NASA Technical Memorandum 100015

Some Experiences with the Viscous-Inviscid Interaction Approach

W.R. Van Dalsem, J.L. Steger, and K.V. Rao

```
(NASA-TM-100015)  SOME EXPERIENCES WITH THE          N88-12466
VISCOUS-INVISCID INTERACTICN APPROACH
(NASA)  48 p                          CSCL 01A
                                                      Unclas
                                              G3/02  0111140
```

November 1987

NASA

National Aeronautics and
Space Administration

TABLE OF CONTENTS

NASA Technical Memorandum 100015

Some Experiences with the Viscous-Inviscid Interaction Approach

W. R. Van Dalsem
J. L. Steger, Ames Research Center, Moffett Field, California

K. V. Rao, Iowa State University, Ames, Iowa

November 1987

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

SOME EXPERIENCES WITH THE VISCOUS-INVISCID INTERACTION APPROACH

W. R. Van Dalsem, J. L. Steger
NASA Ames Research Center, Moffett Field, California

and

K. V. Rao
Iowa State University, Ames, Iowa

Methods for simulating compressible viscous flow using the viscous-inviscid interaction approach are described. The formulations presented range from the more familiar full-potential/boundary-layer interaction schemes to a method for coupling Euler/Navier-Stokes and boundary-layer algorithms. An effort is made to describe the advantages and disadvantages of each formulation. Sample results are presented which illustrate the applicability of the methods.

NOMENCLATURE

$\mathbf{B}$	vector potential
C_f	skin friction coefficient
C_l	lift coefficient
C_p	pressure coefficient
c	chord
c_p, c_v	specific heats at constant pressure and volume, respectively
E	shift operator (e.g., $E_\xi^{\pm 1} u_j = u_{j\pm 1}$)
H	total enthalpy or shape factor
h	enthalpy or time step
M	Mach number
Pr	Prandtl number
p	pressure
R	perfect gas constant
Re	Reynolds number
s	entropy
T	temperature
t	time
U, V, W	contravariant velocity components
u, v, w	velocity components
x, y, z	coordinate axes
β	$(p/\rho U_\infty^2)$
γ	ratio of specific heats
Δ	forward difference operator (e.g., $\Delta_\xi = (E_\xi^{+1} - 1)/(\Delta\xi)$)
∇	gradient or backward difference (e.g., $\nabla_\xi = (1 - E_\xi^{-1})/(\Delta\xi)$) operator
δ	central difference operator (e.g., $\delta_\xi = (E_\xi^{+1} - E_\xi^{-1})/(2\Delta\xi)$)

$\hat{\delta}$	upwind difference operator, see §B.2 of Part I
$\bar{\delta}$	midpoint central difference operator (e.g., $\bar{\delta}_\xi = (E_\xi^{+\frac{1}{2}} - E_\xi^{-\frac{1}{2}})/(\Delta\xi)$)
δ^*	displacement thickness
Θ	flow deflection angle
θ	momentum thickness
κ	total conductivity
μ	total viscosity
ξ, η, ζ	transformed coordinates
ρ	density
ϕ	scalar potential function
ϑ, χ, ψ	vector potential functions
$\vec{\omega}$	vorticity vector
ω	overrelaxation parameter or vorticity magnitude

Superscripts/Subscripts

e	edge of boundary layer
i	inviscid value
j, k, l	ξ, η, and ζ indices
n	iteration number
r	reference value
t	derivative with respect to time
v	viscous value
w	wall value
0	stagnation value
∞	free stream value

INTRODUCTION

The simulation of three-dimensional flows by solving finite-difference approximations to the Navier-Stokes equations is becoming more routine. However, because of large computer memory and time requirements, it is often impractical to compute these flows using the fine grids necessary to resolve all of the high gradients and accurately predict some global parameters, such as the drag. Improving accuracy by grid refinement can be costly because more iterations are required to converge to a solution and each iteration costs more. Increasing the resolution by an order of magnitude may, for example, increase the required computer time by two orders of magnitude.

In many applications the Navier-Stokes equations are solved throughout the entire flow field, although the majority of the flow can be accurately described by simpler equations which can be solved using less computer resources. An alternative is to take advantage of the physics of the specific zones of the flow. For example, the potential flow far from the body can be resolved by solving the scalar potential equation, shear layers can be simulated by solving the boundary-layer equations, while the Euler or Navier-Stokes equations can be used in any remaining regions. The savings will be significant if the optimal algorithm is used to accurately resolve each flow region. However, a potential disadvantage of such a scheme is the increased complexity (and possible numerical instability) of treating multiple solution algorithms, solution domains, and domain interfaces. This difficulty is usually accentuated when the generality of the Navier-Stokes equations must be retained.

This paper describes some of our experiences in applying viscous-inviscid interaction methods as well as efforts towards generalizing and simplifying these procedures. Because efficient boundary-layer algorithms are the "backbone" of all viscous-inviscid interaction efforts, two- and three-dimensional finite-difference boundary-layer algorithms are described in Part I of this paper. Some of the viscous-inviscid interaction schemes which we have explored are described in Part II.

PART I: BOUNDARY-LAYER FORMULATIONS

Many important flows include mildly separated boundary layers which still satisfy the thin shear-layer assumption used in deriving the boundary-layer equations.[1] Therefore, it was natural that early investigators tried to use the boundary-layer equations to study separation. However, it was soon realized[2] that any attempt to march the boundary-layer equations through a separation point met with failure. This motivated Goldstein[3] to study the behavior of the boundary-layer equations at separation. Goldstein showed that near separation the boundary-layer equations may generate a singular solution unless the prescribed pressure distribution satisfies very restrictive relations. Hence, researchers speculated that the pressure distribution should not be merely specified to the boundary-layer equations, but should result from a strong interaction between the viscous and inviscid flow fields.[4] This concept resulted in the successful application of the boundary-layer equations to separated flow.

Two approaches arose for marching the boundary-layer equations through a separation point. The first to appear was based on the concept of solving the inviscid and boundary-layer equations simultaneously. This approach was first applied to the interaction between a supersonic inviscid flow and a separating boundary layer.[5−9] The concept has since been extended to the computation of the transonic separated flow about airfoils,[10] and the computation of steady and unsteady diffuser flows,[11−12] to cite two examples. A variation of this concept known as the quasi-simultaneous approach has also been used. In this approach the full inviscid equations are solved, and the viscous flow is then computed simultaneously with a simplified inviscid flow model. This cycle continues until convergence.[13] However, it has proven challenging to extend either the simultaneous or quasi-simultaneous viscous-inviscid interaction concepts to allow the use of the more complex inviscid and viscous flow models in external flow applications.

The second method of applying the boundary-layer equations to separated flow arose from the observation that if the pressure was not specified to the boundary-layer equations, nonsingular separation point solutions would be obtained. Of course, some forcing function other than the pressure distribution would have to be specified. Catherall and Mangler[14] successfully marched the boundary-layer equations through a separation point by specifying a displacement thickness rather than a pressure distribution. Since the displacement thickness is typically one of the results of a boundary-layer computation, Catherall and Mangler's algorithm was referred to as an inverse method, in contrast to the classical direct pressure specified algorithms. Both integral[15-16] and finite-difference[17-21] inverse boundary-layer algorithms have since been developed.

Solution algorithms for the boundary-layer equations are perhaps equally divided between those which use integral methods and those using either finite-difference or finite-element methods. Integral boundary-layer methods are quite fast and with use of good correlation functions, they can give good accuracy. However, they require modeling of the flow physics in addition to the turbulence model. As a result it is difficult to decide whether errors are due to turbulence modeling or to the integral method modeling. The integral methods are also user intensive since they require extensive correlation with experimental data. Finally, accurate integral methods have not appeared for separated three-dimensional flow. For these reasons we have preferred finite-difference boundary-layer methods. (It has also been our experience that once the boundary-layer and inviscid flow equations are coupled, the computer time savings accrued by using an integral over a finite-difference boundary-layer scheme is minimal.)

A. Steady Two-Dimensional Formulation

1. Equation Set

The nondimensionalized, compressible boundary-layer equations for the steady, two-dimensional flow of a perfect gas are

x-momentum equation

$$\rho(uu_x + vu_y) = -\beta p_x + (\mu u_y)_y \tag{1a}$$

energy equation

$$\rho c_p(uT_x + vT_y) = \beta u p_x + (\kappa T_y)_y + \mu(u_y)^2 \tag{1b}$$

perfect gas equation

$$p = \rho T \tag{1c}$$

continuity equation

$$(\rho u)_x + (\rho v)_y = 0 \tag{1d}$$

where x, y are along and normal to the body or wake centerline, respectively, and u and v are velocity components in the corresponding directions. Viscosity, pressure, temperature, and density are nondimensionalized by their free stream values. The u and x variables are nondimensionalized by the free stream velocity and a characteristic length, respectively. The v and y variables are nondimensionalized by the same quantities divided by $\sqrt{Re_\infty}$.

To solve the boundary-layer equations using finite-difference approximations it is necessary to construct a grid. In the interest of accuracy and computational efficiency it is desirable to use a nonuniform grid. However, if Eqs. (1a-d) are differenced on a nonuniform grid, complicated variable-spacing finite-difference operators must be used. This problem can be avoided if a general x, y to $\xi(x), \eta(x, y)$ coordinate transformation is applied to the boundary-layer equations. A nonuniform x, y grid may then be used while the equations can be solved on a uniform ξ, η grid. Thus, the physical domain x, y grid is adapted to resolve the flow field, while the computational domain ξ, η grid is chosen so that simple equal-spaced finite-difference operators may be used (Fig. 1). Although the equations are slightly more complicated after the coordinate transformation, overall the finite-difference solution of the equations is much simpler.

For computational convenience, Eqs. (1a-d) are transformed from the physical x, y domain to a uniform $\xi(x), \eta(x, y)$ computational domain
x-momentum equation

$$\rho[u(u_\xi \xi_x + u_\eta \eta_x) + v u_\eta \eta_y] = -\beta p_\xi \xi_x + (\mu u_\eta \eta_y)_\eta \eta_y \tag{2a}$$

energy equation

$$\rho c_p[u(T_\xi \xi_x + T_\eta \eta_x) + v T_\eta \eta_y] = \beta u p_\xi \xi_x + (\kappa T_\eta \eta_y)_\eta \eta_y + \mu(u_\eta \eta_y)^2 \tag{2b}$$

perfect gas equation

$$p = \rho T \tag{2c}$$

continuity equation

$$(\rho u)_\xi \xi_x + (\rho u)_\eta \eta_x + (\rho v)_\eta \eta_y = 0 \tag{2d}$$

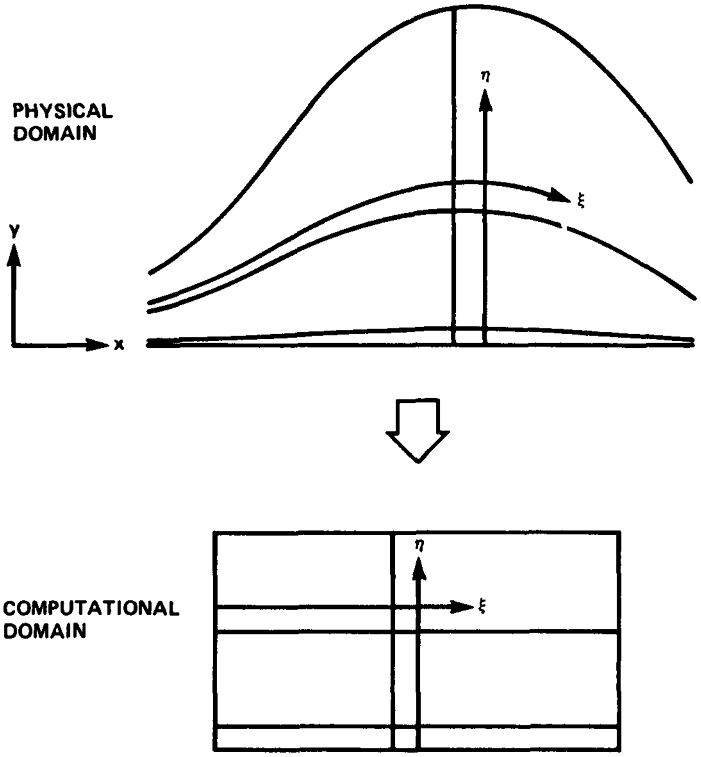

Fig. 1 Two-dimensional boundary-layer physical to computational domain coordinate transformation.

2. Solution Algorithm

Because the steady boundary-layer equations are parabolic in space, it is convenient and economical to solve the equations by marching along the body. In two-dimensional flows, the marching direction is clear and space marching schemes are quite successful. However, in three dimensions the marching direction is not clear and there are difficulties in applying marching schemes to complex three-dimensional flows. (We will return to this point in the next section.)

The relevant length scales in a boundary layer (e.g., the boundary-layer thickness and a characteristic length in the flow direction) differ in order by a power of the Reynolds number. For high Reynolds numbers, this implies a stiff numerical problem and implicit schemes are generally required. One of the more popular of these algorithms is the "box scheme" originally developed by Keller and Cebeci,[22] which Cebeci has shown is applicable to a wide range of thin shear-layer problems.[23] However, this and other implicit algorithms which solve the boundary-layer equations simultaneously require the solution of block tridiagonal matrices. Since the boundary-layer equations are weakly coupled, a numerical algorithm which solves the boundary-layer equations in a sequential manner at each streamwise station may be more appropriate and efficient.

The boundary-layer equations can be solved using an implicit predictor-corrector algorithm.[24–25] Streamwise marching begins by predicting estimates of u, T, ρ, v, μ, and κ. This predictor step uses first-order-accurate ξ operators in the x-momentum and energy equations, but for one step produces second-order-accurate values. Because the predictor step only needs to be first-order accurate, any nonlinear coefficients can be lagged in ξ. A second-order-accurate corrector step is then used to calculate improved values of u, T, ρ, v, μ, and κ. During the corrector step, the nonlinear coefficients are evaluated using the most recently computed flow variables. At each step the equations are solved sequentially. First u is updated from the x-momentum equation. With u available, T is updated from the energy equation, ρ is then obtained from the equation of state, and v can then be updated from the continuity equation. Overall, with this predictor-corrector scheme, second-order-accurate solutions are obtained at the cost of two scalar bidiagonal and four scalar tridiagonal matrix inversions per streamwise station.

In reversed flow regions, the streamwise convection terms in the x-momentum and energy equations can be evaluated using flow-dependent difference operators. These flow-dependent operators must model the proper zone of influence at every grid point. For example, a three-point backward difference operator may be used when $u > 0$, while a three-point forward difference operator would be used in the reversed flow regions. This has been avoided in the past because of the requirement of repeatedly sweeping through a reversed flow region. A popular approximation (called the FLARE[26] approximation) avoids the need for multiple sweeps through separation regions by setting the streamwise convection terms in the momentum and energy equations to zero in separated regions. These one-sweep solutions are achieved at the expense of assuming that the upstream influence in reversed flow regions is small. However, if the viscous algorithm is part of a viscous-inviscid interaction code, repeated sweeping of the viscous flow is required as part of the viscous-inviscid iteration. In this case, upstream influence in the reversed flow region can be obtained during the repeated sweeping required by the viscous-inviscid iteration for little additional computational cost.

Near and in reversed-flow regions, the boundary-layer equations are solved in the inverse mode to avoid singular behavior at the separation point. While various researchers use δ^*, here the wall shear stress τ_w and the wake centerline velocity u_{wc} are used because they can be readily and efficiently implemented. The algorithm is modified to operate in the inverse mode by replacing the pressure term in the x-momentum equation with an expression containing τ_w or u_{wc}. These relations are obtained by applying the x-momentum equation at the wall and wake centerline. For example, at a wall the x-momentum equation yields

$$\beta\xi_x p_\xi = (\mu u_\eta \eta_y)_\eta \eta_y|_w \tag{3}$$

The right-hand side of Eq. (3) is evaluated using first-order finite-difference approximations which allow the elimination of the pressure term $(\beta\xi_x\delta_\xi p)$ from the x-momentum equation. However, in the inverse mode the value of u at $k = 2$, i.e., u_2, appears in the difference equation at each and every k index. If this term is treated implicitly, then the matrix is tridiagonal with an additional column of nonzero coefficients

$$\begin{pmatrix} b_2 & c_2 & & & & \\ f_3 & b_3 & c_3 & & & \\ f_4 & a_4 & b_4 & c_4 & & \\ \cdot & & \cdot & \cdot & \cdot & \\ \cdot & & & \cdot & \cdot & c_{kmax-1} \\ f_{kmax} & & & & a_{kmax} & b_{kmax} \end{pmatrix} \mathbf{u} = \mathbf{d} \tag{4}$$

This augmented scalar tridiagonal matrix system is efficiently solved using an algorithm developed specifically for this application. The algorithm uses an LU decomposition to solve the augmented scalar tridiagonal system[24-25] and costs only 40% more than the standard scalar tridiagonal inversion. This approach results in an implicit treatment of the inverse boundary conditions.

B. Unsteady Three-Dimensional Formulation

As will be shown in Part II of this paper, the steady two-dimensional formulation just described has been quite useful. However, the extension of the space-marching boundary-layer approach to three dimensions is not straightforward because there is not a clearly defined marching direction. Our experience in three dimensions is that a time-relaxation approach to solving the boundary-layer equations is preferable.[27-28]

By relaxing the boundary-layer equations in time, flow-dependent difference operators which automatically adjust to the flow direction can replace problem-dependent space sweeps. Hence, one relatively simple code can be easily applied to a wide variety of three-dimensional flows without any changes to the algorithm. An additional benefit of relaxation schemes is that they are also more robust and forgiving of approximate boundary conditions than space-marching schemes. With a space-marching scheme, all variables must be specified at inflow and some side boundaries, whereas simple outflow, reflection, periodic, or zero-gradient boundary conditions can often be used with a relaxation numerical algorithm. For example, it would be very difficult to specify the flow quantities near a wing tip, as may be required by a space-marching scheme. However, with a relaxation algorithm the tip values can be brought up during the iteration process using, for example, a zero-gradient boundary condition. On the other hand, compared to space-marching algorithms, relaxation schemes can be relatively computationally expensive when used to simulate simple steady flows. This drawback can be minimized if care is taken to design the relaxation algorithm so that a steady state can be reached in few iterations. Also, since a relaxation boundary-layer scheme can be approximately half as expensive as a second-order space-marching scheme on a per-iteration basis, when included in a viscous-inviscid interaction algorithm the relaxation boundary-layer scheme should result in a more efficient code. Finally, the same code can also be used to simulate unsteady flows.

