DEPARTMENT OF MATHEMATICS

EXPERIMENTS WITH 2D SHOCK FITTING FOR THE EULER EQUATIONS

Peter D. Arnold

Numerical Analysis Report 18/90

UNIVERSITY OF READING

UNIVERSITY OF READING

Experiments with 2D shock fitting for the Euler Equations

Peter D. Arnold

Numerical Analysis Report 17/91

University of Reading

Department of Mathematics

P O Box 220

Reading RG6 2AX

Report for RAE Farnborough on work carried out Oct.1989 - Sept 1990

The work reported here forms part of the research programme of the Institute for Computational Fluid Dynamics at Oxford and Reading, and was carried out with the support of R.A.E. Farnborough under contract number DER/1/9/4/2035/065//XR/AERO

INTRODUCTION

This report represents work carried out on R.A.E. contract number DER/1/9/4/2035/065//XR/AERO, since 1st November 1989.

The objective of the contract is to produce a code capable of solving the pseudo unsteady Euler equations (H system), by applying a cell vertex finite volume method to a transonic flow problem, initially flow in a bumpy channel. A code which was not completely satisfactory was already in existence, namely a version written by M. Paisley which was based from the Morton and Paisley report [1]. This code attempts to fit a line of discontinuity into the solution domain, using a grid line which is double valued, and shaped by a least squares cubic polynomial representing the shock. The discontinuity has flow variables specified on both upstream and downstream sides, and is moved according to the local shock speed. However the code is unsatisfactory in that the unknowns do not converge along the shock, and consequently the shock speeds do not converge or reduce to zero. The cause of the non convergence at the time the contact began was thought to be due to the manner in which the shock tip in particular was treated.

The bulk of this report contains the details of the most significant experiments tried in order to improve convergence. Chapter 1 deals with some preliminaries such as, defining the flow equations, boundary and initial conditions. Also it describes the solution procedure in the original method, and some modifications made to the original code to make it more robust or accurate. Chapter 2 describes a set of experiments done under the assumption that the shock was normal to the flow. Chapter 3 describes fewer experiments carried

out using a variable switching method, and Chapter 4 deals with all the experiments that use the actual positions of the shocked nodes to calculate the shock angles, including variations on the cubic polynomial curve fitting. Chapter 5 is the final experimental work and describes experiments which combine both shock capturing and fitting. Each experimental chapter has its own conclusions section which summarise the main results. Finally Chapter 6 draws conclusions from all previous chapters and puts forward possibilities for future work.

CHAPTER 1

Defining the Problem

Here follows a description of the equations to be solved, and the boundary conditions.

Since only the steady state solution is of interest we assume constant enthalpy, and consequently the system of equations to be solved is the H system, i.e.

$$\frac{\partial \underline{u}}{\partial t} = \frac{\partial \underline{F}}{\partial x} + \frac{\partial \underline{G}}{\partial y}$$

where $u = \begin{vmatrix} \rho \\ \rho u \\ \rho v \end{vmatrix}$ $F = \begin{vmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \end{vmatrix}$ $G = \begin{vmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \end{vmatrix}$

u and v are the velocity components in the x and y directions

ρ is density

p is pressure

After non-dimensionalising the variables in the usual way, the above equations remain unchanged while the pressure by Bernoulli's equation is

$$P = \frac{\rho}{\gamma}\left[1 - \frac{(\gamma-1)}{2}(u^2 + v^2)\right]$$

where γ is the ratio of specific heats = 1.4.

The shape of the domain and boundary conditions are as in Hall [2], but to remind the reader the calculation is performed on a circular arc bump with height-chord ratio 1:10. The channel width is 1 chord and the

channel extends one chord from the leading edge (LE) backwards and one chord from the trailing edge (TE) forwards (see Fig. 1).

The boundary conditions for the subsonic inflow and outflow are arrived at using the characteristic theory from simplified unidirectional flow, i.e. at inflow we have two ingoing characteristics and one outgoing characteristic. Hence two pieces of information are specified from outside the solution domain, which are chosen to be $v = 0$ (parallel flow) and $P = \left[\frac{\rho}{\gamma}\right]$ (the isentropy condition). Then ρ is specified by calculation from inside the domain, and u can then be solved for via Bernoulli's equation. At outflow we have two outgoing characteristics and one incoming characteristic. Here we choose to specify the pressure at outflow as

$$P = \left[1 + \frac{(\gamma-1)}{2} M_\infty^2\right]^{\gamma/1-\gamma} \qquad \text{where} \quad M_\infty = 0.675$$

and ρu, ρv are calculated from inside the domain: ρ is then calculated via a rearrangement of Bernoulli's equation.

On the top channel wall we prescribe a symmetry condition which also ensures that the flow is tangential, i.e. $v = 0$. On the bottom wall we enforce tangential flow again, via a symmetry principle.

The initial conditions are $v = 0$ and M_∞ everywhere except along the bump, where the tangential velocity is the tangential component of M_∞ , and the normal velocity is zero. The density is then calculated using the isentropy condition and Bernoulli's equation.

The resulting flow will contain a shock whose foot is positioned on the bump, due to the fluid being accelerated over the bump. The problem is to develop a procedure capable of fitting a discontinuity in the

correct place, which allows the solution in the entire domain to converge and the shock speeds to reduce to zero, all along the shock.

INITIAL SOLUTION PROCEDURE

The procedure used when the problem was first approached was that of a multigrid Euler solver, using a cell vertex method with Lax-Wendroff time-stepping as described in Hall [2], coupled with the shock-fitting procedure used by Morton and Paisley in [1]. Essentially this firstly captures the shock with the help of artificial viscosity, and then attempts to fit the shock along one grid line by use of a weighted least squares cubic polynomial. However this approach was found to be unsatisfactory because

a) the solution vector $\underline{u}$ did not converge sufficiently, particularly around the shock tip

b) problems encountered when reducing the mach number at outflow, namely negative arguments of square roots in

$$\frac{1}{\sqrt{\left\{1 - 0.5\left[1 - \frac{\rho_L}{\rho_R}\right]\right\}}}$$ used for calculation of upstream mach number on shock

and

$$\sqrt{\{1 - 0.5(\gamma-1)(u^2 + v^2)\}}$$

used in calculation of the sound speed and often occuring on the downstream side of the shock, both around the tip and foot particularly

c) i) sensitivity of shock position and orientation to weights used.

ii) Shock speeds continually indicate that the shock wishes to move in a particular direction, but the restraining effects of the cubic fit prevent it from doing so.

NUMERICAL EXPERIMENTS WITH SHOCK FITTING

Any shock fitting solution procedure is faced with the task of calculating

$$\begin{bmatrix} \rho \\ \rho u \\ \rho v \end{bmatrix}_L \quad \text{and} \quad \begin{bmatrix} \rho \\ \rho u \\ \rho v \end{bmatrix}_R$$

together with shock orientation angle α and shock speed s.

Hence there are eight unknowns which determine the shock completely. Four of the unknowns are calculated by the finite volume method, i.e $\begin{bmatrix} \rho & \rho u & \rho v \end{bmatrix}_L^T$ and ρ_R (in an attempt to obey characteristic theory) leaving $(\rho u)_R$, $(\rho v)_R$, α and S to solve for, and only 3 equations from the Rankine-Hugoniot jump conditions to solve with. Hence an extra piece of information is required.

The cubic fit procedure provides the additional piece of information in the form of the shock orientation angle α at each node. However because of the problems associated with this method, an alternative was sought.

The following is an outline of the main alternatives tried and the results obtained. However firstly it is necessary to describe some of the modifications which were made to the method as it stands in order to improve it, before any experiments were carried out.

MODIFICATION OF SHOCK MOVEMENT AND GRID REMESHING

As the code stands the shock is moved in the x direction only, according to $(\sin \alpha)\ S\ \Delta t$, i.e. the x component of the normal shock speed. Then a patch of the mesh, 3 cells either side of the grid line used to represent the shock, was adjusted so as to space these cells evenly. At the same time the nodes which were moved kept the same y values. This was found to lead to grid distortion and consequent instability in situations where the mesh movement was large.

The improvement was made by taking account of the y component of the shock movement, and using a remeshing procedure which was much less likely to cause grid distortions. The y component of the shock movement was calculated as $-(\cos \alpha)\ S\ \Delta T$, and the new features of the remeshing procedure are:

i) Equidistant spacing of grid nodes occur in each horizontal line, between the shock node and $x = 0$ to the left, and the shock node and $x = 1$ to the right, for all horizontal lines which have a shock node on them, whilst those without a shock node are spaced in the same way as the horizontal grid line which passes through the shock tip.

ii) Adjustment of the vertical grid lines so that they correspond to the new y value of each shock node whilst maintaining the same body fitted shape.

iii) Respacing of horizontal grid lines above the shock to even out any grid compression which may have occured above the shock in the vertical direction.

The result of implementing this procedure was to increase the convergence rate ofthe original cubic fitting method, and reduce the residual shock speed. The resultant grid was much more even, which is particularly important for the cell vertex method.

MODIFICATION OF UPDATING PROCEDURE AT SHOCKS

Instead of using the usual non-reflective conditions when updating on one side of the shock, the method was modified so that the areas of the cells whose rediduals were set to zero were the actual cell areas as opposed to the image cell areas, when calculating the first order change. Also the integration cell used to calculate the 2nd order change was that formed by the centroids of the four cells surrounding the point being updated. This was found by experience to be more successful in relation to the instability of the method, which usually shows itself as a code crash due to negative square root occuring, as mentioned previously. The reason for this is that if the grid becomes distorted around the shock, as in Fig. 2a, then an integration cell formed by a one-sided reflection procedure can easily become very unrealistic, which in turn makes the 2nd order update inaccurate, which leads to instability. By contrast the integration cell formed by the centroids of the four surrounding cells has a more robust area.

This was particularly important at the shock foot where previously only one cell and its centroid were used to determine the integration cell area (Fig. 3a). In this case we have the additional factor of a tangential boundary. In the modified method we use an integration cell constructed from the two cells either side of the shock, and the centroids of their image cells (Fig. 3b).

DELETION OF MULTIGRID ACCELERATION IN SHOCK FITTING PHASES

Once the shock had been captured sufficiently, the shock fitting phase was extended. It was decided to do this on the finest grid only, to avoid the added complexity of using multigrid, and so the multigrid part of the fitting code was left out.

CHAPTER 2

The first alternative solution procedure to the cubic fitting method was to assume that the shock was normal to the flow, hence there was no tangential component to the flow on either ustream or downstream side. This makes the calculation of the shock orientation angle to the x axis particularly simple, because once we have calculated u_L and v_L, the velocity components in the x and y directions on the upsteam side, then $\tan(\alpha - \frac{\pi}{2}) = \frac{v_L}{u_L}$, hence providing the additional piece of information to solve the Rankine-Hugoniot jump conditions for the remaining variables.

The assumption of normal flow along the entire shock can be justified, since at the shock foot the flow is normal due to a zero tangential component and, if we examine the many mach contour plots available of channels and aerofoils (see Figs. 4a and 4b), where a shock has been captured or fitted using a cubic, it is easy to observe the sonic line as being continuous with the shock tip. Since the normal component of the mach number of a stationary shock is equal to 1.0 at the tip, the shock tip has no tangential component, consequently it seems reasonable to assume that the flow is normal along the entire shock length.

EXPERIMENTS WITH THE NORMAL FLOW ASSUMPTION

The code was rewritten so that the angles of the shock nodes were calculated as above, and no cubic fitting was used. Also the smallest

time step of the entire solution domain was used to advance the calculation and the shock movement, in order to mimic a time accurate solution closer. This resulted in a solution which converged slowly though not monotonically, with respect to the shock speeds of the shocked nodes. The results of a 10,000 iteration run are recorded in table 1, where the maximum shock speed and x position of the shock foot and tip were noted every 1,000 iterations. The resulting grid is shown in Fig. 5.

These results contrast with those of the captured solution in that the shock front is approximately 0.1 further downstream, and the tip is approximately 0.03 further upstream in the fitted solution. Also it is clear that not all of the shock is being represented as $\rho_L \neq \rho_R$ at the tip.

EXTRAPOLATING THE TIP

For the latter reason a procedure was developed which could elongate or shorten the shock periodically in the direction the tip was already pointing by an amount proportional to the magnitude of the difference between the normal mach number (MNL) and 1.0 at the shock tip. This would hopefully fit a shock in the area where a shock was neither being fitted or captured (since no artificial viscosity was present), and consequently reduce any oscillations produced in this area. In the original (Paisley) method used, an extra grid point is added if the captured shock at the grid point immediately above and upstream of the tip contains a captured shock of sufficient strength. In contrast this new procedure moved the grid point corresponding to the tip in the appropriate direction. Also the grid line index number

corresponding to the shock was increased or decreased if the shock foot moved past the x value corresponding to the original position of the adjacent grid line, and the shock values were transferred onto this grid line.

Various periods at which the shock was elongated or shortened, and different values of the constant of proportionality by which the shock length was altered were tried, but none resulted in convergence of the shock speeds, and more often the program crashed due to a negative argument of a square root in the sound speed calculation.

It was thought that the reason for this instability might have been the cells at the tip becoming too elongated, and so instead of moving the tip grid point upwards an extra grid point was brought down to represent it and the unknowns at this point were given common upstream and downstream values, equal to the original values of the point. If however the shock were to need shortening then the node would simply move downwards as before. The result of doing this was that too many grid points were used resulting in a long shock which moved too far upstream, whilst the foot moved too far downstream causing the grid line which represented the shock to be continually replaced by the grid line upstream of it. The net result was a diverging solution which eventually caused the code to crash.

