NASA
Technical
Paper
3050

February 1991

An Explicit Upwind Algorithm for Solving the Parabolized Navier-Stokes Equations

John J. Korte

NASA

NASA Technical Paper 3050

1991

An Explicit Upwind Algorithm for Solving the Parabolized Navier-Stokes Equations

John J. Korte
Langley Research Center
Hampton, Virginia

NASA
National Aeronautics and Space Administration
Office of Management
Scientific and Technical Information Division

Contents

1. Introduction

Numerical solutions of the parabolized Navier-Stokes (PNS) equations have been used to obtain a better understanding of the qualitative and quantitative physical phenomena in steady supersonic and hypersonic viscous flows. The noniterative numerical schemes used to solve the PNS equations have either required adjustment of solution-dependent coefficients for capturing shocks or been inefficient on vector supercomputers. The purpose of this study is to develop and apply a numerical scheme which (1) eliminates the need to make adjustments for shock capturing and (2) efficiently utilizes vector supercomputers for accurately solving the PNS equations for complicated hypersonic flow fields over realistic vehicle configurations.

The subject of this study is a finite-difference, two-stage, explicit, upwind algorithm for the direct (noniterative) integration of the three-dimensional PNS equations in a generalized coordinate system. The advantages of this type of algorithm are that

1. The use of upwind flux approximations with equation sets containing nonlinear hyperbolic conservation laws, such as the pressure and convection terms in the PNS equations, allows shocks to be numerically captured without artificial damping terms which the user must adjust.
2. An explicit integration scheme provides an extremely efficient numerical method on vector or parallel machines for solving systems of equations because the dependent variables can be explicitly updated using concurrent machine operations.

The new algorithm uses upwind approximations of the numerical fluxes for the pressure and convection terms obtained by combining flux-difference splittings (FDS) formed from the solution of an approximate Riemann problem (RP). The approximate RP is solved by modifying the method developed by Roe (1981) for steady supersonic flow of an ideal gas. Roe's method was extended for use with the PNS equations expressed in generalized coordinates and with Vigneron, Rakich, and Tannehill's (1978) approximation of the streamwise pressure gradient. For the three-dimensional PNS equations, both fully upwind and upwind-biased approximations of the pressure and convection flux derivatives are used. The upwind-biased flux approximation is formed by using an upwind flux approximation in the direction normal to a shock wave and a central flux approximation in all other directions. The upwind-biased flux approximation eliminates a loss of accuracy in the numerical solution experienced in three-dimensional flow when upwind flux approximations were used in directions tangential to a shock wave whose tangential velocity was negligible. The upwind fluxes are used in a two-stage integration scheme that reduces to MacCormack's (1969) method when the FDS terms are identically zero.

The PNS equations can be integrated (marched) in space using either an iterative, a noniterative, or a time relaxation scheme. Time relaxation schemes retain the time-dependent terms and use time integration methods to obtain a steady state solution at a streamwise location before advancing in space. A noniterative method is usually preferable over either a time relaxation or an iterative scheme since the solution at a given streamwise station is obtained directly. A noniterative method is important when each integration step's cost is high. This study considers only the noniterative schemes applied to the PNS equations.

The conservation law form of the PNS equations is usually solved using refinements of the finite-difference codes of Schiff and Steger (1979) or Vigneron, Rakich, and Tannehill (1978). Both codes use a noniterative, implicit, approximate-factorization, finite-difference algorithm for integrating the thin-layer form of the PNS equations. These algorithms are based on numerical schemes developed by McDonald and Briley (1975) and Beam and Warming (1978). These numerical schemes use central differences to approximate the spatial derivatives of the fluxes. These implicit schemes also use recursive operations which are generally more difficult to apply efficiently on vector computers than are explicit schemes. Gielda and McRae (1986) took advantage of the high vectorizing efficiency of a modified form of MacCormack's (1969) method to solve the PNS equations on a Cray 1 supercomputer. They achieved total solution times that were competitive with existing implicit algorithms for certain classes of problems. Conventional central difference schemes such as Beam and Warming's and MacCormack's require manual adjustments of artificial damping terms to maintain numerical stability and to eliminate nonphysical oscillations in the numerical solutions around shock waves. Lawrence, Tannehill, and Chaussee (1986, 1987) developed an implicit finite-volume scheme for solving the PNS equations which used upwind differencing of the convection terms in areas of supersonic flow and standard central differencing in subsonic regions. Central differencing was used in the subsonic region because numerical instabilities occurred when upwind differences were used. The disadvantage of this approach is that it is difficult to vectorize because of the difference

switching and the implicit integration. The finite-difference upwind method developed in this study can be used throughout the flow field and does not require any special switching of differences in the subsonic regime.

This report presents a new three-dimensional, noniterative PNS solver which combines the computational speed and second-order marching accuracy of a two-stage explicit integration scheme with the robust features obtained from upwind approximations of the convection terms. The new algorithm for solving the PNS equations has the following unique features:

1. Use of upwind approximations of the convection terms in the subsonic region
2. Application of a two-stage integration scheme with upwind flux-limited approximations of the convection fluxes
3. A cubic equation defining Vigneron's splitting coefficient in terms of the dependent variables
4. Use of different upwind flux approximations in each stage of the integration algorithm

The outline of this study is as follows: in section 2 the PNS equations are derived; in section 3 the integration of hyperbolic conservation laws in multistage explicit schemes is investigated and applied to two- and three-dimensional PNS equations; and in sections 4 and 5 solutions using the new algorithm are obtained for two- and three-dimensional flows.

2. Governing Equations

The design process for aerospace vehicles has been improved by using computational fluid dynamic (CFD) computer codes to better understand the qualitative and quantitative physical phenomena in the flow field. Future aerospace vehicles, such as the National Aero-Space Plane (NASP), will cruise at hypersonic speeds. At these speeds, the shock waves and boundary layers are merged over portions of the vehicle (fig. 2.1) and cannot be solved independently. Ideally, one should solve the full Navier-Stokes (NS) equations. However, the memory and speed requirements to obtain full NS solutions for a complete vehicle are beyond the capabilities of modern supercomputers. The parabolized Navier-Stokes (PNS) equations approximate the full NS equations and are more amenable to efficient numerical solution for steady hypersonic flow problems. This efficiency arises from spatial rather than temporal integration. Spatial integration reduces the problem by one dimension and thus provides a significant savings in computer memory and time for a given case. PNS numerical solutions agree with NS numerical solutions for steady supersonic and hypersonic viscous flow problems that do not have a strong upstream influence from points downstream. In this section, the strong conservation law form of the PNS equations is presented for use with a generalized coordinate system.

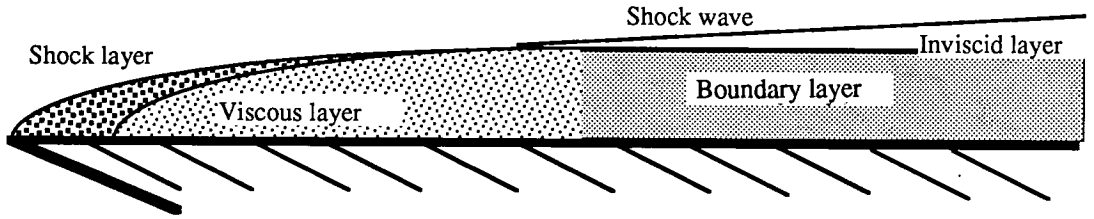


Figure 2.1. Merged shock and viscous layers in a hypersonic flow.

2.1. Navier-Stokes Equations

The flow of a Newtonian fluid can be described by the NS equations, the continuity of mass equation, and the energy equation. These equations expressed in Cartesian conservation law form are

$$\mathbf{U}_t + \mathbf{E}_x + \mathbf{F}_y + \mathbf{G}_z = 0 \tag{2.1}$$

where subscripts x, y, and z indicate partial differentiation with respect to Cartesian coordinates, and subscript t with respect to time. The vectors are defined as

$$\left.\begin{aligned}
\mathbf{U} &= [\rho,\ \rho u,\ \rho v,\ \rho w,\ e_t]^T \\
\mathbf{E} &= \mathbf{E}_i - \mathbf{E}_v \qquad \mathbf{F} = \mathbf{F}_i - \mathbf{F}_v \qquad \mathbf{G} = \mathbf{G}_i - \mathbf{G}_v \\
\mathbf{E}_i &= [\rho u,\ \rho uu + p,\ \rho uv,\ \rho uw,\ (e_t + p)u]^T \\
\mathbf{E}_v &= [0,\ \tau_{xx},\ \tau_{xy},\ \tau_{xz},\ u\tau_{xx} + v\tau_{xy} + w\tau_{xz} - q_x]^T \\
\mathbf{F}_i &= [\rho v,\ \rho vu,\ \rho vv + p,\ \rho vw,\ (e_t + p)v]^T \\
\mathbf{F}_v &= [0,\ \tau_{xy},\ \tau_{yy},\ \tau_{yz},\ u\tau_{xy} + v\tau_{yy} + w\tau_{yz} - q_y]^T \\
\mathbf{G}_i &= [\rho w,\ \rho wu,\ \rho wv,\ \rho ww + p,\ (e_t + p)w]^T \\
\mathbf{G}_v &= [0,\ \tau_{xz},\ \tau_{yz},\ \tau_{zz},\ u\tau_{xz} + v\tau_{yz} + w\tau_{zz} - q_z]^T
\end{aligned}\right\} \tag{2.2}$$

The fluxes are separated into inviscid (subscript i) and viscous (subscript v) components and ρ is the density; u, v, and w are Cartesian velocity components; e_t is total energy; p is pressure;

τ is viscous stress; and q is heat flux. The total energy is defined by

$$e_t = \rho\left(e + \frac{u^2+v^2+w^2}{2}\right) \tag{2.3}$$

where e is the internal energy. The variables have been nondimensionalized using the following relationships:

$$\left.\begin{array}{llll} t = \dfrac{t^*}{L/U_\infty} & x = \dfrac{x^*}{L} & y = \dfrac{y^*}{L} & z = \dfrac{z^*}{L} \\ u = \dfrac{u^*}{U_\infty} & v = \dfrac{v^*}{U_\infty} & w = \dfrac{w^*}{U_\infty} & \rho = \dfrac{\rho^*}{\rho_\infty} \\ p = \dfrac{p^*}{\rho_\infty U_\infty^2} & T = \dfrac{T^*}{T_\infty} & e = \dfrac{e^*}{U_\infty^2} & \mu = \dfrac{\mu^*}{\mu_\infty} \end{array}\right\} \tag{2.4}$$

where T is temperature, μ is viscosity, U is total velocity, L is a characteristic length, * denotes dimensional quantity, and subscript ∞ represents dimensional reference conditions.

2.2. Generalized Transformation

The simulation of a steady supersonic or hypersonic flow field with a dominant flow direction is the problem of interest here. The streamwise direction is defined as being aligned with the ξ-axis. The crossflow plane is defined by the coordinates η and ζ. Equations (2.1) and (2.2) are now changed to a generalized coordinate system using the following transformation:

$$\xi = \xi(x) \qquad \eta = \eta(x,y,z) \qquad \zeta = \zeta(x,y,z) \tag{2.5}$$

The indices i, j, k identify discrete points in the ξ, η, ζ computational coordinate system. The transformation and the formulas for the metrics are given in appendix A.

2.3. Parabolized Navier-Stokes Equations

The PNS equations have evolved from the pioneering work of Rudman and Rubin (1968), who derived and numerically solved a composite set of equations valid in both the inviscid and the viscous region for steady supersonic and hypersonic flow. Researchers have used various forms of composite equation sets to solve steady supersonic viscous flow problems. Common to each of the equation sets is that all viscous stresses and heat fluxes in the streamwise direction are dropped and the subsonic streamwise pressure gradient is modified or dropped. Each numerical scheme obtains a solution at a given streamwise plane, by either time relaxation, iteration, or direct solution, before proceeding forward in the streamwise direction to the next plane.

The PNS equations can be obtained from the transformed NS equations by assuming steady flow and by neglecting the streamwise diffusion terms. The transformed PNS equations in strong conservation law form are

$$\begin{aligned} &\left(\frac{\xi_x \mathbf{E}^*}{J}\right)_\xi + \left(\frac{\eta_x \mathbf{E}}{J} + \frac{\eta_y \mathbf{F}}{J} + \frac{\eta_z \mathbf{G}}{J}\right)_\eta + \left(\frac{\zeta_x \mathbf{E}}{J} + \frac{\zeta_y \mathbf{F}}{J} + \frac{\zeta_z \mathbf{G}}{J}\right)_\zeta \\ &\quad - \left[\mathbf{E}^*\left(\frac{\xi_x}{J}\right)_\xi + \mathbf{E}\left(\frac{\eta_x}{J}\right)_\eta + \mathbf{E}\left(\frac{\zeta_x}{J}\right)_\zeta\right] - \mathbf{F}\left[\left(\frac{\eta_y}{J}\right)_\eta + \left(\frac{\zeta_y}{J}\right)_\zeta\right] - \mathbf{G}\left[\left(\frac{\eta_z}{J}\right)_\eta + \left(\frac{\zeta_z}{J}\right)_\zeta\right] \\ &\quad = -\left(\frac{\xi_x \mathbf{P}}{J}\right)_\xi + \mathbf{P}\left(\frac{\xi_x}{J}\right)_\xi \end{aligned} \tag{2.6}$$

where subscript ξ, η, ζ indicate partial differentiation with respect to the generalized coordinates and J is the Jacobian of the transformation. The streamwise pressure gradient has been split between the two vectors $\mathbf{E}^*$ and $\mathbf{P}$ using the technique developed by Vigneron, Rakich, and Tannehill (1978). The streamwise inviscid flux is split as

$$\mathbf{E}_i = \mathbf{E}^* + \mathbf{P} \tag{2.7}$$

where

$$\mathbf{E}^* = [\rho u,\ \rho uu + \omega p,\ \rho uv,\ \rho uw,\ (e_t + p)u]^T$$
$$\mathbf{P} = [0,\ (1-\omega)p,\ 0,\ 0,\ 0]^T$$

The value of Vigneron's coefficient ω varies from 0 to 1. The determination of ω is covered in a later section.

The shear stress and heat flux terms after the transformation and the parabolizing assumptions become

$$\left.\begin{aligned}
\tau_{xx} &= \frac{2\mu}{3\mathrm{Re}_\infty}\left[2(\eta_x u_\eta + \zeta_x u_\zeta) - (\eta_y v_\eta + \zeta_y v_\zeta) - (\eta_z w_\eta + \zeta_z w_\zeta)\right] \\
\tau_{xy} &= \frac{\mu}{\mathrm{Re}_\infty}\left(\eta_y u_\eta + \zeta_y u_\zeta + \eta_x v_\eta + \zeta_x v_\zeta\right) \\
q_x &= \frac{-\mu}{(\gamma-1)M_\infty^2 \mathrm{Re}_\infty \mathrm{Pr}}\left(\eta_x T_\eta + \zeta_x T_\zeta\right) \\
\tau_{yy} &= \frac{2\mu}{3\mathrm{Re}_\infty}\left[2(\eta_y v_\eta + \zeta_y v_\zeta) - (\eta_x u_\eta + \zeta_x u_\zeta) - (\eta_z w_\eta + \zeta_z w_\zeta)\right] \\
\tau_{xz} &= \frac{\mu}{\mathrm{Re}_\infty}\left(\eta_z u_\eta + \zeta_z u_\zeta + \eta_x w_\eta + \zeta_x w_\zeta\right) \\
q_y &= \frac{-\mu}{(\gamma-1)M_\infty^2 \mathrm{Re}_\infty \mathrm{Pr}}\left(\eta_y T_\eta + \zeta_y T_\zeta\right) \\
\tau_{zz} &= \frac{2\mu}{3\mathrm{Re}_\infty}\left[2(\eta_z w_\eta + \zeta_z w_\zeta) - (\eta_x u_\eta + \zeta_x u_\zeta) - (\eta_y v_\eta + \zeta_y v_\zeta)\right] \\
\tau_{yz} &= \frac{\mu}{\mathrm{Re}_\infty}\left(\eta_y w_\eta + \zeta_y w_\zeta + \eta_z v_\eta + \zeta_z v_\zeta\right) \\
q_z &= \frac{-\mu}{(\gamma-1)M_\infty^2 \mathrm{Re}_\infty \mathrm{Pr}}\left(\eta_z T_\eta + \zeta_z T_\zeta\right)
\end{aligned}\right\} \tag{2.8}$$

The Reynolds, Prandtl, and Mach numbers are denoted by Re, Pr, and M. The ratio of specific heats is denoted by γ and the viscosity is calculated using Sutherland's equation:

$$\mu = T^{3/2}\left(\frac{1 + T_{\mathrm{ref}}}{T + T_{\mathrm{ref}}}\right) \tag{2.9}$$

where

$$T_{\mathrm{ref}} = \frac{110.4\ \mathrm{K}}{T_\infty}$$

The perfect gas relationships are used to completely define the system of equations:

$$T = \frac{\gamma M_\infty^2 p}{\rho} \tag{2.10}$$

Note that the equations presented here differ from those used by others (Chaussee et al. 1981) in that the stress terms are retained in the flux vector $\mathbf{E}$ when the outer derivative is other than ξ.

2.4. Geometric Conservation Law

The second and third lines of equation (2.6) contain the geometric conservation law (GCL) (Anderson, Tannehill, and Pletcher 1984) and metrics associated with returning the governing equations to strong conservation law form. The sum of the metrics in the GCL equals zero when they are analytically evaluated for the full NS equations. In a numerical scheme, the GCL terms may or may not be zero depending on how the full NS equations are approximated and how the equations and the metrics are differenced. Gielda and McRae (1986) have shown that for MacCormack's (1969) method, the GCL terms are not zero for the PNS equation set (eq. (2.6)) for any combination of possible differencing of the metrics. This fact, which is true for all PNS solvers, requires that the GCL terms be evaluated numerically as part of the integration scheme to cancel nonphysical source terms.

2.5. Treatment of the Streamwise Pressure Gradient

In equation (2.6), the streamwise pressure gradient is split between the left and the right side of the PNS equation using Vigneron's coefficient ω. If one is interested in using a space marching method for integrating the PNS equation set, then the inviscid eigenvalues have to be real and the viscous eigenvalues have to be nonnegative and real. These conditions are true for supersonic flow. A linear stability analysis by Vigneron, Rakich, and Tannehill (1978) (and later extended by Davis, Barnett, and Rakich 1986) shows that the inviscid eigenvalues are real and the viscous eigenvalues are real and nonnegative for subsonic flow if a fraction of the streamwise pressure gradient is retained. This fraction is obtained by defining ω as

$$\overline{\omega} = \begin{cases} 1 & (M_\xi \geq 1) \\ \dfrac{\gamma M_\xi^2}{1+(\gamma-1)M_\xi^2} & (M_\xi < 1) \end{cases} \tag{2.11}$$

where the axial Mach number is denoted by M_ξ. The coefficient ω has to be applied with a safety factor σ:

$$\omega = \min(1, \sigma\overline{\omega}) \tag{2.12}$$

so that the eigenvalues of the inviscid PNS equation set are real and the equations are hyperbolic. The effect of σ on ω is shown in figure 2.2.

When ω is less than 1, the source term $\mathbf{P}$ physically represents the mechanism which would allow for upstream propagation of information (Davis, Barnett, and Rakich 1986). Lubard and Helliwell (1974) have shown that when $\mathbf{P}$ is included in a finite-difference method as part of a backward difference, the method is unstable for small marching step sizes. If the source term $\mathbf{P}$ is dropped, the finite-difference method is stable up to the allowable marching step size for the numerical scheme. Therefore, the vector $\mathbf{P}$ on the right side of equation (2.6) is dropped in numerical calculations. Davis, Barnett, and Rakich (1986) demonstrated that this is a good approximation for a high Mach number, weakly interacting flow. However, the source term $\mathbf{P}$ can be included as a forward difference with a global iteration method (Davis, Barnett, and Rakich 1986) for calculating strongly interacting flows after one pass has been made to establish an initial pressure distribution.

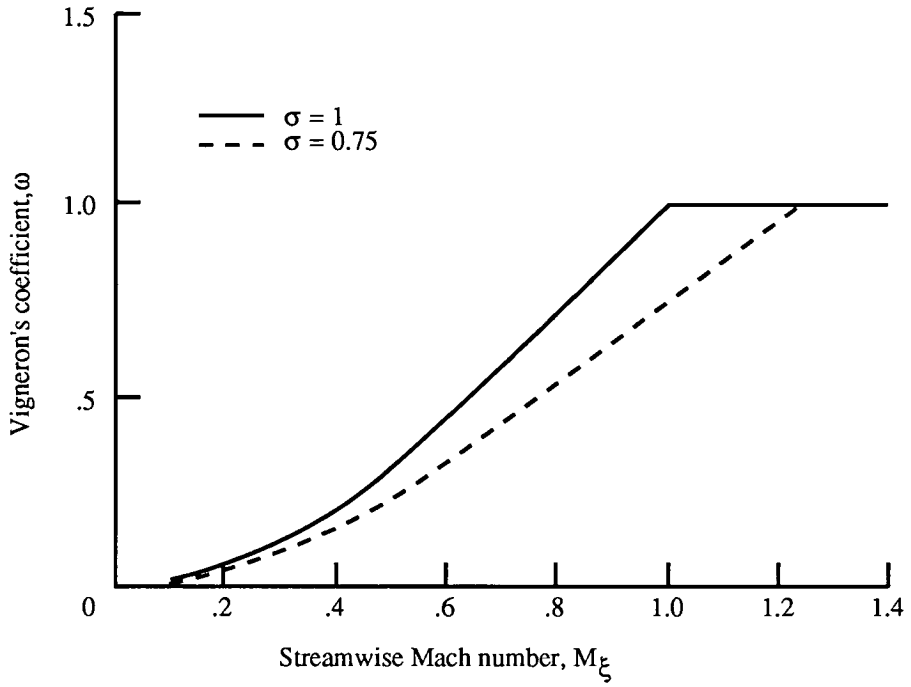


Figure 2.2. Effect of safety factor σ on Vigneron's coefficient ω.

In equation (2.6), ω is included in the differential with respect to ξ. Once the vector $\mathbf{P}$ is dropped in the numerical scheme, there is no longer conservation of the streamwise momentum (neglecting the streamwise viscous terms) in the subsonic region. Care has to be taken when integrating $\mathbf{E}^*$ so that any variation of ω is canceled out in the streamwise (ξ) direction. If the variation of ω is not canceled out, a nonphysical acceleration occurs in the subsonic streamwise direction which affects the accuracy of the PNS solution as an approximation to the full NS solution.[1] The pressure derivative in the streamwise direction after the vector $\mathbf{P}$ has been dropped can be expressed as

$$\left(\frac{\xi_x}{J}\omega p\right)_\xi - \omega p\left(\frac{\xi_x}{J}\right)_\xi = \frac{\xi_x}{J}(\omega p)_\xi$$
$$= \frac{\xi_x}{J}\omega p_\xi + \frac{\xi_x}{J}p\omega_\xi \qquad (2.13)$$

The last term on the right side has to be canceled out to eliminate the nonphysical accelerations caused by the variation of ω. Since ω is directly related to the streamwise Mach number, different solutions can be obtained for the same problem using slightly different numerical grids if the last term is not canceled out, by subtracting its value at the previous solution station from the numerical solution.

2.6. PNS Equations for Use With a Single Pass Method

The equation solved by a single pass space marching method can be obtained from equation (2.6) by dropping the vector $\mathbf{P}$, shifting all the crossflow flux derivatives to the right

[1] The author would like to acknowledge J. H. Morrison, Analytical Services & Materials, Inc. (Hampton, Va.) for discovery of this nonphysical acceleration from the variation of ω.

side, and adding a correction to cancel out any variation of ω in the streamwise direction:

$$\left(\frac{\xi_x \mathbf{E}^*}{J}\right)_\xi = -\left(\frac{\eta_x \mathbf{E}}{J} + \frac{\eta_y \mathbf{F}}{J} + \frac{\eta_z \mathbf{G}}{J}\right)_\eta - \left(\frac{\zeta_x \mathbf{E}}{J} + \frac{\zeta_y \mathbf{F}}{J} + \frac{\zeta_z \mathbf{G}}{J}\right)_\zeta + \left[\mathbf{E}^*\left(\frac{\xi_x}{J}\right)_\xi + \mathbf{E}\left(\frac{\eta_x}{J}\right)_\eta + \mathbf{E}\left(\frac{\zeta_x}{J}\right)_\zeta\right] + \mathbf{F}\left[\left(\frac{\eta_y}{J}\right)_\eta + \left(\frac{\zeta_y}{J}\right)_\zeta\right] + \mathbf{G}\left[\left(\frac{\eta_z}{J}\right)_\eta + \left(\frac{\zeta_z}{J}\right)_\zeta\right] + \frac{\xi_x}{J} p \mathbf{\Omega}_\xi \tag{2.14}$$

where

$$\mathbf{\Omega} = [0,\ \omega,\ 0,\ 0,\ 0]^T \tag{2.15}$$

The last term contains a vector $\mathbf{\Omega}_\xi$ which is added to cancel out any variation of ω in the streamwise direction when the source term $\mathbf{P}$ is dropped. The PNS equations described by equation (2.14) are a mixed set of hyperbolic-parabolic partial differential equations in ξ-space. Given that boundary conditions are known for $\mathbf{E}^*$ on an η-ζ surface and that appropriate initial conditions are known on a surface for $\xi = 0$, the system of equations can be space marched (integrated) in the ξ-direction.

2.7. Defining ω for Decoding $\mathbf{E}^*$

The primitive flow variables are used in the definition of the fluxes and to display and analyze the solution. The primitive flow variables have to be defined in terms of the dependent variable $\mathbf{E}^*$. Previously, numerical schemes used with the conservative form of the PNS equations have required that a change of variables be made from $\mathbf{E}^*$, to eliminate the difficulty of decoding $\mathbf{E}^*$ to obtain the primitive flow variables. This difficulty had to do with choosing the sign on the square-root function used for determining the streamwise velocity with the steady form of Euler's equations when the flow changes from supersonic to subsonic. Gielda and McRae (1986) eliminated this problem by using Vigneron's coefficient ω so that the sign does not change on the square-root function when the flow becomes subsonic. They defined the primitive variables in terms of $\mathbf{E}^*$ and ω as

$$\mathbf{E}^* = [\rho u,\ \rho u u + \omega p,\ \rho u v,\ \rho u w,\ (e_t + p)u]^T = [E_1^*,\ E_2^*,\ E_3^*,\ E_4^*,\ E_5^*]^T \tag{2.16}$$

so that

$$\left.\begin{aligned} v &= \frac{E_3^*}{E_1^*} \qquad w = \frac{E_4^*}{E_1^*} \qquad u = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \\ \rho &= \frac{E_1^*}{u} \qquad p = \frac{E_2^* - u E_1^*}{\omega} \end{aligned}\right\} \tag{2.17}$$

where

$$a = \frac{1}{2} \quad b = \frac{-\gamma E_2^*}{E_1^*[2\gamma - \omega(\gamma - 1)]} \quad c = \frac{\omega(\gamma - 1)}{[2\gamma - \omega(\gamma - 1)]}\left[\frac{E_5^*}{E_1^*} - \frac{1}{2}\left(v^2 + w^2\right)\right]$$

The value of ω must be known before the flow variables are computed from equations (2.16) and (2.17). Gielda and McRae lagged ω in their numerical scheme by defining it as a function of the primitive flow variables from the previous decoding of $\mathbf{E}^*$.

A different approach was taken in this study by defining ω in terms of $\mathbf{E}^*$ and determining it before decoding. This results in a cubic function defining ω in terms of $\mathbf{E}^*$:

$$\omega^3 - \frac{2(1+\sigma)\gamma}{\gamma-1}\omega^2 + \frac{\left[(1+\sigma)^2\gamma^2 + A\right]}{(\gamma-1)^2}\omega - \frac{2A\sigma\gamma}{(\gamma-1)^3} = 0 \tag{2.18}$$

where

$$A = \frac{\gamma^2 E_2^{*2}}{2E_5^* E_1^* - E_3^{*2} - E_4^{*2}}$$

The value of ω can be determined either by solving the cubic exactly or by using Newton-Raphson iteration. The advantage of solving for ω from $\mathbf{E}^*$ is that it allows a larger space marching step size in practice, especially when starting from free-stream or approximate initial conditions.

3. Integration Method

The objective of this study is to develop a single pass space marching numerical scheme for integrating the parabolized Navier-Stokes (PNS) equations which uses upwind approximations for the convection terms, is second-order accurate, and can be executed with vector operations. In this section the following topics are discussed: an upwind approximation of the convection terms using the solution of an approximate Riemann problem (RP), the use of upwind flux approximations in multistage explicit integration schemes which have second-order accuracy and, finally, a finite-difference upwind algorithm which can be used to integrate equation (2.14).

3.1. Selecting an Upwind Scheme for the PNS Equations

Upwind numerical schemes have recently become popular for solving nonlinear, hyperbolic, partial differential equations in conservation law form. First-order upwind approximations of the flux derivatives are used to eliminate numerical oscillations associated with solutions containing discontinuities. This section reviews methods for determining an upwind approximation for the purpose of selecting a method for use with the PNS convection terms in an explicit integration algorithm.

An upwind scheme applicable to the steady form of Euler's equations would be applicable to the PNS equations if a modification is made in the subsonic region to handle the splitting of the streamwise pressure gradient with Vigneron, Rakich, and Tannehill's approximation. For supersonic flow, the steady form of Euler's equations is identical to the convective terms of the PNS equations. The steady form of Euler's equations has hyperbolic character for supersonic flow and has a more complicated set of eigenvalues and eigenvectors than the unsteady form. Most previous applications of upwind schemes have been to the unsteady, conservative form of Euler's equations.

The characteristics of the PNS convection terms in ξ-η space can be examined by first determining the high Reynolds number limit ($1/\mathrm{Re} \rightarrow 0$) form of equation (2.5) and dropping the geometric conservation law (GCL) and source terms:

$$\begin{aligned} &\overline{\mathbf{E}}^*_\xi + \overline{\mathbf{F}}_{i_\eta} = 0 \quad \overline{\mathbf{E}}^*(0,\eta) = \overline{\mathbf{E}}^*_0(\eta) \\ &\overline{\mathbf{E}}^* = \frac{\xi_x}{J}\mathbf{E}^* \qquad \overline{\mathbf{F}}_i = \frac{\eta_x \mathbf{E}_i + \eta_y \mathbf{F}_i + \eta_z \mathbf{G}_i}{J} \end{aligned} \tag{3.1}$$

The system of equations is hyperbolic since the Jacobian matrix $\mathbf{A}$ is

$$\mathbf{A} = \frac{\partial \overline{\mathbf{F}}_i}{\partial \overline{\mathbf{E}}^*} \tag{3.2}$$

has real eigenvalues when $\sigma < 1$. Equation (3.1) represents an initial-value problem for a system of nonlinear hyperbolic conservation laws where the dependent variable $\overline{\mathbf{E}}^*(\xi,\eta)$ is a vector and $\overline{\mathbf{F}}$ is a vector-valued nonlinear function of $\overline{\mathbf{E}}^*$. Solutions of equation (3.1) for a given set of initial conditions may contain or develop discontinuities of the dependent variable. These "weak solutions" of Euler's equations physically represent shock waves or contact surfaces occurring in the flow. Nonphysical weak solutions, called expansion shocks, can also be solutions to equation (3.1). The physical solution obeys an entropy condition and is a solution to the viscous equations in the limit as $\varepsilon \rightarrow 0$:

$$\overline{\mathbf{E}}^*_\xi + \overline{\mathbf{F}}_{i_\eta} = \varepsilon \overline{\mathbf{E}}^*_{\eta\eta} \qquad (\varepsilon > 0) \tag{3.3}$$

The correct physical solution should be predicted by solving the PNS equations since the right side of equation (3.3) is approximated by using the physical viscous stresses and heat fluxes.

The semidiscrete form of equation (3.1) is obtained by approximating the flux on a numerical grid in the η-direction:

$$\overline{\mathbf{E}}^*_\xi + \left(\frac{\overline{\mathbf{F}}_{i_{j+\frac{1}{2}}} - \overline{\mathbf{F}}_{i_{j-\frac{1}{2}}}}{\Delta\eta} \right) = 0 \tag{3.4}$$

A first-order upwind approximation of a flux f for a scalar equation at the $j + \frac{1}{2}$ point can be determined simply by investigating the sign of λ, the eigenvalue (or wave speed):

$$f_{j+\frac{1}{2}} = \left\{ \begin{matrix} f_j & (\lambda > 0) \\ f_{j+1} & (\lambda < 0) \end{matrix} \right\} \tag{3.5}$$

For a system of equations where there can be λ's of mixed sign, the upwind flux approximation is determined from a splitting of either the flux vectors (FVS) or flux differences (FDS). (See summary in Chakravarthy 1987.) Both the FDS and the FVS approach can be used to separate the flux into contributions which can be associated with either the positive or the negative λ's.

