data<sup>6</sup> (CD-WT) and an ADIFOR-generated CFL3D solution (CFL3D.AD). The CFL3D.AD derivatives are dashed lines and the CD-WT derivatives are the symbols with a central-finite-difference step of  $\pm 2$ ,  $\pm 4$ , and  $\pm 6$  deg angle of sideslip for the circle, square, and diamond, respectively. The  $\pm 2$  deg CD-WT data is connected with solid lines because the smallest central-finite-difference step ( $\pm 2$ ) is presumed to be the most accurate of the three finite difference step sizes for small sideslip disturbances. All three finite difference step sizes are shown to give an indication of the nonlinearities or measurement noise in the wind tunnel data. The derivatives in Fig. 7 are presented in the units of  $\text{deg}^{-1}$ .

The effects of the vortical flow structure (Fig. 5) can be seen clearly in the lateral force and moments angle-of-sideslip derivatives (Fig. 7). The initiation and strengthening of vortical flow between 5 and 10 deg angles of attack (Fig. 5a and 5b) can be interpreted to have sharply influenced the angle-of-attack trends of  $CS_\beta$  and  $Cn_\beta$  (Fig. 7a and 7c) computed by CD-WT and CFL3D.AD. Then, the derivatives  $CS_\beta$  and  $Cn_\beta$  (Fig. 7a and 7c) dramatically reversed angle-of-attack trends above 10 deg angle of attack. The CFL3D.AD  $Cl_\beta$  (Fig. 7b) derivative showed excellent agreement with CD-WT for 0 to 15 deg angles of attack. The  $Cl_\beta$  comparison deteriorated at higher (20–30 deg) angles of attack.

As angle of sideslip varies, each wing experiences different effective leading-edge sweep angles. Due to the highly swept (65 deg) leading edge of the ICE configuration the vortical flow field over the wing may be sensitive to changes in effective leading-edge sweep angle. Therefore, the calculation of a vortex burst structure that formed symmetrically at 20 deg angle of attack (Fig. 5d) may be produced asymmetrically at lower (10–15 deg) angles of attack. An asymmetric, bursting vortex structure may have been responsible for the dramatically reversed angle-of-attack trends in the lateral derivatives (Fig. 7). The ICE configuration does not have any vertical surfaces, so the magnitude of  $CS_\beta$  and  $Cn_\beta$  (Fig. 7a and 7c) was reduced as compared to a configuration with vertical surfaces. The small magnitude of  $CS_\beta$  and  $Cn_\beta$  may have hindered measurement accuracy and exacerbated comparison of CD-WT with CFL3D.AD.



### ICE Dynamic Derivatives

Figures 8 and 9 show the dynamic derivatives computed by the DYNAMIC<sup>14</sup> code and CFL3D.NI.AD. The DYNAMIC code utilized strip theory and the results of the high-angle-of-attack stability and control prediction code HASC<sup>15,16</sup> to calculate the dynamic derivatives. The HASC code employs VORLAX<sup>13</sup> and empirical corrections to predict configuration forces and moments at various flow angles and rotational rates. The derivatives were computed at zero rotational rate, zero angle of sideslip, and various angles of attack. The CFL3D.NI.AD dynamic derivatives were computed assuming rotations about the moment center of the configuration, which is located slightly below the body. The longitudinal pitch rate derivatives  $CN_q$  and  $Cm_q$  are shown in Fig. 8a and 8d, respectively. The longitudinal dynamic derivatives were nondimensionalized by dividing by the mean aerodynamic chord (345 in.) and multiplying by two times the free stream velocity.

The rolling moment dynamic derivatives  $Cl_p$  and  $Cl_r$  are shown in Fig. 8b and 8e, respectively. The yawing moment dynamic derivatives  $Cn_p$  and  $Cn_r$  are shown in Fig. 8c and 8f, respectively. The side force dynamic derivatives  $CS_p$  and  $CS_r$  are shown in Fig. 9a and 9b, respectively. The lateral dynamic derivatives were nondimensionalized by dividing by the wingspan  $b$  (450 in.) and multiplying by two times the free stream velocity. There was no forced, oscillatory motion wind tunnel data for comparison.

Both codes, CFL3D.NI.AD and DYNAMIC, showed fairly good comparison. Both of the CFL3D.NI.AD pitch rate ( $q$ ) derivatives (Fig. 8a and 8d) show a local maximum or a minimum near 5 deg angle of attack. Note that the CFL3D.NI.AD calculation of  $Cm_q$  (Fig. 8d) was consistently more negative than the combined analytical and vortex lattice method of DYNAMIC and VORLAX. This trend agrees with those of both SFLOW and CFL3D.NI.AD when compared to VORLAX for the 2-D NACA0012 case (Fig. 4).

The roll rate ( $p$ ) derivatives (Figs. 8b, 8c, and 9a) also showed a reversal of angle of attack trends at 5 deg angle of attack. The reversals of the  $q$  and  $p$  derivative trends at 5 deg angle of attack corresponded to the indication of vortical flow at 10 deg angle of attack in Fig. 5b. These  $p$  derivatives also showed another local extreme at 15–20 deg angle of attack, which was slightly below the indication of vortex bursting in the static pathlines (Fig. 5d). A roll rate creates differential angles of attack on each wing, which may induce asymmetric vortical burst structures at lower angles of attack than a zero-roll-rate symmetric case.

The yaw rate ( $r$ ) derivatives (Figs. 8e, 8f, and 9b) had consistent trends in angle of attack at 15 deg angle

of attack and lower. These trends became less consistent at 20, 25, and 30 deg angles of attack, which corresponded with the initial indication of a symmetric vortex burst structure in Fig. 5d.

The CFL3D.NI.AD differentiated flow solver had convergence difficulties at 20, 25, and 30 deg angles of attack. The 30 deg angle of attack case never reached a steady-state value, so an average of the last 2 thousand iterations is presented. These convergence difficulties may have been due to the presence of bursting vortex structures, with their inherent unsteadiness and increased sensitivity to disturbances. These high angle-of-attack conditions may be more suitable to a time-accurate solution, but in the interest of minimizing computational resource requirements, that approach was not attempted in this study.

### ICE Pathlines at Nonzero Rotational Rates

Figure 10 shows the ICE configuration at 0.3 Mach, 15 deg angle of attack and zero deg angle of sideslip, performing velocity vector rolls at various rotational rates. In these velocity vector rolls, the rotation vector was parallel to the free stream velocity vector; this condition simulated a wind tunnel rotary-balance test. These solutions were computed by the noninertial, modified CFL3D (CFL3D.NI) code. The rotational rate ( $\Omega$ ) was nondimensionalized by multiplying by the wingspan  $b$  (450 in.) and dividing by two times the free stream velocity ( $u_\infty$ ), with a positive rotational rate indicating the starboard wing was descending. The 0.2 and 0.4 rotational rate cases (Fig. 10b and 10c) showed a much tighter vortex core on the ascending, port wing than the descending, starboard wing. The 0.4 rotational rate case (Fig. 10c) depicted a vortex burst on the descending, starboard wing. From this point of view, the vortex wakes in Fig. 10b and 10c appear to be converging, but actually were spiraling around the rotation vector.