After examining the grids produced in the last two cases, it became clear that altering the vertical grid line which represents the shock, if the shock foot has moved sufficiently, is not necessarily helpful. If we consider what will happen should the foot wish to move forward but the tip backwards, we see that although the cells at the foot may be evenly spaced, those at the tip will become unevenly spaced (see Fig. 6).

In view of the above the code was run without any extrapolation or contraction of any kind, and the grid line representing the shock was kept the same, with a constant number of shocked nodes. The result after 20,000 iterations is shown in table 2, where the horizontal lines labelled 1 to 7 correspond to the upstream and downstream values of the shock nodes, 1 being the foot. The vertical columns are respectively

ρ, ρu, ρv, α, S, MVL, x position

The convergence of the shock speeds was not monotonic, and the maximum residual $\simeq \frac{\partial u}{\partial t} = 0.1163$ occurring at the leading edge (L.E). The resulting grid is shown in Fig. 7.

TESTING STABILITY OF SOLUTION TO SHOCK LENGTH

Next, in order to test how sensitive the solution is to representing the shock in its entirety, the same code was run, except the node corresponding to the tip was removed after 2,000 iterations, and the code ran for a further 18,000 iterations. The results are shown in Table 3.

The maximum residual was 0.1157, occurring just upstream of the shock tip. Note the maximum shock speed is nearly 30 times larger than when seven shock nodes were used, and again the convergence was unsteady.

SHOCK SHARPENING VARIATIONS

Several experiments were also carried out with the initialisation of the shock values, immediately after capturing the shock, including

putting upstream values equal to the downstream values, and their common value equal to the average of the extrapolated values either side of the shock, or the interpolated value in the shocked cell depending on its detected position - no significant improvement on the shock speeds was recorded.

DISCUSSION

The experiments performed so far indicate that

i) uneveness of the grid can easily cause instability and should be avoided

ii) the solution is very sensive to how closely the length of the shock is represented

iii) examination of the results after 20,000 iterations indicate that the positions of the shock nodes, and the angles assciated with them, are inconsistent with a continuous curve without kinks in it. (The angles with the x axis are all less than $\frac{\pi}{2}$, yet the nodes have successively smaller x values).

ADDITION OF ARTICIFICAL VISCOSITY IN TIP REGION

In order to eliminate the sensitivity of the solutions to the shock length, and consequent grid uneveness, it was decided to attempt to smear out the shock in the tip region with artifical viscosity, whilst representing the reminder of the shock using a normal flow assumption as before. (Previously when shock fitting, artificial viscosity was switched off everywhere in the solution domain). The viscosity added was of the form

$$\mu_1[|R_A(\rho)| + |R_B(\rho)| + |R_C(\rho)| + |R_D(\rho)|][U_A + U_B + U_C + U_D - 4U_1]$$

when being applied to node 1 (see Fig. 7a), where

$R_A(\rho)$ ... $R_D(\rho)$ are density residuals in cells A ...D

U_A .....U_D are the corresponding components of vector $\underline{u}$ in cells A ...D .

$U_c = \frac{1}{4}[U_1 + U_2 + U_3 + U_4]$

μ_1 is usually taken to equal 0.02,

The experiments conducted applied the viscosity to the nodes marked x in figures 8 to 10. If these points were double-valued shock nodes, then they are now treated as single valued nodes.

The result of applying viscosity as in Fig. 8 was to give a shock foot position = 0.828, a maximum shock speed = 0.04 and a RMS residual = 0.011 after 1,000 iterations. Hence the foot was further downstream than in the solution without the additional viscosity. The configuration used in Fig. 9 produced a shock foot = 0.840, a maximum shock speed = 0.02 and a RMS residual = 0.0467. This was virtually unchanged when varying the constant μ_1 from 0.01 to 0.05. Finally the configuration used in Fig. 10, with μ_1 = 0.10, gave nearly identical results to Fig. 9. None of the configurations used improved the convergence of the shock speeds at the remaining shock nodes.

EXPERIMENTS WITH SHOCK MOVEMENT

A different approach to the problem of getting the solution to converge was to alter the manner in which the shock nodes were allowed to move, in order to find the steady state position of the shock.

A function was defined which measured the state of convergence of

the shock speeds, which was

$$F = \sum_{J=1}^{JMS} |S_J|$$

where S_J is the shock speed of the node J , and JMS is the number of the tip node. It was soon realised that S_J (J = 1 to JMS) is ultimately dependent upon the positions of the other shock nodes, and the unknowns at the other nodes, i.e. a change in the position of just one node will after a few iterations change the value of S at all other nodes. With this in mind it was thought that minimising F for each node may lead to the steady state position of the shock. So the value of F was assessed every 10 iterations, whilst allowing only one of the shock nodes to move, starting with the foot, until F started to increase, then this node was "locked up" and the same procedure applied to the next node up, and so on until every shock node had been treated in this way. Then the process was repeated starting from the foot again.

After 1,000 iterations the maximum shock speed was not significantly different from that without this procedure, though the node positions did not have as much time to converge, since each node was moved JMS times less on average. The position of the nodes seemed to be similar to those of the captured solution and the root mean squared residual (RMS residual) = 0.0066.

A different tactic tried was to simply move the nodes one at a time, every 10 iterations, regardless of the value of F , so that there was no restriction on which direction the node moved in. This resulted in a maximum shock speed slightly higher than would have been without

this procedure, and a RMS residual = 0.005 after 1,000 iterations.

In both of the above cases the convergence appeared to be very slow, and the angles on the shock were not consistent with the position of the shock nodes.

OTHER SHOCK MOVEMENT EXPERIMENTS

These included the use of the global minimum time step to advance the solution, in order to make the shock shape mimic the physical solution more closely, and the use of different periods at which the shock was moved, i.e the number of iterations between shock movement. However none of these produced any significant benefit in terms of shock speed convergence.

SHOCK TRACKING METHOD

The theory behind this scheme can be found in Samuels [3]. An outline of the method and its numerical implementation is as follows.

Suppose we have a shock at some point on which component i of the vector of unknowns $\underline{u} = [\rho, \rho u, \rho v]^T$ has a value U_o^i, (at time t). Our aim is to track this value of U_o^i onto the shock curve at $t + \Delta t$.

To accomplish this, the point on the shock requires a normal velocity component S and a tangential velocity component α_T^i. The normal component S can be calculated as usual from the jump conditions, whilst the tangential component is

$$\alpha_T^i = \frac{V^i - S\underline{v}.\underline{N}^i}{\underline{\tau}.\underline{N}^i}$$

where

$$V^i = \frac{\left[\frac{\partial F^i}{\partial x} + \frac{\partial G^i}{\partial y}\right]}{|\nabla u^i|}$$

$\underline{v}$ is a unit vector normal to the shock

$\underline{\tau}$ is a unit vector tangential to the shock

$\underline{N}^i$ is a unit vector in the direction of ∇u^i

$\frac{\partial F^i}{\partial x} + \frac{\partial G^i}{\partial y}$ is the ith component of the residual which we calculate in our cell vertex method.

∇u^i can be estimated at a point on the shock via a bilinear approximation, $u^i = ax + by + cxy + d$

$$\nabla u^i = \frac{\partial u^i}{\partial x} + \frac{\partial u^i}{\partial y} \quad \text{and} \quad \frac{\partial u}{\partial x} = a + cy, \; \frac{\partial u^i}{\partial y} = b + cx \; .$$

The coefficients a,b,c and d are found by applying the bilinear approximation at four local points (see Fig 11) including the point in question. In practice this is done twice using points 1,2,3 and 4 and then points 3,4,5 and 6 and taking the average value of $\frac{\partial u}{\partial x}$ and $\frac{\partial u}{\partial y}$ at point 3.

Once S and α_T^i have been found we resolve the velocities into x and y directions to give:

$$x^{n+1} = x^n + \Delta t \; S[\sin \alpha] + \Delta t \; \alpha_T^i \cos \alpha$$

$$y^{n+1} = y^n - \Delta t \; S[\cos \alpha] + \Delta t \; \alpha_T^i \sin \alpha$$

where α is the angle the shock makes with the x axis.

The shock movement routine was rewritten to incorporate the tangential velocity component. The first time it was tried, instability resulted because some of the horizontal grid lines were running into each other, or getting very close together, causing grid uneveness, which eventually let to the code crashing. However the sizes of the normal and tangential components of the shock velocity were comparable to each other.

Once the frequency at which the shock was moved was decreased, and the tangential component at the foot was left out (since it was calculated using only one cell as opposed to two) the code ran without crashing, (for the 2,000 iterations it was run for) but convergence was slow and the RMS residual = 0.306 which was comparatively large. Also the shock foot continually moved further downstream. In conclusion this method did not aid the convergence of the shock speeds, and did not behave robustly, due to the grid uneveness caused by horizontal lines becoming compressed, although the latter could possibly be overcome by introducing safety measures into the remaining routine.

CONCLUSIONS FOR NORMAL SHOCK ASSUMPTION

The work carried out so far under the assumption that the flow is normal to the shock has provided useful information about the behaviour of the solution.

Firstly the absence of either artificial viscosity, or a double-valued shock line in a region where a shock exists, will cause oscillations in the update vector, which leads to instability and eventually the code crashes, usually due to a negative square root. Consequently the solution is very sensitive to the way in which the

shock is represented, the tip being the most difficult part to represent, because it is here that the shock vanishes. The technique used to fit the tip by extrapolating the upstream normal mach number (MNL), was on reflection rather arbitrary, and could have been improved by fitting a function which takes account of MNL and the distance along the shock from the foot at several points, as opposed to one. Alternatively interpolation in the captured region of the shock could be helpful to detect where MNL = 1.0 , and then adding an extra grid point, or move the last grid point.

The use of artificial viscosity at the tip, in order to smooth out the discontinuity, produced no significant improvement in the reduction of the shock speeds.

The attempts made to manoeuvre the shock into the correct position by various means, including allowing only one node at a time to move in a direction which minimises the overall shock speed, were not successful in reducing the shock speeds sufficiently. In fact it appeared that the function F used to minimise the shock speeds had many local minima, in which the minimisation process got stuck. The inclusion of a tangential shock velocity in order to track the shock, did not improve the convergence either. Further work is required on this method to ensure that grid distortion does not occur.

All the experiments tried failed to reduce the shock speeds of the shocked nodes to zero, whilst producing a plausible shock shape. However they all had one common result, which was that the shock angles and the position of the nodes were consistent with a curve whose curvature alternated between positive and negative (see tables 2 and 3). In fact all the shock angles were less than $\pi/2$, yet the nodes had

positions with progressively smaller x values.

The method used treats each point individually, and there is no requirement for the points and shock angles to be consistent. The results in table 2 indicate that it is possible for the shock speeds to converge, to nearly zero, but the resulting shock shape is unrealistic.

What seems to be required is a curve that is normal to the direction of flow, and has a curvature which does not change sign. There are several possibilities for such a curve, for example a least squares cubic, whose coefficients are calculated by ensuring $\frac{dy}{dx} = -\frac{u}{v}$, or a cubic spline under the same restraint.

CHAPTER 3

OBTAINING ADDITIONAL INFORMATION FROM THE PREVIOUS TIME LEVEL

This technique provides the additional piece of information required by using a variable value from the previous time level.

Assume at some time that we have all the information to specify the shock completely, provided by say using a least squares cubic curve. At the next time level we have the unknowns u_R, v_R, S and α. The approach is to specify S from the previous time level, and then solve the 3 jump equations for u_R, v_R, and α. Then at the next time level we specify α from the previous level and solve for u_R, v_R, and S. This switching is continued whilst all the time ρ_L, u_L, v_L, S and ρ_R are being updated as usual, hence providing new information to close the system until (hopefully) convergence is reached when the shock is in the correct position.

One drawback with this approach is when solving for α a quadratic is formed. The two roots of the quadratic correspond to two angles. If the shock is close to being normal to the flow, the roots are very close together and which root to choose is not clear.

The equation which presents itself is

$$\cos \alpha = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

where

$$a = u_L^2 + v_L^2$$

$$b = 2v_L[S+a_L M_L]$$

$$c = [S+a_L M_L]^2 - u_L^2$$

where u_L, v_L and a_L are the upstream components of velocity and sound speed, M_L is the upstream relative normal mach no., equal to $\frac{q_{NL} - S}{a_L}$, and q_{NL} is the upstream normal velocity component equal to $u_L \sin\alpha - v_L \cos\alpha$, and S is the shock speed.

If all variables are at the same time level then

$$b^2 - 4ac = 4u_L^2[u_L^2+v_L - q_{NL}^2] = 4u_L^2[u_L\cos\alpha+v_L \sin\alpha]^2 \geq 0 .$$

However because the shock speed is taken from the previous time level (S_o),

$$b^2 - 4ac = 4u_L^2[u_L^2+v_L - (S_o+a_L M_L)^2]$$

and so there is a possibility that $b^2 - 4ac < 0$.

The method is similar to the normal flow assumption, in that there is no guarantee that the angles at the shock nodes will be consistent with the node positions. We assume that the method will manoeuvre them into the correct positions and the angles will be consistent at convergence.