The FVS approach is used in the schemes of Steger and Warming (1981) and Van Leer (1982) for the unsteady Euler equations. The flux vector is split into two new vectors which have either all negative or all positive λ's. A forward or backward difference is applied to the split vectors based on the sign of the λ's to obtain an upwind algorithm. Steger and Warming's method is based on a homogeneous property of the Euler equations which does not apply to the PNS equations as modified by the splitting of the streamwise pressure gradient.

The FDS approach is based on information about the evolution of the flow field obtained from the localized Riemann solutions between adjacent grid cells. A characteristic decomposition of the system of equations defined in the initial-value problem (eq. (3.1)) is made and a Riemann problem (RP) is formulated. The initial data for $\overline{\mathbf{E}}^*$ are assumed to be piecewise constant between points (fig. 3.1). An interface is assumed to exist between the two points at the symbolic location of $j+\frac{1}{2}$. The discontinuous initial data for the initial-value problem define the RP. Since $\overline{\mathbf{E}}^*$ is a vector and $\overline{\mathbf{F}}_i$ is a nonlinear vector, the solution of the RP involves nonlinear algebraic equations and logical conditions for determining whether the solution contains a shock wave or a smooth expansion. A RP is solved to determine the evolution of the interface in ξ-space and the intermediate values of $\overline{\mathbf{E}}^*$. The solution of the RP (for steady flow of a supersonic gas) contains four constant states of $\overline{\mathbf{E}}^*$, separated by five waves evolving from the interface (fig. 3.2). Each wave is associated with a λ of the Jacobian matrix $\mathbf{A}$. The waves can represent a shock wave, a rarefaction fan, or a contact surface. Once the intermediate values of $\overline{\mathbf{E}}^*$ are known, the flux difference across a wave can be determined. The flux difference across a wave with a positive λ is considered a positive flux difference and the flux difference across a wave with a negative λ is considered a negative flux difference. The positive flux differences are used as part of a backward approximation of the flux derivative, and the negative flux differences are used as part of a forward approximation of the flux derivative.

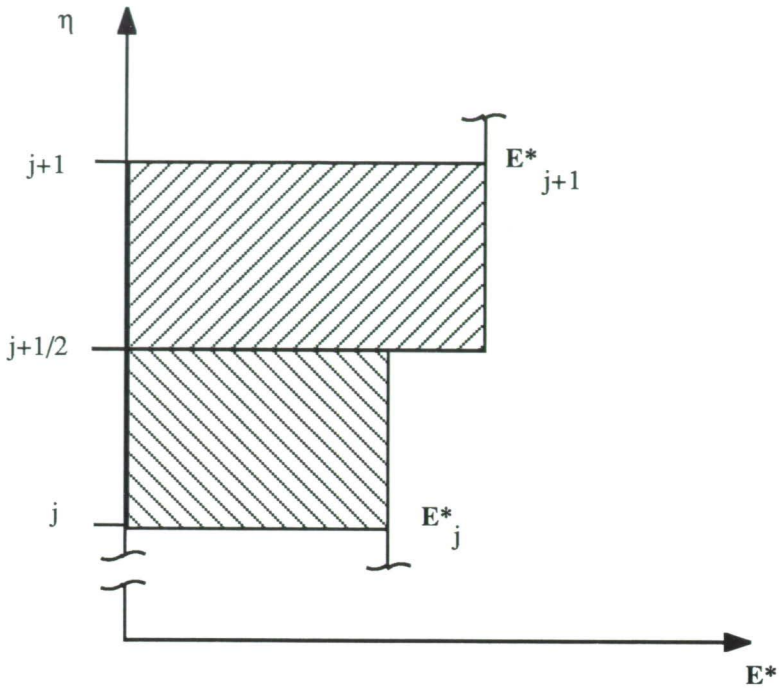


Figure 3.1. Initial data distribution between adjacent points.

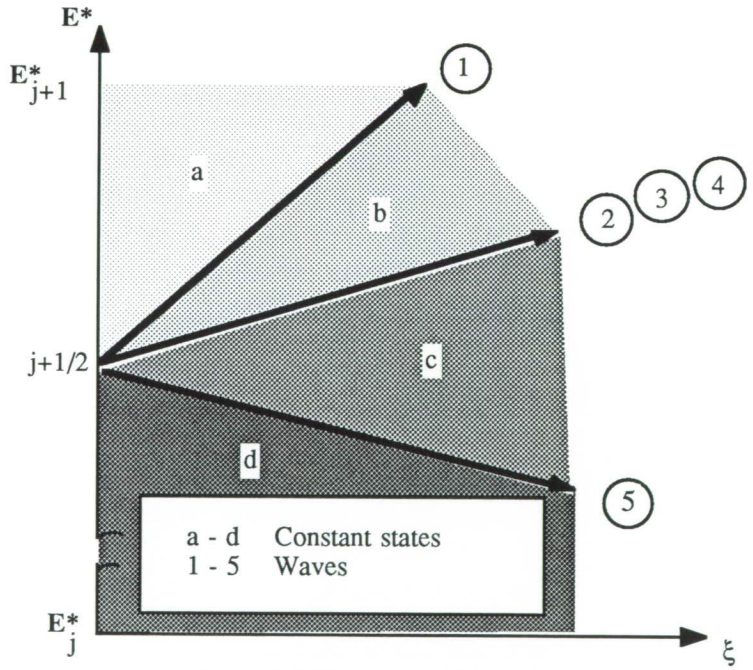


Figure 3.2. Riemann problem solution.

Godunov (1960) developed the first upwind scheme for the unsteady, conservative form of Euler's equations. This scheme was based on an exact solution of the RP. The exact solution to the RP is expensive to compute since it requires an iteration process. Osher and Solomon (1982), Pandolfi (1984), Roe (1981), and others have proposed approximate RP solvers for the unsteady Euler equations. Roe (1981) and Pandolfi (1985) have developed schemes for steady supersonic flow using the steady form of Euler's equations. Pandolfi's schemes are limited to Euler's equations, while Roe's scheme is applicable to any hyperbolic equation set that

has unique eigenvectors. Roe's method has been applied to the PNS equation set in areas of supersonic flow by Lawrence, Tannehill, and Chaussee (1986, 1987). They found that applying the upwind scheme in the subsonic region where the streamwise pressure gradient was split was "detrimental to both the stability and accuracy of the algorithm." They used a noniterative implicit integration scheme and made a change of variables so that the equations were solved in **U**-space instead of $\mathbf{E}^*$-space. For the three-dimensional PNS equations, Lawrence, Chaussee, and Tannehill (1987) used the eigenvectors and eigenvalues for Euler's equations expressed in a rotated Cartesian coordinate system for solving the approximate RP in **U**-space. A modification detailed below of Roe's scheme for use with the three-dimensional PNS equations expressed in a generalized coordinate system can be applied for solving the approximate RP in $\mathbf{E}^*$-space. This modification eliminates the problem experienced by Lawrence, Chaussee, and Tannehill (1987) and enables the scheme to be used throughout the flow field, including locations where the streamwise pressure gradient is split.

3.2. Application of an Approximate Riemann Solver to the PNS Equations

Roe's (1981) method is based on solving an approximate RP exactly. The solution of the RP is used in a numerical scheme to obtain a splitting of the flux differences in the crossflow directions. Consider a discrete grid of η_j points in the η-direction. An exact solution is sought for the following approximate equation (for the convection terms of the PNS equations) between the points j and $j+1$:

$$\widehat{\mathbf{E}}^*_\xi + \widehat{\mathbf{A}}\widehat{\mathbf{E}}^*_\eta = 0 \tag{3.6}$$

with initial conditions,

$$\widehat{\mathbf{E}}^*(0,\eta) = \begin{cases} \widehat{\mathbf{E}}^*_{j+1} & \left(\eta > j+\frac{1}{2}\right) \\ \widehat{\mathbf{E}}^*_j & \left(\eta < j+\frac{1}{2}\right) \end{cases}$$

where $\widehat{\mathbf{A}}$ is a constant matrix (based on local conditions). The discrete matrix $\widehat{\mathbf{A}}$ is formed using specialized square-root averaging of the primitive flow variables at points j and $j+1$ so that conservation properties are maintained. The $\widehat{\ }$ variables are formed using a locally constant value of the streamwise splitting coefficient ω. The matrix $\widehat{\mathbf{A}}$ evaluated at the interface location $\left(j+\frac{1}{2}\right)$ is formed from

$$\widehat{\mathbf{A}}_{j+\frac{1}{2}} = \left.\frac{\partial\left(\frac{\eta_x}{J}\mathbf{E}_i + \frac{\eta_y}{J}\mathbf{F}_i + \frac{\eta_z}{J}\mathbf{G}_i\right)}{\partial\left(\frac{\xi_x}{J}\mathbf{E}^*\right)}\right|_{j+\frac{1}{2}} \tag{3.7}$$

The matrix $\widehat{\mathbf{A}}$ has the following conservative property if it is evaluated using specialized square-root-averaged variables (Roe, 1981):

$$\widehat{\mathbf{F}}_{j+1} - \widehat{\mathbf{F}}_j = \widehat{\mathbf{A}}_{j+\frac{1}{2}}\left(\frac{\xi_x}{J}\right)_{j+\frac{1}{2}}\left(\widehat{\mathbf{E}}^*_{j+1} - \widehat{\mathbf{E}}^*_j\right) \tag{3.8}$$

where

$$\widehat{\mathbf{F}}_k = \left(\frac{\eta_x}{J}\right)_{j+\frac{1}{2}}\mathbf{E}_{i_k} + \left(\frac{\eta_y}{J}\right)_{j+\frac{1}{2}}\mathbf{F}_{i_k} + \left(\frac{\eta_z}{J}\right)_{j+\frac{1}{2}}\mathbf{G}_{i_k}$$

and where k is a dummy index for j or $j+1$ and the metrics are held constant. For the PNS equations, recall that $\widehat{\mathbf{E}}^*$ has been modified to include only a fraction of the streamwise pressure gradient in the subsonic region. The allowable amount of the streamwise pressure gradient can change rapidly in the η-direction. This variation of ω was eliminated in the projection of $\widehat{\mathbf{E}}^*$

into $\widehat{\mathbf{F}}$ in the definition of the RP between points j and $j+1$ by defining $\widehat{\mathbf{E}}^*$ using a locally constant value of ω:

$$\widehat{\mathbf{E}}^*_k = \left[\rho u,\ \rho u^2 + \omega_{j+\frac{1}{2}} p,\ \rho uv,\ \rho uw,\ (e_t + p)u\right]^T_k \tag{3.9}$$

This modification was necessary to eliminate the problem experienced by Lawrence, Tannehill, and Chaussee (1986, 1987) in the subsonic region. The requirement of using a fixed value of ω for the RP increases the difficulty of using an implicit integration scheme when $\omega \neq 1$ (subsonic region). This is because the $\widehat{\mathbf{E}}^*$ at point j in the RP defined at $j+\frac{1}{2}$ is not equal to the $\widehat{\mathbf{E}}^*$ at point j defined in the RP at point $j-\frac{1}{2}$ because of the different values of ω used at the $j \pm \frac{1}{2}$ points.

The objective of solving the RP between points j and $j+1$ is to split the flux difference in equation (3.8) into five parts, one for each eigenvalue, which can be associated with either the positive or the negative eigenvalues. Roe's method for solving the RP consists of first calculating the square-root-averaged variables for the interface at $j+\frac{1}{2}$. The eigenvalues λ and eigenvectors $\hat{\mathbf{e}}$ of the Jacobian matrix $\widehat{\mathbf{A}}$, and the wave strengths α are calculated with the square-root-averaged variables using the equations given in appendix B. The wave strengths are defined as

$$\sum_{m=1}^{5} (\alpha_m \hat{\mathbf{e}}_m)_{j+\frac{1}{2}} = \left(\frac{\xi_x}{J}\right)_{j+\frac{1}{2}} \left(\widehat{\mathbf{E}}^*_{j+1} - \widehat{\mathbf{E}}^*_j\right) \tag{3.10}$$

The flux difference across the mth wave is $\lambda_m \alpha_m \hat{\mathbf{e}}_m$. The sum of the flux difference across all the waves is equal to the difference of the flux between points j and $j+1$.

$$\sum_{m=1}^{5} (\lambda_m \alpha_m \hat{\mathbf{e}}_m)_{j+\frac{1}{2}} = \widehat{\mathbf{F}}_{j+1} - \widehat{\mathbf{F}}_j \tag{3.11}$$

The total flux difference between points j and $j+1$ can now be split into the total positive ($\mathbf{df}^+$) and negative ($\mathbf{df}^-$) flux differences. The $\mathbf{df}^+$ and $\mathbf{df}^-$ vectors are calculated from

$$\widehat{\mathbf{F}}_{j+1} - \widehat{\mathbf{F}}_j = \mathbf{df}^+_{j+\frac{1}{2}} + \mathbf{df}^-_{j+\frac{1}{2}} \tag{3.12}$$

where

$$\mathbf{df}^+_{j+\frac{1}{2}} = \sum_{m=1}^{5} \frac{\lambda_m + |\lambda_m|}{2} (\alpha_m \hat{\mathbf{e}}_m)_{j+\frac{1}{2}} \quad \mathbf{df}^-_{j+\frac{1}{2}} = \sum_{m=1}^{5} \frac{\lambda_m - |\lambda_m|}{2} (\alpha_m \hat{\mathbf{e}}_m)_{j+\frac{1}{2}} \tag{3.13}$$

The eigenvalues and eigenvectors for the three-dimensional inviscid PNS equations in generalized coordinates were determined in part using the symbolic manipulation language MACSYMA. The vectors $\mathbf{df}^+$ and $\mathbf{df}^-$ are used as the building blocks for obtaining an upwind flux approximation at the $j+\frac{1}{2}$ point.

3.3. Upwind Flux Approximations

A first-order upwind flux approximation at $j+\frac{1}{2}$ can be obtained by modifying either a forward, a backward, or a central flux approximation with FDS determined from the solution of the RP:

$$\overline{\mathbf{F}}_{i_{j+\frac{1}{2}}} = \mathbf{H}_{j+\frac{1}{2}} = \begin{cases} \overline{\mathbf{F}}_{i_{j+1}} - \mathbf{df}^+_{j+\frac{1}{2}} \\ \overline{\mathbf{F}}_{i_j} + \mathbf{df}^-_{j+\frac{1}{2}} \\ \frac{1}{2}\left(\overline{\mathbf{F}}_{i_{j+1}} + \overline{\mathbf{F}}_{i_j}\right) - \frac{1}{2}\left(\mathbf{df}^+_{j+\frac{1}{2}} - \mathbf{df}^-_{j+\frac{1}{2}}\right) \end{cases} \tag{3.14}$$

A numerical scheme that uses a first-order upwind approximation of the flux has the advantage of resolving discontinuities without spurious oscillations. Unfortunately, the dissipation inherent in the first-order upwind scheme makes it impractical for global use. In practice, second- and higher-order flux approximations are used with numerical schemes to minimize truncation errors. The higher-order flux approximation cannot resolve a discontinuity such as a shock wave without an overshoot or undershoot. The oscillations around a shock wave are minimized either by adding additional dissipation to the numerical scheme or by modifying the flux approximation with a flux limiter.

The classical way of minimizing the oscillations is to use additional dissipation. Dissipation is added to the numerical scheme by including either a second- or a fourth-order derivative of the dependent variable multiplied by a user-specified constant. The disadvantages of this procedure are determining the best value for the constant and adding a nonphysical stresslike term to the equation set that is being solved.

Another approach is to use a nonlinear method to change or "limit" the higher-order flux approximation to first-order in the neighborhood of a discontinuity to eliminate numerical oscillations (fig. 3.3). A second-order upwind approximation of the flux using flux limiters is

$$\bar{\bar{\mathbf{F}}}_{i_{j+\frac{1}{2}}} = \mathbf{H}_{j+\frac{1}{2}} + \frac{1}{2}\left(\overline{\overline{\mathbf{df}}}^{+}_{j-\frac{1}{2}} - \overline{\mathbf{df}}^{-}_{j+\frac{3}{2}}\right) \tag{3.15}$$

The flux differences designated with an overbar are treated with the minmod flux limiter.

$$\left.\begin{aligned} \overline{\overline{\mathbf{df}}}^{+}_{j-\frac{1}{2}} &= \text{minmod}\left(\mathbf{df}^{+}_{j-\frac{1}{2}},\ \beta\,\mathbf{df}^{+}_{j+\frac{1}{2}}\right) \\ \overline{\mathbf{df}}^{-}_{j+\frac{3}{2}} &= \text{minmod}\left(\mathbf{df}^{-}_{j+\frac{3}{2}},\ \beta\,\mathbf{df}^{-}_{j+\frac{1}{2}}\right) \end{aligned}\right\} \tag{3.16}$$

The minmod flux limiter is defined as

$$\text{minmod}(x, y) = \text{sign}(x)\ \max\{0, \min[|x|,\ y\ \text{sign}(x)]\} \tag{3.17}$$

The minmod function limits the overbar flux differences to a first-order approximation around captured shocks in order to minimize oscillations. The parameter β varies slightly for different flux approximations. For second-order upwind flux approximations, β has a maximum value of 2 for obtaining oscillation-free shock capturing. The advantage of the nonlinear flux limiter is that oscillation-free results can be obtained without adding artificial stresses to the numerical scheme. Different types of flux limiters are in use and the reader is referred to the work of Sweby (1984) for more information.

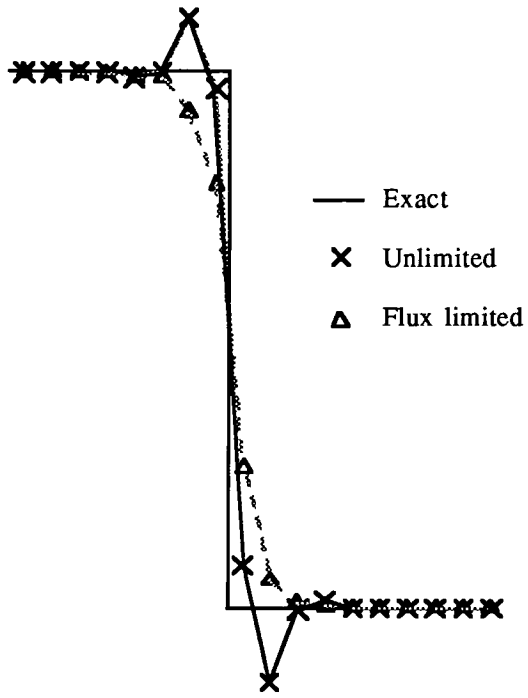


Figure 3.3. Comparison of unlimited and flux-limited shock capturing with a second-order flux approximation.

3.4. Second-Order Explicit Upwind Integration Method

First- and second-order upwind flux approximations for the PNS inviscid equations in ξ-η space were defined in the previous sections. This section presents the integration of the initial-value problem described by equation (3.1) using a method that incorporates upwind flux approximations and is globally second-order accurate. The flux approximations in MacCormack's (1969) scheme are modified to obtain a flux-limited version of Warming and Beam's (1975) second-order explicit upwind scheme.

MacCormack's (1969) method is a two-stage explicit scheme that has been used extensively for solving the Euler, full Navier-Stokes, and, recently by Gielda and McRae (1986), the PNS equations. The one-sided inviscid flux approximations used in MacCormack's method are modified with the FDS obtained from the solution of the RP. The resulting unlimited form of the algorithm is similar to the Warming and Beam upwind (WBU) algorithm. The WBU scheme has twice the linear stability limit of MacCormack's. The larger integration step size of the WBU scheme compensates for some of the additional cost of determining the solution to the RP's. The MacCormack and WBU schemes are classified as Lax and Wendroff (1960) type schemes.

Second-order and higher integration schemes for solving initial-value, boundary-value problems are derived using either the fully discrete method of Lax and Wendroff (1960) or the semidiscrete method of lines. The Lax-Wendroff scheme is a finite-difference method that is derived by satisfying a Taylor series expansion about the solution point. The derivatives with respect to the independent variable of expansion are replaced with the original partial differential equation. Another way to integrate the equation set is the method-of-lines approximation. The first step in the method-of-lines procedure is to express the partial differential equation in semidiscrete form. The remaining partial derivatives are treated numerically as ordinary differential equations. Two- and three-dimensional problems with source terms can be easily

handled, and any appropriate ordinary differential equation solver can be used to perform the integration.

The method-of-lines procedure can be modified for determining a numerical scheme equivalent to the MacCormack and WBU schemes. First apply a second-order modified Euler method, also called Heun's method (Gear 1971), to the initial-value problem expressed in semidiscrete form (eq. (3.4)):

$$\overline{\mathbf{E}}_j^{*(1)} = \overline{\mathbf{E}}_j^{*(0)} - \frac{\Delta\xi}{\Delta\eta}\left(\overline{\mathbf{F}}^{(0)}_{i_{j+\frac{1}{2}}} - \overline{\mathbf{F}}^{(0)}_{i_{j-\frac{1}{2}}}\right) \tag{3.18a}$$

$$\overline{\mathbf{E}}_j^{*(2)} = \overline{\mathbf{E}}_j^{*(0)} - \frac{\Delta\xi}{2\,\Delta\eta}\left(\overline{\mathbf{F}}^{(0)}_{i_{j+\frac{1}{2}}} - \overline{\mathbf{F}}^{(0)}_{i_{j-\frac{1}{2}}}\right) - \frac{\Delta\xi}{2\,\Delta\eta}\left(\overline{\mathbf{F}}^{(1)}_{i_{j+\frac{1}{2}}} - \overline{\mathbf{F}}^{(1)}_{i_{j-\frac{1}{2}}}\right) \tag{3.18b}$$

where the superscript represents the stage of evaluation and (0) represents the initial value. The method-of-lines procedure guarantees a global second-order scheme when the midpoint fluxes are approximated with the same second-order flux approximations in each stage. Equivalent Lax-Wendroff methods can be formulated if different types of flux approximations are used in the different stages. The accuracy of the method must be checked when different flux approximations are used unless the scheme is a known Lax-Wendroff method.

MacCormack's scheme is obtained using one-sided flux approximations:

$$\left.\begin{array}{lll} \text{Stage 1} & \overline{\mathbf{F}}^{(0)}_{i_{j+\frac{1}{2}}} = \overline{\mathbf{F}}^{(0)}_{i_j} & \\ \text{Stage 2} & \overline{\mathbf{F}}^{(0)}_{i_{j+\frac{1}{2}}} = \overline{\mathbf{F}}^{(0)}_{i_j} & \overline{\mathbf{F}}^{(1)}_{i_{j+\frac{1}{2}}} = \overline{\mathbf{F}}^{(1)}_{i_{j+1}} \end{array}\right\} \tag{3.19}$$

Substituting for the flux differencing evaluated at level (0) in the second stage (eq. (3.18b)) with the first stage equation (eq. (3.18a)) results in the traditional form of the second stage of MacCormack's algorithm:

$$\left.\begin{aligned} \overline{\mathbf{E}}_j^{*(1)} &= \overline{\mathbf{E}}_j^{*(0)} - \frac{\Delta\xi}{\Delta\eta}\left(\overline{\mathbf{F}}^{(0)}_{i_j} - \overline{\mathbf{F}}^{(0)}_{i_{j-1}}\right) \\ \overline{\mathbf{E}}_j^{*(2)} &= \frac{1}{2}\left[\overline{\mathbf{E}}_j^{*(0)} + \overline{\mathbf{E}}_j^{*(1)} - \frac{\Delta\xi}{\Delta\eta}\left(\overline{\mathbf{F}}^{(1)}_{i_{j+1}} - \overline{\mathbf{F}}^{(1)}_{i_j}\right)\right] \end{aligned}\right\} \tag{3.20}$$

The upwind scheme is obtained by adding FDS to the one-sided flux approximations:

$$\left.\begin{array}{lll} \text{Stage 1} & \overline{\mathbf{F}}^{(0)}_{i_{j+\frac{1}{2}}} = \overline{\mathbf{F}}^{(0)}_{i_j} + \mathbf{df}^{-(0)}_{j+\frac{1}{2}} & \\ \text{Stage 2} & \overline{\mathbf{F}}^{(0)}_{i_{j+\frac{1}{2}}} = \overline{\mathbf{F}}^{(0)}_{i_j} + \mathbf{df}^{-(0)}_{j+\frac{1}{2}} + \frac{1}{2}\left(\overline{\overline{\mathbf{df}}}^{+(0)}_{j-\frac{1}{2}} - \overline{\mathbf{df}}^{-(0)}_{j+\frac{3}{2}}\right) & \\ & \overline{\mathbf{F}}^{(1)}_{i_{j+\frac{1}{2}}} = \overline{\mathbf{F}}^{(1)}_{i_{j+1}} - \mathbf{df}^{+(1)}_{j+\frac{1}{2}} + \frac{1}{2}\left(\overline{\overline{\mathbf{df}}}^{+(0)}_{j-\frac{1}{2}} - \overline{\mathbf{df}}^{-(0)}_{j+\frac{3}{2}}\right) & \end{array}\right\} \tag{3.21}$$

The flux approximations used in stage 2 differ from those in stage 1 and contain terms evaluated at both levels. The numerical fluxes approximate the flux at a particular point in space, which may differ from stage to stage. The flux definitions in stage 2 are not unique. For example, all the terms evaluated at (0) could be placed together. The above form was preferred since it represented a second-order discrete approximation of the flux in η-space. Different flux approximations used in different stages of Runge-Kutta integration schemes have been used

in developing the third-order schemes of Rusanov (1970) and Burstein and Mirin (1970). The traditional form of the second stage of the WBU scheme is obtained by substituting the equation for the first stage into the second:

$$\left.\begin{aligned}
\overline{\mathbf{E}}_j^{*(1)} &= \overline{\mathbf{E}}_j^{*(0)} - \frac{\Delta\xi}{\Delta\eta}\left[\overline{\mathbf{F}}_{i_j}^{(0)} - \overline{\mathbf{F}}_{i_{j-1}}^{(0)} + \left(\mathbf{df}_{j+\frac{1}{2}}^{-(0)} - \mathbf{df}_{j-\frac{1}{2}}^{-(0)}\right)\right] \\
\overline{\mathbf{E}}_j^{*(2)} &= \frac{1}{2}\left(\overline{\mathbf{E}}_j^{*(0)} + \overline{\mathbf{E}}_j^{*(1)} - \frac{\Delta\xi}{\Delta\eta}\left\{\overline{\mathbf{F}}_{i_{j+1}}^{(1)} - \overline{\mathbf{F}}_{i_j}^{(1)} - \left(\mathbf{df}_{j+\frac{1}{2}}^{+(1)} - \mathbf{df}_{j-\frac{1}{2}}^{+(1)}\right)\right.\right. \\
&\quad \left.\left. + \left[\left(\overline{\overline{\mathbf{df}}}_{j-\frac{1}{2}}^{+(0)} - \overline{\mathbf{df}}_{j+\frac{3}{2}}^{-(0)}\right) - \left(\overline{\overline{\mathbf{df}}}_{j-\frac{3}{2}}^{+(0)} - \overline{\mathbf{df}}_{j+\frac{1}{2}}^{-(0)}\right)\right]\right\}\right)
\end{aligned}\right\} \tag{3.22}$$

The advantage of this upwind scheme is that it can easily be obtained by modifying the inviscid fluxes used in MacCormack's scheme. The above scheme differs in practice from other upwind schemes because the first-order upwind flux approximation is based on modifying either a forward or a backward flux instead of a central flux (eq. (3.14)). The metrics used in the definition of the FDS are defined so that any downwind contributions exactly cancel.

Consider the first-order approximation of the flux derivative including the geometric conservation law (GCL) terms on a grid where $\bar{\eta} = \eta(y)$ with the upwind flux definition based on a backward flux approximation:

$$\begin{aligned}
&\overline{\mathbf{F}}_{i_{j+\frac{1}{2}}} - \overline{\mathbf{F}}_{i_{j-\frac{1}{2}}} - \mathrm{GCL}_j \\
&\qquad = \left(\frac{\eta_y}{J}\mathbf{F}_i\right)_j + \mathbf{df}_{j+\frac{1}{2}}^{-} - \left(\frac{\eta_y}{J}\mathbf{F}_i\right)_{j-1} - \mathbf{df}_{j-\frac{1}{2}}^{-} - \mathbf{F}_{i_j}\left[\left(\frac{\eta_y}{J}\right)_j - \left(\frac{\eta_y}{J}\right)_{j-1}\right] \\
&\qquad = \left(\frac{\eta_y}{J}\right)_{j-1}\left(\mathbf{F}_{i_j} - \mathbf{F}_{i_{j-1}}\right) + \mathbf{df}_{j+\frac{1}{2}}^{-} - \mathbf{df}_{j-\frac{1}{2}}^{-}
\end{aligned} \tag{3.23}$$

If all the eigenvalues are positive, the FDS terms would be zero and result in a backward difference approximation of the flux derivative. If all the eigenvalues are negative, the FDS terms exactly cancel out the backward difference if

$$\left(\frac{\eta_y}{J}\right)_{j-\frac{1}{2}} = \left(\frac{\eta_y}{J}\right)_{j-1} \qquad (\lambda < 0) \tag{3.24}$$

and thus

$$\mathbf{df}_{j-\frac{1}{2}}^{-} = \left(\frac{\eta_y}{J}\right)_{j-\frac{1}{2}}\left(\mathbf{F}_{i_j} - \mathbf{F}_{i_{j-1}}\right) \qquad (\text{All } \lambda < 0)$$

which results in a forward difference approximation of the flux derivative.

If an upwind flux is formed based on a forward flux approximation, then the forward difference would be canceled by the FDS terms if all the eigenvalues are positive and the metrics are defined by

$$\left(\frac{\eta_y}{J}\right)_{j+\frac{1}{2}} = \left(\frac{\eta_y}{J}\right)_{j+1} \qquad (\lambda > 0) \tag{3.25}$$

and thus

$$\mathbf{df}_{j+\frac{1}{2}}^{+} = \left(\frac{\eta_y}{J}\right)_{j+\frac{1}{2}}\left(\mathbf{F}_{i_{j+1}} - \mathbf{F}_{i_j}\right) \qquad (\text{All } \lambda > 0)$$

The metrics for a general transformation, $\eta = \eta(x, y, z)$, would be defined at the same points as above for the positive and negative flux differences. The different approximations for the upwind flux should be alternated after one complete cycle of the algorithm to eliminate biasing.

3.5. Viscous Stress and Heat Flux Differencing

The finite-difference approximations of the stress tensor and the heat fluxes are made so that their derivatives formed in each stage of the algorithm are second-order accurate. This requires separate approximations of the stress tensor and the heat fluxes for each differencing direction. The separate approximations of the stress tensor and heat fluxes result in a source term that is canceled by including the GCL terms. The derivatives of the stress tensor for the x-momentum equation including the first GCL term are

$$\left(\frac{\eta_x\tau'_{xx}+\eta_y\tau'_{xy}+\eta_z\tau'_{xz}}{J}\right)_\eta+\left(\frac{\zeta_x\tau''_{xx}+\zeta_y\tau''_{xy}+\zeta_z\tau''_{xz}}{J}\right)_\zeta-\tau'_{xx}\left(\frac{\eta_x}{J}\right)_\eta-\tau''_{xx}\left(\frac{\zeta_x}{J}\right)_\zeta-\ldots \tag{3.26}$$

where two discrete approximations of the same stress are used. The single prime denotes stresses to be differenced in the η-direction, and the double prime the ζ-direction. Let j represent the grid points in the η-direction, and k the ζ-direction. For example, assume that a forward difference of the viscous stresses in the first stage of the algorithm is required. The discrete approximation of τ_{xy} in a generalized coordinate system ($\Delta\eta=\Delta\zeta=1$) would be

$$\left.\begin{aligned}\left(\tau'_{xy}\right)_{j,k}&=\frac{\mu_{j,k}}{\mathrm{Re}_\infty}\left[\eta_{y_{j,k}}\left(u_{j,k}-u_{j-1,k}\right)+\zeta_{y_{j,k}}\left(\frac{u_{j,k+1}-u_{j,k-1}}{2}\right)+\ldots\right]\\ \left(\tau''_{xy}\right)_{j,k}&=\frac{\mu_{j,k}}{\mathrm{Re}_\infty}\left[\eta_{y_{j,k}}\left(\frac{u_{j+1,k}-u_{j-1,k}}{2}\right)+\zeta_{y_{j,k}}\left(u_{j,k}-u_{j,k-1}\right)+\ldots\right]\end{aligned}\right\} \tag{3.27}$$

This differencing scheme for the stress and heat flux terms results in a source term that has to be canceled by including the geometric conservation law (GCL) in the integration algorithm (see Gielda and McRae 1986).

