NASA CONTRACTOR REPORT 166579

Application of the Implicit MacCormack Scheme To the Parabolized Navier-Stokes Equations

J. L. Lawrence
J. C. Tannehill
D. S. Chaussee

CONTRACT NCA2-OR340-301
May 1984

NASA

NASA CONTRACTOR REPORT 166579

Application of the Implicit MacCormack Scheme To the Parabolized Navier-Stokes Equations

J. L. Lawrence
J. C. Tannehill
D. S. Chaussee
Computational Fluid Dynamic Institute
Iowa State University
Ames, Iowa 50011

Prepared for
Ames Research Center
under Contract NCA2-OR340-301

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

TABLE OF CONTENTS

I. INTRODUCTION

Numerous studies in recent years have demonstrated the usefulness of the PNS equations in the calculation of a definable class of supersonic flows. If the inviscid region of the flow is supersonic and there is no streamwise separation of the flow, the equations of motion can be accurately modelled by a mixed set of hyperbolic-parabolic equations (the PNS equations). These equations can be solved much more efficiently than the complete Navier-Stokes equations since the solution can be marched in space rather than time.

Various versions of PNS equations have been successfully employed. One of the earliest studies involving the PNS equations was performed by Rudman and Rubin (1) in 1968. Rudman and Rubin applied a series expansion technique to the steady Navier-Stokes equations and by eliminating higher-order terms produced a system of strictly parabolic Navier-Stokes equations. A less formal approach was taken by Lubard and Helliwell (2) in which streamwise viscous stresses were assumed small in comparison with the normal viscous stresses. Thus, the Lubard-Helliwell PNS system is derived by dropping viscous terms containing partial derivatives in the streamwise direction. The retention of the pressure gradient in the streamwise momentum equation of this system is the most significant difference between the Lubard-Helliwell PNS equations and the Rudman-Rubin PNS equations. The absence of this pressure gradient term allows for stable space marching but may lead to inaccuracies in flowfields containing moderate streamwise pressure gradients. In this investigation, the more

common Lubard-Helliwell formulation is employed with the pressure gradient term being treated in a manner described in the next section.

The PNS equations have been integrated using a variety of finite-difference schemes. Because of its ease of implementation, a simple explicit scheme was used by Rudman and Rubin (1) for their calculations of the merged layer region near sharp leading edges in hypersonic viscous flow. In an effort to obtain solutions farther downstream, Rubin and Lin (3) proposed a predictor-corrector, semi-implicit, multiple-iteration scheme. Due to the larger allowable marching step size, this scheme was found to require an order-of magnitude less computer time to perform the same calculations than the explicit scheme. In their investigation of hypersonic viscous flow over cones at high angle of attack, Lubard and Helliwell (2) used an implicit differencing of the equations with a Newton-Raphson iteration technique to solve the resulting systems of nonlinear algebraic equations. In the late 1970s, noniterative, implicit, approximate-factorization schemes were developed by Vigneron et al. (4) and Schiff and Steger (5). These schemes were based on a class of ADI schemes developed by Lindemuth and Killeen (6), McDonald and Briley (7), and Beam and Warming (8). Though they require the inversion of block tridiagonal systems of linear algebraic equations in the calculation of flow properties at each step, these schemes were found to be more computationally efficient than the iterative schemes previously used, and they are the schemes most commonly employed in PNS calculations today.

In 1981, MacCormack (9) proposed an implicit scheme which requires only the inversion of block bidiagonal systems rather than block tri-

diagonal systems, thus yielding a savings in computer time and storage requirements. This scheme was designed to solve time dependent equations such as the complete Navier-Stokes equations. It is based on MacCormack's well-proven second-order accurate explicit predictor-corrector method (10) but adds an implicit procedure in the predictor-corrector sequence for points in the flow at which the local CFL number exceeds the stability limit. The method has been applied to two-dimensional internal supersonic flows (11, 12), two-dimensional external flows (13), external axisymmetric flows (14), quasi-one-dimensional flows (15), three-dimensional flow over a biconic with compression flap (16), as well as three-dimensional blunt body flows. In each of these cases, the scheme was applied to either the complete, or thin layer forms of the unsteady, Navier-Stokes equations as well as the viscous shock layer equations.

In the present work, the implicit MacCormack scheme has been modified to solve the parabolized Navier-Stokes equations. This report describes the resulting finite-difference algorithm and presents computational results for two laminar test cases. Results for the case of a flat plate boundary layer are compared with those obtained using the conventional Beam-Warming scheme as well as those obtained from a boundary-layer code. In a more severe test of the method, the hypersonic flow past a 15° compression corner has been computed. For this case, a global iteration on the pressure field of the form developed by Rakich (17) was applied in conjunction with both the implicit MacCormack scheme and the Beam-Warming scheme. Using an iteration of this type, it is possible to include

influences from downstream which are otherwise neglected in a parabolized Navier-Stokes calculation. The computed results are compared with available experimental data and a numerical solution of the complete Navier-Stokes equations.

II. GOVERNING EQUATIONS

A. Navier-Stokes Equations

The equations describing the planar flow of a Newtonian fluid are the two-dimensional, unsteady Navier-Stokes equations. These can be written in nondimensional strong-conservation-law form in Cartesian coordinates as

$$\frac{\partial U}{\partial t} + \frac{\partial (E - E_v)}{\partial x} + \frac{\partial (F - F_v)}{\partial y} = 0 \tag{1}$$

where

$$U = [\rho, \rho u, \rho v, e_t]^T$$

$$E = [\rho u, \rho u^2 + p, \rho uv, (e_t + p)u]^T$$

$$F = [\rho v, \rho uv, \rho v^2 + p, (e_t + p)v]^T$$

$$E_v = [0, \tau_{xx}, \tau_{xy}, u\tau_{xx} + v\tau_{xy} - q_x]^T$$

$$F_v = [0, \tau_{xy}, \tau_{yy}, u\tau_{xy} + v\tau_{yy} - q_y]^T$$

and

$$\tau_{xx} = \frac{\mu}{Re_\infty} \frac{2}{3} \left(2 \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} \right)$$

$$\tau_{yy} = \frac{\mu}{Re_\infty} \frac{2}{3} \left(- \frac{\partial u}{\partial x} + 2 \frac{\partial v}{\partial y} \right)$$

$$\tau_{xy} = \frac{\mu}{Re_\infty} \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)$$

$$q_x = - \frac{\mu}{Re_\infty} \frac{1}{(\gamma - 1) M_\infty^2 Pr} \frac{\partial T}{\partial x}$$

$$q_y = -\frac{\mu}{Re_\infty}\frac{1}{(\gamma - 1)M_\infty^2 Pr}\frac{\partial T}{\partial y}$$

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

The equations have been nondimensionalized (dimensional quantities are denoted by a tilde) in the following manner:

$$t = \tilde{t}/(\tilde{L}/\tilde{V}_\infty) \qquad x = \tilde{x}/\tilde{L} \qquad y = \tilde{y}/\tilde{L}$$

$$u = \tilde{u}/\tilde{V}_\infty \qquad v = \tilde{v}/\tilde{V}_\infty \qquad e = \tilde{e}/\tilde{V}_\infty^2$$

$$\rho = \tilde{\rho}/\tilde{\rho}_\infty \qquad T = \tilde{T}/\tilde{T}_\infty \qquad \mu = \tilde{\mu}/\tilde{\mu}_\infty$$

$$p = \tilde{p}/\tilde{\rho}_\infty \tilde{V}_\infty^2$$

where $\tilde{L}$ is the reference length of unity.

The Reynolds number (Re_∞) appearing in the viscous terms is given by

$$Re_\infty = \frac{\tilde{\rho}_\infty \tilde{V}_\infty \tilde{L}}{\tilde{\mu}_\infty}$$

The coefficient of thermal conductivity has been replaced by assuming a constant Prandtl number and the coefficient of viscosity is calculated using Sutherland's equation

$$\mu = T^{3/2}\left(\frac{1 + 110.4\ \tilde{K}/\tilde{T}_\infty}{T + 110.4\ K/\tilde{T}_\infty}\right)$$

Finally, the system is closed using the perfect gas equation of state which in nondimensional form becomes

$$p = \rho T/\gamma M_\infty^2$$

B. Coordinate Transformation

A transformation of the spatial coordinates of the form

$$\xi = x$$

$$\eta = \eta(x,y)$$

is applied to Eq. (1) so that the equation may be differenced on a uniform computational mesh. The resulting equation in strong-conservation-law form is

$$\frac{\partial \hat{U}}{\partial t} + \frac{\partial \hat{E}}{\partial \xi} + \frac{\partial \hat{F}}{\partial \eta} = 0 \tag{2}$$

where

$$\hat{U} = U/J$$

$$\hat{E} = \frac{1}{J}\,(E - E_v)$$

$$\hat{F} = \frac{1}{J}\,[\eta_x(E - E_v) + \eta_y(F - F_v)]$$

and J is the Jacobian of the transformation given by

$$J = \frac{\partial(\xi,\eta)}{\partial(x,y)}$$

Derivatives in the viscous vectors are transformed using the chain rule:

$$\frac{\partial}{\partial x} = \xi_x \frac{\partial}{\partial \xi} = \frac{\partial}{\partial \xi}$$

$$\frac{\partial}{\partial y} = \eta_x \frac{\partial}{\partial x} + \eta_y \frac{\partial}{\partial y}$$

C. Parabolizing Assumptions

The equations are "parabolized" to permit stable marching in space by making the following assumptions:

1. Steady flow.

2. Streamwise viscous derivatives are negligible in comparison with normal viscous derivatives. This approximation is valid for flows with high Reynolds numbers.

The following system of PNS equations is obtained as a result of these assumptions

$$\frac{\partial \bar{E}}{\partial \xi} + \frac{\partial \bar{F}}{\partial \eta} = 0 \tag{3}$$

where

$$\bar{E} = \frac{1}{J} E$$

$$\bar{F} = \frac{1}{J} [\eta_x (E - E_v) + \eta_y (F - F_v)]$$

and E_v and F_v now contain no ξ derivative terms. The PNS equations are a mixed set of hyperbolic-parabolic equations in the streamwise direction ξ provided that the inviscid flow is supersonic, the streamwise velocity component is everywhere greater than zero, and the streamwise pressure gradient term in the streamwise momentum equation is either omitted or the "departure behavior" is suppressed by a suitable technique.

D. Streamwise Pressure Gradient

The presence of the streamwise pressure gradient term in the streamwise momentum equation permits information to be propagated upstream through subsonic portions of the flowfield such as a boundary layer. As a consequence, a space-marching method of solution is not well-posed and in some cases exponentially growing solutions (departure solutions)

are encountered. A number of different techniques have been proposed to eliminate this difficulty. For this study, the method proposed by Vigneron et al. (4) is used.

The "Vigneron technique" involves splitting the $\bar{E}$ vector into two parts,

$$\bar{E} = \bar{E}^* + \bar{P}$$

where

$$\bar{E}^* = \frac{1}{J}\,[\rho u,\ \rho u^2 + \omega p,\ \rho uv,\ (e_t + p)u]^T$$

$$\bar{P} = \frac{1}{J}\,[0,\ (1 - \omega)p,\ 0,\ 0]^T$$

The $\bar{E}^*$ vector now replaces $\bar{E}$ in the numerical scheme and $\bar{P}_\xi$ is treated as a source term which is evaluated in the supersonic region. The final form of the governing equations becomes

$$\frac{\partial \bar{E}^*}{\partial \xi} + \frac{\partial \bar{F}}{\partial \eta} + \frac{\partial \bar{P}}{\partial \xi} = 0 \qquad (4)$$

An eigenvalue analysis shows that this system will be hyperbolic-parabolic if

$$\omega < \frac{\gamma M_\xi^2}{1 + (\gamma - 1)M_\xi^2}$$

where M_ξ is the local streamwise Mach number. Since this relation was determined using a linear analysis, a safety factor σ is applied and ω is calculated from

$$\omega = \frac{\sigma\gamma M_\xi^2}{1 + (\gamma - 1)M_\xi^2}$$

or is set equal to one when the computed ω is greater than one.

III. NUMERICAL SOLUTION OF EQUATIONS

A. Numerical Scheme

The numerical scheme used in the present study to integrate the PNS equations is an adaptation of the method proposed by MacCormack (9) in 1981. MacCormack demonstrated the method's usefulness for solving the full unsteady Navier-Stokes equations in an application to a two-dimensional shock-boundary-layer interaction problem. The method is implicit in nature and thus allows a much larger marching step size than explicit methods. In addition, the method possesses three advantages over conventional fully implicit methods. First, the method uses two-point, one-sided differences in the implicit part of the algorithm. Thus, block bidiagonal systems of algebraic equations result which are significantly less costly to invert than the block tridiagonal systems found in conventional methods. Second, the method employs the inviscid Jacobians and corrects them using representative viscous terms added to the Eulerian eigenvalues. This maintains stability while avoiding the expensive calculation of the complete viscous Jacobians. Finally, the algorithm allows the implicit step to be skipped in regions where the explicit stability restriction is satisfied, as in the region away from the boundary layer where mesh spacing is large. The method is stable for unbounded Δt and second-order accurate under the condition that $\frac{\mu}{\rho}\frac{\Delta t}{\min(\Delta x^2, \Delta y^2)}$ remains bounded as a 2-D Cartesian mesh is refined.

In adapting this scheme for use in solving Eq. (4) the procedure outlined in Ref. 9 is followed. Equation (4) is differentiated with

respect to ξ to obtain the equation

$$\frac{\partial}{\partial \xi}\left(\frac{\partial \bar{E}^*}{\partial \xi}\right) + \frac{\partial}{\partial \eta}\left(C \frac{\partial \bar{E}^*}{\partial \xi}\right) + \frac{\partial^2 \bar{P}}{\partial \xi^2} = 0 \tag{5}$$

which governs the propagation of the local changes, $\Delta\xi\left(\frac{\partial \bar{E}^*}{\partial \xi}\right)$. In Eq. (5), C is the Jacobian $\partial \bar{F}/\partial \bar{E}^*$. If the ξ-derivative is implicitly differenced, the following equation results

$$\left(I + \Delta\xi \frac{\partial C\cdot}{\partial \eta}\right)\left(\frac{\partial \bar{E}^*}{\partial \xi}\right)^{n+1} = \left(\frac{\partial \bar{E}}{\partial \xi}\right)^n - \left(\frac{\partial \bar{P}}{\partial \xi}\right)^{n+1} \tag{6}$$

where the dot in the equation indicates that the derivative also operates on the factor to the right. Next, define

$$\Delta\bar{E}^n = \Delta\xi\left(\frac{\partial \bar{E}}{\partial \xi}\right)^n$$

and

$$\delta\bar{E}^{*n+1} = \Delta\xi\left(\frac{\partial \bar{E}^*}{\partial \xi}\right)^{n+1}$$

so that Eq. (6) becomes

$$\left(I + \Delta\xi \frac{\partial C\cdot}{\partial \eta}\right)\delta\bar{E}^{*n+1} = \Delta\bar{E}^n - \Delta\bar{P}^{n+1} \tag{7}$$

At this point, it is observed that proceeding to difference this equation in the same manner that MacCormack used for the unsteady equations will yield a scheme which requires the modal analysis (i.e., finding the eigenvalues and eigenvectors) of the C matrix. In addition, the difficult task of decoding the $\bar{E}^*$ vector will be encountered at each step in order to obtain the flow variables.

This problem consists of solving the nonlinear set of algebraic equations

$$\begin{bmatrix} \rho u \\ \rho u^2 + \omega p \\ \rho uv \\ (e_t + p)u \end{bmatrix} = J\bar{E}^* \tag{8}$$

for the dependent variables, ρ, u, v, and p. This system can be reduced to a single quadratic equation for u-velocity, with one root of this equation being associated with a supersonic flow and the other one a subsonic flow. Solution is straightforward when it is known a priori that the flow is supersonic everywhere as when integrating the Euler equations. However, for viscous problems, the solution is more difficult due to the presence of the subsonic region of the boundary layer. For a boundary-layer calculation, the $\bar{E}^*$ vector may be readily decoded at every point normal to the wall except at the point closest to the sonic line. It is possible to determine this point in most cases by linearly interpolating the Mach number between the grid points on either side of the unknown point. The $\bar{E}^*$ vector for this point is then decoded by choosing the branch of the solution yielding a Mach number closest to the interpolated line. However, this is an undesirable method from the standpoint of computer time and code robustness.

