JOURNAL OF COMPUTATIONAL PHYSICS 80, 168–203 (1989)

A Finite Volume Scheme with Shock Fitting for the Steady Euler Equations

K. W. MORTON

Oxford University Computing Laboratory, Oxford OX1 3QD, England

AND

M. F. PAISLEY

Royal Aircraft Establishment, Farnborough, Hants GU14 6TD, England

Received July 29, 1987; revised December 4, 1987

Analysis of two alternative finite volume formulations, in respect of accuracy on non-uniform meshes and number of spurious modes, leads to a preference for the more compact cell vertex scheme over the cell centre scheme. The resulting equations are solved iteratively by using a Lax–Wendroff procedure as a smoother for a multigrid algorithm: then application of boundary conditions in a natural way leads rapidly to all individual residuals being driven close to zero—except at shocks. At shocks the residuals should not be zero and a shock-fitting procedure is introduced to avoid this inconsistency. Sharp, accurate solutions on a relatively coarse mesh are obtained for a channel flow problem in which the Zierep singularity is displayed, and for the NACA 0012 aerofoil.

1. INTRODUCTION

Great advances in the simulation of transonic inviscid flows have been made in the last few years. For genuinely unsteady flows it is highly desirable to work with the characteristic variables in some form; for steady flows the conservation law form is a simpler alternative which is quite adequate for most of the flow domain. Jameson and his collaborators [5, 6] have pioneered the use of a finite volume formulation to approximate the steady Euler equations and have produced a series of codes which are efficient, capable of producing flows over a complete aircraft, and very popular in the aerospace community. The finite volume scheme usually adopted in these codes uses a structured, body-fitted mesh with the flow variables associated with the centres of the cells which are quadrilaterals in two dimensions. In parallel with this work, Ni [13] and Denton [2] have used finite-volume schemes in which the flow variables are associated with the cell vertices and have achieved similar successes. Both approaches use a pseudo time-stepping method to solve the algebraic equations, the former usually adopting a Runge–Kutta procedure and the latter a Lax–Wendroff process.

0021-9991/89 $3.00

In the present paper we begin by comparing the two finite volume formulations in two dimensions. Within the context of a time-stepping iteration to steady state, the cell centre approach has immediate advantages. The residual vector over each cell is directly associated with the state vector at its centre, making the equality of numbers of equations and unknowns obvious and simplifying the task of setting up an iteration. However, once an appropriate procedure has been devised for driving the residuals to zero in the cell vertex scheme, the resultant approximation has several advantages. It retains second-order accuracy when opposite mesh sides differ by $O(h)$, while the cell centre scheme allows only $O(h^2)$ differences if it is to maintain its accuracy. And it supports only one spurious solution mode as opposed to three in the cell centre case. The greater compactness of the scheme which is related to both these advantages, also affects the greater ease with which boundary conditions can be applied.

Ni [13] used a Lax–Wendroff procedure for the iteration of the cell vertex scheme, but it is important that the improvement suggested by Hall [4] be used too; an analysis shows that without it the second-order damping can be nullified by first-order amplification of errors arising from the non-uniformity of the mesh. Rather than the two-step form of Lax–Wendroff favoured because of its conservation properties for unsteady problems, a one-step form is used to ensure that when convergence is achieved all the individual cell residuals are set to zero. Proper treatment of the boundary conditions is also crucial to ensuring this property. Fourier analysis of the scalar case on a uniform rectangular mesh shows that a CFL condition of the familiar form $v_x^2 + v_y^2 \leqslant 1$ is necessary and sufficient for no growth of error. The optimum choice of Δt for maximum error damping is discussed; it bears out the practical observation that the local Δt should be chosen close to the local CFL limit.

When used as a smoother for a multigrid scheme, the Lax–Wendroff method is very efficient at achieving convergence away from shocks; it merely needs a little extra damping in the neighbourhood of sonic lines and to check the sole spurious mode. The position at a shock, however, is very different. The crucial point is that the usual residual should not be zero for a cell crossed by a shock, except in certain special cases. Any iteration that tries to make it so has to have large damping terms added to spread the error over a number of neighbouring cells. Moreover, these will need to contain carefully chosen parameters because simple experiments in one-dimensional nozzle problems readily show that the shock position is a very sensitive function of the form of damping chosen [15].

One is led to the conclusion that some form of shock fitting is needed, as long advocated by Moretti and his colleagues [9]. We present a shock-fitting procedure for a simple two-dimensional shock which is reasonably well aligned with one set of mesh lines. The approach is to capture the shock after a number of iterations and then adapt the mesh in its neighbourhood. The shock can then be treated as an internal boundary, using the Rankine–Hugoniot conditions as boundary conditions. A shock speed is obtained which is used to move the shock and its neighbouring mesh, as the iteration proceeds. Results are given for the channel flow

problem used by Ni [13] and for the NACA 0012 aerofoil at Mach number $M_\infty = 0.8$ and angle of attack $\alpha = 1.25°$ and at $M_\infty = 0.85$, $\alpha = 1.0°$. Even on a relatively coarse mesh, very accurate results are obtained.

2. Problem Formulation

The full unsteady Euler equations describing inviscid flow in two dimensions express the conservation of mass, the two components of momentum, and the energy. We use the notation u, v, ρ, P, E, H for the two Cartesian components of velocity, the density, pressure, total energy, and total enthalpy, respectively: the energy and enthalpy are related by

$$H = E + \frac{P}{\rho} \tag{2.1}$$

and, in addition, we have an equation of state, which for an ideal gas leads to

$$E = \frac{P}{(\gamma - 1)\rho} + \frac{1}{2}(u^2 + v^2), \tag{2.2}$$

where γ is the ratio of specific heats.

In many applications, only the steady state solution with a constant state at infinity is of interest. Then a reduced system can be derived by combining the steady energy equation $(\rho u H)_x + (\rho v H)_y = 0$ with the steady mass conservation equation to give

$$uH_x + vH_y = 0, \tag{2.3}$$

implying that enthalpy is constant along streamlines. Since in the applications considered here all the streamlines originate in the constant free-stream, H must be the same constant H_{const} on each: substituting into (2.1) and (2.2) gives

$$P = \frac{\rho}{\gamma}(\gamma - 1)\left(H_{\text{const}} - \frac{1}{2}(u^2 + v^2)\right). \tag{2.4}$$

This algebraic relation together with the system

$$\rho_t + (\rho u)_x + (\rho v)_y = 0 \tag{2.5a}$$

$$(\rho u)_t + (P + \rho u^2)_x + (\rho uv)_y = 0 \tag{2.5b}$$

$$(\rho v)_t + (\rho uv)_x + (P + \rho v^2)_y = 0 \tag{2.5c}$$

forms the standard pseudo-unsteady system for the Euler equations, termed the H-system by Viviand [27].

The system (2.5) is totally hyperbolic, and the characteristic speeds for unidirectional flow are

$$\lambda = q \qquad \text{and} \qquad \lambda = \frac{\gamma+1}{2\gamma} q \pm \sqrt{\left(\frac{a^2}{\gamma} + \left[\frac{\gamma-1}{2\gamma} q\right]^2\right)}, \tag{2.6}$$

where q is the flow speed and $a = (\gamma P/\rho)^{1/2}$ is the speed of sound.

It is useful for computational purposes to non-dimensionalise the equations, and we follow the usual practice by taking the units of length, speed, and pressure to be a typical length L', stagnation sound speed a_0', and stagnation pressure P_0'. This leaves all Eqs. (2.5) in the same form as before, so they can be viewed as equations in the new non-dimensional variables:

$$x = \frac{x'}{L'}, \qquad y = \frac{y'}{L'}, \qquad t = \frac{a_0'}{L'} t',$$

$$u = \frac{u'}{a_0'}, \qquad v = \frac{v'}{a_0'}, \qquad P = \frac{P'}{P_0'}, \qquad \rho = \frac{(a_0')^2}{P_0'} \rho'.$$

However, the pressure relation (2.4) becomes

$$P = \frac{\rho}{\gamma}\left(1 - \frac{1}{2}(\gamma-1)(u^2+v^2)\right). \tag{2.7}$$

The problems that we shall consider are of two types, flow through an infinite channel and flow past a fixed body in space. On the channel wall and on the body we impose the single boundary condition of flow tangency. Then for both types of problems the flow region is truncated and inflow and outflow boundary conditions are imposed; we shall limit ourselves here to subsonic inflow and outflow conditions. At inflow there are two ingoing characteristics requiring two boundary conditions which we choose to impose by specifying the tangential velocity and the entropy. At outflow only one boundary condition is needed and we specify the outflow pressure. For details see Section 6.

3. Finite Volume Schemes

In the finite volume formulation, the goal of steady state is identified with the net flux into a finite volume cell being equal to zero. In smooth flow regions it is equivalent to the integration of the differential equations in conservation form over an arbitrary cell Ω with boundary $\partial\Omega$ to give

$$\iint_{\Omega} [\mathbf{f}(\mathbf{u})_x + \mathbf{g}(\mathbf{u})_y]\, dx\, dy = 0. \tag{3.1}$$

Here $\mathbf{f}$ and $\mathbf{g}$ are the flux vectors in the x- and y-directions, respectively: for the H-system they and $\mathbf{u}$ have three components defined by writing (2.5) as $\mathbf{u}_t + \mathbf{f}_x + \mathbf{g}_y = 0$. Applying the divergence theorem we obtain the boundary integral

$$\int_{\partial\Omega} [\mathbf{f}(\mathbf{u})\, dy - \mathbf{g}(\mathbf{u})\, dx] = 0. \tag{3.2}$$

For the discrete version, where we now work with $\mathbf{U}$, the approximation to $\mathbf{u}$, the boundary integral (3.2) is replaced by the sum over the four cell sides (of a quadrilateral),

$$\sum_{i=1}^{4} [\mathbf{f}(\mathbf{U})|_i\, \Delta y_i - \mathbf{g}(\mathbf{U})|_i\, \Delta x_i] = 0, \tag{3.3}$$

where $(\Delta x_i, \Delta y_i)$ defines the orientation of side i, and $\mathbf{f}(\mathbf{U})|_i$, $\mathbf{g}(\mathbf{U})|_i$ are flux functions considered as averages along it. Hence $\mathbf{f}(\mathbf{U})|_i\, \Delta y_i - \mathbf{g}(\mathbf{U})|_i\, \Delta x_i$ is the normal flux through that side.

The key discretisation decision is how these averages are to be expressed in terms of $\mathbf{U}$. The two most obvious choices are to keep $\mathbf{U}$ at the centres of cells, as in Jameson [5], and average *across* the cell sides, or to keep $\mathbf{U}$ at the cell vertices, following Ni [13], and average *along* the cell sides. We consider these two below and decide firmly in favour of the latter.

3.1. *Cell Centres or Cell Vertices*

First, consider keeping $\mathbf{U}$ at the cell centres. With reference to Fig. 1a, we construct the discrete steady spatial residual (3.3) for an arbitrary cell by averaging

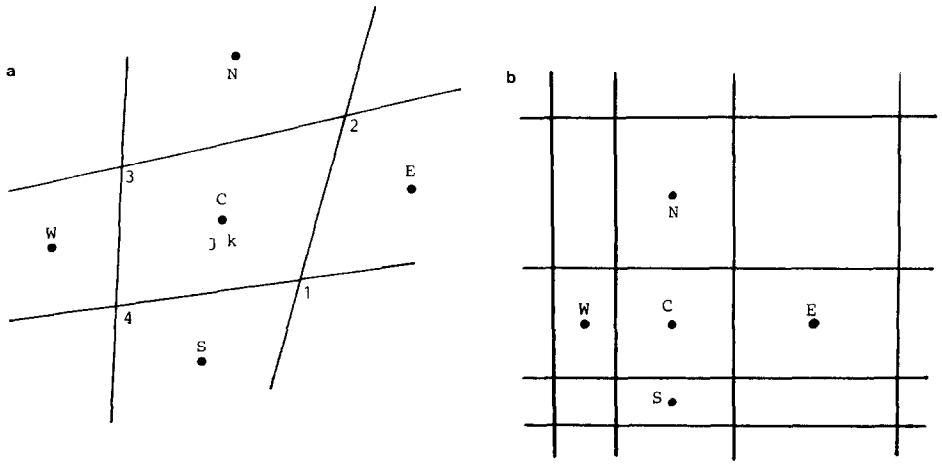

FIG. 1. Geometry for flow variables at cell centres: (a) general quadrilateral mesh; (b) non-uniform rectangular mesh.