3.6. Explicit Upwind Integration Scheme for the Three-Dimensional PNS Equations

An explicit upwind integration scheme for the three-dimensional PNS equations can be formed using the flux-limited form of the WBU scheme and the stress and heat flux differencing developed by Gielda and McRae (1986). A second-order, two-stage, explicit, upwind scheme for the PNS equations is

$$\left.\begin{aligned}
\text{Stage 1}\quad &\left(\frac{\xi_x}{J}\mathbf{E}^*\right)^p_{j,k}=\left(\frac{\xi_x}{J}\mathbf{E}^*\right)^n_{j,k}-\left(\overline{\mathbf{F}}^n_{i_{j+\frac12,k}}-\overline{\mathbf{F}}^n_{i_{j-\frac12,k}}\right)-\left(\overline{\mathbf{G}}^n_{i_{j,k+\frac12}}-\overline{\mathbf{G}}^n_{i_{j,k-\frac12}}\right)\\
&+\left(\overline{\mathbf{F}}^n_{v_{j+1,k}}-\overline{\mathbf{F}}^n_{v_{j,k}}\right)+\left(\overline{\mathbf{G}}^n_{v_{j,k+1}}-\overline{\mathbf{G}}^n_{v_{j,k}}\right)+\mathrm{GCL}^n_{j,k}+\left(\frac{\xi_x}{J}p\right)^n_{j,k}\left(\Omega^n_{j,k}-\Omega^{n-1}_{j,k}\right)\\
\text{Stage 2}\quad &\left(\frac{\xi_x}{J}\mathbf{E}^*\right)^{n+1}_{j,k}=\frac12\left[\left(\frac{\xi_x}{J}\mathbf{E}^*\right)^p_{j,k}+\left(\frac{\xi_x}{J}\mathbf{E}^*\right)^n_{j,k}-\left(\overline{\mathbf{F}}^p_{i_{j+\frac12,k}}-\overline{\mathbf{F}}^p_{i_{j-\frac12,k}}\right)-\left(\overline{\mathbf{G}}^p_{i_{j,k+\frac12}}-\overline{\mathbf{G}}^p_{i_{j,k-\frac12}}\right)\right.\\
&\left.+\left(\overline{\mathbf{F}}^p_{v_{j,k}}-\overline{\mathbf{F}}^p_{v_{j-1,k}}\right)+\left(\overline{\mathbf{G}}^p_{v_{j,k}}-\overline{\mathbf{G}}^p_{v_{j,k-1}}\right)+\mathrm{GCL}^p_{j,k}+\left(\frac{\xi_x}{J}p\right)^p_{j,k}\left(\Omega^p_{j,k}-\Omega^n_{j,k}\right)\right]\\
&\overline{\mathbf{F}}_{v_{j,k}}=\frac{\eta_x}{J}\mathbf{E}'_{v_{j,k}}+\frac{\eta_y}{J}\mathbf{F}'_{v_{j,k}}+\frac{\eta_z}{J}\mathbf{G}'_{v_{j,k}}\\
&\overline{\mathbf{G}}_{v_{j,k}}=\frac{\zeta_x}{J}\mathbf{E}''_{v_{j,k}}+\frac{\zeta_y}{J}\mathbf{F}''_{v_{j,k}}+\frac{\zeta_z}{J}\mathbf{G}''_{v_{j,k}}
\end{aligned}\right\} \tag{3.28}$$

The superscript n stands for the values determined at the initial condition, the p values are based on the result of the first stage, and $n+1$ stands for the solution point. The fluxes $\mathbf{F}_v$ and $\mathbf{G}_v$ denote the diffusion terms. The single prime denotes viscous stress and heat fluxes to be differenced in the η-direction, and the double prime denotes differencing in the ζ-direction.

3.7. Geometric Conservation Law

The GCL term included in the numerical algorithm is used to cancel source terms occurring from the differencing of the stress terms as mentioned previously and the use of the Vigneron coefficient in the dependent flux vector (see Gielda and McRae 1986 for details). The GCL term in equation (3.28) is defined as

$$
\left.
\begin{aligned}
\mathrm{GCL}^n_{j,k} &= \mathbf{E}^{*n}_{j,k}\left[\left(\frac{\xi_x}{J}\right)^{n+1}_{j,k} - \left(\frac{\xi_x}{J}\right)^{n}_{j,k}\right] \\
&\quad + \mathbf{E}'^{n}_{j,k}\left[\left(\frac{\eta_x}{J}\right)^{n}_{j+1,k} - \left(\frac{\eta_x}{J}\right)^{n}_{j,k}\right] + \mathbf{E}''^{n}_{j,k}\left[\left(\frac{\zeta_x}{J}\right)^{n}_{j,k+1} - \left(\frac{\zeta_x}{J}\right)^{n}_{j,k}\right] \\
&\quad + \mathbf{F}'^{n}_{j,k}\left[\left(\frac{\eta_y}{J}\right)^{n}_{j+1,k} - \left(\frac{\eta_y}{J}\right)^{n}_{j,k}\right] + \mathbf{F}''^{n}_{j,k}\left[\left(\frac{\zeta_y}{J}\right)^{n}_{j,k+1} - \left(\frac{\zeta_y}{J}\right)^{n}_{j,k}\right] \\
&\quad + \mathbf{G}'^{n}_{j,k}\left[\left(\frac{\eta_z}{J}\right)^{n}_{j+1,k} - \left(\frac{\eta_z}{J}\right)^{n}_{j,k}\right] + \mathbf{G}''^{n}_{j,k}\left[\left(\frac{\zeta_z}{J}\right)^{n}_{j,k+1} - \left(\frac{\zeta_z}{J}\right)^{n}_{j,k}\right] \\
\mathrm{GCL}^p_{j,k} &= \mathbf{E}^{*p}_{j,k}\left[\left(\frac{\xi_x}{J}\right)^{n+1}_{j,k} - \left(\frac{\xi_x}{J}\right)^{n}_{j,k}\right] \\
&\quad + \mathbf{E}'^{p}_{j,k}\left[\left(\frac{\eta_x}{J}\right)^{n+1}_{j,k} - \left(\frac{\eta_x}{J}\right)^{n+1}_{j-1,k}\right] + \mathbf{E}''^{p}_{j,k}\left[\left(\frac{\zeta_x}{J}\right)^{n+1}_{j,k} - \left(\frac{\zeta_x}{J}\right)^{n+1}_{j,k-1}\right] \\
&\quad + \mathbf{F}'^{p}_{j,k}\left[\left(\frac{\eta_y}{J}\right)^{n+1}_{j,k} - \left(\frac{\eta_y}{J}\right)^{n+1}_{j-1,k}\right] + \mathbf{F}''^{p}_{j,k}\left[\left(\frac{\zeta_y}{J}\right)^{n+1}_{j,k} - \left(\frac{\zeta_y}{J}\right)^{n+1}_{j,k-1}\right] \\
&\quad + \mathbf{G}'^{p}_{j,k}\left[\left(\frac{\eta_z}{J}\right)^{n+1}_{j,k} - \left(\frac{\eta_z}{J}\right)^{n+1}_{j-1,k}\right] + \mathbf{G}''^{p}_{j,k}\left[\left(\frac{\zeta_z}{J}\right)^{n+1}_{j,k} - \left(\frac{\zeta_z}{J}\right)^{n+1}_{j,k-1}\right] \\
\mathbf{E}' &= \mathbf{E}_i - \mathbf{E}'_v \qquad \mathbf{E}'' = \mathbf{E}_i - \mathbf{E}''_v \\
\mathbf{F}' &= \mathbf{F}_i - \mathbf{F}'_v \qquad \mathbf{F}'' = \mathbf{F}_i - \mathbf{F}''_v \\
\mathbf{G}' &= \mathbf{G}_i - \mathbf{G}'_v \qquad \mathbf{G}'' = \mathbf{G}_i - \mathbf{G}''_v
\end{aligned}
\right\} \tag{3.29}
$$

3.8. Implementing Boundary Conditions

Boundary conditions at the wall are implemented by applying the no-slip condition at the wall to the velocity components, applying the constant wall temperature condition, and satisfying a normal momentum equation solved for pressure at the wall. The density at the wall is determined from the wall temperature, pressure, and equation of state. The discrete equation for the wall pressure was established by applying the no-slip velocity condition to the y- and z-momentum equations where η is the direction that intersects the wall. The y- and z-momentum equations

evaluated at the wall are

$$\left.\begin{aligned} \eta_y p_\eta + \zeta_y p_\zeta &= \eta_x(\tau_{xy})_\eta + \eta_y(\tau_{yy})_\eta + \eta_z(\tau_{zy})_\eta \\ &\quad + \zeta_x(\tau_{xy})_\zeta + \zeta_y(\tau_{yy})_\zeta + \zeta_z(\tau_{zy})_\zeta \\ \eta_z p_\eta + \zeta_z p_\zeta &= \eta_x(\tau_{xz})_\eta + \eta_y(\tau_{yz})_\eta + \eta_z(\tau_{zz})_\eta \\ &\quad + \zeta_x(\tau_{xz})_\zeta + \zeta_y(\tau_{yz})_\zeta + \zeta_z(\tau_{zz})_\zeta \end{aligned}\right\} \tag{3.30}$$

The derivative p_ζ is eliminated between the two momentum equations:

$$\begin{aligned} p_\eta = &\left[\frac{2}{3}\frac{\mu}{\mathrm{Re}_\infty}\left(2\overline{V}_\eta - \overline{W}_\zeta\right)\right]_\eta \\ &+ \left[\frac{\mu}{\mathrm{Re}_\infty}\left(-\overline{V}_\eta + \frac{\eta_x^2+\eta_y^2+\eta_z^2}{\zeta_z\eta_y - \zeta_y\eta_z}\overline{\vartheta}_\eta + \frac{\eta_x\zeta_x+\eta_y\zeta_y+\eta_z\zeta_z}{\zeta_z\eta_y - \zeta_y\eta_z}\overline{\vartheta}_\zeta\right)\right]_\eta \\ &+ \left[\frac{\mu}{\mathrm{Re}_\infty}\left(\overline{W}_\eta + \frac{\zeta_x^2+\zeta_y^2+\zeta_z^2}{\zeta_z\eta_y - \zeta_y\eta_z}\overline{\vartheta}_\zeta + \frac{\eta_x\zeta_x+\eta_y\zeta_y+\eta_z\zeta_z}{\zeta_z\eta_y - \zeta_y\eta_z}\overline{\vartheta}_\eta\right)\right]_\zeta \end{aligned} \tag{3.31}$$

where

$$\overline{V} = \eta_x u + \eta_y v + \eta_z w,\ \ \overline{W} = \zeta_x u + \zeta_y v + \zeta_z w,\ \ \overline{\vartheta} = \zeta_z v - \zeta_y w$$

A second-order, finite-difference module for equation (3.31) is solved for the wall pressure. The above formulation differs from the more approximate method obtained by neglecting the stress derivatives. Including the stress terms in equation (3.31) eliminated severe oscillations of the surface pressure experienced at the tip of delta wings in high Mach number flows.

3.9. Selecting a Marching Step Size

The allowable step size is calculated from a combination of the maximum allowable inviscid and viscous step sizes for a linear system. The inviscid upwind algorithm has a Courant-Friedrichs-Lewy (CFL) linear-stability limit of 2, and a viscous limit of ½ for two-dimensional problems and ¼ for three-dimensional problems. If the distance between points in the η- and ζ-directions is denoted as Δr and Δs, respectively, a marching step size Δx can be calculated by neglecting the crossflow velocities and assuming that the first point off the wall is subsonic:

$$\Delta x \le \frac{\Delta d}{\dfrac{\Delta d}{2\min(\Delta r,\ \Delta s)M_\xi\sqrt{\gamma(1-\sigma)}} + \dfrac{2\mu}{\rho u\,\Delta d\,\mathrm{Re}_\infty}} \tag{3.32}$$

where

$$\Delta d = \frac{\Delta r\,\Delta s}{\sqrt{\Delta r^2 + \Delta s^2}}$$

Equation (3.32) is usually applied with a safety factor of about 0.90 for simple geometric shapes. Note that this equation gives a marching step size that is twice that of MacCormack's scheme. The starting step size should be calculated by replacing the local axial Mach number M_ξ with one slightly larger than the desired M_ξ for one point off the surface.

3.10. Three-Dimensional, Upwind-Biased Method

The PNS equations are integrated using space marching procedures. This allows a two-dimensional flow problem to be solved with a one-dimensional numerical method, and a three-dimensional flow problem with a two-dimensional numerical method. The upwind

flux approximations for the pressure and convection terms are formed from one-dimensional numerical operators. The one-dimensional numerical operators are determined analytically by solving a RP. The Riemann solution is for an approximate form of the two-dimensional inviscid PNS equations which locally models the inviscid flow between two points. Unfortunately, there is not a two-dimensional analytical solution of steady three-dimensional flow which can be used in determining an appropriate two-dimensional numerical operator for approximating the numerical fluxes. In practice, the numerical fluxes for a multidimensional numerical problem are usually determined by a sequence of one-dimensional numerical operators. Algorithms for the three-dimensional PNS equations require two different flux approximations in the crossflow plane. The fully upwind algorithm for the three-dimensional PNS equations presented earlier is based on two one-dimensional upwind operators determined from the Riemann solution for two-dimensional steady flow. The combination of two sets of RPs, one for each coordinate direction in the crossflow plane, does not always correctly model the physics of a three-dimensional flow field. The purpose of this section is to investigate alternate ways of approximating the convection fluxes in the crossflow plane to model the physics of the flow field more accurately.

Upwind flux approximations constructed from analytical solutions of RP's are used because of their excellent shock-capturing capability. For each integration step of a one-dimensional upwind numerical method, a shock wave is captured between only two points. This represents a point discontinuity along a line of data. For three-dimensional steady flow, the shock wave represents a discontinuous curve in the crossflow plane. The use of one-dimensional analytical solutions to determine upwind flux approximations for multidimensional schemes can result in difficulties associated with the discretization of a continuous problem containing a shock wave. When the shock wave is mapped onto a discrete grid for the crossflow plane (fig. 3.4), the discontinuity cuts across data cells unless the shock aligns with the numerical grid. The fully upwind algorithm treats the discontinuous data as jumps and attempts to model a shock at each cell interface in all directions. In this case, the upwind flux approximation does not represent the physics as clearly as in the two-dimensional case. When the crossflow velocity tangent to a shock wave is near zero, the numerical dissipation of the scheme in this direction is small. The numerical dissipation that is inherent in the numerical method depends on the magnitude of the eigenvalues. Three out of the five eigenvalues are directly related to the magnitude of the crossflow velocities. Significant errors can occur when the velocity tangent to a shock is near zero and a slight difference occurs between adjacent cells next to the captured shock (fig. 3.5). The numerical results may drift away from the correct solution when a shock wave is not perfectly aligned with the numerical grid in areas where the numerical scheme has little natural numerical dissipation. As a shock wave moves across a numerical grid, errors are made as the discontinuity jumps across finite areas. If the velocity tangent to a strong shock is nearly zero, small disturbances can occur in the tangential velocity that transports large amounts of conserved quantities (Colella and Woodward 1984). These errors are minimized in regions where there is a large enough tangential velocity providing enough dissipation from the numerical method to ensure the correct solution. This problem has been handled by adding dissipation to the numerical method using a number of different devices. A commonly used procedure is a device originally proposed by Harten (1983) for eliminating expansion shocks. Harten's device maintains the absolute value of the eigenvalues above a minimum value to guarantee a minimum amount of numerical dissipation in all flow regions. Specifying the correct amount is difficult when the viscous terms are present since the convection eigenvalues are near zero in the boundary layer. Colella and Woodward (1984) developed a number of different methods for eliminating the multidimensional oscillations with their RP solver for the Euler equations.

Unfortunately, each one of these methods uses coefficients that require tuning by the user. The main reason for using an upwind scheme was to eliminate the need to use additional dissipation that would depend on user-specified parameters.

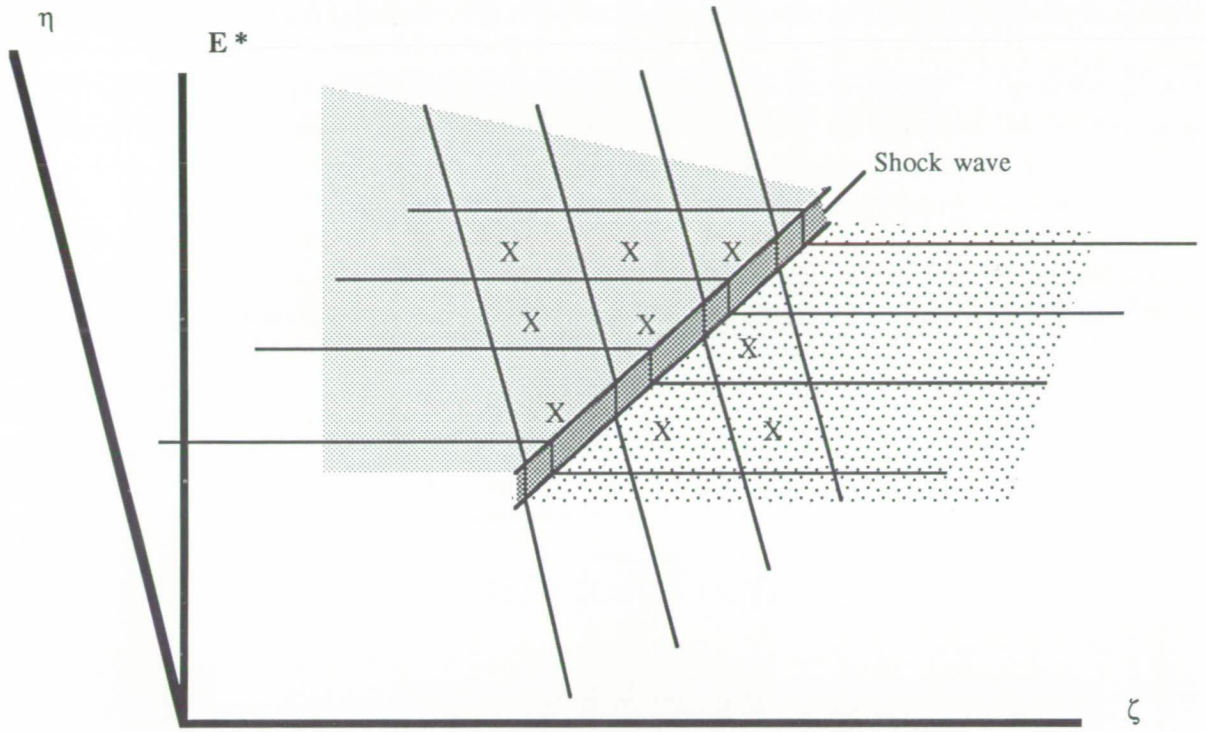


Figure 3.4. Mapping of a shock wave in the crossflow plane.

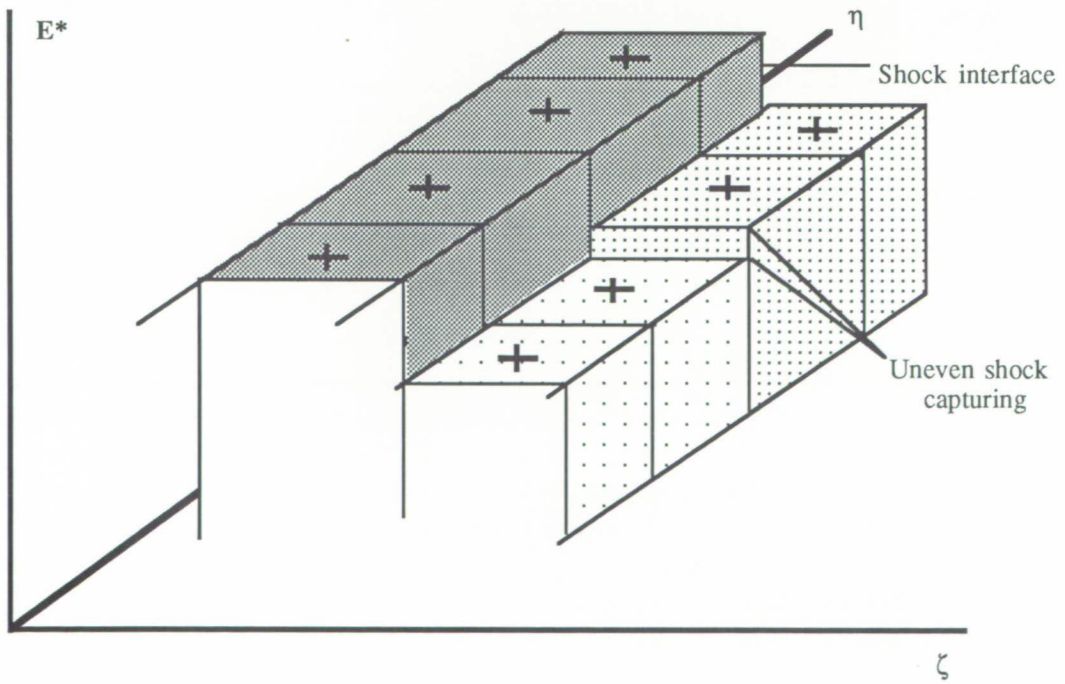


Figure 3.5. Uneven shock capturing in the crossflow plane.

An alternate approach to adding dissipation explicitly is to use a numerical method that models the physics of the flow more accurately. A two-dimensional numerical scheme is defined

as a scheme which selects different flux approximations based on some physical characteristic of the flow field that does not occur in the one-dimensional scheme. Murman and Cole (1971) and Jameson (1974) used two-dimensional schemes for integrating the potential equation in regions of transonic flow. Murman introduced the idea of equation-type-dependent differencing with the use of upwind differencing in hyperbolic regions and central differencing in elliptic regions. Jameson (1974) applied a rotated differencing scheme to the potential equation that used upwind differencing only in the local stream direction when the equation set is hyperbolic (for supersonic flow) and central differencing elsewhere. Davis (1984) developed a rotational upwind-biased difference scheme for use with Euler's equations. The first step of Davis' scheme was to determine the direction in which a shock wave (i.e., RP) would be oriented if it existed in the finite-difference stencil (fig. 3.6). An upwind algorithm would then be used in a local coordinate system rotated to be normal to the orientation of the assumed shock wave. Roe (1986) proposed a two-dimensional RP algorithm which has not been implemented. Hirsch, Lacor, and Deconinck (1987) have investigated a method in which the two-dimensional unsteady form of Euler's equations can be diagonalized for special flow conditions.

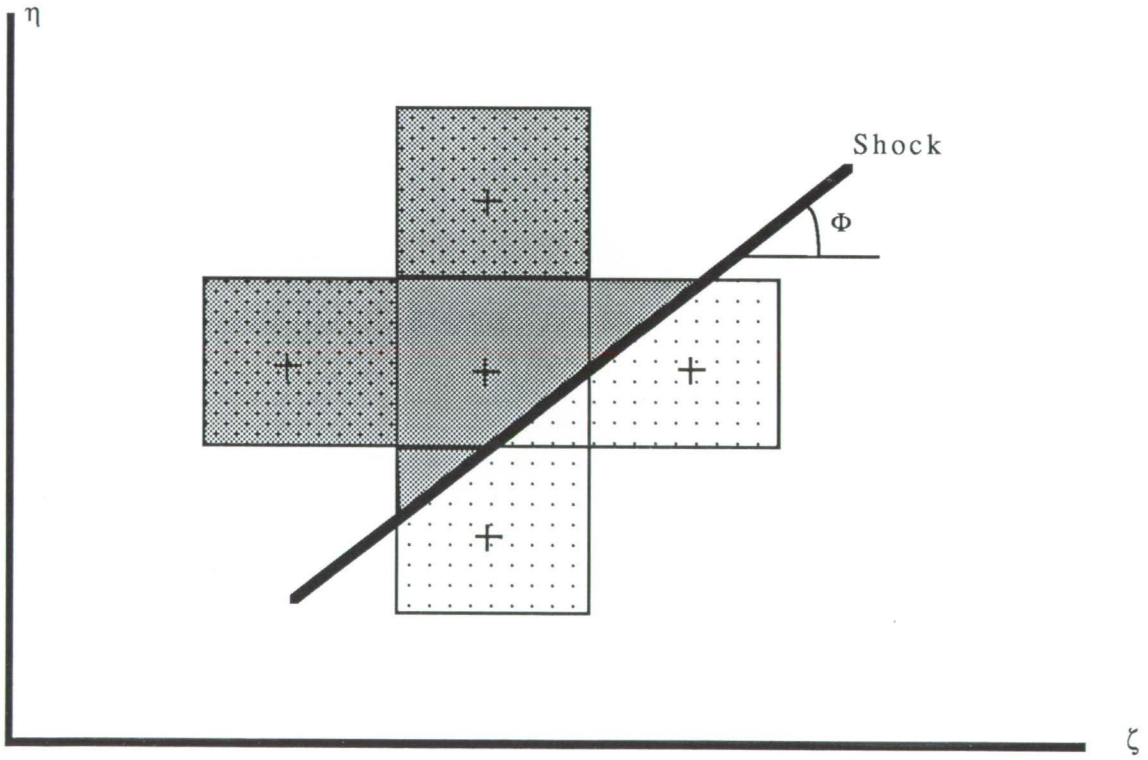


Figure 3.6. Shock angle in the computational plane.

The two-dimensional problem described previously has to do with the mapping of a shock wave as a discontinuous curve. Riemann methods are based on the assumption that the data are piecewise continuous. This assumption is correct in the direction normal to the shock wave. The data in the tangential direction to the shock wave should be continuous without jumps. Davis' (1984) scheme is based on this observation. Recall that the fully upwind scheme described in section 3.6 becomes a central difference scheme based on MacCormack's (1969) method when there is no jump in the initial data used in solving the RP's. A combination method based on Davis' ideas can be incorporated by using the solution of the RP only in the directions normal to an assumed shock wave position. In this case the flux approximation at an interface now depends on the angle of an assumed shock wave. Since the RP is solved for an interface in the

computational plane, the angle that the assumed shock wave would make at this interface has to be determined. The assumed shock angle can be determined using geometry by recognizing that there is no velocity jump in the direction tangential to a shock wave (Davis 1984). A quadratic equation for the shock angle derived from the geometry in the computational plane is

$$\overline{V}_\zeta \tan^2\Phi - \tan\Phi\left(\overline{W}_\zeta - \overline{V}_\eta\right) - \overline{W}_\eta = 0 \tag{3.33}$$

where $\overline{V}$ and $\overline{W}$ are defined as for equation (3.31). Equation (3.33) differs from Davis' formulation of the shock angle since it is not based on an approximation. Once the angle of the shock in the computational plane at the interface is known, then the FDS terms can be modified by

$$\left.\begin{aligned} \mathbf{df}^{+}_{j+\frac{1}{2}} &= \delta_{j+\frac{1}{2}} \sum_{m=1}^{5} \frac{\lambda_m + |\lambda_m|}{2} \left(\alpha_m \hat{\mathbf{e}}_m\right)_{j+\frac{1}{2}} \\ \mathbf{df}^{-}_{j+\frac{1}{2}} &= \delta_{j+\frac{1}{2}} \sum_{m=1}^{5} \frac{\lambda_m - |\lambda_m|}{2} \left(\alpha_m \hat{\mathbf{e}}_m\right)_{j+\frac{1}{2}} \end{aligned}\right\} \tag{3.34}$$

The parameter δ can vary from 0 to 1. For the interface location between points in the η-direction, $\delta = |\cos\Phi|$. For the interface location between points in the ζ-direction, $\delta = |\sin\Phi|$.

To summarize, first the RP's are solved in the computational coordinate system as if fully upwind differencing is to be used for the pressure and convection terms. Second the angle at which a shock wave would occur at the interface is calculated from equation (3.33). Finally the FDS expression is multiplied by either the sine or the cosine of the shock angle for the interface (eq. (3.34)). Following this procedure, an upwind differencing stencil is used for the pressure and convection terms in the direction normal to a shock wave and MacCormack's differencing stencil is used in the direction tangential to a shock wave.

4. Two-Dimensional Results

The explicit upwind algorithm (section 3.6) is used to solve the parabolized Navier-Stokes (PNS) equations for three two-dimensional supersonic and hypersonic viscous flows. The explicit upwind PNS solver applied to two-dimensional flow problems has the advantage that the numerical problem is one-dimensional. The Riemann problem (RP) used in this study models the convection terms of the two-dimensional PNS equations. The successful simulation of the two-dimensional test cases validates the upwind flux approximations determined from the solution of the RP. Solutions are calculated for laminar flow over a flat plate, across a ramp, and through an inlet. The calculations were done on both a Gould minicomputer and the Numerical Aerodynamic Simulation (NAS) Cray 2 supercomputer. The calculations were repeated on the Cray 2 to obtain execution times for the vectorized code.

The numerical grids were generated at each new station using algebraic stretching functions to cluster the spacing of the points near the wall. The stretching transformation can be found in the text of Anderson, Tannehill, and Pletcher (1984):

$$r(j) = r(1) + [r(n_j) - r(1)]\, h\left(\frac{n_j - j}{n_j - 1}\right) \qquad (4.1)$$

where

$$h(\Psi) = 1 - \beta + \frac{2\beta}{1 + \left(\frac{\beta+1}{\beta-1}\right)^{\Psi}}$$

The distance away from the wall ($j = 1$) is denoted by r, and n_j denotes the total number of points. This transformation clusters points around $j = 1$ when the values of the stretching parameter β are close to 1.

4.1. Supersonic Laminar Flow Over a Flat Plate

The first case is for supersonic laminar flow over a flat plate. The conditions are

$M_\infty = 2.0$ $\quad \mathrm{Re}_L = 1.65 \times 10^6$ $\quad L = 1.0$ m
$\mathrm{Pr} = 0.72$ $\quad T_\infty = T_w = 222$ K $\quad \sigma = 0.80$

where T_w is the wall temperature and σ is the safety factor introduced in equation (2.12). The supersonic laminar flow field for a flat plate at Mach 2.0 is comprised of a weak leading edge shock and a Blasius boundary layer profile (fig. 4.1). The correct prediction of both the velocity and the temperature profile verifies that the numerical scheme accurately captures the weak shock and resolves the viscous and heat fluxes. The use of an explicit scheme for this case requires a large number of marching steps because of the small normal grid spacing required to represent the Mach 2.0 boundary layer. The results of the upwind scheme at the final exit station show that the numerical dissipation inherent in the upwind differencing of the convection term does not adversely affect the resolution of the viscous terms.

The computational domain was sized to include the leading edge shock. The vertical height of the grid was initially set to $y = 0.06$ and was made to grow linearly to a maximum height of $y = 0.525$ at $x = 0.93$. The computational grid consisted of 104 points in the η-plane. The code was started using free-stream initial plane conditions with an initial maximum marching step size of $\Delta x = 0.00015$ until $x = 0.025$. Once this point had been reached, the step size was varied based upon the result of equation (3.32). The grid stretching parameter was initially set to 1.01 until after startup was completed and then varied to maintain the first point off the surface at approximately $M = 0.15$. The number of points for the η-plane was determined to ensure having at least 10 points in the axial boundary layer profile at $x = 0.93$.

The results of the explicit upwind code are compared at $x = 0.93$ with the results of Lawrence, Tannehill, and Chaussee (1986) who used an implicit PNS code (Beam and Warming 1978). The temperature and axial velocity profiles at $x = 0.93$ are shown in figures 4.2 and 4.3. The Beam-Warming code was started using the results from a boundary layer code at $x = 0.3$ and used approximately twice the number of equally spaced points in the boundary layer (only half are shown). The boundary layer profiles compare favorably even with the different initial condition and stretched grid used with the explicit upwind code.

This stretched grid was scaled to the boundary layer growth to maintain the same Mach number at the first point off the wall. In so doing, the scaled grid minimized the effect of the source term Ω_ξ in equation (2.14). To illustrate the effect of Ω_ξ, two additional runs were made on a second grid with a fixed point spacing with the same normal point distribution as at the final station of the original scaled grid. On the second fixed grid, the source term Ω_ξ is significant because of the streamwise change of each grid point's Mach number as the boundary layer grows. Figure 4.4 shows the effect on the temperature profile of including and neglecting Ω_ξ in the numerical scheme when obtaining a solution on the two fixed grids. The error of not including Ω_ξ is clearly seen in the drop in the maximum temperature.

The subroutine used to calculate the flux-difference splitting (FDS) required 49 percent of the

central processing unit (CPU) time. The algorithm required 0.216×10^{-4} sec per longitudinal step per normal grid point on the Cray 2 computer.