In order to avoid this procedure, as well as the modal analysis of the C matrix, Eq. (7) is modified using the linearization

$$(\delta\bar{E}^*)^{n+1} = \Delta\xi \left(\frac{\partial \bar{E}^*}{\partial \bar{U}}\right)^n \left(\frac{\partial \bar{U}}{\partial \xi}\right)^{n+1} + O(\Delta\xi)^2$$

or

$$\delta\bar{E}^{*n+1} = A^n \delta\bar{U}^{n+1} + O(\Delta\xi)^2 \tag{9}$$

where

$$A = \frac{\partial \bar{E}^*}{\partial \bar{U}}$$

and

$$\delta\bar{U}^{n+1} = \Delta\xi \left(\frac{\partial \bar{U}}{\partial \xi}\right)^{n+1}$$

Finally, noting that

$$\frac{\partial \bar{F}}{\partial \bar{E}^*} = \left(\frac{\partial \bar{F}}{\partial \bar{U}}\right)\left(\frac{\partial \bar{E}^*}{\partial \bar{U}}\right)^{-1}$$

and substituting Eq. (9) into Eq. (7), we have

$$\left(A + \Delta\xi \frac{\partial B}{\partial \eta} \cdot\right) \delta\bar{U}^{n+1} = \Delta\bar{E}^n - \Delta\bar{P}^{n+1} \tag{10}$$

where

$$B = \frac{\partial \bar{F}}{\partial \bar{U}}$$

The equation now contains the simpler Jacobians, A and B, and the easily decoded $\bar{U}$ vector. Additional justification for this change in the dependent variable is provided by consideration of the case of a more general transformation of the streamwise coordinate. For example, a transformation of the form $\xi = \xi(x,y)$ allows the marching planes to be of general shape and orientation, which is desirable for marching along bodies with large surface slope. However, this transformation also yields an $\bar{E}^*$ vector which is virtually impossible to decode, thus making a change in

marching vector from $\bar{E}^*$ to $\bar{U}$ essential. The disadvantages of the present formulation will be discussed in a subsequent paragraph.

Differencing Eq. (10) in a manner consistent with MacCormack's original explicit scheme (10) yields the following implicit predictor-corrector algorithm for the numerical integration of Eq. (4).

Predictor:

$$\Delta\bar{E}_j^n = -\Delta\xi \frac{\Delta_+ \bar{F}_j^n}{\Delta\eta} \tag{11a}$$

$$\left(A_j^n - \Delta\xi \frac{\Delta_+}{\Delta\eta} |B|\cdot\right) \delta\bar{U}_j^{\overline{n+1}} = \Delta\bar{E}_j^n - \Delta\bar{P}_j^n \tag{11b}$$

$$\bar{U}_j^{\overline{n+1}} = \bar{U}_j^n + \delta\bar{U}_j^{\overline{n+1}} \tag{11c}$$

Corrector:

$$\Delta\bar{E}_j^{\overline{n+1}} = -\Delta\xi \frac{\Delta_- \bar{F}_j^{\overline{n+1}}}{\Delta\eta} \tag{12a}$$

$$\left(A_j^{\overline{n+1}} + \Delta\xi \frac{\Delta_-}{\Delta\eta} |B|\cdot\right) \delta\bar{U}_j^{n+1} = \Delta\bar{E}_j^{\overline{n+1}} - \Delta\bar{P}_j^n \tag{12b}$$

$$\bar{U}_j^{n+1} = \frac{1}{2}\left(\bar{U}_j^n + \bar{U}_j^{\overline{n+1}} + \delta\bar{U}_j^{n+1}\right) \tag{12c}$$

The differencing operators, $\Delta_+/\Delta\eta$ and $\Delta_-/\Delta\eta$, are defined by

$$\frac{\Delta_+ z}{\Delta\eta} = \frac{z_{j+1} - z_j}{\Delta\eta}$$

$$\frac{\Delta_- z}{\Delta\eta} = \frac{z_j - z_{j-1}}{\Delta\eta}$$

The right-hand sides of Eqs. (11a) and (12a) are calculated as in the explicit scheme. That is, one-sided differences are used for the convective terms and central differences for the viscous terms. In the present code, the differencing permutation can be reversed at each marching step depending on the boundary condition used (see subsection C).

The matrix B in Eq. (10) has been replaced with the related matrix $|B|$ in the implicit steps of the algorithm. This substitution is required in order for the block bidiagonal systems to be inverted numerically in a stable manner. The matrix $|B|$ is defined by

$$|B| = S_y^{-1} D_B S_y \tag{13}$$

where S_y^{-1} is the matrix whose rows are the left eigenvectors of the inviscid Jacobian, $B_i = \partial \bar{F}_i / \partial \bar{U}$ (see the Appendix). D_B is the diagonal matrix whose elements d_{B_i} are defined by

$$d_{B_i} = K(|\lambda_i| + \text{VISCOR})$$

where λ_i is the ith eigenvalue of B_i (Appendix). The viscous correction, VISCOR, is related to the eigenvalues of the viscous Jacobian $\partial \bar{F}_v / \partial \bar{U}$ and is given by

$$\text{VISCOR} = \frac{2\mu}{Re_\infty} \frac{\eta_x^2 + \eta_y^2}{\rho \Delta \eta}$$

The coefficient K is determined by

$$K = \max \left[1. - \frac{\sigma(\Delta\xi)_{CFL}}{\Delta\xi(1 + \frac{2}{Re_\Delta})} , 0 \right]$$

where $Re_\Delta = Re_\infty \rho u \Delta\eta / (\mu \eta_y)$ and σ is a safety factor usually set equal to

0.5. Also, $(\Delta\xi)_{CFL}$ is the largest step size which satisfies the CFL condition

$$(\Delta\xi)_{CFL} \leq \frac{\Delta\eta}{\lambda_c}$$

where λ_c is the maximum eigenvalue of $\partial\bar{F}_1/\partial\bar{E}^*$. Thus, when the step size is such that the explicit scheme is locally stable, K is set to zero and the implicit step reduces to

$$[A_j]\delta\bar{U}_j = \Delta\bar{E}_j - \Delta\bar{P}_j \tag{14}$$

This matrix equation reveals a disadvantage of marching with the $\bar{U}$ vector inasmuch as it is necessary to invert a 4×4 matrix at each point including those at which the explicit scheme is stable. The other drawback of this formulation is that the main diagonal blocks of the coefficient matrices which result cannot be easily diagonalized as MacCormack has done. A lower-upper decomposition is used here to invert the main diagonal blocks. It is believed, however, that the computer time saved by diagonalizing the coefficient matrices would be spent in decoding the $\bar{E}^*$ vector. In order to remedy the first disadvantage, steps b) and c) of Eqs. (11) and (12) may be replaced with

Predictor:

$$\bar{E}^*_j{}^{\overline{n+1}} = \bar{E}^*_j{}^n + \Delta\bar{E}_j{}^n - \Delta\bar{P}_j{}^{\bar{n}}$$

Corrector: (15)

$$\bar{E}^*_j{}^{n+1} = \frac{1}{2}\left(\bar{E}^*_j{}^n + \bar{E}^*_j{}^{\overline{n+1}} + \Delta\bar{E}_j{}^{\overline{n+1}} - \Delta\bar{P}^n\right)$$

in regions of the flow where flow properties and grid spacing are such that the marching step size satisfies the stability restriction of the explicit scheme. Generally, the explicit scheme will not be stable in the subsonic region of the boundary layer so that decoding is not a problem unless a more general transformation has been applied.

The implicit smoothing proposed by MacCormack was found to be unnecessary for the test cases of the present study. However, when capturing shocks some explicit damping is required. The pressure smoothing term of Ref. 18 was found to be sufficient to allow stable space marching. This scheme involves adding a term proportional to the quantity $(\Delta\eta)^4 |p_{\eta\eta}| U_{\eta\eta}$ so that the smoothing effect is most pronounced in regions of high pressure gradient changes. In the present code, the term is centrally differenced as

$$\Delta\bar{E}^*_j = C_x \frac{|p_{j+1} - 2p_j + p_{j-1}|}{p_{j+1} + 2p_j + p_{j-1}} \frac{(U_{j+1} - 2U_j + U_{j-1})}{J_j}$$

where C_x is proportional to the marching step size. This term is fourth order in $\Delta\eta$ so that little effect on the viscous forces is observed.

B. Computational Grid

For viscous flow cases, it is desirable to attain as much resolution of the boundary layer as possible in order that the viscous stresses may be accurately modelled. This is accomplished in the present study by clustering the grid points normal to the wall using the following clustering function

$$z(\eta) = \frac{\beta+1 - (\beta-1)\left(\frac{\beta+1}{\beta-1}\right)^{1-\eta}}{\left(\frac{\beta+1}{\beta-1}\right)^{1-\eta} + 1}$$

where $z(\eta)$ becomes equal to one when $\eta = 1$ and is zero when $\eta = 0$. The points become more tightly clustered as β approaches 1.

The physical distance, y, is then obtained from

$$y(\xi,\eta) = y_o(\xi) + \delta(\xi)\ z(\eta)$$

where $\delta(\xi)$ is the distance from the wall to the freestream edge of the grid and $y_o(\xi)$ is the value of y at the wall.

The transformation used is analytic and the metrics could therefore be determined analytically. However, experience has shown that it is desirable to calculate the terms η_x and η_y numerically in a manner consistent with the finite-difference scheme being used. For the MacCormack scheme, one-sided differences are used to evaluate the Jacobian of the transformation, J. The differencing follows that of the convective derivatives. The Jacobian can be calculated by the equation

$$J = \frac{1}{y_\eta}$$

The metrics are then computed using the relations

$$\eta_x = - Jy_\xi$$

$$\eta_y = J$$

and

$$y_\xi(\eta) = y_{o_\xi}(\xi) + \delta_\xi(\xi)\ z(\eta)$$

C. Boundary Conditions

At the wall boundary, no-slip conditions were imposed on the velocities and the temperature was specified. In order to determine the wall pressure, a zero-gradient extrapolation was explicitly applied to the pressure at the end of each predictor and corrector sweep.

At the freestream edge of the grid, zero-gradient extrapolations were explicitly applied to all of the dependent variables at the end of each predictor and corrector step.

One of the weaknesses of the present scheme lies in the difficulty of obtaining a reliable implicit boundary condition. That is, at the start of each predictor and corrector sweep a value for the vector, $\frac{\Delta\xi}{\Delta\eta}\,|B|\,\delta\bar{U}$, is required at the boundary. At boundaries where the flow properties and grid spacing are such that the explicit scheme is stable this presents no problem since $|B|$ is zero. This is generally the case at freestream boundaries. However, at solid wall boundaries in viscous flow, the grid spacing normal to the wall is generally small and implicit treatment is required. The approach generally taken is that described by MacCormack (9) in which the flux, $|B|\,\delta\bar{U}$, crossing the boundary at the end of the predictor step is reinjected into the flow to start the corrector sweep. That is,

$$\left[\frac{\Delta\xi}{\Delta\eta}\,|B|\,\delta\bar{U}\right]_1^{n+1} = [E]\left[\frac{\Delta\xi}{\Delta\eta}\,|B|\,\delta\bar{U}\right]_2^{\overline{n+1}} \tag{16}$$

where

$$[E] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Of course, the use of this boundary condition requires that the differencing permutation for the right-hand side [Eqs. (11a) and (12a)] remain the same for each step such that the predictor step is always swept toward the wall and the corrector step away from the wall.

The boundary condition described above was tested on both test cases and its performance was compared with that of the condition which simply sets $\left[\frac{\Delta\xi}{\Delta\eta}\,|B|\delta\bar{U}\right]_1^{n+1}$ equal to zero. The results will be discussed in the following section.

IV. NUMERICAL RESULTS

In order to evaluate the present implicit method for solving the PNS equations, two test cases were computed.

A. Test Case I

For the first test case, the supersonic laminar flow over a flat plate (see Fig. 1) was computed. The freestream flow conditions for this case are

$M_\infty = 2.0$

$Re_\infty/\tilde{L} = 1.99 \times 10^7/m$

$\tilde{T}_\infty = \tilde{T}_w = 233\ K$

$Pr = 0.72$

Two PNS codes were written to compute this boundary-layer test case. The first code used the conventional Beam-Warming finite-difference scheme while the second employed the implicit MacCormack scheme. The results from the two codes were compared with the results obtained from the compressible boundary-layer code of Pletcher (19). Thus, a fair evaluation of the suitability of using the implicit MacCormack scheme to solve the PNS equations could be made.

The initial conditions at $x = 1.54$ were obtained from the boundary-layer code. To obtain the results presented here, the same equally spaced grid was used in all the calculations with $\Delta y = 6.096 \times 10^{-5}$. Grid points were added to the top of the mesh as required by the growth

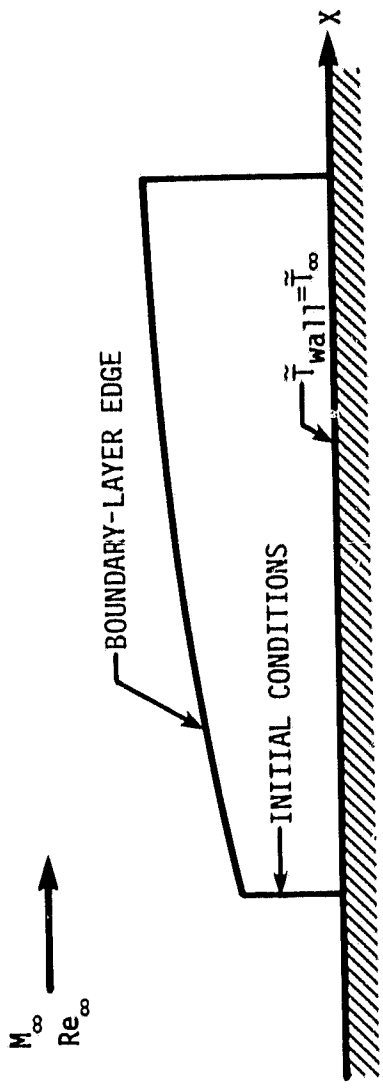

Figure 1. Flat plate boundary-layer test case

of the boundary layer. This followed the procedure of the boundary-layer code and thus allowed a point by point comparison of the results.

Profiles of tangential velocity and temperature at x = 4.57 are shown in Figs. 2 and 3, respectively. Figure 4 shows the streamwise variation of skin friction coefficient as calculated from the formula

$$C_f = \frac{\mu_{wall}}{Re_\infty} \frac{\partial u}{\partial y}$$

Plotted in these three figures are results of the boundary-layer code, the results from the Beam-Warming PNS code, and the results from the implicit MacCormack PNS code. The results are in excellent agreement. Though not shown, calculations were also performed using the explicit MacCormack scheme which proved to be unstable at a Courant number greater than 1.5. The PNS calculations were performed at a Courant number of 330.

As indicated in the previous section for the implicit MacCormack scheme, two different methods of treating the solid boundary were attempted. The first made use of the expression $\left[\frac{\Delta\xi}{\Delta\eta} |B| \delta\bar{U}\right]_1^{n+1} = 0$ which caused the scheme to go unstable at Courant numbers larger than approximately 400. The second method consists of reflection of the quantity $\frac{\Delta\xi}{\Delta\eta} |B| \delta\bar{U}$ at the wall. This condition allows the new scheme to remain stable at Courant numbers greater than 10^4 though calculated skin friction coefficients became inaccurate when the Courant number exceeded $5(10^3)$. The Beam-Warming code which employs straightforward implicit conditions at both boundaries was observed to be stable and to yield reasonable results for the skin friction coefficients at Courant numbers of more than 10^4.