EXPERIMENTS WITH VARIABLE SWITCHING

When the code was run with this variable switching included, the shock tip moved progressively further back as far as it could go, producing a very distorted grid, and the shock angles were not consistent with the shock shape. At this stage the angle chosen in the angle calculation was taken to be the smallest of the two. So the next time the code was ran the angle chosen was that which was closest to the

angle at the previous time level. This results in the same grid distortion. Then the largest of the two angles was chosen, but this resulted in the argument of $\sqrt{b^2 - 4a_L}$ becoming negative. To overcome this the argument was put equal to zero and the calculation continued. After 1,000 iterations this gave a max shock speed of 0.03. A shock foot position of 0.814 and a tip position of 0.561. However if the iteration continues a sawtooth shape begins to form, and after 2,000 iterations the shock speeds were no smaller than at 1,000, i.e. the method was not converging.

MINIMISING THE SHOCK SPEEDS

The minimisation procedure, as used for the normal shock assumption described on page 15, was applied to this variable switching method, to minimize the function F . After 1,000 iterations the maximum shock speed was - 0.03, and the RMS residual was 0.009, the shock shape being in reasonable agreement with the captured solution, but the angles were inconsistent with the shape.

If only one node was moved at a time in order along the shock, giving the solution time to settle before moving the next node, then after 1,000 iterations the shock speed = 0.07, the RMS residual = 0.0077, and the shock shape was in reasonable agreement with the captured solution, but again the shock angles were inconsistent with the node positions.

CONCLUSIONS FOR VARIABLE SWITCHING

The method of variable switching did not aid convergence of the shock speeds, and either resulted in a shock which was sawtooth, or one

which was inconsistent with the shock node angles. However on reflection the method can be critisized. Firstly it should not have been used at the shock foot, since here we can obtain α from the flow tangency condition, and then solve for u_R, v_R and S via the 3 jump equations. Hence the specification of α from flow tangency closes the system. By also using S from the previous time level, the system is overspecified. This may have contributed to the negative argument of $\sqrt{b^2 - 4ac}$ which occurred frequently. Secondly, as well as setting $b^2 - 4ac$ equal to zero when it was less than zero, the shock speed should have also been updated in accordance with $b^2 - 4ac = 0$. This is consistent with the flow being normal to the shock at this point. Thirdly, the problem of deciding which of two angles to use when solving a quadratic equation for α was not answered.

However, even if all these points were resolved, there is still doubt that the method would converge, because like the normal flow assumption there is nothing to make the shock angles consistent with the shock shape.

CHAPTER 4

USING THE SHOCK SHAPE TO DETERMINE THE SHOCK ANGLES

From our previous experience with using shock nodes, at which the angles do not necessarily correspond to the position of the nodes, the shocks speeds do not converge, and the nodes do not generally manoeuvre themselves into positions which correspond to the angles. A method of estimating the angles was therefore derived which used the actual positions. The first method calculates

$$\alpha(J) = \frac{Y(IXS,J) - Y(IXS,J-1)}{X(IXS,J) - X(IXS,J-1)}$$

where IXS is the discrete vertical grid line number which represents the shock. Hence this method uses the gradient of the grid line immediately below the node to estimate the angle of the shock at this point (see Fig. 12). The angle at the shock foot is calculated using flow tangency. The code was ran for 1,000 iterations, moving the shock every 10 iterations. This resulted in a shock shape which was sawtooth, gradually getting worse as the iterations continued (see Table 4 - note the large residuals). The resulting grid is shown in Fig. 15.

To counteract this sawtooth pattern, α was calculated by averaging the gradients both above and below the node (see Fig. 13), except at the foot, where the shock was taken to be normal to the flow, and the tip, where the previous one sided estimated had to be used. The code was ran for 1,000 iterations, the result again being a sawtooth shaped shock but not as severe as before (see table 5). Also the angles had virtually converged after 1,000 iterations. Examination of Table 5

shows that the shock speed was alternately positive and negative, and the angles increasing then decreasing.

MINIMISING THE SHOCK SPEED

The minimisation of the shock speeds via function F , as described on page 15, was tested with the angle/position method just described. After 1,000 iterations a mild sawtooth shaped shock was produced. The maximum residual and RMS residual were larger than when this process was applied to a normal flow assumption, and the shock speeds were generally no smaller than without using the minimisation process.

If instead of minimising F , each node was moved sequentially regardless of the size of F , then the maximum shock speed, residual and RMS residual were slightly smaller than when minimising F . However the improvement was not significant compared to the results achieved without the minimisation process.

APPLICATION OF SHOCK TRACKING METHOD

The shock tracking method as described on page 17 was used in conjunction with this angle/position method. The result was a code crash, which was caused by bunching of horizontal grid lines. The angle position method was then changed by weighting the two gradients used, depending on the distance between the two adjacent nodes (see Fig 14). This resulted in a code crash also, caused by bunching of horizontal lines (see Fig. 16). The tangential component of the shock speed at the foot was then put equal to zero, because it was thought that it may be in error, due to the calculation of grids being one sided at this point. Also the frequency at which the shock was moved was decreased, and the

minimum global time step was used to advance the whole calculation. The sawtooth pattern still emerged and the code eventually crashed after 3,500 iterations, due to bunching of the horizontal grid lines.

FITTING A LEAST SQUARES CUBIC POLYNOMIAL

Having not found a method capable of fitting the shock so that the shock shape is plausible and the shock speeds and variables converge, we return to out starting point. This was the use of a least squares cubic polynomial to respresent the shock and hence provide the orientation angle. The best results obtained so far using this method can be seen in Table 6, where the method was ran for 5,000 iterations. In an effort to improve these results we firstly tried the inclusion of the shock tracking method as previously decribed. An evenly weighting cubic and a weighting which is biased towards the shock foot were both tried, together with the use of the smallest global time step, and a reduction in shock movement frequency. However the result was always that the horizontal grid lines would sooner or later bunch up as in Fig. 16, and the residuals would become large around the leading edge. Another idea was to impose continuity of all flow variables at the shock tip, without the shock tracking method. This resulted in the updates and residuals becoming large around the tip area.

CONCLUSION FOR THE USE OF SHOCK SHAPE TO DETERMINE ANGLES

The estimation of the shock angle at a shock node, by a weighted combination of the gradients between the node and the adjacent shock nodes, appears to have the peculiar effect of producing a sawtooth shaped shock. The shock speed minimisation procedure did not stop this

sawtooth shape emerging, or reduce the shock speeds. The shock tracking method generally produced a distorted grid and a sawtooth shock.

The use of a least squares cubic to represent the shock produces a plausible shock shape, but leaves residual shock speeds, which do not converge themselves, as do not the unknowns along the shock, the maximum shock speed being some 50 times larger than that achieved using a normal shock assumption for 20,000 iterations. In effect we have a boundary which is moving about very slightly, which perturbs the unknowns along it, which in turn moves the boundary, but equilibrium is not found. The imposition of tip continuity, and the inclusion of a tangential shock speed, only hinder the convergence.

CHAPTER 5

EXPERIMENTS WITH SHOCK CAPTURING AND MESH ALIGNMENT

In order to investigate the effect of artificial viscosity on the shock captured solution, the initial part of the code, which captures the shock ready for fitting, was ran on a 17X65 grid and a 33X129 grid with multigrid for 1,000 iterations using various values of the aritificial viscosity parameter μ_1 . The resulting detected shock positions are recorded in Tables 7 and 8.

The 17 x 65 grid results show that the positions for $\mu_1 = 0.01$ to 0.04 are unchanged to 2 decimal places except at the tip for $\mu_1 = 0.04$. For $\mu_1 = 0.005$ however the positions are considerably changed, in particular the shock foot has moved further upstream. If we compare these results with the 33X129 grid, we see that for $\mu_1 \leq 0.02$ the shock foot is again considerably further upstream and the tip also to a lesser extent. The illustrates the arbitrariness in the use of artificial viscosity.

ALIGNING THE MESH

It is shown in Morton and Paisley [1] that the error caused by evaluating the line integral $\int F\,dy - G\,dx$ by the trapezoidal rule, in a cell which is intersected by a shock, can be minimised by positioning the mesh so that the shock exactly bisects the upper and lower sides (see Fig. 17).

A procedure was therefore developed which could capture the shock on the initial grid, detect the shock position, and realign the mesh so that the shock bisects all shocked cells. Then this process was repeated so that eventually the shock would always be detected in the

same position (bisecting a column of cells) and convergence could (hopefully) be achieved.

Theoretically, for a shock whose flux vectors change at an equal rate with distance or are constant for half a cell either side of the shock (see Fig. 18), it should be possible to set all residuals to zero. As a consequence of this the component of the artificial viscosity, which is proportional to the density residual, will equal zero.

This procedure was tested using 1,000 iterations on the finest grid for each mesh, before realigning the mesh to the new detected shock position. The results are shown in Table 9. After 1,000 iterations in the 10th grid position no shock was detected. It is clear from Table 9, that the process was getting out of control, and that the mesh was very distorted by this point.

OTHER OBSERVATIONS ON SHOCK CAPTURING

It was noticed that the converged solution when using multigrid is not identical to the converged solution when iterating on the fine grid only. The can be shown by obtaining a converged solution using multigrid, and then switching to the fine grid. At this point the RMS $\left[\frac{\text{UPDATE}}{\text{DT}}\right]$ value rises and then decreases again as the solution converges on the fine grid. Also the RMS residual obtained using multigrid equals 0.0281, as opposed to 0.0302 on the fine grid only. The largest residuals in both cases occur around the leading edge, and around the shock.

The reason this occurs is thought to be that the multigrid solution does not add any artificial viscosity to the solution, either as a background smoother, or around the shock, because the multigrid process

is a smoother in itself. The solution on the fine grid however adds both types of viscosity, and hence converges to a slightly different solution.

CONCLUSION ON SHOCK CAPTURING AND MESH ALIGNMENT

In this section we have experimented with the use of artificial viscosity to capture shocks. The results without any mesh adjustment, show that the choice of the artificial viscosity parameter μ_1 is rather arbitrary, and is often chosen to give the "best" results in terms of convergence properties and shock position. Next we tried to adjust the mesh so that the shock passed through the middle of the upper and lower faces, of the shocked cells. This if successful would have minimized the residuals and consequently the artifial viscosity present. However in practice the process of repeatedly adjusting the mesh got out of control. Some of the reasons why this may have occured are as follows

i) The shock detection algorithm is not accurate enough at detecting where the shock is in a cell.

ii) The type of viscosity used is not conservatively formulated, and will not therefore necessarily produce a shock in the correct position.

CHAPTER 6

CONCLUSIONS AND DISCUSSION

The object of the work carried out on this report was to produce a method capable of achieving convergence of the unknowns in the entire solution domain and reduction of the shock speeds to zero, whilst representing the shock by a plausible shape. If this had been achieved then the residuals (which are a measure of how well the differential equation is satisfied in each cell) would have been minimised.

The techniques included minimising a shock speed function via restricting the shock node movements, shock tracking by including a tangential component ofthe shock speed, minimum global time-stepping, tip extrapolation, and the use of artificial viscosity in the tip region to smear out the disontinuity.

The above techniques were combined with various assumptions which were used to provide a shock orientation angle, in order to solve for the unknowns along the shock length, the first assumption being that the shock is oriented normally to the direction of flow. This was successful in that convergence of the shock speeds and unknowns was eventually achieved (see Tables 1 and 2), but the resulting shock shape was inconsistent with the shock angles, unless its curvature alternated from positive to negative, which is not plausible.

The second assumption obtained a missing piece of information at the new time level from the previous time level. This did not help convergence of either unknowns or shock speeds, and the resulting shocks were sawtooth shaped, or else the angles were inconsistent with the positions of the nodes.

Thirdly the actual node positions were used to calculate the angles using either unweighted or weighted central differencing of the adjoining gradients. This resulted in a sawtooth shaped shock, along which the shock speeds and unknowns would not converge. Also an unweighted or weighted least squares cubic polynomial was used to fit the node positions, and provide the shock angles. This was fairly successful (see Table 6), but a small residual shock speed was always present which would not converge and neither did the unknowns along the shock.

In the final set of experiments a hybrid of shock capturing and fitting was used, to align the mesh so that the shock exactly bisected all shocked cells, in an effort to minimise the error caused by averaging across the shock, using the trapezoidal rule. The method was unsuccessful in its present state, and improvements are required to the shock detection algorithm, and a conservatively formulated form of artificial viscosity needs to be used, which together should predict the shock location more accurately.

A feature which both the normal flow assumption and the variable switching method have in common is that neither provide any means of linking the shock angles calculated at the shocked nodes with their positions. This seems to be an undesirable feature, and in all cases resulted in a shock shape which was inconsistent with the node angles.

The use of central differencing locally to obtain the shock angles also appears to be undesirable because it does not take into account the remaining shock nodes. The only remaining shock fitting method then is to use a curve of some kind to fit the shock. The curve used here was a least squares cubic polynomial. Let us now look in detail at the way

this was implemented.

The shock is treated as an internal boundary, at which boundary conditions are implemented by using characteristic theory as a guide. Since we can rotate our axes and equations to align with the direction of flow at any point in the flow field, we use the characteristic theory from the simplified undirectional flow unsteady Euler equations.