1. Equation Set

Neglecting surface curvature, the compressible boundary-layer equations for the unsteady, three-dimensional flow of a perfect gas can be written in $\xi(x,z)$, $\eta(x,y,z)$, $\zeta(x,z)$ coordinates as

x-momentum equation

$$\rho(u_t + Uu_\xi + Vu_\eta + Wu_\zeta) = -\beta(p_\xi \xi_x + p_\zeta \zeta_x) + (\mu u_\eta \eta_y)_\eta \eta_y \tag{5a}$$

z-momentum equation

$$\rho(w_t + Uw_\xi + Vw_\eta + Ww_\zeta) = -\beta(p_\xi \xi_z + p_\zeta \zeta_z) + (\mu w_\eta \eta_y)_\eta \eta_y \tag{5b}$$

perfect gas relation

$$p = \rho T \tag{5c}$$

energy equation (H=total enthalpy)

$$\rho(H_t + UH_\xi + VH_\eta + WH_\zeta) = \left[\frac{\mu}{Pr}\left\{H_\eta \eta_y + \frac{Pr-1}{2}(u^2+w^2)_\eta \eta_y\right\}\right]_\eta \eta_y + p_t \tag{5d}$$

continuity equation

$$\rho_t + (\rho u)_\xi \xi_x + (\rho u)_\eta \eta_x + (\rho u)_\zeta \zeta_x + (\rho v)_\eta \eta_y + (\rho w)_\xi \xi_z + (\rho w)_\eta \eta_z + (\rho w)_\zeta \zeta_z = 0 \tag{5e}$$

where U, V, and W are unscaled contravariant velocities

$$\begin{bmatrix} U \\ V \\ W \end{bmatrix} = \begin{bmatrix} \xi_t + & \xi_x u + & & \xi_z w \\ \eta_t + & \eta_x u + & \eta_y v + & \eta_z w \\ \zeta_t + & \zeta_x u + & & \zeta_z w \end{bmatrix}$$

In the interest of simplicity, the boundary-layer equations and algorithm are presented here neglecting surface curvature. To date, we have included the terms to treat body-conforming orthogonal curvilinear coordinates.[29−31] Eventually, a general treatment of surface curvature should be included.[32]

2. Solution algorithm

Once again, because the pressure gradient forcing terms in the momentum equations are treated as given functions, the boundary-layer equations are weakly coupled and can be solved sequentially at each time step. As a result, a semi-implicit algorithm can be used at each time or iterative step, yet only scalar-like uncoupled equations are solved. Spatial derivatives in the momentum (and energy) equations are approximated with implicit second-order-accurate central-difference operators in the direction normal to the body (η), and flow-dependent second-order accurate operators in the other two directions (ξ and ζ). That is, central differencing is used in η, and upwind differencing is used in ξ and ζ. For example, when differencing a term such as u_ξ a backward difference is used if the coefficient ρU is positive and a forward difference is used if the coefficient ρU is negative. In the continuity equation, the ξ and ζ derivatives are approximated with second-order-accurate central-difference operators, and trapezoidal-rule integration is used in the η direction to yield v.

The time-accurate algorithm is presented below using conventional operators (defined in the nomenclature) and the second-order-accurate upwind operator $\widehat{\delta}$

$$\widehat{\alpha}\widehat{\delta}_\xi = \left(\frac{\alpha + |\alpha|}{2}\right) \nabla_\xi \left(\frac{3 - E_\xi^{-1}}{2}\right) + \left(\frac{\alpha - |\alpha|}{2}\right) \Delta_\xi \left(\frac{3 - E_\xi^{+1}}{2}\right)$$

where α is the convective coefficient (e.g., ρU). Using a notation that space-time indices are not written unless changed (e.g., $u = u^n_{j,k,l}, u^{n+1} = u^{n+1}_{j,k,l}$, etc.), the semi-implicit scheme is described below.

From the x-momentum equation, the velocity u at the new time step is updated from

$$\rho(\nabla_t u^{n+1} + \widehat{U}\widehat{\delta}_\xi u^{n+1} + V\delta_\eta u^{n+1} + \widehat{W}\widehat{\delta}_\zeta u) = -\beta(\xi_x \delta_\xi p + \zeta_x \delta_\zeta p) + \eta_y \bar{\delta}_\eta (\mu \eta_y \bar{\delta}_\eta u^{n+1}) \quad (6a)$$

where ($\widehat{U} = \xi_x u + \xi_z w, etc...$) and where if $\widehat{U} < 0$, $\widehat{\delta}_\xi$ will actually operate on explicit data. As noted below, care is taken to maintain stability of terms that are explicitly three-point differenced. This system of equations is easily solved by inverting scalar-tridiagonal matrices in η. In a similar way, w is updated from the z-momentum equation ($\widehat{U} = \xi_x u^{n+1} + \xi_z w, etc...$)

$$\rho(\nabla_t w^{n+1} + \widehat{U}\widehat{\delta}_\xi w^{n+1} + V\delta_\eta w^{n+1} + \widehat{W}\widehat{\delta}_\zeta w) = -\beta(\xi_z \delta_\xi p + \zeta_z \delta_\zeta p) + \eta_y \bar{\delta}_\eta (\mu \eta_y \bar{\delta}_\eta w^{n+1}) \quad (6b)$$

and H is updated from the energy equation ($\widehat{U} = \xi_x u^{n+1} + \xi_z w^{n+1}, etc...$)

$$\begin{aligned} &\rho(\nabla_t H^{n+1} + \widehat{U}\widehat{\delta}_\xi H^{n+1} + V\delta_\eta H^{n+1} + \widehat{W}\widehat{\delta}_\zeta H) \\ &\quad = \eta_y \bar{\delta}_y \left[\frac{\mu}{Pr} \left\{ \eta_y \bar{\delta}_\eta H^{n+1} + \frac{Pr-1}{2} \eta_y \bar{\delta}_\eta (u^2 + w^2)^{n+1} \right\} \right] + \nabla_t p^{n+1} \end{aligned} \tag{6c}$$

Finally, obtain $\rho = p/T$ and integrate the continuity equation for v using updated values of u, w, and ρ.

$$\begin{aligned} \nabla_\eta(\rho v)^{n+1} = -\frac{1+E_\eta^{-1}}{2} [&\xi_x \delta_\xi(\rho u) + \eta_x \delta_\eta(\rho u) + \zeta_x \delta_\zeta(\rho u) \\ &+ \xi_z \delta_\xi(\rho w) + \eta_z \delta_\eta(\rho w) + \zeta_z \delta_\zeta(\rho w) + \nabla_t \rho]^{n+1} \end{aligned} \tag{6d}$$

The coefficients μ and Pr are then evaluated using a turbulence model, here either the Cebeci algebraic turbulence model[33] (generalized to three dimensions) or the Baldwin-Lomax model.[34]

The algorithm just presented is second-order accurate in space and first-order accurate in time. A simple corrector sequence can be used to achieve second-order time accuracy and provide additional stability of the explicitly differenced terms following the same technique employed in the two-dimensional space-marching algorithm described earlier. However, if only a steady-state solution is required, enhanced stability and faster convergence is obtained by implicitly including the diagonal contribution of any explicit operator. In this case the algorithm takes on a similarity to a SLOR scheme. Finally, in many steady-state cases an assumption of constant total enthalpy can replace the partial-differential energy equation.

If no streamwise separation is present, these equations may be solved with the pressure specified (direct mode). If streamwise separation occurs, the inverse mode must be used and this is implemented as outlined for the two-dimensional case. It is interesting to note that it is possible to solve the three-dimensional boundary-layer equations in a mixed direct/inverse mode. For example, the x-momentum equation may be solved in the inverse mode, while the z-momentum equation can be solved in the direct mode, or vice-versa. The authors have observed that the saddle-point behavior at the streamwise separation line may be avoided if just the momentum equation in roughly the free stream direction is solved in the inverse mode (i.e., the other momentum equation can be solved in the direct mode).

This algorithm readily lends itself to vector computer architecture by arranging the computation of the inversion matrix coefficients and the actual inversion in a vectorized loop. To do this, one spatial direction must be chosen along which to create the vectors. It is convenient to choose the direction along the body roughly normal to the free stream. Then at each time step, along each ξ plane, all the coefficient matrices are computed, stacked together, and fed to a vectorized version of a scalar tridiagonal (or augmented tridiagonal) solver. Using this approach, the algorithm requires approximately 2 μs/point/iteration on a CRAY-XMP processor.

3. Results

Detailed experimental data have been obtained for the flow over a 6:1 prolate spheroid at angle of attack.[35-38] The u_∞=45 m/sec transitional flow over the ellipsoid at 30° angle of attack was computed using the relaxation boundary-layer code. The experimental pressure and near-surface particle traces are presented in Figs. 2-3, respectively. The experimental particle traces were obtained by computing the streamlines of particles released on the body (and required to remain on the body) given the experimental wall skin-friction data.

The flow was computed by solving the boundary-layer equations, including the non-unity geometrical metric coefficients outlined by Wang,[29] Cebeci,[30] and others, on a 45 (axial direction) × 50 (normal to the body) × 25 (circumferential direction) mesh. A cubic spline representation of the experimental pressure distribution was specified as the forcing function. In Figs. 4a-c computed surface streamlines are presented for three conditions: fully laminar, transition at the cross-flow separation line, and fully turbulent. From these three figures it is clear that this flow is very sensitive to the laminar/turbulent state of the boundary layer. Specifically, in the fully laminar and transitional computations, the streamlines indicate two counterrotating streamwise vortices (indicated by the coalescence/dispersion/coalescence streamline patterns in the meridional direction), while in the fully turbulent case, only one fairly small streamwise vortex is indicated by the one streamwise coalescence line. By comparison with the experimental particle traces it is apparent that the experimental flow is neither entirely laminar nor turbulent, and that' the transitional result is in the closest agreement to the experimental data. The simple transition model used here has underpredicted the extent of turbulent flow, and as a result a slightly larger separated region is predicted than is observed experimentally. However, it is clear that these flows can be computed routinely with the present code and a refined transitional model could be developed using this code. Figure 5 is a close-up of the captured stagnation point. As mentioned earlier, with the present algorithm there is no special treatment of the stagnation region since there is no requirement to march away from the stagnation point.

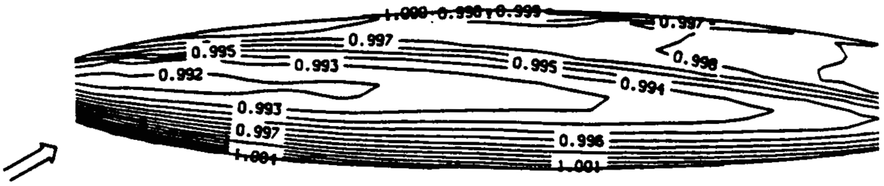

Fig. 2 Measured pressure distribution[37] on a 6:1 prolate spheroid at 30° angle of attack, $Re = 7.2 \times 10^6$, and $M_\infty = 0.13$.

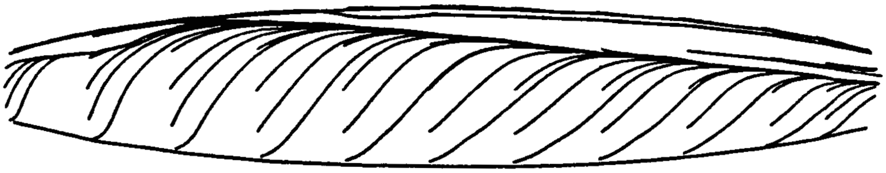

Fig. 3 Measured near-surface particle paths for the flow over the prolate spheroid of Fig. 2.

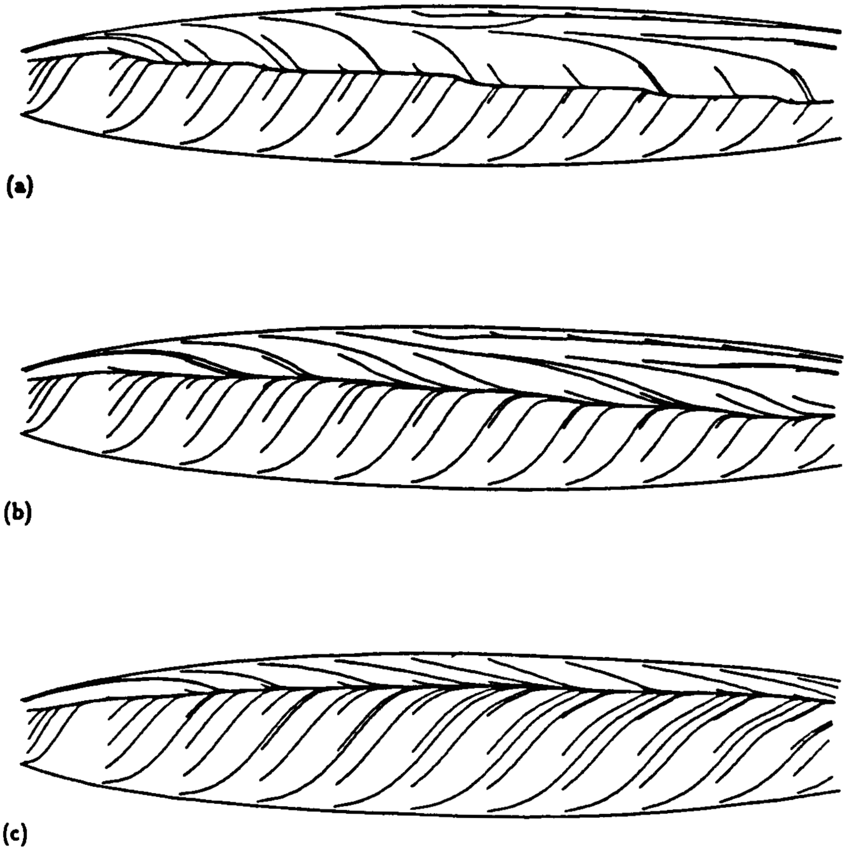

Fig. 4 Computed near-surface particle paths for the flow over the prolate spheroid of Fig. 2: a) completely laminar; b) transition at the cross-flow separation line; c) completely turbulent.

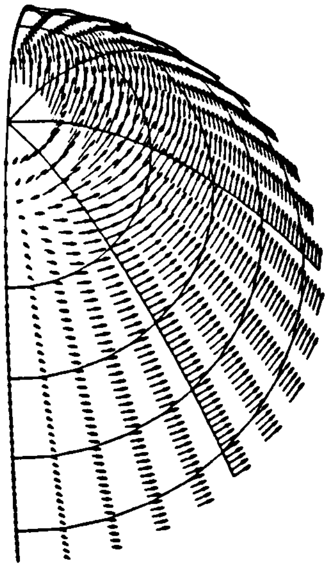

Fig. 5 Closeup of captured stagnation point of the flow over the prolate spheroid of Fig. 2.

The transonic flow over a NACA 0012 wing with a 20° sweep and an aspect ratio of 3 has been studied by Lockman and Seegmiller[39] over a range of free stream Mach numbers and angles of attack. At $M_\infty = 0.826$, $\alpha = 2°$, a strong shock runs across the wing (Fig. 6) which contributes to the formation of complex streamwise and cross-flow separation regions. Unfortunately, the experimental data are not of sufficient detail to supply the input for a boundary-layer computation. However, this flow has also been computed using a Navier-Stokes algorithm[40] (called TNS) which does supply detailed results. In general, the pressure distribution generated by the TNS code is in good agreement with the experimental data. Consequently, the pressure distribution generated by the TNS code can be used as input for a boundary-layer computation, and this was done by operating in the inverse mode and iterating to the TNS pressure distribution. It is important to note that if this iteration procedure were to converge to exactly the specified pressure distribution, saddle-point behavior would be encountered at any streamwise separation line. To avoid this, the relaxation parameter ω which appears in the expression to iteratively update the shear stress $(\tau_w^{n+1} = \tau_w^n + \omega(p_v^n - p_{TNS}^{n+1}))$ is made proportional to the absolute value of the computed wall shear stress. (A complete description of the scheme used to update the wall shear stress, Eq. (8) with $p_i = p_{TNS}$, is given in §A.1 of Part II of this report.) In this way, the specified pressure distribution is accurately matched, but the saddle-point behavior is avoided since the iteration procedure never reaches exactly the specified pressure at the separation line. The experimental oil-flow and computed near-surface particle paths are compared in Figs. 7-8. As shown, details of the flow near the tip, the "mushroom" separation region, and (farther inboard) the continued deflection of the flow through the shock are all captured. Overall, even for this complex case, the agreement between the experimental and computed boundary-layer flows is quite encouraging.