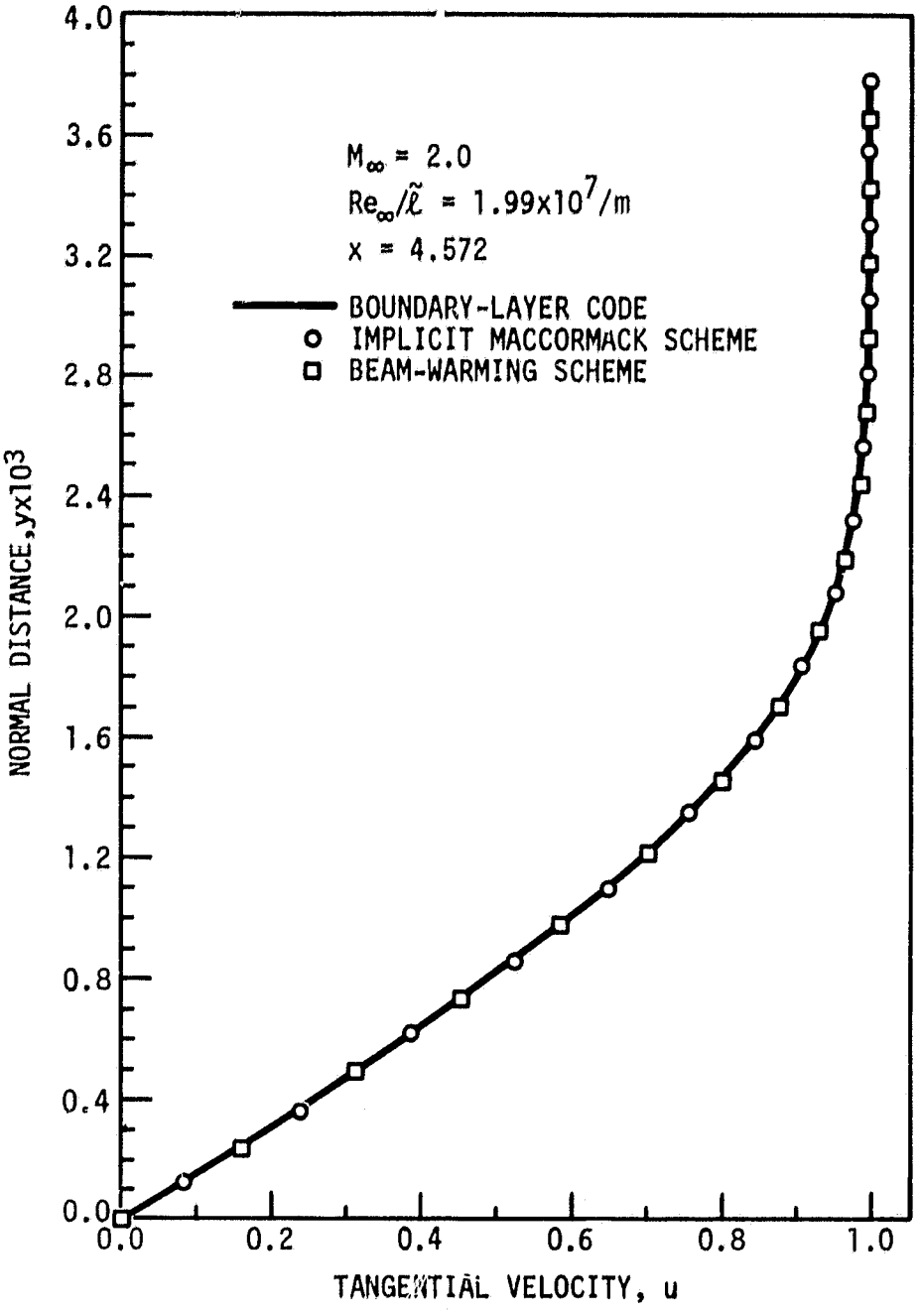

Figure 2. Comparison of velocity profiles

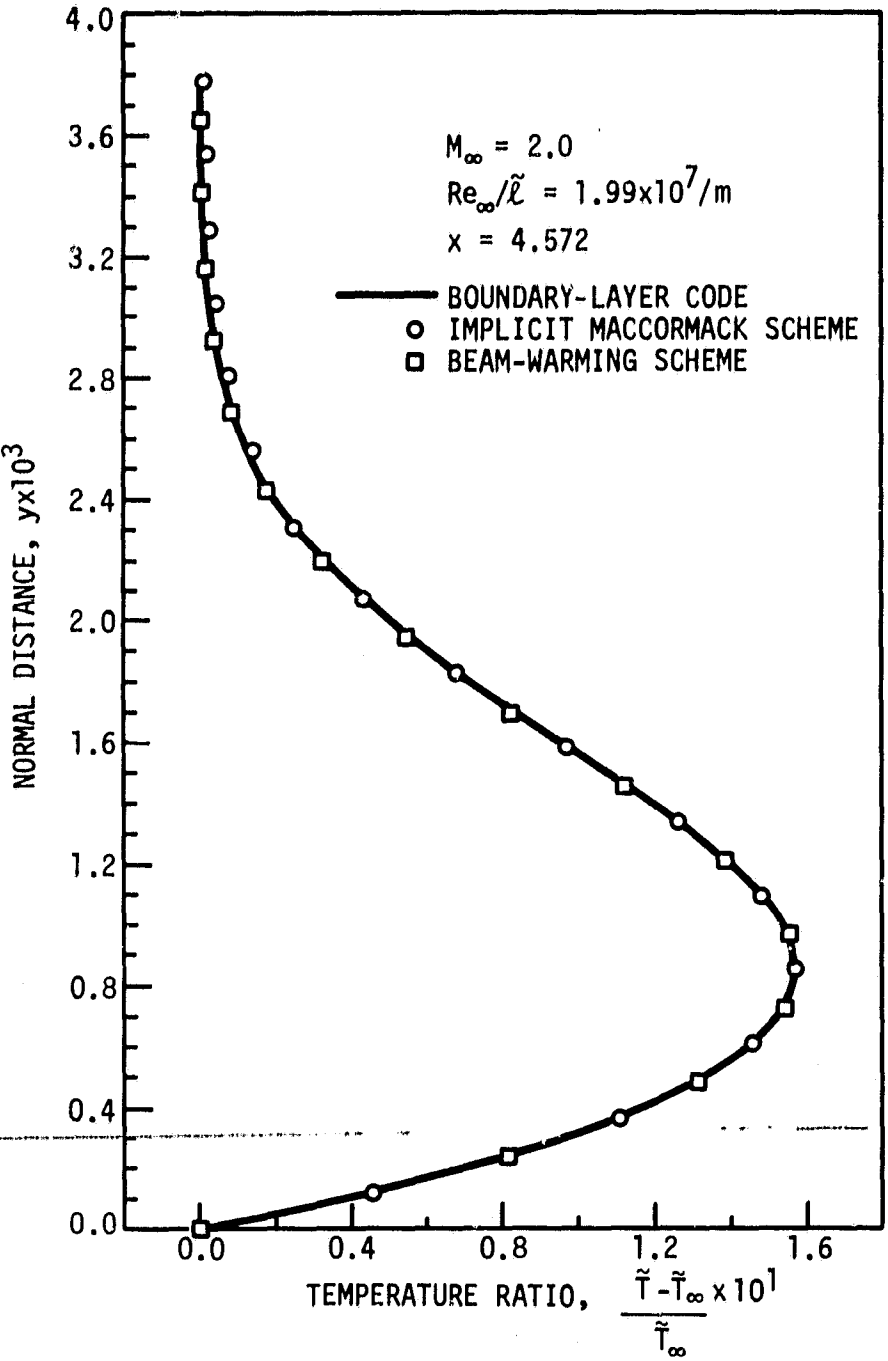

Figure 3. Comparison of temperature profiles

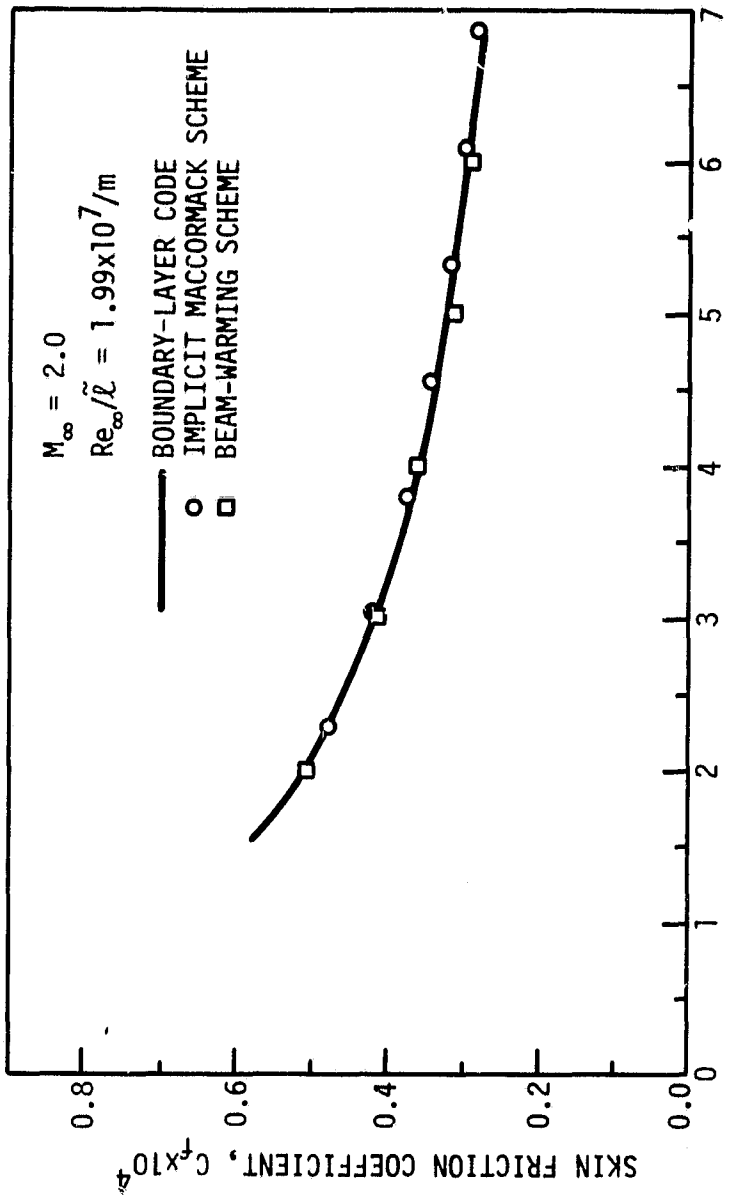

Figure 4. Comparison of skin friction coefficients

The difference in accuracy of the two schemes was originally believed to be due to errors introduced by the approximation of the viscous terms on the left-hand side of the implicit MacCormack algorithm. However, a comparison of the results obtained by using the two boundary condition procedures described above indicates the powerful effect of the boundary conditions on the stability and accuracy of the scheme. Observation of this effect leads one to conclude that the boundary condition procedure is likely to still be the dominant source of inaccuracies in implicit MacCormack scheme calculations on equally spaced grids.

In addition to these calculations, some calculations were performed on stretched grids which varied in height with the streamwise coordinate. Though the results are not included here, these calculations showed that, when using the implicit MacCormack scheme, the allowable Courant number is strongly dependent on the extent to which the grid is clustered in the normal direction. In general, at Courant numbers larger than 10^3, even slight amounts of grid stretching had catastrophic effects on the solutions. Calculations performed in the freestream with extrapolated boundary conditions (i.e., no wall boundary imposed) suggest that, with the new scheme, grid clustering has the effect of injecting nonphysical mass into the flow at the grid points. To eliminate this behavior, Hindman (20) emphasized the need to numerically solve the additional grid conservation-law equation,

$$\frac{\partial}{\partial \xi}\left(\frac{1}{J}\right) + \frac{\partial}{\partial \eta}\left(\frac{\eta_x}{J}\right) = 0$$

This equation must be solved with the same integration scheme used for the transformed equations of motion when the equations are written in strong-conservation-law form. Satisfaction of this geometric conservation law is observed to correct the problem when using the explicit MacCormack scheme; however, extension to the implicit scheme is not straightforward and nonphysical behavior still persists in regions of the flow in which the implicit steps are required for numerical stability. These errors were also present in calculations using the Beam-Warming scheme on these grids; but because this scheme is centrally differenced, the errors are much less significant.

Comparison was also made of the computer time required to perform the calculations. The Beam-Warming scheme required $9.629(10^{-4})$ sec/step/grid point of CPU time on an NAS AS/6 computer. For the implicit MacCormack scheme, $1.274(10^{-3})$ sec/step/grid point were required. It must be pointed out, however, that the latter is a worst case value. That is, since the grid is evenly spaced, all points were calculated implicitly. A timing study was performed to determine the reasons for the greater computer time of the implicit MacCormack scheme. As might be expected, a major contributor is the $|B|$ matrix calculation which must be performed twice per marching step. Another significant factor is the calculation of the right-hand side terms which also must be computed twice per step. Thus, although the study showed that two block bidiagonal systems can be inverted about 10% more quickly than one block tridiagonal system, some points must be calculated explicitly for the implicit MacCormack scheme to be as efficient in solving the PNS equations as the Beam-Warming scheme.

A comparison of the computer storage requirements of the two schemes showed that the Beam-Warming scheme required 108 K bytes of storage while the implicit MacCormack scheme required only 76 K bytes. The reason for the lower requirements of the implicit MacCormack scheme is that block tridiagonal systems need not be stored for the entire grid. The block bidiagonal systems may be formed and inverted with one sweep while storing only two 4×4 matrices at a time.

The two schemes were of about equal difficulty to code for the two-dimensional PNS equations. The difficulty of coding the viscous Jacobians in the Beam-Warming scheme is balanced by the coding of the $|B|$ matrix in the MacCormack scheme. However, if a code exists which employs the explicit MacCormack scheme, the new scheme may be implemented by simply augmenting the existing algorithm with the implicit steps. The Beam-Warming code of the present study contained about 30 percent more FORTRAN source lines than the implicit MacCormack scheme mostly as a result of the block tridiagonal solver.

B. Test Case II

The second case computed was that of hypersonic laminar flow over a 15° wedge. The flow conditions were chosen to correspond with one of the cases studied experimentally by Holden and Moselle (21) and numerically by Hung and MacCormack (18) using the complete Navier-Stokes equations. The flow conditions were

$M_\infty = 14.1$ $\qquad$ $\tilde{\ell} = 0.439$ m

$\tilde{T}_\infty = 72.2$ K $\qquad$ $Pr = 0.72$

$Re_\ell = 1.04 \times 10^5$ $\qquad$ $\tilde{T}_w = 297$ K

The Reynolds number, Re_{ℓ}, is the freestream Reynolds number based on the distance from the leading edge to the beginning of the ramp. This flow is supersonic in the inviscid region and exhibits no streamwise separation. Thus, stable space marching is allowed. The problem is illustrated schematically in Fig. 5. The grid used in the calculation is shown in Fig. 6 with every other grid line omitted. Thirty grid points were spaced normal to the wall with a stretching parameter, β, of 1.08. The grid has an initial height of 0.139 $\tilde{\ell}$ and the top of the grid is initially at an angle, ϕ_{top}, of 5° with respect to horizontal. At the beginning of the ramp, ϕ_{top} changes to 15° to follow the rise of the wall.

The initial conditions at $\tilde{x} = 0$ were provided by specifying freestream conditions everywhere except at the wall where no-slip conditions and constant wall temperature were imposed. The computation then proceeded downstream with a step size of $\Delta\xi = 3.05 \times 10^{-3}$ and was terminated after 266 steps at $\tilde{x} = 1.74\ \tilde{\ell}$. About 13 seconds of CPU time on an NAS AS/6 computer were required for the calculation. This compares with the 32 minutes of computer time on a CDC 7600 which were required by Hung and MacCormack to solve the complete Navier-Stokes equations. It should be noted that about 20 of the 30 points normal to the plate were computed explicitly by the present algorithm.

Comparison of the wall pressures on the flat plate with those computed by Hung and MacCormack is shown in Fig. 7. Also presented in this figure are the theoretical results of the strong-interaction analysis of Bertram and Blackstock (22). Good agreement is observed between the present

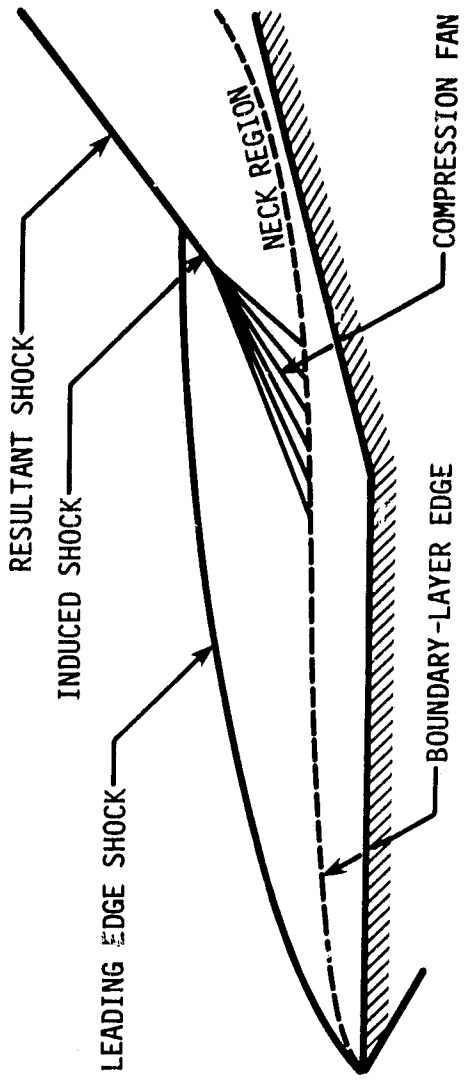

Figure 5. Hypersonic compression corner test case

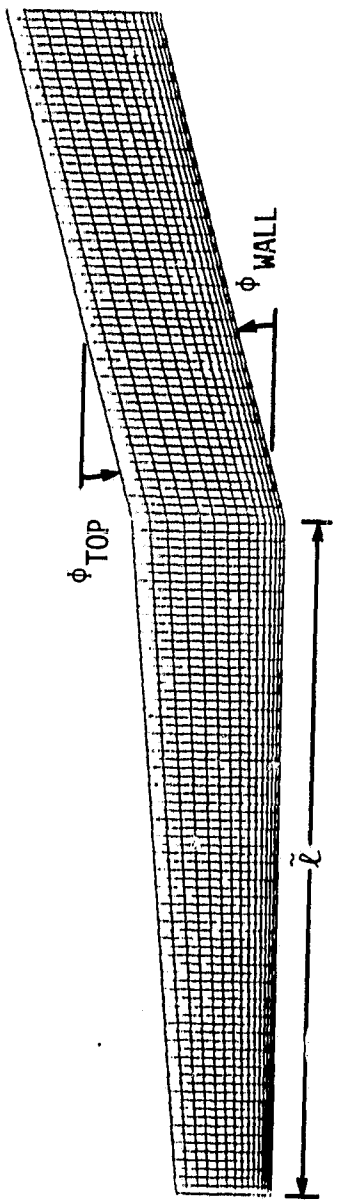

Figure 6. Computational grid

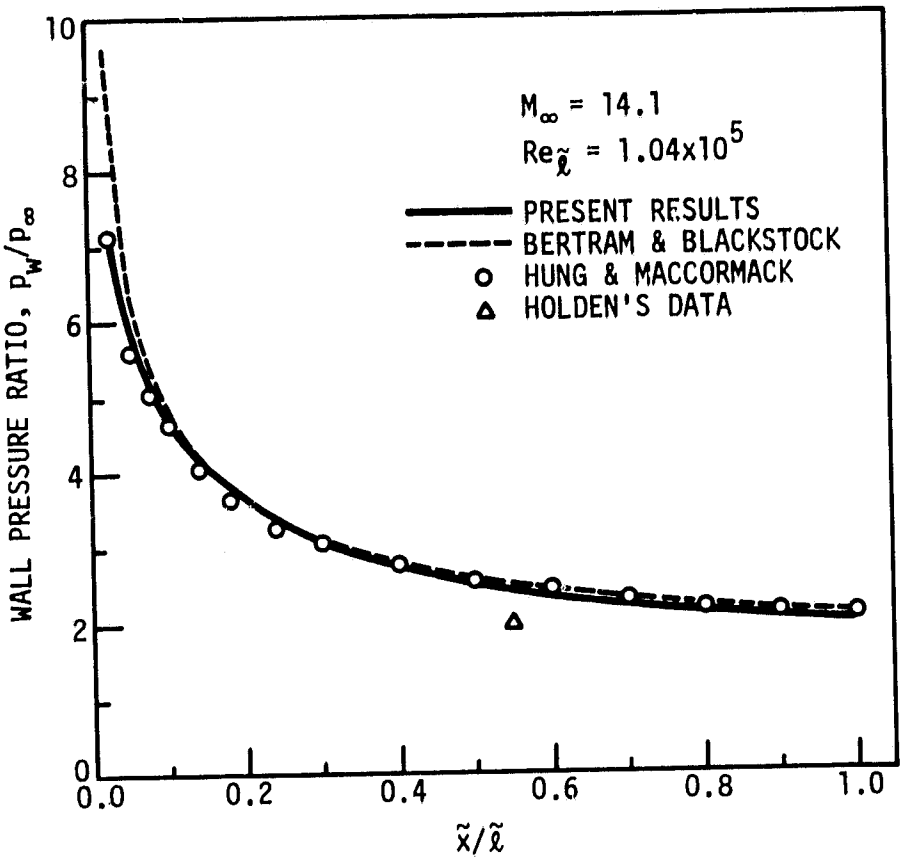

Figure 7. Wall pressure distribution on the flat plate

results and those previously obtained by computational and theoretical methods. A small disagreement with the result of the experiment of Holden and Moselle is observed. The reason for the discrepancy is unknown.