4.2. Hypersonic Laminar Flow Over a Compression Ramp

The second test case is the simulation of two-dimensional hypersonic laminar flow across a flat plate and over a 15° ramp. The conditions are

$$\begin{array}{lll} M_\infty = 14.1 & \mathrm{Re}_L = 1.04 \times 10^5 & L = 0.439 \text{ m} \\ T_\infty = 72.2\text{K} & T_w = 297.0 \text{ K} & \mathrm{Pr} = 0.72 \\ \sigma = 0.75 & & \end{array}$$

where the characteristic length, on which the Reynolds number is based, is the length of the flat plate. The 15° ramp is one of many ramps studied experimentally at the above free-stream conditions by Holden and Moselle (1970). The distinguishing features of this case are the interaction between the viscous and inviscid flows and the lack of evidence of any separation at the base of the ramp in the experimental data. The high Mach number flow on the flat plate results in the pressure profile normal to the surface being curved in the boundary layer. The inviscid flow structure consists of a leading edge shock which intersects the induced shock formed by the ramp. The two right-running shocks intersect to form a single stronger shock, an expansion fan, and a contact surface (fig. 4.5).

The computational grid consisted of 45 points in the η-direction. The top of the grid was defined by

$$y_{\text{top}} = \begin{cases} 0.135 + 0.104x & (\text{For } x < 1.65) \\ 0.3066 + \tan(15°)x & (\text{For } x > 1.65) \end{cases}$$

The grid was stretched in the normal direction using a constant stretching factor of 1.04. The law-of-the-wall coordinate (y^+) for the first point away from the wall varied between 0.87 and 0.33 from the leading edge of the flat plate to the corner of the ramp and reached a maximum of 2.7 on the ramp. The code was started using free-stream initial conditions and a step size of 0.0005. The step size was reduced when the result of equation (3.32) was smaller than 0.0005.

The results of the explicit upwind code are compared with the experimental results of Holden and Moselle (1970) and the numerical results of Hung and MacCormack (1976) for the Navier-Stokes (NS) equations. The results for surface pressure (c_p), heat transfer (c_h), and skin friction (c_f) coefficients are shown in figures 4.6, 4.7, and 4.8. The coefficients are defined as

$$\left.\begin{aligned} c_p &= \frac{p}{\rho_\infty U_\infty^2} \\ c_h &= \frac{\mu_w \left.\frac{\partial T}{\partial n}\right|_w}{\mathrm{Re}_L \mathrm{Pr}\left(\frac{\gamma-1}{2} M_\infty^2 + 1 - T_w\right)} \\ c_f &= \frac{\mu_w}{\mathrm{Re}_L} \left.\frac{\partial u}{\partial n}\right|_w \end{aligned}\right\} \qquad (4.2)$$

where

$$\frac{\partial(\ )}{\partial n} = \sec(\theta)\eta_y \frac{\partial(\ )}{\partial \eta}$$

and subscript w indicates values at the wall, n represents the surface normal direction, and θ represents the angle which the η-coordinate makes with the surface normal direction. A slight overprediction of the experimental data can be observed that is consistent with numerical results using other methods (Gielda and McRae 1986; Lawrence, Tannehill, and Chaussee 1986). Lawrence et al. attributed this difference to a flow misalignment with the ramp. The explicit upwind results agree favorably with the Navier-Stokes solution except around the beginning of the ramp. Since the PNS equations do not allow upstream propagation of information in the subsonic region, the c_p, c_h, and c_f results do not increase until the corner of the wedge is reached, while the experimental and NS results definitely increase before the corner. The comparison of c_f in figure 4.8 shows that the flow was very close to separating at the corner of the ramp. The excellent agreement of the explicit upwind c_f with the Navier-Stokes results before and after the ramp corner was attributed to the addition of the source term Ω_ξ. Recall that Ω_ξ was added to cancel the variation of ω in the streamwise direction (eq. (2.14)).

The interaction of the leading edge shock and the wedge shock is shown in the Mach number contour (fig. 4.9) and the pressure contour (fig. 4.10) from the explicit upwind algorithm. The intersections of the leading edge shock and the wedge shock, the new combination shock, and the expansion fan are sharply defined in the pressure contour. The contact surface can be seen immediately after the shock intersection in the Mach contour. Notice that the pressure contour is oscillation free near the wall. This is in contrast to the implicit application by Lawrence, Tannehill, and Chaussee (1986), which showed pressure oscillations near the wall. They attributed the oscillations to the switch from their upwind scheme in the supersonic region to a central difference algorithm in the subsonic region. When the above

switch was implemented in the explicit upwind algorithm, the stability and accuracy of the scheme deteriorated dramatically. Therefore, it is important to use the same differencing scheme throughout the subsonic region.

4.3. Hypersonic Laminar Flow Through an Inlet

The final two-dimensional test case is for a complicated hypersonic flow field in a converging inlet. The geometry of the inlet is given in figure 4.11. The free-stream conditions are

$$M_\infty = 15 \qquad \mathrm{Re}_L = 8.0 \times 10^4 \qquad L = 0.4 \text{ m}$$
$$T_\infty = 100 \text{ K} \qquad T_w = 1000 \text{ K} \qquad \mathrm{Pr} = 0.72$$
$$\sigma = 0.75$$

The characteristic length is the length of the flat plate at the entrance of the inlet. The NS results (Newsome, Walters, and Thomas 1987) for this inlet flow field indicate a strongly interacting flow which experiences streamwise separation at the compression and expansion corners and at the intersection of the shock waves with the wall. One-sweep PNS solvers can march through streamwise separation points without predicting separation. Therefore, a one-sweep PNS solution does not accurately represent the NS solution in areas around the separation regions. The one-sweep PNS solution can be used as initial conditions for NS solvers or for preliminary evaluation of the inviscid flow field structure. The purpose of this test case is to evaluate the performance of the upwind scheme for an inviscid flow containing a number of intersecting and reflecting shock waves. This is a numerical test case solved by Lawrence, Tannehill, and Chaussee (1986) and Newsome, Walters, and Thomas (1987). Lawrence et al. solved the PNS equations using an implicit noniterative upwind scheme. Newsome et al. solved both the PNS equations and the thin-layer Navier-Stokes equations using an implicit upwind relaxation method.

The numerical domain for the explicit upwind scheme extends in the vertical direction from the wall to the centerline of the inlet. The numerical grid consisted of 45 points in the vertical direction which included the 2 points required for the symmetry boundary condition at the inlet centerline. A constant stretching coefficient of 1.08 was used for the entire length of the inlet. The normal point distribution is identical to what was used by Newsome et al. The centerline and the wall boundaries are

$$y_{\text{top}} = 0.375$$

$$y_{\text{wall}} = \begin{cases} 0 & (x < 1) \\ (x-1)\tan(15^\circ) & (1 < x < 2) \\ \tan(15^\circ) & (x > 2) \end{cases}$$

The code was started using free-stream initial conditions and a streamwise step size of 0.0002. The step size was reduced if the result of equation (3.32) was less than 0.0002. A safety factor of 0.5 was used in the step size calculation to maintain stability when the shock waves intersected the inlet walls. This calculation required 20 220 marching steps to reach the end of the inlet ($x = 4$), using 23 seconds of CPU time on the NAS Cray 2 computer.

Pressure, heat transfer, and skin friction coefficients from the explicit upwind PNS solution are compared with Newsome et al.'s PNS and NS solutions. The pressure coefficients (fig. 4.12) and the heat transfer coefficients (fig. 4.13) agree favorably with their PNS solution and NS solution except in the regions around the flow separation. A slight difference is observed near the location of the second shock reflection in the inlet for both the pressure and the heat transfer coefficient. This difference is attributed in part to the difference in modeling the symmetry condition with the finite-difference formulation used in this study and with the finite-volume formulation used by Newsome et al. The comparison of skin friction coefficients (fig. 4.14) is favorable only for the flat plate region. The complicated shock wave structure for the inlet can be observed in the pressure contour (fig. 4.15) and the Mach number contour (fig. 4.16). The shock waves are sharply captured using the upwind scheme without pre- and post-shock oscillations. While the overall trends of the two PNS solutions are similar, part of the difference may be attributed to the more complete modeling of the viscous terms used in this study and the increased streamwise accuracy of the explicit calculation because of the large number of streamwise stations required. The previous study of Newsome et al. used 220 streamwise planes, while the explicit calculation made here used 20 220 streamwise planes. The same number of points in the vertical direction was used in both studies.

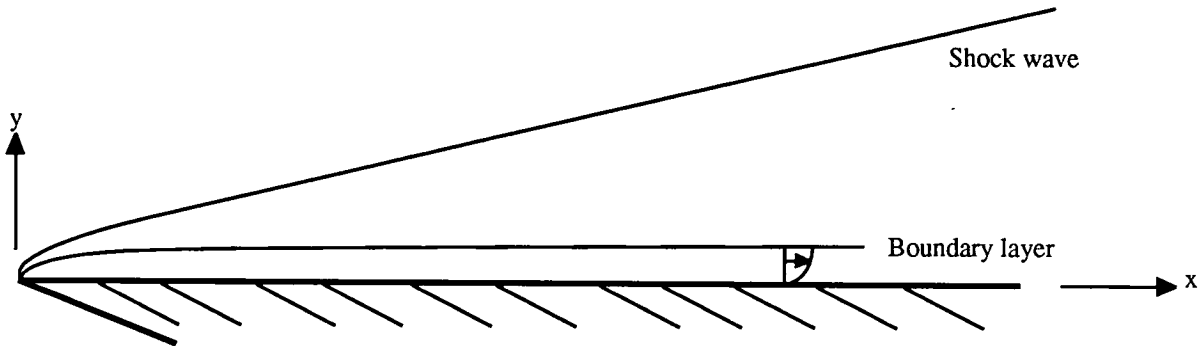


Figure 4.1. Shock and boundary layer in $M = 2.0$ laminar flow over a flat plate.

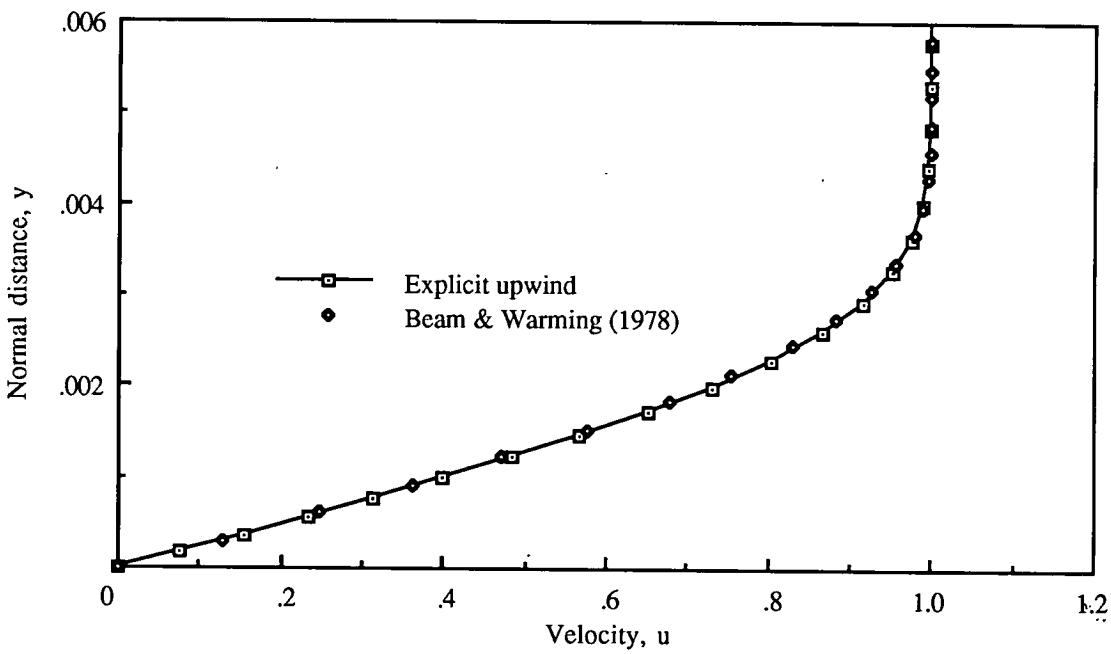


Figure 4.2. Comparison of computed velocity profiles on a flat plate. $M_\infty = 2.0$; $\mathrm{Re}_L = 1.65 \times 10^6$; $x = 0.93$.

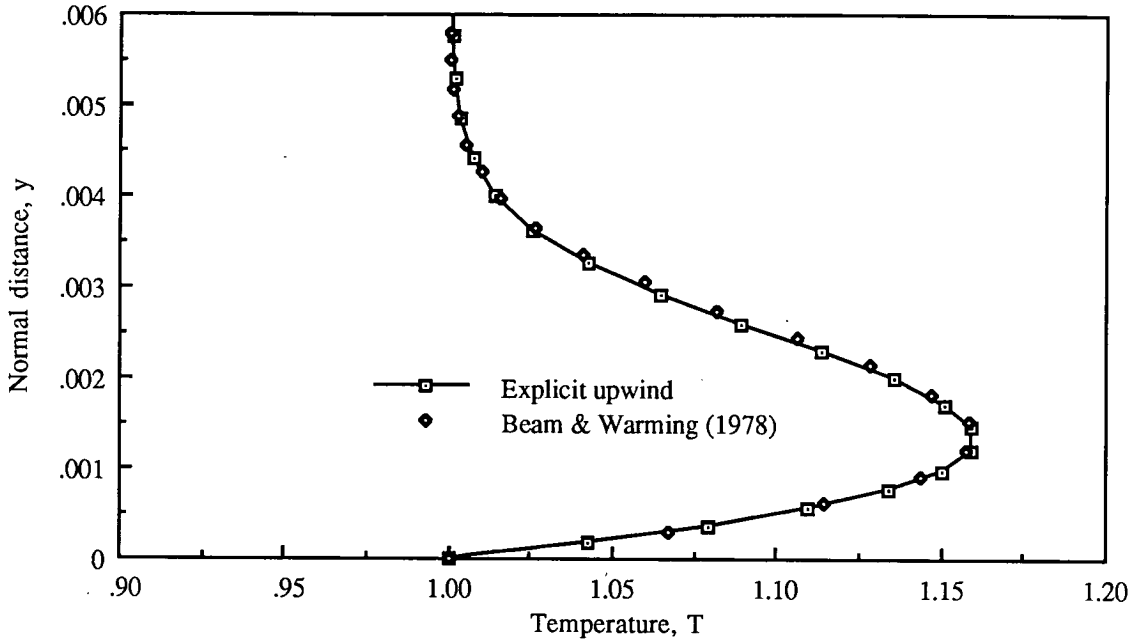


Figure 4.3. Comparison of computed temperature profiles on a flat plate. $M_\infty = 2.0$; $\mathrm{Re}_L = 1.65 \times 10^6$; $x = 0.93$.

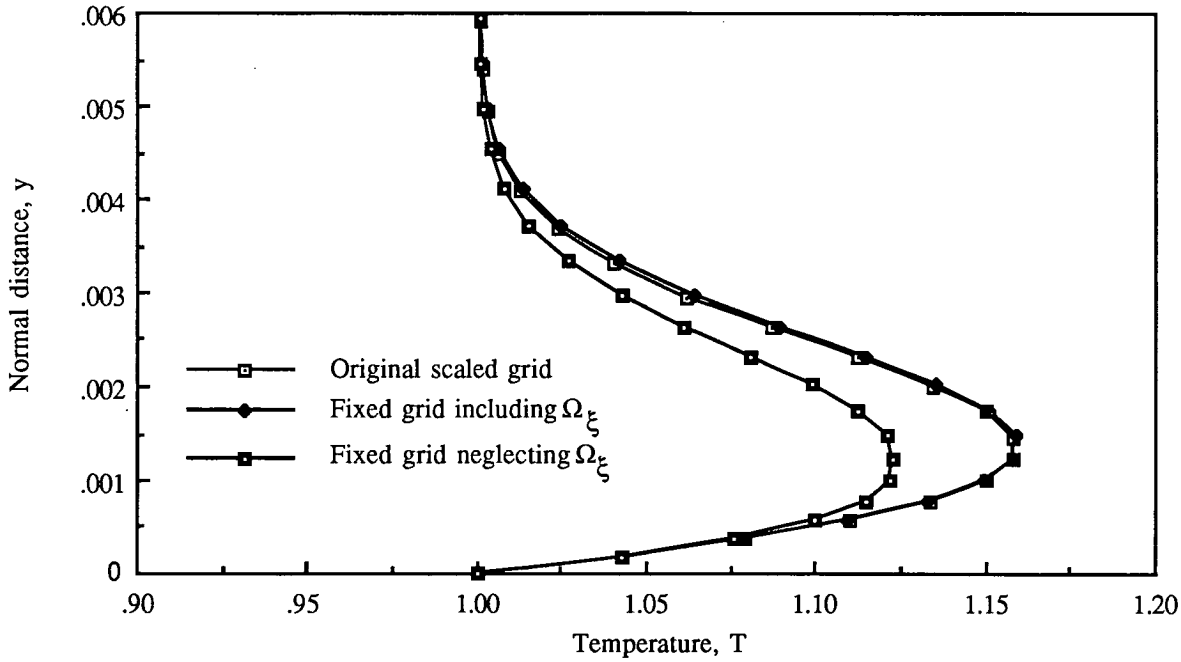


Figure 4.4. Comparison of computed temperature profiles when including and neglecting the source term Ω_ξ. $M_\infty = 2.0$; $\mathrm{Re}_L = 1.65 \times 10^6$; $x = 0.93$.

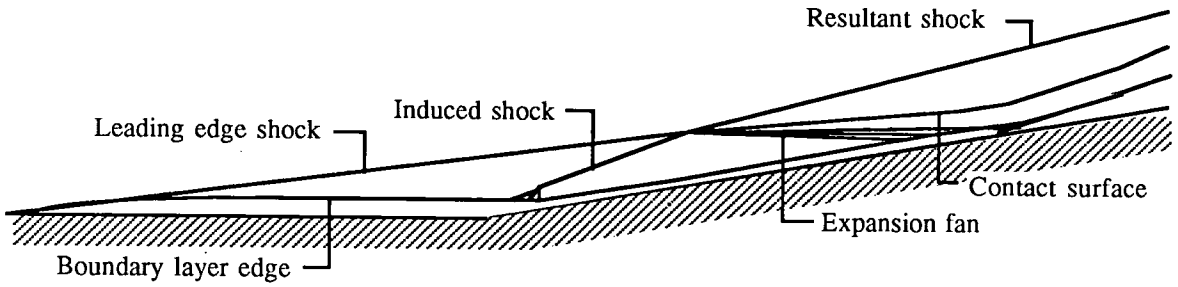


Figure 4.5. Flow field for a hypersonic compression corner.

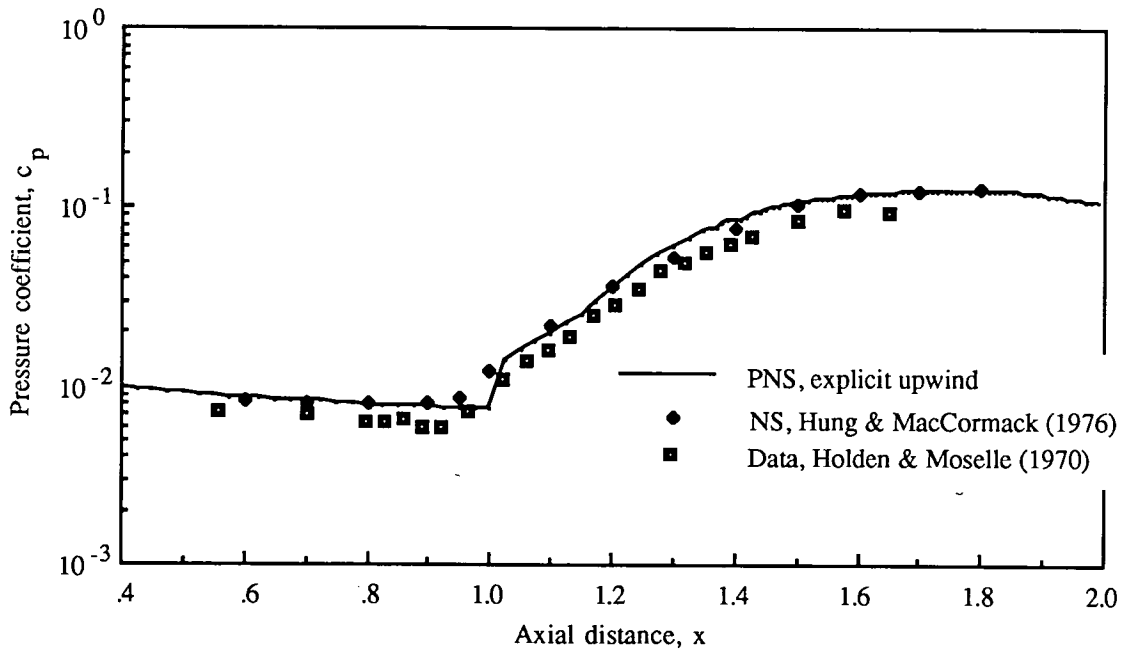


Figure 4.6. Comparison of computed pressure coefficients with experimental data. $M_\infty = 14.1$; $\mathrm{Re}_L = 1.04 \times 10^5$.

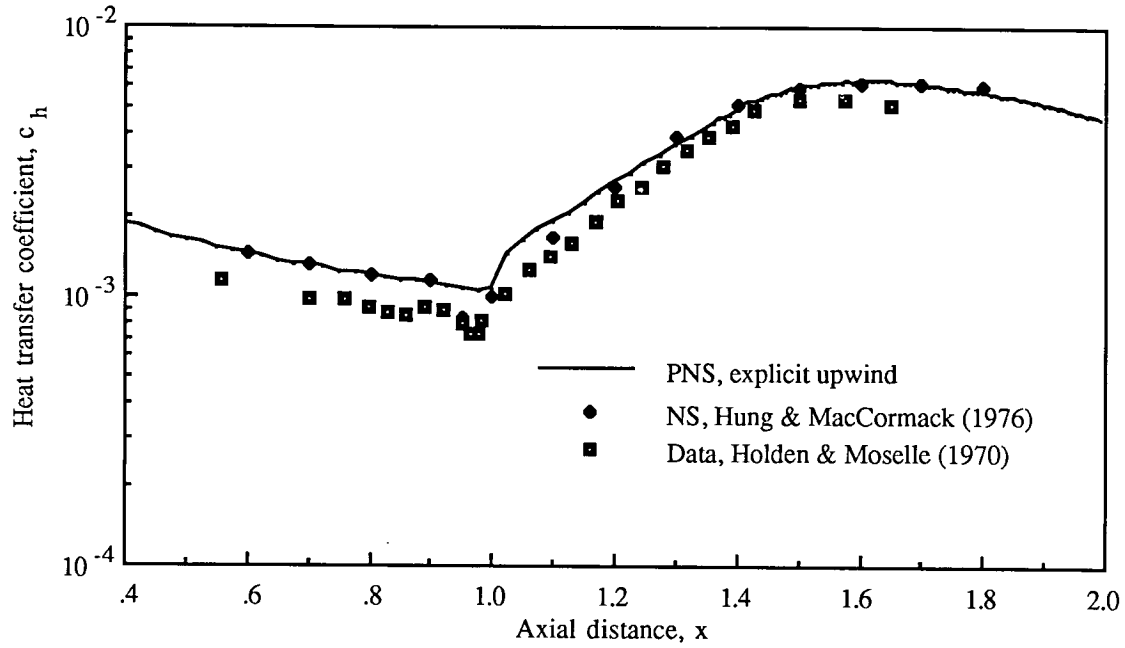


Figure 4.7. Comparison of computed heat transfer coefficients with experimental data. $M_\infty = 14.1$; $\mathrm{Re}_L = 1.04 \times 10^5$.

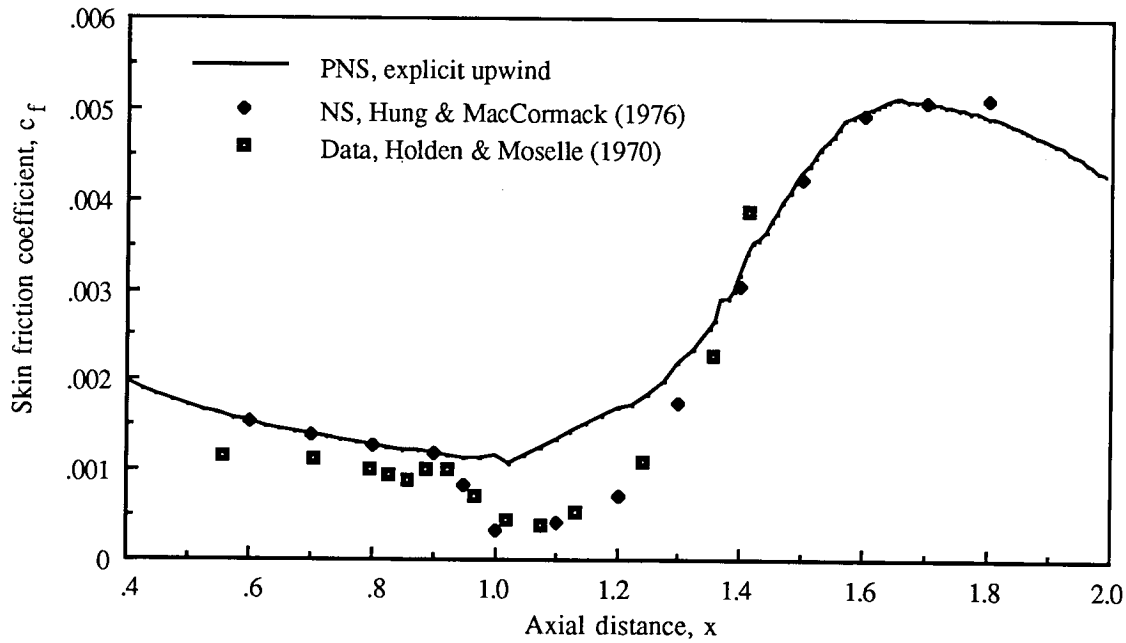


Figure 4.8. Comparison of computed skin friction coefficients with experimental data. $M_\infty = 14.1$; $\mathrm{Re}_L = 1.04 \times 10^5$.

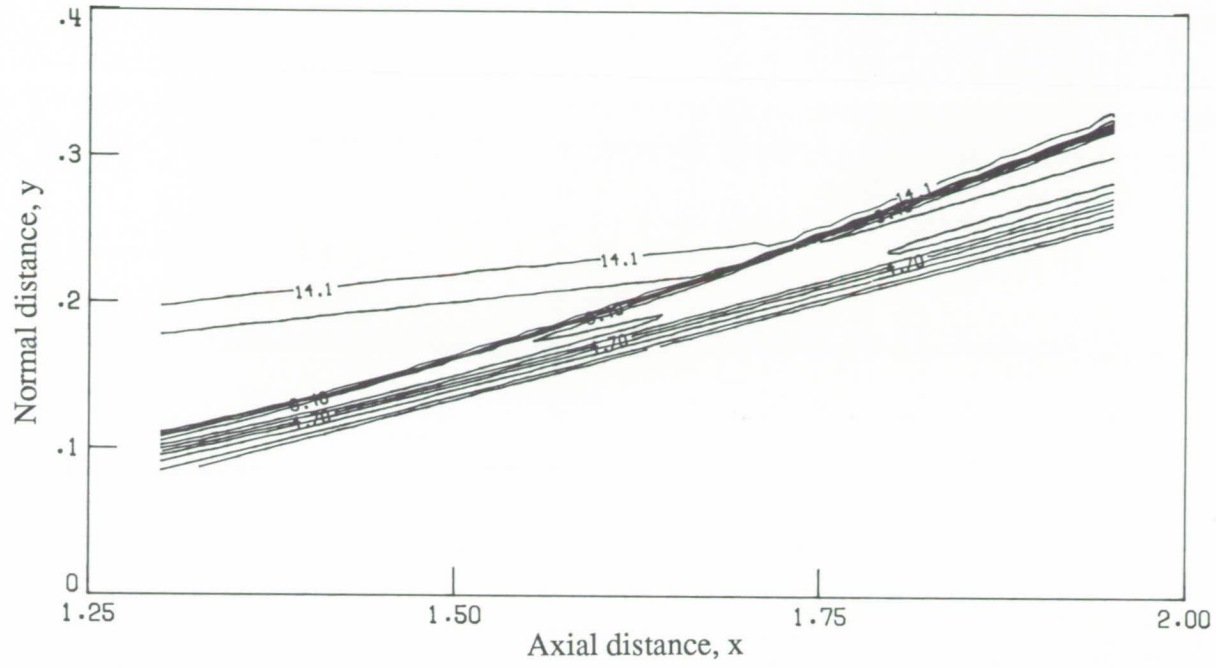


Figure 4.9. Mach number contours on the 15° ramp from explicit upwind solution. $M_\infty = 14.1$; $\mathrm{Re}_L = 1.04 \times 10^4$.

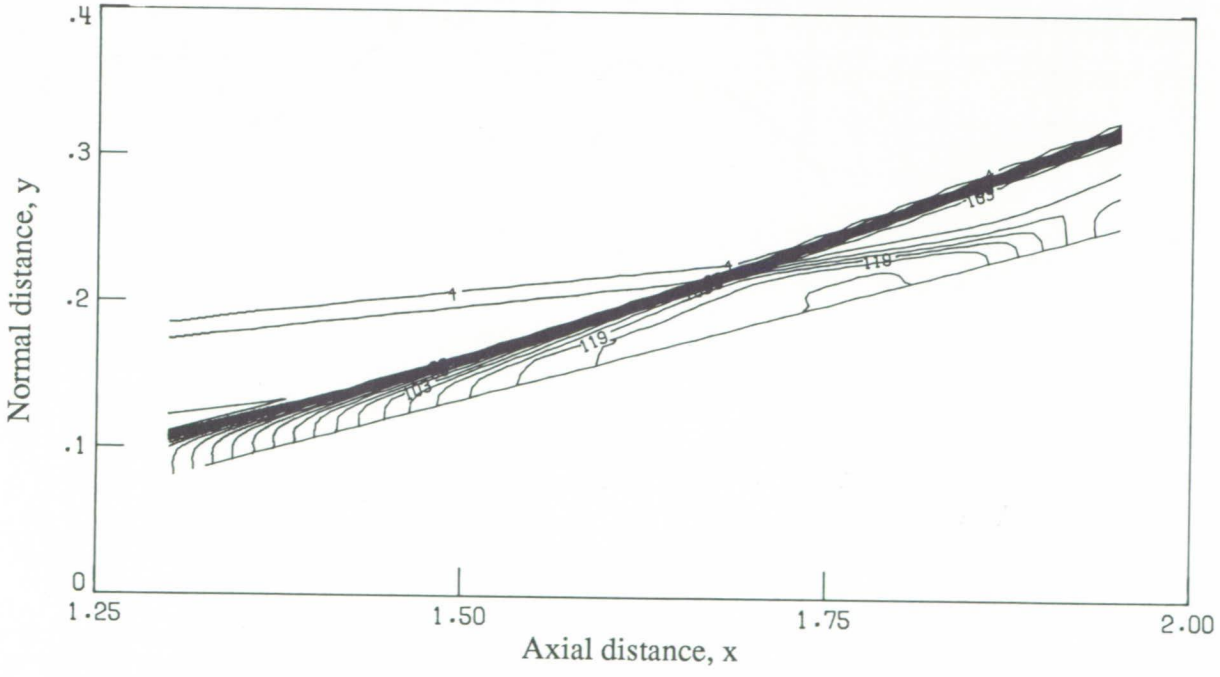


Figure 4.10. Pressure contours on the 15° ramp from explicit upwind solution. $M_\infty = 14.1$; $\mathrm{Re}_L = 1.04 \times 10^5$.

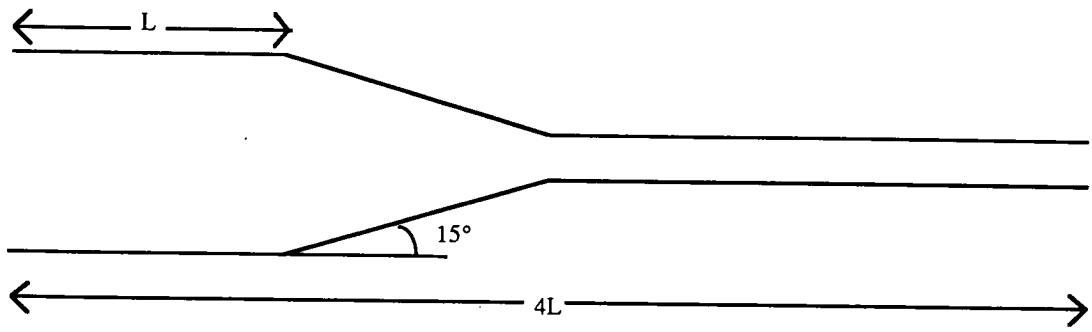


Figure 4.11. Converging inlet.

