NASA Technical Memorandum 101664

The Upwind Control Volume Scheme for Unstructured Triangular Grids

Michael Giles

W. Kyle Anderson

and

Thomas W. Roberts

September 1989

National Aeronautics and Space Administration

Langley Research Center
Hampton, Virginia 23665

The Upwind Control Volume Scheme for Unstructured Triangular Grids

Michael Giles
Massachusetts Institute of Technology
Cambridge, Massachusetts

W. Kyle Anderson
NASA Langley Research Center
Hampton, Virginia

Thomas W. Roberts
Vigyan Research Associates, Inc.
Hampton, Virginia

September 20, 1989

1 Introduction

In this report, a new algorithm for the solution of the steady Euler equations in two-dimensional flow is presented. The development of the scheme was motivated by several considerations. First, the geometric generality of an unstructured grid was desired to allow the ease of treating complex geometries. Second, this generality should not be at the expense of accuracy; it was required that the scheme be second-order accurate in the steady-state. Third, it was desired to keep the data structure as simple as possible to avoid excess overhead in terms of storage, and to minimize communications costs on any massively parallel computer that may be used in the future. This requires a compact difference stencil, so that the grid connectivity information necessary to update the solution at a node is minimal. Finally, the use of upwind differencing was desired to provide sharp resolution of shocks and to minimize the need for adjusting artificial viscosity coefficients, in contrast to some well-established central-difference and finite-element schemes.

The scheme that has emerged from the above requirements is called the Upwind Control Volume (UCV) scheme. The ideas for this algorithm have come from two independent lines of thought. The first comes from the version of the Lax-Wendroff algorithm developed by Ni [1]. Ni's interpretation of his algorithm is that it is a form of upwind scheme, in which the flux residual is evaluated on a cell and then distributed

preferentially towards the downstream nodes. In one dimension, a scalar Lax-Wendroff algorithm with a Courant number of 1.0 becomes an exact upwind scheme, i.e. it produces the exact convection of the initial conditions. However, at smaller values of the Courant number the upwind biasing is lessened because it is a second order term. This is particularly important in solving systems of equations, such as the Euler equations. At a shock, if the Courant number associated with the $u + a$ characteristic is 1.0, then the Courant number of the $u - a$ characteristic is very much smaller, and therefore is only weakly upwinded. However, the latter characteristic is the one that defines the shock. This is the reason that artificial viscosity terms must be added to Ni's Lax-Wendroff calculations to prevent unacceptable oscillations at shocks. The second line of thought comes from the ideas of Moore [2], who has developed a three-dimensional elliptic pressure-correction Navier-Stokes method. A particular feature of her approach is the formulation of the steady-state equations. It is effectively a node-based scheme, but in each cell the momentum and energy equations are assigned to the downstream nodes, and the continuity equation to the upstream nodes. This gives a stable scheme without the addition of any numerical smoothing, and it produces exceptionally good answers on very coarse grids. (It has recently been brought to the authors's attention that Wornam [3] has developed an implicit scheme for the quasi-one-dimensional Euler equations based on a similar idea. In Wornam's scheme, for subsonic flow, the continuity and energy equations are included from the downwind direction and the momentum from upwind, and no numerical smoothing needs to be added.)

Unlike most other upwind schemes that solve a Riemann problem in order to obtain the numerical fluxes, the current approach computes the flux residual for a cell using a cell-vertex trapezoidal rule integration in exactly the same way as the cell-vertex schemes of Ni and Jameson [4]. The present scheme differs from those algorithms in that the distribution of the change in the state vector from the cells to the nodes is performed using a fully upwinded bias. The directions for the upwinding are based on the local flow direction, rather than being oriented with cell-face normals, so that the upwinding is grid independent. Unlike other attempts at grid-independent upwind-differencing that require the construction of an upwind-biased difference stencil (e. g. Davis [5], Levy et al. [6]), the current scheme requires only information at the cell vertices to compute the distribution formulae for the cell. This compactness allows a simple data structure, with cell-to-node pointers being the only connectivity information required. Steady-state solutions are second-order accurate; however, for unsteady problems the scheme is only first-order accurate.

Although the resulting scheme works quite well in predicting the pressure field, the lack of dissipation in the crossflow direction tends to decouple adjacent streamtubes,

resulting in odd-even oscillations in the density across streamtubes. Also, large stagnation density errors were observed near stagnation points. Two additional artificial viscosity terms were added to the scheme that smooth variations in entropy and vorticity. Although these viscosities are formally only first-order accurate, they actually improve the accuracy of the basic scheme in regions of the flow upstream of shocks, as they smooth quantities that should vanish analytically. This is at the cost of slight smearing of shock wakes.

The scheme is described in the next section for a system of time-dependent linear hyperbolic partial differential equations in one space dimension. In Section 3, the extension of the algorithm to the two-dimensional Euler equations is presented. An analysis of the steady-state difference operator for scalar advection is found in Section 4. The entropy and vorticity smoothing are explained in Section 5. Section 6 presents airfoil results in two dimensions on a triangular mesh. Concluding remarks are made in Section 7.