### ICE Rotary-Balance and Noninertial CFL3D Comparison

Figure 11 shows a comparison of wind tunnel rotary-balance data<sup>6</sup> (ROT-BAL, solid line) and CFL3D.NI (dashed line) at 0.3 Mach, 15 deg angle of attack, and zero deg angle of sideslip. Mach 0.3 was chosen to simulate the incompressible conditions of the low-speed, rotary-balance tests. The ICE configuration was rotated about the moment center of the configuration, which is located slightly below the body. The figure shows the change ( $\Delta$ ) in force or moment coefficient between cases nonrotating and rotating about the velocity vector. The rotation rate ( $\Omega$ ) about the velocity vector was nondimensionalized by multiplying by the wingspan  $b$  (450 in.) and dividing by two times the free stream velocity ( $u_\infty$ ), with a positive rotational rate indicating the starboard wing

was descending. Note that nonlinear effects with rotational rate were modeled in CFL3D.NI.

The increase in normal force ( $\Delta C_N$ , Fig. 11a) due to rotation was very similar between ROT-BAL and CFL3D.NI. The change in axial force ( $\Delta C_A$ , Fig. 11b) due to rotation was very similar in magnitude between ROT-BAL and CFL3D.NI, but opposite in sign. The change in pitching moment ( $\Delta C_m$ , Fig. 11c) due to rotation was assumed to be an even function as calculated by CFL3D.NI, but ROT-BAL showed inconclusive trends. The change in side force ( $\Delta C_S$ , Fig. 11d) due to rotation was assumed to be an odd function as calculated by CFL3D.NI, but ROT-BAL showed inconclusive trends. The change in rolling moment ( $\Delta C_l$ , Fig. 11e) due to rotation was the best lateral comparison of CFL3D.NI with ROT-BAL. The change in yawing moment ( $\Delta C_n$ , Fig. 11f) due to rotation calculated by CFL3D.NI was much greater in magnitude and opposite in sign of the ROT-BAL trend.

The nonlinear effects with rotational rate as calculated by CFL3D.NI in Fig. 11 can be correlated to the calculation of a vortex burst structure over the descending wing as illustrated in Fig. 10b and 10c. The difference between ROT-BAL and CFL3D.NI (highlighted in Figs. 11b, 11c, and 11f) is currently under investigation. A possible explanation of ROT-BAL asymmetries may be model asymmetries or installation misalignments. The poor comparisons of ROT-BAL and CFL3D.NI may be due to rotation about different locations for the experimental and computational cases. The CFL3D.NI code simulated rotation about the reported moment center of the configuration, which is outside the model. The ROT-BAL tests may or may not have rotated the model about that moment center location.

### **Timing**

Table 2 describes the processors, wall time, and RAM required by the original CFL3D, CFL3D.NI, and CFL3D.NI.AD. The column labeled "Independents" indicates whether function only (zero independents) or function plus derivatives with respect to angle of attack, angle of sideslip, roll rate, pitch rate, and yaw rate (five independents) were calculated. The column labeled "Processors" indicates the number of SGI Origin 2000™ (O2K) processors employed for the calculations. All three parallel versions of the CFL3D code employed in this study use one of the processors for administrative tasks, so the number of actual computing processors is one less than the number quoted in the "Processors" column. The four-processor runs were performed on a NASA Langley Multidisciplinary Optimization Branch four-processor O2K with 4 Gb RAM. The 14-processor runs were performed on a HPCCP 16-processor O2K with 12 Gb

RAM. By means of a batch queuing system, the 16-processor O2K total wall time was achieved through multiple 45 min runs. The 16 processor O2K had significant shutdown and restart overhead (approximately 10%), which adversely affects total wall time for the CFL3D.NI.AD examples.

Note that CFL3D.NI required 0.5 hour (3.8%) more execution time than the original CFL3D steady-state execution wall time for the ICE configuration with the S-A turbulence model. The corresponding wall time increase for 2-D and 3-D Euler calculations due to noninertial modifications was approximately 15%. The noninertial modifications had a larger penalty for Euler than turbulent N-S solutions because N-S and S-A solutions required more calculations per iteration than Euler solutions. The increased calculations per iteration of the turbulent N-S solution masked the same number of noninertial modification calculations per iteration of the turbulent N-S and Euler solutions.

A time-accurate CFL3D solution that would emulate a CFL3D.NI solution was estimated to require approximately 175 hours, or more than an order of magnitude increase in wall time over a CFL3D.NI calculation. The central-finite-difference estimate wall time was calculated by multiplying the CFL3D.NI time by 11 (one function plus ten perturbed solutions) to yield 148.5 hours, which was scaled between the two O2K computers assuming perfect, linear speedup with a ratio of 3 worker processors to 13 worker processors. In other words,  $13.5 \times 11 = 148.5$  and  $148.5 \times 3 / 13 \approx 34$ . The central-difference estimate required 9.7% more wall time than CFL3D.NI.AD between 0 and 15 deg angles of attack. Compared to the 0–15 deg angle of attack solutions, CFL3D.NI.AD required three to four times the wall time at 20, 25, and 30 deg angle of attack, due to differentiated flow solver convergence difficulties. The vortex burst structures at the higher (20–30 deg) angles of attack (Fig. 5d–5f) may have been responsible for the convergence difficulties.

### **Conclusions**

An initial application of ADIFOR to CFL3D with constant-rate noninertial modifications to compute constant-rate rotary stability derivatives was completed. This application was validated for a 2-D NACA0012 Euler case by comparison to the SFLOW code, a similar formulation. ADIFOR-generated noninertial CFL3D derivatives of a 2-D NACA0012 airfoil showed good comparison with existing methods at 0.1 and 0.5 Mach. Symmetric vortical flow structures for the ICE configuration were identified by means of computational flow visualizations of turbulent N-S calculations at 5–30 deg angles of attack. The nature of these vortical flow structures was correlated to the behavior of forces, moments,

angle-of-sideslip derivatives, and rotational rate derivatives at 0–30 deg angles of attack. Flow visualization techniques were also applied to computational solutions for velocity vector rolls at 15 deg angle of attack; these visualizations depicted asymmetric vortex burst structures at a nondimensional roll rate of 0.4. The effect of these asymmetric vortical flow structures was observed in the nonlinear effects of rotation rate on forces and moments.