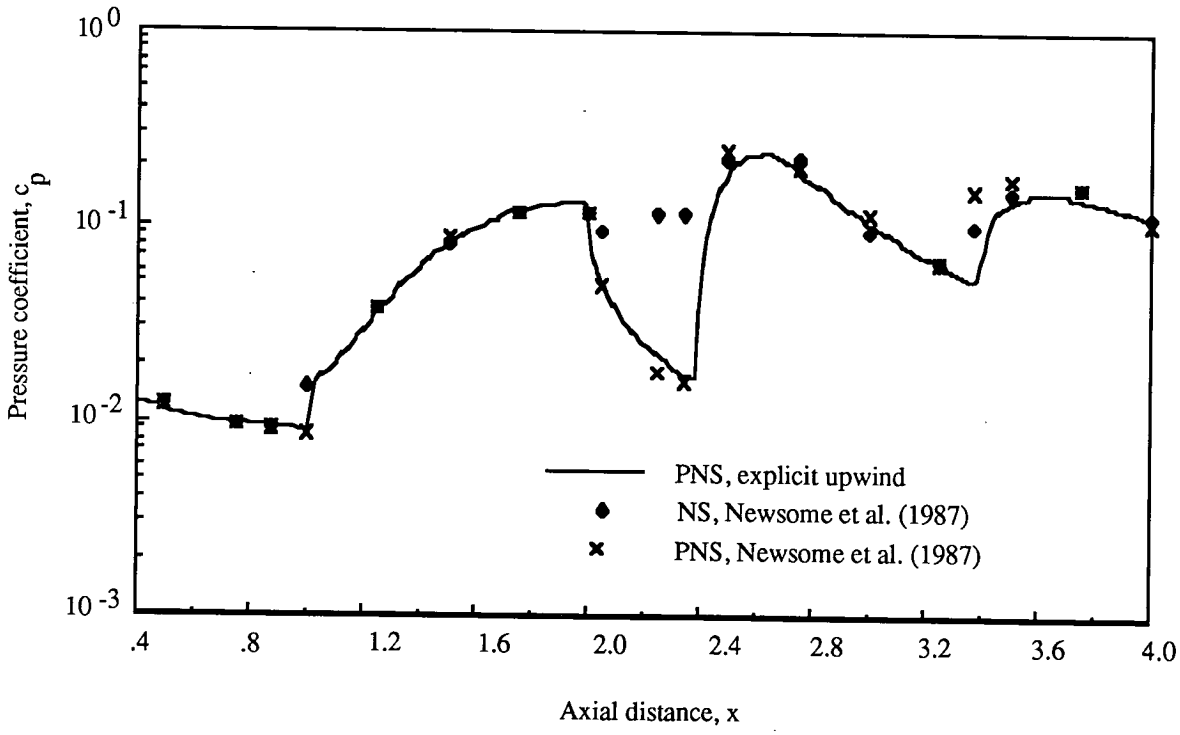


Figure 4.12. Comparison of computed pressure coefficients for a 15° hypersonic inlet. $M_\infty = 15$; $\mathrm{Re}_L = 8.0 \times 10^4$.

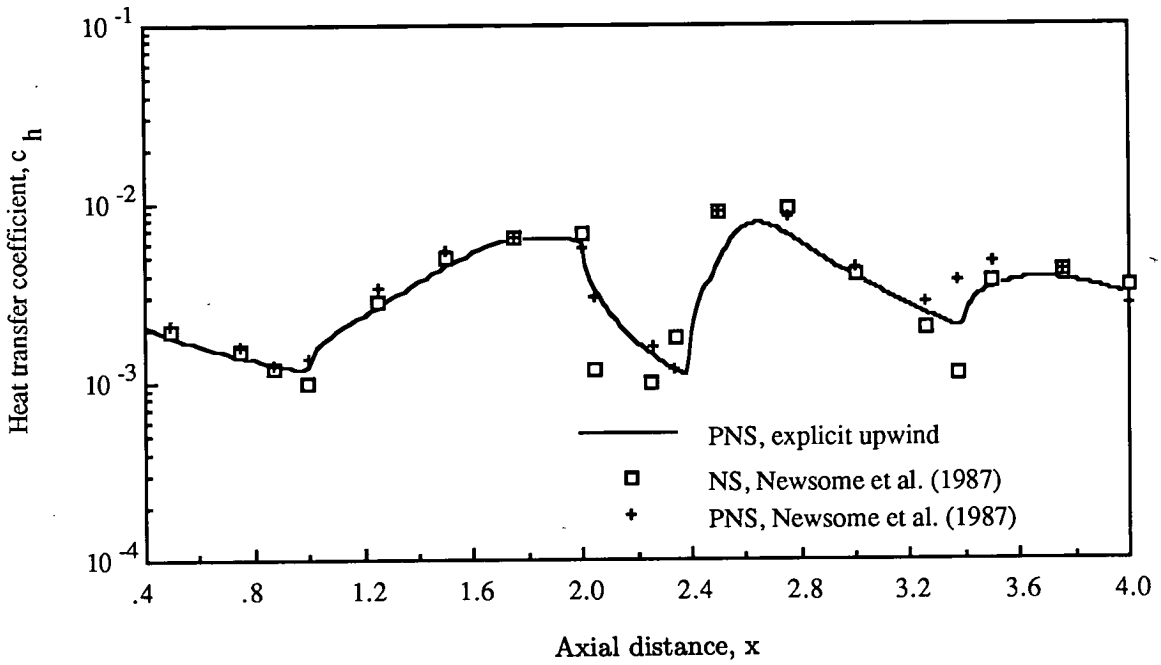


Figure 4.13. Comparison of computed heat transfer coefficients for a 15° hypersonic inlet. $M_\infty = 15$; $\mathrm{Re}_L = 8.0 \times 10^4$.

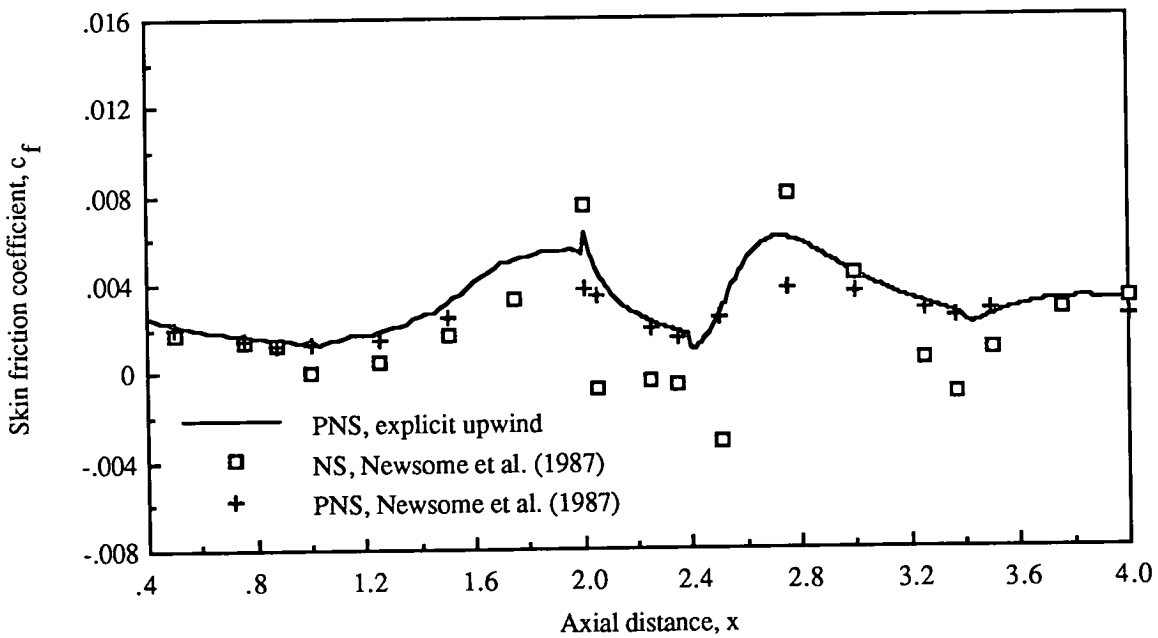


Figure 4.14. Comparison of computed skin friction coefficients for a 15° hypersonic inlet. $M_\infty = 15$; $\mathrm{Re}_L = 8.0 \times 10^4$.

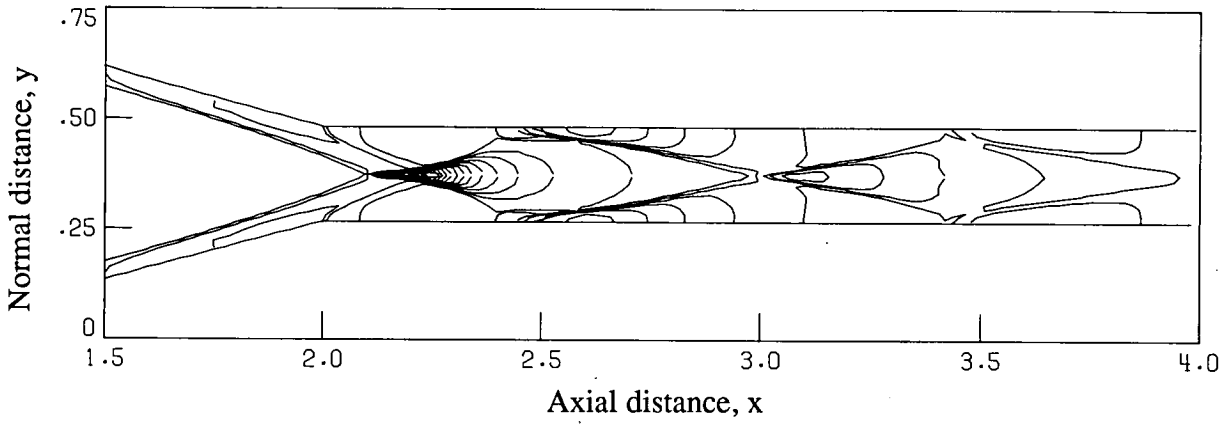


Figure 4.15. Hypersonic inlet pressure contour from explicit upwind solution. $M_\infty = 15$; $\mathrm{Re}_L = 8.0 \times 10^4$.

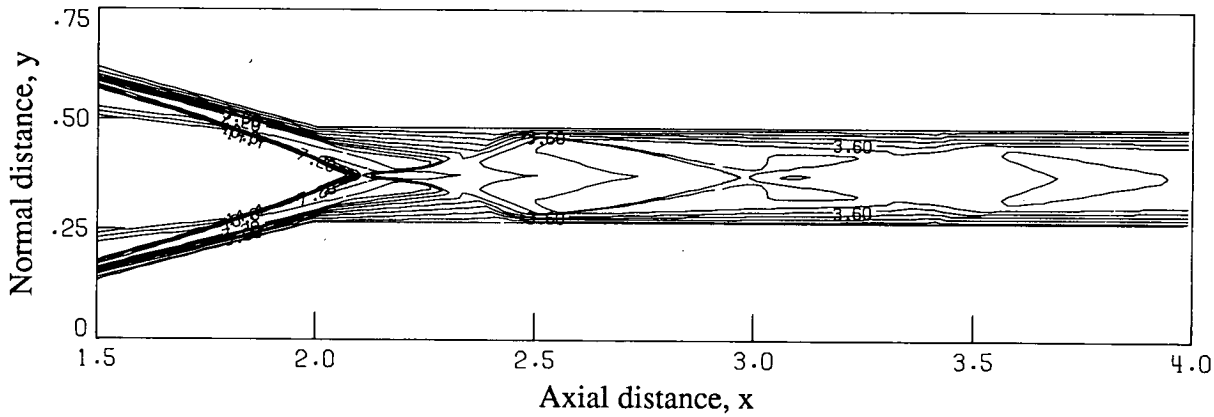


Figure 4.16. Hypersonic inlet Mach number contour from explicit upwind solution. $M_\infty = 15$; $\mathrm{Re}_L = 8.0 \times 10^4$.

5. Three-Dimensional Results

Two three-dimensional hypersonic viscous flow cases were simulated by solving the three-dimensional parabolized Navier-Stokes (PNS) equations. The solution of the three-dimensional PNS equations requires the approximation of two flux derivatives in the crossflow plane. The two-dimensional form of the explicit upwind algorithm for the PNS equations was extended to solve the three-dimensional PNS equations by approximating the convection derivatives with (1) upwind flux approximations (fully upwind method) and (2) a combination of upwind and MacCormack's (1969) flux approximations (upwind-biased method). Both forms of the algorithm are used in the first three-dimensional test case for simulating hypersonic viscous flow over a cone at high angle of attack. A limitation in using the fully upwind method at high angles of attack is discussed. The fully upwind form of the algorithm is used in the last test case to simulate a Mach 24.5 flow field about a generic airplane configuration. A special procedure is used for defining the numerical grid in the crossflow plane during the development of the sharp delta wings.

The computer code for solving the three-dimensional PNS equations used to generate these results can be characterized as a MacCormack code plus a subroutine to execute the equations in the body and appendix of this study for determining the flux-difference splitting (FDS). All inner do loops were vectorized for use on the NAS Cray 2 computer. While the code is highly vectorized, it was developed to verify the algorithms rather than to optimize execution speed. The execution times on the Cray 2 given here can be improved by combining some of the calculations and extending the effective lengths for multiple do loops.

5.1. Hypersonic Flow Over a Cone

The fourth test case simulates laminar, three-dimensional hypersonic flow over a 10° half-angle cone at an angle of attack α of 24° (fig. 5.1). The conditions are

$M_\infty = 7.95$	$\mathrm{Re} = 4.101 \times 10^6/\mathrm{m}$	$L = 0.3048$ m
$T_{\mathrm{total},\infty} = 755.4$ K	$T_w = 309.8$ K	$\mathrm{Pr} = 0.72$
$\sigma = 0.75$		

The above conditions are for the largest angle of attack considered in Tracy's (1963) experimental investigation. The high angle of attack and free-stream Mach number result in a complex flow field because of the interaction of the supersonic crossflow with the boundary layer. The inviscid flow structure is dominated by a conical outer shock. The conical outer shock's shape changes on the leeward side of the cone because of the growth of the viscous layer. The crossflow is stagnated on the windward side and rapidly expands to supersonic speeds as it wraps around the circumference of the cone. The boundary layer gradually thickens as the crossflow moves across the cone toward the leeward side. A crossflow separation occurs as the flow approaches the top of the leeward side. The separation region generates an increased displacement thickness on the leeward side of the cone which expands the position of the outer conical shock on the leeward side. A lambda shock forms inside the outer shock wave as the crossflow approaches the leeward side to provide the necessary transition to subsonic speeds. The complicated flow field is an excellent and demanding test case for establishing the capabilities of numerical codes.

The computational grid for the crossflow plane consisted of 50 points in the normal direction and 56 circumferential points. The computational grid for the crossflow plane is shown in figure 5.2 using every fifth point in the direction normal to the surface. The computational grids and solution contours are displayed in conical coordinates:

$$\left.\begin{aligned} \theta_y &= \frac{y}{R} \arctan\left(\frac{R}{x}\right) \\ \theta_z &= \frac{z}{R} \arctan\left(\frac{R}{x}\right) \end{aligned}\right\} \tag{5.1}$$

where $R = \sqrt{y^2 + z^2}$. The circumferential rays were equally spaced around the cone while the grid was initially stretched in the normal direction with a stretching parameter of 1.12. The outer boundary was set outside the expected shock position. The grid was made to grow in the marching direction in a conical fashion. The code was started from free-stream conditions at $x = 0.015$ with the marching step size determined by equation (3.32). The axial Mach number varied between 0.2 and 0.5 for the first node off the surface at the crossflow stagnation point. Typical values of the law-of-the-wall coordinate (y^+) for the first point off the wall at solution station ($x = 0.266$) varied from 0.7 to a maximum of 5.0 at the crossflow stagnation point.

One reason for using upwind differencing for the convection and pressure terms is to eliminate the need for additional smoothing or damping parameters to maintain numerical stability when capturing shock waves. For three-dimensional flows solved with upwind differences, a problem was encountered having to do with the mapping of a shock wave onto a numerical grid. The perfect grid would be oriented so that the shock wave was contained along

an axis in the computational plane. When a shock wave does not move uniformly in the computational plane, an unrealistic set of initial states occurs when the Riemann problem (RP) is solved tangential to the shock wave. This results in a nonphysical flux error which is not damped out in regions where the numerical scheme has low dissipation. For three-dimensional flow about a cone, this situation occurs along the crossflow symmetry line at high angles of attack. Nonphysical solutions were obtained around the crossflow symmetry line for high angles of attack when using upwind differencing of the pressure and convection terms. Low-angle-of-attack cases were calculated using upwind differences without the above problem. The nonphysical solutions begin to develop when the bow shock wave has moved outside the viscous region, usually at 20 000 to 30 000 marching steps from the initial data crossplane. The pressure slowly becomes either extremely high or low when compared with both the experimental results and the next grid point in the tangential direction. This problem was solved by two different methods: by adding dissipation with Harten's (1983) device when the shocks were not aligned in the computational plane, or by differencing the pressure and convection terms with MacCormack's (1969) method in the direction tangent to a shock wave and using upwind differencing in the normal direction (upwind-biased method). The cross section pressure contour for the fully upwind method (fig. 5.3(a)) shows the difficulty most strongly at the windward crossflow symmetry line, while the contour for the upwind-biased method (fig. 5.3(b)) shows the correct result at the crossflow symmetry. Application of Harten's device was difficult. Harten's device adds dissipation, or smoothing, by artificially preventing the absolute value of eigenvalues from decreasing below a certain level. To correct the problem at the crossflow symmetry line, the amount of smoothing added resulted in a large increase in the boundary layer thickness and smearing of the shock wave. The results obtained with Harten's device are not included since they were poor compared with the solution obtained with the upwind-biased method. The correct solution to the three-dimensional flow problem was obtained without additional damping terms only if a central differencing algorithm (MacCormack's) in the direction tangential to the shock was used.

The surface pressure distribution at $x = 0.325$ (fig. 5.4) agrees well with experimental data on the leeward side and is slightly lower than the experimental results on the windward side. This is typical of previous numerical studies (Gielda and McRae 1986; McRae 1976) and has been attributed to the experimental error associated with the size of the pressure tap compared with the boundary layer thickness on the windward side. The experimental results include cross section surveys of the flow field showing the location of the shock wave, viscous region, and minimum pitot tube pressures. These data were taken along surface normals at $x = 0.2831$ (that is, 8.8 cm) from the apex of the cone measured along the cone's surface. The numerical results are in a plane normal to the centerline of the cone. The numerical results were compared with the experimental data by projecting the numerical results into a conical coordinate system. The numerical results for the crossflow plane that bisects the conical experimental data are shown in figure 5.5 for the Mach number contour at $x = 0.266$ with the experimental determination of shock location, viscous boundary, and minimum pitot pressure. Since the flow field is nearly conical at this point, the locations of the shock and viscous region agree fairly well except on top of the leeward side.

The execution of FDS required 56 percent of the CPU time. The new algorithm achieved a computational rate of 0.434×10^{-4} sec per point for one complete step of the algorithm.

5.2. Hypersonic Flow Past a Generic Vehicle

The last test case simulates a laminar hypersonic flow field about a generic airplane configuration. The purpose of this test case is to demonstrate the capability of the upwind algorithm for solving the PNS equations to simulate a hypersonic flow field about a realistic geometry. The airplane configuration and flow conditions are taken from the numerical study done by Richardson and Morrison (1987). Note that this is a demonstration case since real gas effects are not taken into account. The forward part of the body is a 4.6° half-angle sharp circular cone, which extends 756 in. (19.2 m) from the nose. The cone is connected to a cylindrical body which extends to 1371 in. (34.8 m) from the nose (fig. 5.6). The 12° delta wing has a cross section defined by an angle of 9.327°, set at an angle of attack of 1° relative to the fuselage centerline. The delta wing begins 584.6 in. (14.8 m) from the nose. The configuration geometry is defined by the following equations where the coordinate positions are defined in figure 5.6(b):

$$R = \min\left(x\frac{61 \text{ in.}}{756 \text{ in.}}, 61 \text{ in.}\right)$$

$$y_o = -24.5837 \text{ in.}$$

$$y_{bw} = y_o - (x - 405 \text{ in.}) \tan(1°) \quad z_{bw} = z_{le}$$

$$y_{le} = y_{bw} \quad z_{le} = (x - 405 \text{ in.}) \tan(12°)$$

$$y_{ru} = 2y_o - y_{bw}$$

$$y_{tw}^2 - \left(\frac{2y_{ru}}{1+\beta^2}\right) y_{tw} + \left(\frac{y_{ru}^2 - R^2\beta^2}{1+\beta^2}\right) = 0$$

$$z_{tw} = \sqrt{R^2 - y_{tw}^2}$$

where

$$\beta = 2\left[\frac{\tan(1°)}{\tan(12°)}\right]$$

The flow conditions are

$M_\infty = 24.5$	Re = 12 000/in. (4.7×10^6/m)
L = 1371 in. (34.8 m)	T_w = 2470°R (1372 K)
T_∞ = 490°R (272 K)	Pr = 0.72
$\alpha = 1°$	

For high Mach number flows at large Reynolds numbers, the viscous region becomes extremely thin; thus calculation of flow fields around realistic geometries becomes difficult. The airplane body has a number of discontinuous changes in the surface geometry, which have made calculations of flow fields with traditional numerical methods difficult. Numerical stability problems are often encountered at these locations. In the previous study by Richardson and Morrison, the thin-layer Navier-Stokes (NS) equations were solved with an implicit finite-volume upwind scheme to simulate the flow field.

The NS solution (Richardson and Morrison 1987) was calculated at 52 streamwise stations, with the numerical grid at each cross section containing 65×65 points. The calculation of the PNS solution with an explicit space marching scheme requires the definition of approximately 50 000 to 200 000 streamwise stations. At first, solution to this problem was attempted with the same numerical grid as the previous study, by interpolating (the cross section grids) between the 52 streamwise stations to obtain the intermediate stations. This grid development procedure was satisfactory for space marching the PNS equations until the apex of the delta wings was encountered on the fuselage. The point distribution and the orthogonality of the numerical grid at the body surface required severe changes in the numerical grid in the streamwise direction at the root of the delta wing. The grid points to be used in defining the wing surface were collected at the streamwise station immediately before the apex of the delta wing. All these points were distributed on the wing surface at the next station downstream. Moving all the points onto the wing at one time and requiring the interior domain to be orthogonal at the surface required a large movement of the numerical grid in physical space between these two stations. The severe streamwise change in the numerical grid around the wing region was difficult for the space marching scheme to handle with realistic space marching step sizes. Note that the large streamwise spacing used in the NS calculation effectively smooths out any sharp changes in the streamwise geometry and the development of the wing. Because of the above problems, a different gridding procedure was developed to handle the development of the wing by smoothly adjusting the physical movement of the numerical grid.

The new crossflow numerical grids used for this case were formed with algebraic stretching functions (eq. (4.1)) to cluster the grid at the body surface (fig. 5.7). No attempt was made to make the numerical grid orthogonal at the surface. The point distribution on the outer computational boundary and the body surface was controlled in an attempt to minimize the streamwise changes in the numerical grid. The outer computational boundary was a 7° cone until the wing had grown large enough to compress the numerical grid to 10 percent of the height of the grid along the symmetry line. After this point had been reached, the outer boundary was based on the linear combination of two cones: a 7° cone at the symmetry plane and a cone large enough to include the leading edge of the wing plus 10 percent of the computational domain at the symmetry line. The point distribution on the outer boundary was stretched circumferentially so that the points were clustered along the plane bisecting the wing's leading edge. The body point distribution before the wing is divided into two regions. A constant angular point spacing is used above and below the location where the apex of the delta wing eventually appears. Where the delta wing eventually emerges from the fuselage is defined by three points spaced 0.05° apart (fig. 5.7). These three points are used to define the leading edge of the delta wing. The initial point spacing on the top and bottom of the wing is equal to the point spacing at the leading edge of the wing. As the wing grows in size, the points on the body surface are rotated one at a time onto the wing in a continuous fashion. The amount of rotation is controlled by a ratio of the cross section body and wing perimeters. When 16 points have been rotated onto both the top and the bottom of the wing, the point spacing on the wing begins to increase and the rotation stops. The point spacing for the three points describing the leading edge of the delta wing remains fixed for the complete length of the wing.

The computational grid for half of a crossflow plane was defined by 45 points in the η-direction (away from the body surface) and 63 points in ζ-direction (circumferential). Three crossflow grids are shown in figure 5.7. The plots of the numerical grid show every fifth line in the radial direction. The

symmetry boundary condition required 4 of the 63 points in the ζ-direction. At the final station, the distribution of points in the ζ-direction was as follows: 16 points on the upper fuselage, 12 points on the lower fuselage, and 32 points on the wing. The outer computational boundary at the symmetry plane was equal to the cross section of a 7° circular cone. The point distribution along the outer computational boundary was algebraically stretched ($\beta = 1.3$) to cluster points around the wing tip. The stretching coefficient for the interior domain between the body surface and the outer computational boundary was adjusted to maintain the axial Mach number between 0.50 and 0.85 for the first point off the surface on the windward symmetry line. The code was started using free-stream conditions at 68.55 in. (1.74 m) from the apex of the cone. The step size was calculated from equation (3.32) using a safety factor of 0.95 (inviscid Courant number of approximately 1.9) up to 585 in. (14.9 m) downstream of the cone apex. At this point, the wings began to develop and the safety factor had to be lowered to 0.20 to account for the skewing of the numerical grid and the attachment of a shock wave to the leading edge of the wing. Once the wings began to develop, the pressure boundary condition (eq. (3.31)) had to be used to stabilize the surface pressure on the leading edge of the wings. The total CPU time on the NAS Cray 2 was over 3 hours. It took approximately 1 hour to advance the solution to the point where the wings start, 1 hour to reach the cone-cylinder junction, and 1 hour to reach the end of the cylindrical section. This compares well with the execution time for the NS solution of Richardson and Morrison (1987), which took approximately 21 hours on the Control Data Corp. VPS-32 supercomputer at NASA Langley. The code they used executes on the VPS-32 at roughly the same speed as on the Cray 2. However, had the normal grid spacing at the wall been as refined as in the Richardson and Morrison computation, the total execution times would have been similar.

A numerical difficulty was encountered once the leading edge of the wing intersected the bow shock and moved out into the free stream (at approximately 700 in. (17.8 m) from the cone apex). A slight pressure undershoot occurs for the shock captured on the ray of points emerging from the leading edge of the wing. The pressure undershoot eventually causes a numerical instability. The pressure undershoot was found to be eliminated by either lowering the value of β in the flux limiter or by adding explicit smoothing to the ray of points emerging from the leading edge of the wing. The smoothing terms could then be removed or β increased, but eventually the pressure undershoot would occur again. The undershoot is thought to be caused from the misalignment of the grid around the shock wave adjacent to the leading edge of the wing. The grid movement causes shock wave misalignment with the grid and occurs because of expansion of the outer computational boundary and adjustment of the stretching coefficient to maintain the streamwise Mach number. A large enough grid movement changes the shock position relative to the grid points in the computational plane. The pressure undershoot is considered to be a result of this shock wave position change on the grid. The solution presented here was calculated with an explicit second-order smoothing term added only to the ray emerging from the leading edge of the wing. The explicit smoothing term added to the solution at these points is defined as

$$a_1(\mathbf{E}_i)_{\eta\eta} + a_2(\mathbf{E}_i)_{\zeta\zeta} \tag{5.2}$$

where

$$a_1 = c_1 p_{\eta\eta} \quad a_2 = c_2 p_{\zeta\zeta}$$

The second partial derivatives of $(\mathbf{E}_i)$ and (p) with respect to η and ζ were approximated using second-order central differencing. The explicit smoothing was applied after the 700-in. (17.8-m) station using a coefficient of $c_1 = c_2 = 0.002$.

The PNS results calculated in this study were compared with Richardson and Morrison's (1987) NS results. To compare the PNS finite-difference solution directly with the NS finite-volume results, the cell center locations had to be calculated for the NS solutions. Pressure, temperature, and axial velocity profiles at the windward symmetry plane are compared at three stations in figures 5.8, 5.9, and 5.10. The profiles agree favorably with respect to values before and after the shock, surface values, and profile shape. Slight differences in the various profiles can be partly attributed to the different point spacing used in the two calculations, the different numerical integration methods used, and the different equation set solved. The first station (256 in. (6.5 m), part (a) of each figure) is on the forward part of the cone, before the wings appear. The pressure, temperature, and velocity profiles compare favorably between the two methods. The second station (767 in. (19.5 m), part (b) of each figure) is 11 in. (0.3 m) downstream of the cone-cylinder junction. An expansion wave begins to propagate into the flow field away from this junction to expand the flow around the cone-cylinder corner. The comparison of pressure profiles (fig. 5.8(b)) shows that in the NS solution, the expansion has propagated farther away from the body and is more rounded than the PNS solution. The rounding of the expansion wave can be partially

attributed to the upstream influence permitted by the NS equations. The PNS solution does not round the wave since it does not feel the effect of the corner until it is reached. Also, the large streamwise spacing used in the NS calculation, approximately 27 in. (0.7 m), effectively smooths out the beginning of the expansion wave. The PNS solution predicts a slightly higher maximum temperature (fig. 5.9(b)), while good agreement is observed between the axial velocity distributions (fig. 5.10(b)) at the second station. The third station (1304 in. (33.1 m), part (c) of each figure) is near the end of the airplane. The strength of the outer bow shock was weakened by the expansion of the flow onto the cylindrical body. Slightly different shock locations are predicted by the two calculations while the surface pressure is in agreement (fig. 5.8(c)). The PNS solution predicts a small drop in pressure outside the edge of the boundary layer which is not predicted in the NS solution. This drop in pressure could be caused by the turning of the flow in the crossflow plane near the saddle point on the symmetry plane, as discussed subsequently. The numerical grid used in the PNS calculations contains almost twice as many points on the lower part of the fuselage surface as the numerical grid used in the NS calculations. The resolution of the numerical grid used in the NS calculation around the lower body may not have been adequate to resolve this feature. The PNS solution consistently predicts a higher maximum temperature in the boundary layer (fig. 5.9(c)).

The surface pressures on the wing are compared in figure 5.11. A slight difference between solutions is observed toward the middle of the wing. A portion of this difference may be attributed to the more accurate pressure boundary condition used in this study or the difference in modeling the leading edge of the wing. A solution point is located along the leading edge in the PNS calculation while in the NS calculation the leading edge is between two cells. The pressure contours are compared in figures 5.12, 5.13, and 5.14, using 16 identical contour levels for each station. The PNS solution more sharply captures the bow shock due to the more refined grid at the shock location. The PNS prediction of the bow shock location agrees with the NS solution. The outer bow shock and the decrease in the pressure levels around the cone for flow at an angle of attack are clearly defined in the pressure contour at station 256 in. (6.5 m) for the PNS solution (fig. 5.12). The bow shock, expansion wave, delta-wing shock, and change in pressure around the cone circumference are shown for the PNS solution at station 767 in. (19.5 m) in figure 5.13. The pressure contour at station 1304 in. (33.1 m) (fig. 5.14) shows the pressure increasing from the body to the bow shock and the high pressure region created above and below the wing immediately inboard the corner shock.

The details of the flow field in the crossflow plane can be investigated using projections of the velocity vectors in the appropriate coordinate plane. For locations close to the body, the Cartesian plane normal to the body can be used to visualize the flow field in the crossflow plane. The Cartesian crossflow velocity vectors are shown in figure 5.15. The flow field next to the lower surface of the fuselage is shown in figure 5.16. Note the high crossflow velocity around the corner of the fuselage-wing junction, the vortex located underneath the fuselage-wing junction, and the saddle point located on the symmetry line. The pressure drop outside the edge of the boundary layer in the PNS solution (fig. 5.8(c)) could be caused by the turning of the flow near this saddle point. The Cartesian crossflow velocity vectors for the middle of the delta wing (fig. 5.15) suggest a strong reverse flow.

The Cartesian plane projection of the velocity vectors on the delta wing is misleading since the inviscid flow field on an isolated delta wing is conical. To view the flow field on the delta wing, a conical coordinate system centered at $x = 658.9$ in. (16.7 m), $y = -29.02$ in. (0.7 m), $z = 53.96$ in. (1.4 m) on the leading edge of the delta wing was used to project the velocity vectors onto a conical plane. The location of the conical coordinate system was obtained by extending a line along the lower wing-fuselage junction until it intersected the leading edge. The use of a conical plane to observe the crossflow velocity vectors on the wing is an attempt to examine the crossflow in a more natural plane. Note that different locations of the conical coordinate system yield slightly different results. The conical coordinate system defined above was selected because it represents a projection plane that is normal to the leading edge and the fuselage-wing junction. The conical velocity vectors along the leading edge are shown in figure 5.17. The leading edge shock wave and the flow separation on the upper and lower surfaces are apparent from the velocity profiles. The flow separates at approximately $\theta_z \approx 7.2°$ on the top surface and at $\theta_z \approx 6.2°$ on the lower surface. The conical velocity vectors along the middle of the wing span are shown in figure 5.18. Note that the scaling factor for the magnitude of the velocity vectors has doubled from figure 5.17. The separated flow on the lower surface has reattached at approximately $\theta_z \approx 4.6°$. The flow on the upper surface is more complicated. The flow is reattached to the upper surface at $\theta_z \approx 5.8°$, and the flow separates a second time at approximately $\theta_z \approx 5.5°$ with reattachment at $\theta_z \approx 4.5°$.