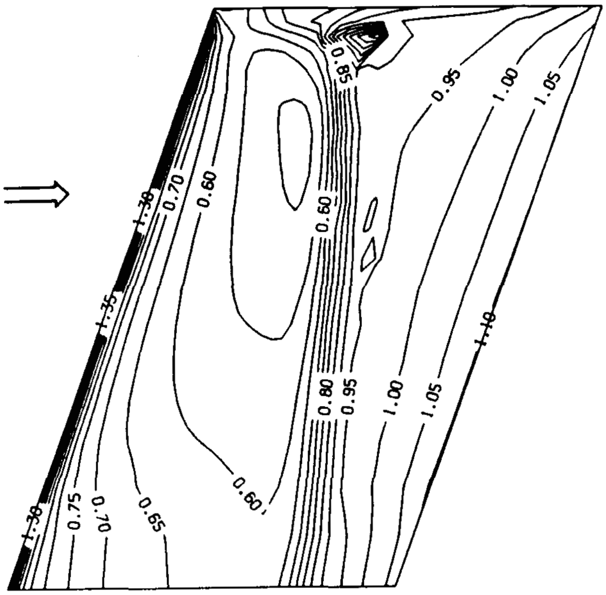

Fig. 6 Computed pressure contours[40] on a NACA 0012 wing with AR=3, 2° angle of attack, $Re = 8.0 \times 10^6$ and $M_\infty = 0.826$.

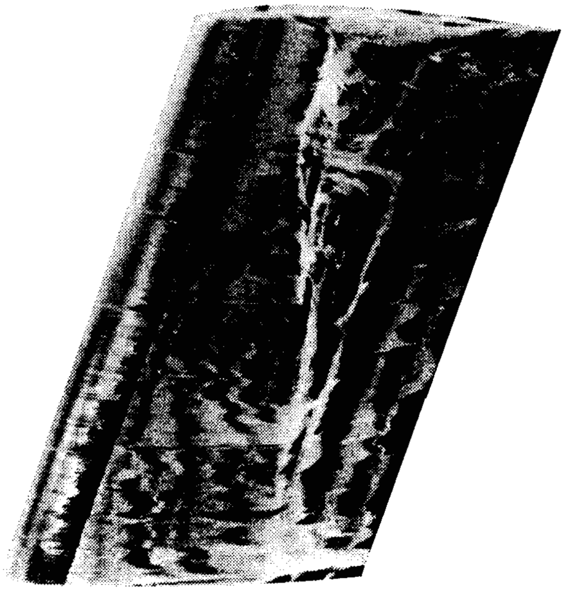

Fig. 7 Experimental oil-flow patterns[39] for the flow over the NACA 0012 wing of Fig. 6.

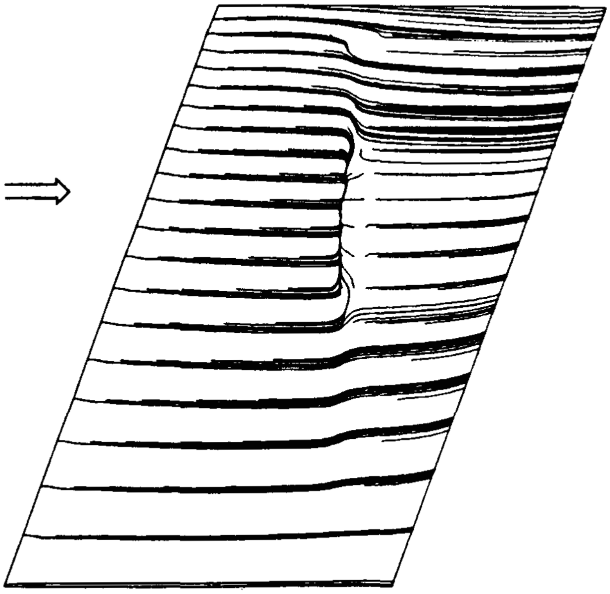

Fig. 8 Computed near-surface particle paths for the flow over the NACA 0012 wing of Fig. 6.

PART II: INTERACTION SCHEMES

Three approaches to the viscous-inviscid interaction problem are briefly reviewed. First, a conventional full-potential/boundary-layer transpiration boundary-condition matching procedure will be discussed. Although this approach is useful and fairly flexible, it is our experience that it is difficult to apply in some cases. For this reason alternative interaction procedures are being explored.[28,41-43] Two of these, a vorticity-interaction scheme and the Fortified Navier-Stokes (FNS) approach, are presented here. Many alternative interaction procedures are being studied. However, we feel that the two presented here are promising and a discussion of them will outline some of the basic requirements for the next generation of interaction procedures.

A. Full-Potential/Boundary-Layer Interaction

The goal of a viscous-inviscid interaction algorithm is to obtain converged and compatible viscous and inviscid flow field solutions. To achieve this the viscous-inviscid iteration scheme must allow each flow to be influenced by the other, yet remain stable and computationally efficient even when this interaction is strong. The "classical" interaction approach is to obtain an approximation to the inviscid flow, extract some information from the inviscid flow (e.g., p) to feed to the viscous algorithm, and finally extract information from the viscous algorithm (e.g., δ^*) and input it to another estimate of the inviscid flow. This cycle (Fig. 9) is continued until the inviscid and viscous solutions are converged and compatible. The interface schemes for transferring information between the inviscid and viscous algorithms are critical and still a subject of active research. In the next two sections, two-dimensional examples of the more "classical" interfaces are discussed.

1. Inviscid to Viscous Interface

If the flow remains attached, the boundary-layer equations can be solved with pressure specified (the direct mode), where the pressure is obtained from the inviscid solution. The need to switch to the inverse mode near regions of reversed flow complicates the inviscid to viscous interface in two ways. First, the interface must "sense" when the viscous flow is about to separate and then change from the direct to the inverse mode. Second, since the inverse forcing functions (τ_w and u_{wc}) cannot be supplied directly by the inviscid algorithm, interaction algorithms must be developed which supply these forcing functions. These interaction algorithms should supply inverse forcing function distributions which result in the inviscid and viscous flow algorithms rapidly converging to compatible solutions in as few viscous-inviscid iterations as possible. Since the converged solutions will not depend upon the form of the interaction algorithms, only the requirement of computational efficiency need influence the design of these interface algorithms.

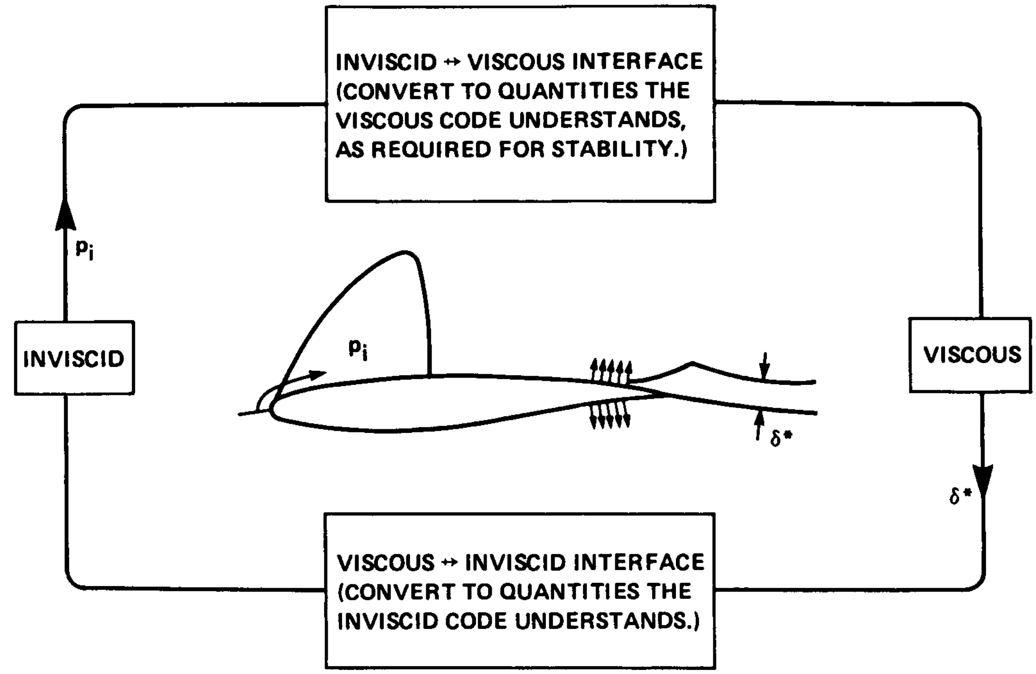
INVISCID ↔ VISCOUS INTERFACE (CONVERT TO QUANTITIES THE VISCOUS CODE UNDERSTANDS, AS REQUIRED FOR STABILITY.)
p_i
INVISCID
p_i
VISCOUS
δ^*
δ^*
VISCOUS ↔ INVISCID INTERFACE (CONVERT TO QUANTITIES THE INVISCID CODE UNDERSTANDS.)

Fig. 9 "Classical" interaction scheme.

Monitoring the evolution of the wall shear stress is a simple way of determining if a boundary layer is about to separate. In a typical boundary-layer computation it is reasonable to begin the computation in the direct mode, switching to the inverse mode only if τ_w falls below a prescribed small positive value. Once beyond the reversed flow regions it is possible to switch back to the direct mode. However, for simplicity, in the present work the viscous flow computation is continued in the inverse mode even if the flow has reattached.

Once in the inverse mode, inviscid to viscous interface algorithms must supply inverse forcing function distributions. A variety of schemes for updating the wall shear stress have been studied. The integral boundary-layer equation[44] indicates that skin friction is a function of the pressure gradient

$$\frac{C_f|_e}{2} = \frac{\theta}{u_e}\frac{du_e}{dx}(2 + H - M_e{}^2) + \frac{d\theta}{dx} \tag{7a}$$

Likewise, evaluation of the x-momentum equation at the wall (Eq. (3)) suggests a relaxation equation of the form

$$\tau_w^{n+1} = \tau_w^n + \omega\left(\frac{\frac{\mu_2+\mu_1}{2}\frac{u_2-u_1}{y_2-y_1} - \tau_w^*}{(y_2-y_1)/2} - \beta p_x|_i^{n+1}\right)\left(\frac{y_2-y_1}{2}\right) \tag{7b}$$

where $\tau_w^* = \tau_w^n$ or τ_w^{n+1} and ω is a relaxation parameter. It was found that schemes based upon this type of equation worked well for some separated flows, such as incompressible diffuser flow or transonic airfoil flow with weak shocks. However, strong shock cases generate large pressure gradients which caused all of the schemes based on pressure gradient to be unreliable. A scheme based simply upon the difference between the viscous and inviscid pressures was found to be more reliable and equally fast. The results presented in this work were obtained using the simple scheme

$$\tau_w^{n+1} = \tau_w^n + \omega(p_v^n - p_i^{n+1}) \tag{8}$$

In all cases, an over-relaxation factor of ten (i.e., $\omega = 10$ in Eq. 8) was used to accelerate convergence.

This scheme is based on an observation about the nature of boundary-layer flow. It was noted that at a given streamwise station an increase in the specified τ_w will result in an increased computed u_e or a decreased pressure (Fig. 10). Hence, if the pressure computed by the viscous algorithm is greater than the desired inviscid pressure, the τ_w should be increased from the value used in the previous iteration. A similar argument holds in the situation where an increased viscous pressure is desired, in which case a decrease in τ_w is required. Carter[45] developed a similar scheme to update the displacement thickness specified to an inverse code. As mentioned previously, the only criterion these schemes must meet is that they produce matched inviscid and viscous solutions in as few iterations as possible.

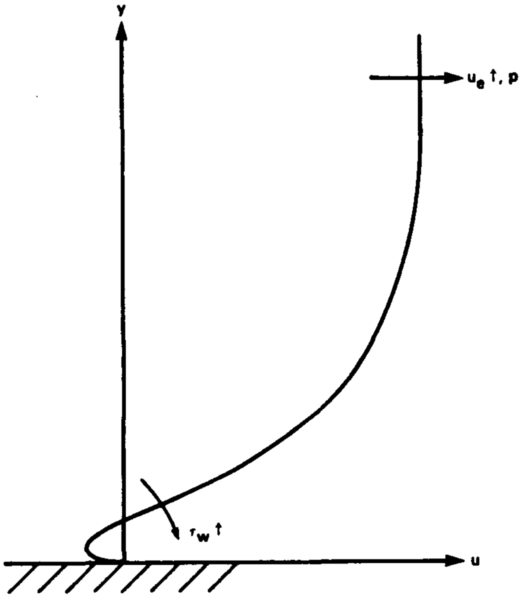

Fig. 10 Relation between τ_w and p at a given streamwise station.

2. Viscous to Inviscid Interface

The classical method of introducing the influence of the viscous flow upon the inviscid flow is to add the displacement thickness distribution to the original body shape, and then compute the inviscid flow about this new body. The advantage of this method is its simplicity. However, there are two disadvantages to this approach.

If the displacement thickness approach is used, the inviscid grid must be regenerated after each viscous iteration. The body-conforming inviscid grid must be regenerated because the body about which the inviscid flow is computed changes when the displacement thickness distribution changes, after each viscous iteration. The computational cost of regenerating the inviscid grid may be minimized by simply shearing the original inviscid grid. However, we have found that this can still be costly and may lead to unsuitable inviscid grids.

The second disadvantage of the displacement thickness approach is the possibility of supercritical viscous-inviscid interaction. First, it should be explained that subcritical and supercritical interactions are distinguished by the response of the flow deflection angle Θ (the ratio of the inviscid velocity components normal and parallel to the body) generated by the viscous flow to the streamwise pressure gradient. In an adverse pressure gradient a subcritical interaction results in an increased Θ, while a supercritical interaction results in a reduced Θ. (This nomenclature was first introduced by Crocco and Lees.[46]) Because of these properties a supercritical interaction cannot result in a smooth transition to separation, a process in which the pressure and boundary-layer thickness both increase. Rather, before a supercritical interaction can produce separated flow, a transition or jump back to a subcritical condition must occur. This required jump or discontinuity corresponds to a saddle-point singularity and is similar to the throat singularity of one-dimensional nozzle flow. Hence, any interaction procedure which leads to supercritical behavior and may encounter this singularity should be avoided. Le Balleur[47] has shown that the classical displacement thickness interaction becomes supercritical when $M_e \approx 1.5 \rightarrow 2.0$, depending on the history of the turbulent boundary layer. This implies that the classical displacement thickness approach should not be used in strong transonic viscous-inviscid interactions.

An alternative to adding the displacement thickness distribution to the original body shape is to impose a transpiration velocity at the body surface, as first suggested by Lighthill.[48] In the transpiration velocity approach, the original inviscid flow tangency condition at the body surface is modified to enforce a velocity component normal to the body. This normal or transpiration velocity component deflects the inviscid flow away from the body, and thus simulates the displacement of the inviscid flow by the viscous flow momentum defect. The advantages of the transpiration velocity approach are that the inviscid grid need not be regenerated after each viscous iteration, and the interaction always remains subcritical.[47] Hence, the transpiration velocity approach is the more computationally efficient and robust viscous to inviscid interface approach.

The transpiration velocity distribution should result in an inviscid streamline coinciding with the equivalent body shape obtained when the displacement thickness distribution is added to the original body shape. Such a transpiration velocity distribution may be computed as[48]

$$V_\delta = \frac{1}{\rho_e} \frac{\partial(\rho_e u_e \delta^*)}{\partial s} \tag{9}$$

where ρ_e and u_e are the inviscid values at the body (or wake centerline), and in this instance s is the distance along the body (or wake centerline). Equation (9) is obtained by integrating the difference between the inviscid and viscous continuity equations from

$y = 0 \rightarrow \delta$ (i.e., by integrating across the boundary layer), and simplifying the obtained result using two assumptions. These are the thin boundary-layer assumption, and the assumption that the inviscid flow is essentially uniform from $y = 0 \rightarrow \delta$. In the flow fields of interest in this work, these are reasonable assumptions. The transpiration velocities are then built into the body and wake boundary conditions of the inviscid code. In the case of the TAIR full-potential code[49] used in this work, V_δ is converted to perturbations of the inviscid contravariant velocities (U_p and V_p) which are included in the inviscid residual computations along the body and wake centerline (see Refs. 24-25 for details).

3. Wake Treatment and Curvature Effects

In many cases it is necessary to account for the viscous flow curvature and the pressure jump which occurs across curved streamtubes. Therefore, the method of accounting for the pressure variation across the shear layers developed by Lock and Firmin[50] is incorporated into the present algorithm. Suffice it to say that these corrections add a substantial degree of complexity to the code, but are important for high angle of attack cases and airfoils with a high aft-loading.[51] The reader is referred to Refs. 25 and 50-51 for details.

4. Complete Interaction Procedure

All of the components of the viscous-inviscid interaction which have been presented are shown schematically in Fig. 11. In an inviscid-viscous iteration it is not necessary, or even desirable, to iterate each equation set to convergence before updating the other. Indeed, the ratio of inviscid to viscous iterations ($r_{i/v}$) has a significant influence upon the efficiency of the computation. If $r_{i/v}$ is too large, then a strong viscous-inviscid interaction will not be resolved and instability will result. For example, if during the iteration process a separation bubble develops under a shock, the shock will often rapidly weaken and move upstream. If the viscous flow is not updated often enough, then the separation bubble will not "follow" the shock upstream, but will generate a spurious reacceleration of the subsonic inviscid flow downstream of the shock. That is, if $r_{i/v}$ is too large, a lag develops between the inviscid and viscous solutions which may result in a breakdown in the viscous-inviscid interaction and numerical instability. In contrast, if the viscous flow is updated more often than is necessary to resolve the viscous-inviscid interaction ($r_{i/v}$ is too small), only additional computational cost is incurred. Therefore, it is prudent to recompute the viscous flow after each inviscid iteration ($r_{i/v} = 1$) when a flow field is first simulated or if it is known that a very strong viscous-inviscid interaction occurs.