The application of noninertial, constant-rate calculations was demonstrated for compressible and viscous flows on an unconventional configuration. This new CFL3D capability proved to be an accurate method to complement or reduce dependency on forced-motion rotary or oscillatory wind tunnel measurements. This noninertial reference frame modification to CFL3D also has direct application to turbomachinery studies. The noninertial reference frame theory utilized to formulate the source terms in CFL3D.NI can easily be extended to include angular or translational acceleration terms to model more generalized aircraft or grid motions. The application of ADIFOR to the modified version of CFL3D has great promise as a dynamic, constant-rate rotary derivative prediction tool for stability and control work in design studies and multidisciplinary design frameworks.

### **Acknowledgements**

The authors would like to thank Thomas Zang for his support and diligent review; HPCCP and the Multidisciplinary Optimization Branch for the use of their O2K computers; Bob Biedron for help with CFL3D source code; Jim Thomas and Gautam Shah for advice and review; Alan Carle and Mike Fagan, developers of ADIFOR, for their assistance.

Michael Park is supported by a NASA grant to George Washington University and would like to thank his advisor, Professor Robert Sandusky, for his guidance.

### **References**

- <sup>1</sup>Krist, E., Biedron, R., and Rumsey, C., "CFL3D User's Manual (Version 5.0)" NASA TM-1998-20844, June 1998.
- <sup>2</sup>Chen, J. P., Ghosh, A. R., Sreenivas, K., and Whitfield, D. L., "Comparison of Computations Using Navier-Stokes Equations in Rotating and Fixed Coordinates for Flow Through Turbomachinery," AIAA Paper 97-0878, Jan. 1997.
- <sup>3</sup>Kandil, O. A., and Chuang, H. A., "Unsteady Vortex-Dominated Flows around Maneuvering Wings over a wide Range of Mach Numbers," AIAA Paper 88-0317, Jan. 1988.
- <sup>4</sup>Kandil, O. A., and Chuang, H. A., "Unsteady Navier-Stokes Computations Past Oscillating Delta Wing at High Incidence," *AIAA Journal*, Vol. 28, No. 9, Sept. 1990, pp. 1565–1572.
- <sup>5</sup>Limache, A. C., and Cliff, E. M., "Aerodynamic Sensitivity Theory for Rotary Stability Derivatives," AIAA Paper 98-4313, Aug. 1998.
- <sup>6</sup>Dorsett, K. M., and Mehl, D. R., "Innovative Control Effectors (ICE)," Wright Laboratory Report, WL-TR-96-3043, Jan. 1996.
- <sup>7</sup>Bischof, C., Carle, A., Corliss, G., Greewank, A., and Hovland, P., "ADIFOR—Generating Derivative Codes from Fortran Programs," *Scientific Programming*, No. 1, 1992, pp. 1–29.
- <sup>8</sup>Bischof, C., Carle, A., Khademi, P., and Mauer, A., "Automatic Differentiation of FORTRAN," *IEEE Computational Science & Engineering*, Fall 1996.
- <sup>9</sup>Park, M., Green, L., Montgomery, R., and Raney, D., "Determination of Stability and Control Derivatives Using Computational Fluid Dynamics and Automatic Differentiation," AIAA paper 99-3136, June 1999.
- <sup>10</sup>Meirovitch, L., "Methods of Analytical Dynamics," McGraw-Hill, New York, NY, 1970.
- <sup>11</sup>Carle, A. and Fagan, M., "Overview of Adifor 3.0," Department of Computational and Applied Mathematics, Rice University, CAAM-TR 00-02, Jan. 2000.
- <sup>12</sup>Youngren, H. H., Bouchard, E. E., Coopersmith, R. M., and Miranda, L. R., "Comparison of Panel Method Formulations and its Influence on the Development of QUADPAN, an Advanced Low Order Method," AIAA Paper 83-1827, July 1983.
- <sup>13</sup>Miranda, L. R., Elliott, R. D., and Baker, W. M., "A Generalized Vortex Lattice Method of Subsonic and Supersonic Flow Applications," NASA CR-2865, Dec. 1977.
- <sup>14</sup>Simon, J. M., "Dynamic Derivative Data for High-Angle-of-Attack Simulation," AIAA paper 92-4355, Aug. 1992.
- <sup>15</sup>Blake, W. B., Dixon, C. J., and Adler, C. O., "Development of a High-Angle-of-Attack Stability and Control Prediction Code," AIAA Paper 92-4354, Aug. 1992.
- <sup>16</sup>Albright, A. E., Dixon, C. J., Hegedus, M. C., "Modification and Validation of Conceptual Design Aerodynamic Prediction Method HASC95 With VTXCHN," NASA CR-4712, Mar. 1996.

**Table 1 Summary of Figures Presented for the ICE Configuration at 0.6 Mach,  $\beta = 0, p = 0, q = 0, r = 0$ .**

	$CN^s$	$CA^s$	$Cm^s$	$CS^s$	$Cl^s$	$Cn^s$
Function	6a	6b	6c	0	0	0
Angle of attack derivative	x, Ref. 6	x, Ref. 6	x, Ref. 6	0	0	0
Angle of sideslip derivative	0	0	0	7a, Ref. 6	7b, Ref. 6	7c, Ref. 6
Pitch rate derivative	8a	x	8d	0	0	0
Roll rate derivative	0	0	0	9a	8b	8c
Yaw rate derivative	0	0	0	9b	8e	8f

<sup>s</sup>0 – Assumed zero for a laterally symmetric configuration; x – not shown, but assumed nonzero.

**Table 2 Execution Time and RAM for CFL3D, CFL3D.NI, and CFL3D.NIAD of ICE N-S and S-A.**

Description	Independents	Processors	Wall Time	Total RAM
CFL3D	0	4	13 hours	1468 Mb
CFL3D.NI	0	4	13.5 hours	1468 Mb
Original time-accurate CFL3D estimate	0	4	175 <sup>†</sup> hours	1468 <sup>†</sup> Mb
Center-finite-difference CFL3D estimate	5	14	34 <sup>†</sup> hours	1468 <sup>†</sup> Mb
CFL3D.NIAD, 0–15 $\alpha$	5	14	31 hours	9828 Mb
CFL3D.NIAD, 20–30 $\alpha$	5	14	90–120 hours	9828 Mb

<sup>†</sup>Estimates.

