JOURNAL OF COMPUTATIONAL PHYSICS 80, 344–361 (1989)

The Relation between Flux Vector Splitting and Parabolized Schemes

CHAU-LYAN CHANG AND CHARLES L. MERKLE

Pennsylvania State University, Department of Mechanical Engineering, University Park, Pennsylvania 16802

Received September 3, 1987; revised March 18, 1988

The relationship between PNS and thin layer Navier–Stokes algorithms is used to develop traditional as well as new PNS procedures. The use of characteristics-based flux vector splitting gives rise to a parabolized system that is based on the predominant physics of the flow, while pressure-gradient-based flux vector splitting is shown to give the traditional parabolized scheme of Vigneron. Comparisons with TLNS results show the characteristics-based PNS system gives results that are at least as accurate as the more traditional pressure-gradient-split PNS system. The use of a safety factor in the pressure-gradient splitting is shown to cause inaccuracies and should be avoided. The interpretation of PNS procedures as the first sweep of a TLNS ADI procedure also suggests an obvious global pressure iteration method that is mathematically well posed and, hence, leads to an efficient rapidly converging global iteration procedure.

I. INTRODUCTION

Parabolized Navier–Stokes procedures have proven to be very popular because of their accuracy and efficiency. For many flowfields they give results that are almost identical to those obtained with the full Navier–Stokes equations even though their CPU requirements are more than an order of magnitude smaller than those needed for the complete equations. Numerous attempts have been made to extend PNS schemes to problems in which the upstream propagation of information cannot be neglected entirely by means of various "global" iteration procedures. These efforts and perhaps the most effective global iteration techniques are summarized by Davis, Barnett, and Rakich [1] and Thompson and Anderson [2].

The major difference between PNS procedures and Navier–Stokes solvers is that PNS schemes are normally formulated in terms of the steady state equations while Navier–Stokes schemes are generally formulated in terms of the time-dependent equations. Because of this, it is difficult to "extend" a PNS algorithm to a Navier–Stokes algorithm. In the present paper, we take the opposite tack. We obtain PNS algorithms as a simplification of the time-dependent Navier–Stokes algorithms. One outcome of this is that we are able to define a family of PNS approximations including one that is based upon the physical characteristics of the equations. This result is obtained by starting from a generalized flux vector split formulation of the Navier–Stokes equations. Because the arguments center on the upstream propagation of "pressure" (inviscid) effects, with no loss of generality, we

0021-9991/89 $3.00

start from the thin layer Navier–Stokes (TLNS) equations in which streamwise diffusion is ignored.

II. Solution of TLNS Equations by Flux Vector Splitting

The two-dimensional thin layer Navier–Stokes equations in generalized axisymmetric coordinates can be written as

$$\frac{\partial Q}{\partial t}+\frac{\partial E}{\partial \xi}+\frac{\partial F}{\partial \eta}=H+\frac{\partial}{\partial \eta}\left(R_1\frac{\partial \tilde{Q}_1}{\partial \eta}+R_2\frac{\partial \tilde{Q}_2}{\partial \eta}\right), \tag{1}$$

where Q, E, and F have the standard definitions:

$$Q=\frac{y}{J}\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ e\end{bmatrix},\qquad E=\frac{y}{J}\begin{bmatrix}\rho U\\ \rho uU+\xi_x p\\ \rho vU+\xi_y p\\ (e+p)\,U\end{bmatrix},\qquad F=\frac{y}{J}\begin{bmatrix}\rho V\\ \rho uV+n_x p\\ \rho vV+n_y p\\ (e+p)\,V\end{bmatrix}. \tag{2}$$

The source term, H, contains both inviscid and viscous terms associated with axisymmetric geometry,

$$H=\frac{1}{J}\begin{bmatrix}0\\ -\frac{2}{3}\eta_x\frac{\partial}{\partial \eta}\mu v\\ p-\frac{4}{3}\mu\frac{v}{y}+\frac{2}{3}\mu n_x\frac{\partial u}{\partial \eta}-\frac{2}{3}v\eta_y\frac{\partial u}{\partial \eta}\\ -\frac{2}{3}\eta_x\frac{\partial}{\partial \eta}\mu uv-\frac{2}{3}\eta_y\frac{\partial}{\partial \eta}\mu v^2\end{bmatrix}. \tag{3}$$

The viscous terms are defined by the vectors

$$\tilde{Q}_1=(\rho, u, v, e)^{\mathrm{T}},\qquad \tilde{Q}_2=(e/\rho, u^2, v^2, uv)^{\mathrm{T}} \tag{4}$$

and the matrices

$$R_1=\begin{bmatrix}0&0&0&0\\ 0&\alpha_1&\alpha_2&0\\ 0&\alpha_2&\alpha_3&0\\ 0&0&0&0\end{bmatrix},$$

$$R_2=\begin{bmatrix}0&0&0&0\\ 0&0&0&0\\ 0&0&0&0\\ \alpha_4&\frac{\alpha_1-\alpha_4}{2}&\frac{\alpha_3-\alpha_4}{2}&\alpha_2\end{bmatrix}, \tag{5}$$

in which

$$\alpha_1 = \frac{y}{J}\mu\left(\frac{4}{3}\eta_x^2 + \eta_y^2\right)$$

$$\alpha_2 = \frac{1}{3}\frac{y}{J}\mu\eta_x\eta_y$$

$$\alpha_3 = \frac{y}{J}\mu\left(\eta_x^2 + \frac{4}{3}\eta_y^2\right) \tag{6}$$

$$\alpha_4 = \frac{y}{J}\frac{\gamma k(\eta_x^2 + \eta_y^2)}{C_p}$$

In this form, the viscous dissipation in the energy equation is separated from the remaining viscous terms so that formulations for which viscous dissipation can be neglected are easily obtained by setting R_2 to zero. A further advantage of this splitting is that the matrices R_1 and R_2 contain only metric terms and properties of the gas (viscosity, thermal conductivity, and specific heats). For cases where μ, k, and C_p are constant, or nearly so, this division makes the linearization of the viscous terms particularly easy. For turbulent flows when these quantities vary rapidly, this form separates them from the dependent variables, making it possible to identify their effects on convergence more closely.

Also, we note that here we deal with the axisymmetric form of the equations. The planar form can be obtained by setting the y's inside the terms in Eqs. (2) and (6) to unity and dropping the source vector, H. Streamwise viscous derivatives are ignored in accordance with the TLNS approximation.