The pressure coefficients defined by $C_p = \tilde{p}_w/\tilde{\rho}_\infty \tilde{V}_\infty^2$ are compared with the previous computational and theoretical results in Fig. 8. Again, the present results compare well with those obtained by other methods. Because of the single sweep marching, there is slight disagreement in the region near the beginning of the ramp since the flow upstream is not "warned" of the oncoming compression. Nevertheless, the results downstream compare with the experiment as well as those obtained by the complete Navier-Stokes code.

As noted by Hung and MacCormack, the intersection of the leading edge shock with the shock induced by the wedge results in a Type VI interference as classified by Edney (23). This interference produces an expansion fan which eventually impinges on the wedge surface. However, this point of impingement falls downstream of the region computed in the present calculation, and therefore the peak of the computed pressure coefficient is expected to lie downstream of the final ξ station.

Heat transfer coefficients as calculated from

$$C_H = \frac{\mu_w}{Pr\ Re_\infty} \frac{\sec\theta}{\frac{\gamma-1}{2} M_\infty^2 + 1 - T_w} \frac{\partial T}{\partial y}$$

are plotted in Fig. 9. The present results show reasonable agreement with the experimental measurements, although some disagreement is observed near the beginning of the ramp.

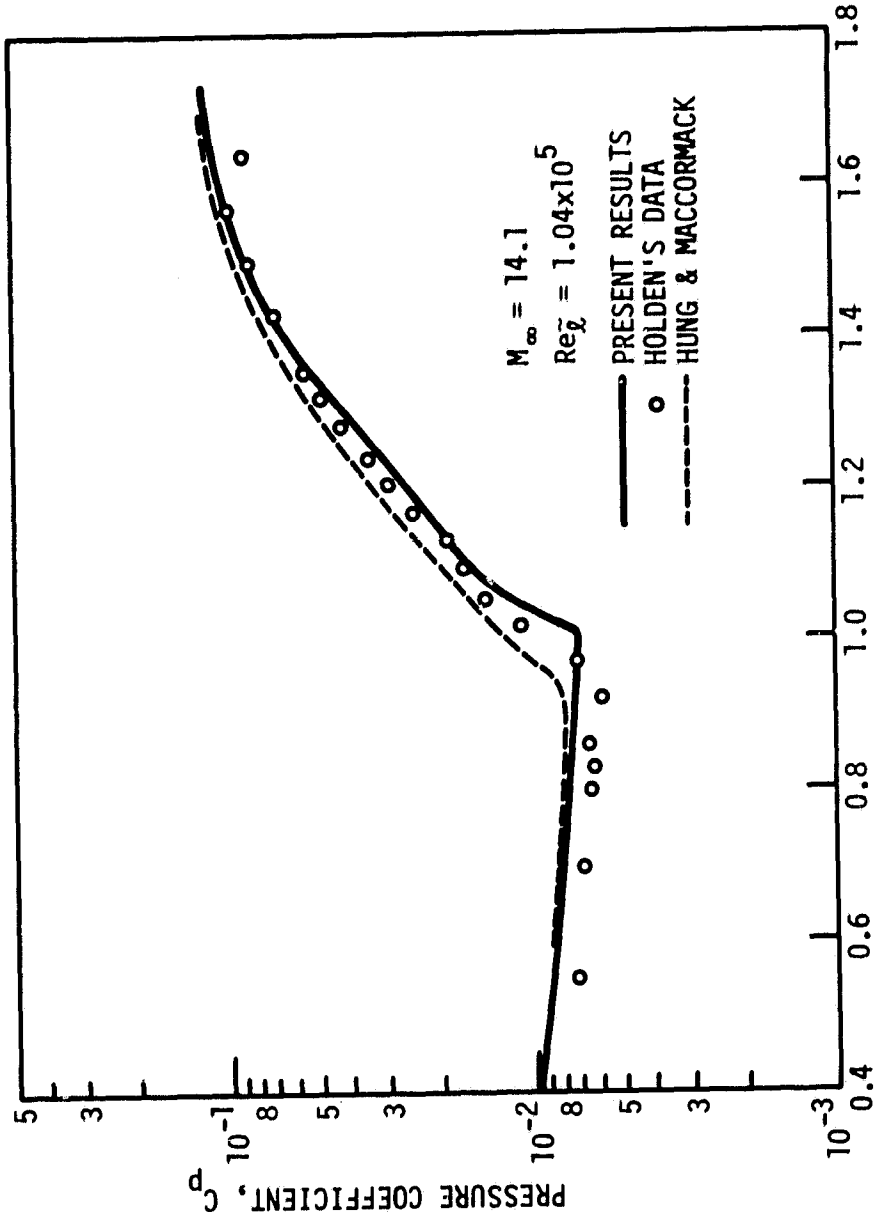

Figure 8. Comparison of wall pressure coefficients

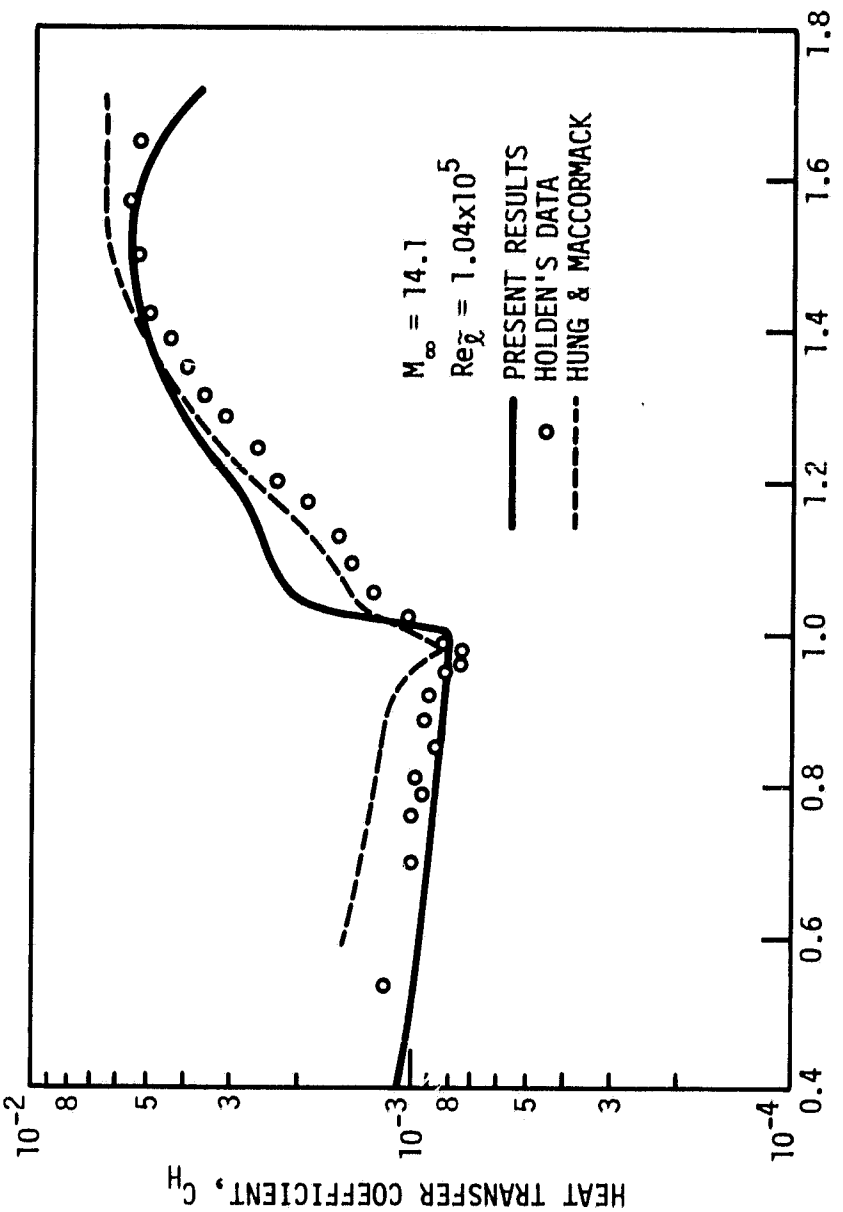

Figure 9. Comparison of heat transfer coefficients

The details of the computed flowfield are illustrated in the contour plots of Figs. 10, 11, and 12. Shock details can be seen in Fig. 10 where the relatively weak leading edge shock is represented by a single pressure contour. Pictured in Fig. 11 is the shear layer which emanates from the shock intersection as well as the strong density gradient regions in the boundary layer and at the induced shock. The Mach contours of Fig. 12 illustrate the rapid decrease in the boundary-layer thickness which results in the large increase in heat transfer to the plate shown in Fig. 9. It should be noted that the vertical coordinate in these contour plots has been unscaled by the factor $\tilde{\ell}$.

In order to include influences from downstream and, thus, hopefully correct the disagreements shown in Figs. 8 and 9, a global iteration on the pressure field was incorporated into the PNS code. PNS methods have proven effective in solving supersonic flows with weak viscous-inviscid interactions characterized by small pressure gradients in the marching direction. In the present hypersonic test case, however, large pressure gradients are generated and the associated upstream influences are therefore expected to be significant. This upstream effect is ignored in the single sweep PNS method and several researchers have turned to global iteration techniques to include these effects.

Recently, Rakich (17) developed an iterative method in which the pressure gradient term on the right side of the momentum equation is evaluated using a combination of values at two iteration levels. That is, $\Delta\bar{P}$ in Eq. (11b) would be evaluated with