2 One-Dimensional Scheme

The standard first-order flux-upwinding algorithm for the linear system,

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}\frac{\partial \mathbf{U}}{\partial x} = 0, \tag{1}$$

is a cell-centered scheme. For a uniform mesh the semi-discrete form is

$$\Delta x \frac{d\mathbf{U}_j}{dt} + (\mathbf{A}^+ \mathbf{U}_j + \mathbf{A}^- \mathbf{U}_{j+1}) - (\mathbf{A}^+ \mathbf{U}_{j-1} + \mathbf{A}^- \mathbf{U}_j) = 0 \tag{2}$$

where as usual

$$\begin{aligned} \mathbf{A} &= \mathbf{T}\mathbf{\Lambda}\mathbf{T}^{-1}, \quad & \mathbf{\Lambda} &= diag(\lambda_i), \\ \mathbf{A}^+ &= \mathbf{T}\mathbf{\Lambda}^+\mathbf{T}^{-1}, \quad & \mathbf{\Lambda}^+ &= diag(\lambda_i^+), \\ \mathbf{A}^- &= \mathbf{T}\mathbf{\Lambda}^-\mathbf{T}^{-1}, \quad & \mathbf{\Lambda}^- &= diag(\lambda_i^-), \end{aligned}$$

with

$$\lambda^+ = \max(0, \lambda), \qquad \lambda^- = \min(0, \lambda).$$

This scheme can be rewritten using the following definition:

$$\mathrm{sgn}(\mathbf{A}) = \mathbf{T}\,\mathrm{sgn}(\mathbf{\Lambda})\mathbf{T}^{-1}, \quad \mathrm{sgn}(\mathbf{\Lambda}) = diag(\mathrm{sgn}(\lambda_i))$$

with

$$\mathrm{sgn}(\lambda) = \begin{cases} 1 & , \quad \lambda > 0 \\ -1 & , \quad \lambda < 0 \\ 0 & , \quad \lambda = 0. \end{cases}$$

Now then,

$$(\mathbf{I} + \mathrm{sgn}(\mathbf{\Lambda}))\,\mathbf{\Lambda} = 2\mathbf{\Lambda}^{+},$$
$$(\mathbf{I} - \mathrm{sgn}(\mathbf{\Lambda}))\,\mathbf{\Lambda} = 2\mathbf{\Lambda}^{-},$$

and hence

$$\mathbf{A}^{+} = \tfrac{1}{2}(\mathbf{I} + \mathrm{sgn}(\mathbf{A}))\,\mathbf{A},$$
$$\mathbf{A}^{-} = \tfrac{1}{2}(\mathbf{I} - \mathrm{sgn}(\mathbf{A}))\,\mathbf{A}.$$

The difference equation may then be written

$$\Delta x \frac{d\mathbf{U}_j}{dt} + \tfrac{1}{2}\left[(\mathbf{I} + \mathrm{sgn}(\mathbf{A}))\mathbf{A}(\mathbf{U}_j - \mathbf{U}_{j-1}) + (\mathbf{I} - \mathrm{sgn}(\mathbf{A}))\mathbf{A}(\mathbf{U}_{j+1} - \mathbf{U}_j)\right] = 0. \quad (3)$$

Notice here the strong similarity to Ni's distribution formulae. For a problem in which all characteristic speeds are identical, and using forward Euler time integration with a CFL number of 1.0, this scheme is identical to Ni's, which becomes purely upwinded under these conditions.

Next, for the quasi-one-dimensional Euler equations,

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = \mathbf{H}, \quad (4)$$

the UCV (upwind control volume) scheme for a nonuniform mesh is

$$(x_{j+1} - x_{j-1})\frac{d\mathbf{U}_j}{dt} = -\Big((\mathbf{I} + \mathrm{sgn}(\mathbf{A}_{j-\frac{1}{2}}))(\mathbf{F}_j - \mathbf{F}_{j-1} - (x_j - x_{j-1})\mathbf{H}_{j-\frac{1}{2}})$$
$$+ (\mathbf{I} - \mathrm{sgn}(\mathbf{A}_{j+\frac{1}{2}}))(\mathbf{F}_{j+1} - \mathbf{F}_j - (x_{j+1} - x_j)\mathbf{H}_{j+\frac{1}{2}})\Big). \quad (5)$$

Notice that for irregular grids, this is an improvement over the usual flux-upwinding scheme that is only first order because now the source term $\mathbf{H}$ is being included with the correct value and the correct Δx so that in the steady state the solution is second order. Also, note that the scheme is conservative. This scheme for the one-dimensional Euler equations turns out to be identical to that proposed independently by van Leer, et al. [7].

3 Two-Dimensional Euler Scheme

Now consider the full Euler equations of gas dynamics. In conservation form, the equations are

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} + \frac{\partial \mathbf{G}(\mathbf{U})}{\partial y} = 0 \quad (6)$$

where

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix}, \qquad \mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (\rho E + p)u \end{pmatrix}, \qquad \mathbf{G}(\mathbf{U}) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ (\rho E + p)v \end{pmatrix}.$$

Here, ρ is the density; u and v, the velocity components in the x and y directions, respectively; p, the pressure; and E, the specific total energy. The working fluid is taken to be an ideal gas with a constant specific heats ratio γ, which gives

$$E = \frac{u^2 + v^2}{2} + \frac{1}{\gamma - 1}\frac{p}{\rho}. \tag{7}$$

In the semi-discrete Euler equations on an unstructured triangular mesh, the change in the state vector at a node has contributions from all of the cells which surround that node.

$$V_j \frac{d\mathbf{U}_j}{dt} = -\sum_{cells} (\mathbf{R}_j)_K, \tag{8}$$

V_j is the volume of the cell associated with node j, which can be defined as one third of the sum of the areas of the triangles around j, and $(\mathbf{R}_j)_K$ is the contribution to the flux residual at node j due to cell K.