We consider the solution of Eq. (1) by flux vector splitting (Steger and Warming [3] and Van Leer [4]). The purpose of flux vector splitting is to separate the flux vectors E and F into parts with definite (positive and negative) eigenvalues. In general, the Jacobian of the vector E has both positive and negative eigenvectors. Its splitting can be formally indicated as

$$E = E^+ + E^-, \tag{8}$$

where the eigenvalues of the Jacobians of E^+ and E^- are positive and negative, respectively. There are an infinite number of ways to accomplish this splitting and we shall consider two specific methods. To demonstrate the approach, we consider the homogeneous case where $E = AQ$ with $A = \partial E/\partial Q$. Flux-splitting of the homogeneous vector is reduced to splitting the matrix A as

$$A = A^+ + A^-, \tag{9}$$

where the eigenvalues of A^+ are positive and those of A^- are negative. Then, from the homogeneous character, we have

$$E^+ = A^+Q \qquad \text{and} \qquad E^- = A^-Q$$

which obviously satisfies Eq. (8).

The flux vector split algorithms can be completely described without defining a precise splitting procedure other than that indicated in Eq. (8). Splitting can be applied to either or both the ξ and η operators. Because central differences are generally used in the cross-stream directions for PNS procedures, we split only the vector E. Combining Eq. (8) with Eq. (1) and discretizing in time gives the finite difference expression

$$\left\{I - D\,\Delta t + \Delta t\left[\frac{\partial}{\partial \xi}(A^+ + A^-) + \frac{\partial}{\partial \eta} B - \frac{\partial}{\partial \eta}\left(R_1 \frac{\partial}{\partial \eta} B_{v1} + R_2 \frac{\partial}{\partial \eta} B_{v2}\right)\right]\right\} \Delta Q = -\Delta t R, \tag{10}$$

where the residual R is given by

$$R = -\Delta t\left[\left(\frac{\partial E^+}{\partial \xi} + \frac{\partial E^-}{\partial \xi} + \frac{\partial F}{\partial \eta} - H - \frac{\partial}{\partial \eta}\left(R_1 \frac{\partial}{\partial \eta} \tilde{Q}_1 + R_2 \frac{\partial}{\partial \eta} \tilde{Q}_2\right)\right]. \tag{11}$$

In Eq. (10), the Jacobians D, B_{v1}, and B_{v2} are

$$D = \partial H/\partial Q, \qquad B_{v1} = \partial \tilde{Q}_1/\partial Q, \qquad \text{and} \qquad B_{v2} = \partial \tilde{Q}_2/\partial Q. \tag{12}$$

These matrices are easily computed and are not given here. In keeping with the above comments, the matrices R_1 and R_2 are not linearized. The spatial discretization in Eq. (10) need not be defined precisely except to note that it is understood that the A^+ and A^- terms must be upwind differenced and that all terms are treated consistently on both sides of the equals sign.

Efficient solution of Eq. (10) requires some sort of operator splitting. Again, in keeping with our purpose, we split the LHS operator as

$$\left\{-D\,\Delta t + \Delta t\left[\frac{\partial}{\partial \xi} A^+ + \frac{\partial}{\partial \eta} B - \frac{\partial}{\partial \eta}\left(R_1 \frac{\partial}{\partial \eta} B_{v1} + R_2 \frac{\partial}{\partial \eta} B_{v2}\right)\right]\right\}$$
$$\{I - D\,\Delta t\}^{-1}\left\{I - D\,\Delta t + \Delta t \frac{\partial}{\partial \xi} A^-\right\} \Delta Q = -\Delta t R. \tag{13}$$

Direct expansion of the LHS of Eq. (13) shows it is equal to the LHS of Eq. (10) except for terms of order Δt^2. Experience with this splitting based upon upwind differencing in the streamwise direction and central differencing in the cross-stream direction has proven to be both efficient and accurate for TLNS solutions of viscous supersonic flow [5]. This type of splitting is based upon ideas suggested by the diagonally dominant (DDADI) splitting of Lombard [6], the Gauss–Seidel method of MacCormack [7], and the LU method of Yoon and Jameson [8].

We now consider two specific flux splittings for the vector E:

(a) *Splitting based on characteristics.* The matrix A can be diagonalized by the similarity transform,

$$\Lambda = M^{-1} A M, \tag{14}$$

where M is a matrix composed of the right eigenvectors of the matrix A. The diagonal matrix, Λ, contains the four entries, $U, U, U \pm C$, where U is the contravariant velocity and $C = (\xi_x^2 + \eta_x^2)^{1/2} c$, where c is the speed of sound. A straightforward splitting of Eq. (14) was suggested by Steger and Warming [3] as

$$\begin{aligned} \Lambda^+ &= [\Lambda + |\Lambda|]^{1/2} \\ \Lambda^- &= [\Lambda - |\Lambda|]^{1/2}, \end{aligned} \tag{15}$$

where $|\Lambda|$ refers to the matrix composed of the absolute values of the elements of Λ. From Eq. (15), we readily obtain

$$A^+ = M \Lambda^+ M^{-1} \qquad A^- = M \Lambda^- M^{-1} \tag{16}$$

with $A = A^+ + A^-$ and $E^+ = A^+ Q$, $E^- = A^- Q$ for the homogeneous case.

(b) *Splitting based on the pressure.* An alternative method for splitting E is to split the pressure term into two parts as proposed by Vigneron [9],

$$E^+ = \frac{y}{J} \begin{bmatrix} \rho U \\ \rho u U + w \xi_x p \\ \rho v U + \omega \xi_y p \\ (e+p) U \end{bmatrix}, \qquad E^- = \frac{y}{J} \begin{bmatrix} 0 \\ (1-\omega)\, \xi_x p \\ (1-\omega)\, \xi_y p \\ 0 \end{bmatrix}. \tag{17}$$

If ω is chosen as

$$\omega = \gamma M_\xi^2 / [1 + (\gamma - 1)\, M_\xi^2], \tag{18}$$

where M_ξ is the streamwise Mach number, we find that the eigenvalues of $\partial E^+ / \partial Q$ and $\partial E^- / \partial Q$ are

$$\begin{aligned} \Lambda^+ &= \text{diag}(U, U, \tfrac{1}{2}\{[(\gamma+1) - \omega(\gamma-1)]\, U \pm [(\gamma-1)^2 (\omega-1)^2 U^2 + 4\omega C^2]^{1/2}\}) \\ \Lambda^- &= \text{diag}(0, 0, 0, -(\gamma-1)(1-\omega)\, U). \end{aligned} \tag{19}$$

Again, we have split the flux vector into positive and negative parts. The reader will note that the flux vector splitting in Eq. (17) is analogous to traditional PNS procedures. Again, this splitting is used on both the LHS and the RHS of Eq. (13).