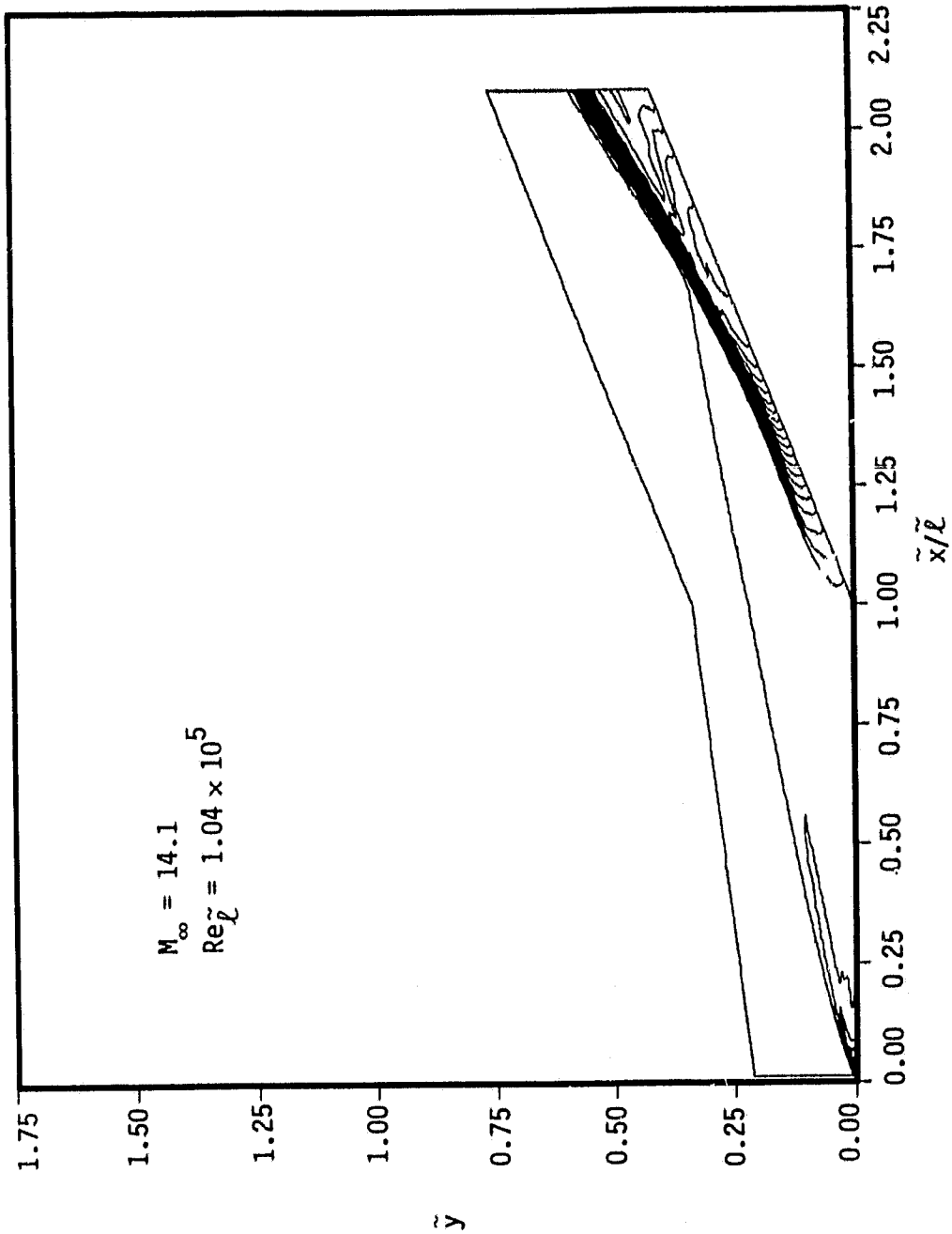

Figure 10. Pressure contours for single-sweep implicit MacCormack method

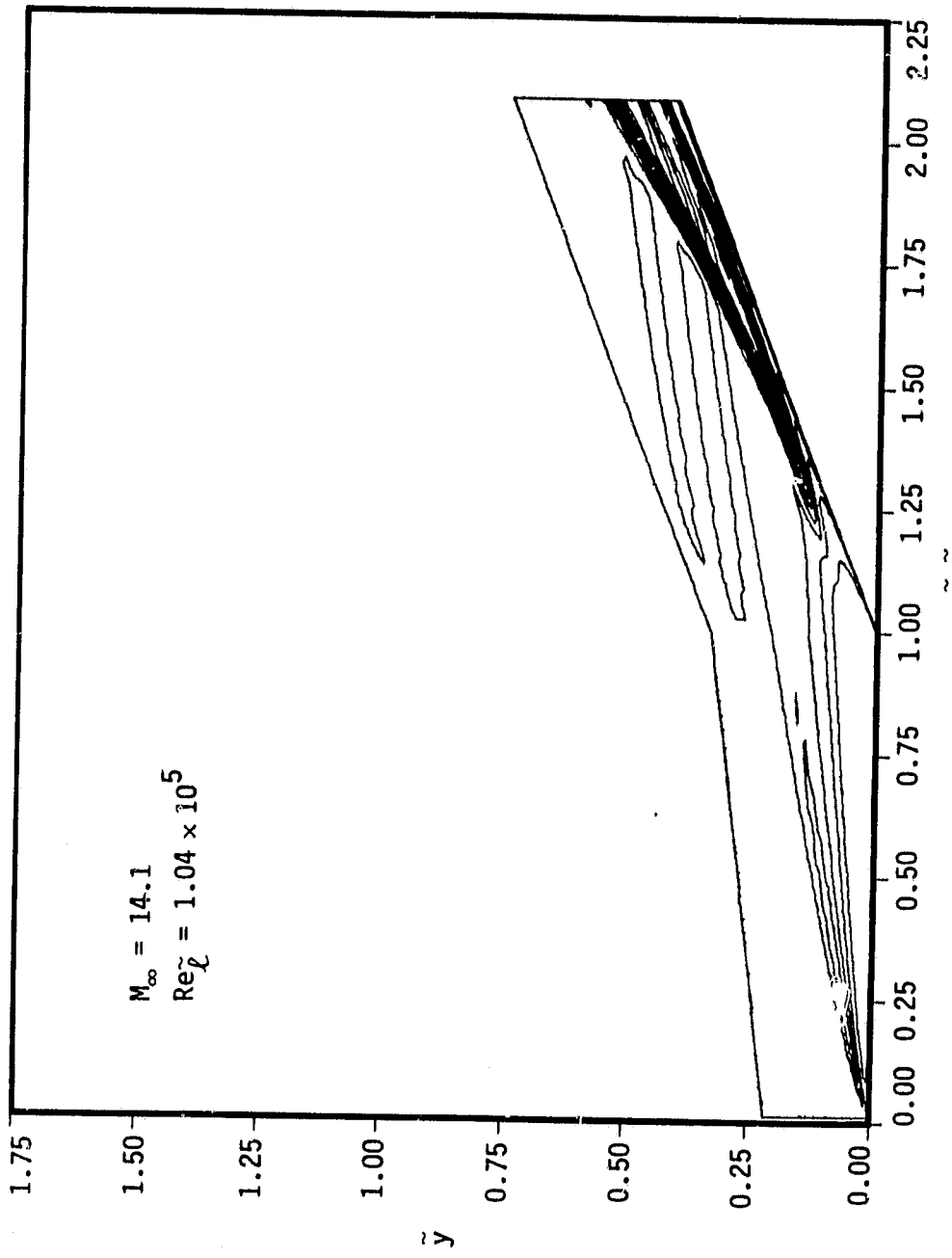

Figure 11. Density contours for single-swcep implicit MacCormack method

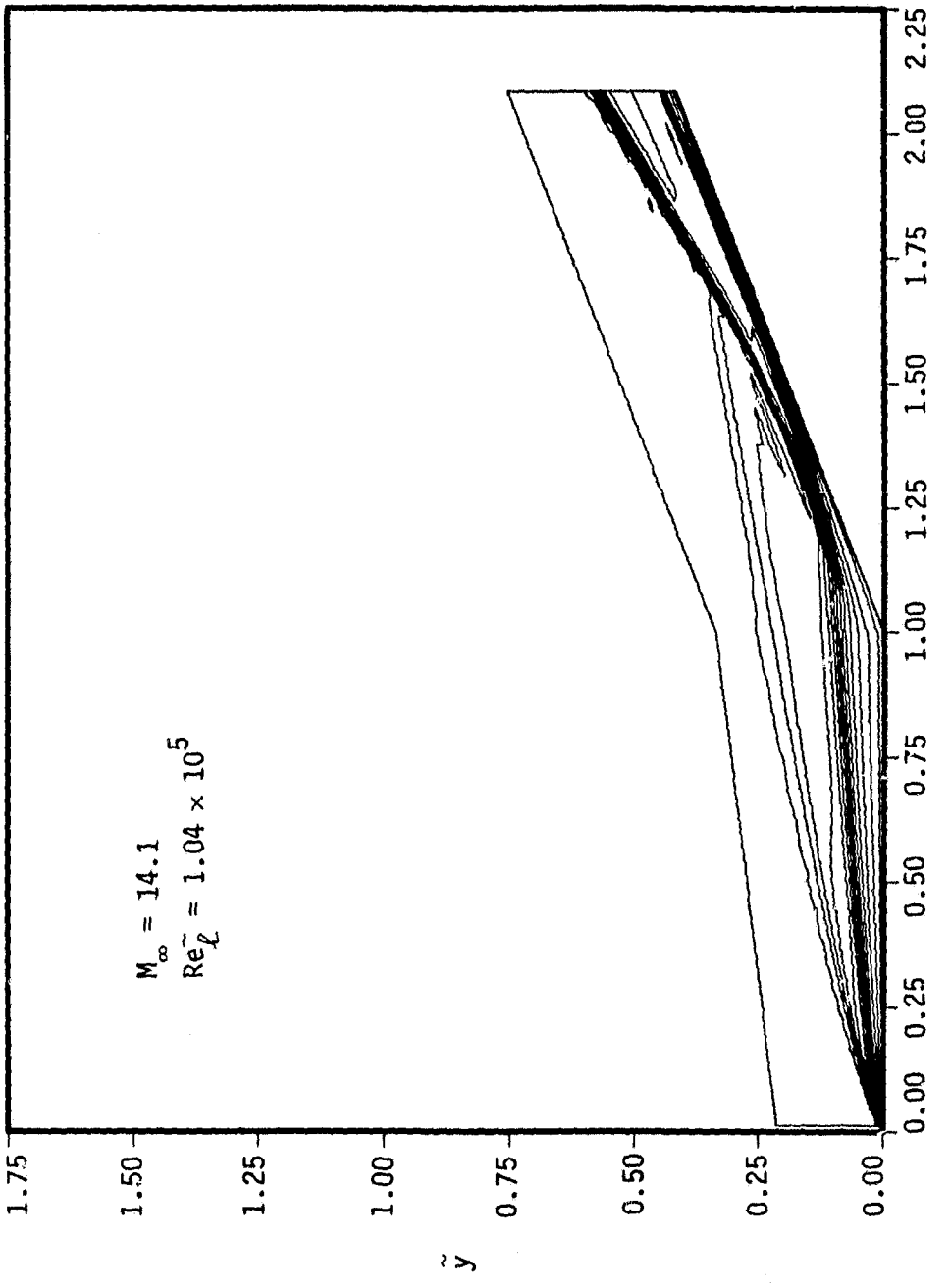

Figure 12. Mach contours for single-sweep implicit MacCormack method

$$\Delta\bar{P}^{\,\bar{n}} = \left[\, 0 \;,\quad (1-\omega)\left(\frac{p_{i-1}^{n+2}}{J} - \frac{\overline{p_i^{n+1}}}{J}\right) \;,\quad 0 \;,\quad 0 \,\right]^T$$

where the subscript i indicates the present iteration level. It is necessary here to move the implicit part of the forward difference, $\overline{p_i^{n+1}}$, to the left-hand side in Eq. (11b). This is done through a series of algebraic manipulations after which the form of the term added to the right-hand sides of Eqs. (11b) and (12b) is

$$\Delta\bar{P} = \left[\, 0 \;,\quad -(1-\omega)\left(\frac{p_{i-1}^{n+2}}{J} - \frac{p_i^{n}}{J}\right) \;,\quad 0 \;,\quad 0 \,\right]^T \tag{17}$$

The effect on the left-hand side is to change $\bar{E}^*$ to

$$\bar{E}^{\#} = \frac{1}{J}\left[\, \rho u,\ \rho u^2 + (2\omega - 1)p,\ \rho uv,\ (e_t + p)u \,\right]^T$$

The new left-hand sides of Eqs. (11b) and (12b) are then obtained by simply replacing ω in the A Jacobian with the term $(2\omega - 1)$.

This iterative method is implemented by taking the first sweep with the standard PNS code. In subsequent iterations, $\bar{E}^*$ is replaced with $\bar{E}^{\#}$ and $\Delta\bar{P}$ is evaluated according to Eq. (17). Only the pressure field is stored after each iteration. Following Rakich's technique for the outflow boundary condition, the pressure gradient is kept constant and equal to the gradient calculated at the end of the first sweep.

The global iteration applied with the implicit MacCormack scheme converged very rapidly. Figures 13 and 14 show pressure and heat transfer coefficient results of calculations which converged in 6 steps and required 70 seconds of CPU time on an NAS AS/6 computer. These figures show that the converged results deviate only slightly from the results

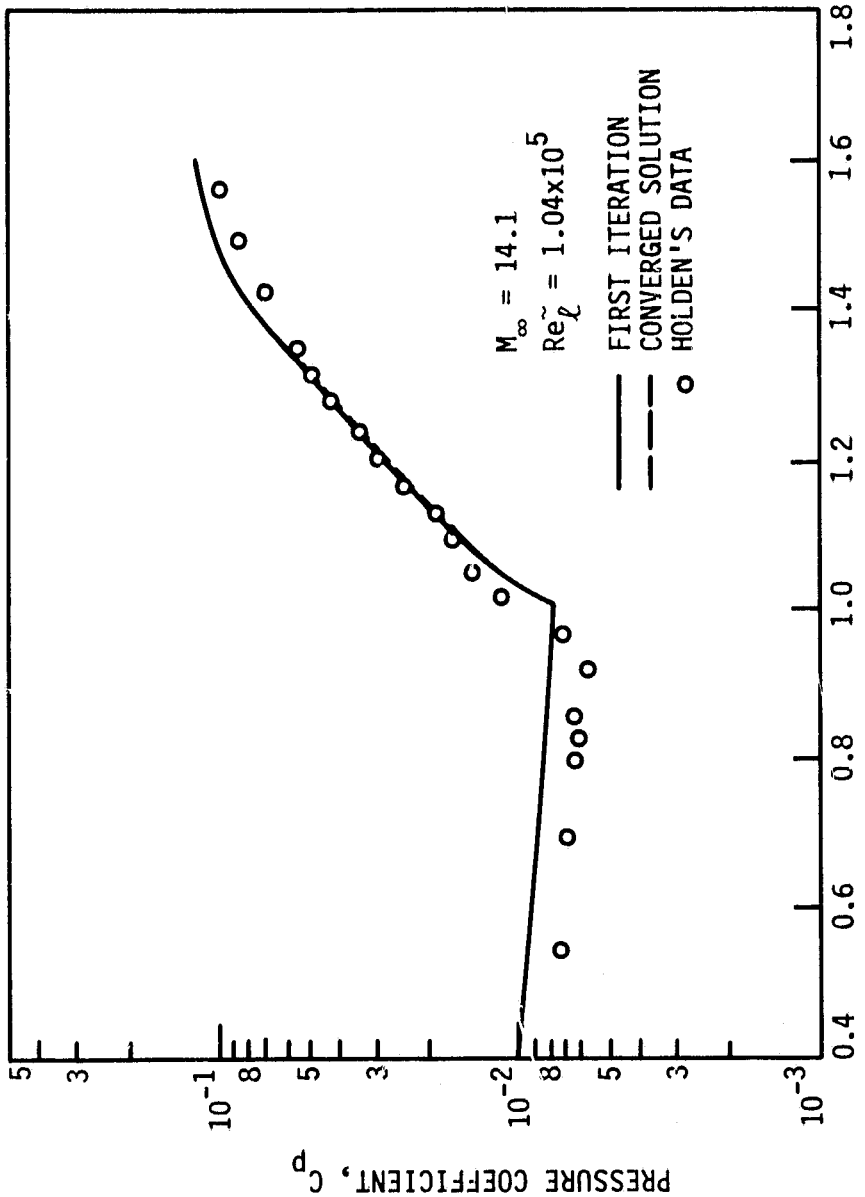

Figure 13. Pressure coefficients for iterated implicit MacCormack method

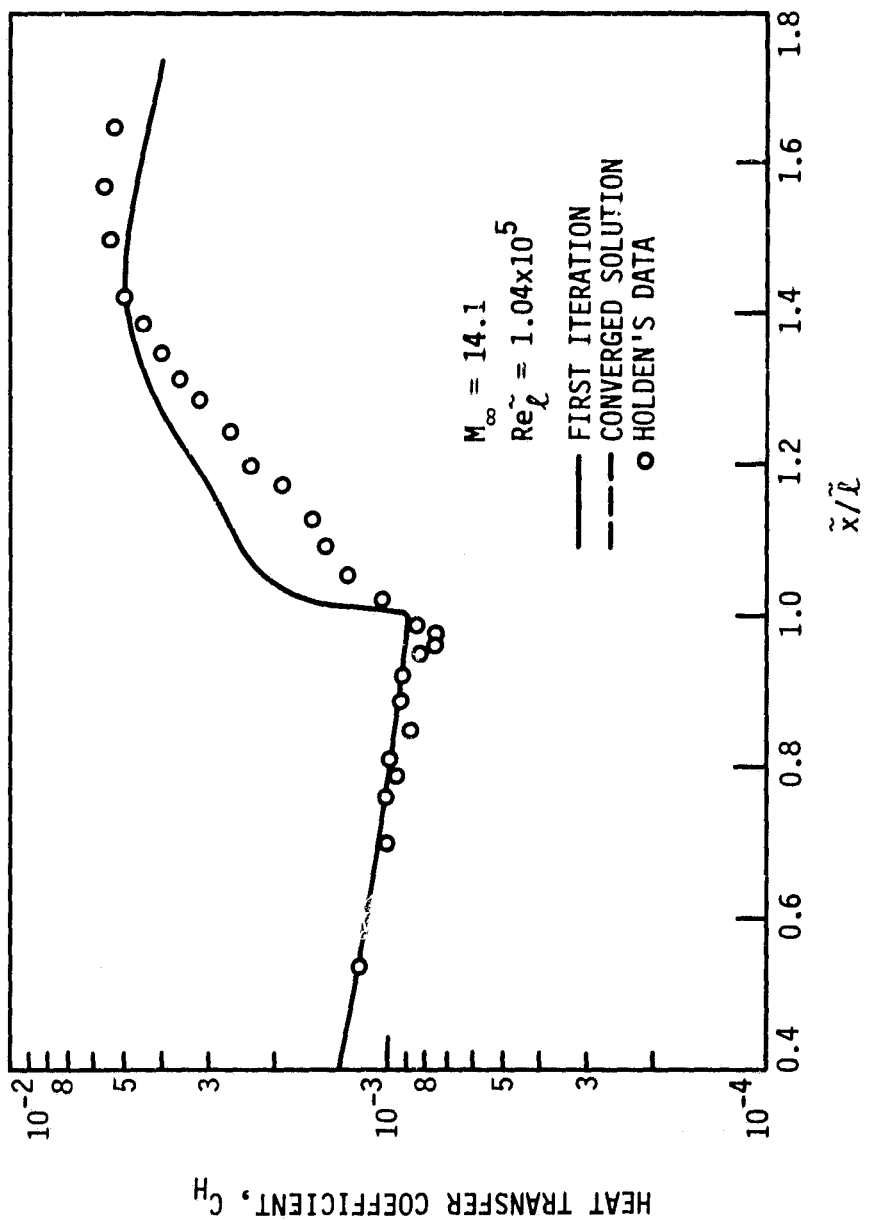

Figure 14. Heat transfer coefficients for iterated implicit MacCormack method

of the first sweep. Even in the vicinity of the corner where the upstream influence should be strongest, the effect of the ramp seems to be felt only one or two grid points upstream.

For purposes of comparison, the global iteration procedure was applied with the Beam-Warming scheme and calculations of the same test case were performed. This method seems to be substantially more sensitive to the forward difference on the pressure than the implicit MacCormack scheme. The consequences are a slower convergence rate and a solution which better displays the effects of upstream influences. Figures 15 and 16 show the pressure and heat transfer coefficients, respectively, after 25 iterations. Regarding the upstream influence, the point of interest in Fig. 15 is the gradual pressure increase in the iterated solution beginning several points ahead of the ramp. The associated feature in Fig. 16 is the slight dip in the heat transfer coefficient near the corner. Both of these characteristics agree well with those of the experimental results though the calculated dip in the heat transfer is displaced slightly toward the ramp.