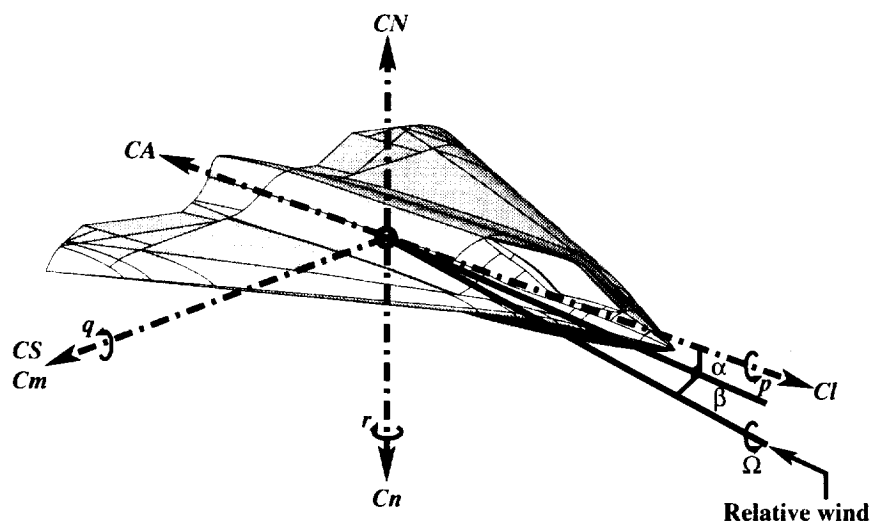


Fig. 1 Lockheed Martin Tactical Aircraft Systems ICE configuration.

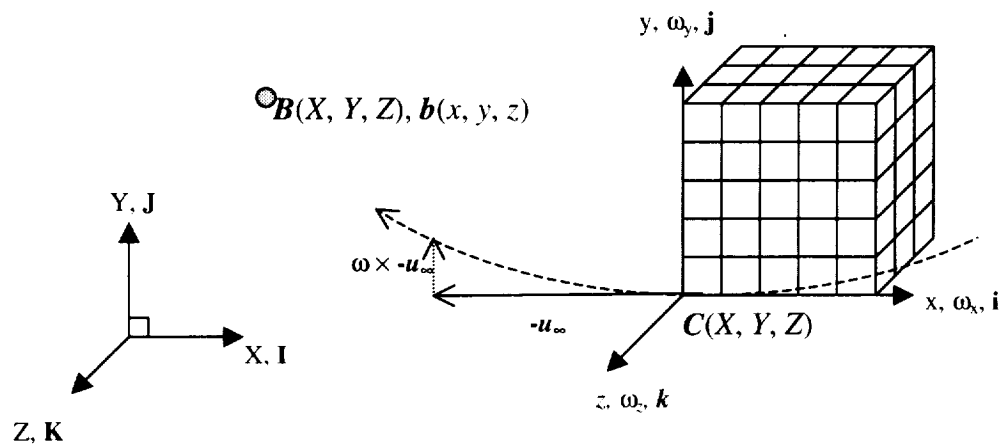
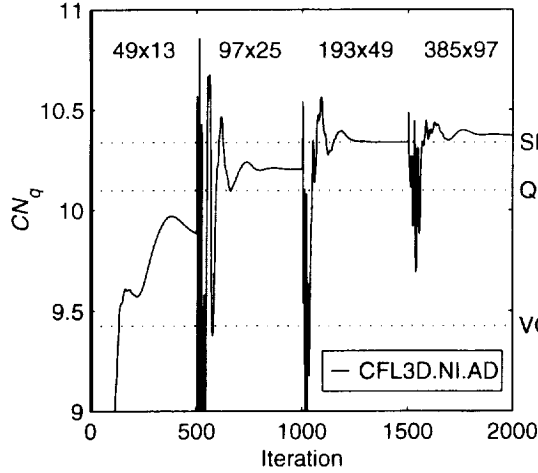
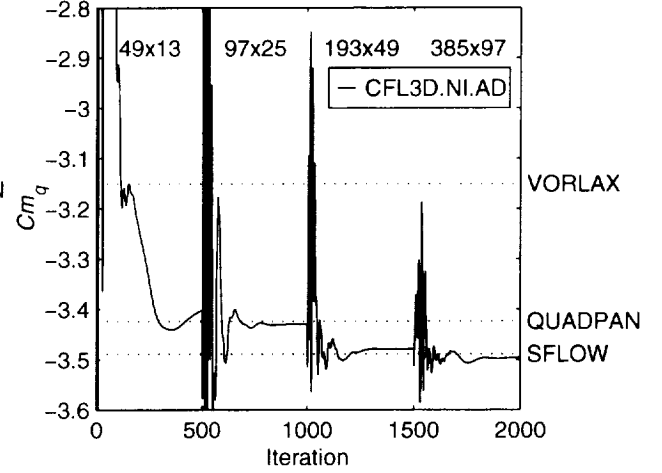


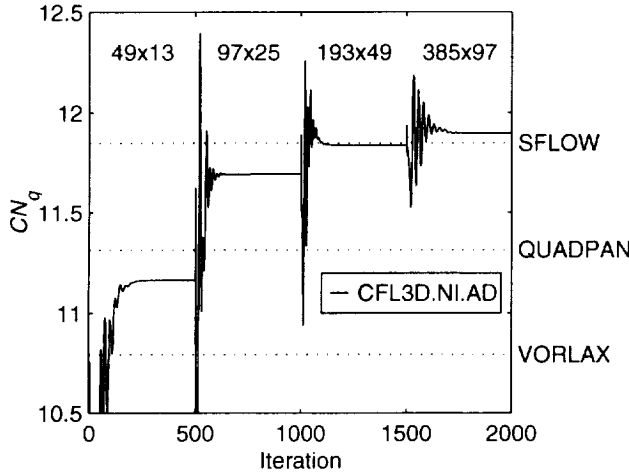
Fig. 2 Inertial and noninertial reference frames.



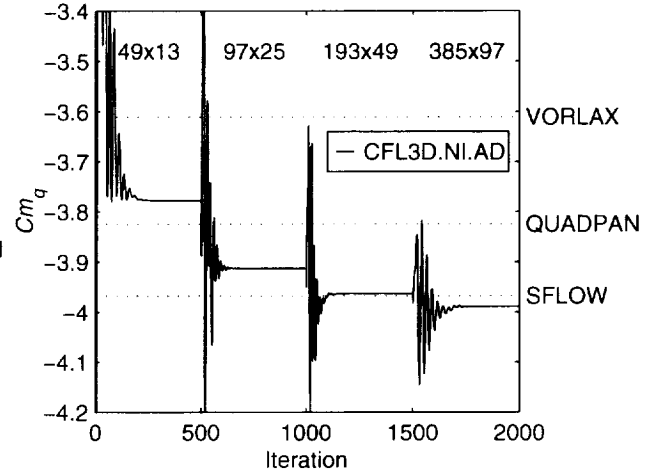
a) Normal force pitch rate derivative, 0.1 Mach