5. Results

McDevitt[52] compiled the experimental peak Mach number on the 18% circular-arc airfoil with a trailing edge splitter plate as a function of free stream Mach Number. Figure 12 compares the results of the present interaction method to this experimental data. The comparison with experiment is quite encouraging. Considering the wind tunnel wall effects documented by Levy[53] the slight discrepancy at the lower Mach numbers is expected. The discrepancy at the higher Mach numbers may be due to wind tunnel wall effects, but could also be due to the assumption of isentropic inviscid flow. The isentropic flow shock jump deviates significantly from the Rankine-Hugoniot flow shock jump as the pre-shock Mach

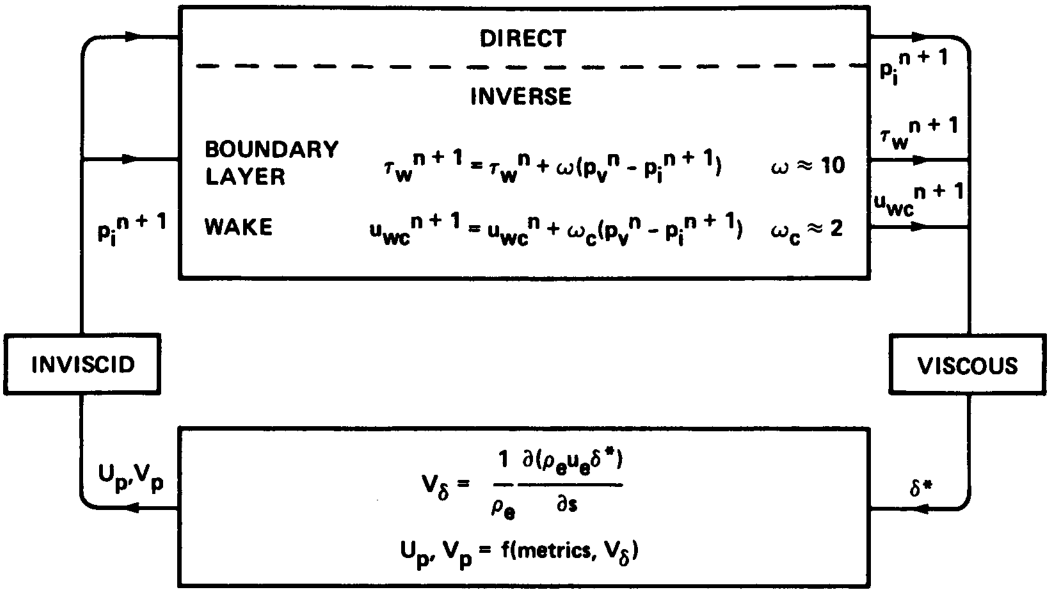
DIRECT
INVERSE
BOUNDARY LAYER
$\tau_w^{n+1} = \tau_w^n + \omega(p_v^n - p_i^{n+1})$
$\omega \approx 10$
WAKE
$u_{wc}^{n+1} = u_{wc}^n + \omega_c(p_v^n - p_i^{n+1})$
$\omega_c \approx 2$
p_i^{n+1}
τ_w^{n+1}
u_{wc}^{n+1}
INVISCID
VISCOUS
U_p, V_p
δ^*
$V_\delta = \frac{1}{\rho_e}\frac{\partial(\rho_e u_e \delta^*)}{\partial s}$
$U_p, V_p = f(\text{metrics}, V_\delta)$

Fig. 11 Complete interaction scheme.

number approaches 1.3. The dramatic difference between the inviscid and viscous peak Mach numbers is indicative of the strong viscous-inviscid interaction being modeled.

Because detailed experimental results are available,[54] the RAE 2822 has become a popular supercritical airfoil for the verification of airfoil codes. Therefore, the M_∞ = 0.730, Re_∞ = 6.5x10^6 flow about the RAE 2822 airfoil at a C_l = 0.803 was computed. Figure 13a compares the experimental C_p distribution, the C_p distribution computed by the present method, and the C_p distribution computed by Mehta[55] using a thin-layer Navier-Stokes code. There is significant wind tunnel wall interference in this data. Hence, the experimentalists suggest that computations not be performed at the measured angle of attack (α = 3.19°), but rather at a corrected angle of attack of approximately α = 2.79° or at the measured C_l = 0.803. Mehta performed his computations at the suggested corrected angle of attack (and computed a C_l = 0.793), while the present computations were performed at α = 2.81° to match the measured C_l = 0.803. It appears that some discrepancy is introduced because of the first order accurate resolution of the supersonic region by the inviscid method used in the present work. In particular, the details of the pressure distribution in the supersonic regions are not resolved. Also, the Navier-Stokes computation was able to resolve the shock profile more accurately than the present method. This discrepancy is due in part to the high degree of clustering in the Navier-Stokes grid in the region of the experimentally observed shock location. However, overall the three pressure distributions are in good agreement.

The experimental and computed $C_f|_e$ distributions for the flow over the upper surface of the RAE 2822 are presented in Fig. 13b. The boundary layers were tripped at 3% percent of chord in both the experiment and the two computations. Considering the difficulties involved in measuring skin friction, the agreement between the computed and experimental $C_f|_e$ distributions is fairly good. Both of the computations predict incipient

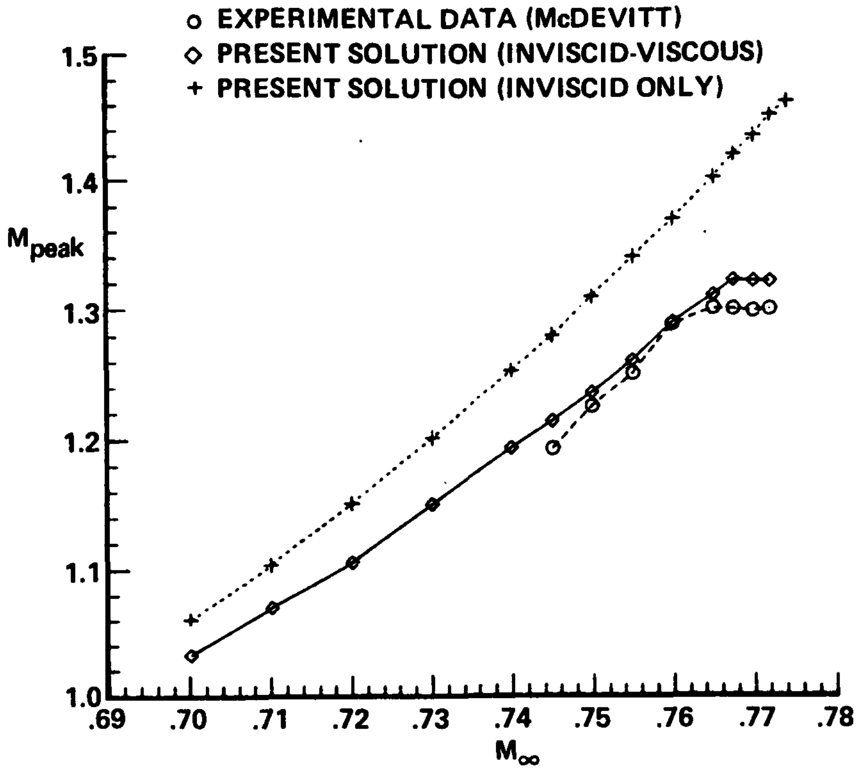

Fig. 12 Calculated and experimental M_{peak} versus M_∞ for the $Re_\infty = 8 \times 10^6$ flow about an 18% circular-arc airfoil with a trailing-edge splitter plate.

shock-induced separation, and the present method predicts a small extent of trailing edge separation.

Figures 13c-d compare the computed boundary-layer displacement thickness and shape factor distributions with those determined experimentally for the flow over the upper surface of the RAE 2822. Because it is difficult to determine the boundary-layer thickness from Navier-Stokes results, the Navier-Stokes integral boundary-layer parameters are not smooth. Nevertheless, the very rapid growth of the displacement thickness under the shock and near the trailing edge are accurately predicted. These results suggest that the eddy-viscosity turbulence models are adequate for computing attached and mildly separated airfoil-type flow fields. Also, both of the computed shape factor distributions (Fig. 13d) predict the basic experimental trends.

Chen et al.[56] extended this technology and a Navier-Stokes formulation[57] to the porous airfoil application shown in Fig. 14. (Porous airfoil research, see Ref. 56 for a complete reference list, has suggested that adding a porous surface to a wing might be a feasible drag reduction approach.) Some representative results are indicated in Figs. 15a-b. A number of interesting characteristics about this flow and the two formulations were observed during the work. In particular, it was found that for moderate cases the Navier-Stokes and viscous-inviscid interaction codes both performed well, with the interaction code being significantly more cost effective. However, for severe cases (very strong shocks or high blowing rates) it was necessary to rely on the Navier-Stokes formulation. It seems that the isentropic

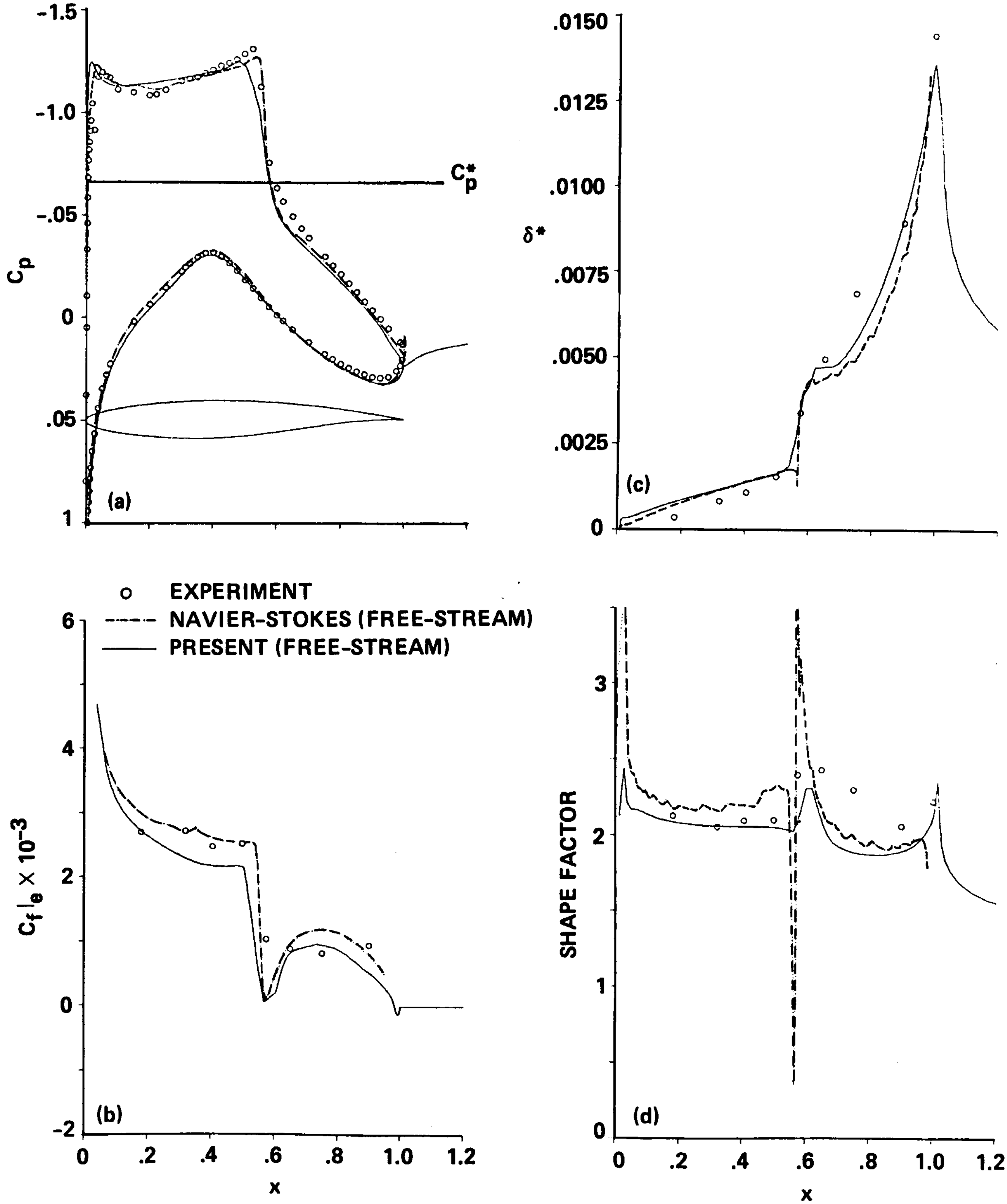

Fig. 13 Calculated and experimental results for an RAE 2822 airfoil at $M_\infty = 0.730, Re_\infty = 6.50 \times 10^6$, $C_l = 0.803$: a) C_p distributions; b) $C_f|_e$ distributions; c) δ^* distributions; d) H distributions.

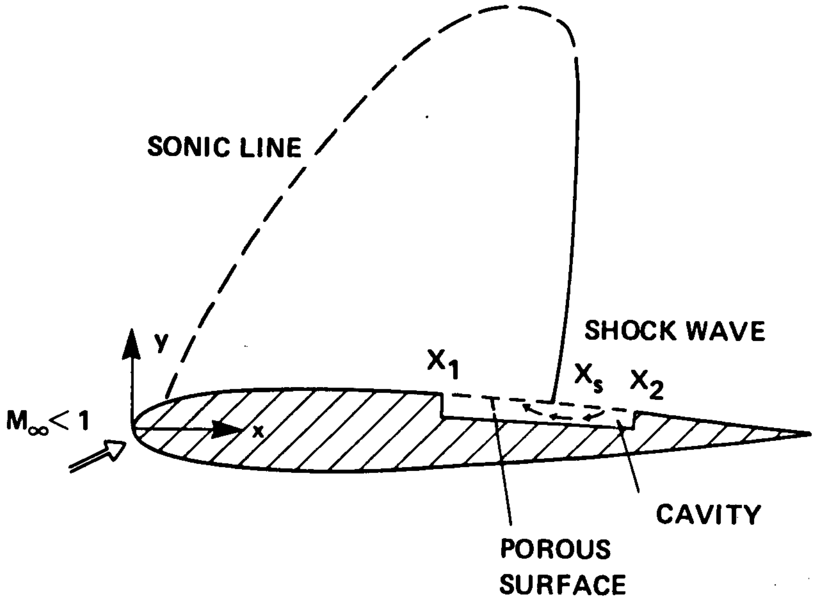

Fig. 14 Porous airfoil in transonic flow.

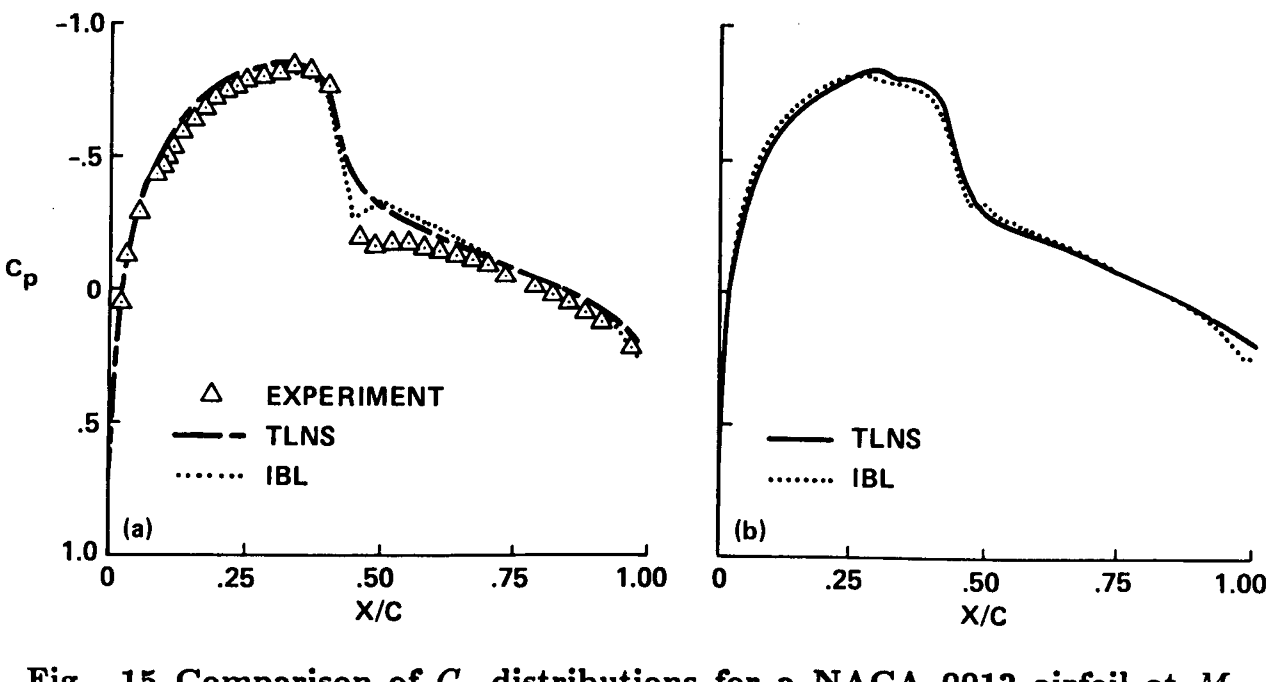

Fig. 15 Comparison of C_p distributions for a NACA 0012 airfoil at $M_\infty = 0.8, \alpha = 0.0°$, and $Re = 4.09 \times 10^6$: a) computed and experimental results for a solid airfoil; b) computed results for a porous airfoil (porous over region of $0.3 \leq x \leq 0.5$). (TLNS-Thin Layer Navier-Stokes, IBL-Interactive Boundary Layer)

assumption (used in the full-potential formulation) and the boundary-layer assumptions are being violated.

This formulation has also been applied to a tri-element augmentor wing configuration by Flores and Van Dalsem.[58] Figures 16a-b indicate representative results from this work. The flow about this configuration has also been computed using a Navier-Stokes formulation.[59] Overall, the agreement between the Navier-Stokes and viscous-inviscid interaction results and the experimental data is only fair. Furthermore, it is not at all clear that the Navier-Stokes results are superior to the viscous-inviscid results. However, the viscous-inviscid interaction computation required at least an order of magnitude less computer time. From this work one might conclude that the viscous-inviscid interaction code gives about the same accuracy for much less cost, and therefore is a better engineering tool. On the other hand, the viscous-inviscid interaction code is more complex because of the logic required to track the six boundary-layers and three wakes.