Solutions of the thin layer Navier–Stokes with either the flux splitting given in Eq. (16) or that given in Eq. (17) using the ADI scheme given in Eq. (13) have proven to be efficient in terms of convergence. The converged solutions obtained with these two flux splittings are virtually identical.

III. Obtaining the PNS Procedure from the Navier–Stokes Algorithm

The above discussion focused on the TLNS equations. We now note that if the matrix A^- is identically zero (as it is in supersonic flow) the algorithm given in Eq. (13) describes an iterative marching procedure in one direction. For those cases where A^- is not zero, we can likewise obtain a "marching" procedure by ignoring the contribution of A^-. As we shall show, the traditional PNS procedures are obtained by omitting this $\partial A^-/\partial\xi$ operator. Again, we note we must maintain consistent operators on both sides of the equation, so we also drop $\partial E^-/\partial\xi$ on the RHS. With this approximation, Eq. (13) becomes the parabolic operator

$$\left\{I - D\,\Delta t + \Delta t\left[\frac{\partial}{\partial\xi}A^+ + \frac{\partial}{\partial\eta}B - \frac{\partial}{\partial\eta}\left(R_1\frac{\partial}{\partial\eta}B_{v1} + R_2\frac{\partial}{\partial\eta}B_{v2}\right)\right]\right\}\Delta Q = -\Delta t R^1, \tag{20}$$

where the modifed residual, R^1, is given by

$$R^1 = \left[\frac{\partial E^+}{\partial\xi} + \frac{\partial F}{\partial\eta} - H\frac{\partial}{\partial\eta}\left(R_1\frac{\partial}{\partial\eta}\tilde{Q}_1 + R_2\frac{\partial}{\partial\eta}\tilde{Q}_2\right)\right]^n. \tag{21}$$

We again consider the two flux splittings described above.

(a) *Pressure Gradient Splitting*

In the special case where E^+ is given by Eq. (17), Eq. (20) becomes the traditional PNS operator as given by numerous authors (see, for example, Refs. [9–13]) except that the time derivative is included. (The source term also may be treated slightly differently but in philosophy the two are precisely the same.) Because the "parabolized" Eq. (20) is now a marching equation, it is clearly better to iterate to convergence in time at each line before advancing to the next streamwise station. Thus, the value of ΔQ on the LHS of Eq. (20) is driven to zero by time marching at one station, and then the procedure "marches" to the next ξ-station and so forth. As will be shown later, this iteration can typically be driven to machine accuracy in less than 10 iterations. These results do differ somewhat from the traditional non-iterative PNS procedures, as discussed later.

(b) *Characteristics Splitting*

A similar time-dependent PNS procedure can also be obtained from the characteristics-split system given in Eq. (16). If the matrix A^- is neglected, this also becomes a marching algorithm. The previous example was likened to traditional PNS algorithms. This PNS-like procedure is more readily described by an appeal to the physics of the flow. If we consider a subsonic flow, dropping the matrix A^- corresponds to omitting the upstream-propagating acoustic wave. Our calculations

show this PNS procedure gives results that are almost identical to those based on the pressure gradient splitting that is traditionally used.

(c) *Dropping the Time Derivative*

The examples of PNS algorithms given above include the temporal derivative. Consequently, they require an iterative procedure at each station rather than a simple marching procedure as is normally used. To obtain this marching procedure, we first re-write Eq. (20) without using the delta form. By cancelling terms on the LHS with those on the RHS (given in Eq. (21)), the time-dependent algorithm becomes

$$\left\{Q + \Delta t\left(\frac{\partial}{\partial \xi} E^+ + \frac{\partial}{\partial \eta} F\right) - \Delta t H - \Delta t \frac{\partial}{\partial \eta}\left(R_1 \frac{\partial}{\partial \eta} B_{v1} + R_2 \frac{\partial}{\partial \eta} B_{v2}\right) Q\right\}^{n+1} = Q^n. \tag{22}$$

Then, by allowing Δt to go to infinity and rearranging terms slightly, we obtain the general form of the PNS procedure,

$$\frac{\partial}{\partial \xi} E^+ = A^+ \frac{\partial Q}{\partial \xi} = -\frac{\partial F}{\partial \eta} + H + \frac{\partial}{\partial \eta}\left(R_1 \frac{\partial}{\partial \eta} B_{v1} + R_2 \frac{\partial}{\partial \eta} B_{v2}\right) Q. \tag{23}$$

This clearly allows marching in ξ. It also shows that a variety of PNS algorithms can be obtained by flux vector splitting the vector E. Each splitting provides a different algorithm. In particular, the pressure gradient splitting noted above gives the widely used PNS algorithms referenced earlier. Equation (23) also shows that the characteristic flux vector splitting suggested by Steger and Warming [3] likewise gives an alternative (and more physically realistic) PNS algorithm.

The logical question to ask at this point is which of the infinitely many flux vector splittings will provide the most accurate parabolized equations if the upwind propagating operator is ignored. We know of no theoretical way to answer this quandary, but our results shown later indicate that the PNS algorithm obtained from the Steger–Warming flux splitting gives results which are essentially identical to the traditional PNS algorithm. The accuracy of both operators is verified by comparison with solutions of the complete TLNS equations.

(d) *A Comparison of Time-Iterative and Space-Marching Algorithms*

The most obvious difference between time-iterative and space-marching PNS algorithms is that the latter does not require iterations at each ξ-station while the former does. There are, however, some additional differences that should be noted. First, we compare the final converged solution from the time-iterative procedure

with that from the space-marching. From Eqs. (20) and (21), the converged solution for the time-iterative equation after ΔQ has been driven to zero is

$$\frac{\partial A^+ Q}{\partial \xi} + \frac{\partial}{\partial \eta} F - \text{V.T.} = 0 \tag{24}$$

where V.T. represents the viscous terms. The space-marching procedure obtained from Eq. (23) yields

$$A^+ \frac{\partial Q}{\partial \xi} + \frac{\partial F}{\partial \eta} - \text{V.T.} = 0. \tag{25}$$

The η-derivatives are identical in both schemes and so the viscous terms are not explicitly stated.