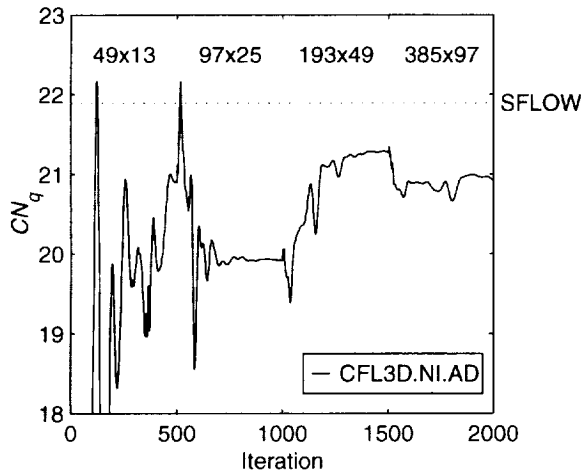
a) Pitching moment pitch rate derivative, 0.1 Mach



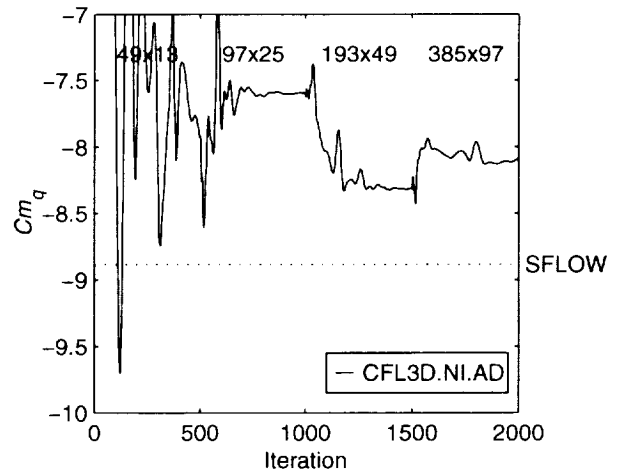
b) Normal force pitch rate derivative, 0.5 Mach



b) Pitching moment pitch rate derivative, 0.5 Mach



c) Normal force pitch rate derivative, 0.8 Mach



c) Pitching moment pitch rate derivative, 0.8 Mach

Fig. 3 Convergence history of 2-D Euler NACA0012 airfoil normal force pitch rate derivatives;  $\alpha = 0$ ,  $q = 0$ .

Fig. 4 Convergence history of 2-D Euler NACA0012 airfoil pitching moment pitch rate derivatives;  $\alpha = 0$ ,  $q = 0$ .

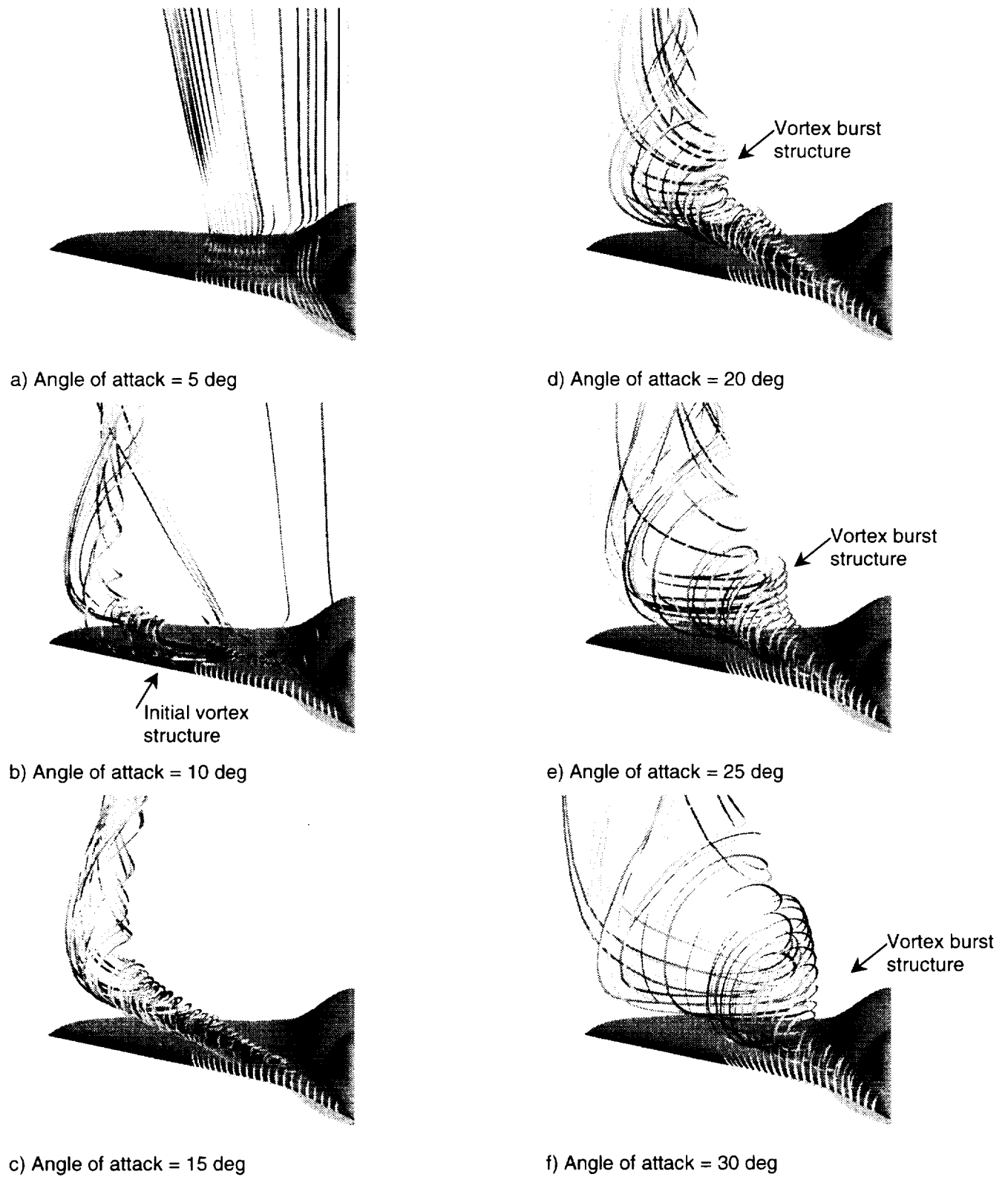


Fig. 5 Forward looking aft at the upper surface of the starboard half-span of ICE configuration, depicting pathlines; 0.6 Mach,  $\beta = 0$ ,  $p = 0$ ,  $q = 0$ ,  $r = 0$ .