$\mathbf{U}$ across cell boundaries. Dividing by the cell area V_C, denoting the residual by $\mathbf{R}_C(\mathbf{U})$, and writing $\mathbf{f}_C$ for $\mathbf{f}(\mathbf{U})_C$, etc., we have

$$\begin{aligned}\mathbf{R}_C(\mathbf{U}) = \frac{1}{V_C}\Big[&\frac{1}{2}(\mathbf{f}_C+\mathbf{f}_E)(y_2-y_1)-\frac{1}{2}(\mathbf{g}_C+\mathbf{g}_E)(x_2-x_1)\\ &+\frac{1}{2}(\mathbf{f}_C+\mathbf{f}_N)(y_3-y_2)-\frac{1}{2}(\mathbf{g}_C+\mathbf{g}_N)(x_3-x_2)\\ &+\frac{1}{2}(\mathbf{f}_C+\mathbf{f}_W)(y_4-y_3)-\frac{1}{2}(\mathbf{g}_C+\mathbf{g}_W)(x_4-x_3)\\ &+\frac{1}{2}(\mathbf{f}_C+\mathbf{f}_S)(y_1-y_4)-\frac{1}{2}(\mathbf{g}_C+\mathbf{g}_S)(x_1-x_4)\Big]\\ =\frac{1}{2V_C}\,&[\mathbf{f}_E(y_2-y_1)-\mathbf{f}_W(y_3-y_4)-\mathbf{f}_N(y_2-y_3)+\mathbf{f}_S(y_1-y_4)\\ &-\mathbf{g}_E(x_2-x_1)+\mathbf{g}_W(x_3-x_4)+\mathbf{g}_N(x_2-x_3)-\mathbf{g}_S(x_1-x_4)]. \qquad (3.4)\end{aligned}$$

For a uniform rectangular mesh of cell dimensions Δx, Δy, this reduces to

$$\mathbf{R}_{jk}(\mathbf{U}) = \left(\frac{\mathbf{f}_E-\mathbf{f}_W}{2\Delta x}+\frac{\mathbf{g}_N-\mathbf{g}_S}{2\Delta y}\right). \qquad (3.5)$$

the familiar central differencing in both x- and y-directions.

Alternatively we can keep $\mathbf{U}$ at the cell vertices, as in Fig. 2a. Again constructing the residual (3.3), and denoting it by $\mathbf{R}_C$, we average $\mathbf{U}$ along the cell boundaries to give

$$\begin{aligned}\mathbf{R}_C(\mathbf{U}) = \frac{1}{V_C}\Big[&\frac{1}{2}(\mathbf{f}_1+\mathbf{f}_2)(y_2-y_1)-\frac{1}{2}(\mathbf{g}_1+\mathbf{g}_2)(x_2-x_1)\\ &+\frac{1}{2}(\mathbf{f}_2+\mathbf{f}_3)(y_3-y_2)-\frac{1}{2}(\mathbf{g}_2+\mathbf{g}_3)(x_3-x_2)\\ &+\frac{1}{2}(\mathbf{f}_3+\mathbf{f}_4)(y_4-y_3)-\frac{1}{2}(\mathbf{g}_3+\mathbf{g}_4)(x_4-x_3)\\ &+\frac{1}{2}(\mathbf{f}_4+\mathbf{f}_1)(y_1-y_4)-\frac{1}{2}(\mathbf{g}_4+\mathbf{g}_1)(x_1-x_4)\Big]\\ =\frac{1}{2V_C}\,&[(\mathbf{f}_1-\mathbf{f}_3)(y_2-y_4)+(\mathbf{f}_2-\mathbf{f}_4)(y_3-y_1)\\ &-(\mathbf{g}_1-\mathbf{g}_3)(x_2-x_4)-(\mathbf{g}_2-\mathbf{g}_4)(x_3-x_1)]. \qquad (3.6)\end{aligned}$$

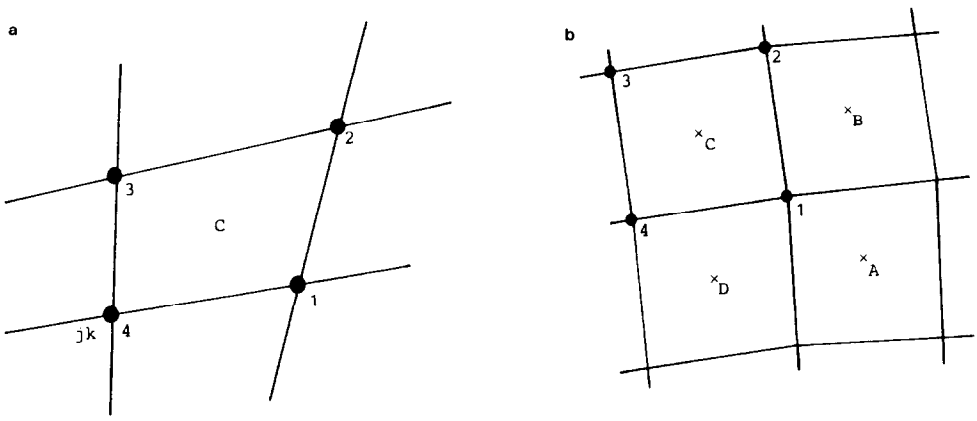

FIG. 2. Geometry for flow variables at cell vertices: (a) the vertices around cell C; (b) the cells used to update vertex 1.

This corresponds to using the trapezoidal rule to approximate the integrals along each of the cell sides. On a uniform rectangular mesh this can be rewritten as the box-scheme average of differences

$$\mathbf{R}_{j+1/2,k+1/2}(\mathbf{U}) = \frac{1}{2}\left(\frac{\mathbf{f}_2-\mathbf{f}_3}{\Delta x}+\frac{\mathbf{f}_1-\mathbf{f}_4}{\Delta x}\right) + \frac{1}{2}\left(\frac{\mathbf{g}_2-\mathbf{g}_1}{\Delta y}+\frac{\mathbf{g}_3-\mathbf{g}_4}{\Delta y}\right). \tag{3.7}$$

The objective, in computing an aproximate steady solution by means of some iterative process, is to make these spatial residuals (3.4) or (3.6) close to zero. In what follows we look at the principal differences in the character of the approximation obtained by setting (3.4) and (3.6) to zero—in particular, the accuracy on non-uniform meshes and the tendency to exhibit spurious solution modes.

3.2. *Accuracy on Non-uniform Meshes*

On a uniform rectangular mesh the truncation error of either scheme, obtained by substituting the true solution into (3.5) or (3.7) and expanding in Taylor series, is second order in the mesh spacing. Moreover, the operators that need to be inverted in order to relate the truncation error to the error in the solution have a similar form: for (3.5) has the form $AD_{0x}\mathbf{U} + BD_{0y}\mathbf{U}$, where A and B are the Jacobian matrices of $\mathbf{f}$ and $\mathbf{g}$ at some point and D_{0x}, D_{0y} the central divided difference operators. Whilc grouping the flux terms in (3.7) at each mesh point shows that it involves two diagonal central differences to the fluxes corresponding to the rotated co-ordinates, namely $(\Delta y\mathbf{f} \pm \Delta x\mathbf{g})/(\Delta x^2 + \Delta y^2)^{1/2}$. Thus we are encouraged to base a comparison of the errors in the two schemes on a comparison of their truncation

errors; we shall continue to use this in the case of non-uniform meshes. A similar approach has been used by Roe [21] and Turkel [24].

For the cell vertex scheme, we know that the trapezoidal rule gives

$$\int_a^b F(z)\,dz = \frac{1}{2}(F_a + F_b)(b-a) - \frac{1}{12}(b-a)^3 F''_{ab}, \tag{3.8}$$

where F''_{ab} is the value of the second derivative of F in the direction of and somewhere along the line ab. Hence, after substituting $\mathbf{u}$ for $\mathbf{U}$ in (3.6) and using (3.2), we have for pairs of opposite sides

$$\begin{aligned}\mathbf{R}_C(\mathbf{u}) &= \frac{1}{V_C}\left\{\frac{1}{12}(y_2 - y_1)^3\,\mathbf{f}''_{12} + \frac{1}{12}(y_4 - y_3)^3\,\mathbf{f}''_{34}\right.\\ &\qquad \left. + \text{three similar pairs of terms}\right\}\\ &= \frac{1}{12V_C}\left\{((y_2 - y_1)^3 - (y_3 - y_4)^3)\,\mathbf{f}''_{12}\right.\\ &\qquad \left. + (y_4 - y_3)^3(\mathbf{f}''_{34} - \mathbf{f}''_{12}) + \cdots\right\}.\end{aligned}$$

If the functions $\mathbf{f}$, $\mathbf{g}$ have continuous second derivatives, then the approximation retains second-order accuracy provided the directions of opposite sides differ by $O(h)$ and

$$(y_2 - y_1)^3 - (y_3 - y_4)^3 = O(h^4)$$

or

$$(y_2 - y_1)/(y_3 - y_4) = 1 + O(h), \tag{3.9}$$

where h characterises the mesh spacing, that is, opposite side lengths should be in a ratio of $1 + O(h)$. In general, this means that the cells must be parallelograms to within $O(h)$. Even when the cells are distorted such that these ratios deteriorate to $1 + O(1)$, the accuracy will still be first order.

The cell centre method is more difficult to analyse in two dimensions, so for simplicity we assume the computational mesh to be rectangular, as in Fig. 1b. The residual is given by (3.5) as

$$\mathbf{R}_C = \frac{\mathbf{f}_E - \mathbf{f}_W}{2\Delta x_C} + \frac{\mathbf{g}_N - \mathbf{g}_S}{2\Delta y_C}, \tag{3.10}$$

where Δx_C, Δy_C are the dimensions of cell C. Expanding about the cell centre, the leading terms in the truncation error are therefore

$$\mathbf{R}_C(\mathbf{u}) = \frac{\Delta x_E + 2\Delta x_C + \Delta x_W}{4\Delta x_C} \mathbf{f}_x|_C + \frac{\Delta y_N + 2\Delta y_C + \Delta y_S}{4\Delta y_C} \mathbf{g}_y|_C + \cdots,$$

where Δx_E is the width of the cell to the right, and so on; since $\mathbf{f}_x + \mathbf{g}_y = 0$, this implies that we need, for first-order accuracy,

$$\frac{\Delta x_E + 2\Delta x_C + \Delta x_W}{4\Delta x_C} - \frac{\Delta y_N + 2\Delta y_C + \Delta y_S}{4\Delta y_C} = O(h). \tag{3.11}$$

In general, this means that each expression should be $1 + O(h)$ so that we require successive mesh lengths to be in a ratio of $1 + O(h)$ even for first-order accuracy; for second-order accuracy they would need to be in a ratio of $1 + O(h^2)$, in general.

Some indication of the effect of a non-rectangular mesh is given from the analysis by Paisley [14] of the case of a uniform rectangular cell with one corner displaced by amounts ε, δ in the x, y directions, respectively. It was shown that for first- and second-order accuracy ε, δ had to be $O(h^2)$, $O(h^3)$, respectively, in agreement with the above.

Thus, the accuracy of the cell vertex scheme clearly shows a greater resilience to distortions in the mesh than does the corresponding cell centre scheme. Of course, as is widely appreciated, the basic flaw with the cell-centred scheme as in (3.4) on an uneven mesh is the evaluation of the flux on a cell boundary as a simple average of the values in neighbouring cells. On a non-uniform mesh this average is not centred at that boundary and immediately errors are introduced. In principle it would be possible to restore second-order accuracy by centering the averages properly via an appropriate weighting based on cell dimensions. However, as Turkel [24] and Rizzi [20] acknowledge and as discussed in Paisley [14], the Runge–Kutta iteration is now more prone to instability and convergence is more difficult to achieve.

From this discussion it is clear that the cell centre method as it stands must be adapted if it is to match the accuracy of cell vertex method on a mesh of a given non-uniformity. Whether the extra labour involved would be justified depends on other features of both implementations, and it is to these that we now turn.

3.3. *Spurious Solution Modes*

It is well known that if we use a difference operator involving more than the minimum number of points necessary to approximate derivatives, then the solution to the resulting difference equations is not unique. That is, spurious solution modes are present which, if not recognised and dealt with, can seriously pollute and corrupt the solution.

Since the central differences in the cell centre residual (3.5) are three-point approximations to first-order derivatives, even the one-dimensional scheme will be prone to spurious oscillations in the converged solution. By contrast, the compact

cell vertex residual (3.7) contains only a two-point difference and in one dimension the steady solution can have no spurious oscillations present.