For a cell K with nodes 1,2,3 (numbered counter-clockwise) the flux residual $\mathbf{R}_K$ is defined in the usual node-based way:

$$\begin{aligned} \mathbf{R}_K &= \sum \overline{\mathbf{F}}\,\Delta y - \overline{\mathbf{G}}\,\Delta x \\ &= \tfrac{1}{2}\big(\mathbf{F}_1(y_2 - y_3) - \mathbf{G}_1(x_2 - x_3) \\ &\quad + \mathbf{F}_2(y_3 - y_1) - \mathbf{G}_2(x_3 - x_1) \\ &\quad + \mathbf{F}_3(y_1 - y_2) - \mathbf{G}_3(x_1 - x_2) \big). \end{aligned} \tag{9}$$

This residual is then distributed to the three nodes as follows:

$$\begin{aligned} \mathbf{R}_{1K} &= \tfrac{1}{3}\left(\mathbf{I} - \sigma_s \Delta\tilde{n}_1 \operatorname{sgn}(\mathbf{A}_s) + \sigma_n \Delta\tilde{s}_1 \operatorname{sgn}(\mathbf{A}_n)\right)\mathbf{R}_K, \\ \mathbf{R}_{2K} &= \tfrac{1}{3}\left(\mathbf{I} - \sigma_s \Delta\tilde{n}_2 \operatorname{sgn}(\mathbf{A}_s) + \sigma_n \Delta\tilde{s}_2 \operatorname{sgn}(\mathbf{A}_n)\right)\mathbf{R}_K, \\ \mathbf{R}_{3K} &= \tfrac{1}{3}\left(\mathbf{I} - \sigma_s \Delta\tilde{n}_3 \operatorname{sgn}(\mathbf{A}_s) + \sigma_n \Delta\tilde{s}_3 \operatorname{sgn}(\mathbf{A}_n)\right)\mathbf{R}_K. \end{aligned} \tag{10}$$

σ_s and σ_n are parameters controlling the amount of upwinding in the streamwise and normal direction. Their usual values will be 1.0, while values of 0.0 will give Jameson's standard node-based algorithm. The streamwise and normal flux Jacobians $\mathbf{A}_s$ and $\mathbf{A}_n$ are defined by

$$\begin{aligned} \mathbf{A}_s &= \mathbf{A}s_x + \mathbf{B}s_y, \\ \mathbf{A}_n &= \mathbf{A}n_x + \mathbf{B}n_y, \end{aligned} \tag{11}$$

where $\mathbf{A} = \partial\mathbf{F}/\partial\mathbf{U}$ and $\mathbf{B} = \partial\mathbf{G}/\partial\mathbf{U}$. The unit normal vector $\vec{n} = (n_x, n_y)^T$ is defined to be normal to the unit streamwise vector $\vec{s} = (s_x, s_y)^T$ which in turn is defined as

$$\vec{s} = \frac{\vec{u}_1 + \vec{u}_2 + \vec{u}_3}{|\vec{u}_1 + \vec{u}_2 + \vec{u}_3|}. \tag{12}$$

Finally, the terms $\Delta\tilde{n}_j$ and $\Delta\tilde{s}_j$ are defined by these equations:

$$\begin{aligned}
\Delta n_1 &= (\vec{x}_3 - \vec{x}_2)\cdot\vec{n}, \\
\Delta n_2 &= (\vec{x}_1 - \vec{x}_3)\cdot\vec{n}, \\
\Delta n_3 &= (\vec{x}_2 - \vec{x}_1)\cdot\vec{n}, \\
\\
\Delta s_1 &= (\vec{x}_3 - \vec{x}_2)\cdot\vec{s}, \\
\Delta s_2 &= (\vec{x}_1 - \vec{x}_3)\cdot\vec{s}, \\
\Delta s_3 &= (\vec{x}_2 - \vec{x}_1)\cdot\vec{s}, \\
\\
\Delta\tilde{n}_j &= \frac{\Delta n_j}{\frac{1}{2}(|\Delta n_1| + |\Delta n_2| + |\Delta n_3|)}, \\
\Delta\tilde{s}_j &= \frac{\Delta s_j}{\frac{1}{2}(|\Delta s_1| + |\Delta s_2| + |\Delta s_3|)}.
\end{aligned} \tag{13}$$

Thus the $\Delta\tilde{n}_j$'s and $\Delta\tilde{s}_j$'s vary from -1 to +1.

4 Analysis of Two-Dimensional Scalar Algorithm

To gain some understanding of the workings of the two-dimensional algorithm, consider its application to the scalar convection equation,

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0, \tag{14}$$

which describes a wave moving in the positive x-direction. To understand the behavior of steady-state solutions, consider the case when the time-derivative vanishes, i.e.

$$\frac{\partial u}{\partial x} = 0. \tag{15}$$

First let us examine the fluxes distributed from the three equilateral triangles shown in Fig. 1:

$$\begin{aligned}
R_{1A} &= 0, \\
R_{2A} &= \tfrac{2}{3}R_A, \\
R_{3A} &= \tfrac{1}{3}R_A,
\end{aligned} \tag{16}$$

$$
\begin{aligned}
R_{1B} &= \tfrac{1}{6} R_B, \\
R_{2B} &= \tfrac{2}{3} R_B, \\
R_{3B} &= \tfrac{1}{6} R_B,
\end{aligned}
\tag{17}
$$

$$
\begin{aligned}
R_{1C} &= \tfrac{1}{2} R_C, \\
R_{2C} &= \tfrac{1}{2} R_C, \\
R_{3C} &= 0.
\end{aligned}
\tag{18}
$$

The downwind nodes receive most of the flux residual, and the upwind nodes relatively little. Combining these results yields the steady-state flux operators for the meshes shown in Fig. 2.