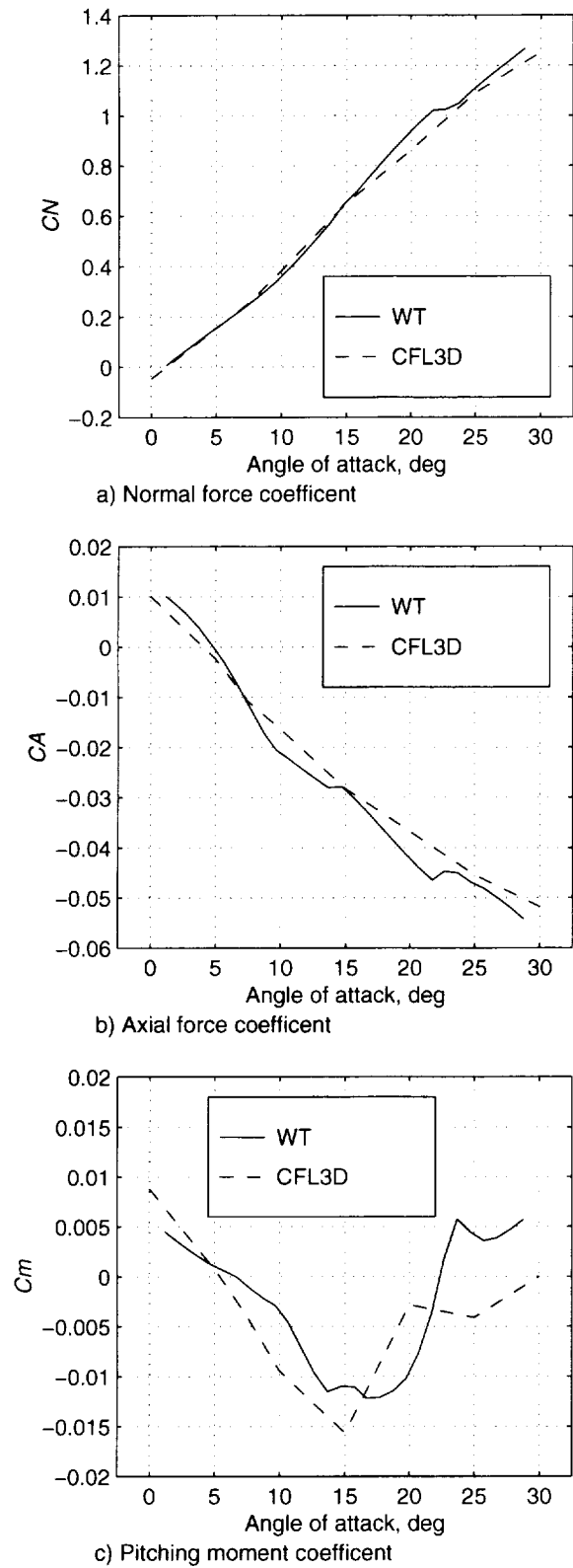


Fig. 6 ICE static longitudinal forces and moment coefficients; 0.6 Mach,  $\beta = 0$ ,  $p = 0$ ,  $q = 0$ ,  $r = 0$ .

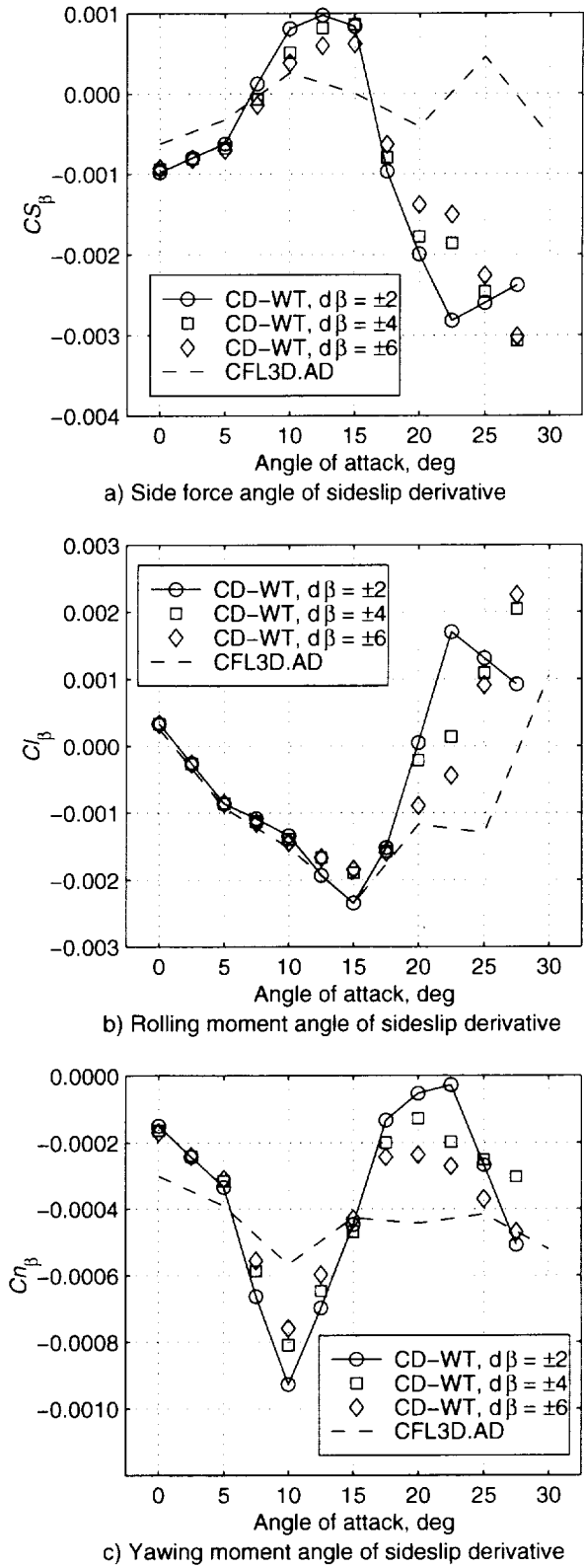


Fig. 7 ICE static lateral force and moments angle of sideslip derivatives,  $\text{deg}^{-1}$ ; 0.6 Mach,  $\beta = 0$ ,  $p = 0$ ,  $q = 0$ ,  $r = 0$ .