In two dimensions, however, both the cell vertex and cell centre residuals admit oscillatory solutions. Consider the scalar linear equation

$$u_t + u_x + u_y = 0 \tag{3.12}$$

in the quarter-plane, with appropriate boundary conditions. Discretising on a uniform rectangular mesh, the cell centre residual for the steady part of (3.12) is

$$R_{jk} = \frac{U_{j+1k} - U_{j-1k}}{2\Delta x} + \frac{U_{jk+1} - U_{jk-1}}{2\Delta y} \tag{3.13}$$

and the vertex residual is

$$R_{j+1/2,k+1/2} = \frac{1}{2}\left(\frac{U_{j+1k+1} - U_{jk+1}}{\Delta x} + \frac{U_{j+1k} - U_{jk}}{\Delta x}\right) + \frac{1}{2}\left(\frac{U_{j+1k+1} - U_{j+1k}}{\Delta y} + \frac{U_{jk+1} - U_{jk}}{\Delta j}\right). \tag{3.14}$$

If the discrete version of (3.12) is iterated until the steady state, then setting these residuals to zero defines the converged solutions. Looking for modes of the form $U_{jk} = \hat{U}\mu^j\nu^k$, we have for the cell centre and cell vertex schemes, respectively,

$$\left(\frac{1}{2\Delta x}(\mu^2 - 1)\,\nu + \frac{1}{2\Delta y}\mu(\nu^2 - 1)\right)\hat{U}\mu^{j-1}\nu^{k-1} = 0 \tag{3.15a}$$

$$\left(\frac{1}{2\Delta x}(\mu - 1)(\nu + 1) + \frac{1}{2\Delta y}(\mu + 1)(\nu - 1)\right)\hat{U}\mu^j\nu^k = 0 \tag{3.15b}$$

and it is seen that if $\mu = \nu$ (the modes are travelling diagonally) these expressions are identical, apart from the extra factor μ or ν in the former. In particular, both residual equations are satisfied by oscillatory solutions of the form $\mu = \nu = -1$. There is still a difference, however; for (3.15a) is quadratic in μ for a fixed ν, while (3.15b) is only linear. Thus for the true mode $\nu = 1 + O(h)$, (3.15a) gives not only the true mode $\mu = 1 + O(h)$, but also the spurious mode $\mu = -1 + O(h)$: then it gives two spurious modes for $\nu = -1$. Thus, in total, the cell-centered scheme gives three distinct spurious modes corresponding to

$$\begin{array}{ccc} + - + & + + + & + - + \\ + - + & - - - & - + - \\ + - + & + + + & + - + \\ \mu = -1,\ \nu = 1 & \mu = 1,\ \nu = -1 & \mu = \nu = -1. \end{array} \tag{3.16a}$$

On the other hand, for the cell vertex residual there is only the one chequer-board mode

$$\begin{matrix} + & - \\ - & + \end{matrix} \qquad \mu = \nu = -1. \tag{3.16b}$$

Modes such as these can be triggered at shocks, boundaries, or even non-uniformities in the mesh and usually have to be damped out. The smaller number of modes in the cell vertex scheme is a distinct advantage.

3.4. *Solution of* $\mathbf{R}_C(\mathbf{U}) = 0$

The spurious modes considered in the previous subsection are those remaining to pollute the approximation when all the residuals have been set sufficiently close to zero. To achieve this in the case of either (3.4) or (3.6) is a major task, to which the bulk of the computation is directed. Some form of iteration is called for; for the structured grids that we have in mind, a multigrid acceleration is highly desirable. The design and analysis of suitable multigrid processes has received considerable attention recently and we shall not go into any details here. What is required in each case is a suitable smoother, which could be used as an iterative method even without multigrid acceleration: its form may well affect the quality of the eventual approximation, which is our main concern in this paper, so we will consider this next.

4. Time-Stepping Iteration

By far the most popular form of iteration or smoother, is that based on a time-stepping approximation to the unsteady equation (2.5). It is here that the cell centre approach has its main attraction, since the residual (3.4) is correctly centred for updating the value $\mathbf{U}_C$. Jameson [5] pioneered the use of Runge–Kutta methods for this purpose and the methods developed by him and his collaborators constitute the present "industry standard." However, once it has been decided how neighbouring residuals can be combined to update the values of unknowns at a vertex, the cell vertex scheme again has definite advantages. These were first realised by Ni [13] and the techniques described below derive directly from his treatment.

4.1. *Lax-Wendroff Algorithms*

In Ni [13], the iteration chosen was one of the many versions of the Lax-Wendroff method; this can be thought of as a two-step method, though it is advantageous for steady problems to consider it as one step. It is based on a Taylor series expansion in time,

$$\delta u^{n+1} := u^{n+1} - u^n = \Delta t u_t + \tfrac{1}{2}\Delta t^2 u_{tt} + O(\Delta t^3), \tag{4.1}$$

in which the time derivatives are replaced by spatial derivatives via the differential equation.

Thus from the unsteady system of differential equations,

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x + \mathbf{g}(\mathbf{u})_y = 0, \tag{4.2}$$

the second time derivative in (4.1) is replaced by spatial derivatives as

$$\begin{aligned}\mathbf{u}_{tt} &= \frac{\partial}{\partial t}(\mathbf{u}_t) = -\frac{\partial}{\partial t}(\mathbf{f}(\mathbf{u})_x + \mathbf{g}(\mathbf{u})_y) \\ &= -\frac{\partial}{\partial x}(A\mathbf{u}_t) - \frac{\partial}{\partial y}(B\mathbf{u}_t) \\ &= \frac{\partial}{\partial x}(A[\mathbf{f}(\mathbf{u})_x + \mathbf{g}(\mathbf{u})_y]) + \frac{\partial}{\partial y}(B[\mathbf{f}(\mathbf{u})_x + \mathbf{g}(\mathbf{u})_y]),\end{aligned} \tag{4.3}$$

where A, B are the Jacobian matrices $\partial\mathbf{f}/\partial\mathbf{u}$, $\partial\mathbf{g}/\partial\mathbf{u}$, respectively. This now gives the change in the solution as

$$\delta\mathbf{U}^{n+1} = -\Delta t(\mathbf{f}_x + \mathbf{g}_y)^n + \frac{1}{2}\Delta t^2\left[\frac{\partial}{\partial x}(A(\mathbf{f}_x + \mathbf{g}_y)) + \frac{\partial}{\partial y}(B(\mathbf{f}_x + \mathbf{g}_y))\right]^n. \tag{4.4}$$

With reference to Fig. 2b, suppose we are calculating the change at point 1; then the first term in (4.4) is given by an average of residuals (3.6) in the neighbouring cells. The straightforward average

$$-\Delta t\, \tfrac{1}{4}(\mathbf{R}_A + \mathbf{R}_B + \mathbf{R}_C + \mathbf{R}_D) \tag{4.5}$$

as recommended by Ni gives a divergent iteration on appreciably non-uniform meshes unless excessive damping is used, but Hall's [4] area-weighted average

$$-\Delta t\, \frac{V_A\mathbf{R}_A + V_B\mathbf{R}_B + V_C\mathbf{R}_C + V_D\mathbf{R}_D}{V_A + V_B + V_C + V_D} \tag{4.6}$$

changes the convergence properties, giving much better behaviour.

On a general mesh the average in (4.6) corresponds to the boundary integral around the perimeter of the group of four cells, ensuring conservation is maintained during the transient phase. It seems, however, that this latter property is incidental to the improved performance of this averaging: what is important is that the values at the central vertex are eliminated. The effect is clear even in one dimension. Suppose the current approximation at three points has values U_0^n in the centre, U_+^n on the right and U_-^n on the left, with intervals h_+ and h_- between the points, respectively. Then we can write the Lax–Wendroff update with simple averaging, and including the second-order terms, as

$$U_0^{n+1} = U_0^n - \tfrac{1}{2}\Delta t(R_+ + R_-) + (\Delta t)^2(a_+ R_+ - a_- R_-)/(h_+ + h_-), \tag{4.7a}$$

where

$$R_+ = (f_+^n - f_0^n)/h_+, \qquad R_- = (f_0^n - f_-^n)/h_-, \qquad a_\pm = (f_\pm^n - f_0^n)/(U_\pm^n - U_0^n). \tag{4.7b}$$

Writing $v_\pm = a_\pm \Delta t/h_\pm$, the coefficient of U_0^n is given by

$$U_0^{n+1} = \left[1 + \frac{1}{2}v_+\left(1 - \frac{2h_+ v_+}{h_+ + h_-}\right) - \frac{1}{2}v_-\left(1 + \frac{2h_- v_-}{h_+ + h_-}\right)\right] U_0^n + \cdots, \tag{4.8}$$

and the condition to ensure that this is less than unity is

$$\frac{a_+}{h_+} - \frac{a_-}{h_-} < \frac{2(a_+ v_+ + a_- v_-)}{h_+ + h_-}. \tag{4.9}$$

The bound on the right is positive and increases with Δt; but if the characteristic speeds are positive and the mesh decreases too fast from left to right or the characteristic speed increases too fast, the error at the central point might be magnified instead of damped. On the other hand, with the averaging of (4.6) we have $h_+ R_+ + h_- R_- = f_+^n - f_-^n$ and only the second-order damping term contributes to the coefficient of U_0^n, giving

$$U_0^{n+1} = \left[1 - \frac{h_+ v_+^2 + h_- v_-^2}{h_+ + h_-}\right] U_0^n + \cdots. \tag{4.10}$$

Thus an isolated error at the central point is reduced so long as $v^2 < 1$ everywhere. A more thorough analysis, based on a discrete energy method, confirms the convergence properties of this scheme.

Returning to two dimensions, the second term of (4.4) is in the form of a divergence and again can be cast as a boundary integral, this time around the inner quadrilateral $ABCD$. The flux functions at the cell centres are now given by the product of the respective Jacobian matrix (where all the values of the conserved variables needed for the matrix entries are evaluated at the cell centres as averages of the vertex values) with the residual vector for that cell. This gives for the second term (cf. (3.6)),

$$\frac{\Delta t^2}{4V_1}[((A\mathbf{R})_A - (A\mathbf{R})_C)(y_B - y_D) + ((A\mathbf{R})_B - (A\mathbf{R})_D)(y_C - y_A)$$
$$- ((B\mathbf{R})_A - (B\mathbf{R})_C)(x_B - x_D) - ((B\mathbf{R})_B - (B\mathbf{R})_D)(x_C - x_A)], \tag{4.11}$$

where V_1 is the area of the quadrilateral formed by the centres of the four cells. The total change at a point is thus given by a weighted average of neighbouring residuals, with relative weights dependent on the geometry and the Jacobians. That is, (4.6) and (4.11) are combined to give

$$\delta\mathbf{U}^{n+1} = -\tfrac{1}{4}\Delta t[D_A\mathbf{R}_A + D_B\mathbf{R}_B + D_C\mathbf{R}_C + D_D\mathbf{R}_D], \tag{4.12}$$

where D_Ω are the "distribution" matrices

$$D_A = \frac{4V_A}{\Sigma V_\Omega} I - \frac{\Delta t}{V_1}(y_B - y_D) A_A + \frac{\Delta t}{V_1}(x_B - x_D) B_A$$

$$D_B = \frac{4V_B}{\Sigma V_\Omega} I - \frac{\Delta t}{V_1}(y_C - y_A) A_B + \frac{\Delta t}{V_1}(x_C - x_A) B_B$$

$$D_C = \frac{4V_C}{\Sigma V_\Omega} I + \frac{\Delta t}{V_1}(y_B - y_D) A_C - \frac{\Delta t}{V_1}(x_B - x_D) B_C$$

$$D_D = \frac{4V_D}{\Sigma V_\Omega} I + \frac{\Delta t}{V_1}(y_C - y_A) A_D - \frac{\Delta t}{V_1}(x_C - x_A) B_D.$$

An important feature of this one-step formulation is that at convergence, when $\delta U^{n+1} \to 0$, the weighted average of the cell residuals in (4.12) also tends to zero. This is crucial to showing that individual residuals tend to zero under appropriate boundary conditions—see below.

The evaluation of the Jacobians necessary for the one-step form can be avoided by casting the iteration in a two-step form as is often done for unsteady problems, for example, see Richtmyer [18] or Richtmyer and Morton [19]. Several ways of doing this for the present equations are presented in Johnson [7]. Of the many variants, the simplest is to predict values at cell centres (denoted here by *) and then obtain corrected values at the vertices. That is,

$$\mathbf{U}_C^* = \tfrac{1}{4}(\mathbf{U}_1^n + \mathbf{U}_2^n + \mathbf{U}_3^n + \mathbf{U}_4^n) - \tfrac{1}{2}\Delta t \mathbf{R}_C \tag{4.13a}$$

$$\mathbf{U}_1^{n+1} = \mathbf{U}_1^n - \Delta t \mathbf{R}_1^*, \tag{4.13b}$$

where $\mathbf{R}_1^*$ is evaluated from $\mathbf{U}_A^*$, $\mathbf{U}_B^*$, $\mathbf{U}_C^*$, and $\mathbf{U}_D^*$. For convergence, however, we must now have $\mathbf{R}_j^* \to 0\ \forall j$ and this does not necessarily imply that $\mathbf{R}_\Omega^n \to 0\ \forall \Omega$. Thus if the two-step form is used the residual formed from average values in neighbouring cells is set to zero, rather than what is actually required—the residual for each individual cell.