The difference between the two formulations appears in the ξ-derivative. The iterative approach allows the conservative form of the equations to be used, while the space-marching form is non-conservative. Using the subscript i for the ξ-direction and for simplicity considering only first-order accurate differencing, the iterative procedure results in

$$(A_i^+ Q_i - A_{i-1}^+ Q_{i-1})/\Delta\xi, \tag{26}$$

while the space-marching procedure gives

$$A_{i-1}^+ (Q_i - Q_{i-1})/\Delta\xi. \tag{27}$$

By performing back-to-back comparisons of the iterative and the space-marching procedures, we have found that the space-marching procedure (based on pressure splitting) tends to become unstable when unequal step sizes in ξ are used. In cases where stability can be maintained, the non-conservative form in a variable ξ-grid leads to global mass errors of order one. The iterative procedure works well for equally spaced or highly stretched grids in ξ. Consequently, part and sometimes all of the CPU penalty incurred by iterating can be offset by using variable (and, hence, larger) steps in ξ. A method for counteracting the global mass flux errors was suggested by Schiff and Steger [11] but does not appear to be widely used. Our calculations indicate that their method is useful but the 75 axial grid lines had to be increased to 300 to enable the modified space-marching procedure to match the conservative time-iterative method in accuracy. Space-marching calculations without the Schiff–Steger modification led to global mass flux errors of more than 50% in this high-expansion nozzle even in the absence of shocks. The advantage of using iterative procedures in space-marching schemes has also been pointed out by Newsome, Walters, and Thomas [14] in an interesting recent article. Although their work is philosophically similar to ours, it differs in detail. Their iteration is based upon Gauss–Seidel procedures rather than the forward–backward time marching used here. They also restrict space-marching procedures to Vigneron splitting based on Van Leer's flux vector splitting.

A second difference between the space-marching and time-iterative PNS procedures is the requirement for a safety factor, σ, in defining the parabolized operator. PNS solutions reported in the literature traditionally replace Eq. (18) by

$$\omega = \sigma\gamma M_\xi^2/[1 + (\gamma - 1)\, M_\xi^2]. \tag{28}$$

where σ is generally chosen as 0.85 or lower. In our tests of a space-marching algorithm, we were able to obtain stable smooth solutions only when σ was 0.85 or smaller, a value generally recommended for space-marching procedures [15]. By contrast, the time-iterative procedure remained completely stable when σ was set equal to one. (The iterative method of Ref. [14] also allows an increase in the value, σ. Their results are based on $\sigma = 0.95$.) Comparisons with TLNS solutions (presented later) show the $\sigma = 1.0$ solution is substantially more accurate than the $\sigma = 0.85$ result, regardless of whether the 0.85 result was obtained by space-marching or iterative procedures. Again, this difference would favor the use of the time-marching PNS procedure.

IV. Stability Analysis of the Time-Iterative PNS Algorithm

To validate the time-iterative algorithm discussed above, the linear stability analysis of Eq. (20) is given as follows. The amplification matrix of the variable, Q, is defined by

$$Q^{n+1} = GQ^n. \tag{29}$$

From a Von Neumann analysis, G can be found to be

$$K_1 G = K_2, \tag{30}$$

where the matrices K_1 and K_2 are

$$\begin{aligned} K_1 &= I - \Delta t D + \Delta t A^+ + i\Delta t B \sin\omega_y \\ &\quad - 2\Delta t \frac{(R_1 B_{v1} + R_2 B_{v2})}{\Delta\eta^2} (\cos\omega_y - 1) \\ K_2 &= I. \end{aligned} \tag{31}$$

and ω_y is the y-direction wavenumber. Figure 1 shows the maximum eigenvalues of the G matrix vs. wavenumber for typical supersonic and subsonic conditions. The results show Eq. (20) is unconditionally stable, and that rapid convergence can be expected for high values of CFL.

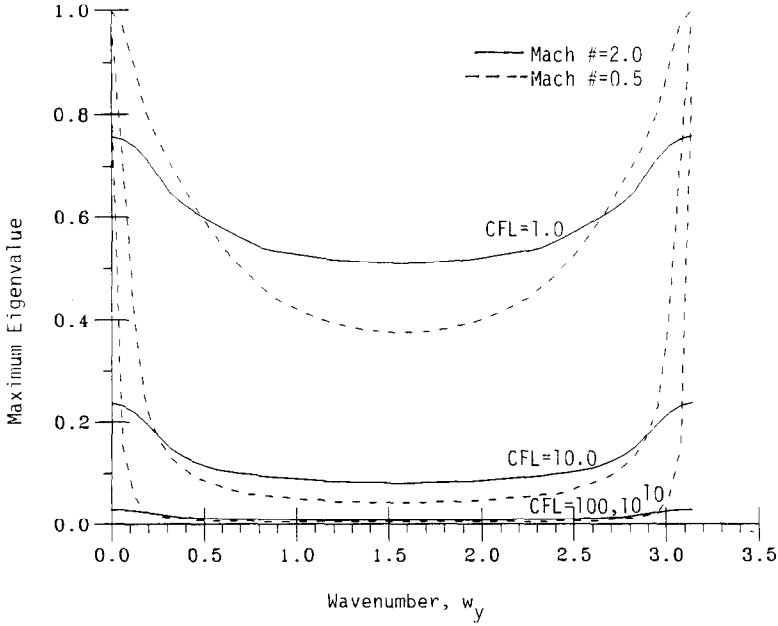

FIG. 1. Maximum eigenvalues of amplification matrix for viscous supersonic flow as a function of CFL; $\mathrm{Re} = 1.0 \times 10^5$.

V. COMPUTATIONAL RESULTS AND DISCUSSION

To test the numerical algorithms described above, the viscous supersonic flow in the diverging section of a high expansion ratio nozzle with an area ratio of 270:1 was computed with the TLNS and the PNS equations. The geometry is shown on Fig. 2 with a 75×50 grid. Calculations were made for a range of Reynolds numbers including both laminar and turbulent conditions, but the results shown here are limited to a throat Reynolds number of 10^5. The gas was treated as perfect with a ratio of specific heats of 1.4. For the inlet line a constant Mach number flow ($M = 1.02$) with zero contravariant velocity was chosen.

The numerical efficiency of the time-iterative PNS procedure is presented on Fig. 3 for representative conditions. This figure shows the rate of convergence of the time-iterative procedure at a particular ξ-station. Convergence rates are plotted for all four equations for both inviscid and viscous calculations. As these results show, convergence reached machine accuracy in less than 10 iterations, and is slightly faster for the inviscid than for the viscous equations. Acceptable convergence (three to four orders of magnitude reduction in the L_2 norm) was reached in four iterations. In general, this convergence characteristic was experienced in all our PNS calculations. The results plotted here are for the pressure gradient split PNS procedure, but are also representative of the characteristics-split system.