The characteristic speeds λ are q, $\left[\frac{\gamma+1}{2\gamma}\right] q \pm \sqrt{\frac{a^2}{\gamma} + \left[\frac{\gamma-1}{2\gamma}\right]^2 q^2}$ where q is the flow speed (see [1]), and a is the sound speed. Hence if $q^2 = a^2$, $\lambda = q$, $\frac{\gamma+1}{2\gamma}(q \pm a)$ so $\lambda = q$, $\frac{\gamma+1}{\gamma} q$, 0 . If $a^2 < q^2$ as in supersonic flow then $\lambda = q$, positive , positive (since in this case $q > 0$) , and if $a^2 > q^2$ as in subsonic flow $\lambda = q$, positive , negative. Hence if $a^2 < q^2$ then all three characteristics head in the same direction as q , but if $a^2 > q^2$ then two characteristics head in the direction of q , whilst one heads in the opposite direction.

The important factor, as far as imposing boundary conditions on the shock is concerned, is: does the component of the characteristic speed, which is normal to the shock, move into or away from the shock. For this we need to know the normal component of both the shock and flow velocities. Therefore our relative normal characteristic speeds become

$$\lambda = (q_N - S) \ , \ \left[\frac{\gamma+1}{2\gamma}\right] (q_N - S) \pm \sqrt{\frac{a^2}{\gamma} + \left[\frac{\gamma-1}{2\gamma}\right]^2 (q_N - S)^2}$$

where q_N and S are the normal components of the flow and shock velocities.

So if $a < q_N - S \Longrightarrow \quad a^2 < (q_N - S)^2$ since $a > 0$ and $q_N >> S$

in this situation. Therefore $\lambda = (q_N - s)$, positive , positive: also $a > q_N - s \implies a^2 > (q_N - s)^2$, therefore $\lambda = (q_N - s)$, positive , negative, where λ is now the relative normal characteristic speed.

Now it can be easily shown that the relative normal upstream and downstream mach numbers MNL, MNR are given by

$$MNL^2 = \frac{(\gamma+1)}{2\gamma} \frac{P_R}{P_L} + \frac{(\gamma+1)}{2\gamma}$$

$$MNR^2 = \frac{(\gamma+1)}{2\gamma} \frac{P_L}{P_R} + \frac{(\gamma+1)}{2\gamma}$$

where MNL is defined equal to $\frac{q_{NL} - S}{a_L}$ and P_L, P_R are upstream and downstream pressures. Now since the shock is compressive $P_R \geq P_L$

$$\implies M_{NL}^2 \geq 1 \quad \text{and} \quad M_{NR}^2 \leq 1$$

and since $q_{NL} - S > 0$ and $q_{NR} - s > 0$ and $a_{L,R} > 0$

$$\implies M_{NL} \geq 1 \quad \text{and} \quad M_{NR} \leq 1$$

$$\implies q_{NL} - s > a_L \quad \text{and} \quad q_{NR} - s < aR \ .$$

The implication of this argument is that all characteristics from upstream carry information from upstream into the shock, whilst on the downstream side, two characteristics carry information away from the shock into the downstream region, whilst one characteristic carries information from downstream back into the shock.

The interpretation of the characteristic theory in the present method is to update all three unknowns on the upstream side of the shock, using the numerical method in the upstream solution domain, and to update only one unknown, chosen to be the density, on the right hand side of the shock, using our numerical method in the downstream solution domain. The remaining two unknowns and the shock speed, are then calculated via the three Rankine Hugoniot jump conditions. However this is not strictly the correct way to proceed.

Consider what happens to the third relative normal characteristic speed as the shock tip is approached. On the upstream side the speed tends to zero from above, and on the downstream side the speed tends to zero from below. Intuitively this would appear to decrease the influence which the waves carried by this characteristic have on the shock. In fact the situation is even more complex. The characteristic speeds are the eigenvalues of the matrix A, when the Euler equations $\underline{U}_t + \underline{F}_x + \underline{G}_y = 0$ are written in the form $\underline{U}_t + A\underline{U}_x + B\underline{U}_y = 0$. Here we ignore the term $B\underline{U}_y$ since we assume unidirectional flow. Each of these eigenvalues has an eigenvector associated with it, and this is the quantity that is travelling with characteristic speed λ. The extent to which this eigenvector influences the shock is dependent not only upon the characteristic speed, but also on the strength of the eigenvector. This strength can be assessed explicitly by measuring the jump in the unknowns at the previous time level (see Roe [4]).

Another point which should be mentioned is that the flow on the downstream side of the shock at the tip, and for a finite region along the shock, has the possibility of being supersonic. If the flow is normal to the shock and for simplicity if we assume $S = 0$, then

$M_{NR} = 1.0$ at the tip $\Rightarrow$ $q_{NR} = a_R$. However if there is a tangential component to the flow then $q_{TR} \neq 0 \Rightarrow q_{NR}^2 + q_{TR}^2 > a_R$ implying supersonic flow. In this case the characteristic speeds in the direction of flow on the downstream side will be q, positive , negative (though the relative normal component of the 3rd characteristic will always be ≤ 0).

If we consider a moving shock then the flow upstream may be subsonic (a shock moving upstream into stationary air) and the flow downstream may be supersonic (at the shock tip where $q_{NR} - S = aR$, for $S > 0$) . The accompanying overturning of the characteristic speeds in the direction of flow, is most likely to occur in the tip region where M_{NL} and M_{NR} are closest to unity. However the relative normal inflow will always be supersonic, and the relative normal outflow will always be subsonic, and consequently the relative normal characteristic speeds will be as described previously.

All of the shock fitting techniques and the hybrid method rely on adapting the mesh to align a grid line either exactly onto the shock, or slightly offset from it. Inevitably the mesh becomes distorted, the extent of which depends upon how curved the shock is, and the number of shocks present. The accuracy of the cell vertex method is dependent upon the smoothness of the mesh, and consequently the accuracy of this shock fitting procedure is limited by the type of shocks present.

FUTURE WORK

There exist three main options for future work. Two have been selected for development from the previous work, and the remaining option relies on a new approach.

The first of the existing methods to be improved is the hybrid approach, which both captures the shock whilst positioning the mesh so that the shock exactly bisects the shocked cells, in an effort to minimise the error due to averaging by the trapezoidal rule. This is described on page 30, and suggested improvements are set out on page 32.

The second of the existing methods is to adapt the present curve fitting techniques so that they obey the characteristic theory, as described in the conclusions on page 37.

Finally the new approach is to add viscous terms corresponding to the Navier Stokes equations, in a limited area, either to the entire shocked region, or to the tip region only, whilst representing the remainder of the shock using one of the existing methods. This can be quite easily incorporated into the present code by using a finite volume discretization of the viscous terms as described by J. Mackenzie [5]. The effect would be to smear out the shock increasingly as the shock becomes weaker. Hopefully this would then allow us to solve the relevant equations in the shock region without generating the large errors and consequent oscillations, caused by averaging across a discontinuity using the trapezoidal rule. However a brief survey of the theory indicates that the shock width is equal to

$$\frac{8\gamma\upsilon_L}{3a_L(\gamma+1)M_L}\left[\frac{\sqrt{q_L}+\sqrt{q_R}}{\sqrt{q_L}-\sqrt{q_R}}\right]$$

for a normally orientated shock, where υ_L is the upstream kinematic viscosity of air q is the flow speed. Hence as $q_L \to q_R$ the shock width becomes infinite. In air a shock with $P_R/P_L = 2.0$, $\rho_R/\rho_L = 1.625$, $M_L = 1.363$ has a shock width equal to 40×10^{-5} mm, and a shock with $P_R/P_L = 1.2$, $\rho_R/\rho_L = 1.139$, $M_L = 1.082$ has a shock width of 190×10^{-5} MM (Hunt [6]). The latter result is typical of

the values found at the shock fitting tip. Hence it is only a very short length of the shock that has an appreciable width. Consequently an extremely fine grid would be required, in order to see the smoothing effect. Alternatively unrealistically high values of the viscosity parameter could be used in order to capture the shock.

REFERENCES

[1] K.W. MORTON & M.F. PAISLEY (1987). A finite volume scheme with shock fitting for the steady Euler equations. Oxford Univ. Comp. Lab., report 87/6.

[2] M.G. HALL (1984). Fast multigrid solution of the Euler equations using a finite volume scheme of Lax Wendroff type. RAE technical report 84013.

[3] P. SAMUELS (1989). Shock behaviour and diffusion. University of Reading, Numerical Analysis Report 12/89, pp 110-118.

[4] P.L. ROE (1981). Approximate Riemann solvers, parameter vectors, and difference schemes. Journal of Computational Physics 43, 357-372 (1981).

[5] J. MACKENZIE (1989). The cell vertex method for viscous transport problems. Oxford University Computing Lab. Report 89/4.

[6] J.N. HUNT. Shock waves in one dimensional flow. University of Reading, Dept. of Math., Lecture Notes III.5.

ACKNOWLEDGEMENTS

I would like to acknowledge the help received from Dr. C.M. Albone of RAE Farnborough as the external monitor, Dr. M.J. Baines for his constant supervision and guidance, Dr. A. Priestley for specific technical support, and Mrs S. Davis for the typing of this report.

List of Symbols

ρ	Density
u	velocity in x direction
v	velocity in y direction
P	pressure
γ	ratio of specific heats
α	shock orientation angle
s	shock speed
x,y	coordinate directions
t	time
q	flow speed
a	sound speed
μ_1	viscosity parameter
R	residual
I	vertical grid line subscript
J	horizontal grtid line subscript
IXS	I value of shock grid line
JMS	shock tip node J value
XS(J)	x coordinate of Jth shock node
L,R	Left and right subscripts
N,T	Normal and tangential subscripts
M	Mach number
$\underline{u}$	vector of unknowns
$\underline{F}$, $\underline{G}$	Flux vectors
λ	Characteristic speed

Fig. 1

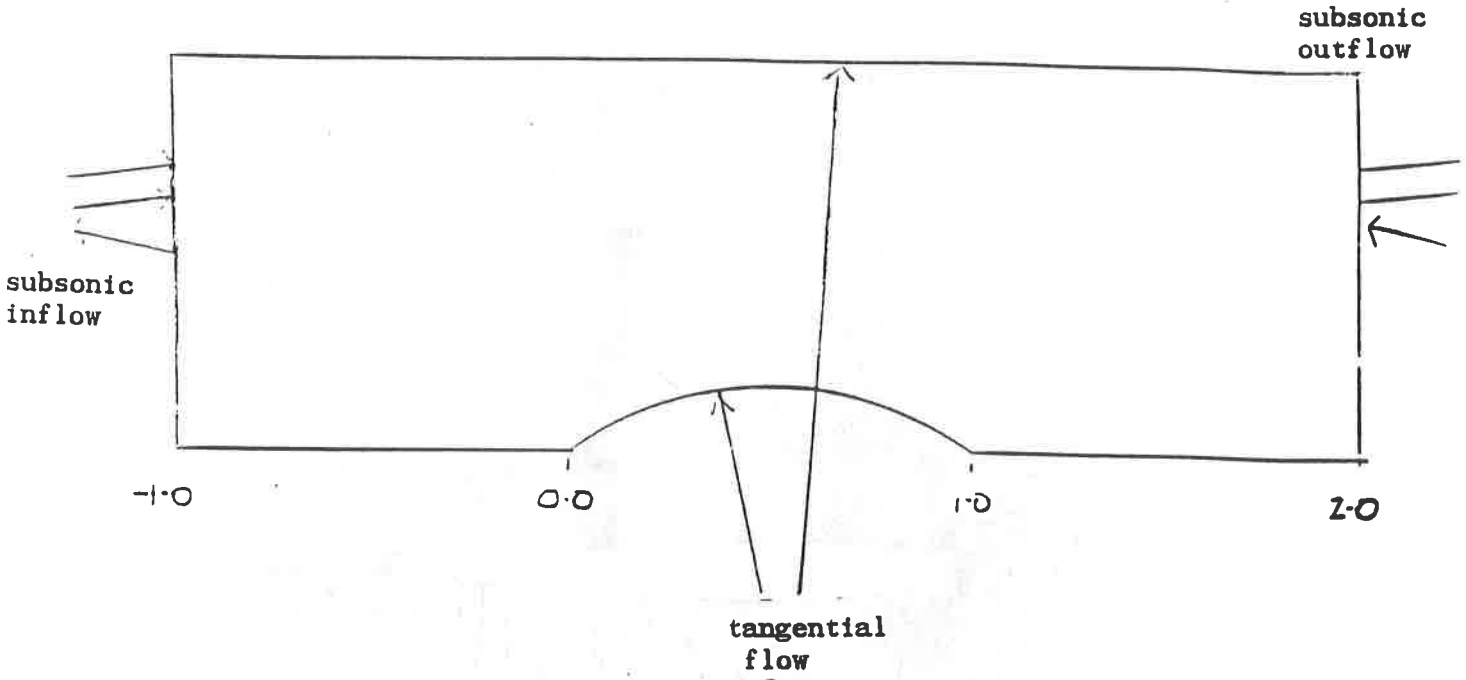
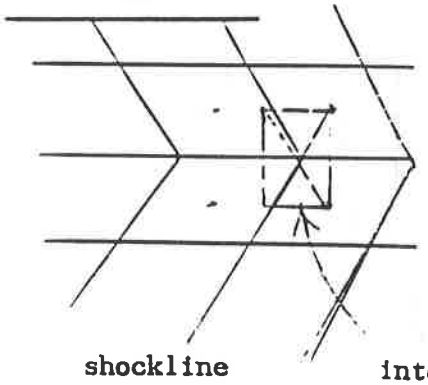
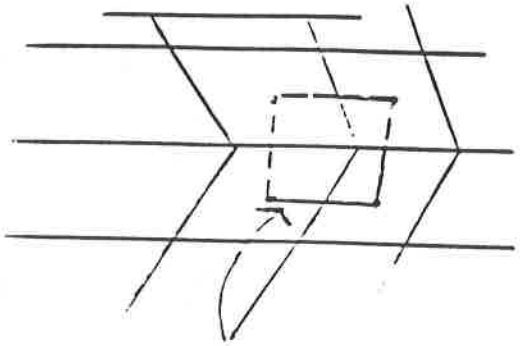