4.2. *Choice of Time-Step*

For the limits on Δt to ensure convergence, and the optimum local choice, we consider first the two-dimensional scalar wave equation, with a and b constant,

$$u_t + a u_x + b u_y = 0 \tag{4.14a}$$

on a uniform rectangular mesh : we have

$$\delta U^{n+1} = -\tfrac{1}{4}\Delta t[(1 - v_x + v_y) R_A + (1 - v_x - v_y) R_B + (1 + v_x - v_y) R_C + (1 + v_x + v_y) R_D] \tag{4.14b}$$

with

$$R_C = \frac{1}{2}a\left(\frac{U_2 - U_3}{\Delta x} + \frac{U_1 - U_4}{\Delta x}\right) + \frac{1}{2}b\left(\frac{U_2 - U_1}{\Delta y} + \frac{U_3 - U_4}{\Delta y}\right), \text{ etc.}$$

and $v_x = a(\Delta t/\Delta x)$ and $v_y = b(\Delta t/\Delta y)$. Then performing the usual Fourier analysis, we assume the solution is of the form $U_{jk}^n = \lambda^n e^{i(j\xi + k\eta)}$, where $\xi = k_x \Delta x$ and $\eta = k_y \Delta y$ and k_x, k_y are the wave numbers in the x and y directions, respectively. The damping factor is

$$\begin{aligned}\lambda = 1 &- i(v_x \sin\xi \cos^2 \tfrac{1}{2}\eta + v_y \cos^2 \tfrac{1}{2}\xi \sin\eta) \\ &- 2v_x^2 \sin^2 \tfrac{1}{2}\xi \cos^2 \tfrac{1}{2}\eta - v_x v_y \sin\xi \sin\eta \\ &- 2v_y^2 \cos^2 \tfrac{1}{2}\xi \sin^2 \tfrac{1}{2}\eta. \end{aligned} \tag{4.15a}$$

Rearranging, using the half-angle formulae and putting $m = v_x \sin \frac{1}{2}\xi \cos \frac{1}{2}\eta + v_y \cos \frac{1}{2}\xi \sin \frac{1}{2}\eta$ gives

$$\lambda = 1 - 2im \cos \tfrac{1}{2}\xi \cos \tfrac{1}{2}\eta - 2m^2, \tag{4.15b}$$

with

$$|\lambda|^2 = 1 - 4m^2[1 - m^2 - \cos^2 \tfrac{1}{2}\xi \cos^2 \tfrac{1}{2}\eta], \tag{4.15c}$$

and for convergence we need $|\lambda|^2 < 1$ for all ξ, η in the range $0 < \xi, \eta < 2\pi$. It is clearly necessary and sufficient for this that we have $0 < m^2 < 1 - c_x^2 c_y^2$, where we write c_x for $\cos \frac{1}{2}\xi$, c_y for $\cos \frac{1}{2}\eta$ and later s_x for $\sin \frac{1}{2}\xi$, s_y for $\sin \frac{1}{2}\eta$.

It can be deduced by use of the Cauchy–Schwarz inequality that for the upper bound on m^2, which corresponds to the stability bound for the unsteady case, it is necessary and sufficient that

$$v_x^2 + v_y^2 < 1, \tag{4.16}$$

a condition observed numerically by Usab [25] and the same as that for the more standard rotated Richtmyer form of the Lax–Wendroff method. Sufficiency follows from the inequalities

$$\begin{aligned} m^2 &\leqslant (v_x^2 + v_y^2)(s_x^2 c_y^2 + s_y^2 c_x^2) \\ &= (v_x^2 + v_y^2)(s_x^2 + s_y^2 - 2s_x^2 s_y^2) \\ &\leqslant (v_x^2 + v_y^2)(s_x^2 + s_y^2 - s_x^2 s_y^2) = (v_x^2 + v_y^2)(1 - c_x^2 c_y^2); \end{aligned}$$

and necessity follows from choosing $s_x : s_y = v_x : v_y$ and letting $s_x, s_y \to 0$.

The stability or convergence limit (4.16) gives the largest Δt that can be used; for the differential equation, which has no dissipation so that the steady state is reached only by driving initial perturbations out of the domain, taking the largest

time-step possible is the optimal strategy. The Lax–Wendroff algorithm (4.14b), however, contains some dissipation which depends on Δt, and a smaller value may therefore be preferable. In one dimension, for example, $|\lambda|^2 = 1 - 4\nu^2(1-\nu^2)\,s^4$, so that one obtains maximum damping for all error modes by taking $\nu^2 = \frac{1}{2}$. Thus we consider next whether such a choice is sensible in two dimensions.

It follows from Section 3.3 that the spurious mode given there, which corresponds here to the highest frequency in both directions $\xi = \eta = \pi$, gives $\lambda = 1$ so that there is neither any damping nor any advection of it to the boundary. Furthermore, all modes with $m = 0$ by (4.15b) have the same property; that is, all those such that

$$(a/\Delta x)\tan\tfrac{1}{2}\xi + (b/\Delta y)\tan\tfrac{1}{2}\eta = 0. \tag{4.17a}$$

At low frequencies this is approximated by $ak_x + bk_y = 0$, corresponding to the wave vector being orthogonal to the direction of wave propagation. Indeed, we can classify all modes according to the value of θ, the angle between the vectors (ν_x, ν_y) and $(\tan\frac{1}{2}\xi, \tan\frac{1}{2}\eta)$, in terms of which we can write

$$m^2 = (\nu_x^2 + \nu_y^2)(1 - c_x^2 c_y^2 - s_x^2 s_y^2)\cos^2\theta. \tag{4.17b}$$

From (4.15c) we see that the damping of any mode for which $m \neq 0$ is increased as m^2 is increased to $\frac{1}{2}(1 - c_x^2 c_y^2)$ and then decreases until the stability limit $m^2 = 1 - c_x^2 c_y^2$ is reached. Thus it is only modes for which the coefficient of $\nu_x^2 + \nu_y^2$ in (4.17b) is greater than $\frac{1}{2}(1 - c_x^2 c_y^2)$ which can be better damped by decreasing the time-step. Even at low frequencies this occurs only for $\theta < \pi/4$ and at high frequencies this "cone of advantage" shrinks to zero.

Hence, even for the simple problem (4.14a), there is little chance of improving the Lax–Wendroff damping by using a time-step below the stability limit. For practical applications with the Euler equations there is even less chance, for several reasons including the following:

(i) For a system of equations there are several characteristic speeds: the time-step is limited by the fastest and, even in one dimension, reducing it may worsen the damping rate for modes corresponding to the slower speeds. For example, with speeds $u \pm a$ and $M = u/a$, the damping is made equal and optimal for both sets of modes not with $\nu_{\max}^2 = \frac{1}{2}$ but with

$$\nu_{\max}^2 = \frac{1}{2} + \frac{M}{1+M^2}.$$

Even with $M = 0.5$, this gives $\nu_{\max} = 0.95$.

(ii) For the Euler system in two dimensions, waves travel in all directions so that the scope for exploiting small values of θ in (4.17b) is further reduced. Also, because the Jacobian matrices A and B in (4.3) do not usually commute, wave modes cannot be simply separated and their damping optimised individually.

For the H-system the maximum wave speed given by (2.6) is $q + a/\gamma^{1/2}$ rather than $q + a$ for the Euler equations; so using the maximum time-step for the latter corresponds to using a reduced time-step for the former. In practice this distinction is obscured by the fact that the time-step can be varied locally to take account of local mesh lengths and the maximum permissible value is unclear when the mesh is non-rectangular. Bearing in mind all these considerations, it is perhaps not surprising that in practice we have found the fastest reliable convergence rate is obtained by using the local time-step given by Ni [13]; this is given by a geometry-independent formula using two one dimensional limits,

$$\Delta t_C < \min\left\{\frac{V_C}{|u\,\Delta y^l - v\,\Delta x^l| + c\,\Delta l}, \frac{V_C}{|u\,\Delta y^m - v\,\Delta x^m| + c\,\Delta m}\right\}, \tag{4.18}$$

where Δx^l, Δy^l, Δl, Δx^m, Δy^m, Δm depend on the cell geometry:

$$\Delta x^l = \tfrac{1}{2}(x_2 + x_3 - x_1 - x_4) \qquad \Delta x^m = \tfrac{1}{2}(x_2 + x_1 - x_3 - x_4)$$
$$\Delta y^l = \tfrac{1}{2}(y_2 + y_3 - y_1 - y_4) \qquad \Delta y^m = \tfrac{1}{2}(y_2 + y_1 - y_3 - y_4)$$
$$\Delta l = \sqrt{((\Delta x^l)^2 + (\Delta y^l)^2} \qquad \Delta m = \sqrt{((\Delta x^m)^2 + (\Delta y^m)^2)}.$$

It is also important for the arguments in the next section that the distribution matrices of (4.12) be guaranteed non-singular, and this condition ensures this.

4.3. *Boundary Conditions and Decoupling of Spurious Modes*

More spurious modes than discussed in Sections 3.3 may be supported by the discrete system if the iteration process leads to only averages of groups of residuals being set to zero. For example, we have seen in the Fourier analysis above that all modes with $m = 0$ given by (4.17) would need damping over and above that given by the Lax–Wendroff process. With certain boundary conditions and geometries, however, we can ensure that all individual cell residuals converge to zero when the update procedure (4.12) converges. Consider the flow in a rectangular channel over an obstacle, for example, a circular arc. The decoupling happens as a result of the treatment of the sides, and particularly the corners, of the domain. For a scalar equation in two dimensions,

$$u_t + f(u)_x + g(u)_y = 0,$$

the Lax–Wendroff iteration (4.12) becomes (for interior points)

$$\delta U_{jk}^{n+1} = -\tfrac{1}{4}\,\Delta t[Q_{j+1/2,k-1/2} + Q_{j+1/2,k+1/2} + Q_{j-1/2,k+1/2} + Q_{j-1/2,k-1/2}]$$
$$j = 1, 2, \ldots, J-1, \qquad k = 1, 2, \ldots, K-1, \tag{4.19}$$

where the distribution factor D has been combined with R to give Q.

THEOREM. *If the scalar Lax–Wendroff iteration* (4.19) *is used on a rectangular two-dimensional domain, then convergence implies* $R_{j-1/2,k-1/2}(U) \to 0$ $(j = 1, ..., J;$ $k = 1, ..., K)$ *provided there are two outflow boundaries, joined by a corner, where non-reflective formulae are used.*

Proof. Suppose the outflow corner is at J, K (top right-hand corner) and we have outflow along $(1, K) \cdots (J-1, K)$ (the top of the domain) and $(J, 1) \cdots (J, K-1)$ (the right boundary). Then when convergence is achieved the right-hand sides of (4.19) tend to zero at all interior points. But for outflow boundary conditions, the iteration is applied at a boundary point with all residuals referring to the exterior of the domain set to zero. Thus we also have

$$Q_{j+1/2,K-1/2} + Q_{j-1/2,K-1/2} \to 0 \qquad j = 1, ..., J-1, \tag{4.20a}$$

$$Q_{J-1/2,k+1/2} + Q_{J-1/2,k-1/2} \to 0 \qquad k = 1, ..., K-1, \tag{4.20b}$$

$$Q_{J-1/2,K-1/2} \to 0. \tag{4.20c}$$

Combining (4.20c) with (4.20a) for $j = J-1$, shows that $Q_{J-3/2,K-1/2} \to 0$; putting $k = K-1$ in (4.13b), similarly gives $Q_{J-1/2,K-3/2} \to 0$. Finally, putting $j = J-1$, $k = K-1$ in (4.19) and using these results gives $Q_{J-3/2,K-3/2} \to 0$. The result now follows by repeating this process inductively.

Such decoupling for a system of equations is more difficult to establish. Furthermore, setting all the cell residual vectors to zero does not necessarily define a unique solution so that the iteration procedure has to have other properties: for example, in the linear wave equation $\rho_t + \operatorname{div} \mathbf{u} = 0$, $\mathbf{u}_t + \operatorname{grad} \rho = 0$, the specification of the vorticity follows only from the unsteady form of the second equation. Nevertheless, by specifying only boundary variables associated with in-going characteristics and using non-reflective formulae for the remainder, we ensure as far as possible that the argument given for the scalar case still holds. This point will be taken up again for the applications in Section 6. Further details are given in Paisley [15].

Although the residuals will usually decouple in this way, the analysis of spurious modes in Section 3.3 shows that the resulting solution may still be corrupted by a chequer-board oscillation. However, when solving the two-dimensional channel problem, it is usual to specify as boundary conditions $v = 0$ at inflow and, for the top and bottom walls, a flow tangency condition, again $v = 0$. That is, for the top-left and bottom-left cells, v is specified at three of the vertices; at the fourth, only the mode of the true solution can exist, and v here must be the true value. For subcritical flows this seems, in practice, to be enough to eliminate all spurious modes from all other cells and for all three flow variables of the system. However, in the case of the aerofoil calculation, the physical boundary conditions are less constraining and this does not happen. Consequently, even for subcritical flow around aerofoils, a small amount of damping is necessary that is not needed for subcritical channel flow. For internal points, this is a nine-point formula given in Section 5.2;

for inflow and outflow boundaries, it is six-point; and along a solid boundary, it is a three-point formula.