The shock waves in the crossflow plane consist of a bow shock, leading edge shock, and a corner shock. The corner shock is similar to the shock wave structure obtained for supersonic flow in the corner of intersecting wedges (West and Korkegi 1972). The bow shock and the leading edge shock of the wing do not intersect, but are joined by a third corner shock. The compression corner formed by the cone–delta-wing junction is similar to the corner made by two wedges. The expansion around the cone-cylinder junction weakens the bow shock and complicates the flow field around the corner shock. The conical crossflow Mach number contours (fig. 5.19) are parallel with the corner shock and indicate that the corner shock is essentially conical. The pressure contours (fig. 5.20) and the density contours (fig. 5.21) show the complicated corner shock wave structure in the crossflow plane. A crossflow shock wave is shown by the collection of crossflow Mach number contours, pressure contours, and density contours on the upper wing surface centered at approximately $\theta_y \approx -0.5°$, $\theta_z \approx 5.5°$. The interaction of the expansion wave has diffused the effect of the refracted leading edge and bow shock wave shown in the pressure contour at station 767 in. (19.5 m) (fig. 5.13).

To demonstrate the complete flow field more clearly, the pressure and Mach number contours are shown in color in figures 5.22 and 5.23. The separated flow on the wing and the vortex underneath the wing-fuselage junction can easily be seen in the color Mach number contour. The high pressure region inboard of the corner shock and the gradual pressure decay to the body surface from the outer shock wave are shown in the pressure contour.

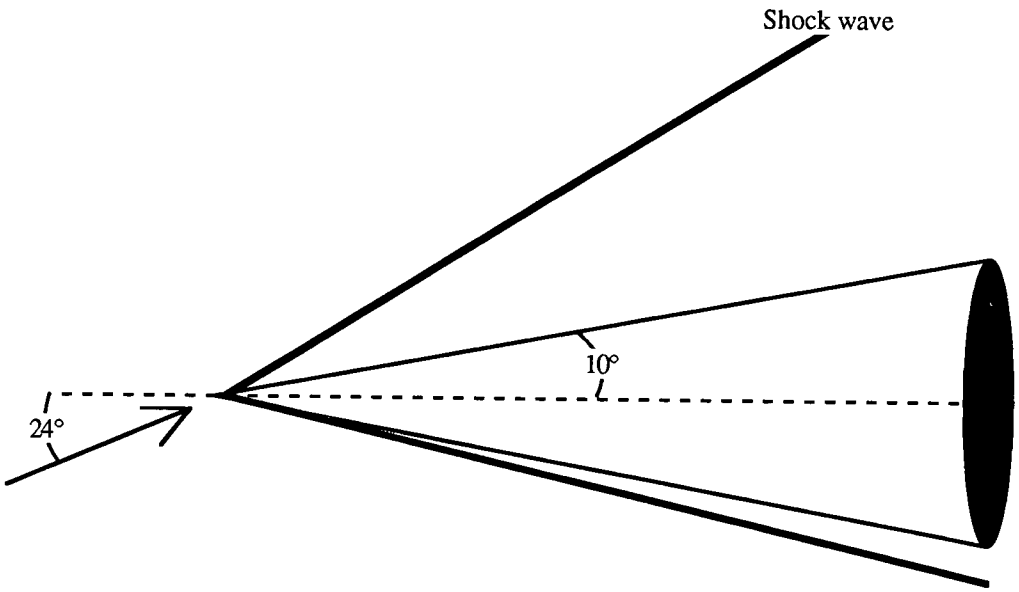


Figure 5.1. Hypersonic flow over a cone at an angle of attack.

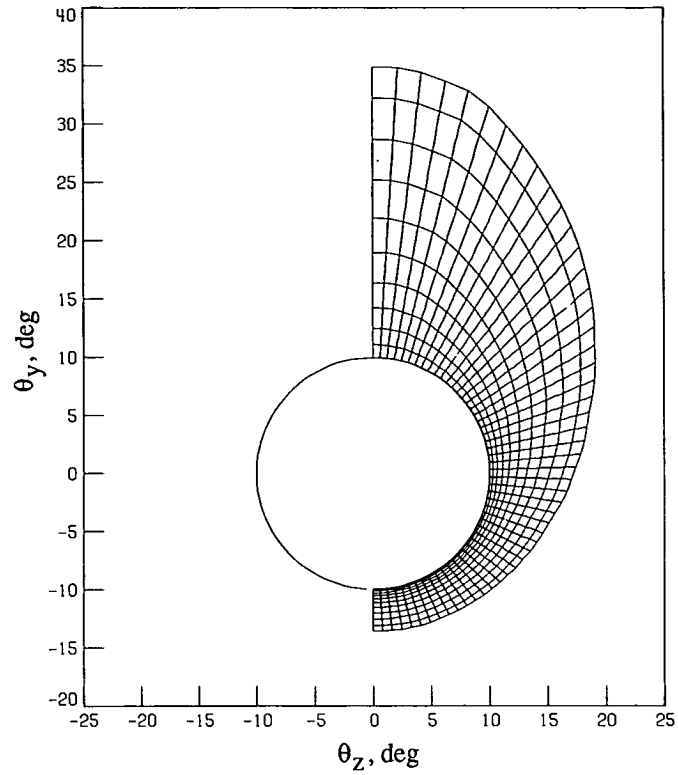
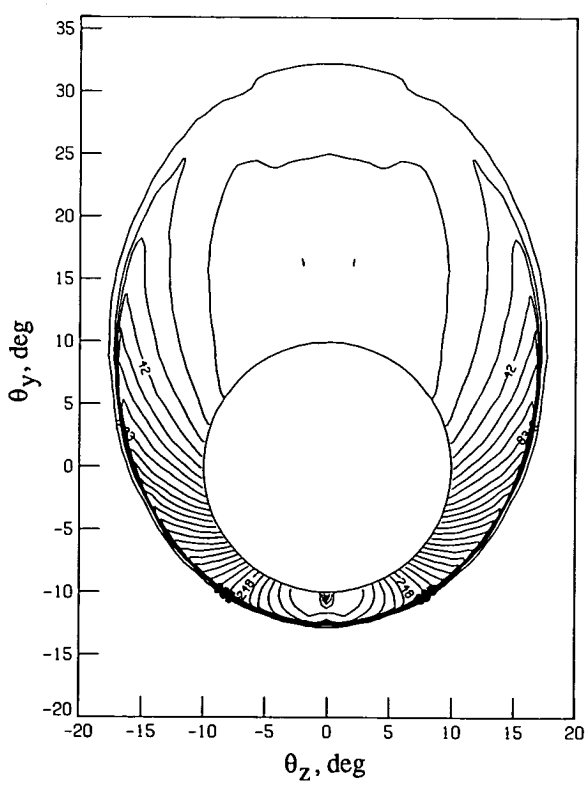
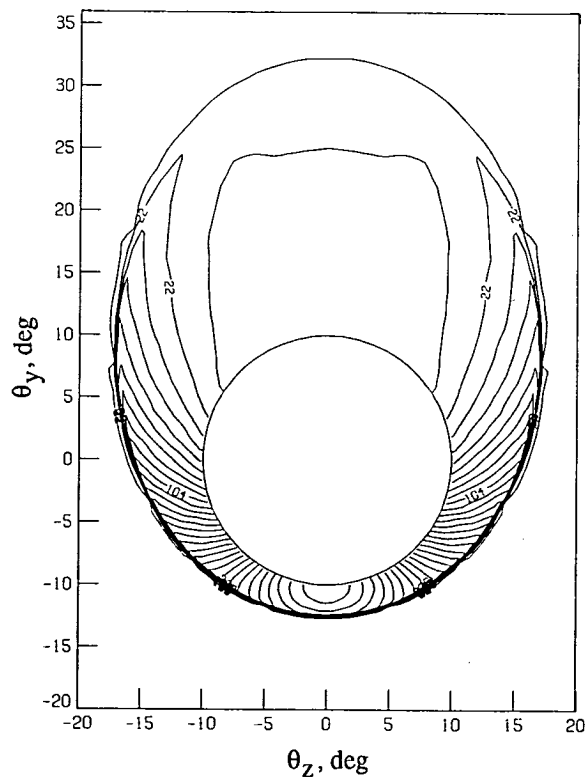


Figure 5.2. Computational grid for a crossflow plane of a cone. $\alpha = 24°$.



(a) Fully upwind method.



(b) Upwind-biased method.

Figure 5.3. Comparison of computed pressure contours for a 10° half-angle cone. $M_\infty = 7.95$; $\mathrm{Re}_x = 3.6 \times 10^5$; $T_w/T_{\mathrm{total},\infty} = 0.40$; $x = 0.266$.

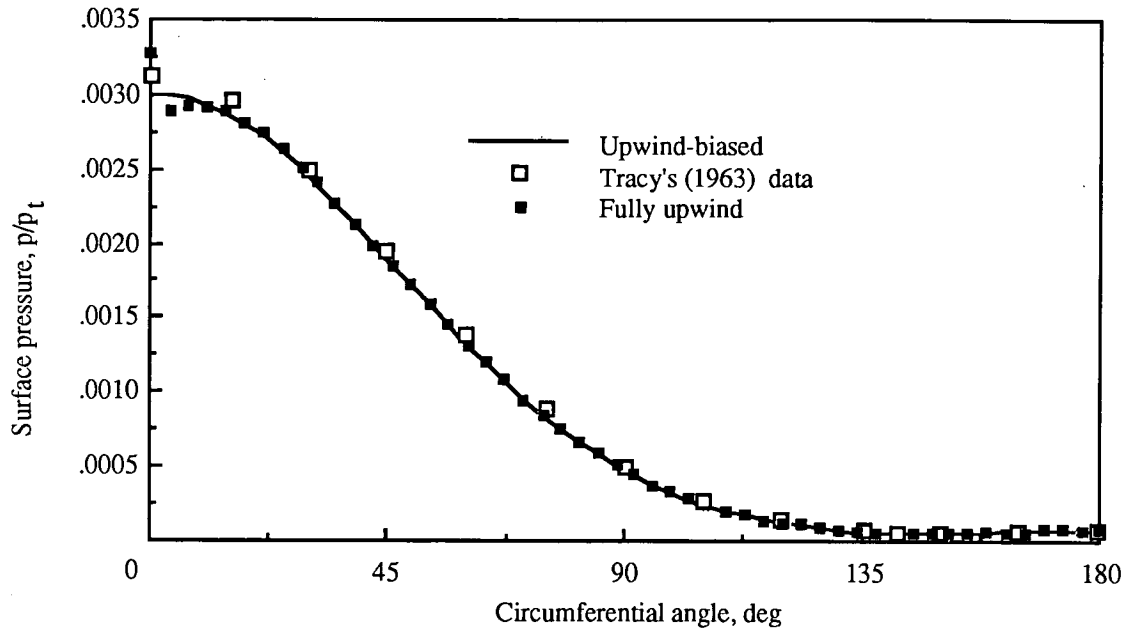


Figure 5.4. Comparison of computed surface pressures with experimental data. $M_\infty = 7.95$; $\alpha = 24°$; $\mathrm{Re}_x = 4.2 \times 10^5$; $T_w/T_{\mathrm{total},\infty} = 0.40$; $x = 0.325$.

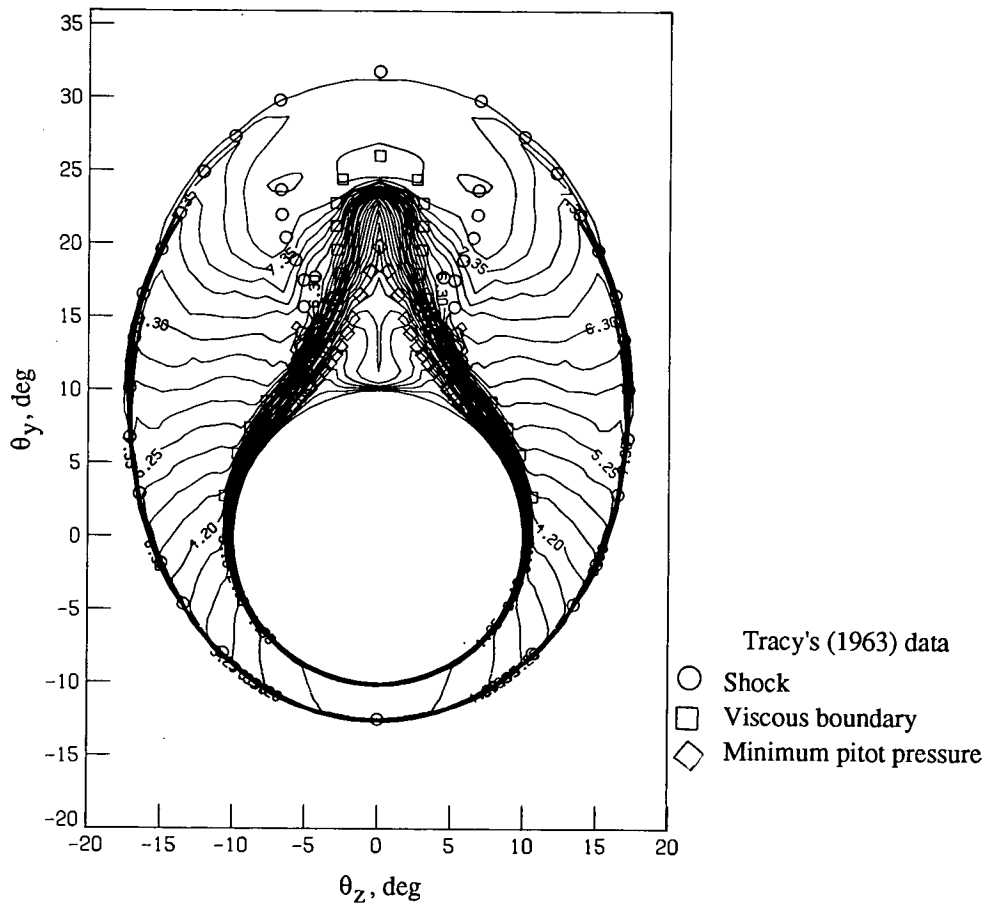
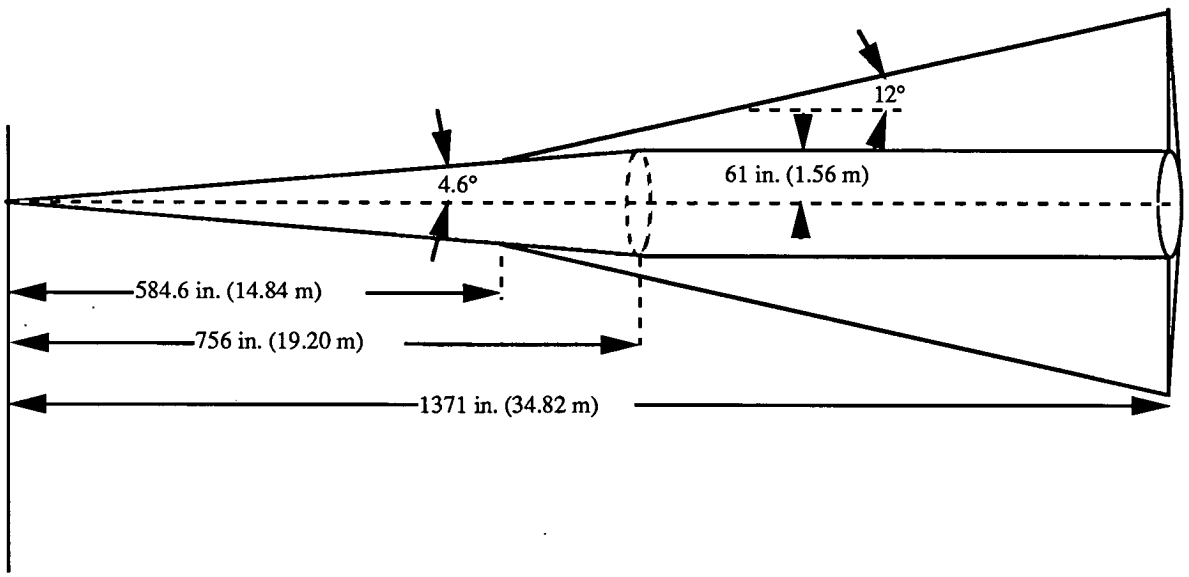
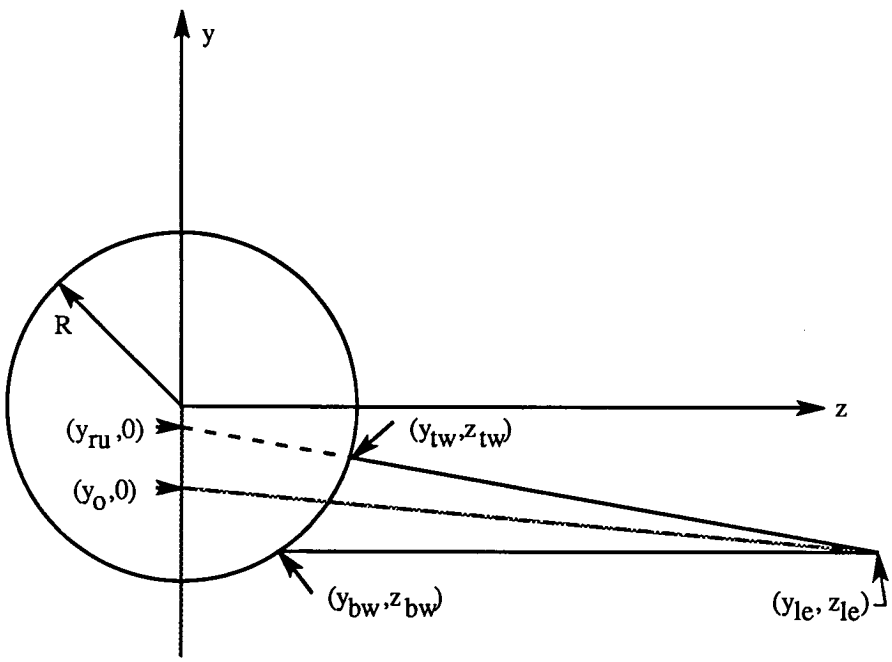


Figure 5.5. Comparison of computed Mach number contours with Tracy's flow field survey. $M_\infty = 7.95$; $\alpha = 24°$; $\mathrm{Re}_x = 3.6 \times 10^5$; $x = 0.266$.

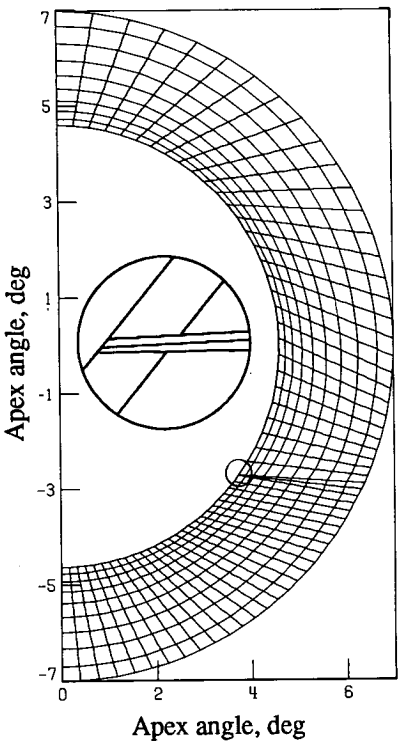


(a) Top view.

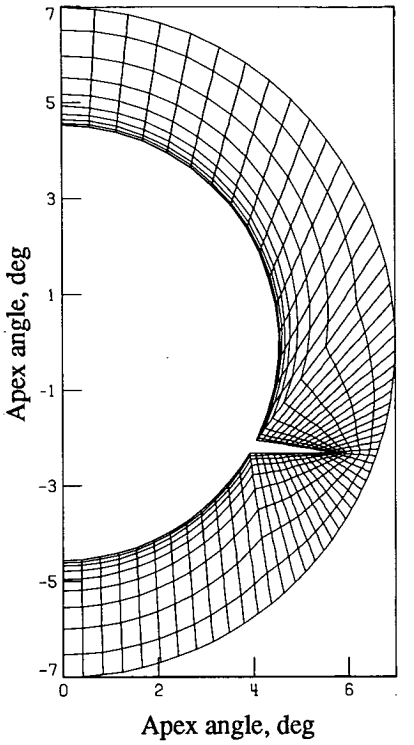


(b) Geometry.

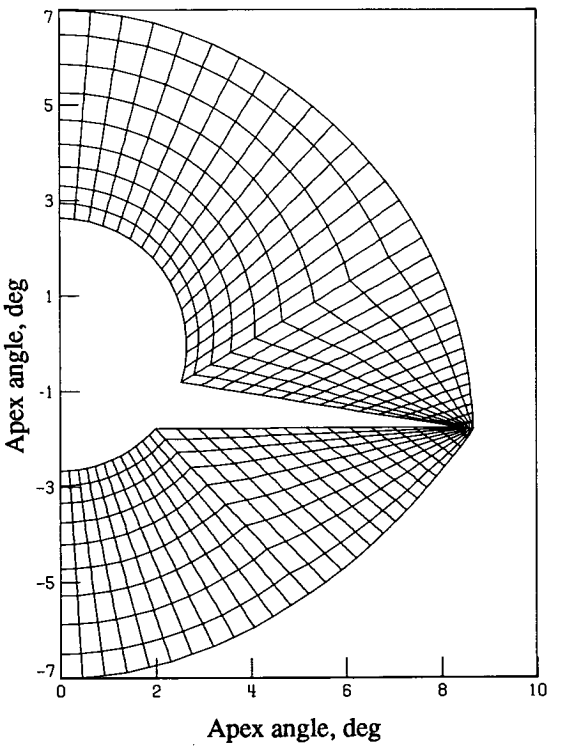
Figure 5.6. Generic airplane configuration.



(a) $x = 256$ in. (6.5 m).

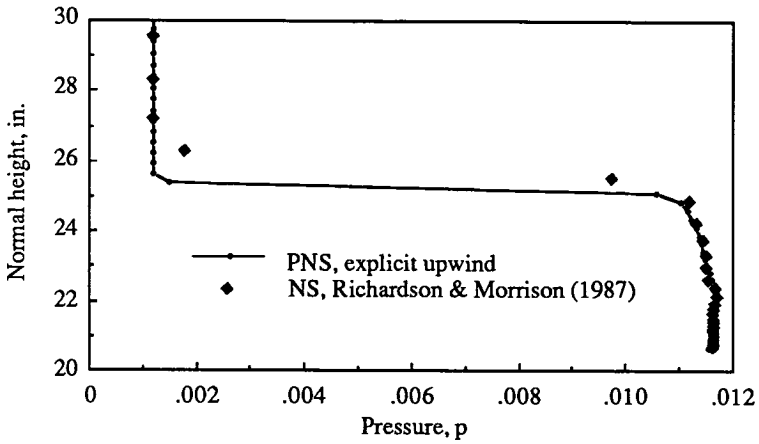


(b) $x = 767$ in. (19.5 m).

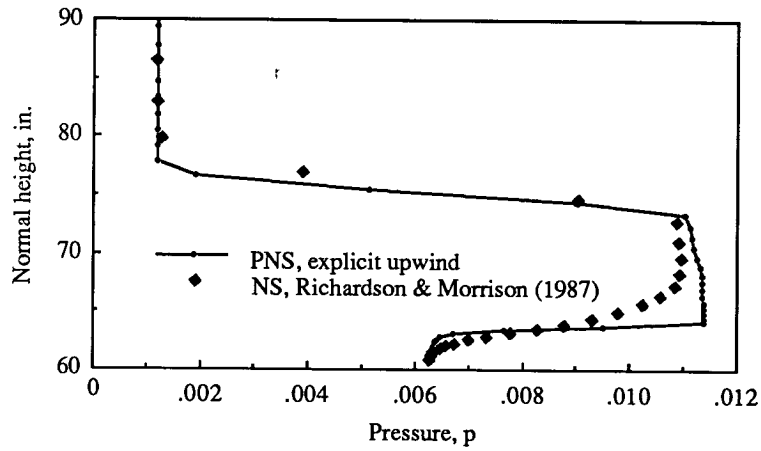


(c) $x = 1304$ in. (33.1 m).

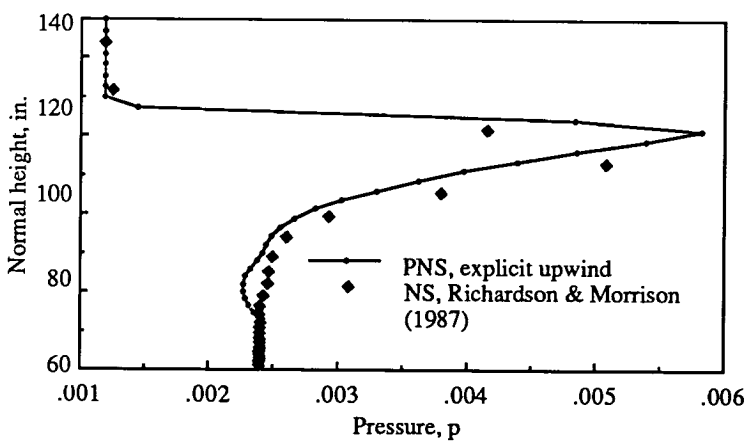
Figure 5.7. Computational grids for three crossflow planes of the airplane.



(a) $x = 256$ in. (6.5 m).

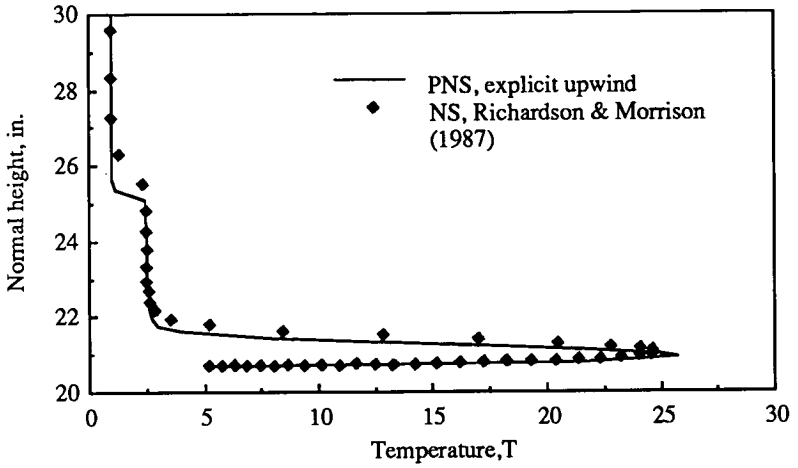


(b) $x = 767$ in. (19.5 m).

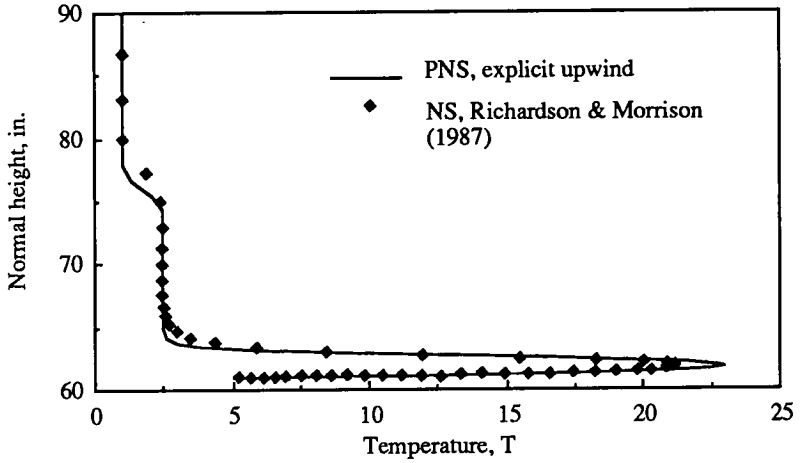


(c) $x = 1304$ in. (33.1 m).

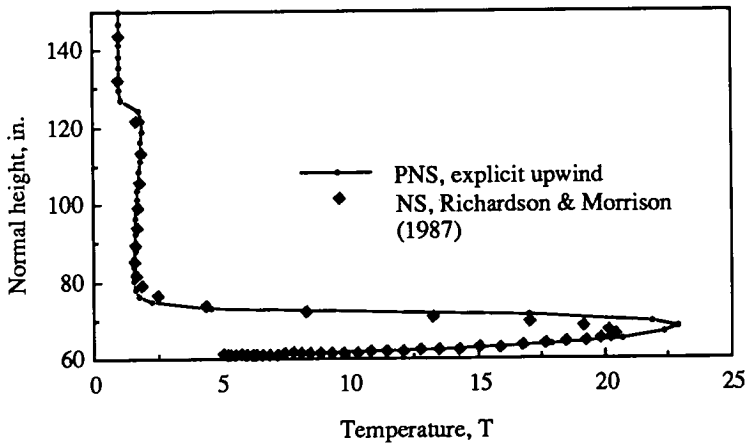
Figure 5.8. Comparison of computed pressure profiles on the windward symmetry plane. $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$. (1 in. = 0.0254 m.)



(a) $x = 256$ in. (6.5 m).

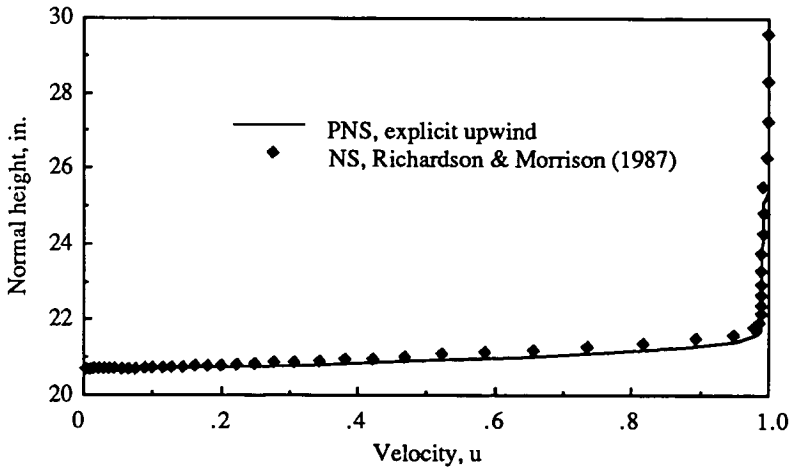


(b) $x = 767$ in. (19.5 m).

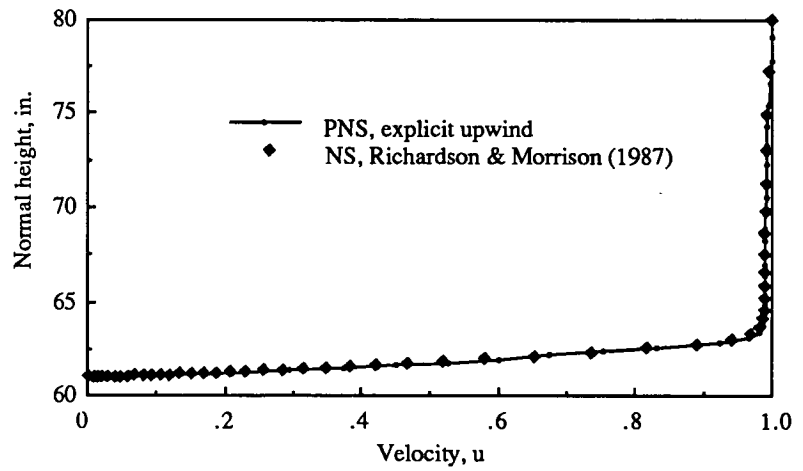


(c) $x = 1304$ in. (33.1 m).

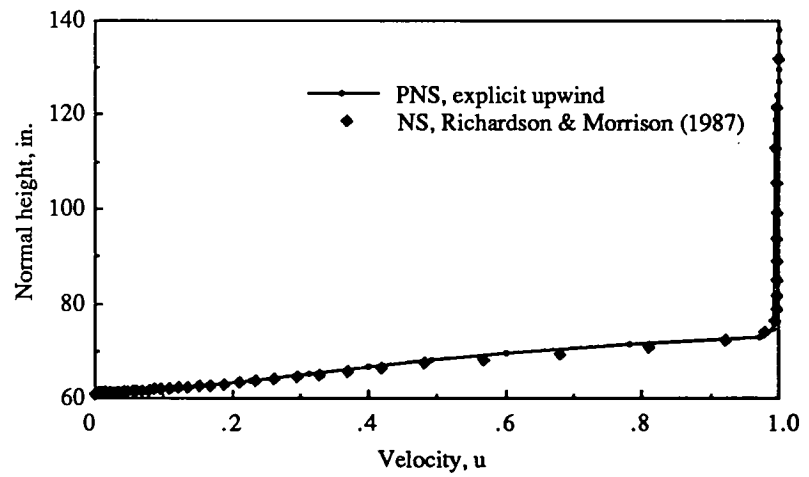
Figure 5.9. Comparison of computed temperature profiles on the windward symmetry plane. $M_\infty = 24.5$; Re = 12000/in. (4.7×10^6/m); $\alpha = 1°$. (1 in. = 0.0254 m.)