Fig. 2a



integration cell
used in righthand update
for original method

Fig. 2b



modified method uses integration cell, constructed from centroids of surrounding cells, for left and righthand updates

Fig. 3a

integration cell used in lefthand update of original method

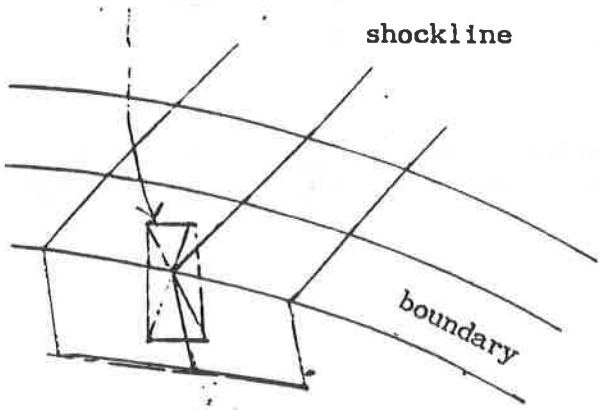


Fig. 3b

modified method uses integration cell constructed from two surrounding cells and their images for both left and righthand updates

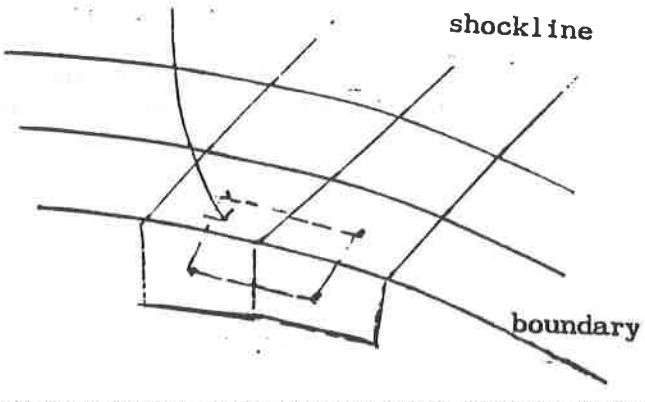


Fig. 4a

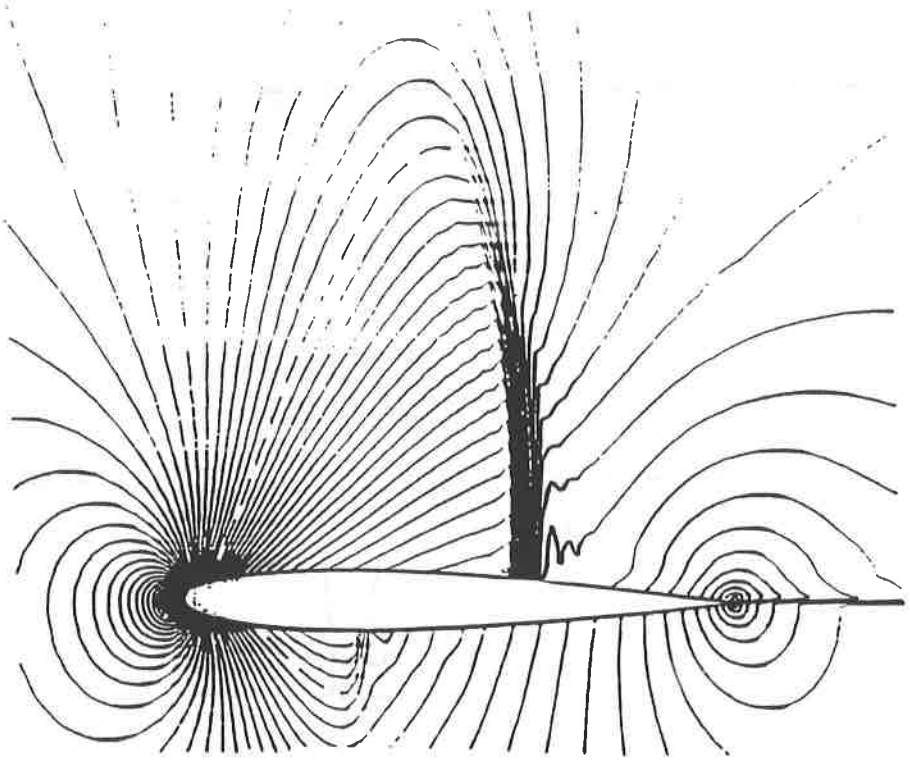
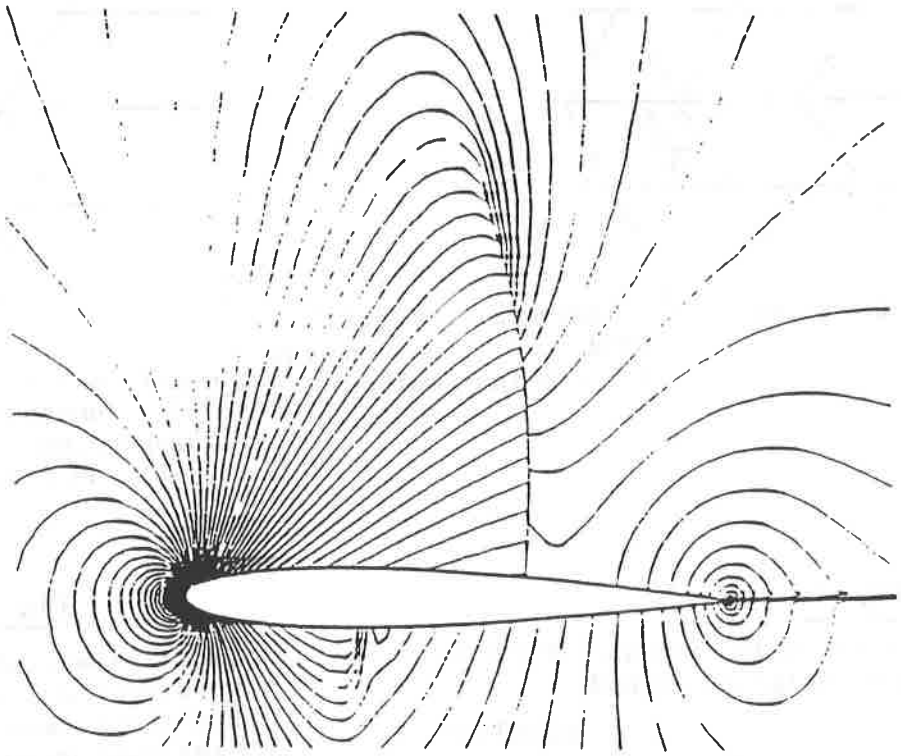


Fig. 4b



Mach contours for flow past the NACA0012 aerofoil (M_∞ = 0.8, α = 1.25) on the fine grid with shock capturing (above) and shock fitting (below). The dotted line is sonic and ΔM = 0.02.

Fig. 5

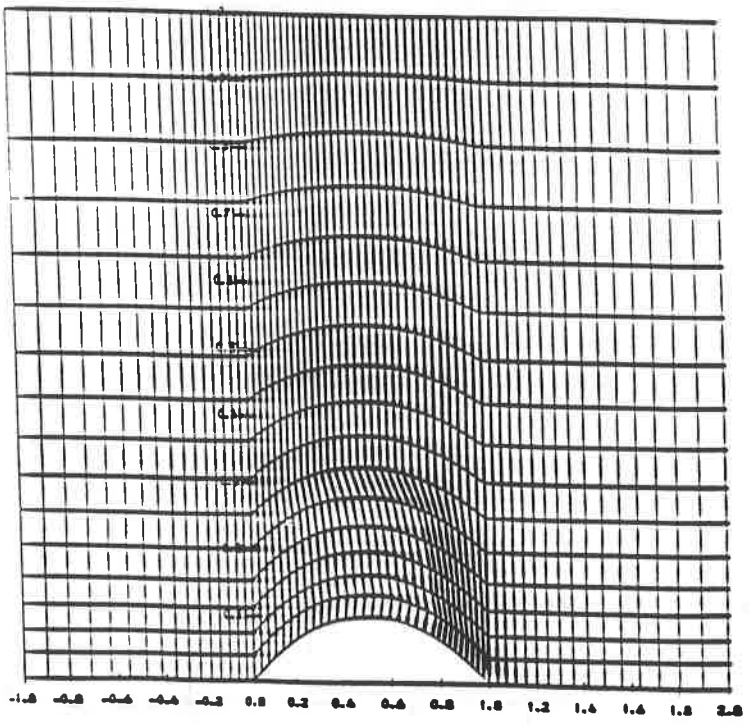


Fig. 7

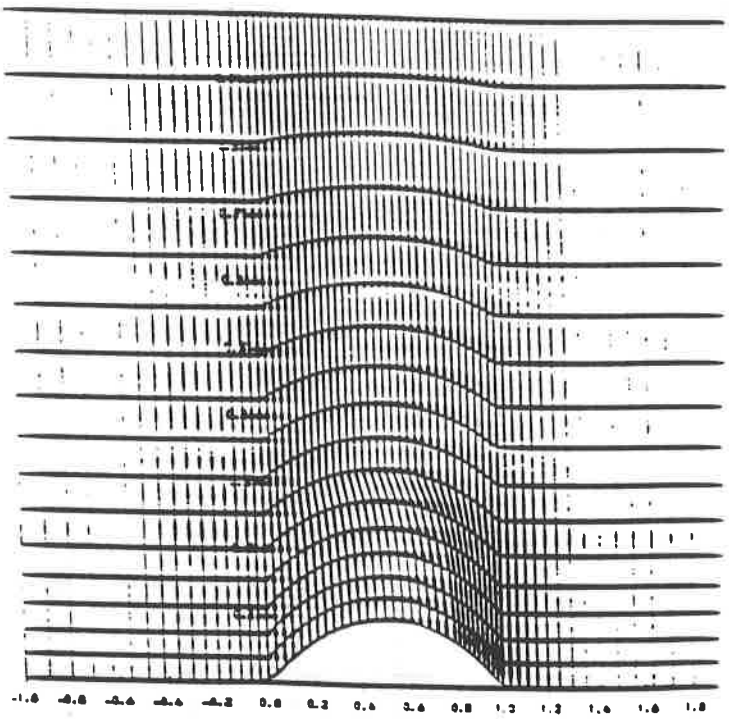


Fig. 6

Example of grid distortion, which resulted from increasing the grid line number corresponding to the shock.

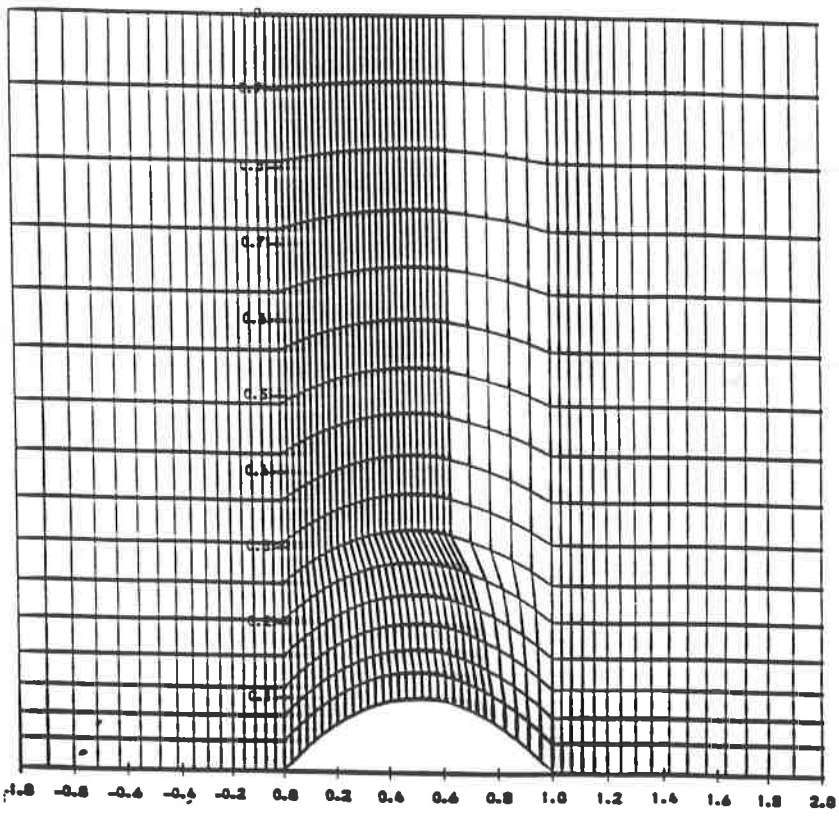


Fig. 7a

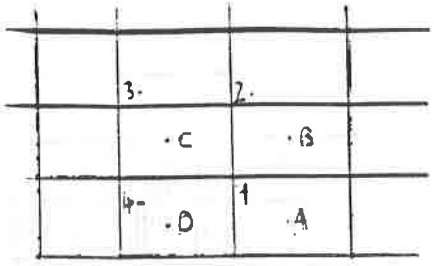
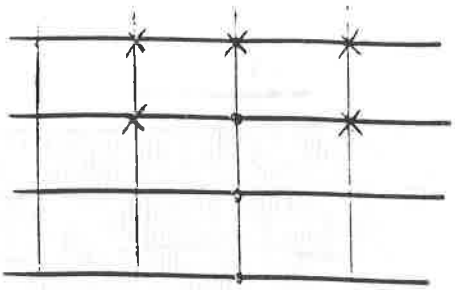
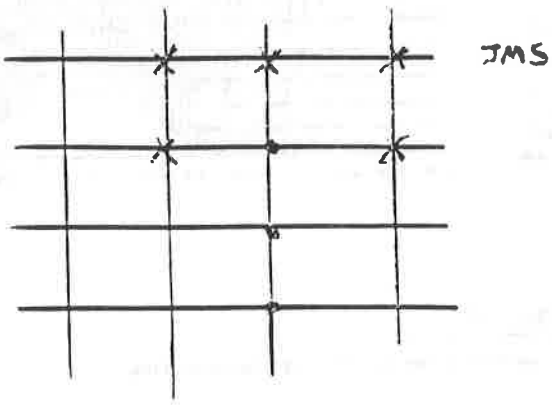