Supercritical flow introduces two more problems. First, and most easily dealt with, is the fact that the Lax–Wendroff algorithm is not entropy satisfying, and instabilities are likely to arise around the sonic line. These can be controlled by locally applying the same damping mentioned above. The solution is smooth in such regions and no real loss of accuracy is incurred. However, a much more serious challenge is presented by the presence of a shock wave, when the residuals (3.6) are not generally accurate representations of the flow in cells containing a shock. This will be examined in more detail below.

5. Shock Recovery and Fitting

5.1. *Error in Quadrature across a Shock*

Although the boundary integral relation (3.2) holds even when the cell is crossed by a stationary shock, approximation of the line integrals by the trapezoidal rule in this case can cause substantial errors.

Consider Fig. 3 in which cell C contains a shock as shown. Suppose the shock is located such that

$$(x_Q, y_Q) = (x_R, y_R) = (x_3, y_3) + \theta_1[(x_2, y_2) - (x_3, y_3)]$$
$$(x_P, y_P) = (x_S, y_S) = (x_4, y_4) + \theta_2[(x_1, y_1) - (x_4, y_4)].$$

The standard sum over the four faces is obtained by applying the trapezoidal rule to each integral in $\int_1^2 + \int_2^3 + \int_3^4 + \int_4^1$, but taking account of the shock would require

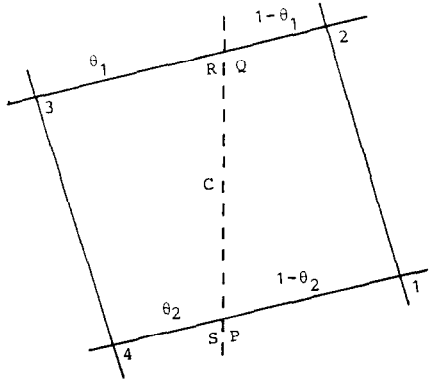

FIG. 3. Vertex scheme with a shock passing through a cell.

using $\int_1^2 + \int_2^Q + \int_R^3 + \int_3^4 + \int_4^S + \int_P^1$. The trapezoidal rule would give a good approximation to the latter, so that the error in the standard sum results from the difference

$$\int_2^Q + \int_R^3 - \int_2^3 + \int_4^S + \int_P^1 - \int_4^1. \tag{5.1}$$

For simplicity we assume constant states on either side of the shock; then the difference becomes

$$\begin{aligned} &\tfrac{1}{2}(1-2\theta_1)(y_3-y_2)[f] - \tfrac{1}{2}(1-2\theta_1)(x_3-x_2)[g] \\ &\quad + \tfrac{1}{2}(1-2\theta_2)(y_1-y_4)[f] - \tfrac{1}{2}(1-2\theta_2)(x_1-x_4)[g], \end{aligned} \tag{5.2}$$

where $[f]$, $[g]$ are the jumps in f, g. No error will be made when $\theta_1 = \theta_2 = \frac{1}{2}$, i.e., when the shock cuts the faces exactly at their midpoints, and other special cases exist when the error made on face 23 is exactly balanced by that made along face 41. In general, however, the expression in (5.2) will be $O(h)$ (giving a truncation error of $O(1/h)$ after division by the cell area) and over the length of the shock will contribute an $O(1)$ error to the solution.

In practice, the magnitude of the error is reduced by roughly aligning the mesh with the shock but it still remains the major source of inaccuracy; it is usually disguised by the addition of a non-linear dissipation which smears it over a number of neighbouring cells. However, for sensitive cases, for example, most aerofoils, this makes the position of the shock somewhat arbitrary. By adjusting the dissipation parameters the shock can be made to settle virtually anywhere on the aerofoil surface and careful tuning of the parameters together with a priori knowledge of the true position is required to give the shock in roughly the correct place. Such a captured shock is still typically spread over 3 or 4 cells—about 10% of the aerofoil chord for a typical 128×16 mesh with 96 cells around the aerofoil. If the eventual aim is to predict shockwave/boundary-layer interactions for viscous calculations then this is a very thick region indeed; the shock ought to be then compared to the boundary-layer thickness, which is typically around 0.5%. To predict narrow shock regions and correct locations (and hence lift and drag coefficients) with certainty therefore requires a very fine mesh (see Pulliam and Barton [17]).

5.2. *Outline of Shock Fitting Procedure*

An alternative to this approach is to recognise the presence of the shock and to use a fitting technique. Of the various ways of achieving this, one is to allow the shock to "float" between the fixed mesh points and to modify the difference formulae close to the shock as discussed in Richtmyer and Morton [19], Salas [23], and Moretti [9, 10]. An alternative is to consider the shock as defining a line of adjustable mesh points treated as an internal boundary as in De Neef and Moretti [12], Zhu and Chen [28], and Veuillot and Cambier [26]. This last contribution

is the only one of these which uses an underlying conservative scheme—MacCormack's method applied to the H-system (2.5), in a sub-domain approach using characteristic compatibility relations at the boundaries, including shocks. Albone's [1] technique is a hybrid approach, using a separate shock-oriented mesh which floats through the fixed underlying mesh.

The approach chosen here is based on the second alternative with double values of the flow quantities introduced along a shock line which forms part of a mesh line. In the first phase of the computation, a normal shock capturing technique is used, so that dissipative terms have to be added to the basic iteration (4.12). These can be relatively unsophisticated and, combining them with those already referred to for damping the spurious modes, leads to an additional term of the form (see Hall [4])

$$(\delta U^{n+1})_{\text{diss}} := \mu(\bar{U}^n_A + \bar{U}^n_B + \bar{U}^n_C + \bar{U}^n_D - 4U^n), \tag{5.3a}$$

where the cell values are averages over the vertices (see Fig. 2a); that is,

$$\bar{U}^n_C = \tfrac{1}{4}(U^n_1 + U^n_2 + U^n_3 + U^n_4). \tag{5.3b}$$

and the coefficient μ has th form

$$\mu = \mu_0 + \mu_1[|R_A(\rho)| + |R_B(\rho)| + |R_C(\rho)| + |R_D(\rho)|]. \tag{5.3c}$$

Here $R_C(\rho)$, etc. are the first components of the residuals (3.6), corresponding to the density, and μ_0, μ_1 are numerical parameters dependent on the geometry and the time-step. The coefficient μ_0 gives the light damping used to control spurious modes, a typical value being $\mu_0 = 0.006$. The coefficient μ_1 gives the more severe damping needed in the shock capturing phase, a typical value being $\mu_1 = 0.02$: note, however, that this term drops out if the residuals are truly zero.

During this first phase the presence of a shock is detected by scanning along the set of mesh lines which are roughly aligned with the streamlines and looking for a jump in pressure. When reasonably consistent results are obtained, the shock parameters are recovered from the local values of the variables in a manner described in more detail below. Only a limited range of shock configurations are handled by the present version of the program—namely, a shock attached to the body boundary and crossing the body-fitted mesh lines. This is then fitted by a global cubic function and a patch of the mesh containing it is adjused so that the shock itself forms part of a complementary mesh line—see Fig. 4 for an example.

In the new mesh, shocks are treated as internal boundaries and the Lax–Wendroff iteration is used to drive all the cell residuals to zero. The Rankine–Hugoniot relations are used to relate the double values introduced along the shock lines and these also determine the local shock speeds. Hence, an iteration can be set up to adjust the shock position, as described below.

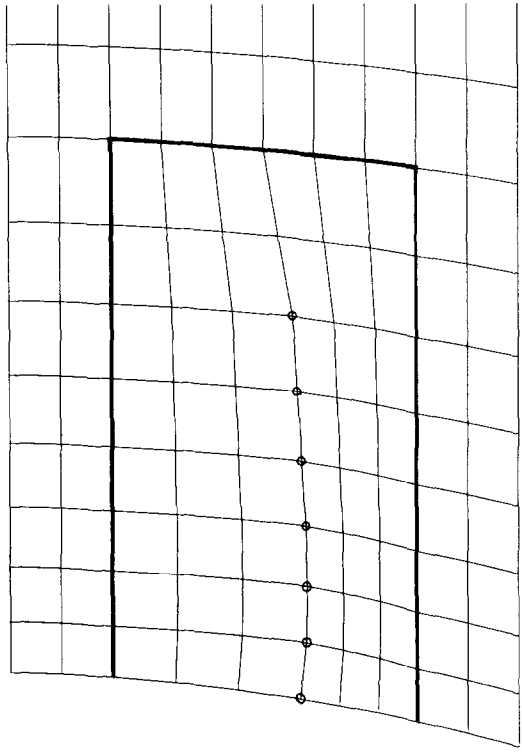

FIG. 4. A mesh patch adapted around a shock-orientated mesh-line, for channel flow over a circular arc.

5.3. *Shock Calculation*

Consider a typical shock configuration, as in Fig. 4, with the flow from left to right. In the supersonic flow ahead of the shock all three characteristics, of the equation system normal to the shock, point downstream and therefore take information into the shock. For the subsonic flow behind the shock, on the other hand, only two characteristics point downstream and that corresponding to the negative sign on the right of (2.6) points upstream and so carries information into the shock. Hence in the update process all flow quantities on the upstream side of the shock are updated in the normal way, but of course with all the integrals confined to one side of the shock, while on the downstream side only one quantity is updated in this way and two are imposed as boundary conditions. The ideal choices for these quantities would seem to be characteristic variables, but this is inconvenient when working with conserved variables. We have therefore chosen the density to be updated and the two momenta to be imposed by the Rankine–Hugoniot relations: no numerical difficulties have been encountered with this choice.

The way in which the shock calculation is carried out is as follows. If we suppose the shock has a normal velocity S, makes an angle α to the x-axis, and $U := u \sin\alpha - v\cos\alpha$, $V := u\cos\alpha + v\sin\alpha$ are the flow velocities normal and tangential to the shock, the shock relations for the full Euler equations give

$$V_L = V_R \tag{5.4a}$$

$$\frac{\rho_R}{\rho_L} = \frac{U_L - S}{U_R - S} = \frac{(\gamma+1)M_L^2}{(\gamma-1)M_L^2 + 2}, \tag{5.4b}$$

where $M_L = (U_L - S)/a_L$ and $a_L^2 = 1 - \frac{1}{2}(\gamma-1)(U_L^2 + V_L^2)$. For the H-system (2.5), rather more complicated relations hold; however, they are the same when the steady state is reached and (5.4) have proved adequate for the iteration process. All quantities on the left with subscript L are known after the update, as well as the density ρ_R. Then (5.4b) determines M_L from which we can then obtain

$$S = U_L - M_L[1 - \tfrac{1}{2}(\gamma-1)(U_L^2 + V_L^2)]^{1/2}; \tag{5.4c}$$

hence U_R and V_R can be obtained from (5.4b) and (5.4a).

This calculation is carried out at all the double-valued points on the shock, including the foot of the shock on the body where, in addition, flow tangency on the upstream side is imposed. Imposition of the condition that the shock be normal to the body then ensures the same tangency condition holds on the downstream side because of (5.4a). The line of the shock can be updated by using the shock speed, which at convergence should be everywhere zero. This is done by first calculating new x-coordinates for where the shock crosses the body-fitted mesh lines,

$$x_S' = x_S + \beta\, \Delta t S \sin\alpha, \tag{5.5}$$

where Δt is the smaller of the two time-steps used for the update of the flow variables at the shock point and β is a parameter which allows for the shock adjustment not to be made at each time-step. Then the new shock line is calculated and the surrounding mesh patch adjusted, as described below.

5.4. *Shock Detection, Recovery, and Adjustment*

After the initial shock capturing phase, the presence of a shock is detected by

supersonic at the mesh point immediately upstream of where the maximum occurs. The recovery of the shock position and its parameters can be done in a variety of ways but only a relatively crude procedure has so far been found necessary. The position on each mesh line is first found by interpolation for $\delta^2 P = 0$ between the maximum and minimum values. Then this is used to set up a local mesh patch onto which the flow variables are interpolated.

Because the shock is strongest at the body, its penetration into the field is terminated at the first mesh-line, counting from the body at which no shock is detected. The set of x-coordinates calculated as above is fitted by least squares to a global cubic constrained to be normal to the body at its foot so that we then have a shock line lying obliquely to the fixed mesh. This is used to define a mesh patch dependent on an integer parameter m, as follows: suppose the mesh-line crossing those on which the shock is detected and whose foot on the body is closest to the foot of the shock has index I_S: this is replaced by the shock line and is joined up smoothly with two points beyond its end, so defining the extent of the patch away from the body. Finally, the neighbouring mesh-lines $I_S \pm 1, \ldots, I_S \pm m$ are adjusted to give a relatively smooth spacing either side of the shock. A typical value for the parameter is $m = 3$ and a typical mesh patch is shown in Fig. 4.

Values of all flow variables are interpolated from the captured flow field onto the new mesh before double values are introduced to start the shock fitting procedure. Both the double values and the values on mesh lines $I_S \pm 1$ are presently obtained by linear extrapolation from the values on lines $I_S \pm 2$ and $I_S \pm 3$ but more elaborate procedures could easily be introduced.

