Patched Grid Solutions of The Two Dimensional

Euler and Thin-Layer Navier-Stokes Equations

by

George Frederick Switzer

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Aerospace & Ocean Engineering

APPROVED:

Dr. Robert W. Walters, Chairman

Dr. Joseph A. Schetz

Dr. Bernard Grossman

May, 1987

Blacksburg, Virginia

Patched Grid Solutions of The Two Dimensional

Euler and Thin-Layer Navier-Stokes Equations

by

George Frederick Switzer

Dr. Robert W. Walters, Chairman

Aerospace & Ocean Engineering

(ABSTRACT)

The development of the patched grid solution methodology for both the Euler and the Navier-Stokes equations in two dimensions is presented. The governing equations are written in the integral form and the basic numerical algorithm is finite volume. The method is capable of first through third order accuracy in space. The flux vectors associated with the Euler equations are split into two sub-vectors (based on the signs of the characteristic speeds) and discretized separately. The viscous and heat flux contributions are treated with central differences. Patched grid results are demonstrated on shock reflection, subsonic boundary layer, and shock-boundary layer interaction flow problems. The results are compared with non-patched or single zone grids. The patched grid approach shows an improvement in resolution while minimizing storage and computer time.

Table of Contents

List of Illustrations

Introduction

The field of aerodynamic simulation is composed of different areas including theoretical, experimental and numerical investigation or computational fluid dynamics (CFD). The ability of computers to perform many computations a second and the large amount of storage available has only recently reached levels where full scale CFD simulation may be accomplished within a reasonable amount of computing time. All of the different costs of each method must be weighed to determine which is most practical. One advantage of numerical simulation is the ability to simulate environments that are difficult to be experimentally duplicated. One such example is the National Aerospace Plane during reentry. In order to achieve the stand alone capability, CFD codes must be developed that are reliable and fast. In the early stages of code development the comparison of numerical results with test results (either experimental or analytical) is necessary to demonstrate validity.

This thesis will obtain the steady state solution of problems by breaking the computational domain into smaller sub-regions or zones. Furthermore, the adjacent zones meet along a common line and contain no overlaps. These grids will be referred to as patched grids. There are three advantages to using sub-regions: storage, computing time, and accuracy. The first two advantages are related and obvious while the third is not so obvious.

One of the first tasks of numerical implementation is to discretize the physical domain. This involves generating a grid. The requirements of this grid are dictated by the problem of interest. Some domains or geometries are so complex that it is not even possible to generate a suitable grid, and for others the resulting grid will contain regions where there are unnecessary grid points. Some examples of problems with complex geometry are the Space Shuttle and the National Aerospace Plane. One advantage of the patched grid scheme is the ability to obtain an adequately resolved mesh spacing or grid everywhere in the domain. This is achieved by selectively refining the grid in local regions in the domain. These refinements are placed where better resolution of the flow is desired as compared to the overall coarser grid. Another advantage of the patching of grids is the reduction in storage and computer (CPU) time. This results from each cell or grid point corresponding to a finite volume over which the governing equations are to be solved. Therefore, the reduction in points is directly related to storage and CPU time. The patched grids further reduce storage requirements by facilitating block-processing computations. The advantages of block-processing is that only the information for a single zone must reside in computer memory. The storage limitation can now apply to individual zones rather than the complete region. This will allow the computational domain to contain the necessary refinement by breaking the region into individual zones that can be handled by the core memory. For some computers, block-processing coupled with patched grids will enable problems to be solved that currently cannot be solved due to memory limitations. The final advantage is in accuracy. Grid skewedness can be shown to introduce errors.[1] The patched grid approach allows the independent generation of zones within the computational domain thereby allowing greater control of the grid quality. The resulting patched grids are simpler to construct, contain fewer grid points for a given level of accuracy and allow greater control of the overall grid features.

The result of the patched grid approach is an efficient and versatile solution of the governing equations in general regions. Investigations in patched grid solutions in both two and three

dimensions for the Euler equations have been done by Rai and Hessenius.[2-5] The extension here will be to incorporate the viscous effects by using the thin- layer Navier-Stokes equations.

This investigation is aimed at validating patched grid solutions of the thin-layer Navier-Stokes equations. The process will start with the Euler equations and then add the viscous terms. The viscous patched grid solutions will be compared to both analytical and experimental results. It is hoped that the chosen test cases demonstrate the usefulness of patched grid solutions. The ultimate objective is to obtain patched grid solutions for the thin-layer Navier-Stokes equations on arbitrary grids in three dimensions.

Governing Equations

Discretization

The non-dimensional integral form of the conservation laws are

$$\frac{\partial}{\partial t}\int_V Q dV + \oint_S (\mathbf{F}_i - \mathbf{F}_v) \bullet \mathbf{n} dS = 0 \qquad [2.1.1]$$

(refer to appendix A for the nondimensionalization of the governing equations.)

where V is the volume, S is the surface of that volume, and **n** is the outward facing unit normal. F_i is the flux vector associated with the Euler equations and F_v contains the viscous and heat flux contributions. The two-dimensional equations are obtained by setting all gradients and velocity components in the third dimension to zero. The resulting volume and surface integrals are then reduced to area and line integrals, respectively.

This integral representation is valid over any region and in the presence of discontinuities in the flow, such as shocks. Also any region in space will satisfy the integral form, but for this

study the region will be quadrilaterals, which I denote as cells. This simplifies the evaluation of the surface integral by limiting the unit normal to four distinct values for each cell. The next task is to evaluate the integrals. The fluxes are interpreted as cell face averages so the surface integral can be written as

$$\oint_S (\mathbf{F}_i - \mathbf{F}_v) \bullet \mathbf{n} dS = \sum_{l=1}^{4} [(\mathbf{F}_i + \mathbf{F}_v) \bullet \mathbf{n} \Delta S]_l \qquad [2.1.2]$$

and the volume integral can be viewed as a cell average value given by

$$Q_{j,k}^{(t)} = \frac{1}{V_{j,k}} \int_V Q(x,y,t) dV \qquad [2.1.3]$$

Each cell has a cell center where the cell average dependent variables are stored. Upwind differencing is incorporated in the Euler or inviscid contribution by the splitting of the fluxes into positive and negative components based on the signs of the characteristic speeds. The positive contribution is associated with waves travelling from left to right and the negative contribution is associated with right to left travelling waves. This is flux vector splitting (FVS). The reason for splitting the flux into components is to better model the physical (characteristic) information of the flow. FVS separates each component of the flux by interpolating in the proper direction to obtain the flux at a cell face. The result is a difference scheme that has been constructed based on the physics of the flow. Van Leer's method of FVS will be used[6] because it has smoothly varying flux contributions at sonic and stagnation points[7]. For the inviscid contribution in the x- direction the flux splitting is

$$F_i = F^+_i(Q_L) + F_i^-(Q_R) \qquad [2.1.4]$$

where Q_L and Q_R corresponds to upwind interpolation of the dependent variables to the cell faces and for $|M_x| < 1$, F_i is given by

$$F^{\pm}(Q_{L,R}) = \begin{bmatrix} f^{\pm}_{mass} \\ f^{\pm}_{mass}\dfrac{[(\gamma-1)u \pm 2c]}{\gamma} \\ f^{\pm}_{mass}v \\ f^{\pm}_{mass}\left[\dfrac{[(\gamma-1)u \pm 2c]^2}{[2(\gamma^2-1)]} + \dfrac{v^2}{2}\right] \end{bmatrix} \qquad [2.1.5]$$

where

$$f^{\pm}_{mass} = \pm\rho c\left[\frac{1}{2}(M_x \pm 1)\right]^2$$

For $M_x \geq 1$, $F_i^+ = F_i$ and $F_i^- = 0$; for $M_x \leq -1$, $F_i^+ = 0$ and $F_i^- = F_i$. M_x is the local Mach number based on u and the local speed of sound c. The calculation of the fluxes at the cell faces is done by interpolating the cell average dependent variables to the cell face and then calculating the flux which is referred to as MUSCL differencing[8] (Monotone Upwind Scalar Conservation Law) after the first code incorporating such a treatment. The method of interpolating the dependent variables is given by the following interpolation equations written in operator form

$$\begin{aligned} (Q_L)_{j+\frac{1}{2},k} &= Q_{j,k} + \frac{S_{j,k}}{4}[(1 - S_{j,k}\kappa_x)\nabla + (1 + S_{j,k}\kappa_x)\Delta]Q_{j,k} \\ (Q_R)_{j+\frac{1}{2},k} &= Q_{j+1,k} - \frac{S_{j+1,k}}{4}[(1 - S_{j+1,k}\kappa_x)\Delta + (1 + S_{j+1,k}\kappa_x)\nabla]Q_{j+1,k} \end{aligned} \qquad [2.1.6]$$

Where $j \pm \frac{1}{2}$ are the location of the cell faces and $S_{j,k}$ is a limiter that will reduce the interpolation to first- order in the presence of large gradients. The type of limiter used is the Van Albeda limiter[9]. This is a continuously operating limiter unlike other limiters which only act only in the presence of large gradients in the flow. The limiter is calculated by the following equation, also in operator form

$$S_{j,k} = \frac{2(\Delta\nabla)\,Q_{j,k} + \varepsilon}{(\Delta^2 + \nabla^2)\,Q_{j,k} + \varepsilon} \qquad [2.1.7]$$

where ε is a small number ($\sim 10^{-6}$) used to prevent division by zero in null-gradient regions of the flowfield.

The operators are given by

$$\begin{aligned} \Delta Q_{j,k} &= Q_{j+1,k} - Q_{j,k} \\ \nabla Q_{j,k} &= Q_{j,k} - Q_{j-1,k} \end{aligned} \qquad [2.1.8]$$

The treatment of the flux and interpolation in the y-direction is similar.

The interpolation equations are a family based on the choice of $S_{j,k}$ and κ. For unlimited solutions $S_{j,k}$ is either 1 or 0, and for limited solutions equation [2.1.7] is used. For $S_{j,k} = 0$ the equations are first-order accurate in space and for $S_{j,k} = 1$, κ determines the order of spatial accuracy. In order to be consistent with FVS the dependent variables should be interpolated in an upwind fashion. Two common choices are $\kappa = -1$ and 1/3 corresponding to fully upwind second- order or third-order upwind biased interpolation, respectively. These will be the only values of κ used. The resulting equations with this splitting are

$$V_{j,k}\frac{\partial}{\partial t}Q^n_{j,k} + \left[\oint_S (\mathbf{F}^+_i + \mathbf{F}^-_i - \mathbf{F}_v)\bullet \mathbf{n}dS\right]_{j,k} = 0 \qquad [2.1.9]$$

It is to be noted that this discretization follows the work of Thomas and Walters[10].

Thin layer approximation

For a large class of problems, namely, for high Reynolds number flows, the flow field is inviscid dominated and the viscous effects are only important in the thin boundary layer near the surface. In addition, in a body orientated coordinate system, the shear and heat flux terms in the direction parallel to the body surface can be neglected relative to the terms in the direction normal to the body surface. Thus, the thin-layer approximation simplifies the computational problem by neglecting the viscous terms in directions other than normal to a surface. The principle advantage of the thin-layer approximation is that it simplifies the computational requirements which results in reduced computer time and storage requirements. It should be noted that the thin-layer assumption is not a necessary simplification for the patched grid formulation. See appendix C for the treatment of the viscous terms.

Integration and linearization

There are many different schemes for integrating the governing equations in time. With the objective of steady state results the order of accuracy in time is not important. The Euler implicit integration scheme is used because it has the advantage of being unconditionally stable and, as a result, has no theoretical limitation on the size of the time step. The Euler implicit scheme written in delta form is

$$\Delta Q = \Delta t\left(\frac{\partial Q}{\partial t}\right)^{n+1} + O(\Delta t^2) \qquad [2.3.1]$$

where

$$\Delta Q = Q^{n+1} - Q^n \qquad \Delta t = t^{n+1} - t^n$$

Substituting into equation [2.1.9] yeilds

$$V_{j,k}\frac{\Delta Q}{\Delta t} + \left[\oint_S (\mathbf{F}_i - \mathbf{F}_v)^{n+1} \bullet \mathbf{n} dS\right]_{j,k} = 0 \qquad [2.3.2]$$

Now the equation is at the n+1 time step. The problem is that the fluxes are non-linear functions of Q at the n+1 time step resulting in a set of coupled non-linear equations. The fluxes are then linearized by use of a Taylor series expansion about the nth time step

$$\mathbf{F}^{n+1} = \mathbf{F}^n + \left(\frac{\partial \mathbf{F}}{\partial Q}\right)^n \Delta Q + O(\Delta t^2) \qquad [2.3.3]$$

The substitution of the linearization into the governing equations gives

$$V_{j,k}\frac{\Delta Q}{\Delta t} + \oint_S \left[\left(\frac{\partial \mathbf{F}}{\partial Q}\right)^n \Delta Q\right] \bullet \mathbf{n} dS = -\oint_S (\mathbf{F})^n \bullet \mathbf{n} dS \qquad [2.3.4]$$

where $\mathbf{F} = \mathbf{F}_i - \mathbf{F}_v$.

Now the equation is linear in the unknown variable ΔQ. All the flux terms and the Jacobians associated with them are at the current time step. The remaining task is to set up the equations for each cell in the computational domain and solve the resulting system of equations

Algorithm Development

The governing equations are applied to each cell. A single grid will have as many equations as cells. Thus, application of equation [2.3.4] will yield a system of equations in matrix form

$$[M]\{\Delta Q\} = -\{R\} \qquad [3.1]$$

where each row of the above equation corresponds to the governing equations at a cell. Each element of [M] is a block 4 by 4 matrix for the two dimensional equations, and $\{\Delta Q\}$ and $\{R\}$ are vectors. The matrix [M] is a large sparse banded matrix and the particular method of ordering the equations will dictate its structure. The right hand side (RHS) is the discrete representation of the surface integral for a cell at the current time.