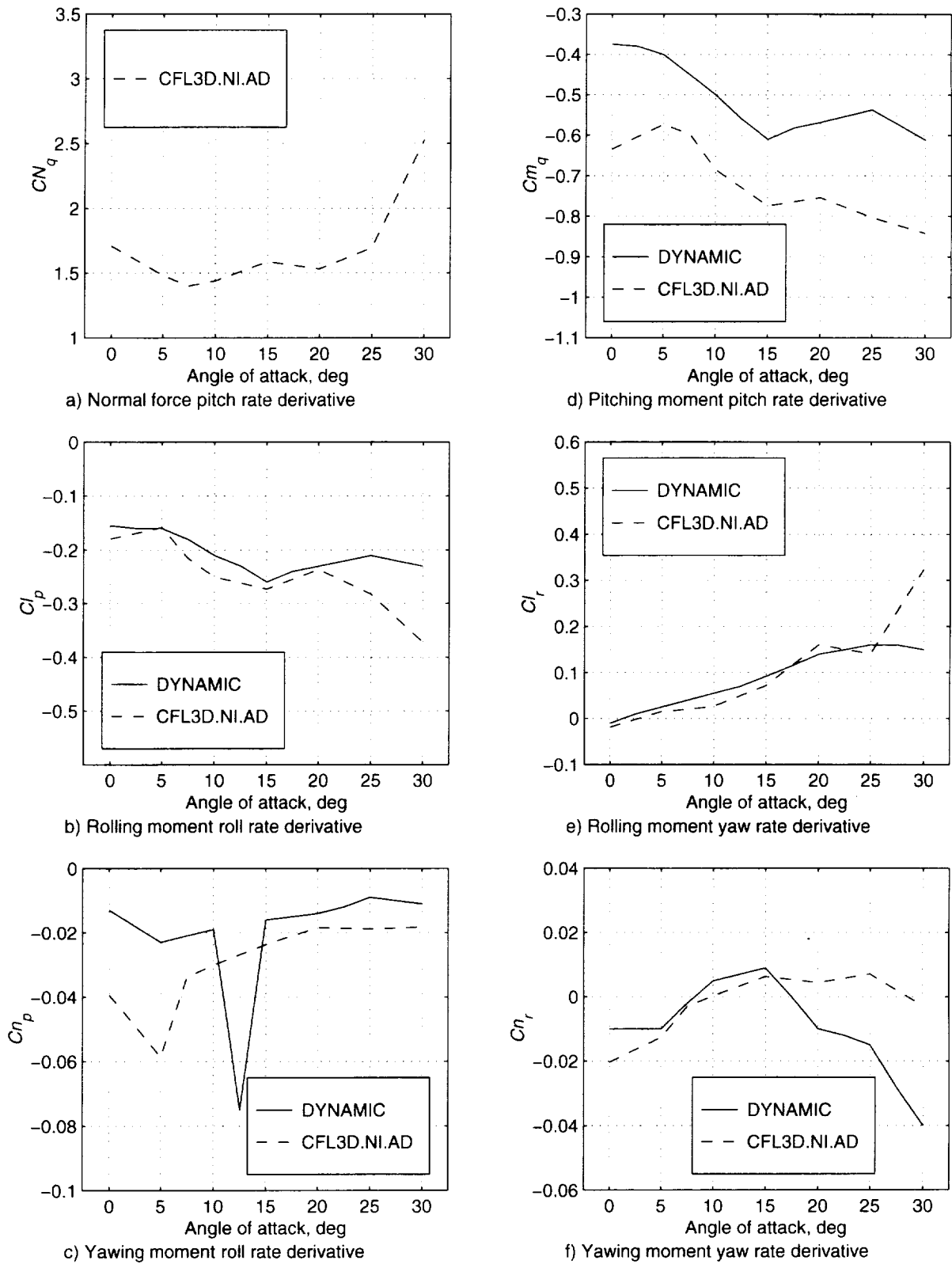
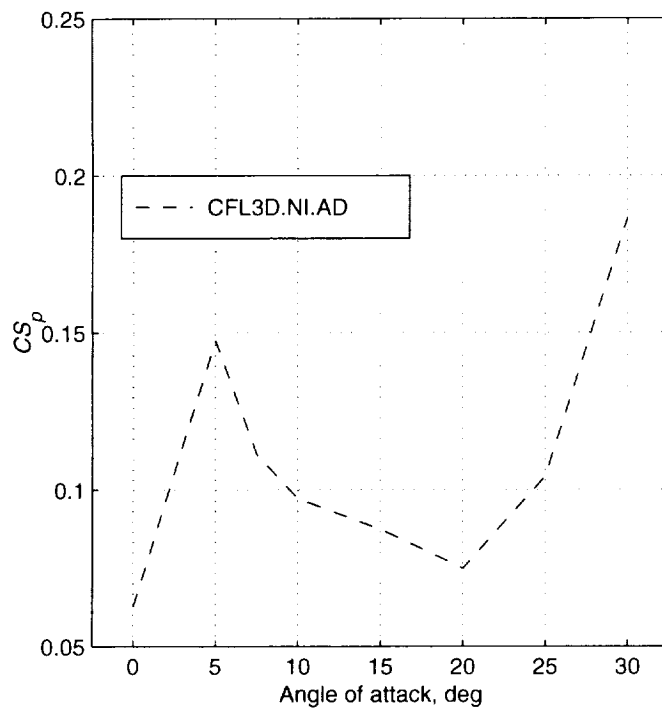
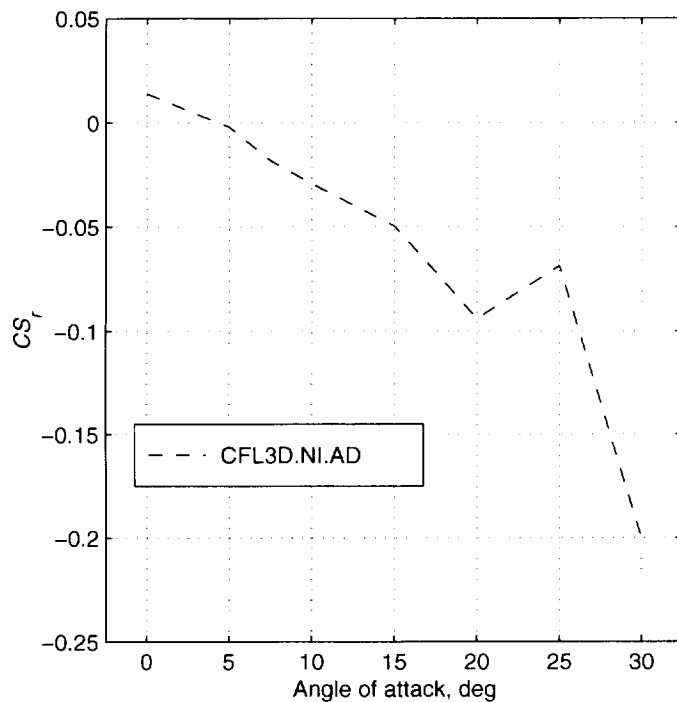


Fig. 8 ICE configuration rate derivatives;  
0.6 Mach,  $\beta = 0$ ,  $p = 0$ ,  $q = 0$ ,  $r = 0$ .