The pressure, density, and Mach number contours of Figs. 17, 18, and 19, respectively, illustrate the details of the computed flowfield after 25 iterations with the Beam-Warming scheme. These figures, especially the pressure contours, reveal some oscillatory behavior also present to a lesser degree in the wall coefficient plots of Figs. 15 and 16. However, the streamwise oscillations of Figs. 15 and 16 are thought to stem from the severity of the initial conditions, whereas the oscillations of Fig. 17 are probably caused by nonlinear effects associated with the induced

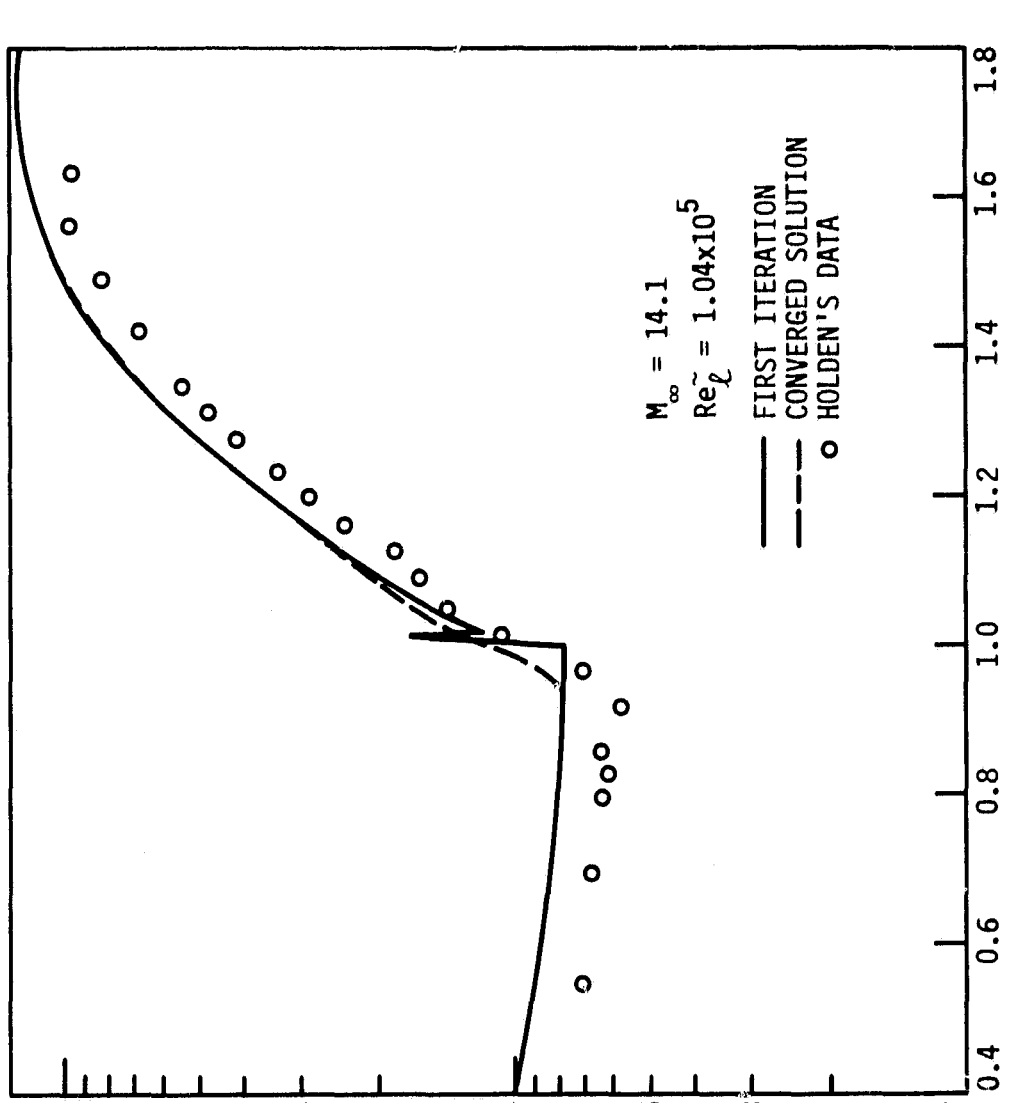

Figure 15. Pressure coefficients for iterated Beam-Warming method

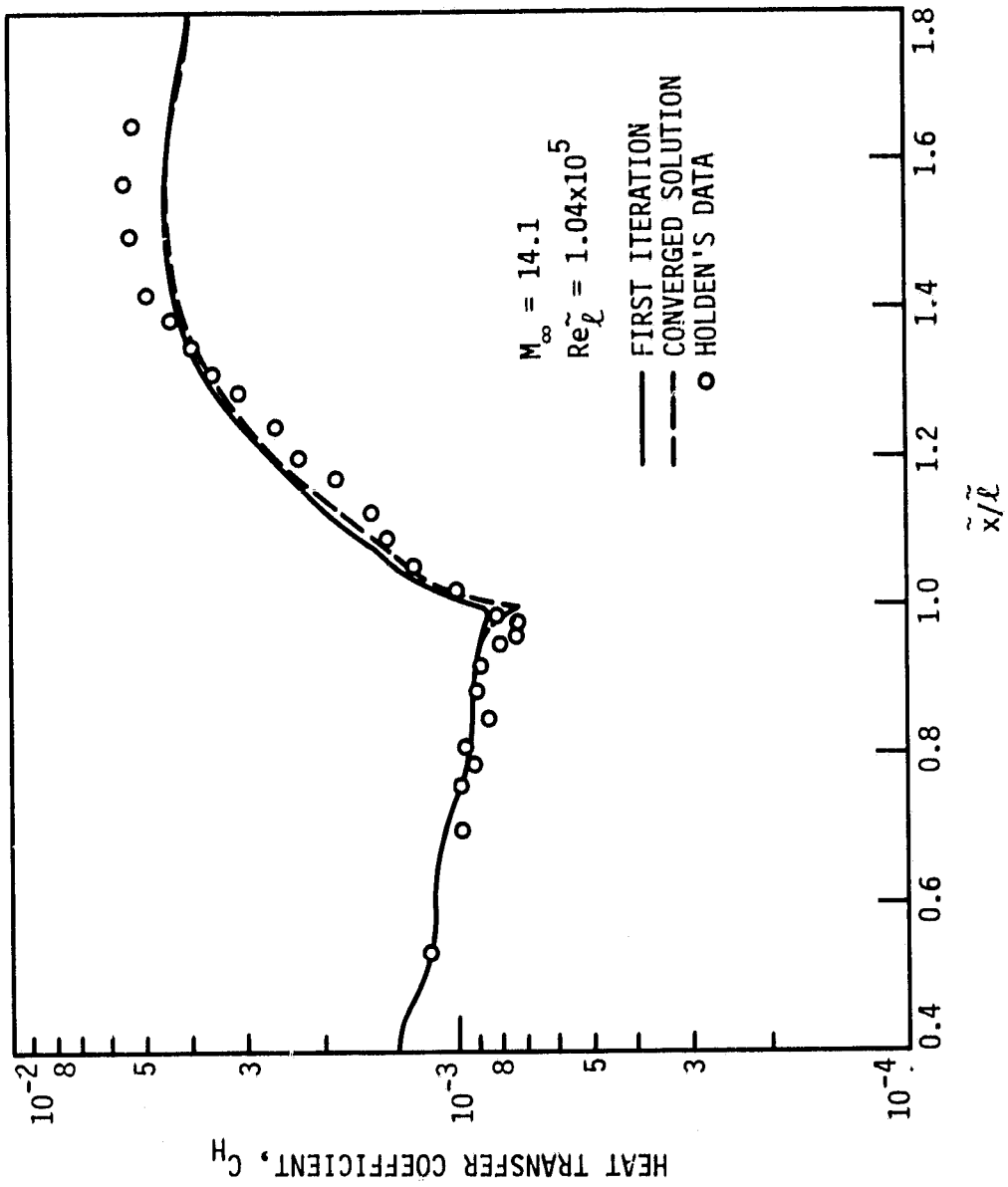

Figure 16. Heat transfer coefficients for iterated Beam-Warming method

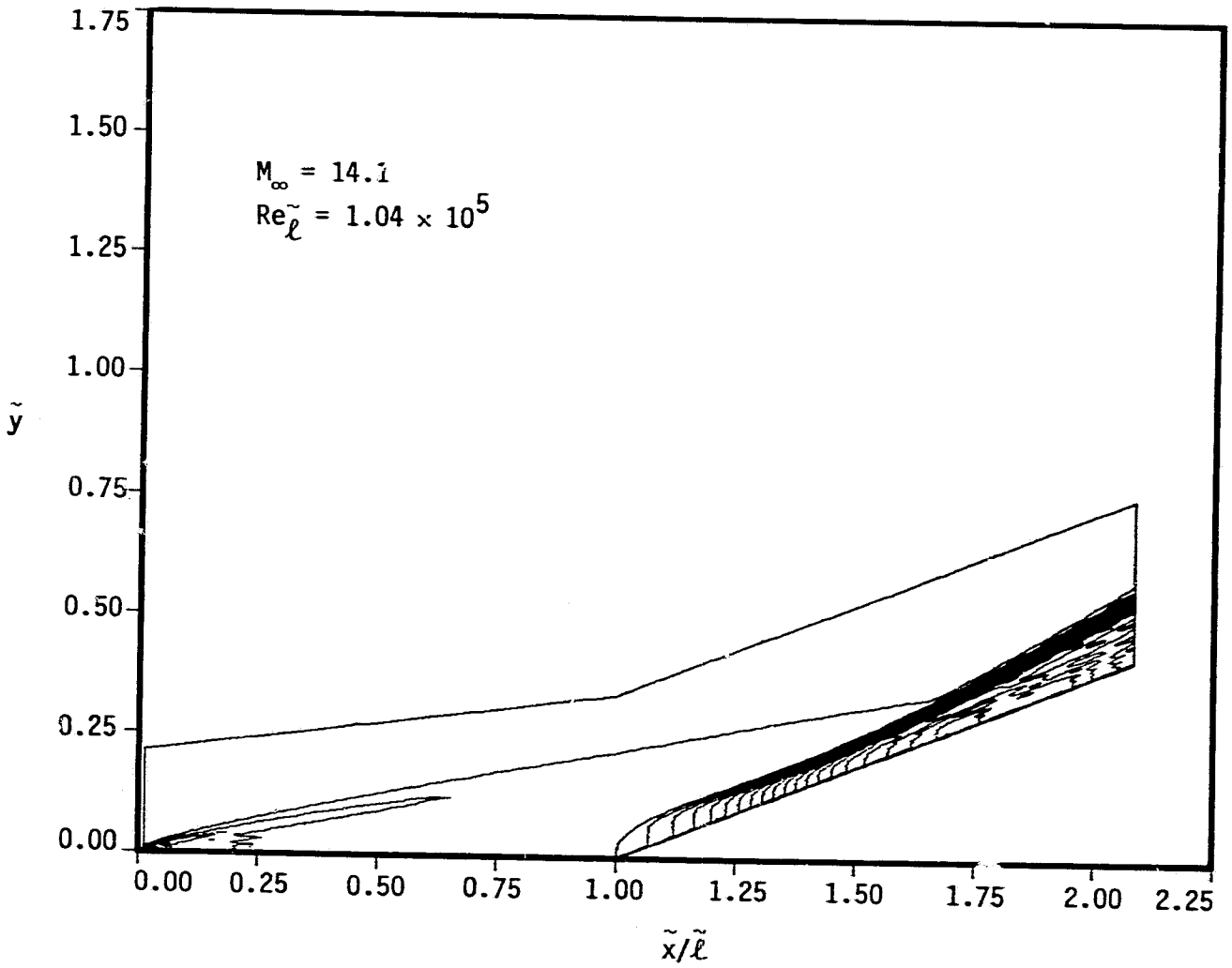

Figure 17. Pressure contours for iterated Beam-Warming method

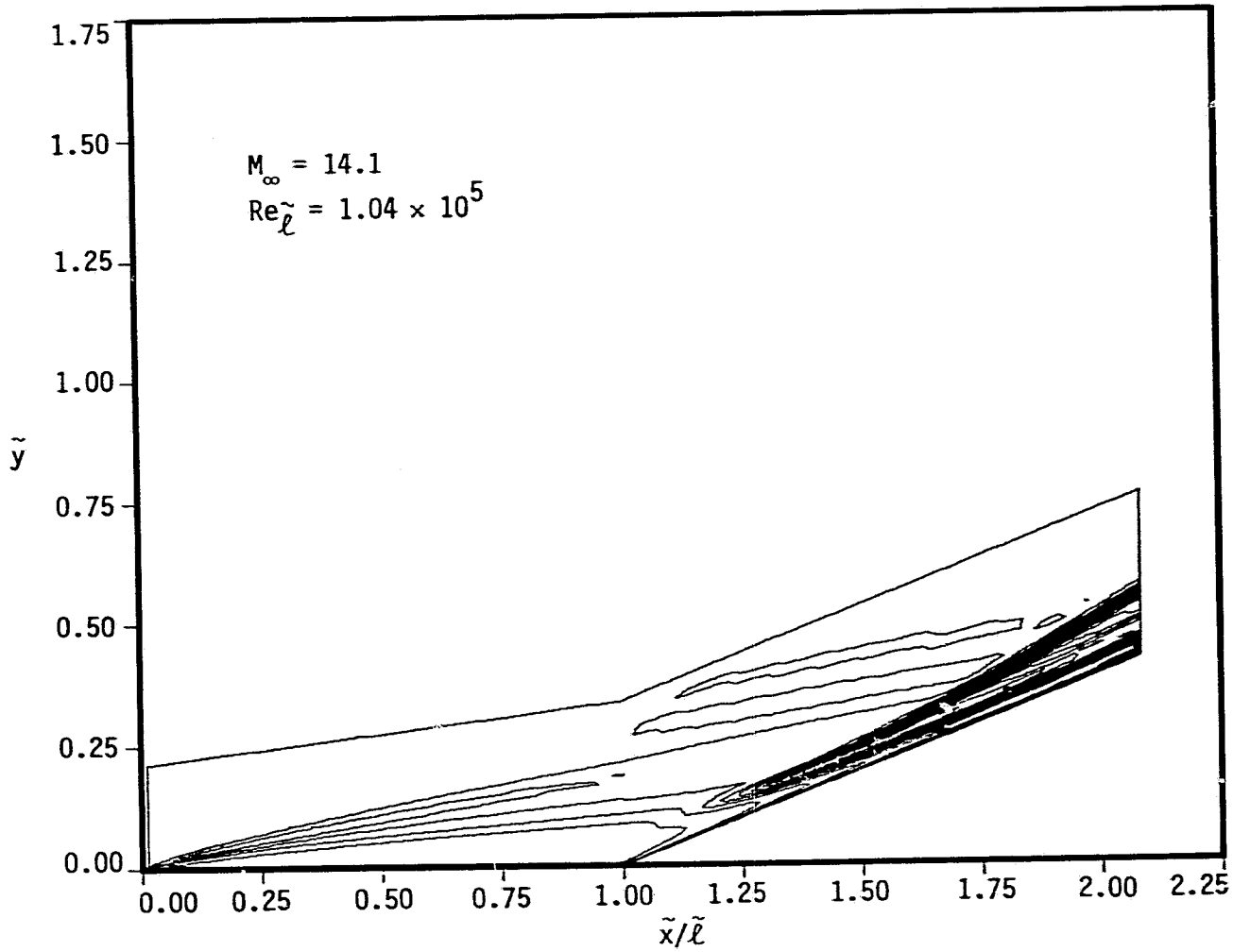

Figure 18. Density contours for iterated Beam-Warming method

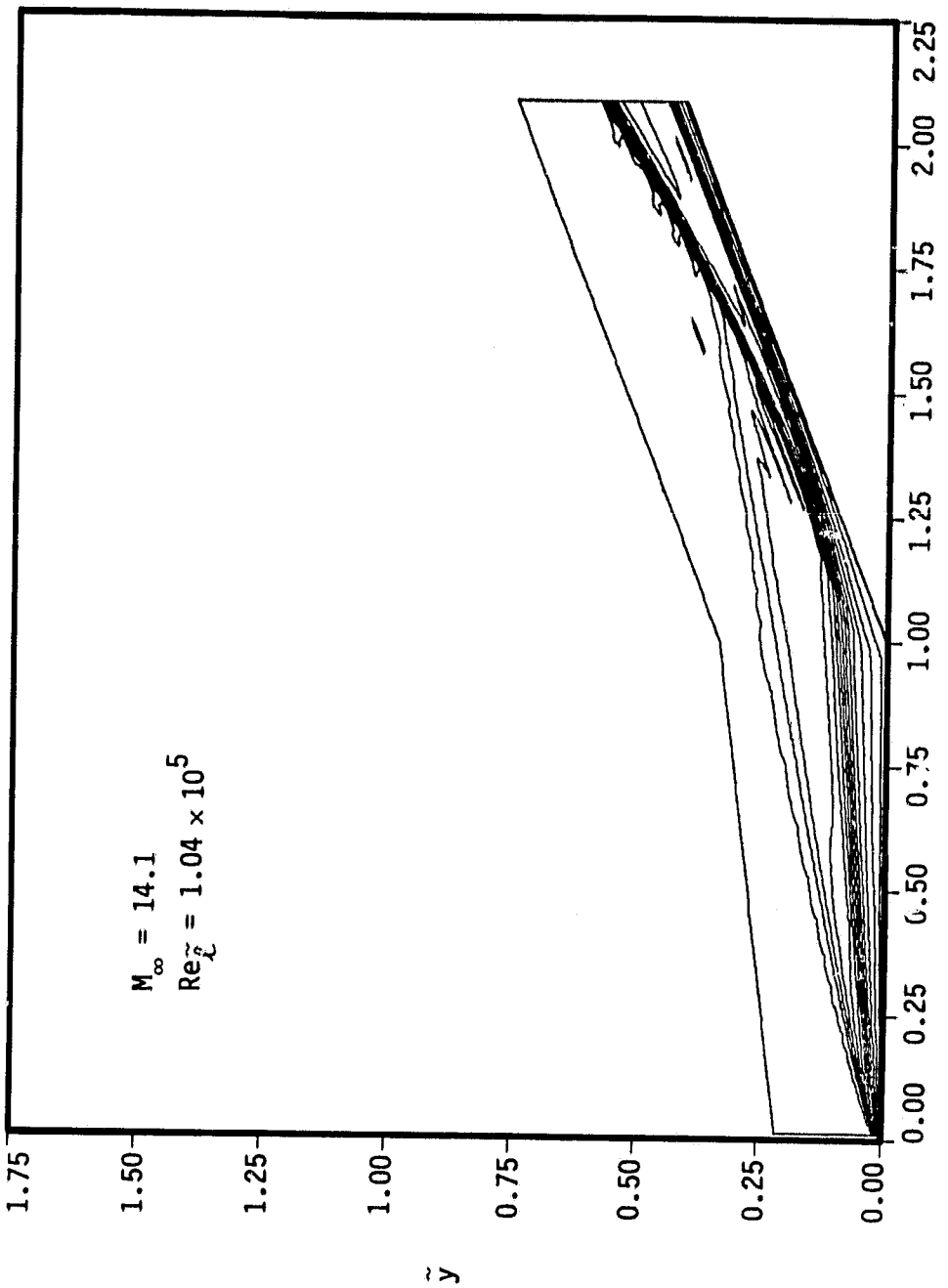

Figure 19. Mach contours for iterated Beam-Warming method

shock. These oscillations illustrate the lack of inherent artificial dissipation in the Beam-Warming scheme.