The adjustment of the shock and the mesh patch, after the shock calculation described in Section 5.3, is carried out in a very similar manner. The extent of the shock may first be adjusted: if the upstream Mach number at the last shock point is recalculated to be less than 1.01, the shock is shrunk by one point since the orientation of the weak end is not well defined and slows convergence. But if the Mach number there is greater than 1.05 and that at the next point is greater than 1.01, the shock is extended by one point. The set of adjusted shock positions given by (5.5), as modified by the above process, can then be fitted by a cubic as already described. Finally, if by comparison with the original mesh the spacing in the patch has become too distorted, a grid line ahead of (or behind) the shock may need to be removed and inserted behind (or ahead of) it before the whole patch is recomputed and new values interpolated onto it.

In this way the shock is free to shrink, extend, or shift laterally as it takes up its steady position, while keeping the local mesh distortion to reasonable limits.

6. Numerical Examples and Applications

6.1. *Laval Nozzle*

It is useful to start with the one-dimensional Laval nozzle problem. If the cross-sectional area is $A(x)$ and this factor is included in the density ρ and pressure P, the equations for iso-energetic flow become

$$\rho_t + (\rho u)_x = 0 \tag{6.1a}$$

$$(\rho u)_t + (P + \rho u^2)_x - \frac{P}{A} A_x = 0, \tag{6.1b}$$

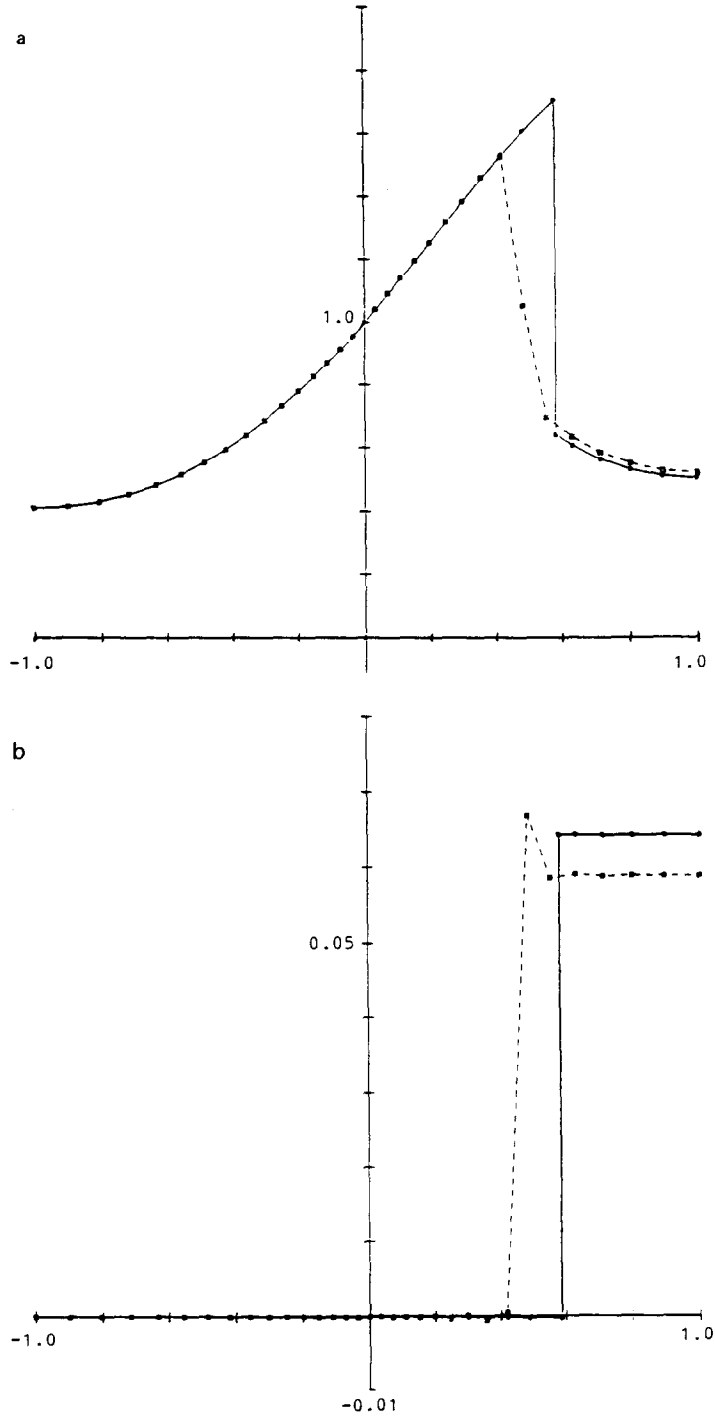

FIG. 5. (a) Mach number and (b) entropy function for nozzle flow at $M_\infty = 0.7$: ——— shock fitted, – – – shock captured.

where

$$P = \frac{\rho}{\gamma}\left(1 - \frac{1}{2}(\gamma - 1)\, u^2\right). \tag{6.1c}$$

We approximate these in an obvious specialisation of the cell-vertex scheme. For a transonic converging–diverging nozzle, both inflow and outflow are subsonic and one boundary condition needs to be imposed at each end. At inflow the density is calculated using a non-reflective (one-sided) Lax–Wendroff update and the pressure calculated from the isentropy condition $P/\rho^\gamma = (P/\rho^\gamma)_\infty$ which in our non-dimensionalisation becomes $P = (\rho/\gamma)^\gamma$. Equating this with (6.1c) then yields the velocity u and hence the updated momentum. At outflow the momentum is updated from the non-reflective Lax–Wendroff process, the exit pressure is given from the free-stream Mach number as

$$P = [1 + \tfrac{1}{2}(\gamma - 1)\, M_\infty^2]^{\gamma/(1-\gamma)} \tag{6.2}$$

and (6.1c) is used this time to give the density.

In Paisley [15] results are given for a nozzle section given by $y(x) = \pm[1.0 - 0.1(1 + \cos \pi x)]$ for $-1 \leqslant x \leqslant 1$ and for $M_\infty = 0.5$ and 0.7. On a graded mesh of 32 cells, in both these supercritical cases the shock-capturing algorithm without damping developed oscillations and diverged within a few hundred steps, after initially appearing to converge. With the addition of damping as in (5.3), and with an appropriate choice of damping parameters, a reasonable plot of the Mach number could be obtained, including approximately the correct position for the shock. On the other hand, the shock-fitting algorithm detected a shock within about 100 iterations of the initial capturing phase and then converged with residuals less than 10^{-4} in about 500 steps without multigrid. Plots of the Mach number and the entropy function $(P/\rho^\gamma)/(P/\rho^\gamma)_\infty - 1$ are shown in Fig. 5. Compared with an analytic calculation which gave the shock position at $x_S = 0.5$ and an entropy rise of 0.04861 for exit pressure 0.7603, the computation gave $x_S = 0.50031$ and downstream entropy which varied between 0.04854 and 0.04859, an accuracy of 0.1%.

6.2. *Channel Flow*

The problem of flow down a channel containing a 10% circular arc bump, as in Ni [13] and Hall [3], was used as the first test of the two-dimensional algorithm. Inflow and outflow boundary conditions were imposed as in [3]; that is, they were treated as in the nozzle problem above, except that at inflow the tangential velocity v was set to zero and at outflow both components of momentum were updated from Eqs. (4.12). The condition that the flow be tangential to the walls is enforced along the sides of the channel and is achieved by calculating non-reflective updates and determining the resulting tangential velocity component. This is then resolved in the x, y directions to give corrected values for u and v.

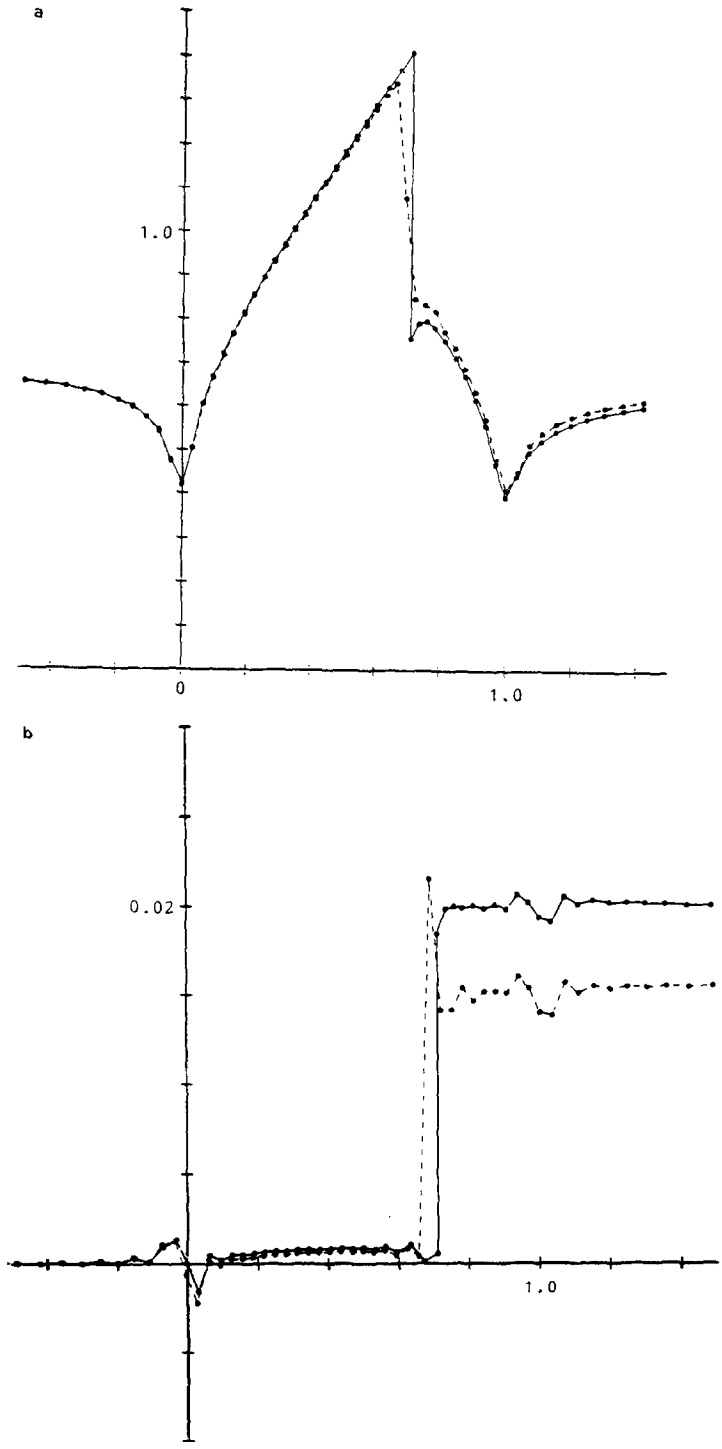

FIG. 6. (a) Mach number and (b) entropy function on the channel wall for channel flow at $M_\infty = 0.675$: ——— shock fitted, – – – shock captured.

Results are given in Fig. 6 for $M_\infty = 0.675$ and for a 64×16 mesh in which the corners at the ends of the circular arc are fared over two cells to eliminate the jump in normal direction. In Fig. 6a the Mach number clearly resolves the Zierup singularity on the body, which is completely smeared in the shock-captured approximation shown for comparison. The plots of the entropy function in Fig. 6b give a more sensitive indication of accuracy; that for the fitted shock shows a much larger rise because the shock is some 20% stronger; that for the captured shock shows the familiar spike at the shock due to the addition of artificial viscosity (see Pike [16] or the earlier reference [8] for an explanation of this non-monotonicity); both plots show oscillations at the beginning and end of the circular arc despite the faring.

6.3. *NACA 0012 Aerofoil*

Greater care needs to be taken with boundary conditions for the aerofoil problem, since the circulation generated by the aerofoil needs to be taken into account in their imposition on a finite boundary. Generally the approach used by Hall [4] has been followed. A "C"-mesh, partly shown in Fig. 7, was used with a point on the trailing edge and a cut in the physical plane along the wake, across which continuity of all the flow quantities was imposed.

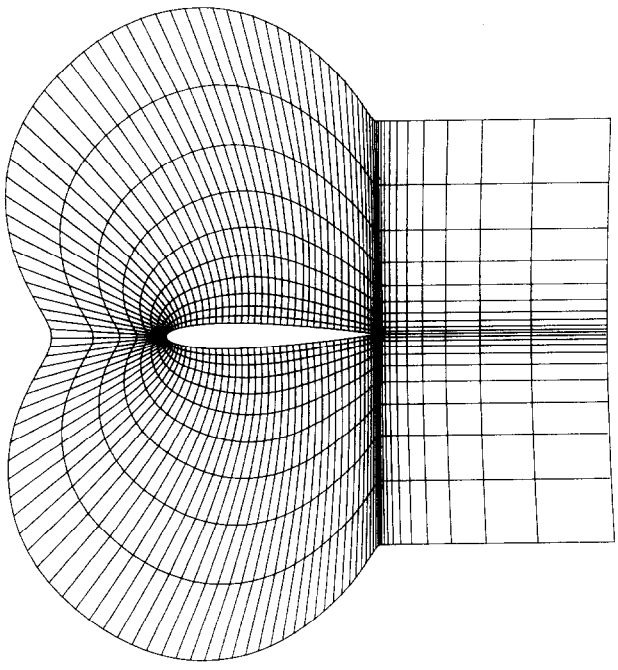

FIG. 7. Part of the coarse mesh used for the NACA 0012 aerofoil at $M_\infty = 0.8$, $\alpha = 1.25°$.