The problem is to determine the values of $\{\Delta Q\}$ over the grid. The conservative variables can then be updated by

$$Q^{n+1} = Q^n + \Delta Q \qquad [3.2]$$

and the iteration process continued. One approach would be to solve the linear system [3.1] with a direct method, but for direct inversion of a banded matrix, the operation count is

$$\text{WORK} \cong n\,[\beta]^2 \qquad [3.3]$$

where WORK is the number of floating point operations (FLOPs), n is the number of unknowns, and β is the bandwidth of the system[11] .

For large sparse systems, direct inversion is not as efficient as iteration schemes[12]. Relaxation methods approximate the coefficient matrix [M] so that inversion of the relaxed system requires much less work. First, however, the structure of the original matrix must be looked at and a relaxation strategy adopted from there. The cell molecule in addition to the ordering of equations will further dictate the structure of this matrix. The upwind interpolation equations [2.1.6] result in a cell molecule of nine cells as shown in Figure 1. The corresponding structure of the coefficient matrix with column-wise ordering of $Q_{j,k}$ is shown in Figure 2. Each row of the linear system will have the following form

$$\begin{aligned} h\,\Delta Q_{j-2,k} + f\,\Delta Q_{j-1,k} + d\,\Delta Q_{j,k-2} + a\,\Delta Q_{j,k-1} + b\,\Delta Q_{j,k} + c\,\Delta Q_{j,k+1} \\ + e\,\Delta Q_{j,k+2} + g\,\Delta Q_{j+1,k} + i\,\Delta Q_{j+2,k} = -R_{j,k} \end{aligned} \qquad [3.4]$$

where the coefficients (a,b,c,d,e,f,g,h,i) for the two-dimensional Navier-Stokes equations are given in appendix D.

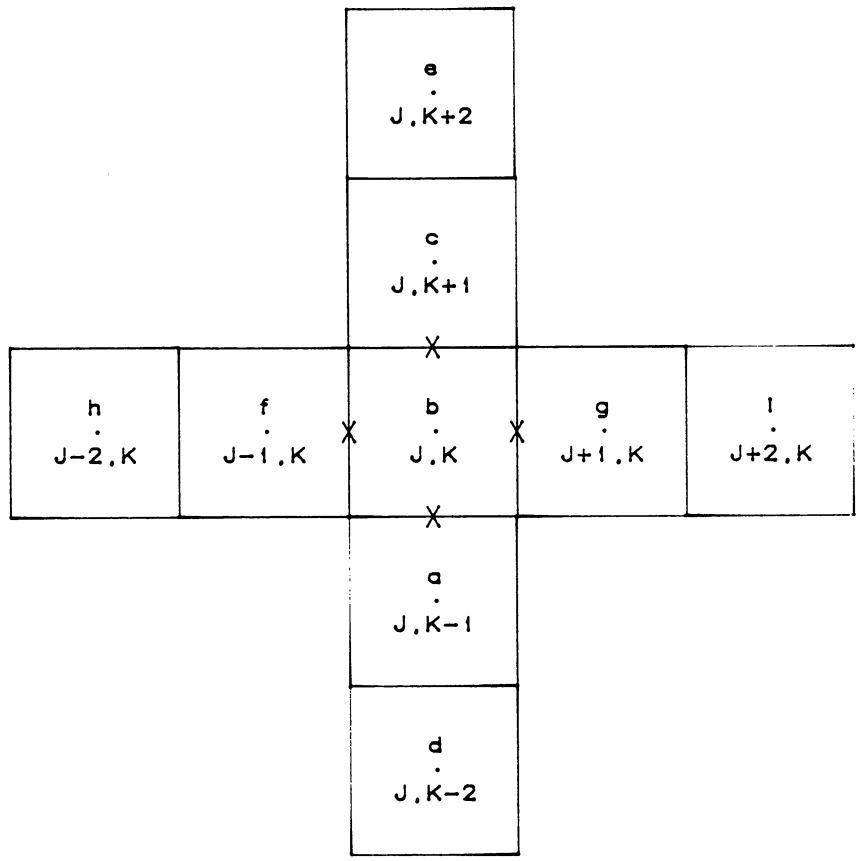


Figure 1. Cell molecule for upwind interpolation

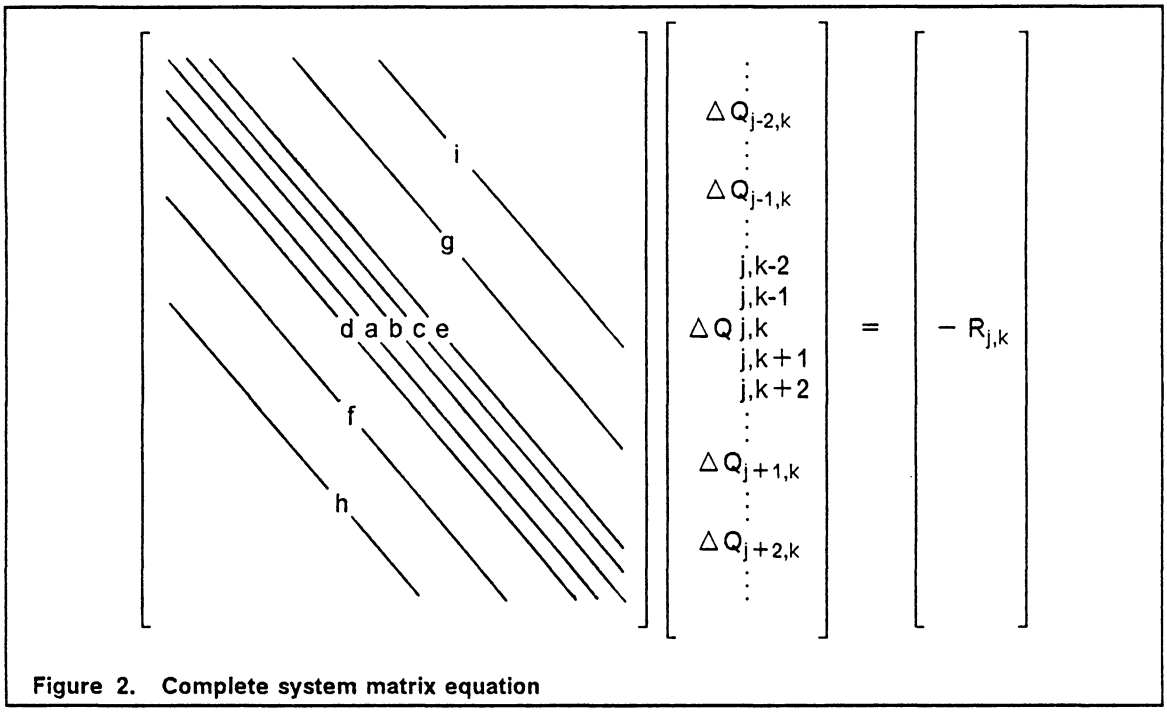


Figure 2. Complete system matrix equation

The grid in the computational space is shown in Figure 3. For the column-wise ordering of the equations, the distance in the vector from j,k to j+1,k is KMAX where KMAX is the number of cells per column. The algorithm used here to simplify the coefficient matrix is vertical line Gauss-Seidel (VLGS). Referring to the linear system (Figure 2) VLGS can be illustrated by breaking the coefficient matrix into three components and rewriting the system as

$$[L + B + U]\,\Delta Q = -[R] \qquad [3.5]$$

where L is a matrix with non-zero entries only on the lower two diagonals, B is a pentadiagonal matrix, and U is a matrix with non- zero entries only on the upper two diagonals. VLGS sets the coefficients in the lower triangle to zero ($f_{j,k}$, $h_{j,k} = 0$) for a right to left sweep through the mesh and likewise zeros the coefficients in the upper triangle ($g_{j,k}$, $i_{j,k} = 0$) for a right to left sweep. This can be written as

$$\left[\overleftarrow{L}^{O(R\rightarrow L)} + B + \overleftarrow{U}^{O(L\rightarrow R)} \right] \Delta Q = -[R] \qquad [3.6]$$

where the arrows indicate that the particular matrix is zeroed for the corresponding sweep direction.

Note that other common relaxation schemes can be readily obtained by reordering the grid and setting other combinations of the coefficients in the difference equation to zero. For example, a bottom to top horizontal relaxation scheme is obtained by ordering in a row-wise fashion and setting $c_{j,k}$, $e_{j,k} = 0$.

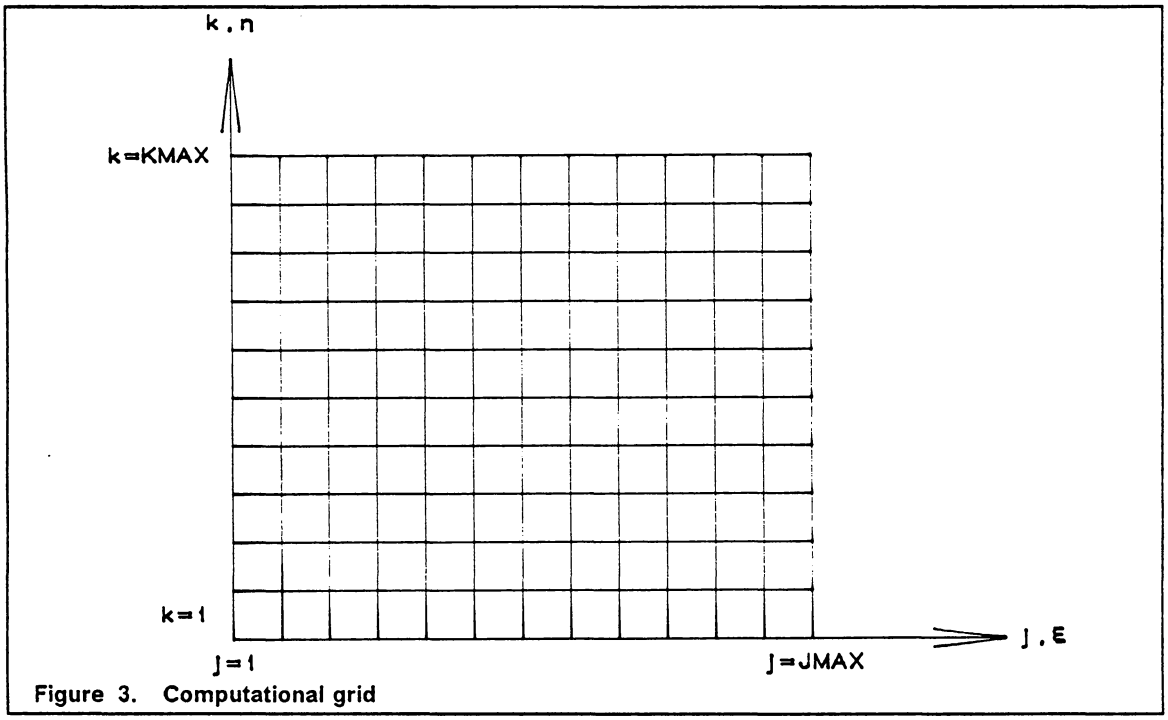


Figure 3. Computational grid

VLGS solves the equations on a ξ = constant line (Figure 3) updates $Q_{j,k}$ (equation [3.2]) and then steps to the next line and continues. This is repeated for all the ξ = constant lines. The entire process is repeated in order to obtain a steady state solution. The direction of sweep is either forward (left to right) or backward (right to left). The flow direction is in the direction of increasing ξ, so a forward sweep is in the direction of the flow. Using an ξ = constant line for VLGS amounts to implicitly treating all information in the direction normal to the surface (η) and explicitly treating upstream or downstream information, depending on the sweep direction. For supersonic inviscid flow all the characteristic information is travelling downstream. With upwind interpolation and sweeping in the flow direction, VLGS becomes a direct solver and recovers Newton's method for large time steps[13]. Therefore, supersonic inviscid flows can be space marched with this method. Also, in supersonic viscous flows almost all of the information is contained in the VLGS molecule. For subsonic flows the information will travel in all directions so the sweep direction must be alternated to obtain the necessary propagation of information[10]. If the direction of sweep is in the increasing ξ-direction then the lower two diagonals, corresponding to previous ξ = constant lines, of the coefficient matrix have already been computed for the current iteration and can be transferred to the RHS. For the backward sweep, the upper two diagonals are similarly transferred. The result is a pentadiagonal matrix on a ξ = constant line which can be easily decomposed into upper and lower triangle matrices. ΔQ can then be solved for by forward and backward substitution.

The criteria for convergence used is the L_2 norm of the residual or steady state governing equations given as

$$||R^n||_2 = \left[\sum_{k=1}^{KMAX} \sum_{j=1}^{JMAX} \sum_{l=1}^{4} (R^n_{j,k,l})^2 \right]^{1/2} \qquad [3.7]$$

where l refers to each of the four governing equations.

The sweeping is continued until the residual or the L_2 norm of R is reduced to machine zero.

Zonal logic

The introduction of patching complicates the transfer of flux across interfaces. For this study, the interface has been constrained to be an ξ = constant line, and ,at the interface, interpolation is first or second-order upwind. Also, spacing is constant in the ξ- direction.