For grid 1, the steady-state equation is

$$
\begin{aligned}
&\frac{2}{3}\left(\frac{\Delta y}{2} u_1 - \frac{\Delta y}{2} u_6\right) + \frac{2}{3}\left(\frac{\Delta y}{2} u_1 - \frac{\Delta y}{2} u_6\right) + \\
&\frac{1}{3}\left(\frac{\Delta y}{2} u_4 - \frac{\Delta y}{2} u_5\right) + \frac{1}{3}\left(\frac{\Delta y}{2} u_2 - \frac{\Delta y}{2} u_7\right) \\
&\quad = \frac{2\Delta y}{3}\left(u_1 + \tfrac{1}{4} u_2 + \tfrac{1}{4} u_4 - u_6 - \tfrac{1}{4} u_5 - \tfrac{1}{4} u_7\right) = 0.
\end{aligned}
\tag{19}
$$

For grid 2, the steady-state equation is

$$
\begin{aligned}
&\frac{2}{3}\left(\frac{\Delta y}{2} u_1 - \frac{\Delta y}{4} u_6 - \frac{\Delta y}{4} u_7\right) + \\
&\frac{1}{2}\left(\frac{\Delta y}{4} u_1 + \frac{\Delta y}{4} u_2 - \frac{\Delta y}{2} u_7\right) + \frac{1}{2}\left(\frac{\Delta y}{4} u_1 + \frac{\Delta y}{4} u_5 - \frac{\Delta y}{2} u_6\right) + \\
&\frac{1}{6}\left(\frac{\Delta y}{2} u_3 - \frac{\Delta y}{4} u_1 - \frac{\Delta y}{4} u_2\right) + \frac{1}{6}\left(\frac{\Delta y}{2} u_4 - \frac{\Delta y}{4} u_1 - \frac{\Delta y}{4} u_5\right) \\
&\quad = \frac{\Delta y}{2}\left(u_1 + \tfrac{1}{6} u_2 + \tfrac{1}{6} u_3 + \tfrac{1}{6} u_4 + \tfrac{1}{6} u_5 - \tfrac{5}{6} u_6 - \tfrac{5}{6} u_7\right) = 0.
\end{aligned}
\tag{20}
$$

The best way in which to determine whether these discretizations are diffusive is to perform a Fourier analysis of the steady solutions. Define

$$
u_{jk} = z^j \exp(ik\theta)
$$

where θ is the Fourier mode in the y-direction and z is the spatial growth factor in the x-direction. Because the grid is made of equilateral triangles, u_{jk} must be carefully interpreted. The node numbering system is defined such that for even values of j the nodes are at integer values of k, while for odd values of j the nodes are at the half-integer positions. This indexing system is indicated in Fig. 3.

Substituting the Fourier mode into the steady-state equation for grid 1 yields a cubic equation for z:

$$
(z^2 - 1)(\cos(\theta/2) z + 2) = 0. \tag{21}
$$

The roots of this are

$$z = 1, -1, -\frac{2}{\cos(\theta/2)}. \tag{22}$$

The first two roots correspond to perfect convection of the quantity u along grid lines. This is possible because the convection velocity is perfectly aligned with the one set of edges of the computational domain. The third root is negative and has a magnitude which is greater than unity; this corresponds to an oscillatory, exponential type of solution. It is related to the fact that the discrete stencil includes nodes downstream of the central node, and it shows how the incorrect specification of the outflow boundary condition can lead to a local error which decays exponentially as one moves into the domain away from the boundary.

Substituting the Fourier mode into the steady-state equation for grid 2 yields a quadratic equation for z:

$$\cos(\theta/2)z^2 + 2\left(1 + \cos^2(\theta/2)\right) z - 5\cos(\theta/2) = 0. \tag{23}$$

One root of this quadratic is

$$\begin{aligned} z_1 &= \frac{-1 - \cos^2(\theta/2) + \sqrt{(1 + cos^2(\theta/2))^2 + 5\cos^2(\theta/2)}}{\cos(\theta/2)} \\ &= \frac{5\cos(\theta/2)}{1 + \cos^2(\theta/2) + \sqrt{(1 + cos^2(\theta/2))^2 + 5\cos^2(\theta/2)}} \\ &= \frac{5}{\frac{1}{\cos(\theta/2)} + \cos(\theta/2) + 3\sqrt{1 + \frac{\sin^4(\theta/2)}{9\cos^2(\theta/2)}}} \\ &= \frac{5}{2 + \left(\frac{1}{\sqrt{\cos(\theta/2)}} - \sqrt{\cos(\theta/2)}\right)^2 + 3\sqrt{1 + \frac{\sin^4(\theta/2)}{9\cos^2(\theta/2)}}}. \end{aligned} \tag{24}$$

From this last result it is clear that z_1 is real and lies in the range $0 < z_1 < 1$. Also, when $\theta \ll 1$,

$$z_1 \approx \frac{1}{1 + \theta^4/192} \approx 1 - \theta^4/192. \tag{25}$$

This shows that the effect of the upwinding is comparable in nature to a fourth-difference smoothing when the computational grid is not perfectly aligned with the convection direction.

The other root of the quadratic is

$$z_2 = -\frac{5}{z_1}. \tag{26}$$

This root lies in the range $z < -5$ and so it again gives the oscillatory boundary-layer behavior at the outflow boundary. In an Euler equation application this would also give the exponential decay on either side of a shock.