Overall, it is felt that these results indicate that a "conventional" full-potential/boundary-layer interaction scheme is a practical and useful engineering tool for flows of moderate complexity. However, as either the flow physics or the geometry become more complex, it becomes much more difficult to apply a "conventional" viscous-inviscid interaction formulation. In this case one might consider using a Navier-Stokes formulation, or alternative viscous-inviscid interaction schemes.

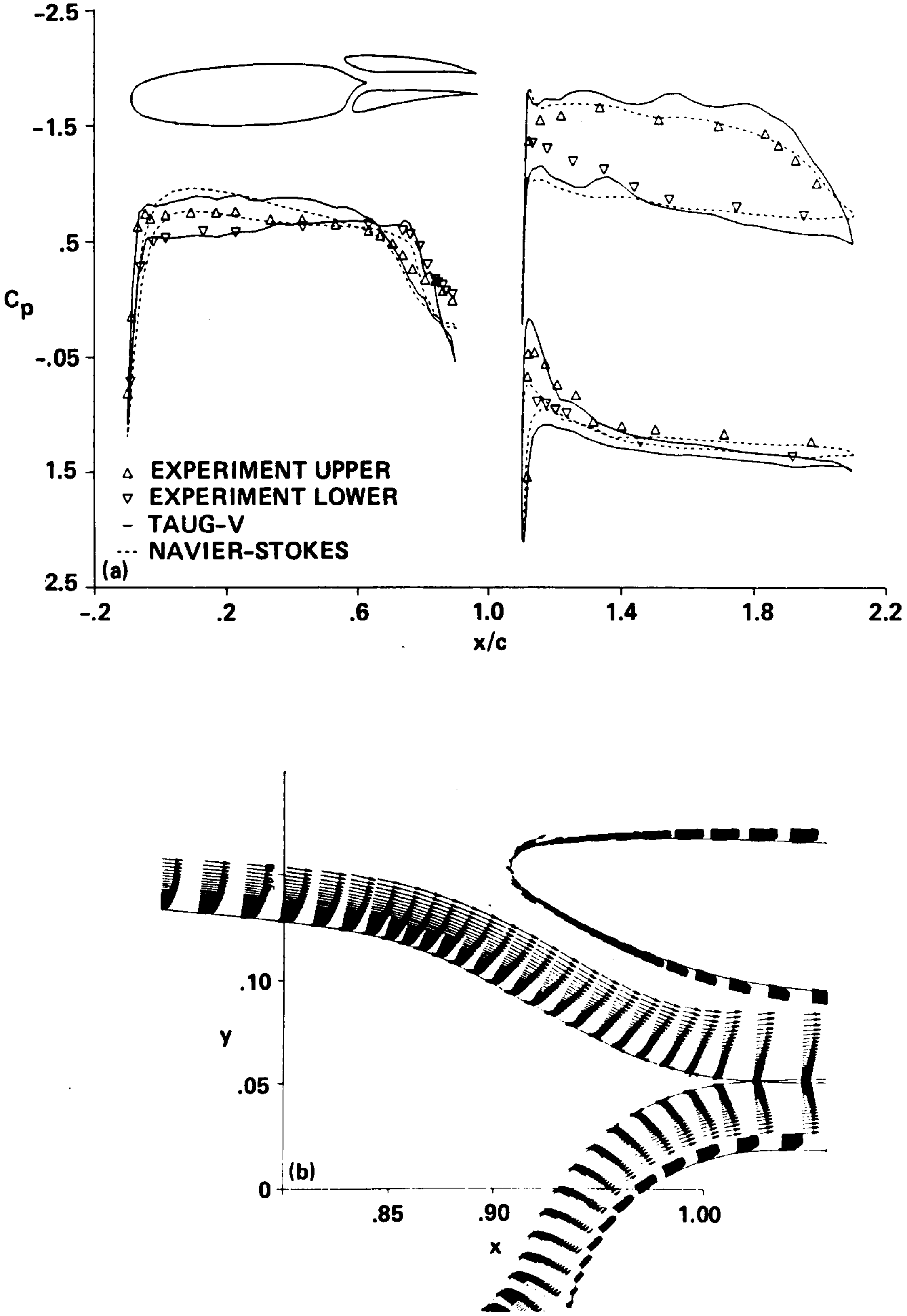

Fig. 16 Example results of application of viscous-inviscid interaction (TAUG-V) to the augmentor wing geometry at $M_\infty = 0.7, \alpha = 1.05°$, and $Re = 12.6 \times 10^6$: a) C_p distributions; b) Viscous velocity vectors in the throat region.

B. Vector-Potential/Boundary-Layer Interaction

An advantage of conventional viscous-to-inviscid interaction using either transpiration or an effective body displacement is that the inviscid flow can be computed using a grid that is relatively coarse compared to what is needed for the viscous grid. Moreover, the inviscid grid need not necessarily be body conforming (e.g., a Cartesian inviscid grid can be used) in the way that a high Reynolds number Navier-Stokes grid must be.

The disadvantage of conventional viscous-to-inviscid interaction is that features such as wakes require special treatment, and matching to a highly rotational inviscid outer flow is not straightforward. Our own experience with the previously described algorithm is that too much logic (such as the wake curvature correction) must be programmed. A thin-layer Navier Stokes code, while costly, is a less complex code which is more readily adapted to new problems.

Instead of imposing the effect of the viscous flow on the inviscid flow through a boundary condition, the viscous flow effect can be embedded into the inviscid flow by means of a forcing function. In particular, the vorticity of the viscous flow can be embedded, through a field forcing term, directly into the "inviscid" algorithm. Such a viscous-inviscid interaction procedure has some advantages over the approach described in the previous section and is briefly discussed in the next section (others are pursuing related ideas; e.g., Halim and Hafez,[42] and Whitfield[43]).

1. Vorticity Transfer Interaction

The Euler equations in nonconservative form can be related to the stream function and other potential-like decompositions of the velocity field and the vorticity. In this form, the Euler equations can be solved very efficiently. For example, the steady inviscid governing equations in three dimensions can be written in terms of the dual-potential velocity decomposition defined by

$$\mathbf{q} = \nabla\phi + \nabla\times\mathbf{B}; \quad \mathbf{B} = \begin{pmatrix} \vartheta \\ \chi \\ \psi \end{pmatrix}$$

or

$$\mathbf{q} = \begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} \phi_x + \psi_y - \chi_z \\ \phi_y + \vartheta_z - \psi_x \\ \phi_z + \chi_x - \vartheta_y \end{pmatrix} \tag{10}$$

where ϕ is the usual potential function and $\mathbf{B}$ is the vector potential. Because four functions are used to define three velocity components, a constraint needs to be imposed on the potential functions to uniquely specify the velocity field. Subject to the consistency constraint

$$\nabla\cdot\mathbf{B} = \vartheta_x + \chi_y + \psi_z = 0 \tag{11}$$

the governing equations take on the symmetric form[60] continuity

$$(\rho\phi_x)_x + (\rho\phi_y)_y + (\rho\phi_z)_z = -[(\psi_y - \chi_z)\rho]_x - [(\vartheta_z - \psi_x)\rho]_y - [(\chi_x - \vartheta_y)\rho]_z \qquad (12a)$$

vorticity

$$\nabla^2 \mathbf{B} = -\vec{\omega} \qquad (12b)$$

Crocco equation

$$\mathbf{q} \times \vec{\omega} = \nabla h_0 - T\nabla s \qquad (12c)$$

entropy convection

$$us_x + vs_y + ws_z = 0 \qquad (12d)$$

stagnation enthalpy convection

$$u(h_0)_x + v(h_0)_y + w(h_0)_z = 0 \cdot \qquad (12e)$$

nonisentropic Bernoulli

$$\frac{\rho}{\rho_r} = \left[\left(\frac{h_0}{h_r}\right) - \frac{1}{2}\left(\frac{q^2}{h_r}\right)\right]^{\frac{1}{\gamma-1}} e^{-\left(\frac{s-s_r}{R}\right)} \qquad (12f)$$

The vorticity appears explicitly in Eq. 12b. Hence, it is straightforward to impose a vorticity field generated by any source (e.g., viscosity) on the velocity field. In place of an effective displacement-thickness boundary condition, viscous flow effects can be embedded into the dual-potential inviscid flow equations as sketched in Fig. 17. The dual-potential equations are solved throughout, while in viscous flow regions, the boundary-layer equations are also solved to resolve the vorticity, entropy, and stagnation enthalpy fields. Within the viscous layer, the inviscid convection equations for entropy and stagnation enthalpy are discarded, and the boundary-layer vorticity is fed directly into the right-hand side of Eq. (12b). Likewise the entropy needed in the Bernoulli equation (Eq. (12f)) can be obtained from the boundary-layer equations, or the boundary-layer equations can directly supply density. Outside of the viscous layer, Eqs. (12a-c) can be used to account for vorticity in the flow, and although they do not generate vorticity, they can adequately convect vorticity. Like a Navier-Stokes approach, essentially one governing set of equations (the dual-potential form of the Euler equations) can be used throughout the flow field. The boundary-layer equations are also solved on a subset of the computational domain, but only to generate the vorticity, entropy, and stagnation enthalpy gradients. In this way, the inviscid and viscous equations do not interact on a boundary alone, but by embedded forcing functions interact over the entire viscous portion of the computational domain.

As this approach allows vorticity transfer into the outer inviscid flow, it would seem to be a more general viscous-inviscid interaction than an effective displacement thickness boundary condition. However, for this procedure to work the inviscid mesh must be sufficiently refined to retain the concentrated vorticity near the wall. Therefore, if this procedure is to be practical, it must be possible to efficiently solve the global inviscid equations on a grid which is refined in the normal direction in the vicinity of the wall. The Poisson-like equations which arise with the dual-potential velocity decomposition can be efficiently solved on such a refined grid.

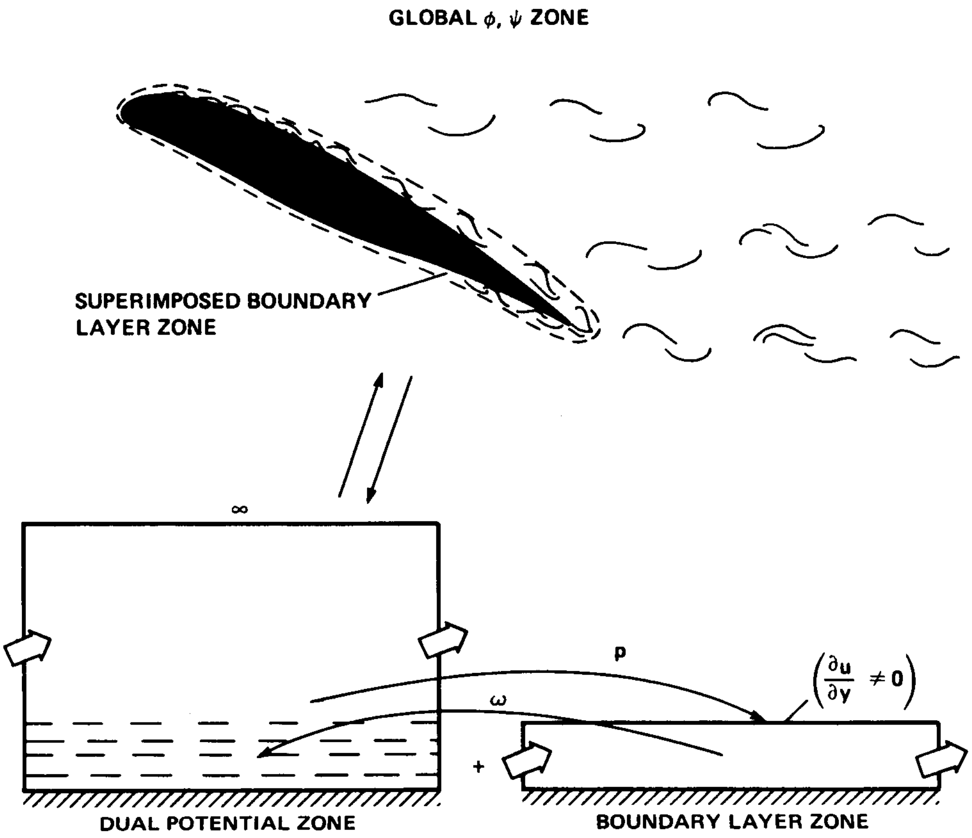
GLOBAL ϕ, ψ ZONE
SUPERIMPOSED BOUNDARY LAYER ZONE
∞
p
ω
$\left(\frac{\partial u}{\partial y} \neq 0\right)$
+
DUAL POTENTIAL ZONE
BOUNDARY LAYER ZONE

Fig. 17 Sketch of computational domains in which the dual-potential equations are solved throughout on a Navier-Stokes-like grid and the boundary-layer equations are also solved near the wall to generate vorticity.

2. Results

Preliminary applications of this vorticity interaction procedure have been reported by Steger and Van Dalsem[41] to a two-dimensional inlet/diffuser flow, and by Rao et al.[61] for three-dimensional incompressible flow over a trough. In both test cases vorticity was confined to the viscous layer and did not convect into the main flow.

The entire two-dimensional inlet/diffuser flow (indicated in Fig. 18), including the flow exterior to the inlet, was solved. On the lower surface of the inlet the grid was refined, and the boundary-layer equations were also solved (Fig. 18). The two-dimensional boundary-layer scheme described previously was used to solve the compressible boundary-layer equations at a low Mach number. Near and in regions of reversed flow the boundary-layer equations were solved in the inverse mode. Computed velocity profiles obtained from the "inviscid" equations are shown in Fig. 19. Figures 20a-c show the profiles in more detail, and also show the velocity profiles computed from the boundary-layer equations using $M_\infty = 0.2$ and $Re = 4.4 \times 10^5$ based on the inlet length. Although the boundary-layer equations are solved on a finer grid, the agreement is excellent except in the region $x = 0.6$ to $x = 0.8$ where small errors are observed. These errors may be partially attributed to interpolation error. (Vorticity was interpolated only as a function of y. The x-variation of the grid was ignored.) A typical separated profile shown in Fig. 21 demonstrates that if the boundary-layer vorticity is resolved, the inviscid equations can give good results even when too coarse an inviscid grid is used. The no-slip boundary condition was not imposed on the Poisson equation, and the departure from no slip at the wall is an indication of any error. In this case the maximum slip velocity reached 2% of the free stream at the first profile shown in Fig. 20c.

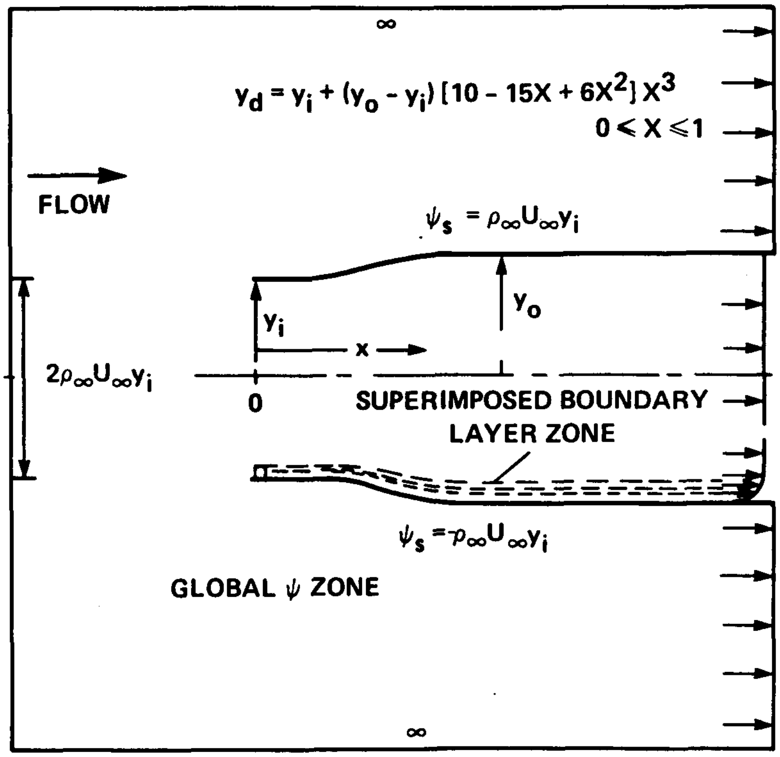
∞
$y_d = y_i + (y_o - y_i)[10 - 15X + 6X^2]X^3$
$0 < X \leqslant 1$
FLOW
$\psi_s = \rho_\infty U_\infty y_i$
y_o
y_i
x
$2\rho_\infty U_\infty y_i$
0
SUPERIMPOSED BOUNDARY
LAYER ZONE
$\psi_s = -\rho_\infty U_\infty y_i$
GLOBAL ψ ZONE
∞

Fig. 18 Sketch of zonal domains in which the stream-function equations are solved everywhere and the boundary-layer equations are superimposed only on the inner lower surface of the inlet/diffuser wall. (Computations performed with $y_i = 1$ and $y_o = 1.07$.)