Figure 4 shows a simple interface between two Cartesian grids. In order to conserve the flux, the line integral of

$$\int_1^K (\mathbf{F} \bullet \mathbf{n}) dS \qquad [4.1]$$

must be the same in each zone. By introducing a discrete line integral along the interface, this can be expressed as

$$\sum_{k^{(i)}=1}^{K-1^{(i)}} \int_{S^{(i)}_{k-\frac{1}{2}}}^{S^{(i)}_{k+\frac{1}{2}}} F_k^{(i)} \, dS \quad = \quad \text{constant; } i = 1,2,... \qquad [4.2]$$

where $F_k^{(i)}$ is the flux evaluated on the interface of the k^{th} cell in zone i.

Equation [4.2] is a mathematical statement that the total area under the curve of $F^{(i)}$ versus S is constant, independent of the interface discretization. Using this relation the flux can be

transferred across an interface. This is done by first noting that the discretized fluxes are piece-wise constant for each cell. The line integral of the fluxes in the source zone is then calculated. From this information and enforcing the conservation requirement of equation [4.2] the flux can be transferred to the target zone. The geometric interpretation of this statement is shown in Figure 5 for a transfer to the k^{th} cell in zone 2 from zone 1 and is expressed as

$$F_k^{(2)}\left(S_{k+\frac{1}{2}}^{(2)} - S_{k-\frac{1}{2}}^{(2)}\right) = \sum_{k^{(1)}=1}^{K-1^{(1)}} F_k^{(1)}\left(S_{k+\frac{1}{2}}^{(1)} - S_{k-\frac{1}{2}}^{(1)}\right) N_{k^{(1)}k^{(2)}} \qquad [4.3]$$

where

$$N_{k^{(1)}k^{(2)}} = \begin{cases} 0\,, & S_{k+\frac{1}{2}}^{(1)} \le S_{k-\frac{1}{2}}^{(2)} \text{ or } S_{k-\frac{1}{2}}^{(1)} \ge S_{k+\frac{1}{2}}^{(2)} \\ \dfrac{\min\left[S_{k+\frac{1}{2}}^{(1)}\,,\, S_{k+\frac{1}{2}}^{(2)}\right] - \max\left[S_{k-\frac{1}{2}}^{(1)}\,,\, S_{k-\frac{1}{2}}^{(2)}\right]}{\left(S_{k+\frac{1}{2}}^{(1)} - S_{k-\frac{1}{2}}^{(1)}\right)}, & \text{otherwise} \end{cases}$$

If this requirement is met for each cell in zone 2 then conservation will be enforced for a transfer from zone 1 to zone 2.

The spatial order of accuracy must be looked at next. For this work, the flux transfer is achieved by the trapezoidal rule, therefore the transfer across the zonal boundary is at most second-order accurate in space.

The implementation of this procedure may be accomplished in the computational plane by the expression

$$\left[\int_S (F^+ + F^-)\, dS\right]_1 = \left[\int_S (F^+ + F^-)\, dS\right]_2 \qquad [4.4]$$

where the subscripts 1 and 2 refer to upstream and downstream zones, respectively.

By breaking the integrals into parts the zonal logic can be implemented as

$$\left[\int_S F^+ dS\right]_1 + \underline{\left[\int_S F^- dS\right]_1} = \underline{\left[\int_S F^+ dS\right]_2} + \left[\int_S F^- dS\right]_2 \qquad [4.5]$$

The underlined integrals require information from the adjacent zone and the arrows show where the information is to be obtained. The general conservative flux transfer is applied to obtain the interpolated information. The Jacobians of the flux are to be transferred in the same way.

For second-order upwind differencing in the streamwise direction, there is one more step to preserve the second-order treatment of the interface. Consider again the simple interface shown in Figure 4. A conservative flux evaluation is assured at the interface, but at the $j+1/2$ and j-3/2 cell faces, the second-order upwind interpolation for Q_L and Q_R requires values of Q at locations in the adjacent zone. This can be corrected by linear interpolating in the k-direction in the adjacent zone for the necessary values of Q. The interpolated values can then be used to maintain the second-order nature of the solution.

The above method has not mentioned the viscous terms because the thin- layer treatment coupled with a body-orientated orthogonal grid will not require the calculation of the viscous terms on the $j \pm 1/2$ cell faces. Therefore, the above method will conserve the flux and maintain a spatial second-order accurate solution across zonal boundaries for both inviscid and viscous calculations.

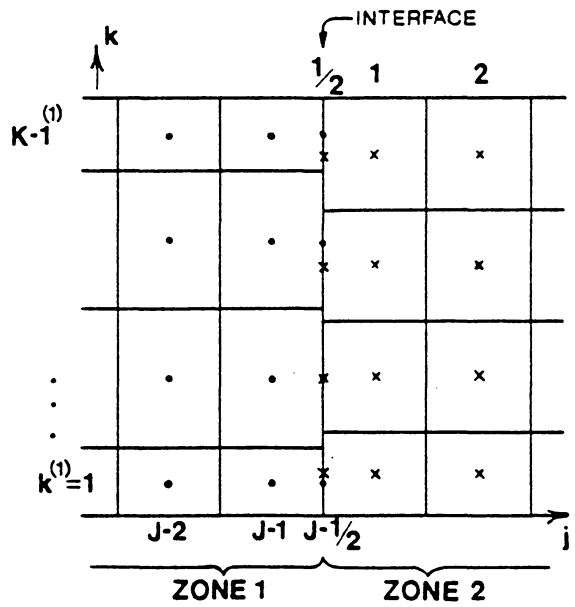


Figure 4. Cell notation for a simple interface

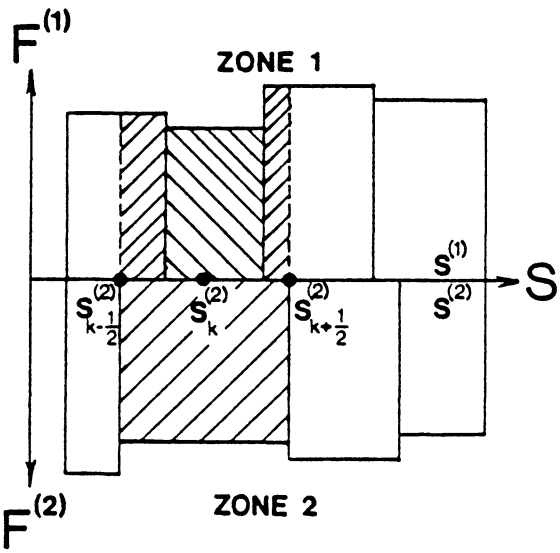


Figure 5. Geometric interpretation of the flux conservation condition

Test cases and results

The results will be presented in order of increasing complexity. The simplest is the inviscid shock reflection case. This is the simplest case since it only involves the Euler equations. For this test case the order of spatial accuracy has been varied from first to third. Having verified that the zonal logic is working, the next step is to incorporate the thin-layer terms. Two cases are chosen to test the viscous terms. The first is laminar flow over a flat plate, and the final case is a shock-boundary layer interaction problem. The laminar boundary layer problem will be compared to the analytical results of Blasius. The shock-boundary layer interaction problem will demonstrate the ability to improve the accuracy of the solution on a patched grid as well as be compared to experimental data.

All calculations have been completed on a Perkin-Elmer 3230 system computer.

Shock reflection case

The physical space is a rectangular region 4 by 1.5 inches. The freestream Mach number M_∞ is 2.9. A shock enters the domain at 1.0 inch above the surface at an angle of 29.0 ° with respect to the surface. It then reflects off the surface and leaves the domain at the outflow boundary. The exact steady state solution is made up of three different regions separated by shocks. The boundary conditions for the inflow are fixed and extrapolated on the outflow. The top boundary conditions are over-specified by enforcing the exact solution. The surface boundary conditions used in the calculations are

$$1)\ p_{j,1} = p_{j,2}$$
$$2)\ \rho_{j,1} = \rho_{j,2}$$
$$3)\ \mathbf{V} \bullet \mathbf{n} = 0$$
$$4)\ H_{j,1} = H_{j,2}$$

where k=1 is the surface and **V** is the velocity vector

The shock reflection case has been run on three different grids. The three grids are shown in Figure 6, Figure 7, and Figure 8. All three are unstretched Cartesian grids. The single zone grid is 61 by 31 mesh points. The double zone grid is 9 by 31 and 53 by 32 points. The second zone is made by sliding the grid one half cell down at the zonal interface. The interface for the double zone grid is at 0.533 inches from the inflow. The triple zone grid is 9 by 31 and 45 by 32 and 9 by 31 points. The third zone is equivalent to the single zone grid in all aspects. The second zonal interface is at 3.533 inches from the inflow.

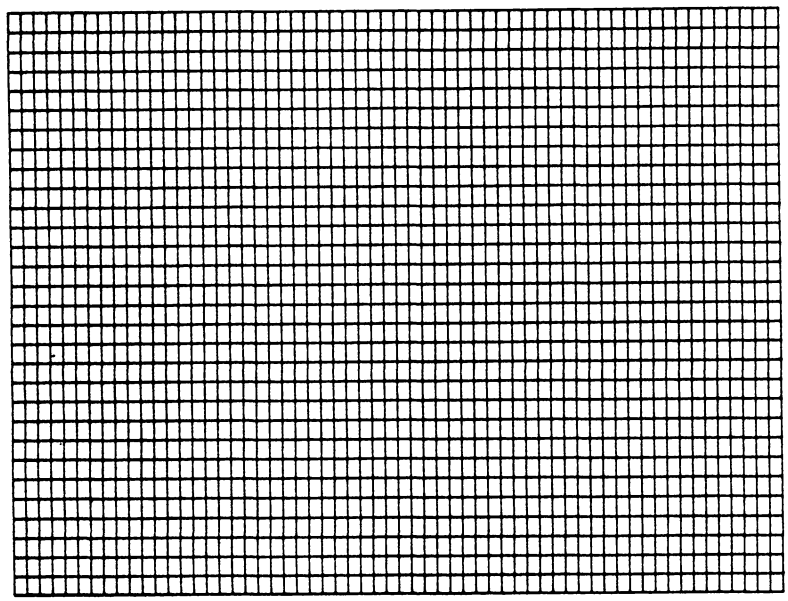


Figure 6. Single zone shock reflection grid

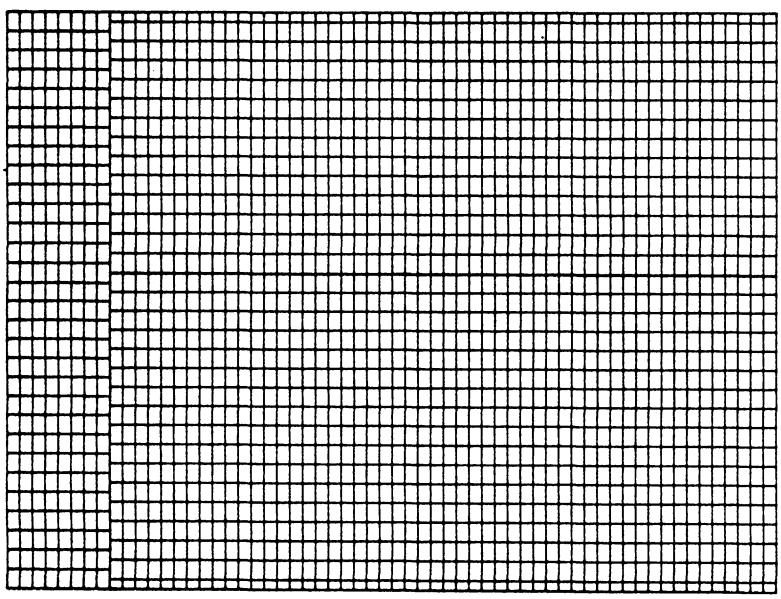


Figure 7. Double zone shock reflection grid

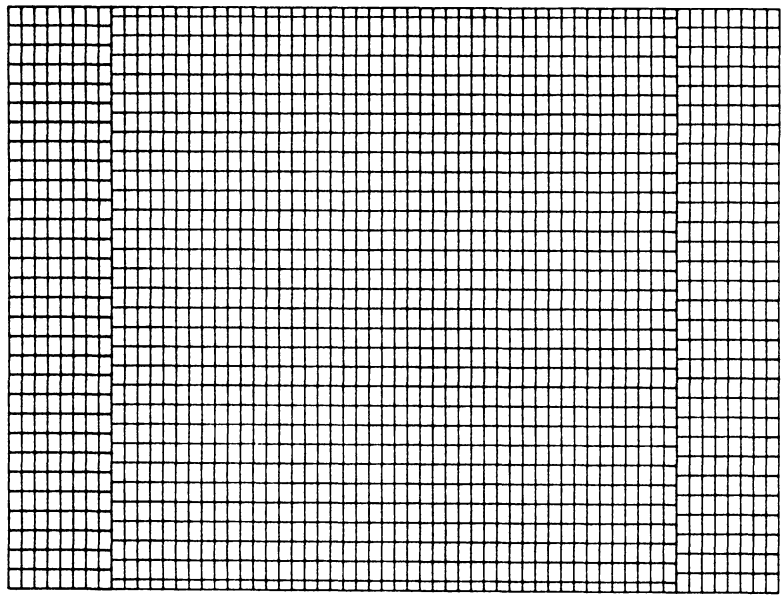
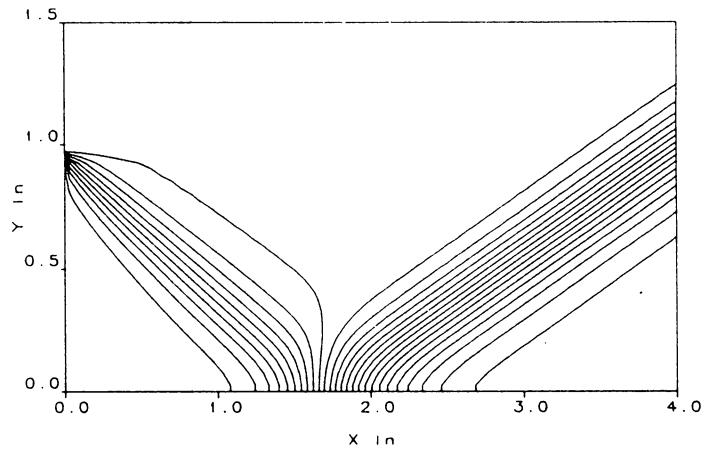