5 Entropy and Vorticity Smoothing

When applied to airfoil flows, the scheme as described above accurately predicts pressure distributions. However, two types of errors have been observed. The first is a significant error ($\sim$ 30 to 50%) in the stagnation density. It is highly localized, occurring at the grid point on the surface nearest the stagnation point. This error is not surprising, as the distribution formulae depend on the flow direction which is singular at the stagnation point. The second error is due to the low dissipation in the crossflow direction, which tends to decouple adjacent streamtubes. This decoupling results in an odd-even oscillation of the density and velocity in the crossflow direction. The pressure field in any case is quite accurate and non-oscillatory, and no instabilities are observed.

To fix these errors, two additional forms of artificial viscosity are added to the basic scheme: the first is a smoothing based on a second difference of entropy; the second is a smoothing based on vorticity. Both forms of the smoothing are conservative, and they have the property that they will return zero in an isentropic, irrotational flow. For airfoils in a uniform freestream, the entropy and vorticity smoothing do not corrupt the flow upstream of any shocks, and smear post-shock wakes only slightly. Also it has been found that the smoothing does not substantially smear the shocks. These artificial dissipation models were developed by the first author for his turbomachinery Euler code UNSFLO and are described in reference [8].

The entropy smoothing is based on a second difference in entropy. The (nondimensional) entropy for an ideal gas is defined as

$$S \equiv \ln \frac{p}{p_\infty} - \gamma \ln \frac{\rho}{\rho_\infty},$$

where the subscript ∞ refers to freestream conditions. Since it is desired that only those entropy variations that are decoupled from the pressure and velocity field are to be smoothed, the smoothing flux is based on

$$\delta \mathbf{U}^e = \left. \frac{\partial \mathbf{U}}{\partial S} \right|_{p,\vec{u}} \delta S$$

where $\partial \mathbf{U}/\partial S$ is evaluated at constant pressure and velocity. This gives

$$\left. \frac{\partial \mathbf{U}}{\partial S} \right|_{p,\vec{u}} = -\frac{\rho}{\gamma} \begin{pmatrix} 1 \\ u \\ v \\ \frac{u^2+v^2}{2} \end{pmatrix}. \tag{27}$$

Note that this is merely the projection of the right eigenvector of the flux Jacobians $\partial \mathbf{F}/\partial \mathbf{U}$ and $\partial \mathbf{G}/\partial \mathbf{U}$ corresponding to entropy waves onto the state vector of con-

served quantities. This is simply a restatement that only entropy variations are being smoothed.

The smoothing flux is then computed by taking the average of the entropy of the three nodes making up a cell, and distributing changes conservatively from each cell back to the nodes. The contribution of the smoothing flux from cell K to its i^{th} node is then

$$\delta \mathbf{U}_{iK}^{e} = \nu_e \left(\frac{\Delta t}{A}\right)_i \left(\frac{A}{\Delta t}\right)_K \left.\frac{\partial \mathbf{U}}{\partial S}\right|_{p,\vec{u}} (S_K - S_i). \tag{28}$$

Here, $(\Delta t/A)_i$ is the timestep over the area for node i, $(A/\Delta t)_K$ is the area over the local timestep of cell K, $S_K \equiv (S_1 + S_2 + S_3)/3$ is the average entropy over the cell, and ν_e is an arbitrary smoothing coefficient.

Although this formula is formally only first order, in fact it improves the accuracy of the basic scheme upstream of shocks because the flow should be isentropic in that region. Thus any entropy variations in that region are due to the truncation error of the scheme. In shock wakes, of course, there exist entropy gradients, and the addition of the smoothing does smear the wake. For moderate values of ν_e this smearing is small.

Inspection of Equation 27 shows that there is no smoothing of the crossflow momentum component. As a result, there will still be a decoupling of adjacent streamtubes, and a crossflow odd-even mode is allowed. To get rid of this error, a smoothing based on the vorticity is used. This smoothing is based on the observation that

$$\nabla \times \vec{\omega} = -\nabla^2 \vec{u} + \nabla (\nabla \cdot \vec{u})$$

where $\vec{\omega} \equiv \nabla \times \vec{u}$ is the vorticity. Thus, the addition to the momentum equations of a term proportional to $\nabla \times \vec{\omega}$ should smooth the flow when there are vorticity variations, but not corrupt irrotational flow.

In two-dimensional flow, $\vec{\omega} = \hat{k}\omega$, and the average vorticity in cell K is

$$\omega = \frac{1}{A} \oint_{\partial K} \vec{u} \cdot d\vec{s},$$

and the average value of $\nabla \times \left(\hat{k}\omega\right)$ is

$$\nabla \times \left(\hat{k}\omega\right) = \frac{1}{A} \oint_{\partial K} \omega d\vec{s}.$$

Based upon these observations, the vorticity smoothing is achieved by adding to the flux residuals of each node a term

$$\delta \mathbf{U}_i^{v} = - \left(\frac{\Delta t}{A}\right)_i \oint_{\partial A_i} \Delta \mathbf{F}\, dy - \Delta \mathbf{G}\, dx \tag{29}$$

where $\Delta\mathbf{F}$ and $\Delta\mathbf{G}$ for the cell are defined as

$$\Delta\mathbf{F} = -\begin{pmatrix} 0 \\ \nu_v \rho a \,\mathrm{curl}\,(\vec{u}) \\ 0 \\ 0 \end{pmatrix}, \qquad \Delta\mathbf{G} = \begin{pmatrix} 0 \\ 0 \\ \nu_v \rho a \,\mathrm{curl}\,(\vec{u}) \\ 0 \end{pmatrix}, \tag{30}$$

where ν_v is an arbitrary smoothing coefficient, a is the speed of sound for the cell, and $\mathrm{curl}\,(\vec{u})$ is a scaled flow circulation,