On the outer "C"-mesh line, which was treated like an inflow boundary, and on the right-hand outflow boundary the boundary conditions are dealt with as follows. An asymptotic solution given by assuming a compressible vortex centred on the aerofoil, as in Thomas and Salas [22], is given by the potential

$$\phi = q_\infty R\cos(\theta-\alpha) - \left(\frac{\Gamma}{2\pi}\right)\tan^{-1}[(1-M_\infty^2)^{1/2}\tan(\theta-\alpha)], \tag{6.3}$$

where (R, θ) are polar co-ordinates, α is the angle of attack of the aerofoil, and the circulation, which with our normalisation is $\Gamma = \frac{1}{2}M_\infty C_L$, is obtained at each iteration from calculating the lift coefficient C_L by integrating the pressure around the

$(q_\infty \cos\alpha, q_\infty \sin\alpha)$ of the flow at infinity to give the velocity (u_f, v_f) at each point on the outer mesh-line. Only the tangential component is calculated in this way: the normal component is updated from the appropriate combination of the momentum components of (4.12), with the density also obtained from (4.12). However, this then defines the Mach number M through Bernoulli's equation

$$M^2 = q^2/[1-\tfrac{1}{2}(\gamma-1)\,q^2], \tag{6.4}$$

and the isentropic relation in the form

$$\rho = \gamma[1+\tfrac{1}{2}(\gamma-1)\,M^2]^{-1/(\gamma-1)} \tag{6.5}$$

is used to impose the value of the density.

On the outflow boundary both components of the velocity are updated as above from the three components of (4.12); however, the two components of the velocity calculated as above from (6.3) define a Mach number M_f from Bernoulli's equation, and this is used to determine an exit pressure through the relation (6.2) with M_f replacing M_∞. Finally the density is determined as usual from (2.7).

The results are shown in Figs. 9–12 for the AGARD test cases $M_\infty = 0.8$, $\alpha = 1.25°$ and $M_\infty = 0.85$, $\alpha = 1°$. The two shocks on the aerofoil for the latter case make it particularly severe, the lift coefficient being very sensitive to their relative positions. For both cases plots of Mach number and entropy function are shown for a coarse mesh and a fine mesh shown in Fig. 7 and Fig. 8, respectively. Note that for the $M_\infty = 0.8$ case the shock is well aligned with the original mesh, and little distortion is seen. This is not so for the $M_\infty = 0.85$ case, where the strong upper shock lies rather more obliquely.

As for the channel results, these were produced using multigrid acceleration, with the convergence criterion being the average relative change in density

$$(\rho^{n+1}-\rho^n)/\Delta t$$

reaching 10^{-7} in the field, with a maximum value on the shock of 10^{-5}. Details of the multigrid procedure as adapted for shock fitting will be given in a further paper.

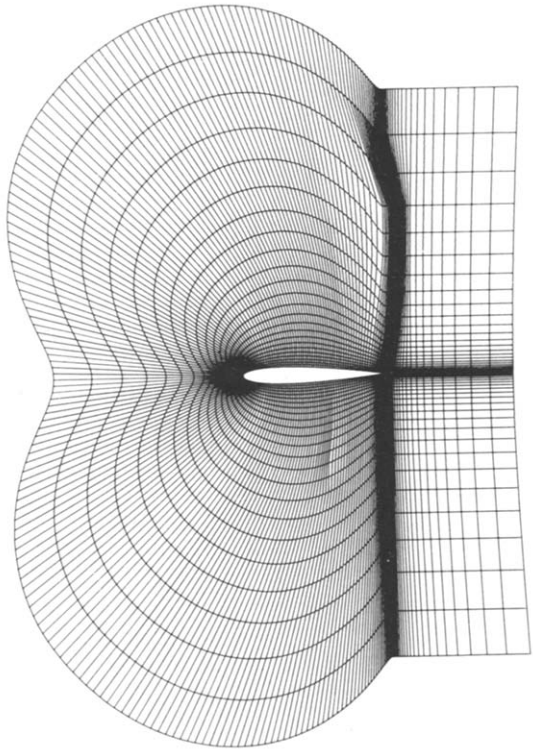

FIG. 8. Part of the fine mesh used for the NACA 0012 aerofoil at $M_\infty = 0.85$, $\alpha = 1°$.

The lift coefficients are summarised in the following table, comparison being made with the shock capturing results of Hall [4] and those of Pulliam and Barton [17] on a 560×65 mesh.

		C_L		
Case	Mesh	Fitted	Captured	[17]
$M_\infty = 0.8$	128×16	0.3745	0.3604	0.3618
$\alpha = 1.25$	256×32	0.3710	0.3598	
$M_\infty = 0.85$	128×16	0.4217	0.3823	0.3938
$\alpha = 1.0$	256×32	0.4138	0.3946	

It is seen that the shock fitting method produces markedly higher lift than the corresponding shock capturing methods. For the relatively straightforward $M_\infty = 0.8$ case this difference is around $2\frac{1}{2}\%$, while for the other case it is rather more and, as expected, the fitted shocks are generally further downstream than the captured ones.

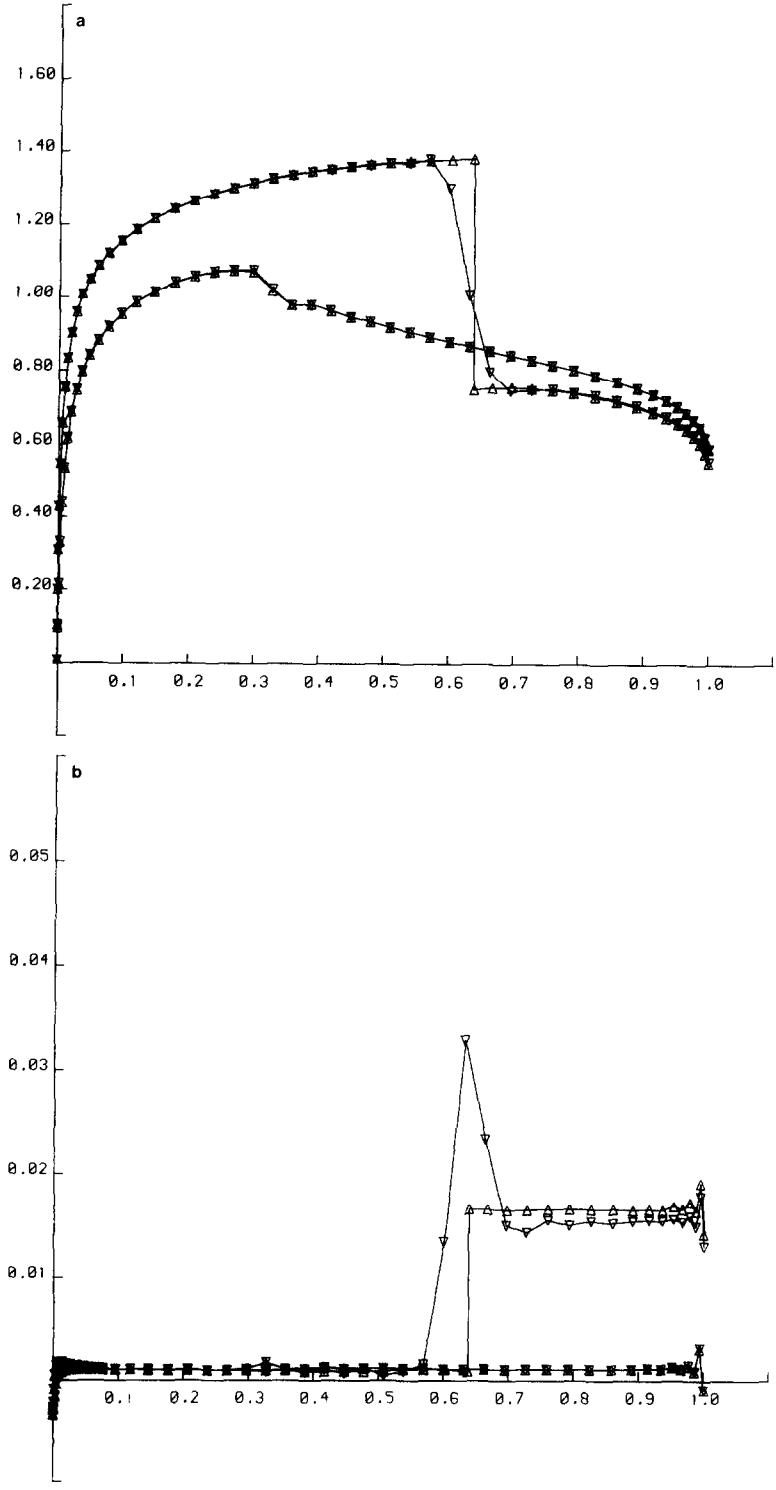

FIG. 9. As in Fig. 6 for the NACA 0012 aerofoil at $M_\infty = 0.8$, $\alpha = 1.25°$ on the coarse mesh of Fig. 7: $\triangle$ = shock fitted, ∇ = shock captured.

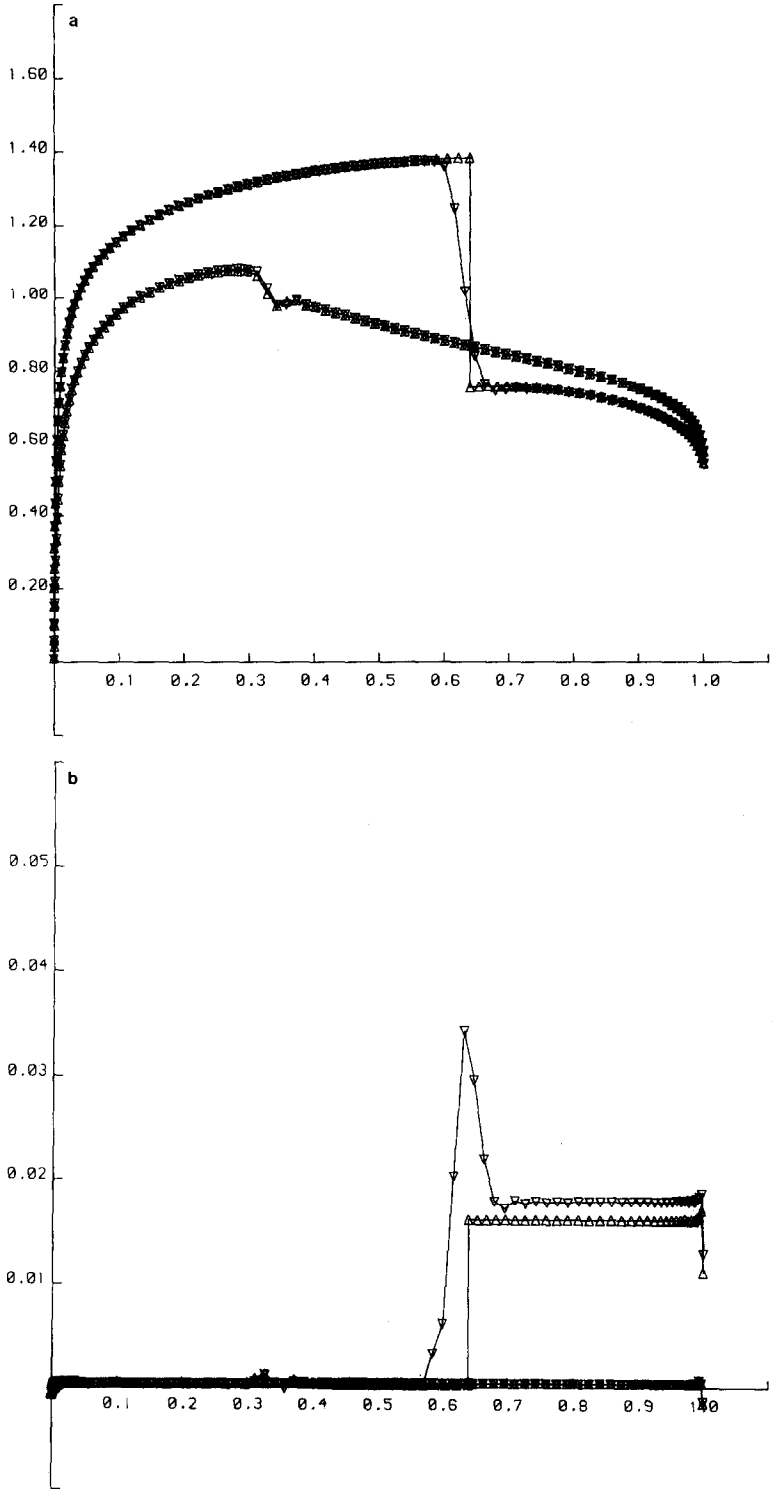

FIG. 10. As in Fig. 9 on the fine mesh.