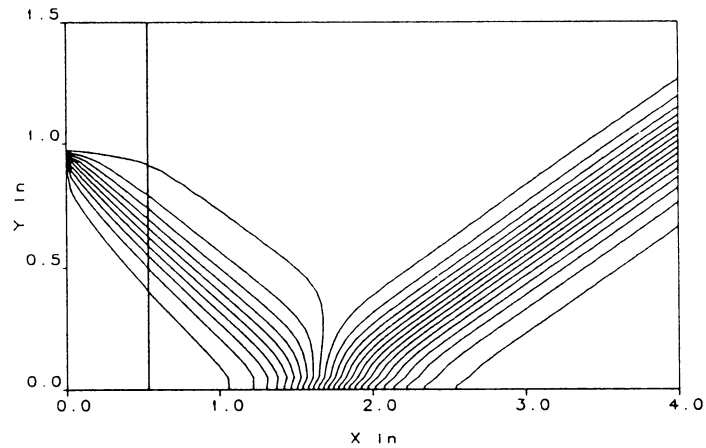
Figure 8. Triple zone shock reflection grid

First-order solution

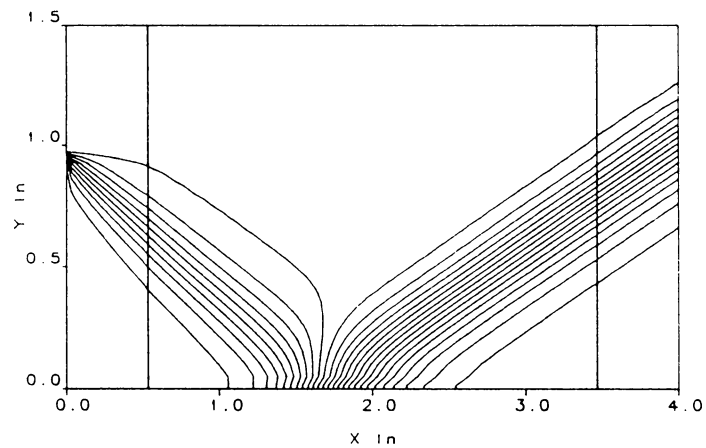
Due to the numerical dissipation associated with the first-order scheme the shock is smeared (or spread) over a wide range of cells as shown in Figure 9. The double zone contours are the same as the single zone for the incoming shock. The difference between zonal and non-zonal solutions arises on reflection from the surface. This can be explained by the difference in grids. The grid spacing next to the surface is smaller in the zonal grid so one would expect less smearing of the reflected shock as compared to the single zone solution. The triple zone pressure contours are the same as the double zone contours. This shows that the third grid has negligible effect on the solution. The important quality about the comparison of the different pressure contour plots is that the shock is passed through the zonal boundary without any distortion in size or orientation. This is because of the conservative treatment at the interfaces. One of the interesting aspects of the shock reflection case is the wall pressure (Figure 10). The addition of the second zone gives a better match of the ideal pressure discontinuity. The third zone has no effect on the wall pressure. This is due to the wall pressure being constant at the second zonal interface. The convergence history shows nearly quadratic convergence to the steady-state solution (Figure 11). The VLGS algorithm reaches a machine zero convergence after 10 iterations for all three grids.



a) Single zone



b) Double zone



c) Triple zone

Figure 9. First-order pressure contours

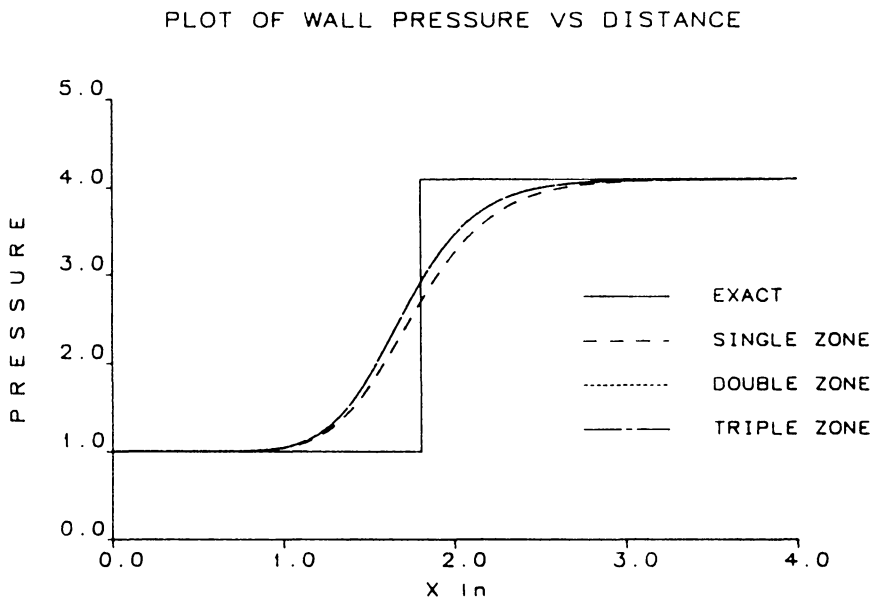


Figure 10. First-order wall pressure comparison

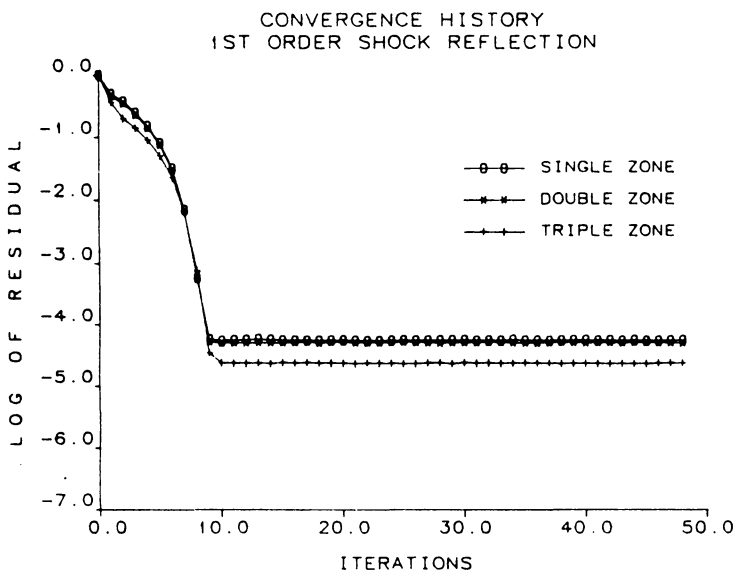
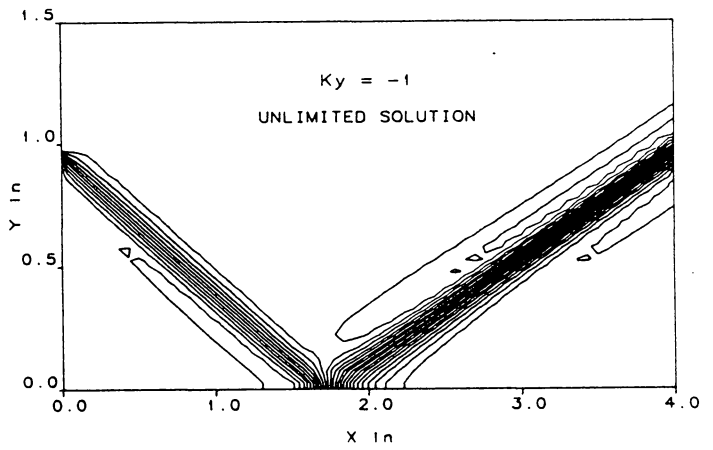


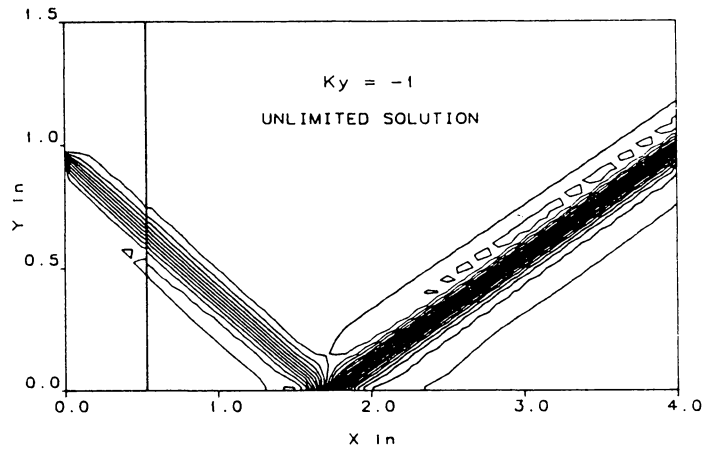
Figure 11. First-order convergence history

Second-order solution

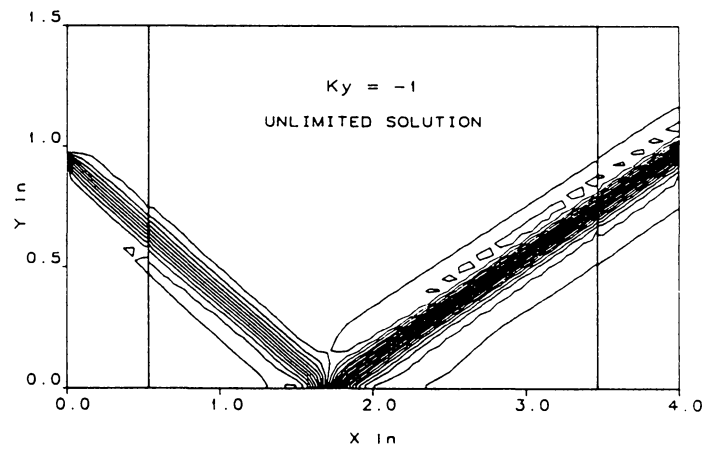
The second-order solution is fully upwinded ($\kappa=-1$). The comparison with the first-order pressure contours shows better capturing of both the incident and reflected shocks (Figure 12). In addition, the pressure contours on the two and three zone grids show less smearing of the reflected shock as compared with the single zone grid. The shock passes through the zonal interface without any change in the strength or shock angle, but as can be seen, the contour lines do not match up at the zonal interface. This is due the fact that the dependent variables are known at cell centers, not on the zonal interface and must be interpolated before being passed to the contour plotting package presently in use. The wall pressure comparison is shown in Figure 13. The second-order over and under shoots of pressure, which are typical of unlimited solutions ($S_{j,k} = 1$), are larger in the multiple zones as compared to the single zone. The wall pressure comparison also shows that the pressure rise is better on the multiple zone grids. The convergence history plot in Figure 14 shows that all of the grids have achieved a steady-state after 15 iterations.