Fig. 8



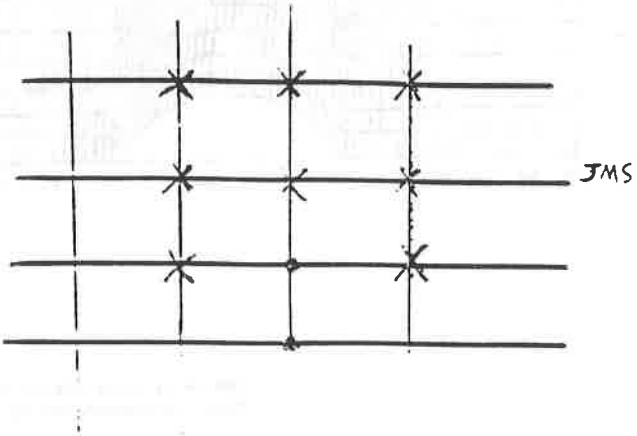
IXS

Fig. 9



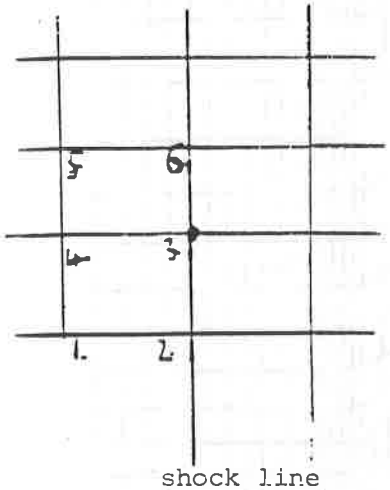
IXS

Fig. 10



IXS

Fig. 11



shock line

• marks a double valued shock node

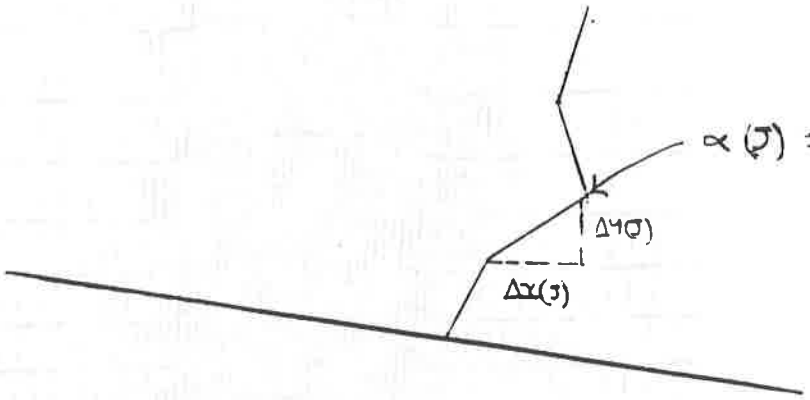
X marks a normal node to which viscosity has been added

JMS is the η index corresponding to the shock top

IXS is the X index corresponding to the shock foot

Fig. 12

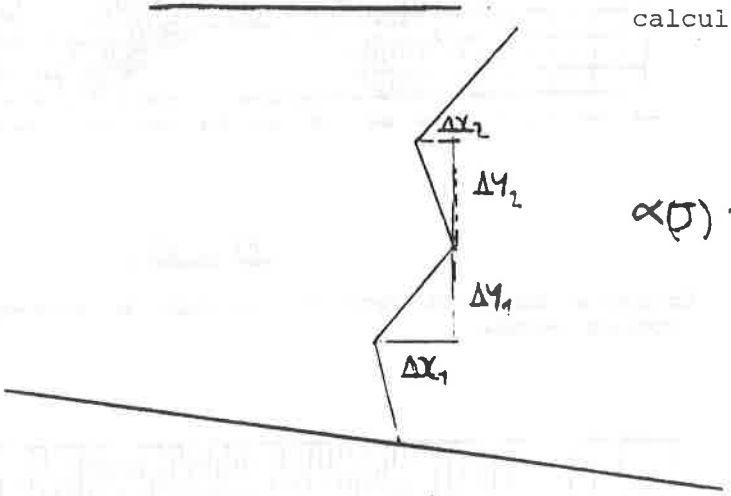
backward difference used to calculate $\alpha(J)$



$$\alpha(J) = \tan^{-1}\left[\frac{\Delta Y(J)}{\Delta X(J)}\right]$$

Fig. 13

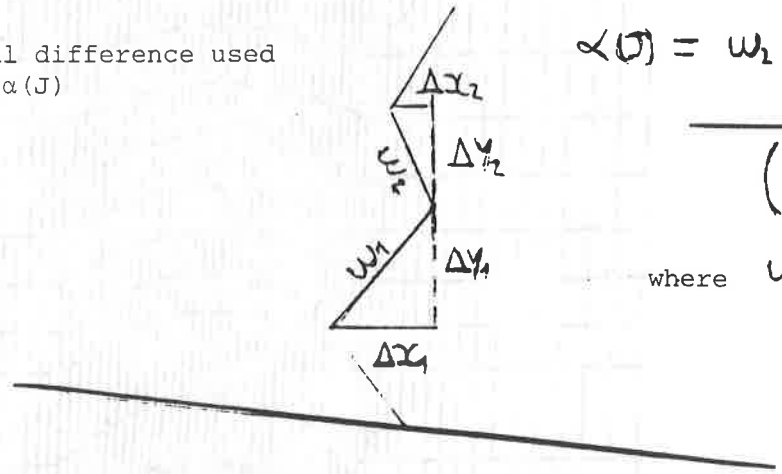
central difference used to calculate $\alpha(J)$



$$\alpha(J) = \frac{\tan^{-1}\left(\frac{\Delta Y_1}{\Delta X_1}\right) + \tan^{-1}\left(\frac{\Delta Y_2}{\Delta X_2}\right)}{2}$$

Fig. 14

weighted central difference used to calculate $\alpha(J)$



$$\alpha(J) = \frac{W_2 \tan^{-1}\left(\frac{\Delta Y_1}{\Delta X_1}\right) + W_1 \tan^{-1}\left(\frac{\Delta Y_2}{\Delta X_2}\right)}{(W_1 + W_2)}$$

where $W_i = \sqrt{\Delta X_i^2 + \Delta y_i^2}$

Fig. 15

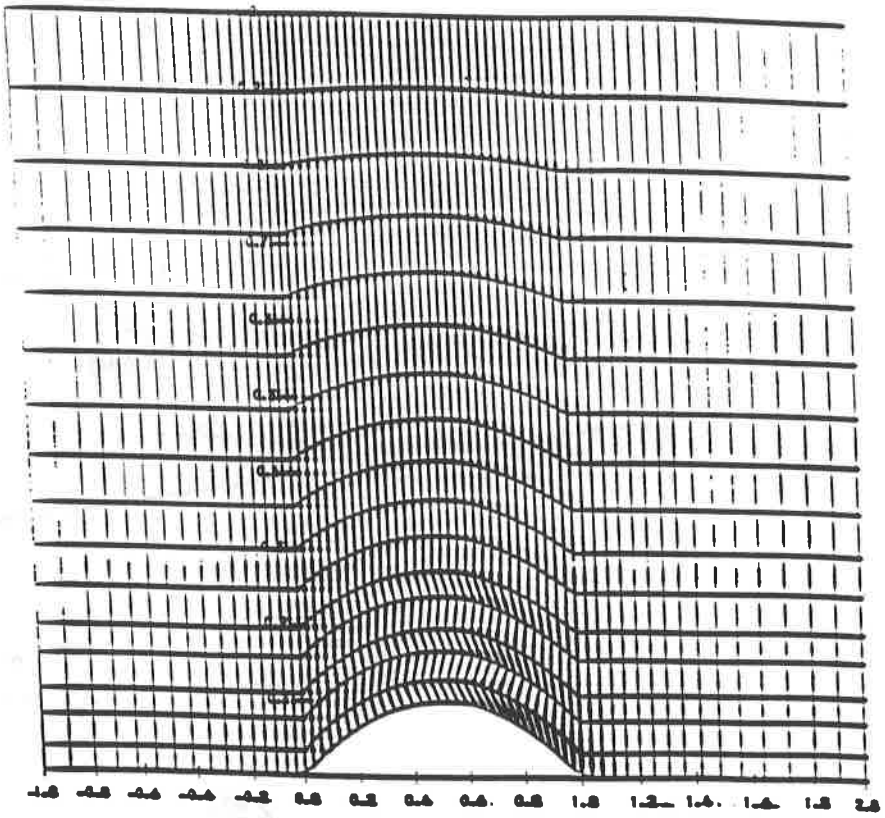


Fig. 16

Example of horizontal grid line bunching as observed when using front tracking methods.

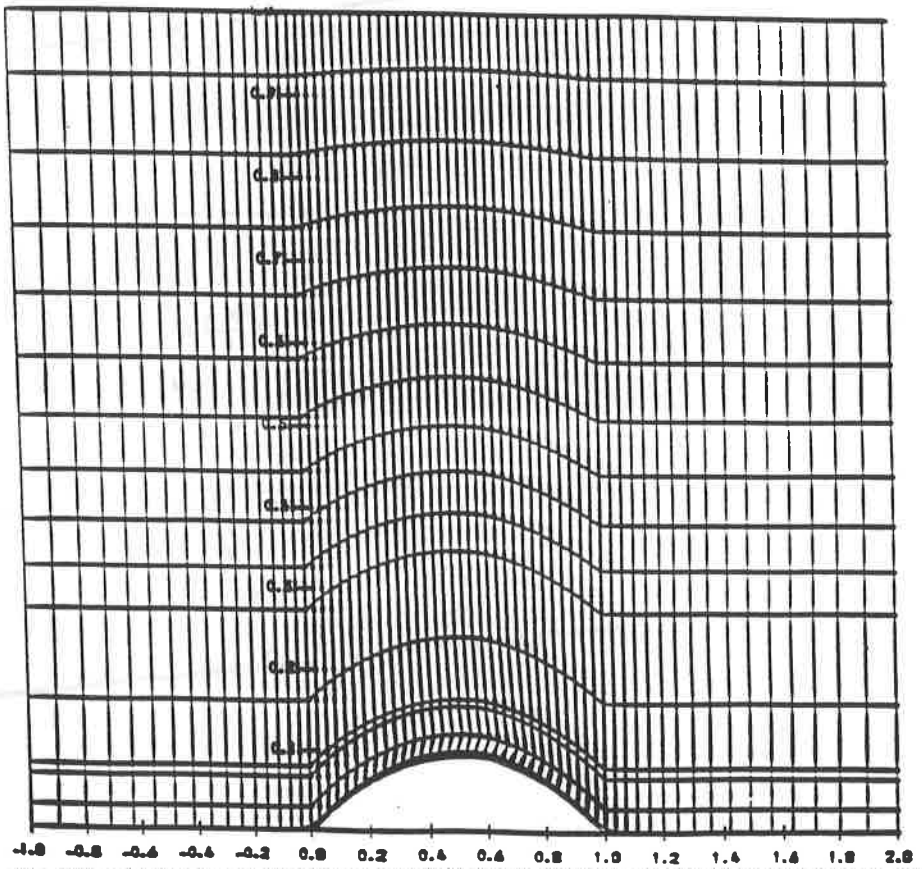
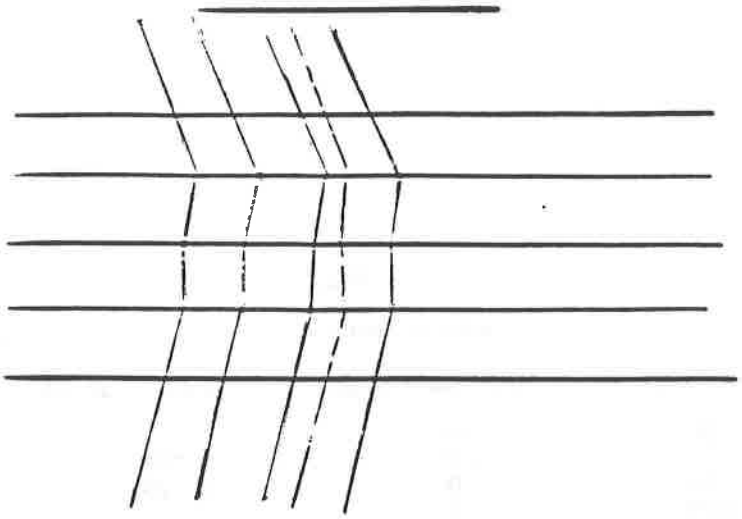