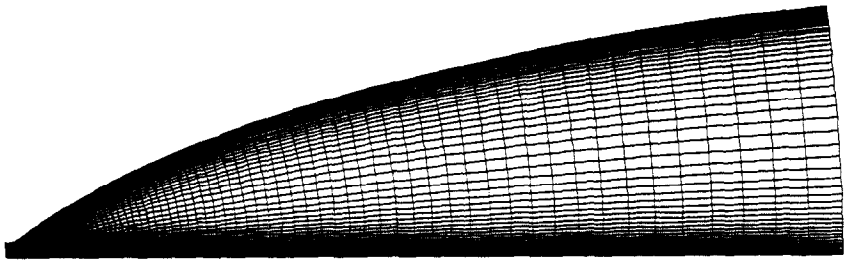

FIG. 2. Geometry and 75 (axial) × 50 (radial) grid for nozzle flow calculations, nozzle expansion ratio: 270:1.

The results of the PNS algorithm are compared with the thin layer Navier–Stokes solutions in Fig. 4. The top curve shows the Mach number contours in the nozzle as computed by the PNS procedure while the bottom curve shows similar results for the TLNS equations. As can be seen, the results are almost identical. Although not shown here, the PNS procedure based on characteristic splitting gives results that are even more similar to the TLNS results than the pressure-gradient splitting.

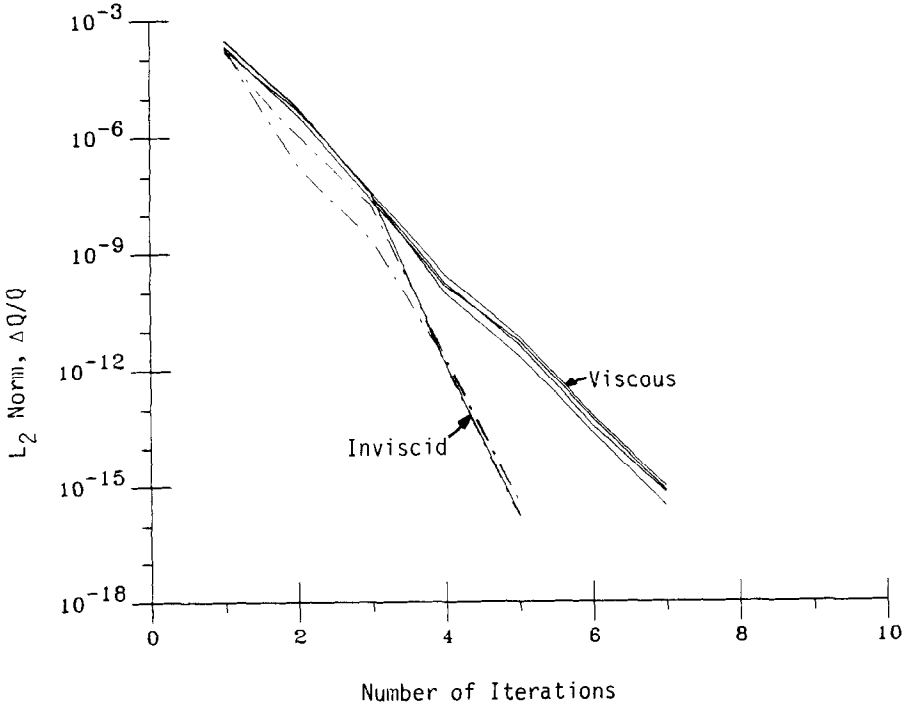

FIG. 3. Convergence rate of time-iterative PNS procedure at typical axial location.

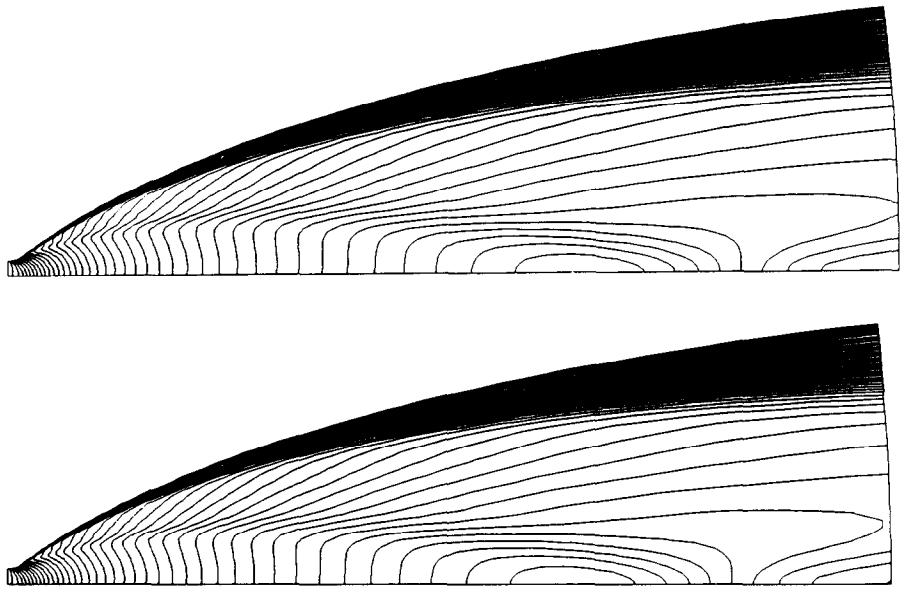

FIG. 4. Mach number contours in nozzle as computed by PNS algorithm based on pressure gradient splitting (top) and thin-layer Navier–Stokes algorithm (bottom).

A more sensitive comparison between the two PNS procedures and the TLNS procedure is given on Figs. 5 and 6. These figures show the cross-stream profiles of the pressure and velocity at the exit plane. The pressure profiles on Fig. 5 show that the two PNS procedures are in excellent agreement with the Navier–Stokes results, although the pressure-gradient-split system shows a modest overshoot near the centerline. Also shown on this figure are the results for the pressure-gradient-split system with a safety factor $\sigma = 0.85$ included. Setting this factor less than one deteriorates the accuracy of the solution.

The use of the safety factor causes the pressure to exceed the Navier–Stokes results by about 5% near the wall and to undershoot by about 25% at the centerline. The effect of setting $\sigma = 0.85$ also causes a noticeable change in the global flowfield structure as can be seen by comparing the Mach number contours for the $\sigma = 0.85$ calculation (Fig. 7) with those for the $\sigma = 1.0$ and the TLNS calculations (Fig. 4). These differences are, of course, amplified by the strong expansion that takes place in the flowfield. These and other results show the $\sigma = 1.0$ calculations are to be preferred over those with $\sigma < 1.0$. In addition, they demonstrate that the characteristics-split system is at least competitive with the pressure-gradient-split system (with $\sigma = 1.0$) in terms of accuracy.