Velocity profiles computed during the first sweep by the implicit MacCormack scheme and the Beam-Warming scheme are shown in Figs. 20 and 21, respectively. The two figures are similar except for the profile on the ramp nearest the corner. For this profile, the Beam-Warming scheme results seem to show a slightly greater tendency toward separation than the results of the implicit MacCormack code. An expanded view of the velocity profiles in the corner region is given in Figs. 22 and 23. The presence of an inflection point in the Beam-Warming scheme results at $\tilde{x}/\tilde{\ell} = 1.014$ is clear, whereas the results of the implicit MacCormack scheme remain nearly linear throughout this region. Thus, the code using the Beam-Warming scheme, even in the first sweep, seems to exhibit a greater sensitivity to the sharp corner than the implicit MacCormack scheme code.

An iterated Beam-Warming scheme calculation produced the results shown in Fig. 24. Comparison with Fig. 23 shows that the effects of iteration are, again, to propagate the effect of the ramp upstream and, in this case, to slow the flow near the wall. The results of the iterated implicit MacCormack scheme calculation were indistinguishable from those of Fig. 22 and, therefore, have been omitted.

One source of error in the implicit MacCormack code may, as in the flat plate boundary-layer test case, be the solid wall boundary treatment. For that test case, the reflective condition, Eq. (16), proved to be the

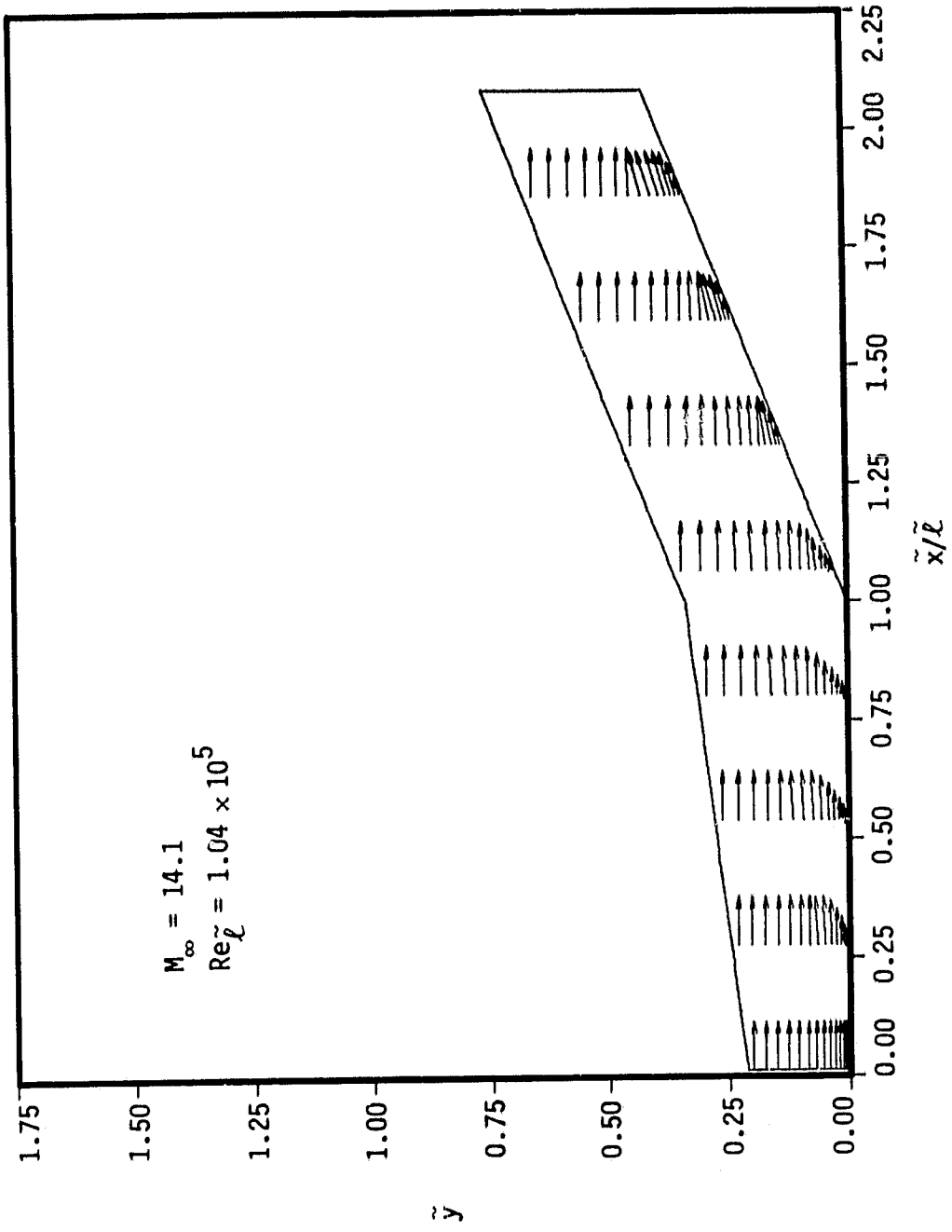

Figure 20. Velocity vectors for single-sweep implicit MacCormack method

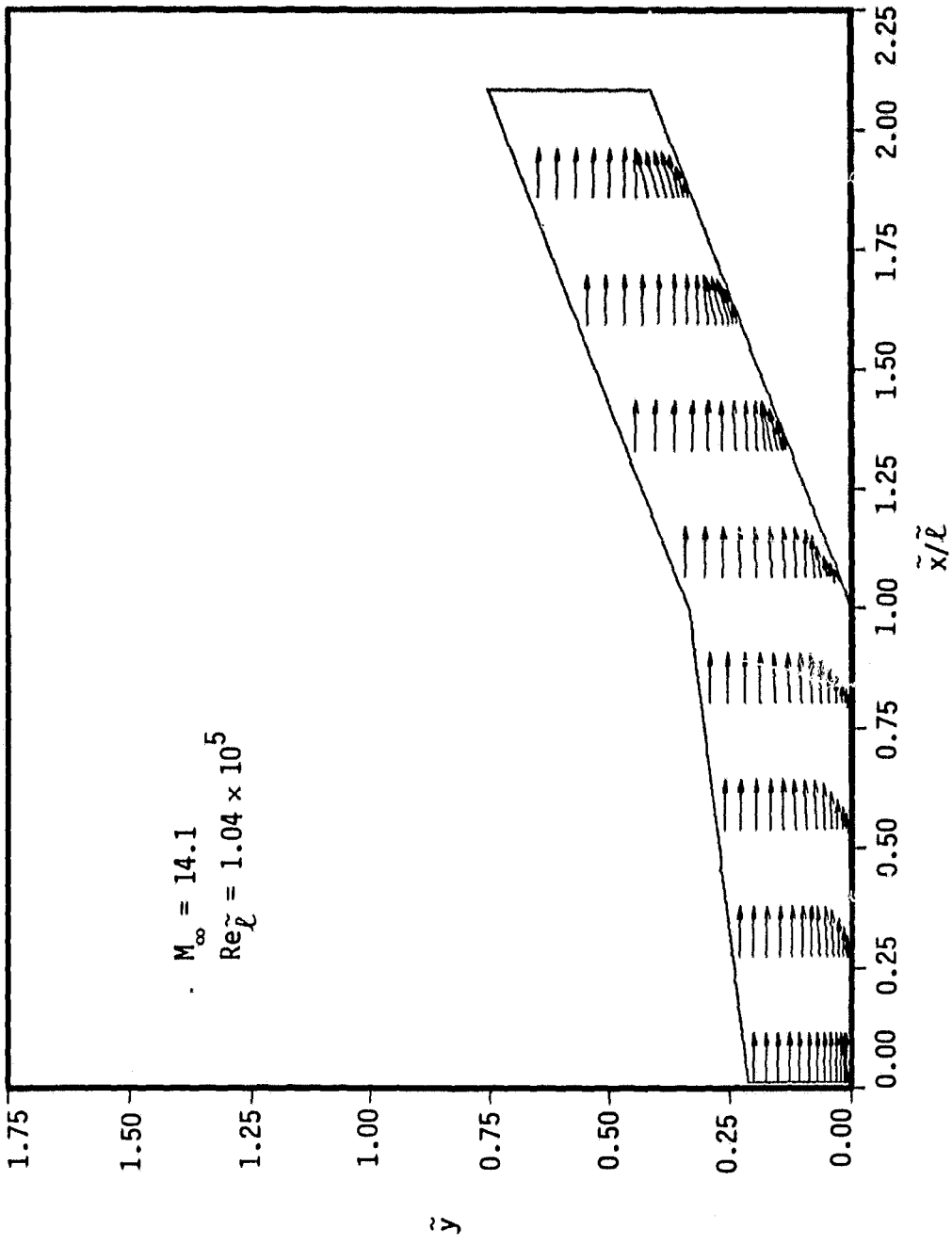

Figure 21. Velocity vectors for single-sweep Beam-Warming method

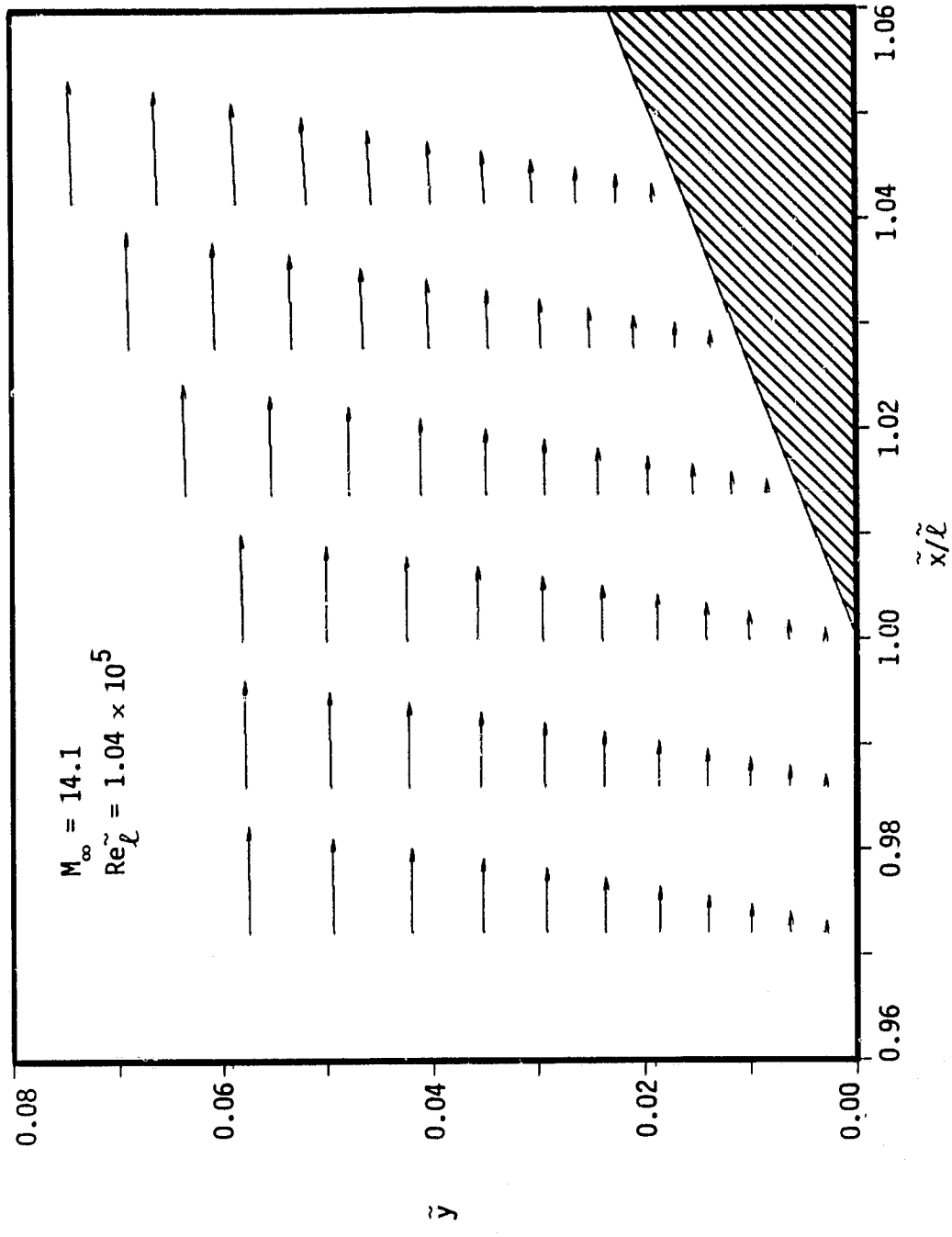

Figure 22. Velocity vectors near corner for single-sweep implicit MacCormack method