Fig. 17



mesh positioned so that shock exactly bisects upper and lower faces of shocked cells

Fig. 18

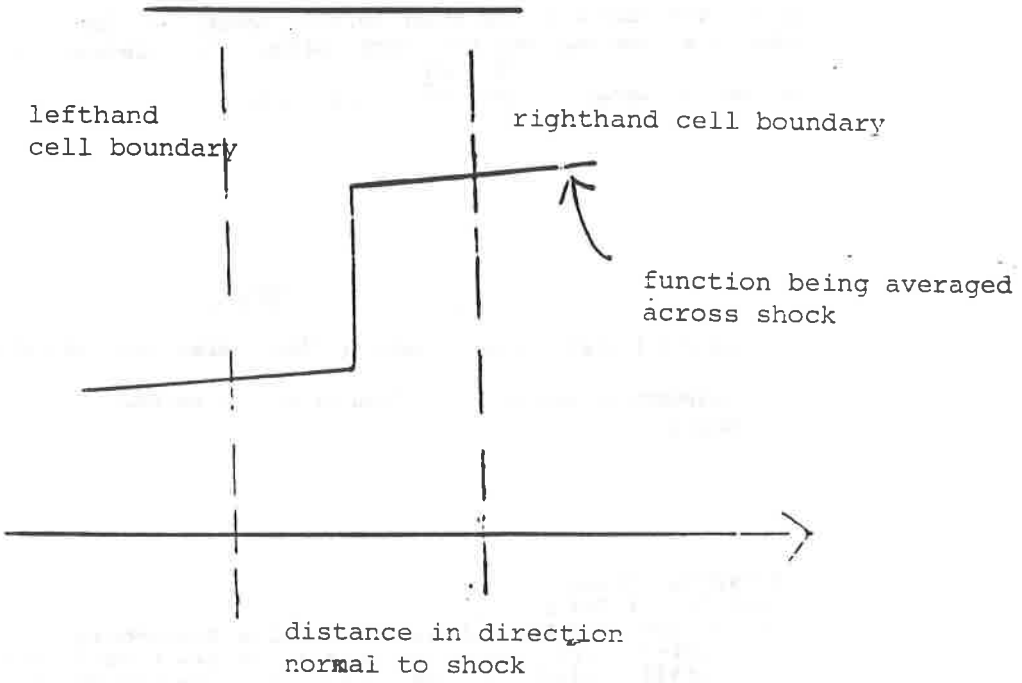


TABLE 1

(10.000 iterations using normal flow assumption)

ITERATION NUMBER	MAXIMUM SHOCK SPEED	FOOT POSITION	TIP POSITION
1.000	0.01		
2.000	0.001	0.779	0.669
3.000	0.0005	0.774	0.668
4.000	0.0003		
5.000	0.0005	0.771	0.667
6.000	0.0003		
7.000	0.0003	0.769	0.667
8.000	0.0003	0.769	0.666
9.000			
10.000	0.00007	0.768	0.666

Maximum residual = 0.1196 after 10.000 iterations. largest residuals occuring at leading edge and along bottom wall. (Residual = discrete approx to 1st component of $\frac{\partial \underline{F}}{\partial x} + \frac{\partial \underline{G}}{\partial y}$ in a cell)

TABLE 2

(20.000 iterations using normal flow assumption. no tip extrapolation or contraction, and no alteration of grid line number which corresponds to shock)

```
ITERATION  19900
  IXS IS   39 JMS IS    7
  1  0.6258   0.7118 -0.1601 ALFA= 1.349 S=-.000082   ML= 1.365   XS=0.785
  1  1.0195   0.7113 -0.1601 ALFA= 1.349 S=-.000082   ML= 1.365   XS=0.785
  2  0.6537   0.7295 -0.1619 ALFA= 1.352 S=-.000027   ML= 1.330   XS=0.783
  2  1.0251   0.7295 -0.1619 ALFA= 1.352 S=-.000027   ML= 1.330   XS=0.783
  3  0.6359   0.7459 -0.1557 ALFA= 1.365 S=-.000065   ML= 1.280   XS=0.778
  3  1.0159   0.7459 -0.1557 ALFA= 1.365 S=-.000065   ML= 1.280   XS=0.778
  4  0.7259   0.7655 -0.1457 ALFA= 1.393 S=-.000021   ML= 1.228   XS=0.766
  4  1.0038   0.7655 -0.1457 ALFA= 1.393 S=-.000021   ML= 1.228   XS=0.766
  5  0.7842   0.7787 -0.1293 ALFA= 1.406 S=0.000047   ML= 1.155   XS=0.750
  5  1.0785   0.7787 -0.1293 ALFA= 1.406 S=0.000047   ML= 1.155   XS=0.750
  6  0.8044   0.7924 -0.1041 ALFA= 1.440 S=0.000066   ML= 1.139   XS=0.723
  6  1.0523   0.7924 -0.1041 ALFA= 1.440 S=0.000066   ML= 1.139   XS=0.723
  7  0.8552   0.8101 -0.0726 ALFA= 1.481 S=-.000098   ML= 1.089   XS=0.670
  7  1.0272   0.8101 -0.0726 ALFA= 1.481 S=-.000098   ML= 1.089   XS=0.670
```

TABLE 3

(20,000 iterations of normal flow assumption, remove one shock node after

2,000 iterations no tip extrapolation contraction, or alteration of grid line
number corresponding to shock).

	ρ	ρu	ρv	α	S	M_L	XS
1	0.6210	0.7099	-0.1505	ALFA= 1.362	S=0.002132	ML= 1.368	XS=0.77
1	1.0148	0.7108	-0.1507	ALFA= 1.362	S=0.002132	ML= 1.369	XS=0.77
2	0.6571	0.7301	-0.1536	ALFA= 1.356	S=0.000337	ML= 1.322	XS=0.77
2	1.0209	0.7306	-0.1536	ALFA= 1.356	S=0.000337	ML= 1.322	XS=0.77
3	0.6957	0.7510	-0.1560	ALFA= 1.356	S=0.002130	ML= 1.265	XS=0.76
3	1.0117	0.7517	-0.1561	ALFA= 1.366	S=0.002130	ML= 1.265	XS=0.766
4	0.7351	0.7666	-0.1460	ALFA= 1.383	S=0.001255	ML= 1.205	XS=0.766
4	0.9928	0.7671	-0.1460	ALFA= 1.383	S=0.001255	ML= 1.205	XS=0.759
5	0.7733	0.7814	-0.1241	ALFA= 1.413	S=-.000419	ML= 1.152	XS=0.759
5	0.9710	0.7817	-0.1241	ALFA= 1.413	S=-.000419	ML= 1.152	XS=0.740
6	0.8103	0.7937	-0.0976	ALFA= 1.450	S=-.002507	ML= 1.096	XS=0.740
6	0.9512	0.7934	-0.0975	ALFA= 1.450	S=-.002507	ML= 1.096	XS=0.693

ITERATION 20000
I= 39J= 5IU= 2MAX UPDATE=0.00024729
RMS UPDATE =0.00002152
I= 37J= 6IU= 1MAX RESIDUAL=0.11567539
RMS RESIDUAL =0.00940242

TABLE 4

1,000 iteration result using one sided difference to calculate shock angles.

	ρ	ρu	ρv	α	S	M_L	XS
IXS IS 39 JMS IS 7							
1	0.7104	0.7645	-0.1181	ALFA= 1.418	S=-.105496	ML= 1.368	XS=0.69
1	1.1504	0.7176	-0.1108	ALFA= 1.418	S=-.105496	ML= 1.369	XS=0.69
2	0.7375	0.7959	-0.0867	ALFA= 3.007	S=********	ML= 1.239	XS=0.66
2	1.1041	1.1212	-0.5410	ALFA= 3.007	S=********	ML= 1.239	XS=0.66
3	0.7605	0.7934	-0.0703	ALFA= 1.086	S=-.115213	ML= 1.224	XS=0.69
3	1.0519	0.8187	0.0496	ALFA= 1.086	S=-.115213	ML= 1.224	XS=0.69
4	0.7935	0.7957	-0.0599	ALFA= 2.154	S=-.243744	ML= 1.179	XS=0.67
4	1.0222	0.8290	-0.2159	ALFA= 2.154	S=-.243744	ML= 1.179	XS=0.67
5	0.8100	0.7969	-0.0529	ALFA= 1.194	S=-.087039	ML= 1.143	XS=0.69
5	1.0063	0.8027	0.0095	ALFA= 1.194	S=-.087039	ML= 1.143	XS=0.69
6	0.8417	0.7977	-0.0485	ALFA= 1.822	S=-.099476	ML= 1.108	XS=0.66
6	0.9954	0.7941	-0.0958	ALFA= 1.822	S=-.099476	ML= 1.108	XS=0.66
7	0.8908	0.7978	-0.0467	ALFA= 1.494	S=-.069152	ML= 1.068	XS=0.68
7	0.9919	0.7909	-0.0445	ALFA= 1.494	S=-.069152	ML= 1.068	XS=0.66

ITERATION 1000
I= 39 J= 2 IU= 2 MAX UPDATE= 0.00123765
RMS UPDATE = 0.00005059
I= 39 J= 1 IU= 3 MAX RESIDUAL= **********
RMS RESIDUAL = **********

TABLE 5

SHOCK ANGLES CALCULATED USING ACTUAL POSITIONS OF NODES
AND AVERAGING GRADIENT EITHER SIDE OF EACH NODE.

```
      x          y          ν          α          S          ML         XS
IXS IS  39 JMS IS   7
 1  0.7792  0.7592 -0.1110 ALFA= 1.426 S=0.066627  ML= 1.023  XS=0.68
 1  0.8086  0.7612 -0.1113 ALFA= 1.426 S=0.066627  ML= 1.023  XS=0.68
 2  0.7915  0.7620 -0.1122 ALFA= 1.573 S=0.045478  ML= 1.013  XS=0.68
 2  0.8159  0.7631 -0.1157 ALFA= 1.573 S=0.045478  ML= 1.018  XS=0.68
 3  0.8107  0.7760 -0.1040 ALFA= 1.549 S=0.045965  ML= 1.013  XS=0.68
 3  0.8284  0.7768 -0.1059 ALFA= 1.549 S=0.045965  ML= 1.013  XS=0.68
 4  0.8438  0.7850 -0.1001 ALFA= 1.526 S=0.016809  ML= 1.011  XS=0.68
 4  0.8594  0.7852 -0.1013 ALFA= 1.526 S=0.016809  ML= 1.011  XS=0.68
 5  0.8790  0.7915 -0.0964 ALFA= 1.514 S=-.016525  ML= 1.009  XS=0.69
 5  0.8917  0.7913 -0.0972 ALFA= 1.514 S=-.016525  ML= 1.009  XS=0.69
 6  0.9129  0.7966 -0.0916 ALFA= 1.506 S=-.047338  ML= 1.005  XS=0.69
 6  0.9212  0.7962 -0.0920 ALFA= 1.506 S=-.047338  ML= 1.005  XS=0.69
 7  0.9376  0.7986 -0.0840 ALFA= 1.509 S=-.070032  ML= 1.002  XS=0.69
 7  0.9410  0.7984 -0.0841 ALFA= 1.509 S=-.070032  ML= 1.002  XS=0.69
ITERATION    5
 1  0.6289  0.7258 -0.1213 ALFA= 1.405 S=0.031217  ML= 1.336  XS=0.71
 1  0.9932  0.7371 -0.1231 ALFA= 1.405 S=0.031217  ML= 1.335  XS=0.71
 2  0.6625  0.7420 -0.0975 ALFA= 1.616 S=0.001029  ML= 1.299  XS=0.68
 2  0.9898  0.7452 -0.1620 ALFA= 1.616 S=0.001029  ML= 1.288  XS=0.68
 3  0.7053  0.7668 -0.1033 ALFA= 1.557 S=0.024846  ML= 1.221  XS=0.70
 3  0.9725  0.7729 -0.1395 ALFA= 1.557 S=0.024846  ML= 1.221  XS=0.70
 4  0.7304  0.7699 -0.0982 ALFA= 1.736 S=0.000045  ML= 1.159  XS=0.68
 4  0.9274  0.7794 -0.1451 ALFA= 1.736 S=0.000045  ML= 1.159  XS=0.68
 5  0.7312  0.7954 -0.0981 ALFA= 1.631 S=0.016125  ML= 1.176  XS=0.69
 5  0.9218  0.8011 -0.1192 ALFA= 1.631 S=0.016125  ML= 1.176  XS=0.69
 6  0.8090  0.7936 -0.0754 ALFA= 1.775 S=-.010125  ML= 1.050  XS=0.67
 6  0.8910  0.7976 -0.0989 ALFA= 1.775 S=-.010125  ML= 1.050  XS=0.67
 7  0.8682  0.8097 -0.0756 ALFA= 1.500 S=0.001904  ML= 1.035  XS=0.67
 7  0.8153  0.8097 -0.0767 ALFA= 1.500 S=0.001904  ML= 1.035  XS=0.67
ITERATION   500
  IXS IS  39 JMS IS   7
 1  0.6272  0.7212 -0.1309 ALFA= 1.391 S=0.028127  ML= 1.339  XS=0.71
 1  0.9921  0.7311 -0.1327 ALFA= 1.391 S=0.028127  ML= 1.338  XS=0.71
 2  0.6600  0.7416 -0.0991 ALFA= 1.600 S=-.003148  ML= 1.301  XS=0.68
 2  1.0021  0.7423 -0.1600 ALFA= 1.600 S=-.003148  ML= 1.301  XS=0.68
 3  0.7021  0.7653 -0.1039 ALFA= 1.544 S=0.018742  ML= 1.215  XS=0.72
 3  0.9849  0.7697 -0.1447 ALFA= 1.544 S=0.018742  ML= 1.215  XS=0.72
 4  0.7247  0.7670 -0.0999 ALFA= 1.747 S=-.000378  ML= 1.151  XS=0.68
 4  0.9233  0.7777 -0.1500 ALFA= 1.747 S=-.000378  ML= 1.151  XS=0.68
 5  0.7750  0.7940 -0.1004 ALFA= 1.599 S=0.013499  ML= 1.113  XS=0.70
 5  0.9238  0.8007 -0.1264 ALFA= 1.599 S=0.013499  ML= 1.113  XS=0.70
 6  0.8015  0.7925 -0.0731 ALFA= 1.825 S=-.000829  ML= 1.043  XS=0.67
 6  0.8592  0.7974 -0.0919 ALFA= 1.825 S=-.000829  ML= 1.043  XS=0.67
 7  0.8573  0.8093 -0.0748 ALFA= 1.335 S=0.004992  ML= 1.035  XS=0.68
 7  0.8103  0.8104 -0.0724 ALFA= 1.335 S=0.004992  ML= 1.035  XS=0.68
ITERATION   1000
I=  39  J=   5  IU=   2 MAX UPDATE=  0.00013642
RMS UPDATE =  0.00002959
I=  37  J=   1 IU=   3 MAX RESIDUAL= 0.19682190
RMS RESIDUAL =  0.01618513
```