The corresponding results for velocity are almost identical for all variations considered as can be seen on Fig. 6. Correct values of velocity in conjunction with incorrect values of static pressure are an indication that entropy (stagnation pressure) is not being properly conserved, a phenomenon that is frequently encountered in numerical schemes.

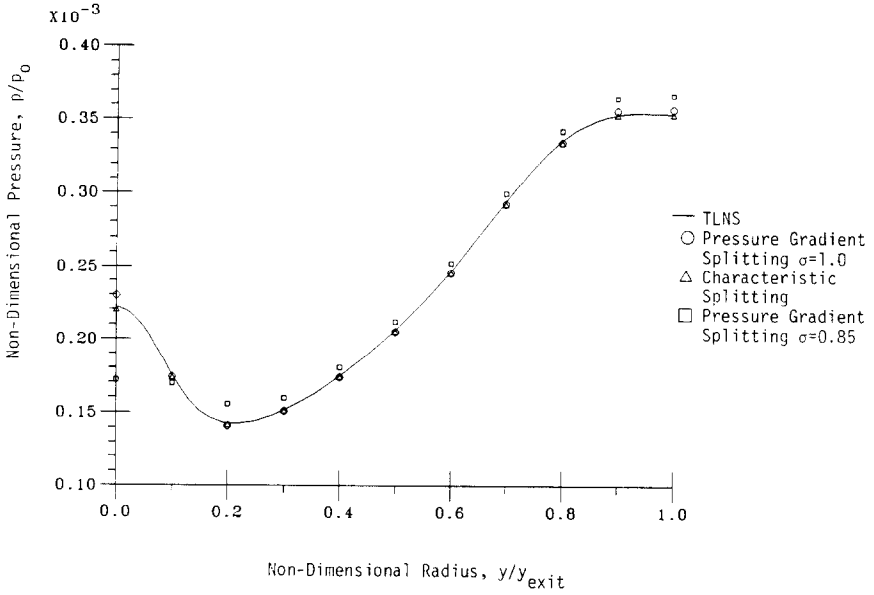

FIG. 5. Comparison of cross-stream pressure profile at exit plane for various PNS results with TLNS calculations.

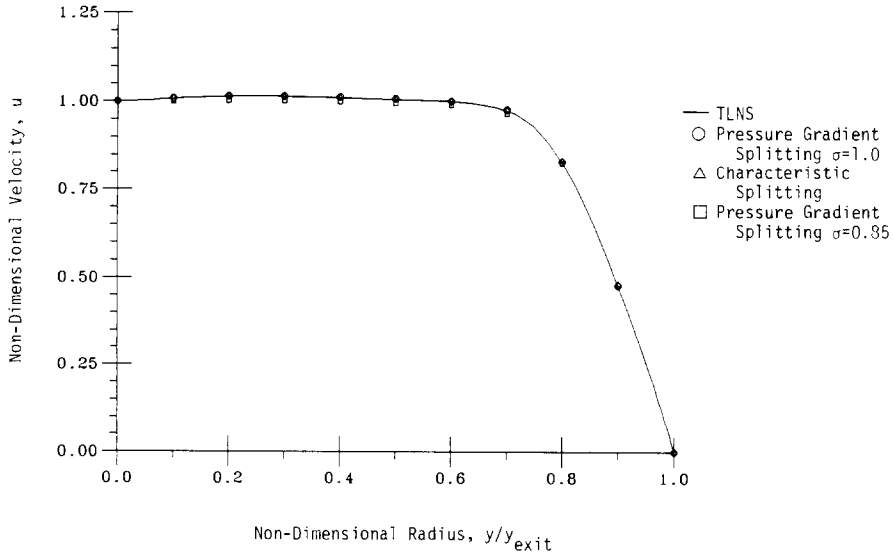

FIG. 6. Comparison of cross-stream velocity profile at exit plane for various PNS calculations with TLNS results.

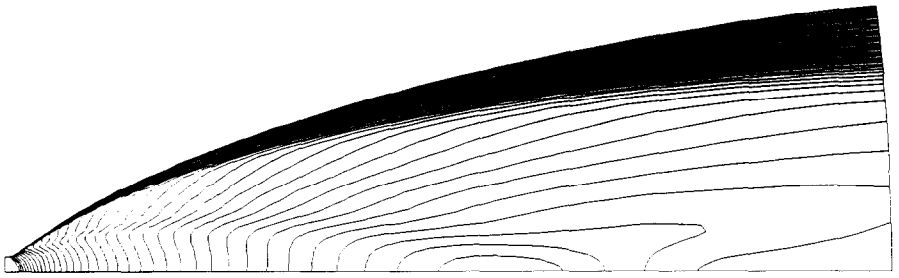

FIG. 7. Mach number contours in 270:1 expansion ratio nozzle for PNS solution-based on pressure gradient splitting and using a safety factor, $\sigma = 0.85$; $\mathrm{Re} = 1.0 \times 10^5$.

Further comparisons between the TLNS results and the three PNS calculations are given in Figs. 8 and 9. Figure 8 shows the pressure distribution on the wall. The three orders of magnitude variation in the wall pressure distribution is too strong to show the relatively minute differences between the PNS solutions, but they do show that the PNS results are globally accurate throughout the expansion process. The thickness of the subsonic layer (distance from the wall to the sonic line) is shown in Fig. 9. This figure shows the three PNS results (characteristics-split, pressure-gradient-split with $\sigma = 1.0$, and pressure-gradient-split with $\sigma = 0.85$) give almost identical locations for the sonic line. As the exit plane is approached, all PNS procedures underpredict the thickness of the subsonic layer by about 1%.

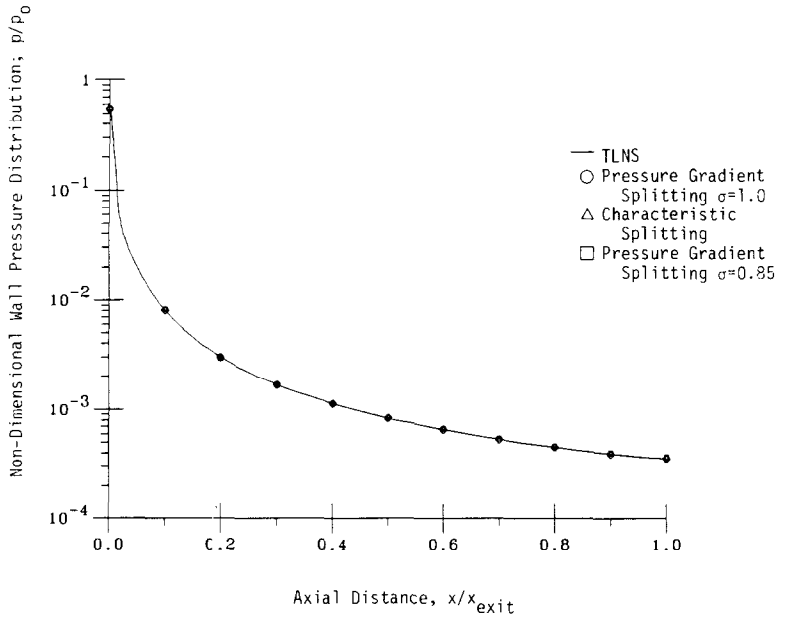

FIG. 8. Pressure distribution on the wall for TLNS and PNS procedures.