$$\mathrm{curl}\,(\vec{u}) = \frac{\Delta u_{21}\Delta x_{31} - \Delta u_{31}\Delta x_{21} + \Delta v_{21}\Delta y_{31} - \Delta v_{31}\Delta y_{21}}{\sqrt{\Delta x_{21}\Delta y_{31} - \Delta x_{31}\Delta y_{21}}}, \tag{31}$$

where $\Delta()_{ij} \equiv ()_i - ()_j$. The contributions of the vorticity smoothing fluxes of cell K to its surrounding nodes are

$$\begin{aligned} \delta\mathbf{U}^v_{1K} &= -\left(\frac{\Delta t}{A}\right)_1 \left(\Delta\mathbf{F}_K\,(y_3 - y_2) - \Delta\mathbf{G}_K\,(x_3 - x_2)\right), \\ \delta\mathbf{U}^v_{2K} &= -\left(\frac{\Delta t}{A}\right)_2 \left(\Delta\mathbf{F}_K\,(y_1 - y_3) - \Delta\mathbf{G}_K\,(x_1 - x_3)\right), \\ \delta\mathbf{U}^v_{3K} &= -\left(\frac{\Delta t}{A}\right)_3 \left(\Delta\mathbf{F}_K\,(y_2 - y_1) - \Delta\mathbf{G}_K\,(x_2 - x_1)\right). \end{aligned} \tag{32}$$

The most attractive features of the entropy and vorticity smoothing are that they have a compact difference stencil, and they are well suited to transonic flows with a uniform freestream. However, these smoothing terms are not suited to high Mach number flows, in which there are strong shocks with large gradients of entropy and vorticity downstream of the shocks. For such flows, the smoothing errors are truly first order in most of the flow field. Also, the present scheme should be extended to the Navier-Stokes equations. The use of such smoothing for viscous flows would result in first-order errors thin-shear-layer regions. This latter problem may be moot, in that physical viscosity should suppress the odd-even mode and the stagnation density error as long as the viscous layers are properly resolved. In this case, there may be no need for the entropy and vorticity smoothing in the viscous regions of the flow. This question will be addressed in future work. However, the unsuitability of the present smoothing for inviscid flows with strong shocks remains.

For flows with freestream Mach numbers significantly greater than 1, a background fourth-difference artificial viscosity similar to that of Jameson et al. [4] is preferable. Preliminary computations using fourth-difference smoothing have been run for airfoil flows from transonic to Mach 3.5 freestreams, and it works quite well; for transonic flows, the results are very similar to those obtained with the entropy and vorticity smoothing. Because the latter smoothing requires less memory, it is preferred for transonic flows.

6 Results

To illustrate the capability of the scheme, solutions have been obtained for a standard inviscid test case given in [9], and referred to as AGARD 01. This case consists of flow over the NACA 0012 airfoil with a freestream Mach number $M_\infty = 0.8$ and angle of attack $\alpha = 1.25°$. The first set of results were obtained on the grid illustrated in Fig. 4. This grid was generated from a structured O-grid of 5248 nodes (128×41) by dividing each quadrilateral cell across a diagonal. The structured grid itself was used to obtain a benchmark solution using the proven code CFL2D [10], which uses an upwind-differencing algorithm.

Figure 5 presents the surface c_p distribution on the airfoil surface for CFL2D, the basic UCV scheme, and the UCV scheme with entropy and vorticity smoothing. Both σ_s and σ_n were equal to 1, and the smoothing coefficients ν_e and ν_v for the third case were both taken to be 0.01. All three solutions agree very well with each other. The UCV scheme gives very sharp shocks, with only moderate pre- and post-shock overshoots. There are no undamped oscillations around the shocks, as will typically occur with central-differencing algorithms. Note that this is true of the solution without entropy and vorticity smoothing; the basic scheme UCV scheme gives nonoscillatory pressures. Also, note that the entropy and vorticity smoothing do not smear the shock significantly.

From these results, it is not apparent why the entropy and vorticity smoothing are needed. To show why they are necessary, contours of constant density for the basic UCV scheme are presented in Fig. 6. Note the boundary-layer behavior at the wall, as well as the significant odd-even oscillations in the crossflow direction. These oscillations are particularly severe behind the shock, although they do not result in any instability. More serious, but not apparent in Fig. 6, is a large stagnation density error at the leading edge. The density at that point is 1.786, compared to the exact value of 1.351. Although not shown, the pressure contours are smooth, and the error in the stagnation pressure is much less severe, which indicates that the error lies in the failure of the scheme to compute the entropy correctly.

With the addition of the entropy and vorticity smoothing (Fig. 7) both the boundary-layer and the odd-even mode are eliminated. The stagnation density now comes down to a more reasonable value, 1.388. It was found that the entropy smoothing alone failed to eliminate the odd-even mode, as there is no smoothing contribution to the crossflow momentum equation where the normal velocity vanishes. Both entropy and vorticity smoothing were found to be necessary.

This test case has also been run using a background fourth-difference smoothing like

that of Jameson et al. [4] in place of the entropy and vorticity smoothing. A smoothing coefficient of 0.005 was found to be adequate to eliminate most of the stagnation density error (the computed stagnation density is 1.349 for this case). The surface c_p distribution is shown in Fig. 8. Note that the primary difference between this result and the result obtained with entropy and vorticity smoothing is more smearing of the shocks, especially the very weak shock on the lower surface. Density contours are plotted in Fig. 9; they show that the fourth-difference smoothing also works well to eliminate the crossflow odd-even mode.