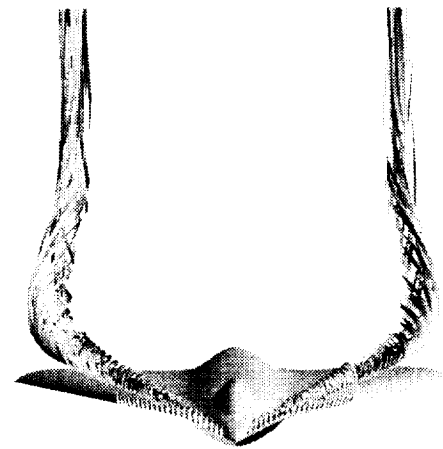


a) Side force roll rate derivative

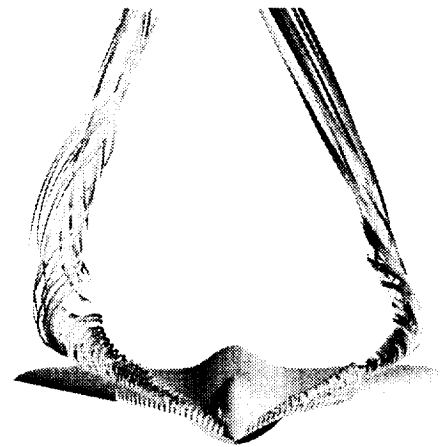


b) Side force yaw rate derivative

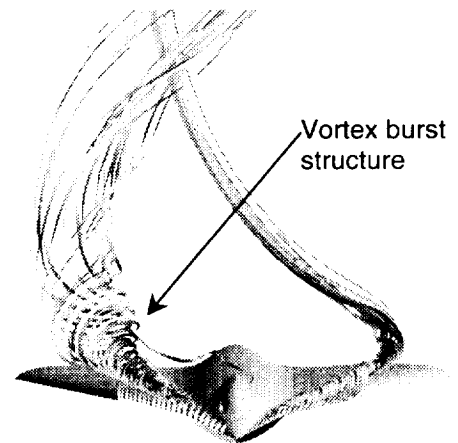
Fig. 9 ICE configuration side force rate derivatives; 0.6 Mach,  $\beta = 0$ ,  $p = 0$ ,  $q = 0$ ,  $r = 0$ .



a) Rotating at  $(\Omega b) / (2 u_\infty) = 0$



b) Rotating at  $(\Omega b) / (2 u_\infty) = 0.2$



c) Rotating at  $(\Omega b) / (2 u_\infty) = 0.4$

Fig. 10 ICE configuration velocity vector rolls; 0.3 Mach,  $\alpha = 15$ ,  $\beta = 0$ .

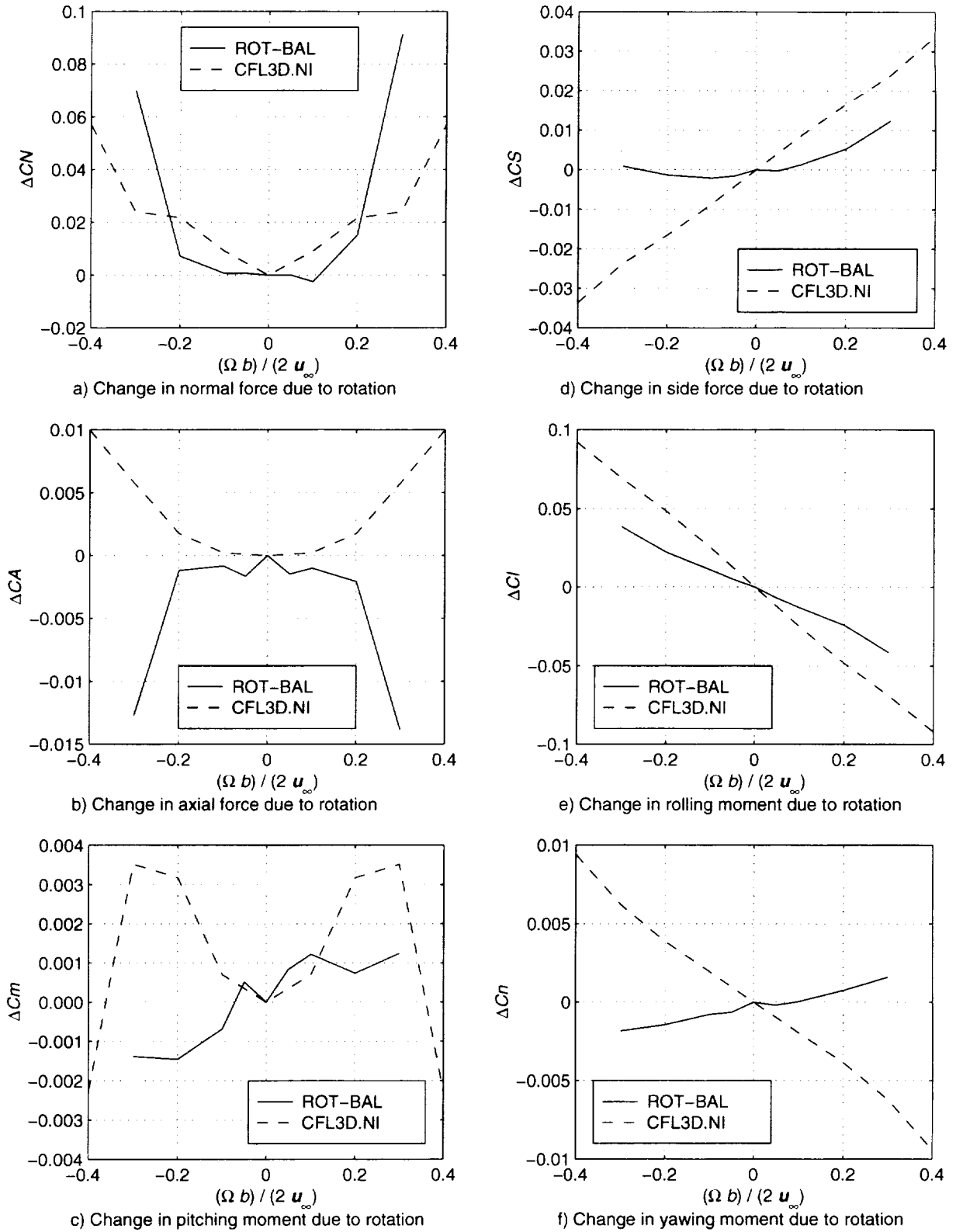


Fig. 11 Comparison of ICE configuration rotary balance wind tunnel data to noninertial CFL3D; 0.3 Mach,  $\alpha = 15^\circ$ ,  $\beta = 0^\circ$ .