a) Single zone



b) Double zone



c) Triple zone

Figure 12. Second-order pressure contours

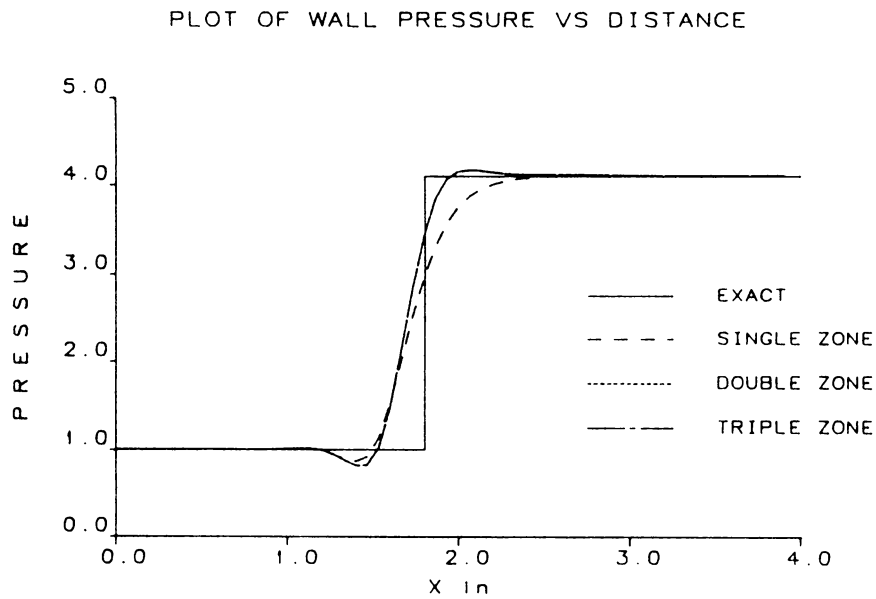


Figure 13. Second-order wall pressure comparison

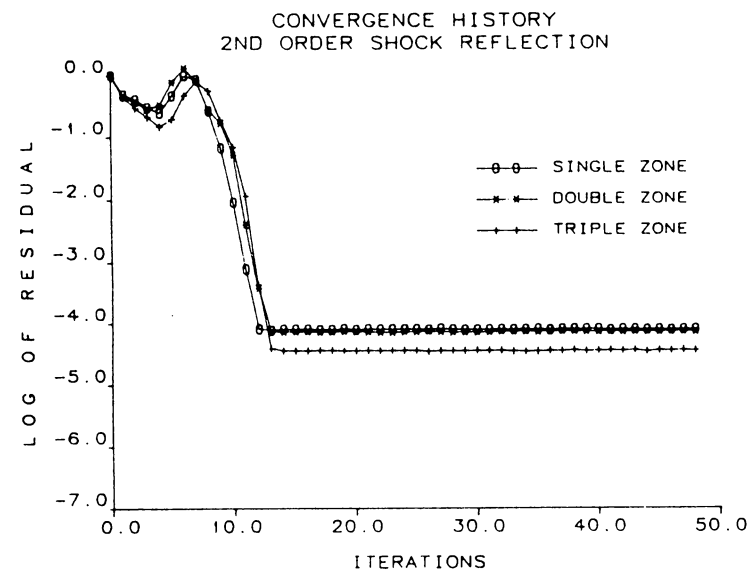
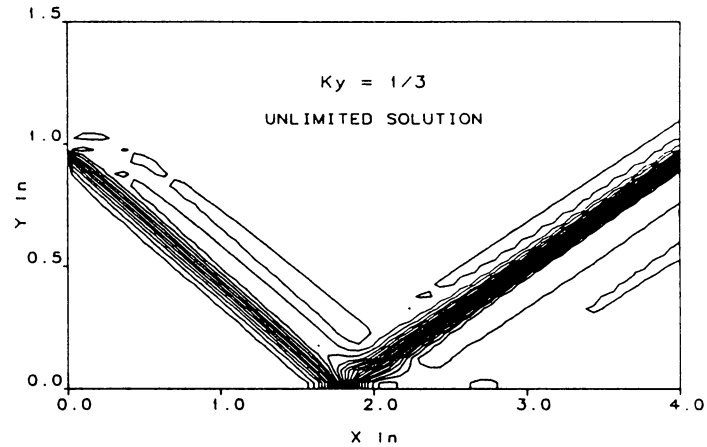


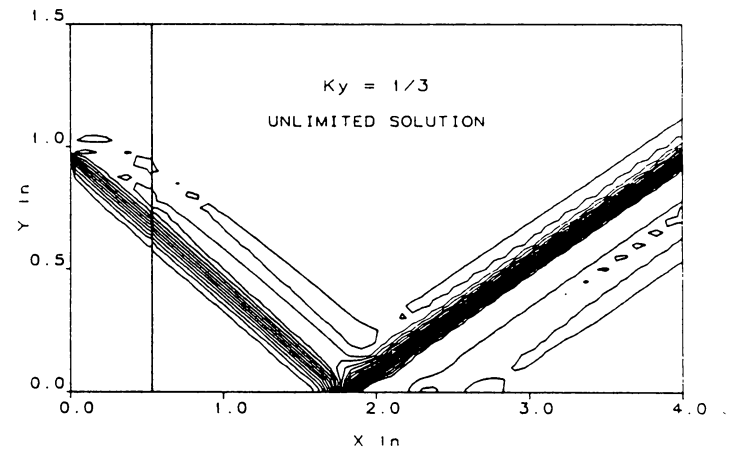
Figure 14. Second-order convergence history

Third-order solution ($\kappa = 1/3$)

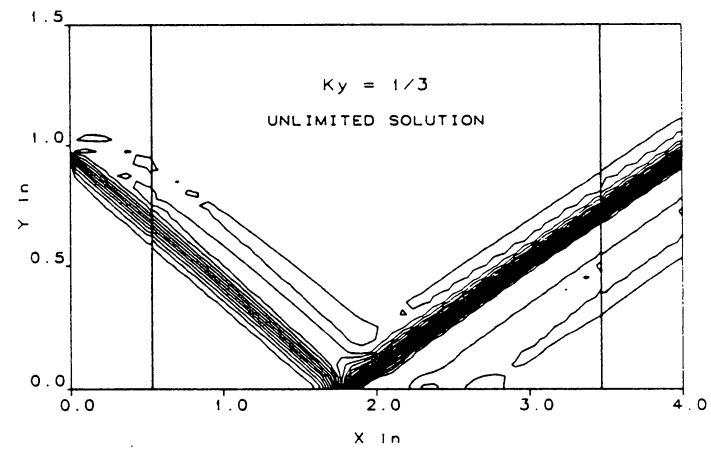
The single zone pressure contours from the upwind biased solution show better capturing of the incident and reflected shock (Figure 15) as compared to the second-order results. The unlimited solution ($S_{j,k} = 1$) show oscillations above the incident and surrounding the reflected shock. The double zone pressure contours again lie on top of the single zone for the incident shock and display better capturing of the reflected shock near the wall. The triple zone results are the same as the double zone. The wall pressure comparison in Figure 16 shows over and under shoots on all grids, most notably in the multiple zones. The convergence history in Figure 17 again shows similar convergence histories for all grids.



a) Single zone



b) Double zone



Triple zone

Figure 15. Third-order pressure contours

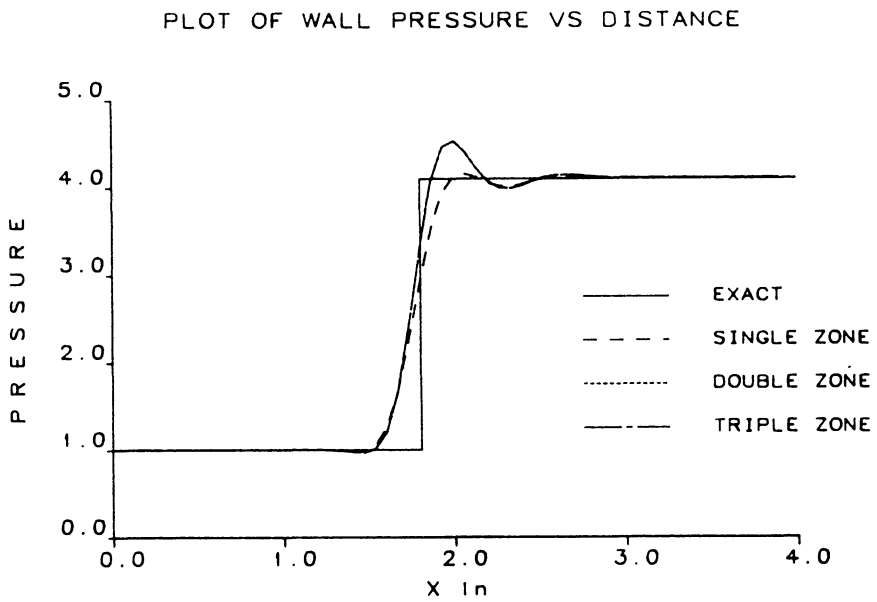


Figure 16. Third-order wall pressure comparison

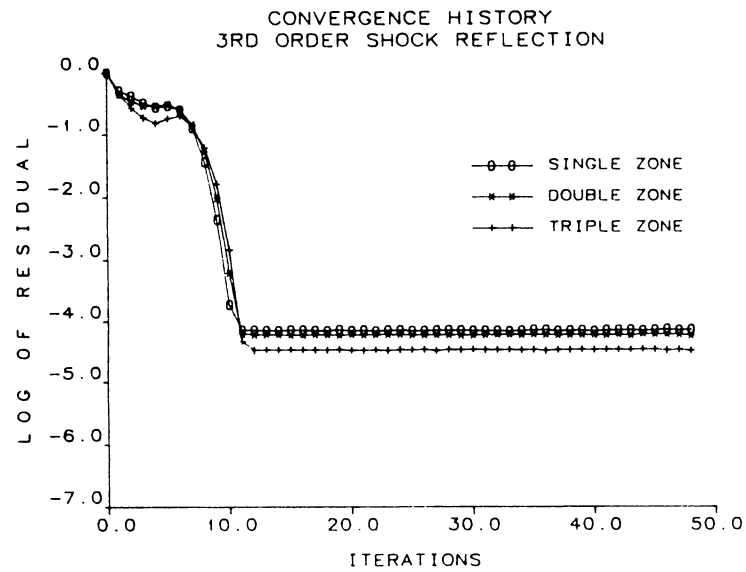


Figure 17. Third-order convergence history

Subsonic boundary layer

The physical space is a rectangular region 0.30 feet wide by 0.20 feet high, and the freestream Mach number M_∞ is 0.5. This case will be used to test the viscous terms and verify that the subsonic flow of information is being correctly transferred across the zonal boundary. The computational results will be compared with the incompressible Blasius similarity solution. The case is second order upwind in the streamwise direction and κ is 1/3 in the normal direction. No limiting has been used for this case ($S_{j,k} = 1$), and the sweep direction is reversed every iteration. The boundary conditions that are used at the inflow boundary ($j = 1$) are

1) constant total enthalpy
2) constant total pressure
3) $v = 0$
4) $u_{1,k} = u_{2,k}$

The boundary conditions on the outflow ($j = jmax + 1$) are

1) $\rho_{jmax+1,k} = \rho_{jmax,k}$
2) $u_{jmax+1,k} = u_{jmax,k}$
3) $v_{jmax+1,k} = v_{jmax,k}$
4) back pressure specified

The top boundary conditions are obtained by extrapolation of all conservative variables. The lower boundary conditions ($k = 1$) are

1) $p_{j,1} = p_{j,2}$
2) $\rho_{j,1} = \rho_{j,2}$
3) $v_{j,1} = 0$
4) $H_{j,1} = H_{j,2} \quad j < j_{plate}$
   $u_{j,1} = 0 \quad j \geq j_{plate}$

where j_{plate} refers to the starting location of the plate.

The single zone grid used for the boundary layer profile has a constant spacing in the streamwise direction and an exponential stretching in the vertical direction (Figure 18). The grid is 61 by 57 in the x and y directions, respectively. The smallest vertical spacing is $\Delta y_{min} = 1.X10^{-4}$ and the largest is $\Delta y_{max} = 6.39X10^{-3}$ The equation used to generate the grid is

$$y_1 = 0$$

$$y_j = y_{j-1} + \Delta y_{max}\left(\frac{\Delta y_{min}}{\Delta y_{max}}\right)^{\frac{j^* - (j-1)}{j^* - 1}} \qquad j = 2, j^*$$

$$y_j = y_{j-1} + \Delta y_{max} \qquad j \geq j^*$$

where j^* is 33 for the single zone grid. The two zone grid also has a constant spacing in the x-direction. The first zone is the same as the single zone grid, but the second zone contains more points in the normal direction (Figure 19). The values of Δy_{max} and Δy_{min} are half of the single zone spacings. The value of j^* is 65. The double zone grid is 29 by 57 and 33 by 113. The zonal interface is at x=0.14 feet.

The plate starts at 0.05 feet from the inflow so the plate itself is 0.25 feet long. From the results five different body stations are chosen to compare with the Blasius similarity solution. The comparison with the Blasius profile is done to give a reference but the main objective of the comparison is to show that the zonal interface does not effect the profiles. In figure Figure 20 the five different profiles lie very close to the Blasius profiles and directly on top of each other. The distances in Figure 20 are measured from the leading edge of the flat plate so they correspond to 13%, 29%, 41%, 61%, and 81% locations on the plate. For the double zone results the zonal interface is at 0.09 feet or 36% on the plate. The three profiles that lie in the second zone give a better correlation with the Blasius profile and also are very close to each other. This suggests that the single zone grid may not be refined enough to correctly

resolve the boundary layer. The convergence history plot (Figure 22) shows similar behavior between the two grids and very little difference in the number of iterations to convergence.