The major reason for using an unstructured grid is the ability to handle arbitrary geometries. For complicated objects, it may be difficult to get a smooth grid. To illustrate how the scheme is suitable for distorted grids, a solution for AGARD 01 on a grid generated by Tim Barth and Dennis Jespersen [11] of NASA Ames Research Center, shown in Fig. 10, has been obtained. This grid has 6691 nodes, and was constructed by generating weighted pairs of pseudo-random numbers and Delaunay triangulation. Barth and Jespersen obtained a solution for the same test case on this grid, using their upwind algorithm presented in [11]. The UCV solution on this grid is shown in Figs. 11 and 12. This solution was obtained with the entropy and vorticity coefficients set at 0.01, as before. Again, the surface pressures are in good agreement with CFL2D. (Note: the CFL2D results are those obtained on the grid shown in Fig. 4, which has roughly 2.5 times fewer nodes on the surface as the Barth grid.) Sharp shocks with small overshoot are obtained. The density contours in Fig. 12 are also seen to be very smooth, with no boundary-layer behavior.

7 Conclusions

A new Upwind Control Volume (UCV) scheme has been developed for obtaining numerical solutions to the Euler equations. This scheme has several very attractive features compared to existing methods. It is well suited to unstructured grids, providing geometric flexibility. It is much simpler than most upwind schemes, in that it does not require complicated flux-limiting. The data structure required to implement the scheme on a triangular mesh is minimal: only cell-to-node pointers are needed. Although it is necessary to add smoothing to the basic UCV scheme in order to reduce stagnation density errors, the choice of entropy and vorticity smoothing does not corrupt the flow in irrotational, isentropic regions. The ability of the scheme to handle distorted grids without loss of accuracy has also been demonstrated.

References

[1] Ni, R.-H., "A Multiple Grid Scheme for Solving the Euler Equations," *AIAA Journal*, vol. 20, Oct. 1981, pp. 1565-1571.

[2] Moore, J. G., "Calculation of 3-D Flow Without Numerical Mixing," *Numerical Methods for Flows in Turbomachinery*, VKI Lecture Series 1989-06, May 1989.

[3] Wornam, S. F., "Application of Two-Point Implicit Central-Difference Methods to Hyperbolic Systems," to be published in *Computers and Fluids*.

[4] Jameson, A., Baker, T. J., and Weatherill, N. P., "Calculation of Inviscid Transonic Flow over a Complete Aircraft," AIAA Paper 86-0103, Jan. 1986.

[5] Davis, S. F., "A Rotationally-Biased Upwind Difference Scheme for the Euler Equations," *Journal of Computational Physics*, vol. 56, 1984.

[6] Levy, D. G., van Leer, B., and Powell, K. G., "An Implementation of a Grid-Independent Upwind Scheme for the Euler Equations," AIAA Paper 89-1931CP, June 1989.

[7] van Leer, B., Lee, W.-T., and Powell, K. G., "Sonic-Point Capturing," AIAA Paper 89-1945-CP, in *AIAA* 9^{th} *Computational Fluid Dynamics Conference* proceedings, June 1989.

[8] Giles, M., *UNSFLO: A Numerical Method For Unsteady Inviscid Flow In Turbomachinery*, Massachusetts Institute of Technology, Gas Turbine Laboratory Report 195, Oct. 1988.

[9] AGARD Subcommittee C., *Test Cases for Inviscid Flow Field Methods*, AGARD Advisory Report 211, 1986.

[10] Anderson, W. K., Thomas, J. L., and van Leer, B., "A Comparison of Finite Volume Flux Vector Splittings for the Euler Equations," *AIAA Journal*, vol. 24, Sept. 1986, pp. 1453-1460.

[11] Barth, T. J., and Jespersen, D. C., "The Design and Application of Upwind Schemes on Unstructured Meshes," AIAA Paper 89-0366, Jan. 1989.