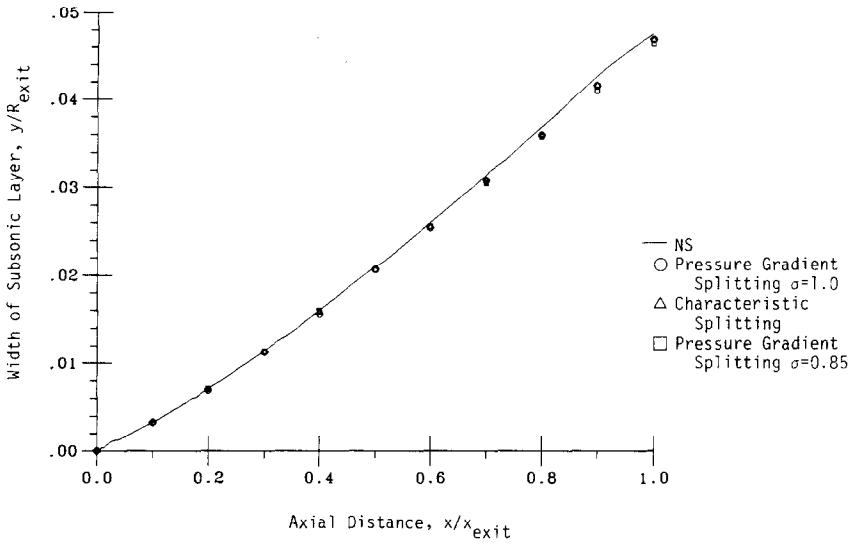

FIG. 9. Width of subsonic layer as a function of axial distance along the wall showing results of TLNS and various PNS calculations.

(Note the location of the sonic line was ascertained by linear interpolation.) This relatively accurate location of the sonic line gives further evidence of the relative and absolute accuracies of these PNS procedures.

VI. GLOBAL PNS PROCEDURE

For flows with strong viscous–inviscid interaction, the marching type PNS procedure as discussed in Section III can lead to serious errors in the numerical solution due to the suppression of the streamwise ellipticity. To allow the upstream propagation of acoustic waves inside the subsonic layer, thus preserving the streamwise elliptic behavior, the $\partial E^-/\partial\xi$ term in Eq. (10) cannot be ignored. Numerous attempts have been made to take into account the effect of $\partial E^-/\partial\xi$ by using a global pressure iteration. The basic idea is to update the pressure field by providing a stable differencing scheme for retaining the $\partial E^-/\partial\xi$ derivative in the steady state equation. This is usually done by evaluating $\partial E^-/\partial\xi$ from a forward difference and using the updated value of E^- at downstream locations, as given in the works of Rakich [10], and Lin and Rubin [16]. Davis *et al.* [1], and Barnett and Davis [17] also developed a global pressure iteration by appending a fictitious unsteady term, $\partial P/\partial\tau$ on the steady state equation, then updating the pressure field by a two-step alternating direction explicit procedure. These global pressure iterations are summarized by Thompson and Anderson [2].

In the present study, the marching type PNS procedure is derived from the unsteady TLNS equations. This suggests that the corresponding global pressure

iteration procedure is recovered by returning to the complete TLNS equations (Eq. (13)). A straightforward forward-backward iteration can be used for this purpose. Our experience is that the most effective initial condition for the fully elliptic solution is obtained by starting from a converged PNS solution. Thus, we suggest:

(a) Obtain an initial PNS solution by marching from upstream to downstream using Eq. (20).

(b) Solve Eq. (13) by the two-step ADI, i.e.,

$$\left\{I-\Delta tD+\Delta t\left[\frac{\partial}{\partial\xi}A^{+}+\frac{\partial}{\partial\eta}B-\frac{\partial}{\partial\eta}\left(R_1\frac{\partial}{\partial\eta}B_{v1}+R_2\frac{\partial}{\partial\eta}B_{v2}\right)\right]\right\}\Delta Q^{*}=-\Delta tR \tag{32}$$

$$\left[I-\Delta tD+\Delta t\frac{\partial}{\partial\xi}A^{-}\right]\Delta Q=(I-\Delta tD)\,\Delta Q^{*}. \tag{33}$$

(c) Update the flow variable, Q, according to,

$$Q^{n+1}=Q^{n}+\Delta Q \tag{34}$$

until the converged steady state is reached.

In supersonic regions, A^{-} is identically zero and the LHS operator in Eq. (33) becomes the identity matrix; hence, only Eq. (32) needs to be solved. In subsonic regions, Eq. (33) provides a mechanism to allow upstream propagation from downstream boundaries. This global iteration algorithm can be easily implemented in existing time-iterative PNS codes.

Figure 10 shows the convergence rate of this global iteration procedure applied to the same high-expansion nozzle calculation given in Section III. It requires only 110 iterations to reach machine accuracy; acceptable convergence is achieved in about 25 iterations. The convergence rate of the global iteration will, of course, depend on the degree of "ellipticity" in the problem, but because it is based upon more general concepts than the global procedures cited above, we anticipate the present procedure will be comparable to or more efficient than those for "nearly" hyperbolic problems. Further, it provides a direct procedure for going to fully elliptic problems. The convergence presented on Fig. 10 is for the characteristics-split system. Thin layer Navier–Stokes solutions based on pressure-gradient-splitting using this same procedure converge at essentially the same rate.