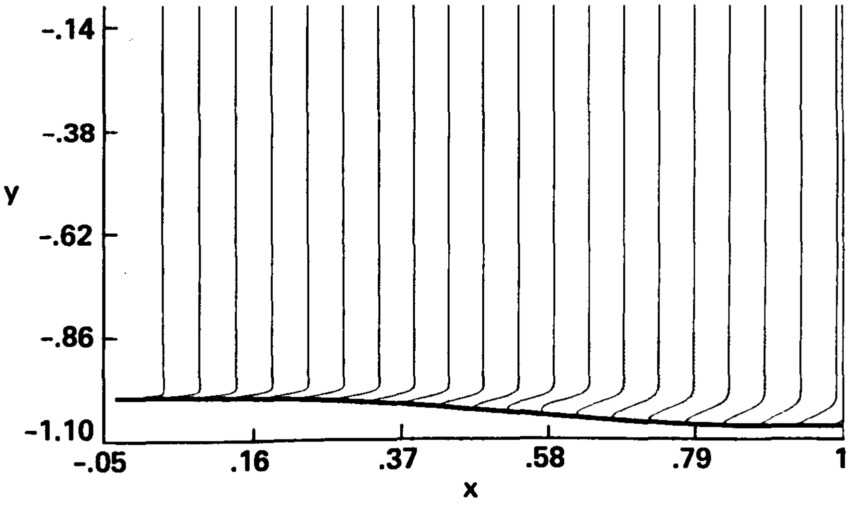

Fig. 19 Profiles on the lower wall of the inlet/diffuser which are computed from the inviscid stream function with vorticity prescribed from the boundary-layer solution.

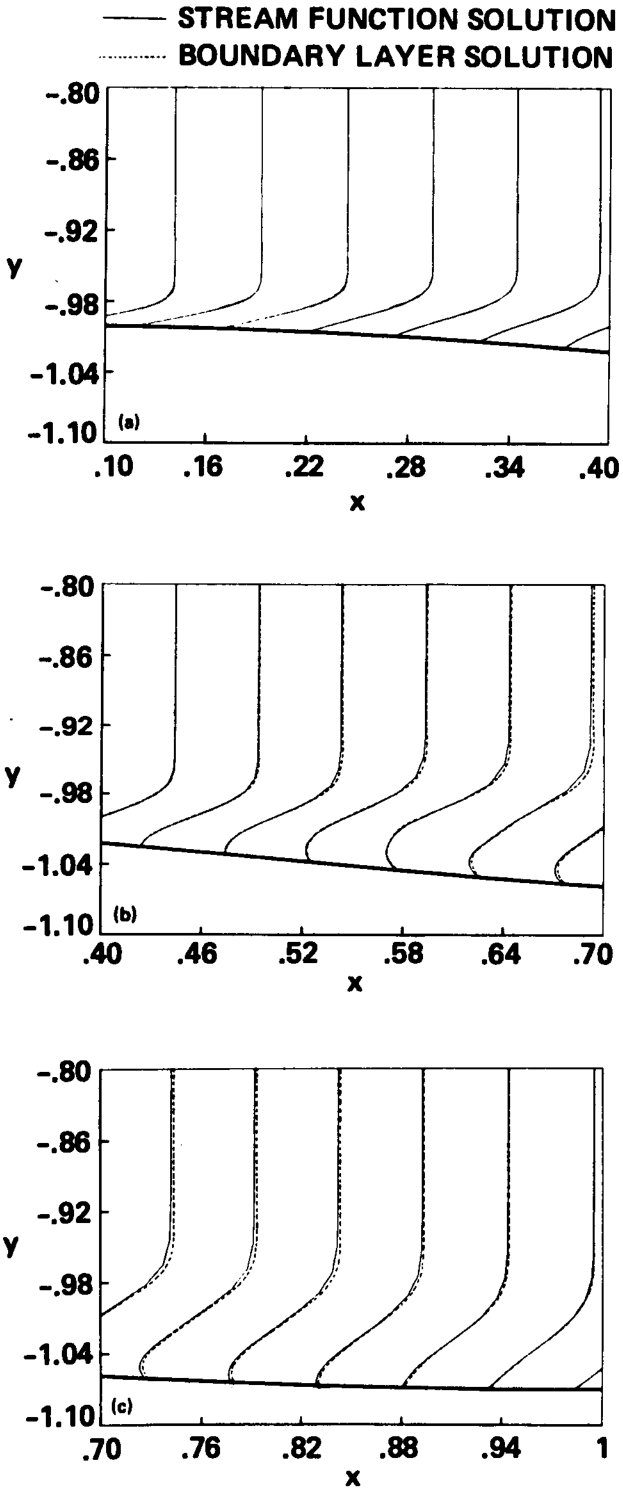

Fig. 20 Comparison of inviscid stream function and boundary-layer computed profiles.

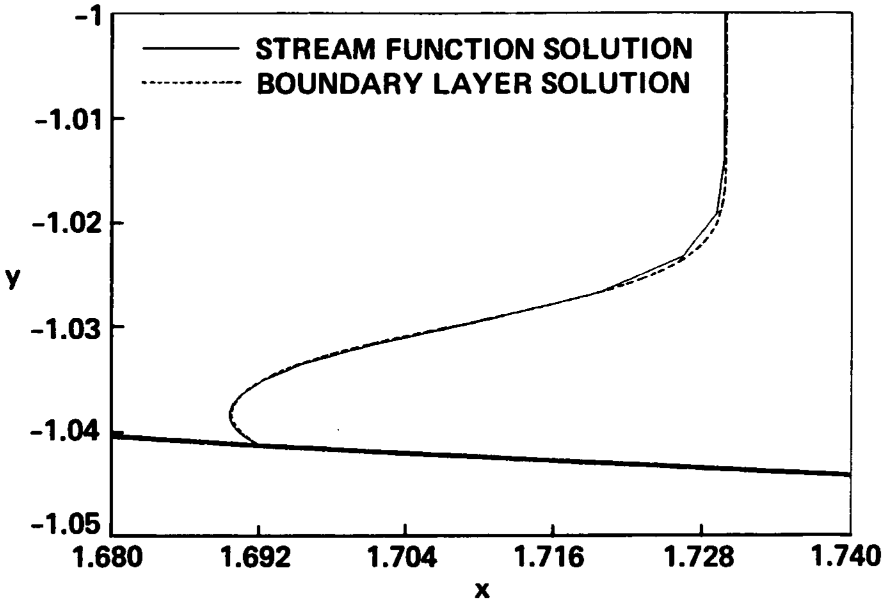

Fig. 21 Good agreement for a typical separated profile in which the boundary-layer resolved vorticity is embedded into the inviscid flow equations with inadequate grid resolution.

Using this scheme, three-dimensional calculations have also been carried out by Rao et al.[61] For the three-dimensional trough defined by

$$z(x,y) = -0.03\,\mathrm{sech}(4x - 10)\,\mathrm{sech}(4y - 6) \tag{13}$$

solutions are presented in Figs. 22-23 which were obtained at an upstream reference Reynolds number of 8000. Figure 22 is a carpet plot of the displacement thickness. The vorticity components were evaluated from the three-dimensional boundary-layer algorithm just described. Comparison with other numerical results as obtained from Davis et al.[62] are presented in Fig. 23. Figure 23 is a plot of the variation of the x-component of shear stress in the streamwise direction at the plane of symmetry. The agreement is relatively good. Edwards[63] obtained a solution for this geometry using an interacting boundary-layer algorithm coupled with an inviscid small-disturbance analysis, while Davis et al.[62] obtained a viscous flow solution using a vorticity/stream-function type formulation for the Navier-Stokes equations. The present method worked well for attached flows, however some difficulty in obtaining converged solutions in separated flow regions was reported.[61]

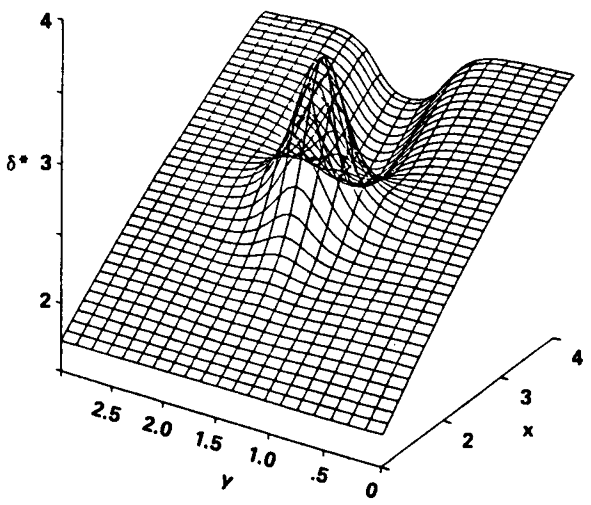

Fig. 22 Carpet plot of the predicted displacment thickness distribution for the three-dimensional trough geometry.

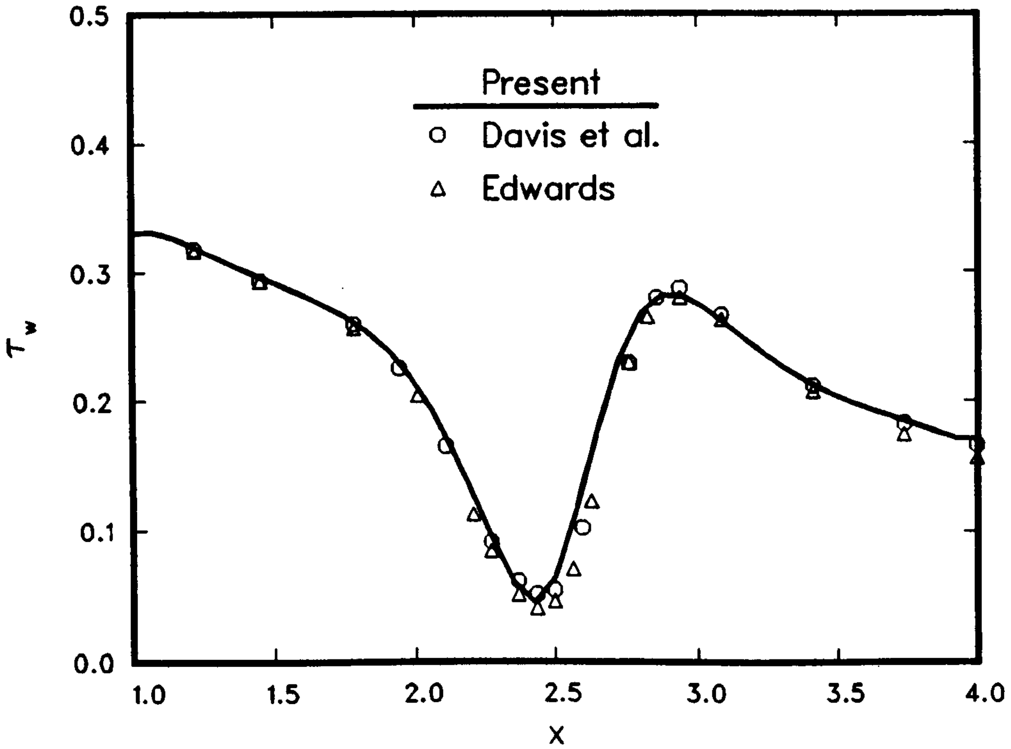

Fig. 23 Comparison of the x-component of the shear stress in the streamwise direction along the trough centerline.

C. Fortified Navier-Stokes Scheme (FNS)

As an alternative to the more familiar "zonal" concept of solving the various flow zones on separate grids and patching the zones together, the Navier-Stokes equations can be applied throughout and the solution of simplified or subset equations can be used to force or "fortify" the general or global algorithm (Fig. 24). By adding subset equation forcing terms to the Navier-Stokes equations, the Navier-Stokes and subset equations interact strongly over entire regions rather than just at interface boundaries. Moreover, the subset equations can be applied selectively to only those regions where they are clearly valid. Specifically, a boundary-layer solver (capable of quickly resolving thin shear layers) can be used to force a global Navier-Stokes scheme. However, if during the iteration procedure the boundary-layer approximations become suspect in a certain region, the boundary-layer to Navier-Stokes forcing can be turned off, and the region can be resolved with only the global Navier-Stokes scheme. Because of this flexibility, the generality of the Navier-Stokes equations is retained, while some of the efficiency of the subset algorithms is recovered. A method for applying this idea, the FNS approach, is described in this section.

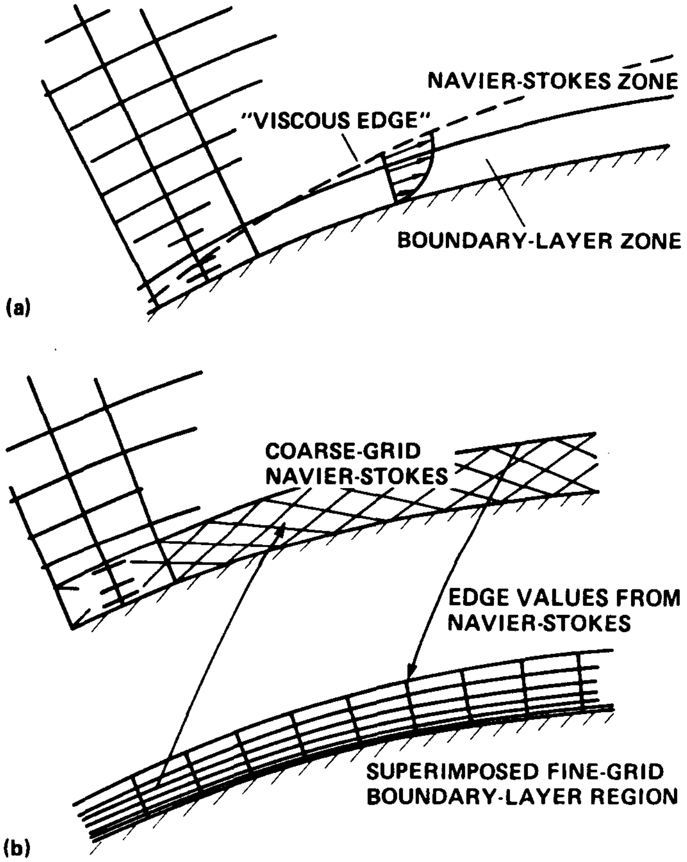

Fig. 24 Boundary-layer forced Navier-Stokes schemes: a) Zonal scheme in which the boundary-layer and Navier-Stokes solutions are matched at a common boundary of a patched grid; b) Spatial-forcing scheme in which the boundary-layer equations are solved on an independent grid, and the boundary-layer solution is forced into the Navier-Stokes algorithm over the entire viscous region.

1. Concept Development

To illustrate the forcing concept which will be used with the Navier-Stokes equations, it is useful to begin with a linear system of partial-differential equations

$$\partial_t \mathbf{Q} + \mathcal{L}\mathbf{Q} = \beta \tag{14}$$

where $\mathbf{Q}$ is a vector, $\mathcal{L}$ is a linear spatial differential operator, and β may contain terms from linearizing a nonlinear system of partial-differential equations. With finite-difference approximation the original nonlinear system of partial-differential equations can be replaced by a large system of linear ordinary differential equations

$$\partial_t \mathbf{Q} + \mathbf{L}\mathbf{Q} = \mathbf{b} \tag{15}$$

where $\mathbf{Q}$ is now the vector of the variables at each grid point, $\mathbf{L}$ is a matrix, and $\mathbf{b}$ contains β and boundary conditions. In general, Eq. (15) is then solved using any one of a variety of time or relaxation schemes. However, in the case of the Navier-Stokes equations, Eq. (15) can be a very large and stiff system of equations.

Assuming that a solution $\mathbf{Q}_f$ is known in some region (for example, from the solution of a simpler but still valid set of equations) then this solution can be used to reduce the computational domain of the original problem. Alternatively, $\mathbf{Q}_f$ can be embedded into the system of equations by means of a forcing function such as

$$\partial_t \mathbf{Q} + \mathbf{L}\mathbf{Q} = \mathbf{b} + \chi(\mathbf{Q}_\mathbf{f} - \mathbf{Q}) \tag{16}$$

where the parameter χ is set to a large positive value if $\mathbf{Q}_f$ is known and is set to zero everywhere else.

The steady-state solution of Eq. (15) is

$$\mathbf{Q} = \mathbf{L}^{-1}\mathbf{b}$$

while the steady-state solution of Eq. (16) is

$$\mathbf{Q} = (\mathbf{L} + \chi \mathbf{I})^{-1}(\mathbf{b} + \chi \mathbf{Q}_f)$$

For sufficiently large values of χ, $\mathbf{Q} \rightarrow \mathbf{Q}_f$, while for $\chi = 0$ everywhere $\mathbf{Q} = \mathbf{L}^{-1}\mathbf{b}$, and some blend of these solutions is obtained for moderate or variable values of χ.

For a constant value of χ the exact solution of Eq. (16) is

$$\mathbf{Q} = ce^{-(\mathbf{L}+\chi\mathbf{I})t} + (\mathbf{L} + \chi \mathbf{I})^{-1}(\mathbf{b} + \chi \mathbf{Q}_f)$$

so that a time-accurate path to a steady state will be damped as $e^{-(\mathbf{L}+\chi\mathbf{I})t}$. Hence as χ is increased, the steady state is reached faster.

For an iterative scheme in which at least $\chi\mathbf{Q}$ is treated implicitly, such as

$$(1 + h\chi)\mathbf{Q}^{n+1} = (\mathbf{I} - h\mathbf{L})\mathbf{Q}^n + h(\mathbf{b} + \chi \mathbf{Q}_f) \tag{17}$$

the iterative solution is

$$\mathbf{Q}^{n+1} = \left(\frac{\mathrm{I} - h\mathrm{L}}{1 + h\chi}\right)^n \mathbf{Q}^1 + h \sum_{m=0}^{n-1} \frac{(\mathrm{I} - h\mathrm{L})^m}{(1 + h\chi)^{m+1}} (\mathbf{b} + \chi \mathbf{Q}_f) \tag{18}$$

$$\text{and if} \quad \left| \left(\frac{\mathrm{I} - h\mathrm{L}}{1 + h\chi} \right) \right| < 1 \text{ then}$$

$$(\mathrm{L} + \chi \mathrm{I})^{-1} \doteq h \sum_{m=0}^{n-1} \frac{(\mathrm{I} - h\mathrm{L})^m}{(1 + h\chi)^{m+1}}$$

This simple analysis indicates that both the transient solution decays more quickly and the series approximation to the steady-state solution converges more quickly as χ is increased. Hence, if a solution $\mathbf{Q}_f$ is known, it can be used to quickly drive an iteration scheme to a steady state or damp errors in transient solutions. Finally, although χ is denoted as a parameter, it could be any positive definite operator that enhances convergence. The reader is referred to Ref. 64 for further details.