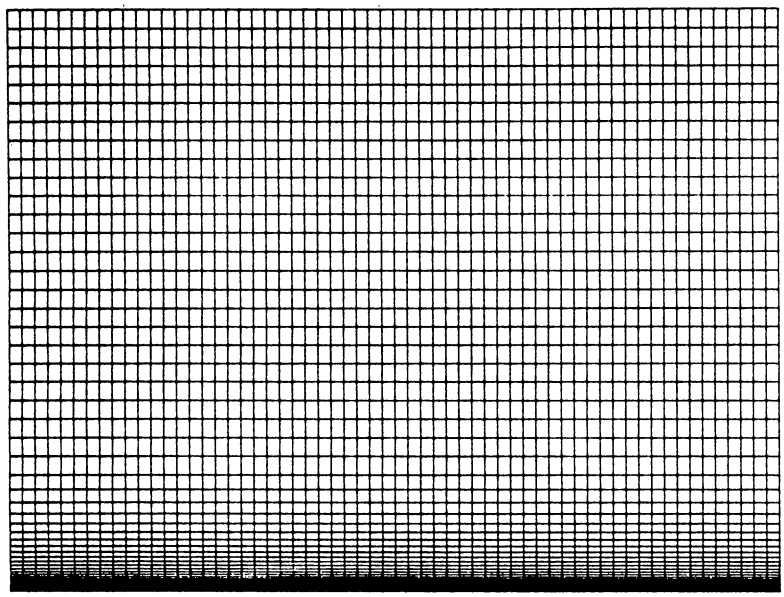


Figure 18. Single zone viscous grid

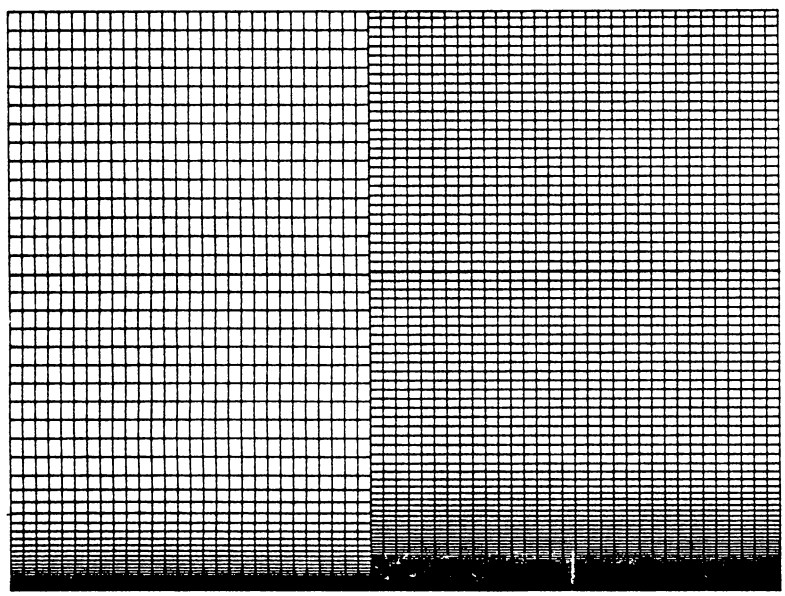


Figure 19. Double zone viscous grid

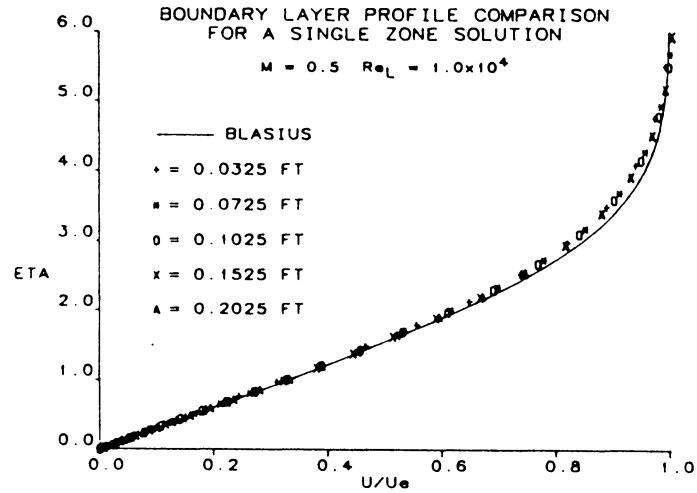


Figure 20. Single zone velocity profile comparison

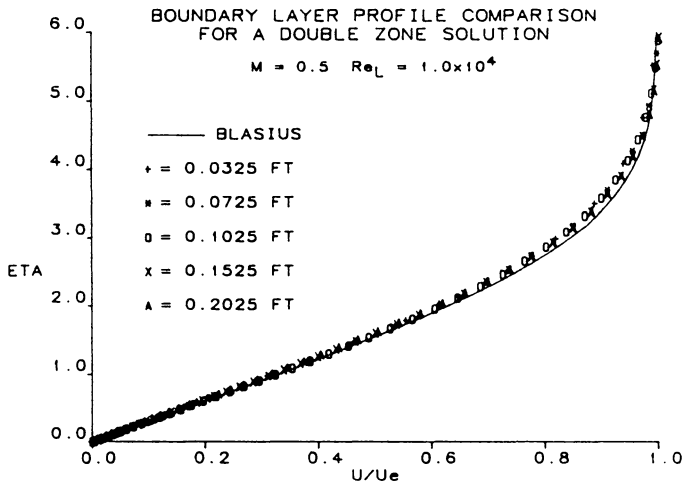


Figure 21. Double zone velocity profile comparison

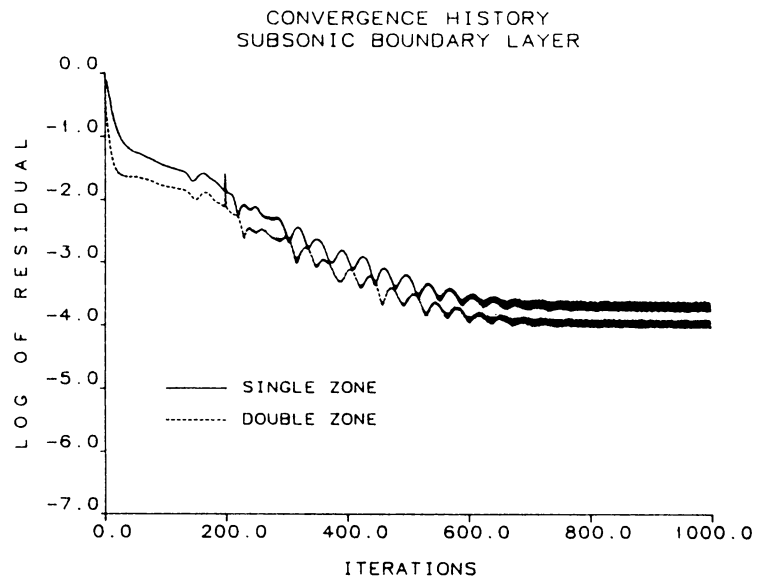


Figure 22. Convergence history comparison

Shock-boundary layer interaction

The physical space is a rectangular region 0.30 feet by 0.20 feet. The freestream Mach number M_∞ is 2.0. The shock enters the domain at 0.116 feet above the flat plate, at an incident angle of 32.6 ° with respect to the plate. The shock then impinges on a laminar boundary layer with sufficient strength to cause separation. The Reynolds number based on the shock location, $Re_{x_{shk}}$, is $2.96X10^5$ and the distance from the leading edge of the plate to the shock, x_{shk}, is 0.162 feet. This is a problem that has been computed by many other researchers on single grids[10,14-16]. The purpose of this problem is to demonstrate the usefulness of patched grids and compare the results with experimental data[17]. The accuracy is second-order upwind ($\kappa=-1$) and third-order upwind biased ($\kappa=1/3$) in the streamwise and normal directions, respectively. No limiting has been used for this case ($S_{j,k} = 1$), and the sweep direction is alternated every iteration. The boundary conditions for the inflow are prescribed to be the freestream conditions below and post-shock conditions above the incoming shock. The outflow conditions are extrapolated. The upper boundary is set to the post-shock conditions and the lower boundary ($k=1$) is

$$
\begin{aligned}
&1)\ p_{j,1} = p_{j,2} \\
&2)\ \rho_{j,1} = \rho_{j,2} \\
&3)\ v_{j,1} = 0 \\
&4)\ H_{j,1} = H_{j,2} \quad j < j_{plate} \\
&\quad\ u_{j,1} = 0 \quad\ \ j \geq j_{plate}
\end{aligned}
$$

where j_{plate} refers to the starting location of the plate.

The grids used are the same as the boundary layer problem as shown in Figure 18, and Figure 19. The pressure contours for the single zone results are shown in Figure 23 and the double zone results in Figure 24. The physical interaction of the shock with the laminar

boundary layer will produce three reflected waves; a compression waves at separation, an expansion wave, and another compression wave at reattachment. The single zone grid resolves the incoming shock and the reflected compression waves but not the expansion wave inbetween. The second zone of the double zone grid is refined to properly show the physics of all the reflected waves. There is no noticeable difference in the pressure contours for the single zone and the first zone of the double zone grid, and the interface displays no distortion of the solution. The number of points for the single zone grid is 3477 and for the double zone grid 5382. Other researchers have demonstrated that the necessary single zone grid to correctly resolve the problem is 61 by 113 or 6893 points[10] . This is a 28% increase over the patched grid. The addition of the patched grid results in about 25% savings in computer time per iteration. The difference of 3% is due to the added calculations to correctly transfer information across zonal boundaries. The comparison of the present solutions and experimental results[17] for the wall pressure are shown in Figure 25. Both the single and double zone wall pressures closely follow the experimental results with the plateau of the double zone being longer. The pressure recovery for both grids is very close to the experimental results. The skin friction shown in Figure 26 also compares the computational and experimental results. Both the single and double zones show the shock-boundary layer interaction in the separated region where the experimentalists were unable to obtain measurements[17] . The double zone shows more details of the interaction, and the single zone has a smaller separated region due to the coarser grid. The comparison between computational and experimental results show a general agreement between the two. For the convergence history the shape of the curves are similar (Figure 27). The double zone requires about 400 more iterations to reach convergence as compared to the single zone. The addition of more grid points that would be necessary to obtain a correct single zone solution would slow the rate of convergence even more than the patched grid solution.

The results of the shock-boundary layer interaction problem have demonstrated the potential savings by incorporating patched grids. The favorable comparison between experimental and

computational results indicate that the zonal boundaries have not adversely effected the solutions, and has in fact increased the resolution while saving on computational time.