(a) $x = 256$ in. (6.5 m).



(b) $x = 767$ in. (19.5 m).



(c) $x = 1304$ in. (33.1 m).

Figure 5.10. Comparison of computed axial velocity profiles on the windward symmetry plane. $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$. (1 in. = 0.0254 m.)

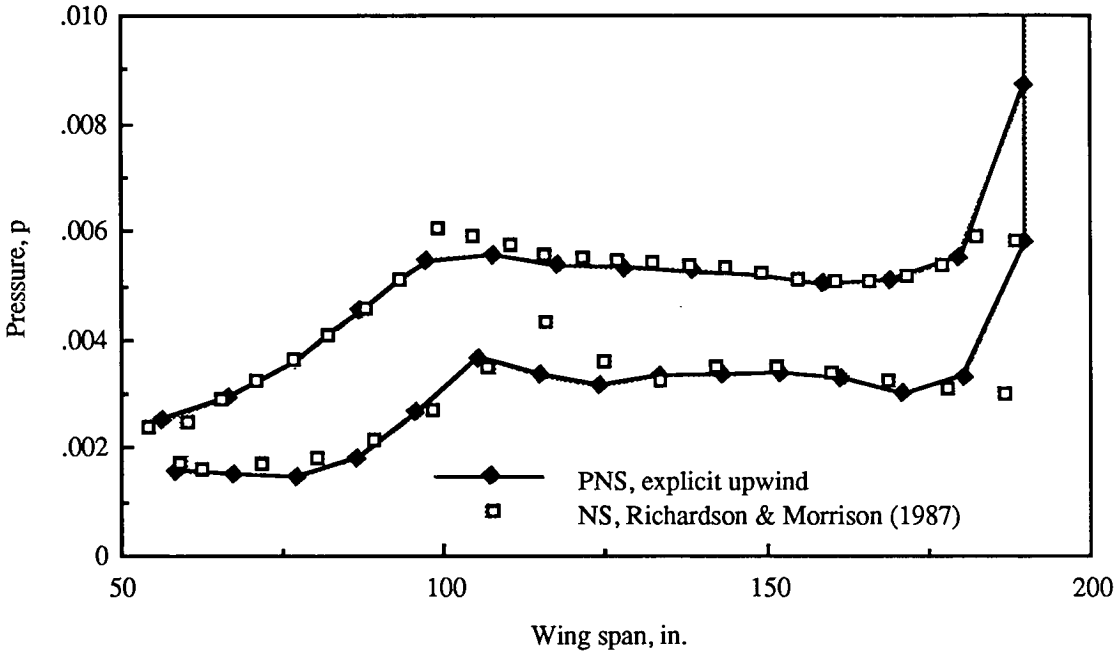
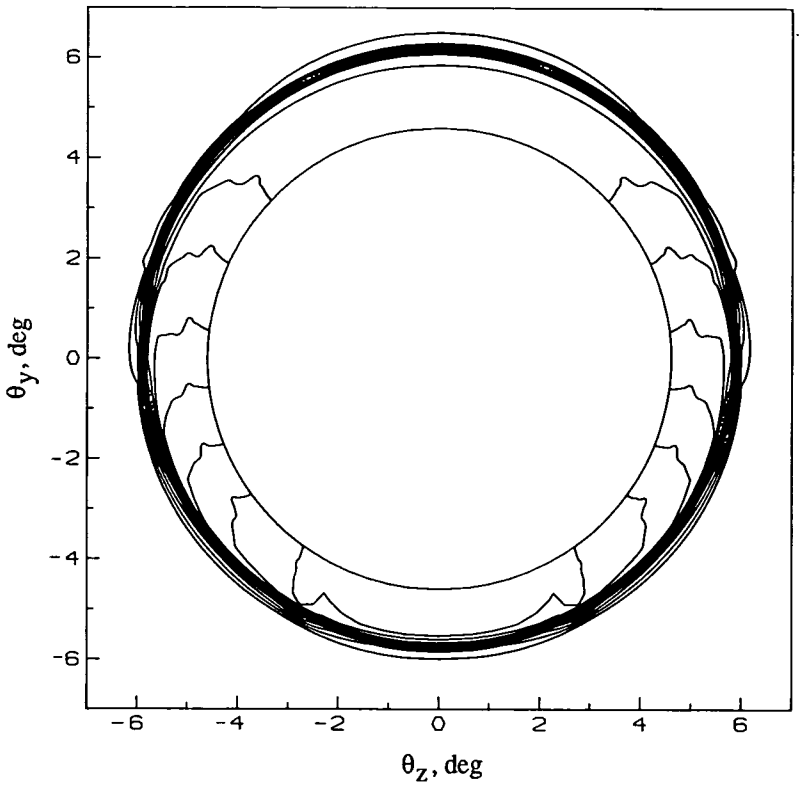
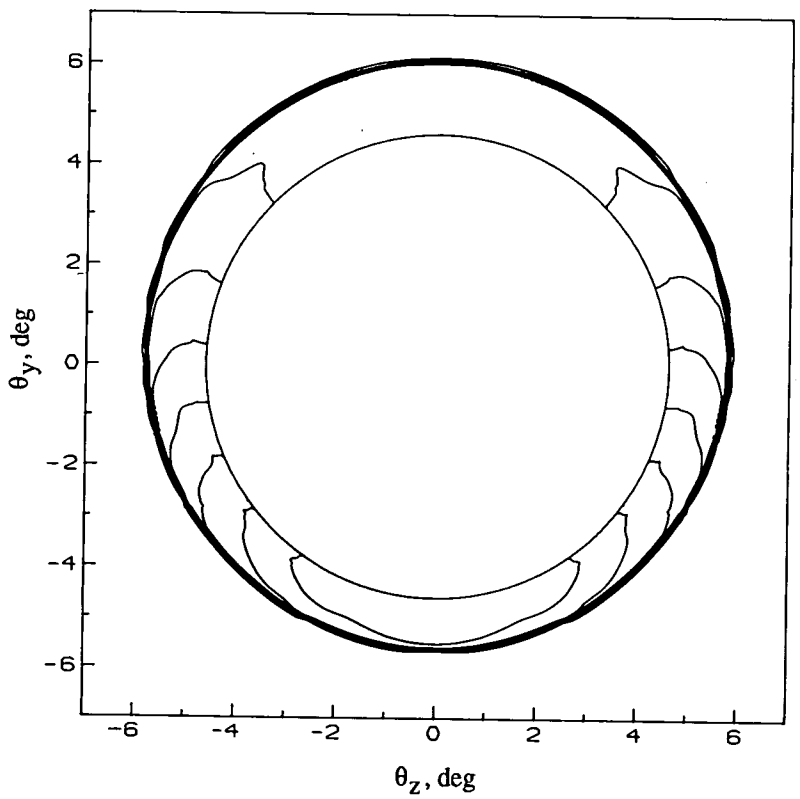


Figure 5.11. Comparison of computed surface pressures on the wing at station 1304 in. (33.1 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$. (1 in. = 0.0254 m.)

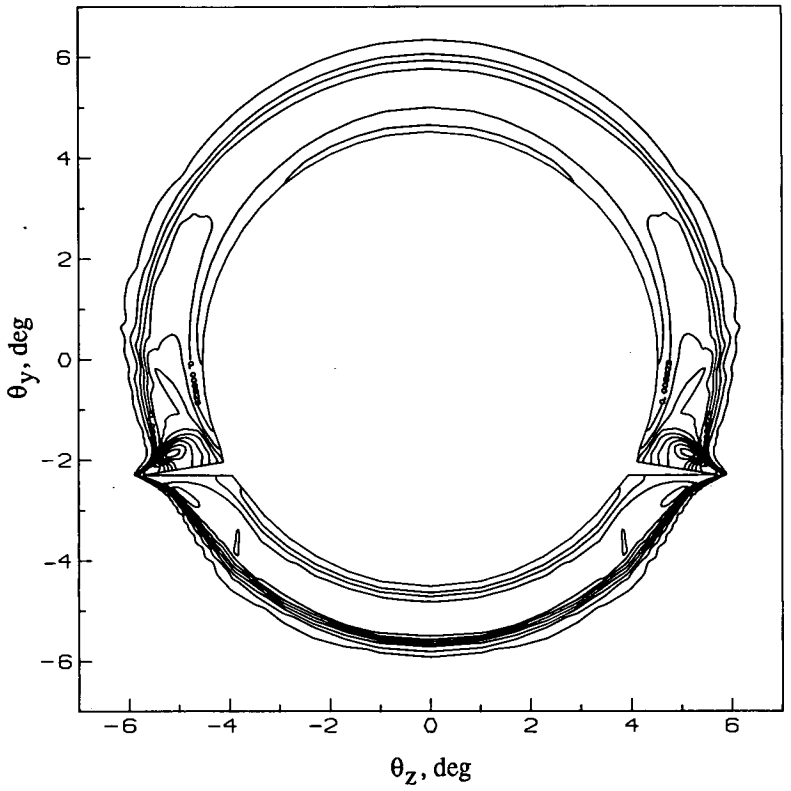


(a) NS solution (Richardson and Morrison 1987).

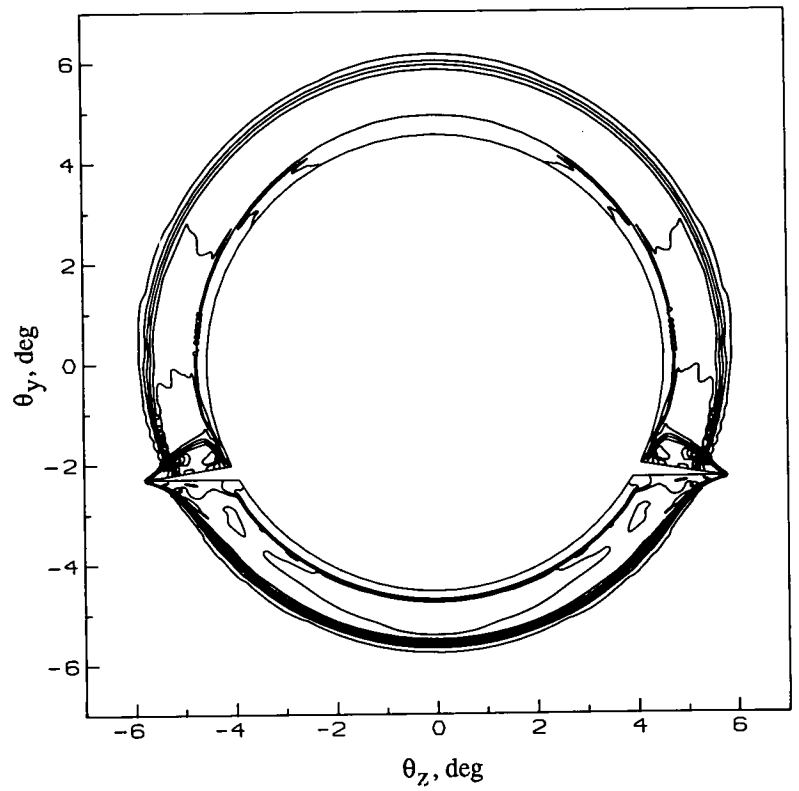


(b) PNS explicit upwind solution.

Figure 5.12. Comparison of computed pressure contours at station 256 in. (6.5 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$.

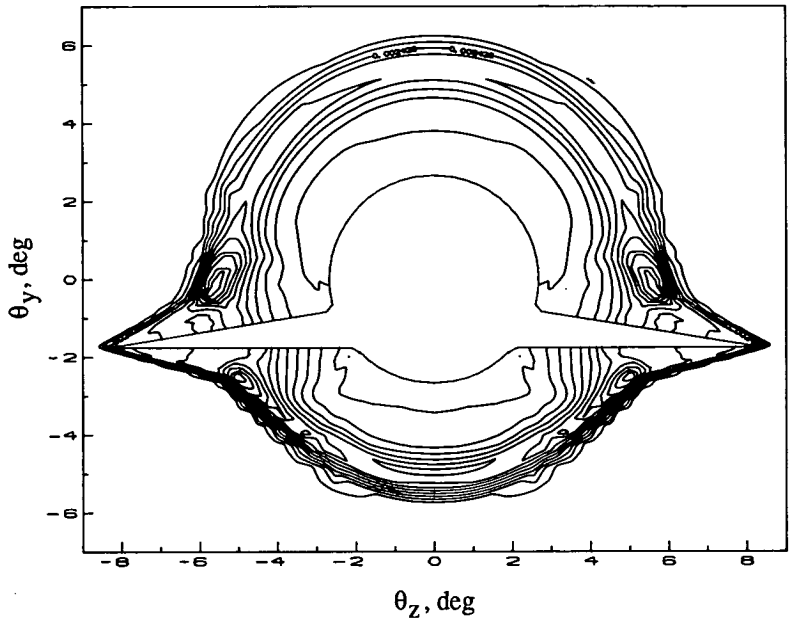


(a) NS solution (Richardson and Morrison 1987).

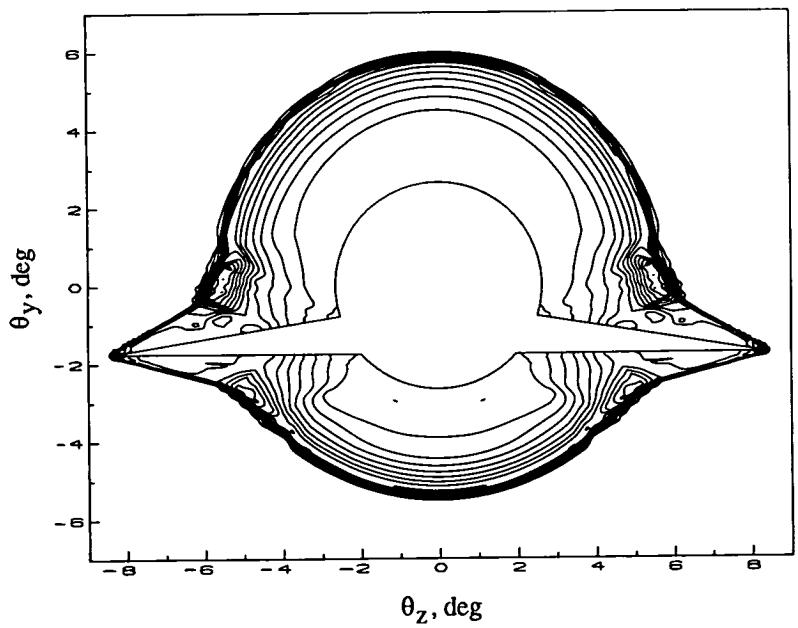


(b) PNS explicit upwind solution.

Figure 5.13. Comparison of computed pressure contours at station 767 in. (19.5 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$.



(a) NS solution (Richardson and Morrison 1987).



(b) PNS explicit upwind solution.

Figure 5.14. Comparison of computed pressure contours at station 1304 in. (33.1 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$.

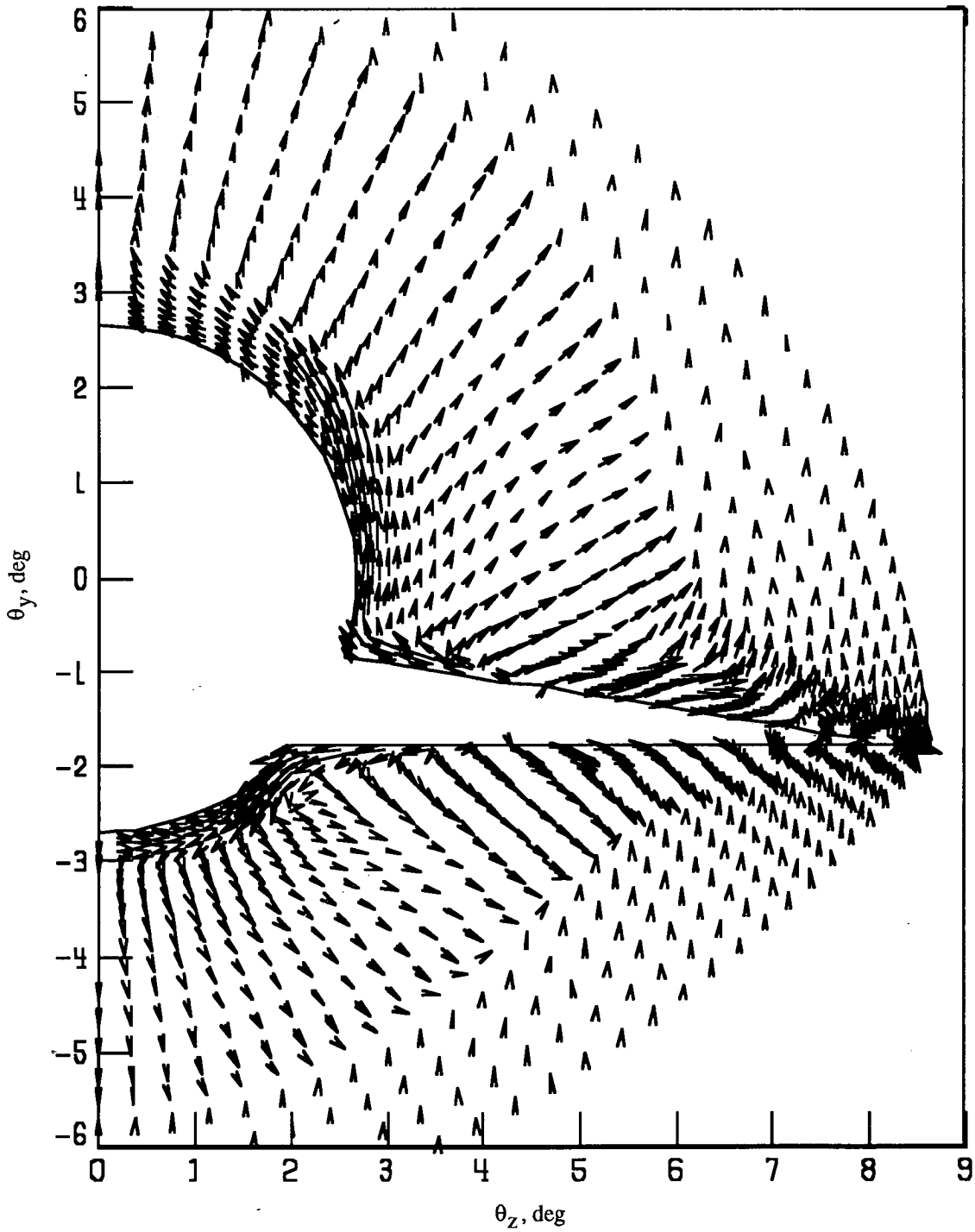


Figure 5.15. Cartesian crossflow velocity vectors at station 1304 in. (33.1 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$.

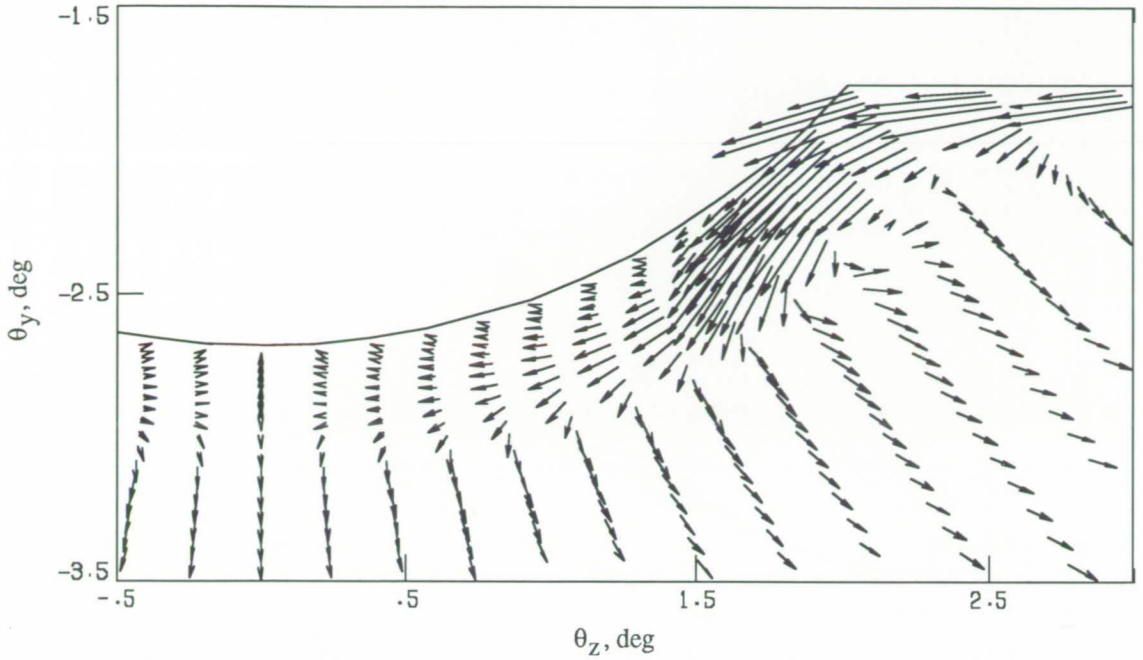


Figure 5.16. Cartesian crossflow velocity vectors on the windward side of the fuselage at station 1304 in. (33.1 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$.

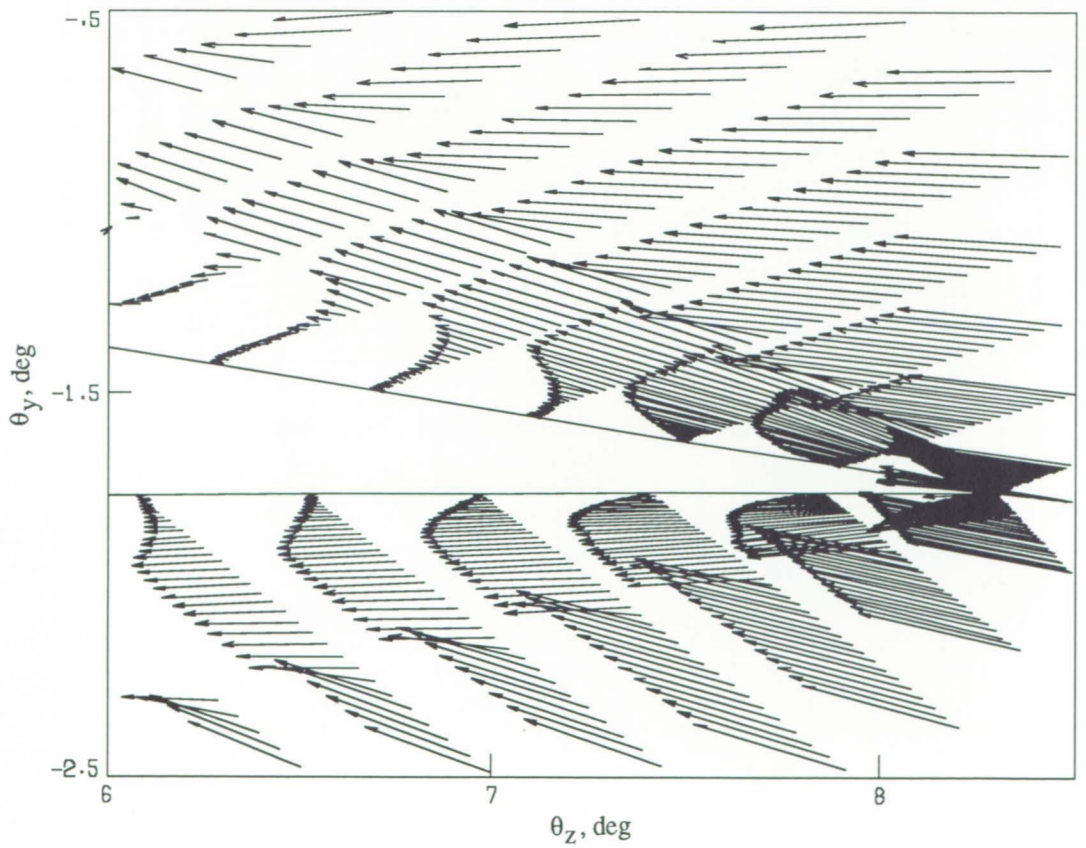


Figure 5.17. Conical crossflow velocity vectors near the leading edge of the delta wing at station 1304 in. (33.1 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$.

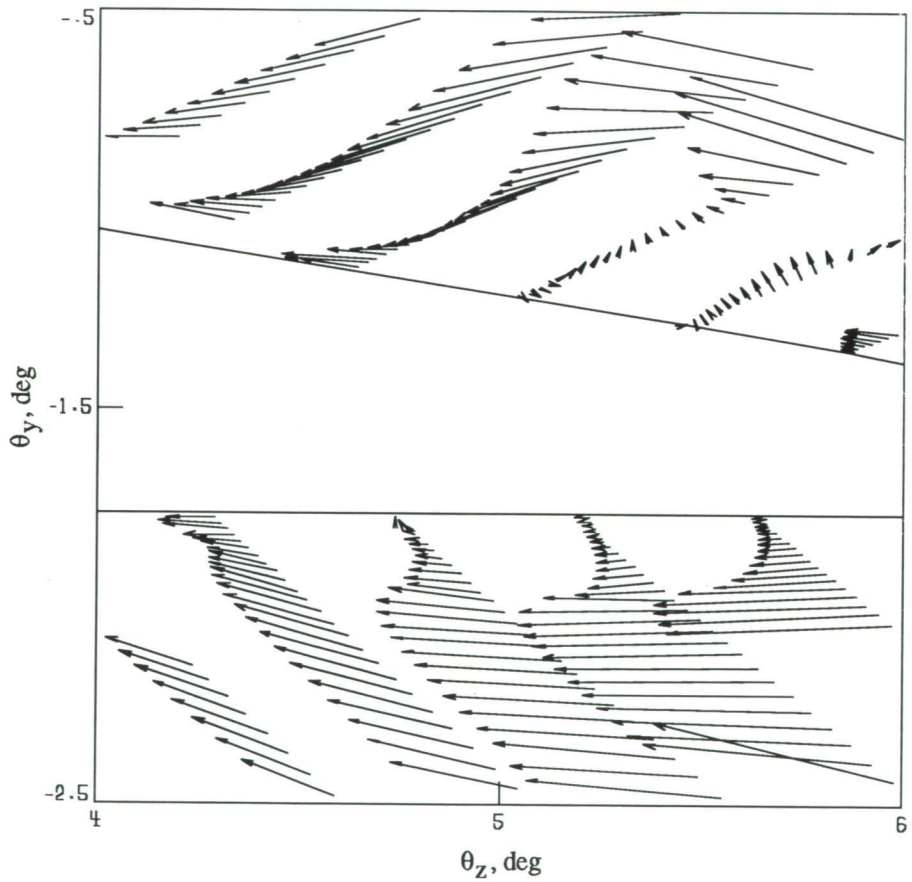


Figure 5.18. Conical crossflow velocity vectors on the delta wing at station 1304 in. (33.1 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$.

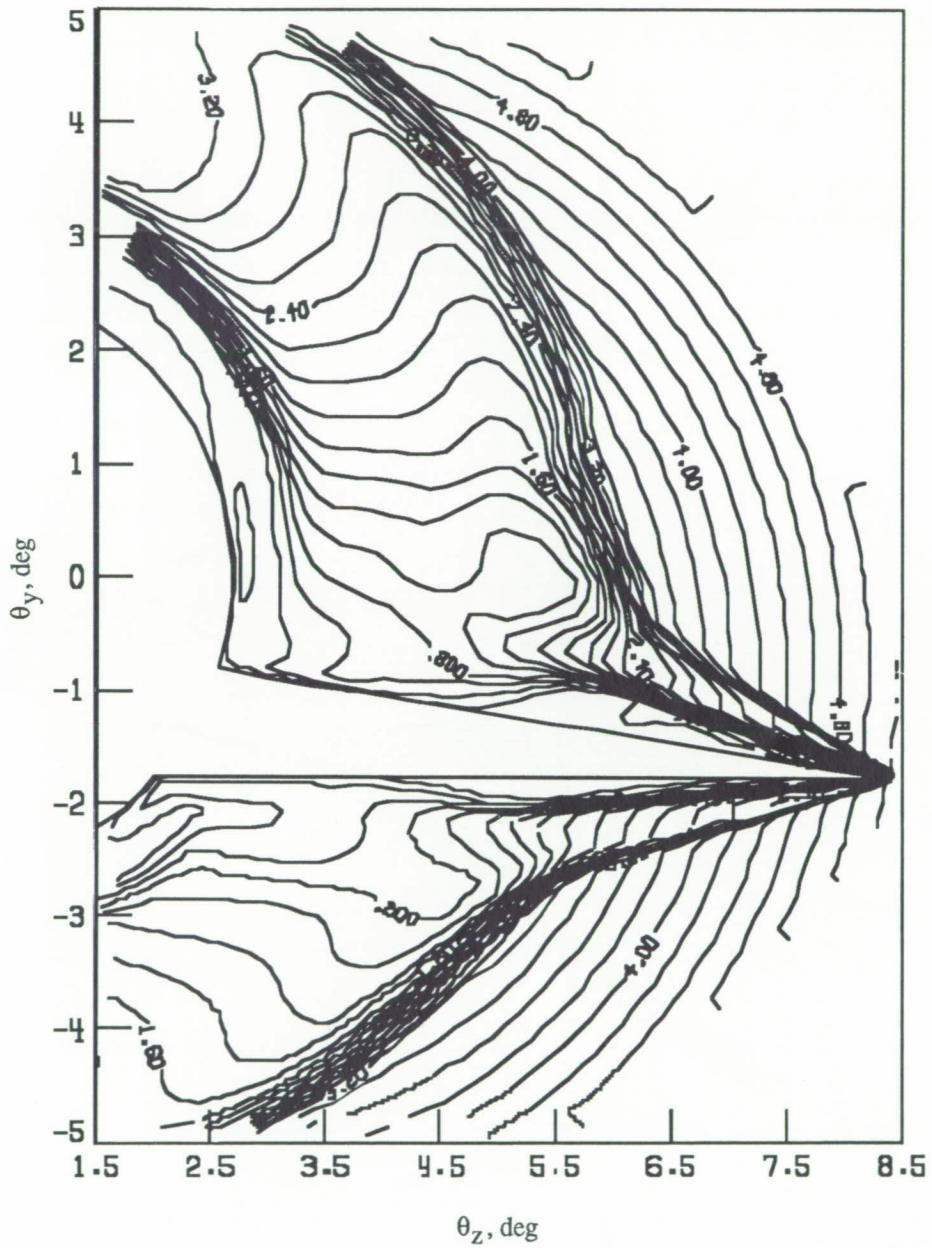


Figure 5.19. Conical crossflow Mach number contours at station 1304 in. (33.1 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$.

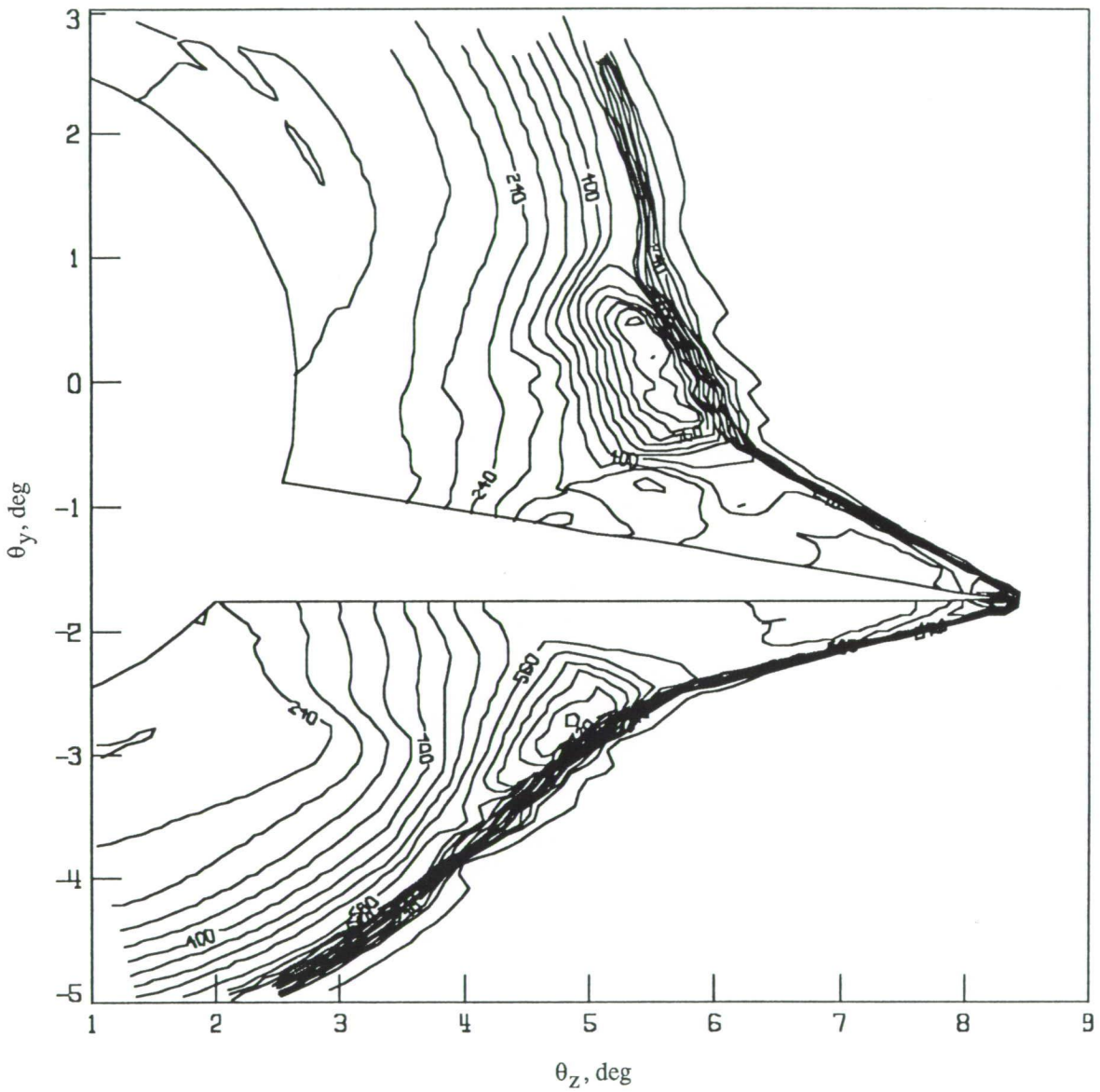


Figure 5.20. Computed wing pressure contours at station 1304 in. (33.1 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$.