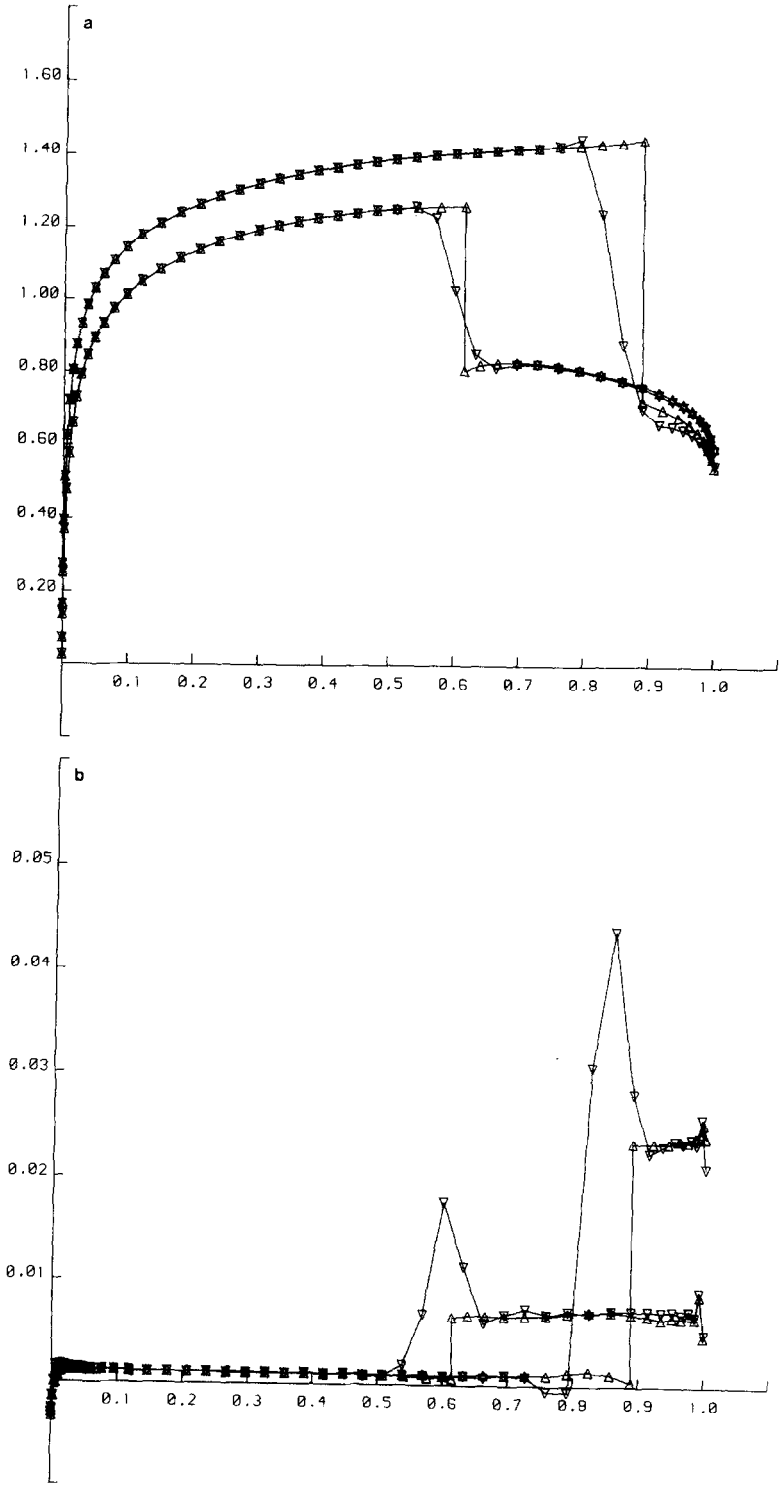

FIG. 11. As in Fig. 9 for the NACA 0012 aerofoil at $M_\infty = 0.85$, $\alpha = 1^\circ$ on the coarse mesh.

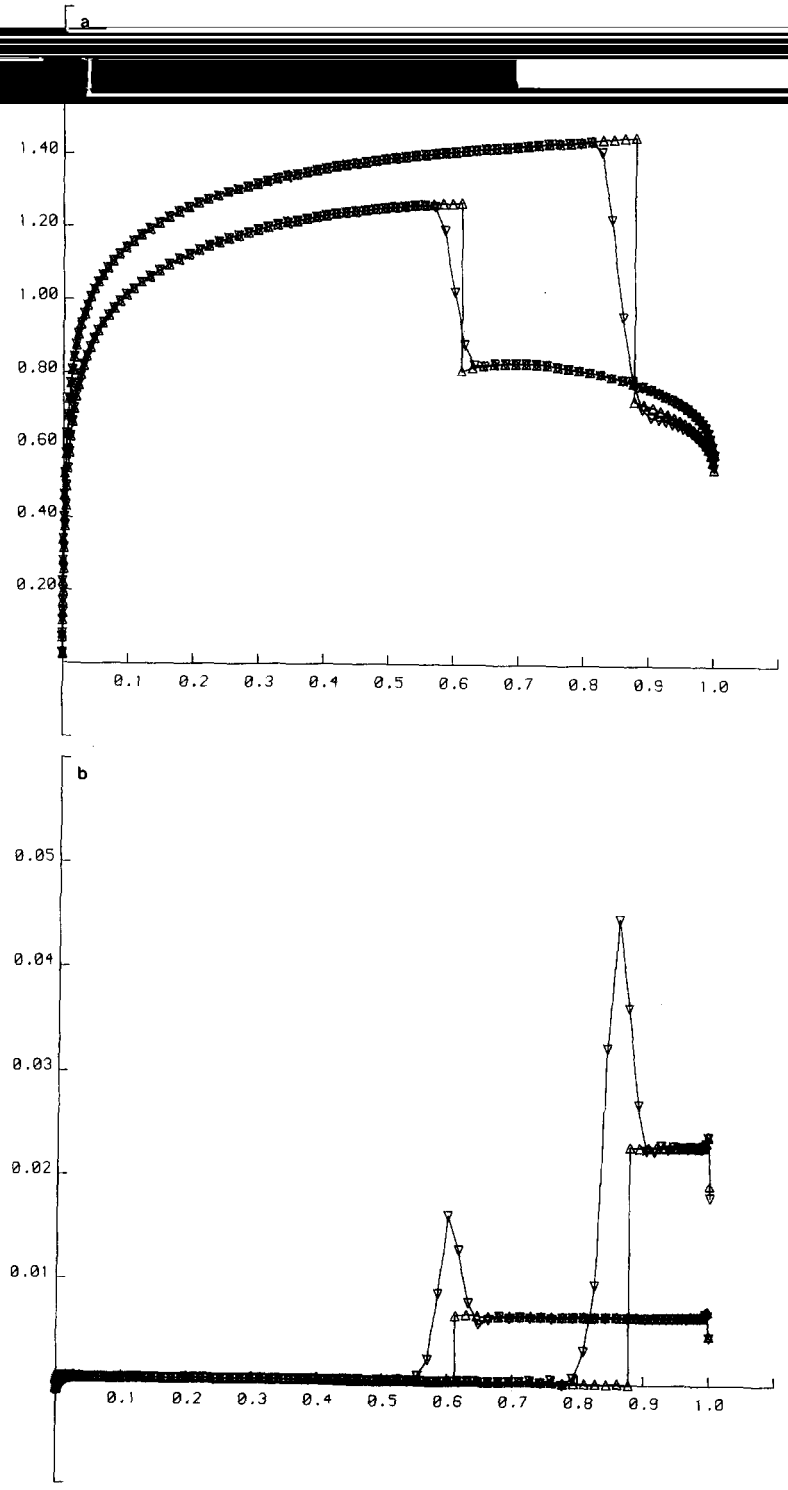

FIG. 12. As in Fig. 11 on the fine mesh of Fig. 8.

It is well known, however, that shock capturing codes require tuning of the artificial dissipation, on which shock positions seem critically dependent. Even on a very fine mesh there is no guarantee that the captured shocks are correctly located. Fitted shocks are free from that source of ambiguity, though the calculation does rely on capturing the stagnation point and slip line at the trailing edge. Furthermore, a series of experiments to be reported elsewhere shows that the fitting procedure is insensitive to such features as the treatment given to the vanishing weak end and the effect of the Zierep singularity on the downstream foot of the shock. This is particularly true on a fine mesh.

A feature deserving improvement in the fitting procedure is the use of a global cubic curve to represent the shock shape. Clearly it would be better if each point on the shock were allowed more freedom to move locally without affecting all others as would be afforded by the use of a stiffened spline. Work is in progress on this aspect, as well as on fitting procedures which do not rely on local mesh adjustment and so could deal with multiple shocks.

ACKNOWLEDGMENTS

The authors would like to thank RAE, Farnborough, for partially supporting the work described here, particularly, Drs. M. G. Hall and C. M. Albone for advice and encouragement in discussions and Dr. M. G. Hall for helpfully providing access to his aerofoil code on CRAY 1-S. The second author also gratefully acknowledges financial support from the Science and Engineering Research Council.

REFERENCES

1. C. M. ALBONE, in *Proceedings, Conference on Numerical Methods for Fluid Dynamics, University of Reading, 1985* edited by K. W. Morton and M. J. Baines (Oxford Univ. Press, London, 1986), p. 427.
2. J. D. DENTON, in *Proceedings, IMA Conference on Numerical Methods in Aeronautical Fluid Dynamics, University of Reading, 1981*, edited by P. L. Roe (Academic Press, London, 1982), p. 189.
3. M. G. HALL, RAE Technical Report 84013, 1984 (unpublished).
4. M. G. HALL, in *Proceedings, Conference on Numerical Methods for Fluid Dynamics, University of Reading, 1985*, edited by K. W. Morton and M. J. Baines (Oxford Univ. Press, London, 1986), p. 303.
5. A. JAMESON, in *Proceedings, IMA Conference on Numerical Methods in Aeronautical Fluid Dynamics, University of Reading, 1981*, edited by P. L. Roe (Academic Press, London, 1982), p. 298.
6. A. JAMESON, T. J. BAKER, AND N. P. WEATHERILL, AIAA Paper 86-0103, 1986 (unpublished).
7. G. M. JOHNSON, NASA Technical Memorandum 82843, 1982 (unpublished).
8. L. D. LANDAU AND E. M. LIFSHITZ, *Fluid Mechanics* (Pergamon, Elmsford, NY, 1959), p. 340.
9. G. MORETTI, Polytechnic Institute of New York, PIBAL Report 73-18, 1973 (unpublished).
10. G. MORETTI, AIAA Paper 74-10, 1974 (unpublished).
11. K. W. MORTON, in *Proceedings, Conference on the Mathematics of Finite Elements and their Applications, Brunel University, 1987*, edited by J. R. Whiteman (Academic Press, London, 1988) p. 353.
12. T. DE NEEF AND G. MORETTI, *Comput. Fluids* **8**, 327 (1980).
13. R.-H. NI, *AIAA J.* **20**, No. 11, 1565 (1981).

14. M. F. PAISLEY, Oxford University Computing Laboratory Report 86/1, 1986 (unpublished).
15. M. F. PAISLEY, D. Phil. thesis, Oxford University, 1986 (unpublished).
16. J. PIKE, *Aero. J.* **89**, 335 (1985).
17. T. H. PULLIAM AND J. T. BARTON, AIAA Paper 85-0018, 1985, (unpublished).
18. R. D. RICHTMYER, National Center for Atmospheric Research Technical Note 63-2, Boulder, 1962 (unpublished).
19. R. D. RICHTMYER AND K. W. MORTON, *Finite Difference Methods for Initial Value Problems* (Wiley-Interscience, New York, 1967).
20. A. W. RIZZI, The Aeronautical Research Institute of Sweden, Bromma, Sweden, private communication (1985).
21. P. L. ROE, ICASE Report 87-6, 1987; *AIAA J.*, in press.
22. J. L. THOMAS AND M. D. SALAS, AIAA Paper 85-0020, 1985 (unpublished).
23. M. D. SALAS, in *Proceedings, 2nd AIAA Computational Fluid Dynamics Conference, 1975*, p. 47.
24. E. TURKEL, ICASE Report 85-43, 1985 (unpublished).
25. J. W. USAB, Ph.D. thesis, Aero and Astro Dept., Massachusetts Institute of Technology, 1983 (unpublished).
26. J. P. VEUILLOT AND L. CAMBIER, ONERA Publication 1884-61, 1984 (unpublished).
27. H. VIVIAND, ONERA Publication 1984-69, 1984 (unpublished).
28. Y.-L. ZHU AND B. CHEN, *Sci. Sinica* **23**, No. 12, 1491 (1980).