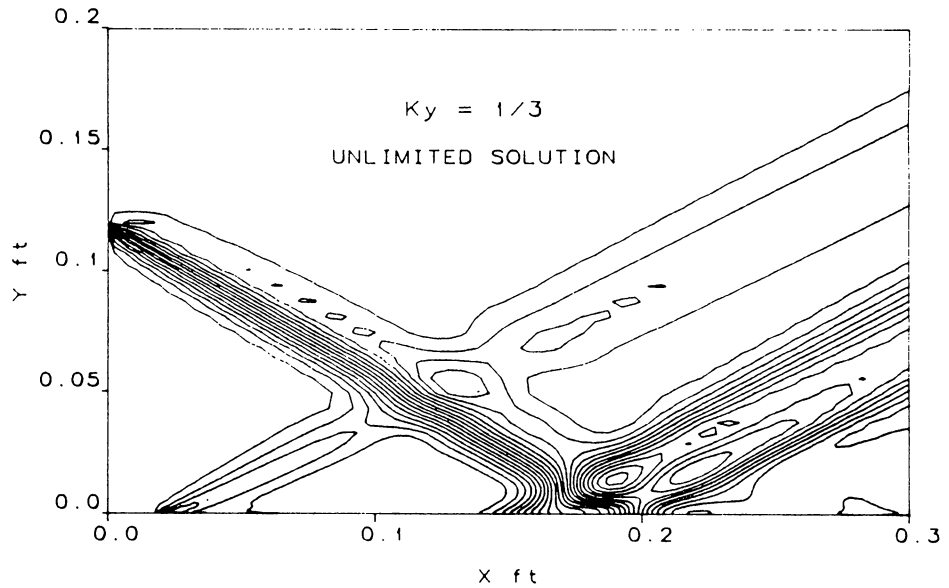


Figure 23. Single zone shock-boundary layer interaction pressure contours

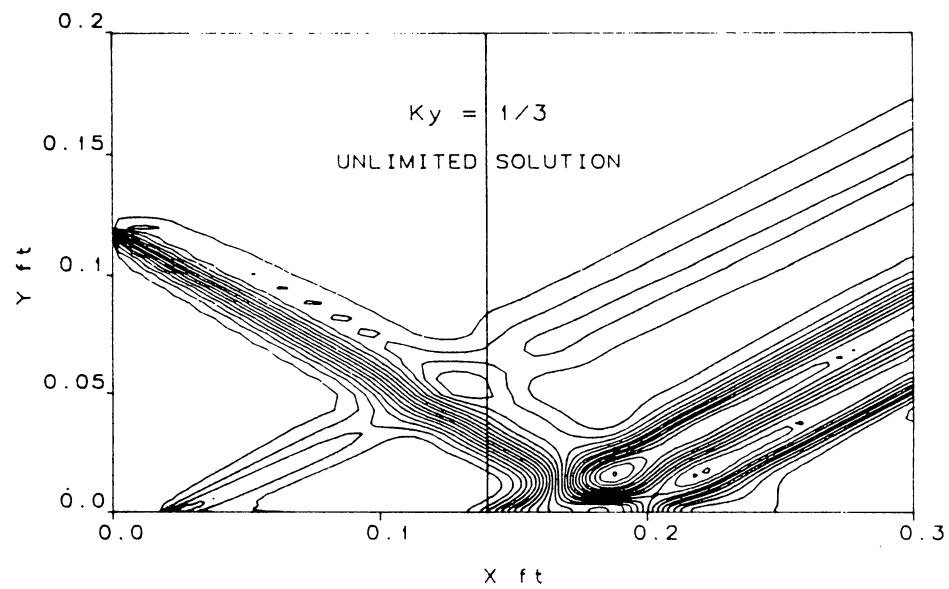


Figure 24. Double zone shock-boundary layer interaction pressure contours

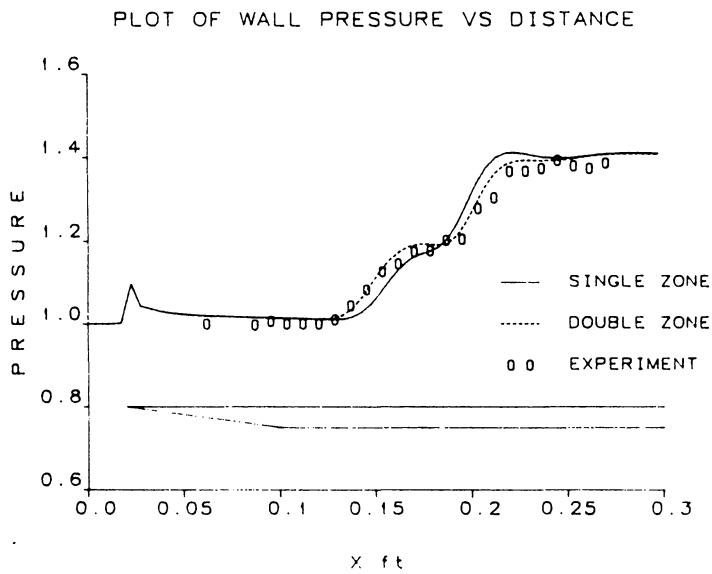


Figure 25. Shock-boundary layer interaction wall pressure comparison

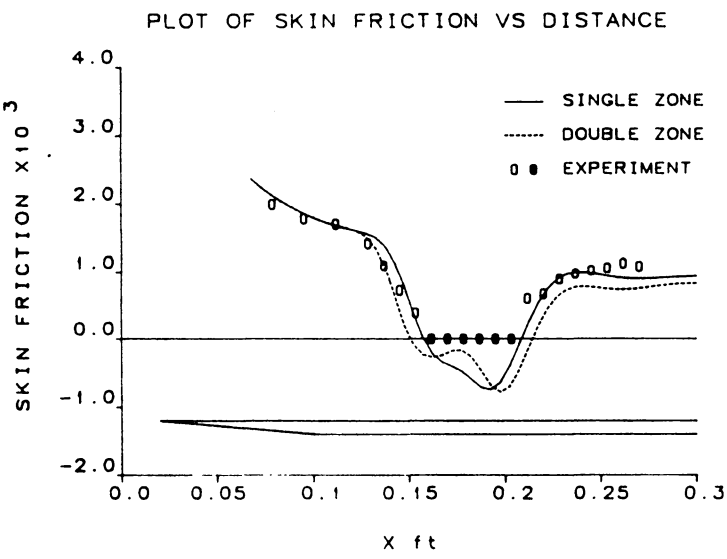


Figure 26. Shock-boundary layer interaction skin friction comparison

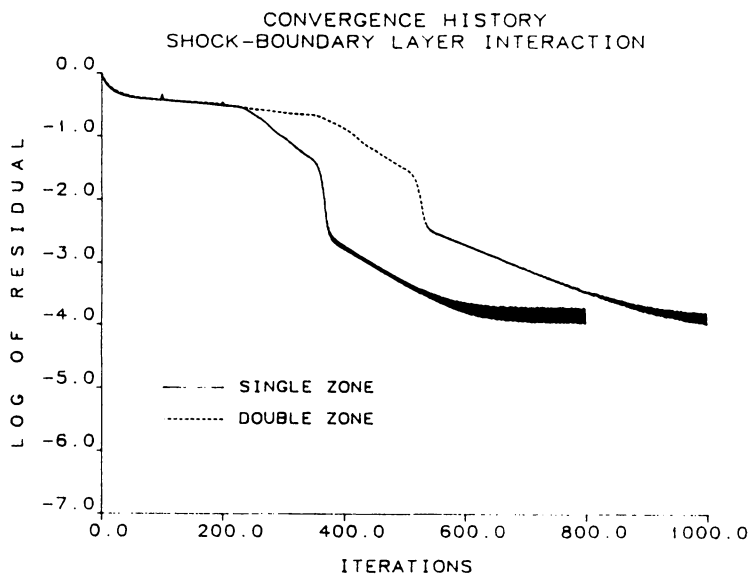


Figure 27. Shock-boundary layer interaction convergence history comparison

Concluding Remarks

The emphasis of this report has been to examine the use of patched grids to obtain solutions of the Euler and thin-layer Navier-Stokes equations. A conservative treatment of zonal boundaries allows information to pass through without any distortion or loss of accuracy for both first and second-order fully upwind solutions. The inviscid results show that the zonal treatment is conservative, and the thin- layer results show the benefits of incorporating patched grids. Furthermore, the results of the viscous cases have closely agreed with both theory and experiment. The patched grid scheme has been demonstrated to be a conservative way to obtain the necessary local grid refinement while saving on computer time and storage.

References

[1] Mastin, W. C., "Numerical Grid Generation", Thompson, J. F., ed., North-Holland, Elsevier Science Publishing Co., New York, pp. 31-40.

[2] Rai, M. M., "An Implicit, Conservative, Zonal-Boundary Scheme for Euler Equation Calculations", AIAA Paper 85-0488.

[3] Rai, M. M., "A Relaxation Approach to Patched-Grid Calculations with the Euler Equations", AIAA Paper 85-0295.

[4] Rai, M. M., "A Conservative Treatment of Zonal Boundaries for Euler Equation Calculations", AIAA Paper 84-0164.

[5] Hessenius, K. A., "Three-Dimensional, Conservative, Euler Computations using Patched Grid Systems and Explicit Methods", AIAA Paper 86-1061.

[6] Van Leer, B., "Flux-Vector Splitting for the Euler Equations", ICASE Report No. 82-30, September 1982,: also, Lecture Notes in Physics, Vol. 170, 1982, pp. 507-512.

[7] Anderson, W. K., Thomas, J. L., and Van Leer, B., "A Comparison of Finite Volume Flux Vector Splittings for the Euler Equations", AIAA Paper 85-0122.

[8] Van Leer, B., "Towards the Ultimate Conservative Difference Scheme V. A Second-Order Sequel to Godonov's Method", Journal of Computational Physics, Vol. 32, 1979, pp. 101-136.

[9] Mulder, W. A., Van Leer, B., "Implicit Upwind Methods for the Euler Equations", AIAA Paper 83-1930.

[10] Thomas, J. L., Walters, R. W., "Upwind Relaxation Algorithms for the Navier Stokes Equations", AIAA Journal, Vol. 25, No. 4, April 1987.

[11] Golub, G. H., Van Loan, C. F., "Matrix Computations", John Hopkins University Press, Baltimore, Maryland, 1985, pp. 92-93.

[12] Varga, R. S., "Matrix Iterative Analysis", Prentice- Hall, Englewood Cliffs, N. J., 1962, pp. 1-3.

[13] Anderson, D. A., Tannehill, J. C., Pletcher, R. H., "Computational Fluid Mechanics and Heat Transfer", McGraw-Hill, Hemisphere Publishing Corporation, 1984, p. 369.

[14] MacCormack, R. W., "A Numerical Method for Solving the Equations of Compressible Flow", AIAA Journal, Vol. 20, No. 9, September 1982.

[15] Chakravarthy, S. R., Szema, K-Y., Goldberg, U. C., Gorski, J. J., and Osher, S., "Application of a New Class of High Accuracy TVD Schemes to the Navier-Stokes Equations", AIAA Paper 85-0165.

[16] Beam, R. W., Warming, R. F., "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations", AIAA Journal, Vol. 16, No. 4, April 1978.

[17] Hakkinen, R. J., Greber, I., Trilling, L., Abarbanel, S. S., "The Interaction of an Oblique Shock Wave with a Laminar Boundary Layer", NASA Memo-2-18-59W, March 1959.

Appendix A. Nondimensionalization of the governing equations

The governing equations without body forces or external heat addition in integral form are (the bar superscript denotes a dimensional quantity)

$$\frac{\partial}{\partial \bar{t}} \int_{\bar{V}} \overline{Q} d\overline{V} + \oint_{\bar{S}} (\overline{\mathbf{F}} \bullet \mathbf{n}) d\overline{S} = 0 \qquad [A.1]$$

where

$$\overline{\mathbf{F}} = \overline{F}\mathbf{i} + \overline{G}\mathbf{j} \qquad [A.2]$$

where $\overline{Q}$, $\overline{F}$, and $\overline{G}$ are given by

$$\overline{Q} = \begin{bmatrix} \overline{\rho} \\ \overline{\rho u} \\ \overline{\rho v} \\ \overline{e} \end{bmatrix}$$

$$\bar{F} = \begin{bmatrix} \overline{\rho u} \\ \overline{\rho u^2} + \bar{p} - \bar{\tau}_{xx} \\ \overline{\rho uv} - \bar{\tau}_{xy} \\ (\bar{e} + \bar{p})\bar{u} - \bar{u}\bar{\tau}_{xx} - \bar{v}\bar{\tau}_{xy} + \bar{q}_x \end{bmatrix} \qquad G = \begin{bmatrix} \overline{\rho v} \\ \overline{\rho uv} - \bar{\tau}_{xy} \\ \overline{\rho v^2} + \bar{p} - \bar{\tau}_{yy} \\ (\bar{e} + \bar{p})\bar{v} - \bar{u}\bar{\tau}_{xy} - \bar{v}\bar{\tau}_{yy} + \bar{q}_y \end{bmatrix}$$

The viscous shear stress and heat flux terms are given by

$$\begin{aligned} \bar{\tau}_{xx} &= (\bar{\lambda} + 2\bar{\mu})\bar{u}_x + \overline{\lambda v}_y \\ \bar{\tau}_{xy} &= \bar{\mu}(\bar{u}_y + \bar{v}_x) \\ \bar{\tau}_{yy} &= (\bar{\lambda} + 2\bar{\mu})\bar{v}_y + \overline{\lambda u}_x \\ \bar{q}_x &= -\overline{kT}_x \\ \bar{q}_x &= -\overline{kT}_y \end{aligned}$$

The gas is considered to be a perfect gas and thus, the equation of state is

$$\bar{p} = \overline{\rho RT} \qquad [A.3]$$

where $\bar{R}$ is the ideal gas constant.

The equations are nondimensionalized by a reference length $\bar{L}$, velocity $\bar{V}_\infty$, density $\bar{\rho}_\infty$, temperature $\bar{T}_\infty$, and dynamic viscosity $\bar{\mu}_\infty$. The nondimensionalization used here is given by

$$x = \frac{\bar{x}}{\bar{L}} \qquad y = \frac{\bar{y}}{\bar{L}} \qquad t = \frac{\bar{t}}{\bar{L}/\bar{V}_\infty}$$

$$u = \frac{\bar{u}}{\bar{V}_\infty} \qquad v = \frac{\bar{v}}{\bar{V}_\infty} \qquad \rho = \frac{\bar{\rho}}{\bar{\rho}_\infty}$$

$$p = \frac{\bar{p}}{\bar{\rho}_\infty \bar{V}_\infty^2} \qquad T = \frac{\bar{T}}{\bar{T}_\infty} \qquad e = \frac{\bar{e}}{\bar{\rho}_\infty \bar{V}_\infty^2}$$

$$\mu = \frac{\bar{\mu}}{\bar{\mu}_\infty} \qquad \lambda = \frac{\bar{\lambda}}{\bar{\mu}_\infty}$$

If these relations are substituted into the governing equations the following nondimensional equations are obtained

$$\frac{\partial}{\partial t}\int_V Q dV + \oint_S (\mathbf{F} \bullet \mathbf{n}) dS = 0 \qquad [A.4]$$

where the flux vector is now broken into the inviscid and the viscous contributions given as

$$\mathbf{F} = \mathbf{F}_I - \mathbf{F}_v \qquad [A.5]$$

where

$$\mathbf{F}_I = F_i\mathbf{i} + G_i\mathbf{j} \qquad \mathbf{F}_v = F_v\mathbf{i} + G_v\mathbf{j}$$

the nondimensional Q, F_i, G_i, F_v, and G_v are given by

$$Q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix} \qquad F_i = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (e + p)u \end{bmatrix} \qquad G_i = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ (e + p)v \end{bmatrix}$$

$$F_v = \frac{1}{Re_L^-}\begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ u\tau_{xx} + v\tau_{xy} + q_x \end{bmatrix} \qquad G_v = \frac{1}{Re_L^-}\begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ u\tau_{xy} + v\tau_{yy} + q_y \end{bmatrix}$$

The nondimensional viscous shear stress and heat flux terms are given by

$$
\begin{aligned}
\tau_{xx} &= (\lambda + 2\mu)u_x + \lambda v_y \\
\tau_{xy} &= \mu(u_y + v_x) \\
\tau_{yy} &= (\lambda + 2\mu)v_y + \lambda u_x \\
q_x &= \frac{\mu}{(\gamma - 1)M_\infty^2 Pr} T_x \\
q_y &= \frac{\mu}{(\gamma - 1)M_\infty^2 Pr} T_y
\end{aligned}
$$

The Reynolds number is based on the reference length, $\bar{L}$ as

$$
Re_{\bar{L}} = \frac{\bar{\rho}_\infty \bar{V}_\infty \bar{L}}{\bar{\mu}_\infty} \qquad [A.6]
$$

The Prandtl number is

$$
Pr = \frac{\bar{c}_p \bar{\mu}}{\bar{k}} \qquad [A.7]
$$

The Mach number is

$$
M = \frac{\bar{V}}{\bar{a}} \qquad [A.8]
$$

where $\bar{a}$ is the speed of sound. The nondimensional equation of state is

$$
p = \frac{\rho T}{\gamma M_\infty^2} \qquad [A.9]
$$

The temperature is given by

$$
T = \frac{\gamma(\gamma - 1)M_\infty^2}{\rho}\left[e - \frac{1}{2}\rho(u^2 + v^2)\right] \qquad [A.10]
$$

where γ is the ratio of specific heats and is 1.4 for air The coefficient of viscosity, μ is calculated from Sutherland's equation

$$\mu = T^{3/2}\left(\frac{1 + C}{T + C}\right) \qquad [A.11]$$

where C is $198.7\,^{\circ}\frac{R}{\bar{T}_{\infty}} = 110.4^{\circ}\frac{K}{\bar{T}_{\infty}}$ for air. The coefficient of bulk viscosity relates λ to μ and is

$$\kappa = \frac{2}{3}\mu + \lambda \qquad [A.12]$$

κ is set to zero to give

$$\lambda = -\frac{2}{3}\mu \qquad [A.13]$$

which is Stokes' hypothesis.