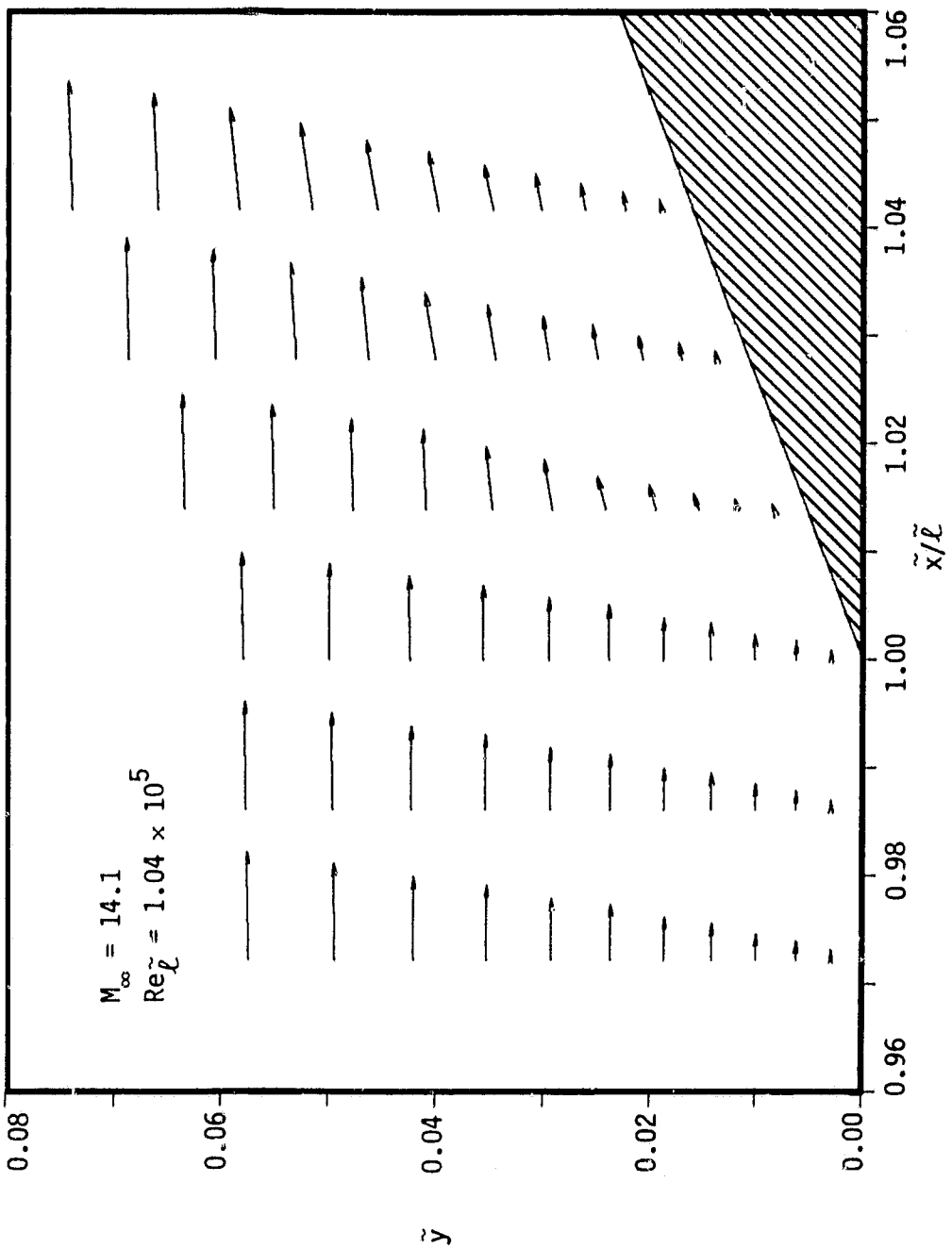

Figure 23. Velocity vectors near corner for single-sweep Beam-Warming method

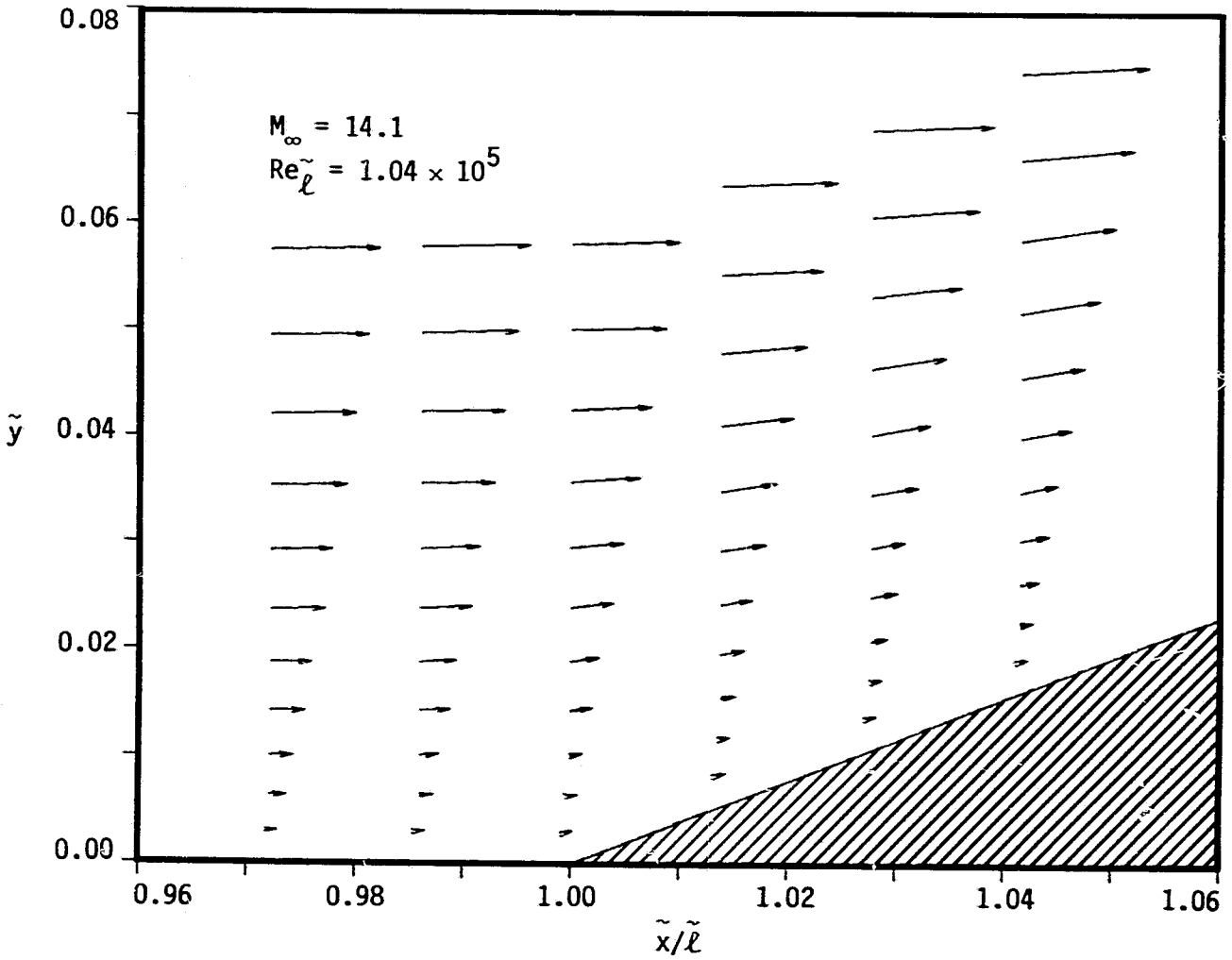

Figure 24. Velocity vectors near corner for iterated Beam-Warming method

more stable and accurate procedure. However, in the present test case, application of Eq. (16) at the wall seems to trigger instabilities near the beginning of the calculations. For this reason, the quantity $\left[\frac{\Delta\xi}{\Delta\eta}\,|B|\,\delta\bar{U}\right]_1^{n+1}$ was set to zero in these computations. Thus, the boundaries are treated only explicitly in the implicit MacCormack code whereas the implicit conditions used in the Beam-Warming code caused no difficulty for this case.

Another possible source of error in the code employing the implicit MacCormack scheme is the effect of subjecting the governing equations in strong-conservation-law form to a finite-difference calculation without properly satisfying the geometric conservation law. This potential problem has been discussed previously in relation to Test Case I where small amounts of grid stretching caused very large errors in that high Courant number case. In the present test case, however, a Courant number of approximately 20 was the largest encountered and only moderate clustering of the grid was employed. In addition, "differencing the freestream" seemed to indicate that these effects should not be significant. Nevertheless, it is possible that the nonlinearity of the solution itself would accentuate these nonphysical effects and cause observable errors in the solution.

V. CONCLUDING REMARKS

The implicit MacCormack scheme has been applied to the two-dimensional, parabolized Navier-Stokes equations for the computation of steady supersonic flowfields. In order to test this method, two flowfields were computed. These included a laminar flat plate boundary-layer case and the hypersonic laminar flow over a 15° compression corner. Present results compare very well with previously published computational results and experimental data. Comparisons were also made with results of the conventional Beam-Warming scheme in terms of accuracy, stability, computer time, computer storage, and ease of implementation.

Also presented in this report are results of a global iteration technique which allows upstream influences to be included in PNS calculations. Very good results were obtained when this procedure was implemented with the Beam-Warming scheme, but the response of the implicit MacCormack scheme to the iteration was somewhat disappointing. In order to improve the accuracy of the implicit MacCormack scheme in general, further investigation is suggested into the development of a reliable implicit boundary condition procedure. In addition, to reduce the strong grid dependence observed here, use of either the chain-rule-conservation-law form of the equations (see Refs. 16 and 20), or a finite-volume approach (see Ref. 13) is strongly recommended.

VI. REFERENCES

1. S. Rudman and S. G. Rubin. "Hypersonic Viscous Flow over Slender Bodies having Sharp Leading Edges." AIAA Journal, 7, No. 10 (October 1968), 1883-1889.

2. S. C. Lubard and W. S. Helliwell. "Calculation of the Flow on a Cone at High Angle of Attack." AIAA Journal, 12, No. 7 (July 1974), 965-974.

3. S. G. Rubin and T. C. Lin. "Numerical Methods for Two- and Three-Dimensional Viscous Flow Problems: Application to Hypersonic Leading Edge Equations." Journal of Computational Physics, 9, (April 1972), 339-364.

4. Y. C. Vigneron, J. V. Rakich, and J. C. Tannehill. "Calculation of Supersonic Viscous Flow over Delta Wings with Sharp Subsonic Leading Edges." AIAA Paper 78-1137, Seattle, Wash., July 1978.

5. L. B. Schiff and J. L. Steger. "Numerical Simulation of Steady Supersonic Viscous Flow." AIAA Paper 79-0130, New Orleans, La., January 1979.

6. I. Lindemuth and J. Killeen. "Alternating Direction Implicit Techniques for Two-Dimensional Magnetohydrodynamics Calculations." Journal of Computational Physics, 13, (October 1973), 181-208.

7. H. McDonald and W. R. Briley. "Three-Dimensional Supersonic Flow of a Viscous or Inviscid Gas." Journal of Computational Physics, 19, (October 1975), 150-178.

8. R. Beam and R. F. Warming. "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations." AIAA Journal, 16, No. 4 (April 1978), 393-401.

9. R. W. MacCormack. "A Numerical Method for Solving the Equations of Compressible Viscous Flow." AIAA Paper 81-0110, St. Louis, Mo., January 1981.

10. R. W. MacCormack. "The Effect of Viscosity on Hypervelocity Impact Cratering." AIAA Paper 69-354, Cincinnati, Oh., April-May 1969.

11. E. von Lavante and W. T. Thompkins. "An Implicit Bidiagonal Numerical Method for Solving the Navier-Stokes Equations." AIAA Journal, 21, No. 6 (June 1983), 828-833.

12. A. Kumar. "Some Observations on a New Numerical Method for Solving the Navier-Stokes Equations." NASA TP-1934, 1981.

13. W. Kordulla and R. W. MacCormack. "Transonic Flow Computation Using an Explicit-Implicit Method." Proceedings of the Eighth International Conference on Numerical Methods, Aachen, Germany, June-July 1982.

14. R. N. Gupta, P. A. Gnoffo, and R. W. MacCormack. "A Viscous Shock-Layer Flowfield Analysis by an Explicit-Implicit Method." AIAA Paper 83-1423, Montreal, Canada, June 1983.

15. M. E. White and J. D. Anderson Jr. "Application of MacCormack's Implicit Method to Quasi-One-Dimensional Nozzle Flows." AIAA Paper 82-0992, St. Louis, Mo., June 1982.

16. J. S. Shang and R. W. MacCormack. "Flow Over a Biconic Configuration with an Afterbody Compression Flap--A Comparative Numerical Study." AIAA Paper 83-1668, Danvers, Mass., July 1983.

17. J. V. Rakich. "Iterative PNS Method for Attached Flows with Upstream Influence." AIAA Paper 83-1955, Danvers, Mass., July 1983.

18. C. M. Hung and R. W. MacCormack. "Numerical Solutions of Supersonic and Hypersonic Laminar Compression Corner Flows." AIAA Journal, 14, No. 4 (April 1976), 475-481.

19. R. H. Pletcher. "On a Calculation Method for Compressible Turbulent Boundary Layer Flows with Heat Transfer." AIAA Paper 71-165, New York, New York, January 1971.

20. R. G. Hindman. "Geometrically Induced Errors and Their Relationship to the Form of the Governing Equations and the Treatment of Generalized Mappings." AIAA Paper 81-1008, Palo Alto, Calif., June 1981.

21. M. S. Holden and J. R. Moselle. "Theoretical and Experimental Studies of the Shock Wave-Boundary Layer Interaction on Compression Surfaces in Hypersonic Flow." CALSPAN Rept. AF-2410-A-1, Buffalo, New York, October 1969.

22. M. H. Bertram and T. A. Blackstock. "Some Simple Solutions to the Problem of Predicting Boundary-Layer Self-Induced Pressure." NASA TN D-798, April 1961.

23. B. Edney. "Anomalous Heat-Transfer and Pressure Distributions on Blunt Bodies at Hypersonic Speeds in the Presence of an Impinging Shock." FFA Rept. 116, Aeronautical Research Inst. of Sweden, Stockholm, Sweden, February 1968.

VII. ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to Dr. J. C. Tannehill for his guidance and support throughout this work, and to Denny S. Chaussee and the other people of NASA Ames' Applied Computational Aerodynamics Branch. I am also grateful to Mrs. Shirley Riney for her skill and speed in putting together this manuscript.

This work was supported by NASA Ames Research Center under Grant NCA2-OR340-301 and the Computational Fluid Dynamics Institute, Iowa State University, Ames, Iowa.

VIII. APPENDIX: $\partial \bar{E}^*/\partial \bar{U}$, $\partial \bar{F}_1/\partial \bar{U}$, S_y^{-1}, and S_y

The Jacobian $\partial \bar{E}^*/\partial \bar{U} = A$ is given by

$$\frac{\partial \bar{E}^*}{\partial \bar{U}} = \frac{\partial}{\partial \bar{U}}\left(\frac{E^*}{J}\right) = \frac{\partial E^*}{\partial U}$$

and the Jacobian $\partial E^*/\partial U$ is given below

$$\frac{\partial E^*}{\partial U} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -u^2 + \frac{1}{2}\omega(\gamma-1)(u^2+v^2) & [2-(\gamma-1)\omega]u & -\omega(\gamma-1)v & \omega(\gamma-1) \\ -uv & v & u & 0 \\ \left[-\frac{\gamma e_t}{\rho} + (\gamma-1)(u^2+v^2)\right]u & \frac{\gamma e_t}{\rho} - \frac{1}{2}(\gamma-1)(3u^2+v^2) & -(\gamma-1)uv & \gamma u \end{bmatrix}$$

The Jacobian $\partial \bar{F}_i/\partial \bar{U}$, using the fact that $\bar{U} = U/J$, becomes

$$\frac{\partial \bar{F}_i}{\partial \bar{U}} = \eta_x \frac{\partial E}{\partial U} + \eta_y \frac{\partial F}{\partial U}$$

where

$$\frac{\partial E}{\partial U} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \phi^2 - u^2 & (3-\gamma)u & -(\gamma-1)v & \gamma-1 \\ -uv & v & u & 0 \\ \left[-\frac{\gamma e_t}{\rho} + 2\phi^2\right]u & \frac{\gamma e_t}{\rho} - \phi^2 - (\gamma-1)u^2 & -(\gamma-1)uv & \gamma u \end{bmatrix}$$

and

$$\frac{\partial F}{\partial U} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -uv & v & u & 0 \\ \phi^2 - v^2 & -(\gamma-1)u & (3-\gamma)v & \gamma-1 \\ \left[-\frac{\gamma e_t}{\rho} + 2\phi^2\right]v & -(\gamma-1)uv & \frac{\gamma e_t}{\rho} - \phi^2 - (\gamma-1)v^2 & \gamma v \end{bmatrix}$$

where $\phi^2 = \frac{1}{2}(\gamma-1)(u^2 + v^2)$. Defining $V = \eta_x u + \eta_y v$, $\partial \bar{F}_i / \partial \bar{U}$ can be written as

$$\frac{\partial \bar{F}_i}{\partial \bar{U}} = \begin{bmatrix} 0 & \eta_x & \eta_y & 0 \\ \eta_x\phi^2 - uV & V + \eta_x(2-\gamma)u & \eta_y u - \eta_x(\gamma-1)v & (\gamma-1)\eta_x \\ \eta_y\phi^2 - vV & \eta_x v - \eta_y(\gamma-1)u & V + (2-\gamma)v\eta_y & (\gamma-1)\eta_y \\ \left[-\frac{\gamma e_t}{\rho} + 2\phi^2\right]V & \eta_x\left(\frac{\gamma e_t}{\rho} - \phi^2\right) - (\gamma-1)uV & \eta_y\left(\frac{\gamma e_t}{\rho} - \phi^2\right) - (\gamma-1)vV & \gamma V \end{bmatrix}$$

which has eigenvalues

$$\lambda_{1,2} = V$$

$$\lambda_3 = V + c\sqrt{\eta_x^2 + \eta_y^2}$$

$$\lambda_4 = V - c\sqrt{\eta_x^2 + \eta_y^2}$$

where c is the speed of sound.

Defining

$$\alpha = \rho/(\sqrt{2}\,c)\,, \quad \beta = 1/(\sqrt{2}\,\rho c)$$

$$\hat{\eta}_x = \eta_x/\sqrt{\eta_x^2 + \eta_y^2}\,, \quad \hat{\eta}_y = \eta_y/\sqrt{\eta_x^2 + \eta_y^2}$$

and

$$\hat{V} = \hat{\eta}_x u + \hat{\eta}_y v$$

the matrix of left eigenvectors of $\partial \bar{F}_i / \partial \bar{U}$ is

$$S_y^{-1} = \begin{bmatrix} 1 & 0 & \alpha & \alpha \\ u & \hat{\eta}_y \rho & \alpha(u + \hat{\eta}_x c) & \alpha(u - \hat{\eta}_x c) \\ v & -\hat{\eta}_x \rho & \alpha(v + \hat{\eta}_y c) & \alpha(v - \hat{\eta}_y c) \\ \frac{\phi^2}{\gamma-1} & \rho(\hat{\eta}_y u - \hat{\eta}_x v) & \alpha\left(\frac{\phi^2 + c^2}{\gamma-1} + c\hat{V}\right) & \alpha\left(\frac{\phi^2 + c^2}{\gamma-1} - c\hat{V}\right) \end{bmatrix}$$

and its inverse is

$$S_y = \begin{bmatrix} 1 - \phi^2/c^2 & (\gamma-1)u/c^2 & (\gamma-1)v/c^2 & -(\gamma-1)/c^2 \\ -(\hat{\eta}_y u - \hat{\eta}_x v)/\rho & \hat{\eta}_y/\rho & -\hat{\eta}_x/\rho & 0 \\ \beta(\phi^2 - c\hat{V}) & \beta[\hat{\eta}_x c - (\gamma-1)u] & \beta[\hat{\eta}_y c - (\gamma-1)v] & \beta(\gamma-1) \\ \beta(\phi^2 + c\hat{V}) & -\beta[\hat{\eta}_x c + (\gamma-1)u] & -\beta[\hat{\eta}_y c + (\gamma-1)v] & \beta(\gamma-1) \end{bmatrix}$$