2. An Application

In a typical high-Reynolds number Navier-Stokes simulation the fine-grid resolution is generally provided in a thin zone near the body surface, and the outer flow is effectively resolved as rotational inviscid flow (unless the turbulent coefficients are quite large, in which case extensive modeling is required). Because of the fine-grid resolution required near the body, a given algorithm often operates much less efficiently on the Navier-Stokes equations than it does on the Euler equations, even though the work per step may be similar and the viscous terms may enhance stability. However, on this same refined viscous grid, the boundary-layer equations can be efficiently solved. As a result one can speculate that by using the FNS approach and a boundary-layer algorithm to supply $\widehat{\mathbf{Q}}_f$ in the shear layers, it may be possible to significantly improve the productivity of a Navier-Stokes algorithm.

In this application, an implicit approximately factored algorithm with central differencing in the η and ζ directions and upwinding in the ξ direction is used to solve the three-dimensional thin-layer Navier-Stokes equations. The basic two-factor solution algorithm is written as[65]

$$\begin{aligned}
&\left[I + h\delta_\xi^b(\widehat{A}^+)^n + h\delta_\zeta \widehat{C}^n - hRe^{-1}\bar{\delta}_\zeta J^{-1}\widehat{M}^n J - D_i|_\zeta\right] \\
&\times \left[I + h\delta_\xi^f(\widehat{A}^-)^n + h\delta_\eta \widehat{B}^n - D_i|_\eta\right] \Delta\widehat{\mathbf{Q}}^n = \\
&\quad - \Delta t\{\delta_\xi^b[(\widehat{\mathbf{F}}^+)^n - \widehat{\mathbf{F}}_\infty^+] + \delta_\xi^f[(\widehat{\mathbf{F}}^-)^n - \widehat{\mathbf{F}}_\infty^-] + \delta_\eta(\widehat{\mathbf{G}}^n - \widehat{\mathbf{G}}_\infty) + \delta_\zeta(\widehat{\mathbf{H}}^n - \widehat{\mathbf{H}}_\infty) \\
&-Re^{-1}\bar{\delta}_\zeta(\widehat{\mathbf{S}}^n - \widehat{\mathbf{S}}_\infty)\} - (D_e|_\eta + D_e|_\zeta)(\widehat{\mathbf{Q}}^n - \widehat{\mathbf{Q}}_\infty)
\end{aligned} \tag{19}$$

Here $h = \Delta t$ (first order in time), or $h = \frac{\Delta t}{2}$, (second order in time) and a free stream base solution is subtracted out to improved accuracy in the far field. The operators δ_ξ^b and δ_ξ^f are backward and forward three-point difference operators. The flux $\widehat{\mathbf{F}}$ has been eigensplit and the matrices $\widehat{A}, \widehat{B}, \widehat{C}$, and $\widehat{M}$ result from local linearization of the fluxes about the previous time level. Because central-space-difference operators are used in η and ζ, implicit D_i and explicit D_e numerical dissipation terms are included in Eq. (19).

There are a number of possible ways to add the forcing term to Eq. (19). The most straightforward way would be to add the term $h\chi$ to either the first or second left-hand side factor, and the term $h\chi(\widehat{Q}_f - \widehat{Q}^n)$ to the right-hand side. However, this would lead to a factorization error on the left-hand side of the order of χh^2, which can become large as $\chi \gg 1$. A factorization which results in an error of the order of $\chi^{-1}h^2$ is given by

$$\begin{aligned}
&\left[I(1+h\chi) + h\delta_\xi^b(\widehat{A}^+)^n + h\delta_\zeta\widehat{C}^n - hRe^{-1}\bar{\delta}_\zeta J^{-1}\widehat{M}^n J - D_i|_\zeta\right] \\
&\times [I(1+h\chi)]^{-1} \times \left[I(1+h\chi) + h\delta_\xi^f(\widehat{A}^-)^n + h\delta_\eta\widehat{B}^n - D_i|_\eta\right]\Delta\widehat{\mathbf{Q}}^n = \\
&-\Delta t\{\delta_\xi^b[(\widehat{\mathbf{F}}^+)^n - \widehat{\mathbf{F}}_\infty^+] + \delta_\xi^f[(\widehat{\mathbf{F}}^-)^n - \widehat{\mathbf{F}}_\infty^-] + \delta_\eta(\widehat{\mathbf{G}}^n - \widehat{\mathbf{G}}_\infty) + \delta_\zeta(\widehat{\mathbf{H}}^n - \widehat{\mathbf{H}}_\infty) \\
&-Re^{-1}\bar{\delta}_\zeta(\widehat{\mathbf{S}}^n - \widehat{\mathbf{S}}_\infty)\} - (D_e|_\eta + D_e|_\zeta)(\widehat{\mathbf{Q}}^n - \widehat{\mathbf{Q}}_\infty) + h\chi(\widehat{\mathbf{Q}}_f - \widehat{\mathbf{Q}}^n)
\end{aligned} \tag{20}$$

With this implementation, χ can become very large without concern for large factorization errors. Another advantage of this form is that as χ becomes large it contributes to the diagonal dominance of both left-hand factors.

3. Results

The FNS method was tested on a geometry which roughly simulates the separated flow on a swept infinite wing, as studied by Van den Berg and Elsenaar.[66] The flow was computed with both the standard thin-layer Navier-Stokes[65] and the FNS algorithms. The flow was first computed using the standard Navier-Stokes algorithm alone on both a fine mesh (29 points in the flow direction, 50 points normal to the wall, 5 points in the span direction) and a coarse mesh which has only 20 points in the critical normal direction. The same minimum normal spacing at the wall was used in both the coarse- and fine-mesh computations. This flow was also computed with the FNS approach using the coarse Navier-Stokes mesh (20 points in the normal direction) and, near the wall, a superimposed fine boundary-layer mesh (50 points in the normal direction). In all the FNS computations presented here, χ was proportional to the vorticity; hence, it is automatically large within the boundary-layer and rapidly drops to zero near the edge of the boundary layer. The drag history in Fig. 25 shows that the coarse-grid standard Navier-Stokes computation does not predict the drag accurately, and that the FNS method obtains essentially the same drag level in 50 iterations that the fine-grid standard Navier-Stokes computation reached in 400 iterations. The computed near-surface particle traces are shown in Figs. 26a-c. Both the FNS method and the fine-grid standard Navier-Stokes computations predict a constant chord-line separation line, whereas the coarse-grid standard Navier-Stokes computation does not quite capture this qualitative feature. Thus, a coarse-mesh standard Navier-Stokes computation should not be relied on to predict even the basic flow features. A detailed comparison of the FNS and fine-grid standard Navier-Stokes computations near the separation line shows that the FNS computation has predicted slightly larger turning angles. It is thought that this slight discrepancy is due at least in part to differences in the amount of numerical dissipation in the Navier-Stokes and boundary-layer algorithms.

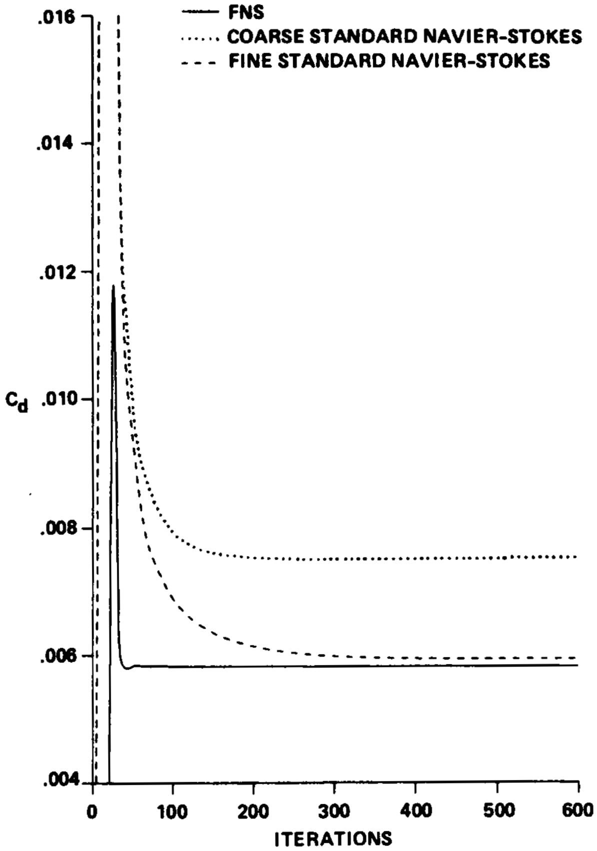

Fig. 25 Drag coefficient history versus iterations for the turbulent flow over a 7.75% sine wave bump with a 35° leading-edge sweep, $Re = 5 \times 10^5$, and $M_\infty = 0.5$.

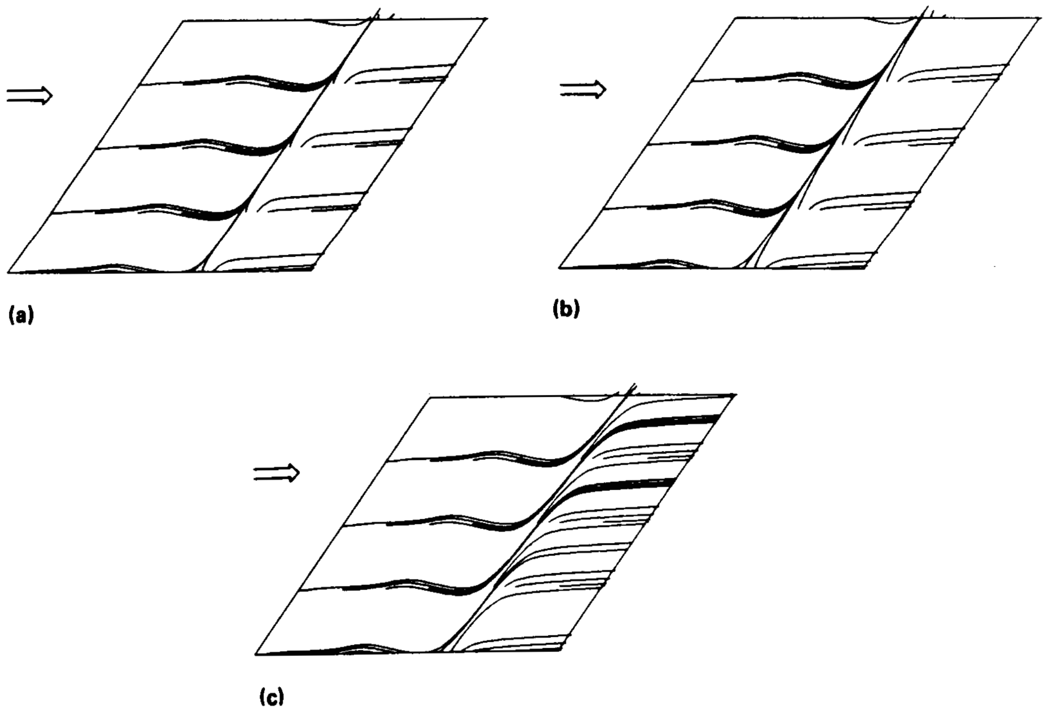

Fig. 26 Computed near-surface particle traces for the flow of Fig. 25: a) Fortified Navier-Stokes; b) standard Navier-Stokes (fine mesh); c) standard Navier-Stokes (coarse mesh).

The swept infinite wing geometry described above was modified to yield a truly three-dimensional flow by reducing the aspect ratio of the wing to one. Also, to resolve spanwise changes, the grid dimension in this direction was increased from 5 to 15. The resulting particle traces (Figs. 27a-d) show the same trends as described for the infinite span example. Also shown in Fig. 27d is the result obtained when the viscous terms and no-slip boundary conditions are removed from the global numerical algorithm (thus making it an Euler formulation). In this case, the entire influence of viscosity must be carried by the boundary-layer algorithm, which is not a difficulty for this case. The drag history versus CRAY-XMP CPU time for these computations is presented in Fig. 28. As before, the coarse-grid standard Navier-Stokes computation is not accurate, and the fine-grid standard Navier-Stokes computation is expensive, whereas the FNS (and Euler) computations yield the same drag level as does the fine-grid standard Navier-Stokes computation, but for one-twentieth of the cost.

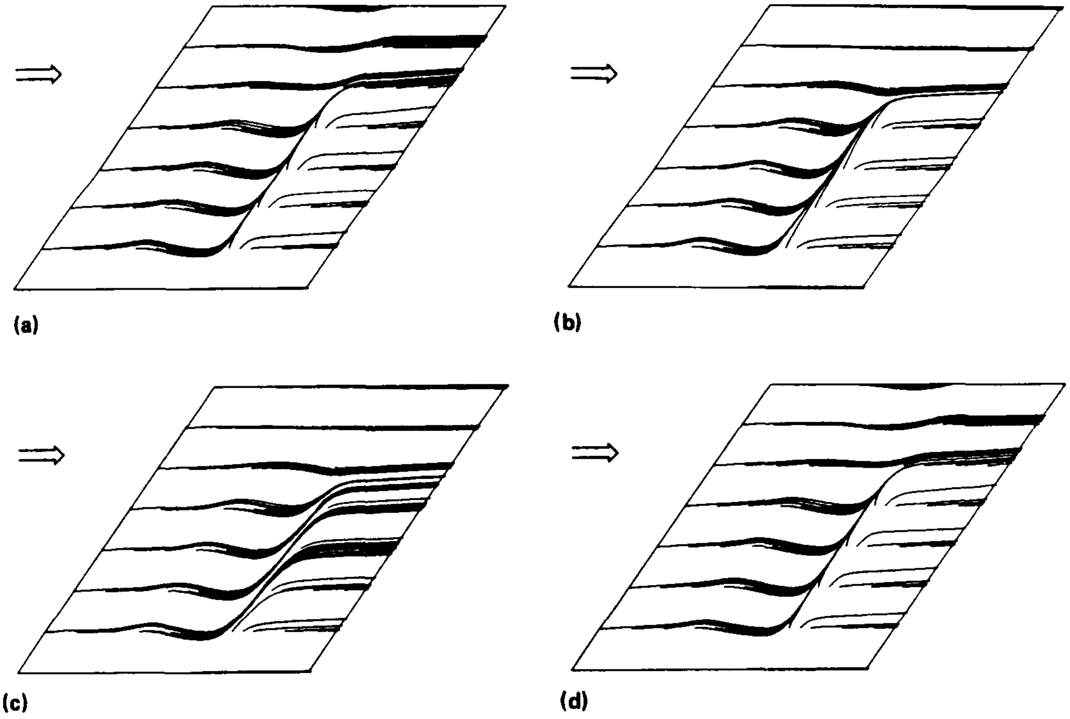

Fig. 27 Computed near-surface particle traces for the turbulent flow over a 7.75% sine wave bump with a 35° leading-edge sweep, $Re = 5 \times 10^5$, $M_\infty = 0.5$, and AR=1: a) Fortified Navier-Stokes; b) standard Navier-Stokes (fine mesh); c) standard Navier-Stokes (coarse mesh); d) Fortified Euler.

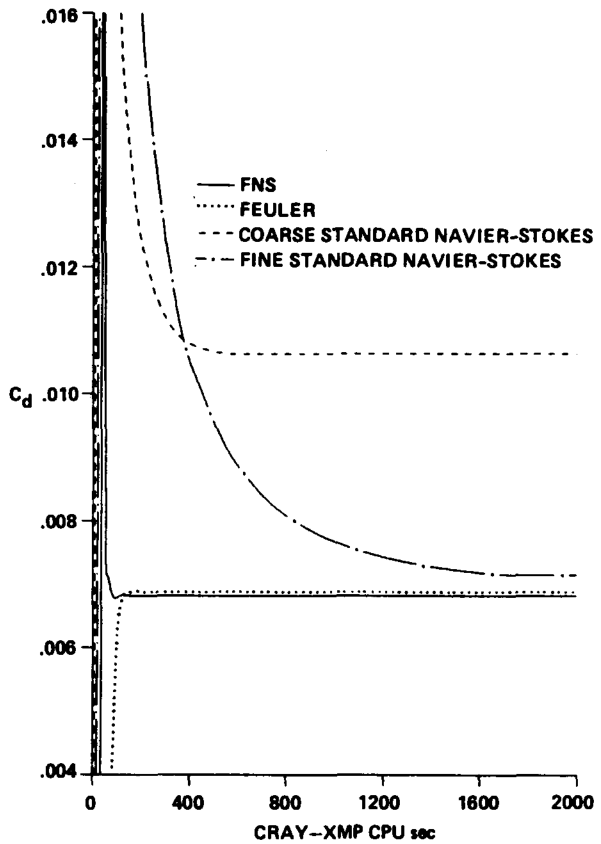

Fig. 28 Drag coefficient history versus CRAY-XMP processor time for flow of Fig. 27.

CONCLUSIONS

A variety of viscous-inviscid interaction schemes have been presented. A "classical" full-potential/boundary-layer interaction has been applied to the transonic separated flow about solid and porous airfoils, and a tri-element augmentor wing. This scheme is representative of the majority of viscous-inviscid interaction codes presently in use. These codes have proven very useful for moderately complex, mildly separated, two-dimensional flows (e.g., the augmentor wing application), and fairly simple three-dimensional flows. However, it is the authors' opinions that it will be difficult to extend the effective displacement thickness class of interaction schemes much beyond present capabilities. The normal pressure gradient and free shear-layer tracking logic needed by these schemes requires a great deal of attention for each new problem and is quite tedious for truly complex flows.