Appendix B. Differential and integral form of the governing equations

Differential form

The differential form of the governing equations can be written as

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = 0 \qquad [B.1.1]$$

Where F and G contain both the inviscid and the viscous terms.

The equations can then transformed to a computational space by the relation

$$\xi = \xi(x,y) \qquad \eta = \eta(x,y) \qquad [B.1.2]$$

where ξ and η are the body-orientated coordinate directions.

The partial derivative of ξ and η by the chain rule become

$$\frac{\partial}{\partial \xi} = x_\xi \frac{\partial}{\partial x} + y_\xi \frac{\partial}{\partial y} \qquad \frac{\partial}{\partial \eta} = x_\eta \frac{\partial}{\partial x} + y_\eta \frac{\partial}{\partial y} \qquad [B.1.3]$$

In matrix notation these equations are

$$\begin{bmatrix} x_\xi & y_\xi \\ x_\eta & y_\eta \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{bmatrix} \qquad [B.1.4]$$

Inverting the above system gives

$$\frac{\partial}{\partial x} = J\left(y_\eta \frac{\partial}{\partial \xi} - y_\xi \frac{\partial}{\partial \eta}\right), \qquad \frac{\partial}{\partial y} = J\left(x_\xi \frac{\partial}{\partial \eta} - x_\eta \frac{\partial}{\partial \xi}\right) \qquad [B.1.5]$$

where J is the inverse of the determinant of the matrix and is

$$J = \frac{1}{x_\xi y_\eta - x_\eta y_\xi}$$

By applying the chain rule to the reverse transformation, the following metric relations follow

$$\begin{aligned} \xi_x &= J y_\eta \qquad \xi_y = -J x_\eta \\ \eta_x &= -J y_\xi \qquad \eta_y = J x_\xi \end{aligned} \qquad [B.1.6]$$

The Jacobian of the coordinate transformation, J, can be written as

$$J = \left|\frac{\partial(\xi,\eta)}{\partial(x,y)}\right| = \begin{vmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{vmatrix} = \xi_x \eta_y - \xi_y \eta_x = \frac{1}{x_\xi y_\eta - x_\eta y_\xi} \qquad [B.1.7]$$

Now the derivatives with respect to x, and y can be transformed to ξ, and η to yield

$$\frac{\partial}{\partial x} = \xi_x \frac{\partial}{\partial \xi} + \eta_x \frac{\partial}{\partial \eta} \qquad \frac{\partial}{\partial y} = \xi_y \frac{\partial}{\partial \xi} + \eta_y \frac{\partial}{\partial \eta} \qquad [B.1.8]$$

By using the above equations the metric relations given in [B.1.6] and assuming that $x_{\xi\eta} = x_{\eta\xi}$ and $y_{\xi\eta} = y_{\eta\xi}$ the governing equations can be written in the form

$$\frac{1}{J}\frac{\partial Q}{\partial t} + \frac{\partial \tilde{F}}{\partial \xi} + \frac{\partial \tilde{G}}{\partial \eta} = 0 \qquad [B.1.9]$$

where $\tilde{F}$ and $\tilde{G}$ are defined as

$$\tilde{F} = \frac{(\xi_x F + \xi_y G)}{J} \qquad \tilde{G} = \frac{(\eta_x F + \eta_y G)}{J} \qquad [B.1.10]$$

Integral form

The integral form of the governing equations is given by

$$\frac{\partial}{\partial t}\int_V Q dV + \oint_S (\mathbf{F} \bullet \mathbf{n}) dS = 0 \qquad [B.2.1]$$

Applying this to a single quadrilateral, with volume $V_{j,k}$ and zone average $Q_{j,k}$ yields

$$\left(\frac{\partial Q}{\partial t} V\right)_{j,k} + \left[\sum_{l=1}^{4} \oint_S (\mathbf{F} \bullet \mathbf{n}) dS\right]_{j,k} = 0 \qquad [B.2.2]$$

where the summation is over the four sides of the quadrilateral. Approximating the surface integrals by average values gives

$$\left[\frac{\partial Q}{\partial t} V\right]_{j,k} + [(\mathbf{F} \bullet \mathbf{n})\Delta S]_{j+\frac{1}{2},k} + [(\mathbf{F} \bullet \mathbf{n})\Delta S]_{j-\frac{1}{2},k} + [(\mathbf{F} \bullet \mathbf{n})\Delta S]_{j,k+\frac{1}{2}} + [(\mathbf{F} \bullet \mathbf{n})\Delta S]_{j,k-\frac{1}{2}} = 0 \qquad [B.2.3]$$

where $\Delta S_{j\pm\frac{1}{2},k}$ are the cell faces at $j \pm \frac{1}{2}$ which corresponds to ξ = constant cell faces in computational space. They are computed by

$$\Delta S_{j\pm\frac{1}{2},k} = \left[(\Delta x)^2_{j\pm\frac{1}{2},k} + (\Delta y)^2_{j\pm\frac{1}{2},k}\right]^{\frac{1}{2}}$$

which can be written in terms of the metric relations as (since $\Delta\xi = \Delta\eta = 1$)

$$\Delta S_{j\pm\frac{1}{2},k} = \left[x_\eta^2 + y_\eta^2\right]^{\frac{1}{2}}_{j\pm\frac{1}{2},k}$$

$$\Delta S_{j\pm\frac{1}{2},k} = \left[\frac{\xi_x^2 + \xi_y^2}{J^2}\right]^{\frac{1}{2}}_{j\pm\frac{1}{2},k}$$

$$\Delta S_{j\pm\frac{1}{2},k} = \left[\frac{|\nabla\xi|}{J}\right]_{j\pm\frac{1}{2},k}$$

Likewise, $\Delta S_{j,k\pm\frac{1}{2}}$ represents the lengths of η = constant cell faces and can be written as

$$\Delta S_{j,k\pm\frac{1}{2}} = \left[\frac{|\nabla\eta|}{J}\right]_{j,k\pm\frac{1}{2}}$$

as a result

$$\mathbf{F}\bullet\mathbf{n} = F\mathbf{n}_x + G\mathbf{n}_y \qquad [B.2.4]$$

and, using the metric relations,

$$(\mathbf{n}_x)_{j+\frac{1}{2},k} = \left[\frac{\xi_x}{|\nabla\xi|}\right]_{j+\frac{1}{2},k} \quad (\mathbf{n}_y)_{j+\frac{1}{2},k} = \left[\frac{\xi_y}{|\nabla\xi|}\right]_{j+\frac{1}{2},k}$$

and

$$(\mathbf{n}_x)_{j,k+\frac{1}{2}} = \left[\frac{\eta_x}{|\nabla\eta|}\right]_{j,k+\frac{1}{2}} \quad (\mathbf{n}_y)_{j,k+\frac{1}{2}} = \left[\frac{\eta_y}{|\nabla\eta|}\right]_{j,k+\frac{1}{2}}$$

Substitution of the above into equation [B.2.3] gives

$$\left[V\frac{\partial Q}{\partial t}\right]_{j,k} + \left[\frac{\xi_x F + \xi_y G}{|\nabla\xi|}\frac{|\nabla\xi|}{J}\right]_{j+\frac{1}{2},k} - \left[\frac{\xi_x F + \xi_y G}{|\nabla\xi|}\frac{|\nabla\xi|}{J}\right]_{j-\frac{1}{2},k}$$
$$+ \left[\frac{\eta_x F + \eta_y G}{|\nabla\eta|}\frac{|\nabla\eta|}{J}\right]_{j,k+\frac{1}{2}} - \left[\frac{\eta_x F + \eta_y G}{|\nabla\eta|}\frac{|\nabla\eta|}{J}\right]_{j,k-\frac{1}{2}} = 0 \qquad [B.2.5]$$

Noting that the cell volume is given by the inverse of the Jacobian, J, and making use of the definition of $\widetilde{F}$ and $\widetilde{G}$ from equation [B.1.10] gives for all j,k

$$\frac{1}{J}\frac{\partial Q}{\partial t} + \frac{\delta_j\widetilde{F}}{\delta\xi} + \frac{\delta_k\widetilde{G}}{\delta\eta} = 0 \qquad [B.2.6]$$

where δ is the centered difference operator, $\delta_j\widetilde{F} = \widetilde{F}_{j+\frac{1}{2}} - \widetilde{F}_{j-\frac{1}{2}}$ and, for convenience, $\delta\xi = \delta\eta = 1$. Thus, the connection between the numerical implementation of the integral (or finite-volume) form (equation [B.2.6]) and the differential form via a computational mapping (equation [B.1.9]) has been accomplished.

Appendix C. Treatment of the viscous vector

The difficulty in the viscous portion of the governing equations arises in the need to evaluate the individual terms in those vectors. The individual terms contain partial derivatives in x or y. They are calculated by the use of the chain rule from the physical to the computational space. From there the thin layer approximation is applied. The transformations with the thin layer approximation are

$$\frac{\partial}{\partial x} = \eta_x \frac{\partial}{\partial \eta} \qquad \frac{\partial}{\partial y} = \eta_y \frac{\partial}{\partial \eta} \qquad [C.1.1]$$

since viscous and heat flux terms containing ξ derivatives are neglected

Using these transformations results in the following relations for the shear stress and heat flux terms.

$$
\begin{aligned}
\tau_{xx} &= (\lambda + 2\mu)\,\eta_x \frac{\partial u}{\partial \eta} + \lambda \eta_y \frac{\partial v}{\partial \eta} \\
\tau_{xy} &= \mu\left(\eta_y \frac{\partial u}{\partial \eta} + \eta_x \frac{\partial v}{\partial \eta}\right. \\
\tau_{yy} &= (\lambda + 2\mu)\,\eta_y \frac{\partial v}{\partial \eta} + \lambda \eta_x \frac{\partial x}{\partial \eta} \\
q_x &= \frac{\mu}{(\gamma - 1)\mathrm{Pr}M_\infty^2}\,\eta_x \frac{\partial T}{\partial \eta} \\
q_y &= \frac{\mu}{(\gamma - 1)\mathrm{Pr}M_\infty^2}\,\eta_y \frac{\partial T}{\partial \eta}
\end{aligned}
\qquad [C.1.2]
$$

where $\lambda = -2/3\mu$ by Stoke's Hypothesis

The transformation requires the determination of μ, η_x, and η_y at the cell faces. μ is obtained by the averaging the cell centered values that surround the cell face. η_x and η_y are obtained by the metric relations given in appendix B. which are repeated here

$$
\eta_x = -Jy_\xi \qquad \eta_y = Jx_\xi \qquad [C.1.3]
$$

where J is the inverse of the cell volume and x_ξ and y_ξ are the components of the cell face length in the x and y directions, respectively. The area at the cell face is determined in the same fashion as μ. Now all of the viscous terms can be calculated. The resulting viscous thin layer vectors have the same appearance as the complete vectors.

Appendix D. Delta coefficients for the thin-layer Navier-Stokes equations

The 4 by 4 block matrices forming the coefficient matrix in Figure 2 on page 13 are given below.

$$h = -\alpha_y B^{+}_{k-\frac{1}{2}}$$

$$f = -\beta_x A^{+}_{j-\frac{1}{2}} + \alpha_x A^{+}_{j+\frac{1}{2}} - \gamma_x A^{-}_{j-\frac{1}{2}}$$

$$d = -\alpha_y B^{+}_{k-\frac{1}{2}}$$

$$a = -\beta_y B^{+}_{k-\frac{1}{2}} + \alpha_y B^{+}_{k+\frac{1}{2}} - \gamma_y B^{-}_{k-\frac{1}{2}} - \frac{1}{Re}\left[\mu_{k-\frac{1}{2}} H_{k-1}\right]$$

$$b = \beta_x \left(A^{+}_{j+\frac{1}{2}} - A^{-}_{j-\frac{1}{2}}\right) - \gamma_x A^{+}_{j-\frac{1}{2}} + \gamma_x A^{-}_{j+\frac{1}{2}} + \beta_y \left(B^{+}_{k+\frac{1}{2}} - B^{-}_{k-\frac{1}{2}}\right) - \gamma_y B^{+}_{k-\frac{1}{2}} + \gamma_y B^{-}_{k+\frac{1}{2}} - \frac{1}{Re}\left[-\left(\mu_{k+\frac{1}{2}} + \mu_{k-\frac{1}{2}}\right) H_k\right] + \frac{1}{J\Delta t}$$

$$c = \gamma_y B^+_{k+\frac{1}{2}} - \alpha_y B^-_{k-\frac{1}{2}} + \beta_y B^-_{k+\frac{1}{2}} - \frac{1}{Re}\left[\mu_{k+\frac{1}{2}} H_{k+1}\right]$$

$$e = \alpha_y B^-_{k+\frac{1}{2}}$$

$$g = \gamma_x A^+_{j+\frac{1}{2}} - \alpha_x A^-_{j-\frac{1}{2}} + \beta_x A^-_{j+\frac{1}{2}}$$

$$i = \alpha_x A^-_{j+\frac{1}{2}}$$

where $A^{\pm}$, $B^{\pm}$, and H are the Jacobians of $F^{\pm}$, $G^{\pm}$, and F_v, respectively. α, β, and γ are given by

$$\alpha_x = -\frac{\varphi}{4}(1 - \kappa_x)$$
$$\beta_x = 1 - \frac{\varphi}{2}\kappa_x$$
$$\gamma_x = \frac{\varphi}{4}(1 + \kappa_x)$$

And similarly for α_y, β_y, γ_y in terms of κ_y.

The two page vita has been removed from the scanned document. Page 1 of 2

The two page vita has been removed from the scanned document. Page 2 of 2