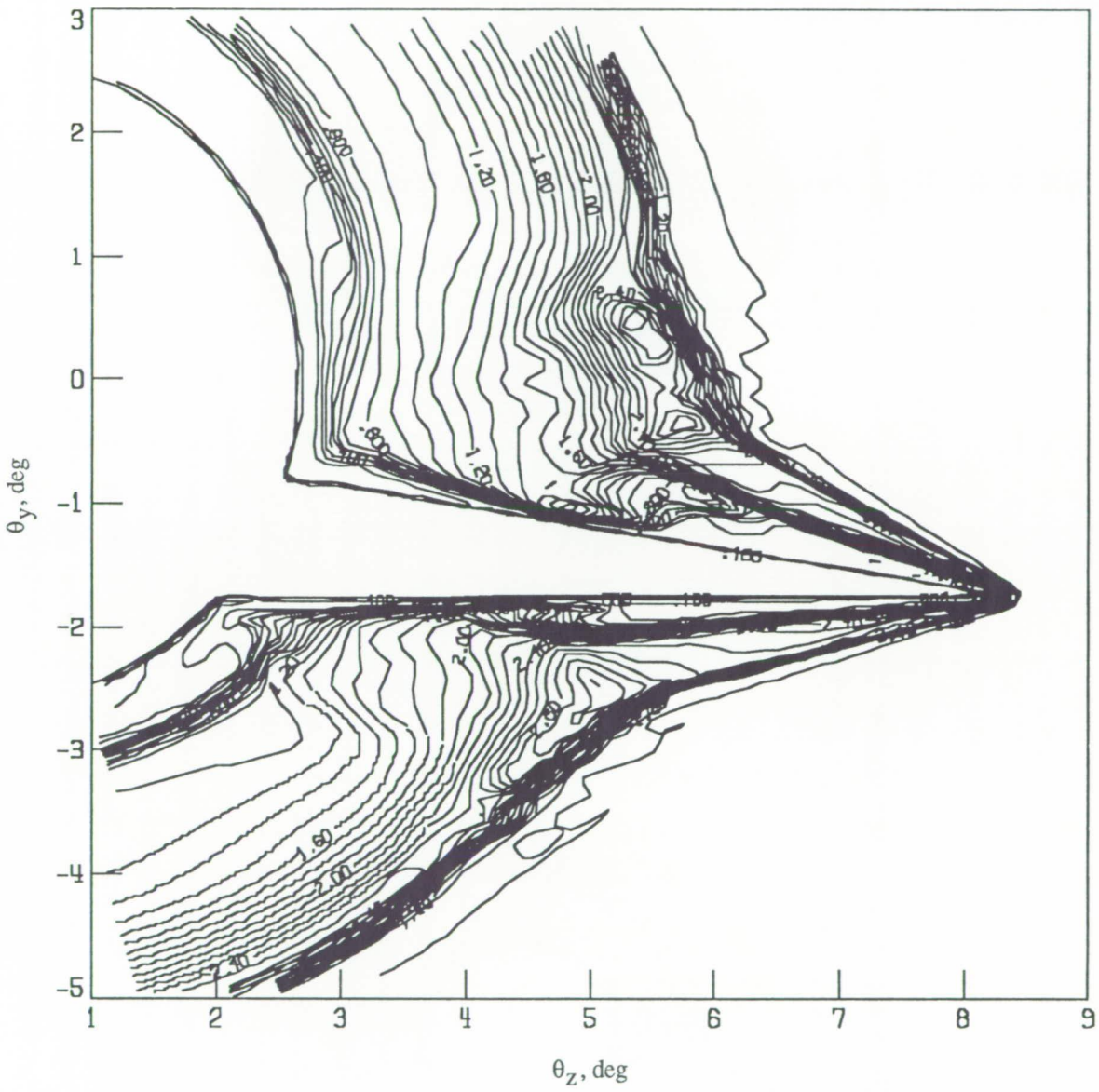


Figure 5.21. Computed wing density contours at station 1304 in. (33.1 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$.

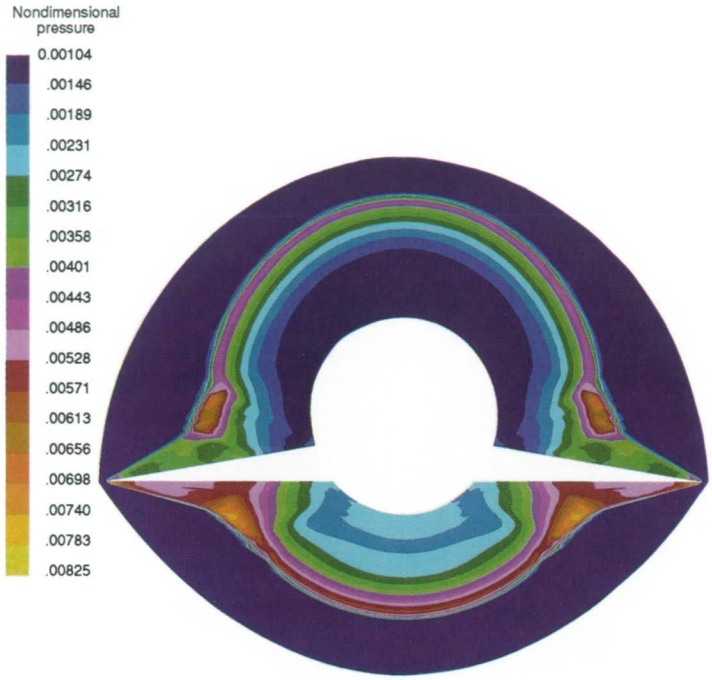


Figure 5.22. Computed pressure contours at station 1304 in. (33.1 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$.

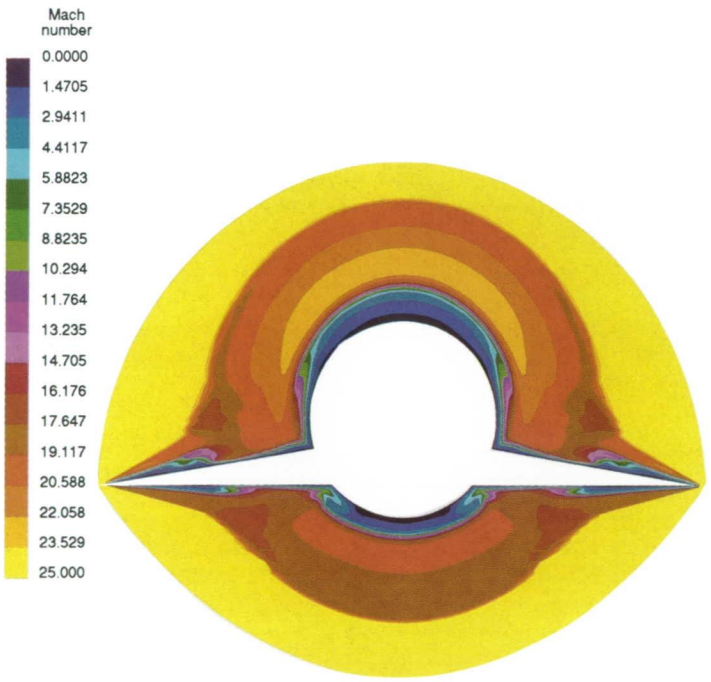


Figure 5.23. Computed Mach number contours at station 1304 in. (33.1 m). $M_\infty = 24.5$; Re = 12 000/in. (4.7×10^6/m); $\alpha = 1°$.

6. Concluding Remarks

A new algorithm for solving the three-dimensional parabolized Navier-Stokes (PNS) equations has been developed. The new algorithm is an explicit finite-difference scheme which uses upwind flux approximations for the pressure and convection terms and central differencing for the viscous and heat flux derivatives. The upwind flux approximations for the pressure and convection terms are based on the solution of an approximate Riemann problem (RP) for the PNS equations using a modification of the method proposed by Roe for steady supersonic flow of an ideal gas. Roe's method is extended to solve an approximate RP in $\mathbf{E}^*$ space for the three-dimensional PNS equations transformed into generalized coordinates and to include the subsonic pressure splitting technique of Vigneron. The algorithm is shown to capture strong shock waves without additional damping terms that depend on adjustment of solution-dependent coefficients. The execution time for the new algorithm is approximately the same as a central difference code, since the upwind differencing of the pressure and convection terms uses approximately 50 percent of the central processing unit time and doubles the Courant-Friedrichs-Lewy stability limit. The algorithm has proven to be efficient for use on vectorized computing machines since all inner and some outer do loops are vectorized.

The new algorithm is demonstrated for two- and three-dimensional supersonic and hypersonic laminar flow test cases. The test cases agree favorably with both experimental data and numerical results obtained using other numerical methods. Accurate flat plate boundary layer profiles are calculated using the new algorithm started from free-stream initial conditions. Previous numerical calculations by others of hypersonic flow over a ramp have demonstrated nonphysical pressure oscillations in the solution. The new algorithm clearly resolves the pressure field for the ramp without this difficulty. A complicated inlet flow field containing intersecting and reflecting shock waves is computed and demonstrates the robust shock capturing obtained with upwind flux approximations of the pressure and convection terms.

Multidimensional numerical algorithms using upwind differencing of the pressure and convection terms have been applied with additional dissipation terms to eliminate a loss in accuracy at certain locations in the flow field. The flow field over a cone at an angle of attack of 24° is computed using a combination of upwind and MacCormack differencing for the pressure and convection terms. This combination scheme eliminated a loss of accuracy at the symmetry plane without additional numerical dissipation. The flow field about a generic hypersonic airplane at Mach 24.5 and an angle of attack of 1° is calculated using the new algorithm. This flow field has not been previously solved using a noniterative space marching method. A special algebraic grid generation routine is used which eliminated difficulties associated with the numerical grid at the apex of the delta wings. In summary, the numerical results obtained with the new algorithm more clearly and accurately resolve the flow field features than previous results obtained with other methods for solving the PNS equations.

The research performed in the course of this study produced the following additional significant results:

1. The eigenvectors were determined for use in solving the approximate RP in $\mathbf{E}^*$-space for the three-dimensional PNS equations transformed into a generalized coordinate system and including Vigneron's splitting of the subsonic pressure gradient.
2. The difficulty associated with applying Roe's method in the subsonic region with a noniterative space marching scheme for solving the PNS equations including Vigneron's pressure splitting procedure was identified and overcome.
3. A simple method was developed for modifying the one-sided differencing in MacCormack's method into an upwind differencing scheme.
4. An increase in the stability of the scheme was obtained by solving for the value of Vigneron's pressure splitting coefficient using a cubic equation in terms of the dependent flux vector.

NASA Langley Research Center
Hampton, VA 23665-5225
November 15, 1990

Appendix A

Generalized Transformation

A general transformation was used in this study to transform the governing equations from the physical domain (x, y, z) to the computational domain (ξ, η, ζ). The transformation is made so that the governing equations can be solved on a uniformly spaced computational grid. One of the advantages of generalized transformation is that it eliminates the need to interpolate the body surface onto the numerical grid. The transformation is of the following form:

$$\left.\begin{aligned} \xi &= \xi(x) \\ \eta &= \eta(x, y, z) \\ \zeta &= \zeta(x, y, z) \end{aligned}\right\} \tag{A.1}$$

The derivations of the formulas for the metrics (ξ_x, η_x, η_y, η_z, ζ_x, ζ_y, ζ_z) of a generalized transformation are given in the text by Anderson, Tannehill, and Pletcher (1984). The formulas for the metrics of the above transformation are

$$\left.\begin{aligned} \xi_x &= J(y_\eta z_\zeta - y_\zeta z_\eta) \\ \xi_y &= \xi_z = 0 \\ \eta_x &= -J(y_\xi z_\zeta - y_\zeta z_\xi) \\ \eta_y &= J x_\xi z_\zeta \\ \eta_z &= -J x_\xi y_\zeta \\ \zeta_x &= J(y_\xi z_\eta - y_\eta z_\xi) \\ \zeta_y &= -J x_\xi z_\eta \\ \zeta_z &= J x_\xi y_\eta \end{aligned}\right\} \tag{A.2}$$

where the subscripts indicate differentiation and J is the Jacobian of the transformation

$$J = \frac{1}{x_\xi \left(y_\eta z_\zeta - y_\zeta z_\eta\right)} \tag{A.3}$$

Gielda and McRae (1986) have shown that for the PNS equations solved by MacCormack's (1969) method, the geometric conservation law (GCL) terms are not zero for any combination of possible differencing of the metrics. Therefore, the metrics are calculated once for each new space marching step using a single differencing approximation. The partial derivatives of x, y, and z with respect to η and ζ are numerically formed using second-order central differences. The partial derivative x_ξ is approximated with a first-order backward difference.

Appendix B

Eigenvalues, Eigenvectors, and Wave Strengths for the Three-Dimensional PNS Equations

The formulas used in constructing the approximate Riemann solution are given in this appendix. The formulas are valid for a Riemann problem (RP) constructed in either the ξ–η or the ξ–ζ plane when the appropriate metrics are used. The index j is used to represent the points in the plane in which the RP is being solved. The determination of the eigenvalues and the eigenvectors shown here for the three-dimensional inviscid PNS equations in generalized coordinates was accomplished in part using the symbolic manipulation language MACSYMA.

B.1. Square-Root Averaging

The matrix $\widehat{\mathbf{A}}$ defined by equation (3.7) is formed from square-root-averaged variables so that conservative properties are maintained. The square-root-averaged variables result from averaging a special parameter vector. The parameter vector $\boldsymbol{p}$ has properties which not only maintain conservation but also simplify the matrix algebra used in determining the eigenvalues and eigenvectors (see Roe 1981). The averaged components of $\boldsymbol{p}$ are defined as the square-root-averaged Cartesian velocities and enthalpy. The definition of $\boldsymbol{p}$ is

$$\boldsymbol{p}_j = \sqrt{\rho'_j}\left[1,\ u'_j,\ v'_j,\ w'_j,\ h'_j\right]^T \tag{B.1}$$

where ρ', u', v', w', h' are the original unaveraged density, Cartesian velocities, and enthalpy. The square-root-averaged velocities and enthalpies are

$$\left.\begin{aligned}
R_{j+\frac{1}{2}} &= \sqrt{\rho'_{j+1}/\rho'_j} \\
u_{j+\frac{1}{2}} &= \frac{0.5(\boldsymbol{p}_{2,j+1}+\boldsymbol{p}_{2,j})}{0.5(\boldsymbol{p}_{1,j+1}+\boldsymbol{p}_{1,j})} = \frac{R_{j+\frac{1}{2}}u'_{j+1}+u'_j}{R_{j+\frac{1}{2}}+1} \\
v_{j+\frac{1}{2}} &= \frac{R_{j+\frac{1}{2}}v'_{j+1}+v'_j}{R_{j+\frac{1}{2}}+1} \\
w_{j+\frac{1}{2}} &= \frac{R_{j+\frac{1}{2}}w'_{j+1}+w'_j}{R_{j+\frac{1}{2}}+1} \\
h_{j+\frac{1}{2}} &= \frac{R_{j+\frac{1}{2}}h'_{j+1}+h'_j}{R_{j+\frac{1}{2}}+1}
\end{aligned}\right\} \tag{B.2}$$

where the first subscript of $\boldsymbol{p}$ indicates the vector component. Unless otherwise stated, all the variables that are defined in the following are understood to be at $j+\frac{1}{2}$ and are formed using the above defined square-root-averaged variables.

B.2. Metrics

The metrics are defined so that if the nonmodified flux is forward (so the modified flux term contains elements based on the positive eigenvalues) and all the eigenvalues are positive, the resulting difference is backward. This happens if we use the metrics at $j+1$ to form $\mathbf{df}^+$ and at j to form $\mathbf{df}^-$. For example, if the

RP is being solved in ξ–η plane, the metrics would be defined as

$$\left.\begin{aligned} n_x\Big|_{j+\frac{1}{2}} &= \frac{\eta_x}{J}\Big|_m \\ n_y\Big|_{j+\frac{1}{2}} &= \frac{\eta_y}{J}\Big|_m \\ n_z\Big|_{j+\frac{1}{2}} &= \frac{\eta_z}{J}\Big|_m \end{aligned}\right\} \tag{B.3}$$

where $m = j+1$ for $\mathbf{df}^+$ and $m = j$ for $\mathbf{df}^-$. For the ξ–ζ plane, η would be replaced by ζ and the differencing would be done in the other plane.

B.3. Eigenvalues

The eigenvalues for the three-dimensional, inviscid PNS equation set in generalized coordinates are

$$\begin{aligned} \lambda_1 &= \frac{-a_2 - \sqrt{a_2^2 - 4a_1a_3}}{2a_1} \\ \lambda_2 = \lambda_3 = \lambda_4 &= \frac{\bar{v}}{u} \\ \lambda_5 &= \frac{-a_2 + \sqrt{a_2^2 - 4a_1a_3}}{2a_1} \end{aligned} \tag{B.4}$$

where

$$\begin{aligned} \bar{v} &= n_x u + n_y v + n_z w \\ a_1 &= \omega(u^2 - c^2) + \gamma(1-\omega)u^2 \\ a_2 &= (\omega+1)(-\bar{v}u + n_x c^2) - \gamma(1-\omega)\bar{v}u \\ a_3 &= \bar{v}^2 - c^2(n_x^2 + n_y^2 + n_z^2) \\ c^2 &= (\gamma-1)\left[h - \frac{1}{2}(u^2+v^2+w^2)\right] \end{aligned}$$

and ω is Vigneron's (1978) coefficient and γ is the ratio of specific heats.

B.4. Eigenvectors

The eigenvectors for the three-dimensional, inviscid PNS equations set in generalized coordinates are

$$\left.\begin{aligned} \hat{\mathbf{e}}_1 &= \begin{bmatrix} 1 \\ u + v_g R_n \\ v - n_y u R_n \\ w - n_z u R_n \\ h \end{bmatrix} \qquad \hat{\mathbf{e}}_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ -h \end{bmatrix} \qquad \hat{\mathbf{e}}_3 = \begin{bmatrix} \overline{w} \\ 0 \\ -2n_z h \\ 2n_y h \\ \overline{w}h \end{bmatrix} \\ \hat{\mathbf{e}}_4 &= \begin{bmatrix} 1 + \frac{q^2}{2h} \\ 2u \\ 2v \\ 2w \\ h + \frac{q^2}{2} \end{bmatrix} \qquad \hat{\mathbf{e}}_5 = \begin{bmatrix} 1 \\ u - v_g R_p \\ v + n_y u R_p \\ w + n_z u R_p \\ h \end{bmatrix} \end{aligned}\right\} \tag{B.5}$$

where

$$\overline{w} = n_y w - n_z v \qquad n_t^2 = n_y^2 + n_z^2$$
$$q^2 = u^2 + v^2 + w^2 \qquad v_t = \bar{v} - n_x u$$
$$v_g = \omega\bar{v} - n_x u \qquad \mathcal{L}^2 = a_2^2 - 4a_1a_3$$
$$\mathcal{N} = (1-\omega)\left[(\gamma-1)u\bar{v} + n_x c^2\right]$$
$$\mathcal{D} = 2\left\{\omega\left[(u^2-c^2)n_t^2 + v_t^2\right] + \gamma u(1-\omega)(un_t^2 - n_x v_t)\right\}$$
$$R_n = \frac{\mathcal{L}+\mathcal{N}}{\mathcal{D}} \qquad R_p = \frac{\mathcal{L}-\mathcal{N}}{\mathcal{D}}$$

B.5. Wave Strengths

The wave strengths can be determined by solving equation (3.11). This results in the following:

$$\left.\begin{aligned}
\alpha_2 &= \frac{h\,\Delta E_1^* - \Delta E_5^*}{2h} \\
S &= \frac{n_t^2 u\,\Delta E_2^* + v_g\left(n_y\,\Delta E_3^* + n_z\,\Delta E_4^*\right)}{n_t^2u^2 + v_g v_t} \\
\alpha_3 &= \frac{n_y\,\Delta E_4^* - n_z\,\Delta E_3^* - \overline{w}S}{2n_t^2 h} \\
\alpha_4 &= \frac{\alpha_3\overline{w} - \Delta E_1^* + \alpha_2 + S}{1 - \frac{q^2}{2h}} \\
\alpha_1 &= \frac{(\mathcal{L}-\mathcal{N})(S-2\alpha_4)}{2\mathcal{L}} + \frac{\mathcal{D}\left[v_t\,\Delta E_2^* - u(n_y\,\Delta E_3^* + n_z\,\Delta E_4^*)\right]}{2\mathcal{L}(n_t^2u^2 + v_g v_t)} \\
\alpha_5 &= S - \alpha_1 - 2\alpha_4
\end{aligned}\right\} \qquad \text{(B.6)}$$

$$\Delta E_m^* = \left(\widehat{E}_{j+1}^* - \widehat{E}_j^*\right)_m$$

References

Anderson, Dale A.; Tannehill, John C.; and Pletcher, Richard H. 1984: *Computational Fluid Mechanics and Heat Transfer.* Hemisphere Publ. Corp., pp. 247–251.

Beam, Richard; and Warming, R. F. 1978: An Implicit Factored Scheme for the Compressible Navier-Stokes Equations. *AIAA J.*, vol. 16, no. 4, Apr., pp. 393–402.

Burstein, Samuel Z.; and Mirin, Arthur A. 1970: Third Order Difference Methods for Hyperbolic Equations. *J. Comput. Phys.*, vol. 5, no. 3, pp. 547–571.

Chakravarthy, Sukumar R. 1987: *Development of Upwind Schemes for the Euler Equations.* NASA CR-4043.

Chaussee, D. S.; Patterson, J. L.; Kutler, P.; Pulliam, T. H.; and Steger, J. L. 1981: A Numerical Simulation of Hypersonic Viscous Flow Over Arbitrary Geometries at High Angle of Attack. AIAA-81-0050, Jan.

Colella, P.; and Woodward, P. R. 1984: The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations. *J. Comput. Phys.*, vol. 54, pp. 174–201.

Davis, R. T.; Barnett, M.; and Rakich, J. V. 1986: The Calculation of Supersonic Viscous Flows Using the Parabolized Navier-Stokes Equations. *Comput. & Fluids,* vol. 14, no. 3, pp. 197–224.

Davis, Stephen F. 1984: A Rotationally Biased Upwind Difference Scheme for the Euler Equations. *J. Comput. Phys.*, vol. 56, no. 1, pp. 65–92.

Gear, C. William 1971: *Numerical Initial Value Problems in Ordinary Differential Equations.* Prentice-Hall, Inc., pp. 27–35.

Gielda, Thomas P.; and McRae, D. Scott 1986: An Accurate, Stable, Explicit, Parabolized Navier-Stokes Solver for High Speed Flows. AIAA-86-1116, May.

Godunov, S. K. 1960: *A Difference Method for the Numerical Calculation of Discontinuous Solutions of Hydrodynamic Equations.* JPRS 7225, U.S. Dep. of Commerce, Nov. 29.

Harten, Ami 1983: High Resolution Schemes for Hyperbolic Conservation Laws. *J. Comput. Phys.*, vol. 49, no. 2, Feb., pp. 357–393.

Hirsch, Ch.; Lacor, C.; and Deconinck, H. 1987: Convection Algorithms Based on a Diagonalization Procedure for the Multidimensional Euler Equations. *A Collection of Technical Papers—AIAA 8th Computational Fluid Dynamics Conference,* June, pp. 667–676. (Available as AIAA-87-1163.)

Holden, M. S.; and Moselle, J. R. 1970: *Theoretical and Experimental Studies of the Shock Wave-Boundary Layer Interaction on Compression Surfaces in Hypersonic Flow.* ARL 70-0002 (Contract No. F33615-67-C-1298), U.S. Air Force, Jan. (Available from DTIC as AD 706 135.)

Hung, C. M.; and MacCormack, R. W. 1976: Numerical Solutions of Supersonic and Hypersonic Laminar Compression Corner Flows. *AIAA J.*, vol. 14, no. 4, Apr., pp. 475–481.

Jameson, Antony 1974: Iterative Solution of Transonic Flows Over Airfoils and Wings, Including Flows at Mach 1. *Commun. Pure & Appl. Math.*, vol. XXVII, no. 3, May, pp. 283–309.

Lawrence, Scott L.; Tannehill, J. C.; and Chaussee, Denny S. 1986: An Upwind Algorithm for the Parabolized Navier-Stokes Equations. AIAA-86-1117, May.

Lawrence, S. L.; Chaussee, D. S.; and Tannehill, J. C. 1987: Application of an Upwind Algorithm to the Three-Dimensional Parabolized Navier-Stokes Equations. AIAA-87-1112-CP, June.

Lax, Peter; and Wendroff, Burton 1960: Systems of Conservation Laws. *Commun. Pure & Appl. Math.*, vol. XIII, no. 2, May, pp. 217–237.

Lubard, Stephen C.; and Helliwell, William S. 1974: Calculation of the Flow on a Cone at High Angle of Attack. *AIAA J.*, vol. 12, no. 7, July, pp. 965–974.

MacCormack, Robert W. 1969: The Effect of Viscosity in Hypervelocity Impact Cratering. AIAA Paper No. 69-354, Apr.–May.

McDonald, H.; and Briley, W. R. 1975: Three-Dimensional Supersonic Flow of a Viscous or Inviscid Gas. *J. Comput. Phys.*, vol. 19, no. 2, Oct., pp. 150–178.

McRae, David S. 1976: A Numerical Study of Supersonic Viscous Cone Flow at High Angle of Attack. AIAA Paper No. 76-97, Jan.

Murman, Earll M.; and Cole, Julian D. 1971: Calculation of Plane Steady Transonic Flows. *AIAA J.*, vol. 9, no. 1, Jan., pp. 114–121.

Newsome, Richard W.; Walters, Robert W.; and Thomas, James L. 1987: An Efficient Iteration Strategy for Upwind/Relaxation Solutions to the Thin-Layer Navier-Stokes Equations. *A Collection of Technical Papers—AIAA 8th Computational Fluid Dynamics Conference,* June, pp. 126–142. (Available as AIAA-87-1113.)

Osher, Stanley; and Solomon, Fred 1982: Upwind Difference Schemes for Hyperbolic Systems of Conservation Laws. *Math. Comput.*, vol. 38, no. 158, Apr., pp. 339–374.

Pandolfi, Maurizio 1984: A Contribution to the Numerical Prediction of Unsteady Flows. *AIAA J.*, vol. 22, no. 5, May, pp. 602–610.

Pandolfi, Maurizio 1985: Computation of Steady Supersonic Flows by a Flux-Difference/Splitting Method. *Comput. & Fluids,* vol. 10, no. 1, pp. 37–46.

Richardson, Pamela F.; and Morrison, Joseph H. 1987: Displacement Surface Calculations for a Hypersonic Aircraft. AIAA-87-1190, June.

Roe, P. L. 1981: Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes. *J. Comput. Phys.*, vol. 43, no. 2, Oct., pp. 357–372.

Roe, Philip L. 1986: Discrete Models for the Numerical Analysis of Time-Dependent Multidimensional Gas Dynamics. *J. Comput. Phys.*, vol. 63, no. 2, pp. 458–476.

Rudman, S.; and Rubin, S. G. 1968: Hypersonic Viscous Flow Over Slender Bodies With Sharp Leading Edges. *AIAA J.*, vol. 6, no. 10, Oct., pp. 1883–1890.

Rusanov, V. V. 1970: On Difference Schemes of Third Order Accuracy for Nonlinear Hyperbolic Systems. *J. Comput. Phys.*, vol. 5, no. 3, pp. 507–516.

Schiff, Lewis B.; and Steger, Joseph L. 1979: Numerical Simulation of Steady Supersonic Viscous Flow. AIAA Paper 79-0130, Jan.

Steger, Joseph L.; and Warming, R. F. 1981: Flux Vector Splitting of the Inviscid Gasdynamic Equations With Application to Finite-Difference Methods. *J. Comput. Phys.*, vol. 40, no. 2, Apr., pp. 263–293.

Sweby, P. K. 1984: High Resolution Schemes Using Flux Limiters for Hyperbolic Conservation Laws. *SIAM J. Numer. Anal.*, vol. 21, no. 5, Oct., pp. 995–1011.

Tracy, Richard R. 1963: *Hypersonic Flow Over a Yawed Circular Cone.* Hypersonic Res. Proj. Memo. No. 69 (Contract No. DA-31-124-ARO(D)-33), Graduate Aeronautical Lab., California Inst. of Technology, Aug. 1.

Van Leer, Bram 1982: *Flux-Vector Splitting for the Euler Equations.* ICASE Rep. No. 82-30.

Vigneron, Yvon C.; Rakich, John V.; and Tannehill, John C. 1978: Calculation of Supersonic Viscous Flow Over Delta Wings With Sharp Subsonic Leading Edges. AIAA 78-1137, July.

Warming, R. F.; and Beam, Richard M. 1975: Upwind Second-Order Difference Schemes and Applications in Unsteady Aerodynamic Flows. *AIAA 2nd Computational Fluid Dynamics Conference—Proceedings,* June, pp. 17–28.

West, John W.; and Korkegi, Robert H. 1972: Supersonic Interaction in the Corner of Intersecting Wedges at High Reynolds Numbers. *AIAA J.*, vol. 10, no. 5, May, pp. 652–656.

NASA
National Aeronautics and Space Administration

Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TP-3050		

4. Title and Subtitle	5. Report Date
An Explicit Upwind Algorithm for Solving the Parabolized Navier-Stokes Equations	February 1991
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
John J. Korte	L-16753

9. Performing Organization Name and Address	10. Work Unit No.
NASA Langley Research Center Hampton, VA 23665-5225	506-80-11-01
	11. Contract or Grant No.

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
National Aeronautics and Space Administration Washington, DC 20546-0001	Technical Paper
	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

An explicit upwind algorithm has been developed for the direct (noniterative) integration of the three-dimensional parabolized Navier-Stokes (PNS) equations in a generalized coordinate system. The new algorithm uses upwind approximations of the numerical fluxes for the pressure and convection terms obtained by combining flux difference splittings (FDS) formed from the solution of an approximate Riemann problem (RP). The approximate RP is solved using a method developed by Roe that is extended to the three-dimensional PNS equations and to include Vigneron's technique of splitting the streamwise pressure gradient. The second-order upwind differencing of the flux derivatives is obtained by adding FDS to an original forward or backward differencing of the flux derivative to modify an explicit MacCormack differencing scheme into an upwind differencing scheme. The second-order upwind flux approximations, applied with flux limiters, provide a method for numerically capturing shocks without the need for additional artificial damping terms which the user must adjust. In addition, a flow field solution is presented for a generic hypersonic aircraft at a Mach number of 24.5 and angle of attack of 1°. The computed results compare well with both experimental data and numerical results from other algorithms. Computational times required for the upwind PNS code are approximately equal to an explicit PNS MacCormack code and existing implicit PNS solvers.

17. Key Words (Suggested by Authors(s))	18. Distribution Statement
Parabolized Navier-Stokes equations Upwind method Riemann solver Numerical methods Hypersonic flow Flux-difference splitting	Unclassified—Unlimited Subject Category 02, 34

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	70	A04

For sale by the National Technical Information Service, Springfield, Virginia 22161-2171