The vorticity interaction scheme presented here is one of a number recently developed specifically to overcome the deficiencies of the effective displacement thickness interaction schemes. In this case, the displacement thickness concept is not used. The outer "inviscid" algorithm is tightly coupled to the viscous flow over the entire viscous flow domain. If necessary, the inviscid flow can convect vorticity downstream (i.e., as in a wake). To date, this approach has been applied to separated, but fairly simple, flows. However, it is straightforward to include the effect of normal pressure gradient in this scheme, and free shear-layer tracking is automatic. Hence some of the deficiencies of the "classical" effective displacement thickness interaction have been avoided.

The Fortified Navier-Stokes scheme is the next step toward complete generality. In this case, a general global algorithm is "fortified" (i.e., the accuracy and/or convergence rate are improved) in any region where a less complete, but more efficient, numerical formulation can be used. For example, the boundary-layer equations can be used to resolve thin shear layers. It has been shown that this approach can significantly improve the efficiency of the overall algorithm.

ACKNOWLEDGMENTS

The porous airfoil study was performed in conjunction with C. L. Chen, C. Y. Chow, and T. L. Holst, and the augmentor wing work in conjunction with J. Flores.

REFERENCES

[1]Prandtl, L., "Über Flüssigkeitsbewegung bei sehr kleiner Reibung," *Proceedings of the Third International Mathematics Congress*, Heidelberg, 1904.

[2]Hartree, D. R., "On an Equation Occurring in Falkner-Skan's Approximate Treatment of the Equations of the Boundary Layer,"*Proceedings of the Cambridge Philosophical Society*, Vol. 33, 1937, pp. 223-39.

[3]Goldstein, S., "On Laminar Boundary-Layer Flow Near a Position of Separation," *Quarterly Journal of Mechanics and Applied Mathematics*, Vol. 1, 1948, pp. 43-69.

[4]Chapman, D. R., Kuehn, D. M., and Larson, H. K., "Investigation of Separated Flows in Supersonic and Subsonic Streams with Emphasis on the Effects of Transition," NACA TN-3869, March 1957.

[5]Abbott, D. E., Holt, M., and Nielsen, J. N., "Investigation of Hypersonic Flow Separation and Its Effects on Aerodynamic Control Characteristics," Vidya Rept. 81, 1962; also "Studies of Separated Laminar Boundary Layers at Hypersonic Speed With Some Low Reynolds Number Data," AIAA Paper 63-172, 1963.

[6]Nielsen, J. N., Lynes, L. L., and Goodwin, F. K., "Calculation of Laminar Separation With Free Interaction by the Method of Integral Relations: Part I - Two-Dimensional Supersonic Adiabatic Flows," AFFDL-TR-65-107, Air Force Flight Dynamics Lab., Wright-Patterson AFB, Ohio, October 1965.

[7]Nielsen, J. N., Lynes, L. L., and Goodwin, F. K., "Calculation of Laminar Separation With Free Interaction by the Method of Integral Relations: Part II - Two-Dimensional Supersonic Nonadiabatic Flow and Axisymmetric Supersonic Adiabatic and Nonadiabatic Flows," AFFDL-TR-65-107, Air Force Flight Dynamics Lab., Wright-Patterson AFB, Ohio, January 1966.

[8]Lees, L, and Reeves, B. L., "Supersonic Separated and Reattaching Laminar Flows: I. General Theory and Application to Adiabatic Boundary-Layer/Shock-Wave Interactions," *AIAA Journal*, Vol. 2., No. 11, 1964, pp. 1907-1920.

[9]Reeves, B. L., and Lees, L. "Theory of Laminar Near Wake of Blunt Bodies in Hypersonic Flow," *AIAA Journal*, Vol. 3, No. 11, 1965, pp. 2061-2074.

[10]Fujii, K., "An Approximate Numerical Approach to Separated Flows at Transonic Speeds," Institute of Space and Aeronautical Science, University of Tokyo, Report No. 595, March 1981.

[11]Lyrio, A. A., Ferziger, J. H., and Kline, S. J., "An Integral Method for the Computation of Steady and Unsteady Turbulent Boundary-Layer Flows," Department of Mechanical Engineering, Stanford University, Report PD-23, March 1981.

[12]Bardina, J., Kline, S. J., and Ferziger, J. H., "Computation of Straight Diffusers at Low Mach Number Incorporating an Improved Correlation for Turbulent Detachment and Reattachment," Department of Mechanical Engineering, Stanford University, Report PD-22, February 1982.

[13]Veldman, A. E. P., "Calculation of Incompressible Boundary-Layers with Strong Viscous-Inviscid Interaction," AGARD 293, Paper 12, 1981.

[14]Catherall, D., and Mangler, K. W., "The Integration of the Two-Dimensional Laminar Boundary-Layer Equations Past the Point of Vanishing Skin Friction," *Journal of Fluid Mechanics*, Vol. 26, Part 1, 1966, pp. 163-182.

[15]East, L. F., Smith, P. D., and Merryman, P. J., "Prediction of the Development of Separated Turbulent Boundary Layers by the Lag-Entrainment Method," Royal Aircraft Establishment, Technical Report 77046, March 1977.

[16]Le Balleur, J. C., "Strong Matching Method for Computing Transonic Viscous Flows Including Wakes and Separations, Lifting Airfoils," *Research Aerospatiale*, No. 1981-3, English Edision, pp. 21-45.

[17]Klineberg, J. M., and Steger, J. L., "The Numerical Calculation of Laminar Boundary-Layer Separation," NASA TN D-7732, July 1974.

[18]Cebeci, T., "Separated Flows and Their Representation by Boundary-Layer Equations," Mech. Eng. Rept. ONR-CR215-234-2, California State University at Long Beach, 1976.

[19]Carter, J. E., "Inverse Boundary-Layer Theory and Comparison With Experiment," NASA TP 1208, September 1978.

[20]Pletcher, R.H., Kwon, O.K., and Chilukuri, R., "Prediction of Separating Turbulent Boundary Layers Including Regions of Reversed Flow," Iowa State University, Report ISU-ERI-Ames-80112, February 1980.

[21]Arieli, R., and Murphy, J. D., "Pseudo-Direct Solution to the Boundary-Layer Equations for Separated Flow," *AIAA Journal*, Vol. 18, No. 8, 1980, pp. 883-891.

[22]Keller, H. B., and Cebeci, T., "Accurate Numerical Methods for Boundary-Layer Flows, Part 1, Two-Dimensional Laminar Flows," in Lecture Notes in Physics, Vol. 8, *Proceedings of the Second International Conference on Numerical Methods in Fluid Dynamics*, pp. 92-100, Springler-Verlag, New York, 1971.

[23]Cebeci, T. and Bradshaw, P., *Momentum Transfer in Boundary Layers*, McGraw-Hill, New York, 1977.

[24]Van Dalsem, W. R., and Steger, J. L., "Finite-Difference Simulation of Transonic Separated Flow Using a Full-Potential Boundary-Layer Interaction Approach," AIAA Paper 83-1689, Danvers, Massachusetts, 1983.

[25]Van Dalsem, W. R., "Simulation of Separated Transonic Airfoil Flow by Finite-Difference Viscous-Inviscid Interaction," Ph.D. thesis, Stanford University, June 1984.

[26]Reyhner, T. A., and Flugge-Lotz, I., "The Interaction of a Shock Wave with a Laminar Boundary Layer," *International Journal of Non-Linear Mechanics*, Vol. 3, 1968, pp. 173-199.

[27]Van Dalsem, W. R., and Steger, J. L., "The Efficient Simulation of Separated Three-Dimensional Viscous Flows Using the Boundary-Layer Equations," *AIAA Journal*, Vol. 25, 1987, pp. 395-400.

[28]Van Dalsem, W. R., and Steger, J. L., "Using the Boundary-Layer Equations in Three-Dimensional Viscous Flow Simulation," *Proceedings of the 58th Meeting of the AGARD Fluid Dynamics Panel Symposium on Applications of Computational Fluid Dynamics in Aeronautics*, Aix-en-Provence, France, 1986.

[29]Wang, K. C., "Boundary Layer Over a Blunt Body at High Incidence with an Open-Type of Separation," *Proceedings of the Royal Society*, Vol. A340, 1974, pp. 33-55.

[30]Cebeci, T., Khattab, A. K., and Stewartson, K., "On Nose Separation," *Journal of Fluid Mechanics*, 1980, Vol. 97, Part 3, pp. 435-454.

[31]Anderson, D. A., Tannehill, J. C., and Pletcher, R. H., *Computational Fluid Mechanics and Heat Transfer*, McGraw-Hill Book Company, New York, 1984.

[32]Hirschel, E. H., and Kordulla, W., *Shear Flow in Surface-Oriented Coordinates*, Notes on Numerical Fluid Mechanics, Vol. 4, Friedr. Vieweg & Sohn, Braunschweig, Germany, 1981.

[33]Cebeci, T., "Calculation of Compressible Turbulent Boundary Layers with Heat and Mass Transfer," AIAA Paper 70-741, June 1970.

[34]Baldwin, B. S., and Lomax, H., "Thin-Layer Approximation and Algebraic Model for Separated Turbulent Flow," AIAA Paper 78-257, Huntsville, Alabama, 1978.

[35]Meier, H. U., and Kreplin, H. P., "Experimental Investigation of the Boundary-Layer Transition and Separation on a Body of Revolution," *Zeitschrift f. Flugwiss. Weltraumforsh.* Vol. 4, 1981, pp. 65-71.

[36]Meier, H. U., Kreplin, H. P., and Ming, X., "Problems Associated with Artificial Boundary-Layer Transition," AIAA Paper 83-1673, July 1983.

[37]Meier, H. U., Kreplin, H. P., and Vollmers, H., "Development of Boundary Layers and Separation Patterns on a Body of Revolution as Incidence," *Proceedings of the Second Symposium on Numerical and Physical Aspects of Aerodynamic Flows*, California State University, Long Beach, Calif., January 17-20, 1983.

[38]Meier, H. U., and Cebeci, T., "Flow Characteristics of a Body of Revolution at Incidence," *Proceedings of the Third Symposium on Numerical and Physical Aspects of Aerodynamic Flows*, California State University, Long Beach, Calif., January 21-24, 1985.

[39]Lockman, W. K., and Seegmiller, H. L., "An Experimental Investigation of the Subcritical and Supercritical Flow About a Swept Semispan Wing," NASA TM-84367, June 1983.

[40] Holst, T. L., Kaynak, U., Gundy, K. L., Thomas, S. D., Flores, J., and Chaderjian, N., "Numerical Solution of Transonic Wing Flows Using an Euler/Navier-Stokes Zonal Approach," AIAA Paper 85-1640, July 1985.

[41] Steger, J. L., and Van Dalsem, W. R., "Developments in the Simulation of Separated Flows Using Finite-Difference Methods," *Proceedings of the Third Symposium on Numerical and Physical Aspects of Aerodynamic Flows*, California State University, Long Beach, California, 1985.

[42] Halim, A. and Hafez, M. M. "Calculation of Separated Bubbles Using Boundary-Layer Type Equations -Part I and II," AIAA Paper No. 84-1585, Snowmass, Colorado, 1984.

[43] Whitfield, D., "Viscous-Inviscid Interaction Computations Using a Pseudo Navier-Stokes Approach," *Proceedings of the Third Symposium on Numerical and Physical Aspects of Aerodynamic Flows*, California State University, Long Beach, California, 1985.

[44] White, F. M., *Viscous Fluid Flow*, McGraw-Hill, New York, 1974.

[45] Carter, J. E., "A New Boundary-Layer Interaction Technique for Separated Flows," NASA TM 78690, June 1978.

[46] Crocco, L., and Lees, L., "A Mixing Theory for the Interaction Between Dissipative Flows and Nearly Isentropic Streams," *Journal of the Aeronautical Sciences*, Vol. 19, No. 10, 1952, pp. 649-676.

[47] Le Balleur, J. C., "Couplage Visqueux-Non Visquex: Analyse du Probleme Incluant Decollements et Ondes de Choc," *La Recherche Aerospatiale*, No. 1977-6, 1977, pp. 349-358.

[48] Lighthill, M. J., "On Displacement Thickness," *Journal of Fluid Mechanics*, Vol. 4, 1958, pp. 383-392.

[49] Dougherty, F. C., Holst, T. L., Gundy, K. L., and Thomas, S. D., "TAIR - A Transonic Airfoil Analysis Computer Code," NASA TM-81296, 1981.

[50] Lock, R. C., and Firmin, M. C. P., "Survey of Techniques for Estimating Viscous Effects in External Aerodynamics," RAE Technical Report 1900, April 1981.

[51] Melnik, R. E., and Mead, H. R., "A Multi-Grid Method for the Computation of Viscid/Inviscid Interaction on Airfoils," AIAA Paper 83-0234, Reno, Nevada, 1983.

[52] McDevitt, J. B., "Supercritical Flow About a Thick Circular-Arc Airfoil," NASA TM-78549, January 1979.

[53] Levy, L. L., "Experimental and Computational Steady and Unsteady Transonic Flows about a Thick Airfoil," *AIAA Journal*, Vol. 16, 1978, pp. 564-572.

[54] Cook, P. H., McDonald, M. A., and Firmin, M. C. P., "Aerofoil RAE 2822-Pressure Distributions and Boundary-Layer and Wake Measurements," AGARD AR 138, Paper A6, 1979.

[55] Mehta, U., "Reynolds Averaged Navier-Stokes Computations of Transonic Flows Around Airfoils," Presented at the Second Symposium on Aerodynamic Flows, Long Beach, California, January 17-20, 1983.

[56] Chen, C. L., Chow, C. Y., Van Dalsem, W. R., and Holst, T. L., "Computation of Viscous Transonic Flow Over Porous Airfoils," AIAA Paper 87-0359, Reno, Nevada, 1987.

[57]Pulliam, T. H., "Euler and Thin-Layer Navier-Stokes Codes," Notes for Computational Fluid Dynamics User's Workshop, The University of Tennessee Space Institute, Tullahoma, Tenn., March 1984.

[58]Flores, J., and Van Dalsem, W. R., "Transonic Separated Solutions for an Augmentor Wing," *AIAA Journal of Aircraft*, Vol. 23, 1986, pp. 837-842.

[59]Lasinski, T. A., Andrews, A. E., Sorenson, R. L., Chaussee, D. S., Pulliam, T. H., and Kutler, P., "Computation of the Steady Viscous Flow over a Tri-Element 'Augmentor Wing' Airfoil" AIAA Paper 82-0021, January 1982.

[60]Chaderjian, N. M., and Steger, J. L., "The Numerical Simulation of Steady Transonic Rotational Flow Using a Dual-Potential Formulation," AIAA Paper 85-0368, Reno, Nevada, 1985.

[61]Rao, K. V., Pletcher, R. H., Steger, J. L., and Van Dalsem, W. R., "A Three-Dimensional Dual-Potential Procedure with Applications to Wind Tunnel Inlets and Interacting Boundary Layers," Iowa State University, Report ISU-ERI-87296, June 1987.

[62] Davis, R. L., Carter, J. E. and Hafez, M., "Three-Dimensional Viscous Flow Solutions With a Vorticity-Stream Function Formulation," AIAA Paper 87-0601, Reno, Nevada, 1987.

[63] Edwards, D. E., "Analysis of 3-D Separated Flow Using Interacting Boundary-Layer Theory," *IUTAM Symposium on Boundary-Layer Separation,* London, England, 1986.

[64]Van Dalsem, W. R., and Steger, J. L., "The Fortified Navier-Stokes Approach," *Proceedings of the Workshop on Computational Fluid Dynamics*, Institute of Nonlinear Sciences, University of California, Davis, California, 1986.

[65]Steger, J. L., Ying, S. X., and Schiff, L.B., "A Partially Flux-Split Algorithm for Numerical Simulation of Compressible Inviscid and Viscous Flow," *Proceedings of the Workshop on Computational Fluid Dynamics*, Institute of Nonlinear Sciences, University of California, Davis, California, 1986.

[66]Van den Berg, B., and Elsenaar, A., "Measurements in a Three-Dimensional Incompressible Turbulent Boundary Layer in an Adverse Pressure Gradient Under Infinite Swept Wing Conditions," NLR TR-72092, 1972.

NASA
National Aeronautics and
Space Administration

Report Documentation Page

1. Report No.
NASA TM 100015

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
Some Experiences with the Viscous-Inviscid Interaction Approach

5. Report Date
November 1987

6. Performing Organization Code

7. Author(s)
W. R. Van Dalsem, J. L. Steger, and K. V. Rao (Iowa State University, Ames, Iowa)

8. Performing Organization Report No.
A-87308

10. Work Unit No.
505-60

9. Performing Organization Name and Address
Ames Research Center
Moffett Field, CA 94035

11. Contract or Grant No.

13. Type of Report and Period Covered
Technical Memorandum

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, DC 20546-0001

14. Sponsoring Agency Code

15. Supplementary Notes
Point of Contact: Bill Van Dalsem, MS 258-1, Ames Research Center, Moffett Field, CA 94035
(415) 694-4469 or FTS 464-4469

16. Abstract

Methods for simulating compressible viscous flow using the viscous-inviscid interaction approach are described. The formulations presented range from the more familiar full-potential/boundary-layer interaction schemes to a method for coupling Euler/Navier-Stokes and boundary-layer algorithms. An effort is made to describe the advantages and disadvantages of each formulation. Sample results are presented which illustrate the applicability of the methods.

17. Key Words (Suggested by Author(s))
Viscous-inviscid interaction
Boundary-layer algorithm
Fortified Navier-Stokes scheme

18. Distribution Statement
Unclassified – Unlimited

Subject Category – 02

19. Security Classif. (of this report)
Unclassified

20. Security Classif. (of this page)
Unclassified

21. No. of pages
45

22. Price
A03

For sale by the National Technical Information Service, Springfield, Virginia 22161