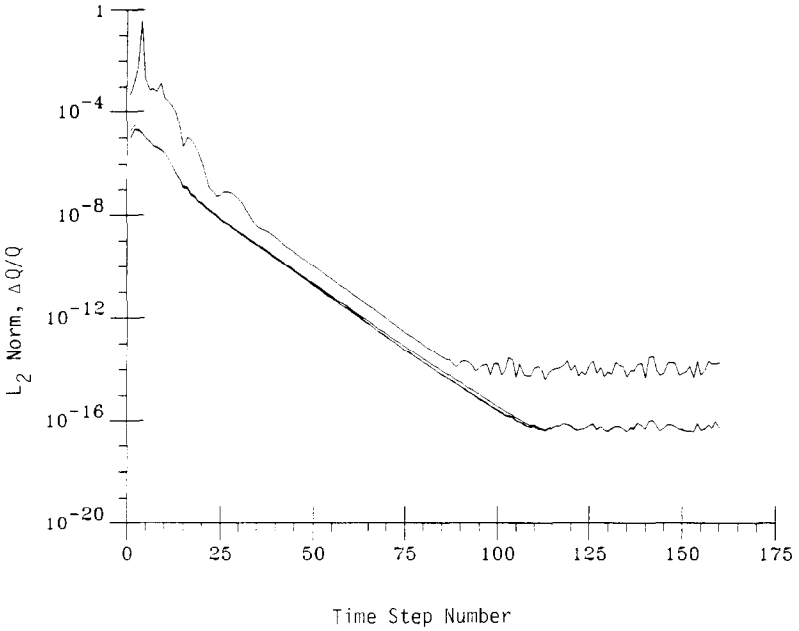

FIG. 10. Convergence of global pressure iteration for the TLNS equations for 270:1 expansion ratio nozzle.

VII. SUMMARY AND CONCLUSIONS

Traditional flux split algorithms for the thin layer Navier–Stokes equations are shown to contain the parabolized Navier–Stokes equations as a subset. If the TLNS algorithm is approximately factored into a forward sweep and a backward sweep, the forward sweep can be shown to be identical to traditional PNS formulations. Thus, by omitting the backward sweep, the TLNS formulation becomes a PNS formulation. This inter-relationship implies a rational manner for developing a "global" iteration procedure for the PNS equations.

The interpretation of one factor of flux vector split TLNS algorithms as a PNS algorithm suggests that a distinct PNS formulation can be obtained for each type of flux vector splitting considered. Two examples show this is true. The traditionally used PNS formulation is obtained by using a pressure gradient splitting to accomplish the flux vector splitting. The use of a characteristics-based flux splitting yields a PNS algorithm that includes only the downrunning characteristics while omitting the upstream propagating waves. Stability results show this characteristics-based PNS formulation is stable for space-marching and numerical results show that it provides results that are identical to the classical pressure gradient split PNS formulation and in excellent agreement with TLNS solutions.

The use of a time-based PNS formulation by necessity requires an iteration at

each ζ-plane. In comparison to non-iterative space-marching procedures, this represents a disadvantage, however, it is partially offset in that the iteration allows the ξ-derivative to be formulated in a conservative form so that variable step sizes in ξ can be readily accomodated. The iteration also obviates the need for using a safety factor when the pressure gradient splitting is used. Comparison of pressure gradient split PNS calculations with TLNS solutions showed that solutions using the safety factor were inferior to those without it. Although this result is not surprising, many space-marching codes require the use of a safety factor for stability.

Finally, the interpretation of PNS algorithms as a subset of TLNS algorithms offers a robust, stable, and efficient way to accomplish the global iteration for pressure. Computations of the TLNS system are shown to converge in some 25 iterations with each iteration requiring an amount of work commensurate with a traditional ADI sweep.

REFERENCES

1. R. T. DAVIS, M. BARNETT, AND J. V. RAKICH, *Comput. Fluids* **14**, 197 (1986).
2. D. S. THOMPSON AND D. A. ANDERSON, "A Pseudo-Unsteady Approach for Predicting Steady Supersonic Flows," AIAA Paper 87–0541, Reno, NV, 1987.
3. J. L. STEGER AND R. F. WARMING, *J. Comput. Phys.* **40**, 263 (1981).
4. B. VAN LEER, *J. Comput. Phys.* **23**, 276 (1977).
5. C.-L. CHANG, Y. KRONZON, AND C. L. MERKLE, "Time-Iterative Solutions of Viscous Supersonic Flows," AIAA Paper 87–1289, Honolulu, HI, 1987.
6. C. K. LOMBARD, E. VENKATAPATHY, AND J. BARDINA, "Universal Single Level Implicit Algorithm for Gasdynamics," AIAA Paper 84–1533, Snowmass, CO, 1984.
7. R. W. MACCORMACK, "Current Status of Numerical Solutions of the Navier–Stokes Equations," AIAA Paper 85–0032, Reno, NV, 1985.
8. S. YOON AND A. JAMESON, "An LU-SSOR Scheme for the Euler and Navier–Stokes Equations," AIAA Paper 87–0600, Reno, NV, 1987.
9. Y. C. VIGNERON, J. V. RAKICH, AND J. C. TANNEHILL, "Calculations of Supersonic Viscous Flow Over Delta Wings with Sharp Subsonic Leading Edges," AIAA Paper 78–1137, Seattle, WA, 1978.
10. J. V. RAKICH, "Iterative PNS Method for Attached Flows with Upstream Influence," AIAA Paper 83–1955, Danvers, MA, 1983.
11. L. B. SCHIFF AND J. L. STEGER, "Numerical Solution of Steady Supersonic Viscous Flow," AIAA Paper 79–0130, New Orleans, LA, 1979.
12. T. CHITSOMBOON, A. KUMAR, AND S. N. TIWARI, "Numerical Study of Finite-Rate Supersonic Combustion Using Parabolized Equations," AIAA Paper 87–0088, Reno, NV, 1987.
13. P. G. BUNING AND J. L. STEGER, "Solution of the Two-Dimensional Euler Equations with Generalized Coordinate Transformation Using Flux Vector Splitting," AIAA Paper 82–0971, Orlando, FL, 1982.
14. R. W. NEWSOME, R. W. WALTERS, AND J. L. THOMAS, "An Efficient Iteration Strategy for Upwind/Relaxation Solutions to the Thin-Layer Navier–Stokes Equations," AIAA Paper 87–1113, Honolulu, HI, 1987.
15. D. A. ANDERSON, J. C. TANNEHILL, AND R. H. PLETCHER, *Computational Fluid Mechanics and Heat Transfer* (McGraw–Hill, New York, 1984), p. 439.
16. A. LIN AND S. G. RUBIN, *AIAA J.* **20**, 1500 (1982).
17. M. BARNETT AND R. T. DAVIS, "A Procedure for the Calculation of Supersonic Flows with Strong Viscous-Inviscid Interaction," AIAA Paper 85–0166, Reno, NV, 1985.