TABLE 6

Best results obtained using least squares cubic with multigrid after 5.000 iterations.

```
ITERATION   4920
  IXS IS  39 JMS IS   6
 1  0.6118  0.7159 -0.1160 ALFA= 1.410 S=0.001364  ML= 1.396  XS=0.708
 1  1.0300  0.7165 -0.1160 ALFA= 1.410 S=0.001364  ML= 1.396  XS=0.708
 2  0.6436  0.7356 -0.1089 ALFA= 1.511 S=-.002370  ML= 1.347  XS=0.712
 2  1.0286  0.7324 -0.1473 ALFA= 1.511 S=-.002370  ML= 1.347  XS=0.712
 3  0.6782  0.7526 -0.1018 ALFA= 1.595 S=-.003317  ML= 1.281  XS=0.712
 3  1.0059  0.7530 -0.1599 ALFA= 1.595 S=-.003317  ML= 1.281  XS=0.712
 4  0.7171  0.7712 -0.0982 ALFA= 1.669 S=0.003879  ML= 1.204  XS=0.710
 4  0.9669  0.7781 -0.1583 ALFA= 1.669 S=0.003879  ML= 1.204  XS=0.710
 5  0.7614  0.7839 -0.0914 ALFA= 1.728 S=-.001158  ML= 1.128  XS=0.705
 5  0.9264  0.7910 -0.1371 ALFA= 1.728 S=-.001158  ML= 1.128  XS=0.705
 6  0.8098  0.7992 -0.0913 ALFA= 1.768 S=-.005255  ML= 1.061  XS=0.697
 6  0.8935  0.8038 -0.1163 ALFA= 1.768 S=-.005255  ML= 1.061  XS=0.697
```

TABLE 7 (17 x 65 grid)

$\mu_1 = 0.005$

XS(1) = 0.7253	YS(1) = 0.0803
XS(2) = 0.7246	YS(2) = 0.1118
XS(3) = 0.7243	YS(3) = 0.1455
XS(4) = 0.7261	YS(4) = 0.1816
XS(5) = 0.7292	YS(5) = 0.2202
XS(6) = 0.7059	YS(6) = 0.2654
XS(7) = 0.6897	YS(7) = 0.3124
XS(8) = 0.6958	YS(8) = 0.3601

$$\text{RMS}\left[\frac{\text{UPDATE}}{\text{DT}}\right] = 2.0 \times 10^{-7}$$

$\mu_1 = 0.01$

XS(1) = 0.6844	YS(1) = 0.0869
XS(2) = 0.6836	YS(2) = 0.1182
XS(3) = 0.6839	YS(3) = 0.1516
XS(4) = 0.6861	YS(4) = 0.1874
XS(5) = 0.6892	YS(5) = 0.2258
XS(6) = 0.6932	YS(6) = 0.2670
XS(7) = 0.6968	YS(7) = 0.3116

$$\text{RMS}\left[\frac{\text{UPDATE}}{\text{DT}}\right] = 2.0 \times 10^{-8}$$

$\mu_1 = 0.02$

XS(1) = 0.6870	YS(1) = 0.0865
XS(2) = 0.6869	YS(2) = 0.1177
XS(3) = 0.6870	YS(3) = 0.1512
XS(4) = 0.6886	YS(4) = 0.1870
XS(5) = 0.6907	YS(5) = 0.2256
XS(6) = 0.6939	YS(6) = 0.2669
XS(7) = 0.6971	YS(7) = 0.3115

$$\text{RMS}\left[\frac{\text{UPDATE}}{\text{DT}}\right] = 5.0 \times 10^{-8}$$

$\mu_1 = 0.03$

XS(1) = 0.6868	YS(1) = 0.0865
XS(2) = 0.6875	YS(2) = 0.1176
XS(3) = 0.6880	YS(3) = 0.1510
XS(4) = 0.6893	YS(4) = 0.1869
XS(5) = 0.6910	YS(5) = 0.2255
XS(6) = 0.6942	YS(6) = 0.2669
XS(7) = 0.6976	YS(7) = 0.3115

$$\text{RMS}\left[\frac{\text{UPDATE}}{\text{DT}}\right] = 1.3 \times 10^{-7}$$

$\mu_1 = 0.04$

XS(1) = 0.6865	YS(1) = 0.0866
XS(2) = 0.6873	YS(2) = 0.1176
XS(3) = 0.6877	YS(3) = 0.1511
XS(4) = 0.6891	YS(4) = 0.1869
XS(5) = 0.6909	YS(5) = 0.2255
XS(6) = 0.6945	YS(6) = 0.2669
XS(7) = 0.6765	YS(7) = 0.3138

$$\text{RMS}\left[\frac{\text{UPDATE}}{\text{DT}}\right] = 2.3 \times 10^{-7}$$

$\mu_1 = 0.013$

XS(1) = 0.6858	YS(1) = 0.0867
XS(2) = 0.6856	YS(2) = 0.1179
XS(3) = 0.6859	YS(3) = 0.1513
XS(4) = 0.6877	YS(4) = 0.1871
XS(5) = 0.6904	YS(5) = 0.2256
XS(6) = 0.6939	YS(6) = 0.2669
XS(7) = 0.6971	YS(7) = 0.3116

$$\text{RMS}\left[\frac{\text{UPDATE}}{\text{DT}}\right] = 2.0 \times 10^{-8}$$

TABLE 8 (33 x 129 grid)

$\mu_1 = 0.01$

XS(1) = 0.7027	YS(1) = 0.0841
XS(2) = 0.7034	YS(2) = 0.0997
XS(3) = 0.7033	YS(3) = 0.1159
XS(4) = 0.7033	YS(4) = 0.1327
XS(5) = 0.7033	YS(5) = 0.1502
XS(6) = 0.7036	YS(6) = 0.1682
XS(7) = 0.7040	YS(7) = 0.1868
XS(8) = 0.7048	YS(8) = 0.2061
XS(9) = 0.7060	YS(9) = 0.2261
XS(10) = 0.6963	YS(10) = 0.2481
XS(11) = 0.6872	YS(11) = 0.2702
XS(12) = 0.6891	YS(12) = 0.2988
XS(13) = 0.6905	YS(13) = 0.3157
XS(14) = 0.6912	YS(14) = 0.3396

$$\text{RMS}\left[\frac{\text{UPDATE}}{\text{DT}}\right] = 3.3 \times 10^{-5}$$

$\mu_1 = 0.02$

XS(1) = 0.7021	YS(1) = 0.0842
XS(2) = 0.7027	YS(2) = 0.0998
XS(3) = 0.7027	YS(3) = 0.1160
XS(4) = 0.7029	YS(4) = 0.1328
XS(5) = 0.7031	YS(5) = 0.1502
XS(6) = 0.7035	YS(6) = 0.1682
XS(7) = 0.7045	YS(7) = 0.1868
XS(8) = 0.6954	YS(8) = 0.2074
XS(9) = 0.6877	YS(9) = 0.2284
XS(10) = 0.6883	YS(10) = 0.2491
XS(11) = 0.6891	YS(11) = 0.2705
XS(12) = 0.6903	YS(12) = 0.2927
XS(13) = 0.6758	YS(13) = 0.3173
XS(14) = 0.6768	YS(14) = 0.3411

$$\text{RMS}\left[\frac{\text{UPDATE}}{\text{DT}}\right] = 1.9 \times 10^{-4}$$

$\mu_1 = 0.03$

XS(1) = 0.6865	YS(1) = 0.0866
XS(2) = 0.6867	YS(2) = 0.1022
XS(3) = 0.6865	YS(3) = 0.1184
XS(4) = 0.6868	YS(4) = 0.1351
XS(5) = 0.6869	YS(5) = 0.1525
XS(6) = 0.6869	YS(6) = 0.1705
XS(7) = 0.6872	YS(7) = 0.1891
XS(8) = 0.6874	YS(8) = 0.2084
XS(9) = 0.6877	YS(9) = 0.2284
XS(10) = 0.6881	YS(10) = 0.2490
XS(11) = 0.6890	YS(11) = 0.2723
XS(12) = 0.6735	YS(12) = 0.2945
XS(13) = 0.6739	YS(13) = 0.3174
XS(14) = 0.6748	YS(14) = 0.3412

$$\text{RMS}\left[\frac{\text{UPDATE}}{\text{DT}}\right] = 1.2 \times 10^{-5}$$

$\mu_1 = 0.04$

XS(1) = 0.6863	YS(1) = 0.0866
XS(2) = 0.6864	YS(2) = 0.1022
XS(3) = 0.6863	YS(3) = 0.1184
XS(4) = 0.6866	YS(4) = 0.1352
XS(5) = 0.6868	YS(5) = 0.1525
XS(6) = 0.6870	YS(6) = 0.1705
XS(7) = 0.6872	YS(7) = 0.1891
XS(8) = 0.6875	YS(8) = 0.2084
XS(9) = 0.6878	YS(9) = 0.2284
XS(10) = 0.6885	YS(10) = 0.2491
XS(11) = 0.6739	YS(11) = 0.2722
XS(12) = 0.6741	YS(12) = 0.2944
XS(13) = 0.6749	YS(13) = 0.3174
XS(14) = 0.6758	YS(14) = 0.3412

$$\text{RMS}\left[\frac{\text{UPDATE}}{\text{DT}}\right] = 3.2 \times 10^{-5}$$

TABLE 9

Grid position 1
XS(1) = 0.6878
XS(2) = 0.6874
XS(3) = 0.6873
XS(4) = 0.6689
XS(5) = 0.6907
XS(6) = 0.6936
XS(7) = 0.6966

Grid position 2
XS(1) = 0.7063
XS(2) = 0.7072
XS(3) = 0.7076
XS(4) = 0.7103
XS(5) = 0.7144
XS(6) = 0.6966
XS(7) = 0.6864

Grid position 3
XS(1) = 0.6941
XS(2) = 0.6951
XS(3) = 0.6957
XS(4) = 0.6990
XS(5) = 0.7062
XS(6) = 0.6961
XS(7) = 0.7125

Grid position 4
XS(1) = 0.7174
XS(2) = 0.7195
XS(3) = 0.7206
XS(4) = 0.7252
XS(5) = 0.6941
XS(6) = 0.6902
XS(7) = 0.7044
XS(8) = 0.7071

Grid position 5
XS(1) = 0.7081
XS(2) = 0.7098
XS(3) = 0.7121
XS(4) = 0.7183
XS(5) = 0.7139
XS(6) = 0.7162
XS(7) = 0.7061

Grid position 6
XS(1) = 0.6992
XS(2) = 0.7005
XS(3) = 0.7035
XS(4) = 0.7109
XS(5) = 0.7094
XS(6) = 0.7130
XS(7) = 0.7067
XS(8) = 0.7282

Grid position 7
XS(1) = 0.6907
XS(2) = 0.6906
XS(3) = 0.6945
XS(4) = 0.7033
XS(5) = 0.7042
XS(6) = 0.7095
XS(7) = 0.7058

Grid position 8
XS(1) = 0.7192
XS(2) = 0.7192
XS(3) = 0.7245
XS(4) = 0.6937
XS(5) = 0.6974
XS(6) = 0.7047
XS(7) = 0.7059
XS(8) = 0.7077

Grid position 9
XS(1) = 0.7090
XS(2) = 0.7119
XS(3) = 0.7210
XS(4) = 0.7207
XS(5) = 0.7298
XS(6) = 0.7090
XS(7) = 0.7087
XS(8) = 0.7431

Grid position 10
XS(1) = 0.8618
XS(2) = 0.7073
XS(3) = 0.7163
XS(4) = 0.7167
XS(5) = 0.7144
XS(6) = 0.7268
XS(7) = 0.7169
XS(8) = 0.7618