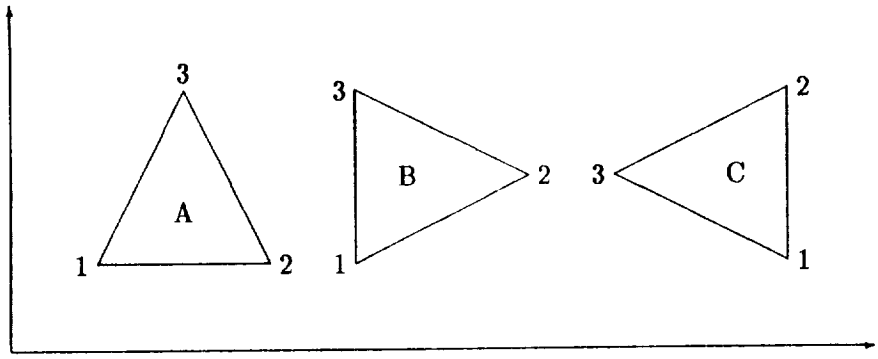

Figure 1: Three equilateral triangles

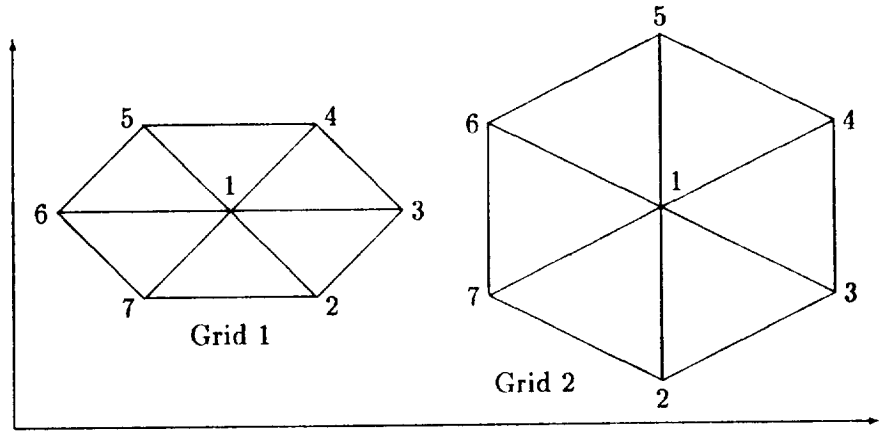

Figure 2: Two triangular meshes

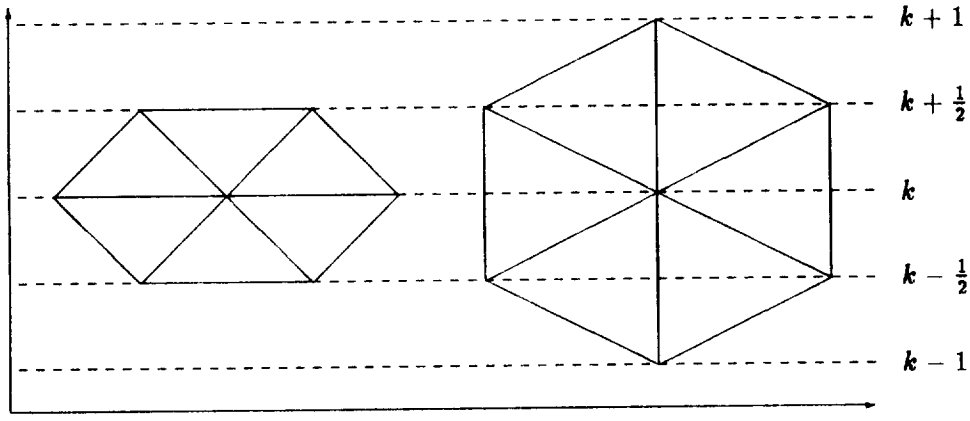

Figure 3: Node indexing system

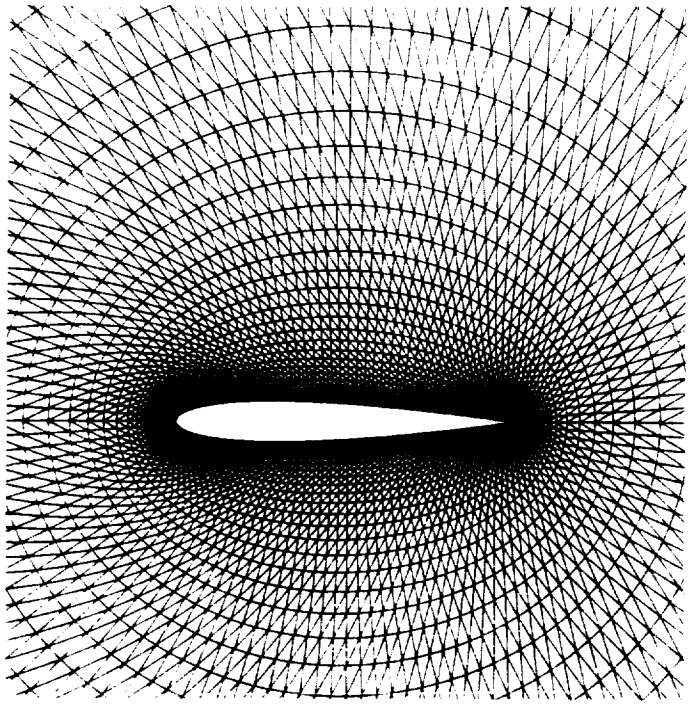

Figure 4: Smooth grid

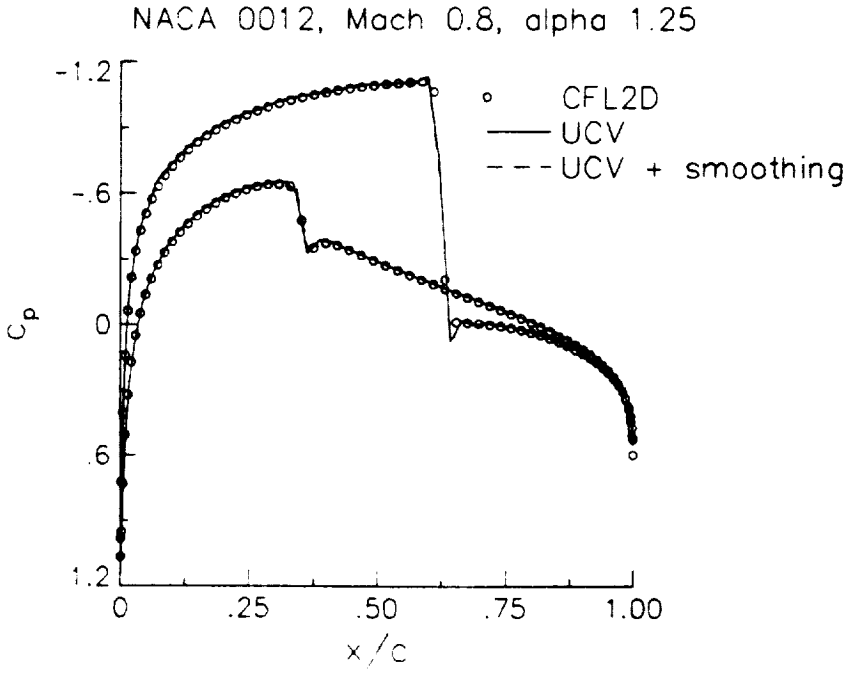

Figure 5: Surface c_p comparison, smooth grid

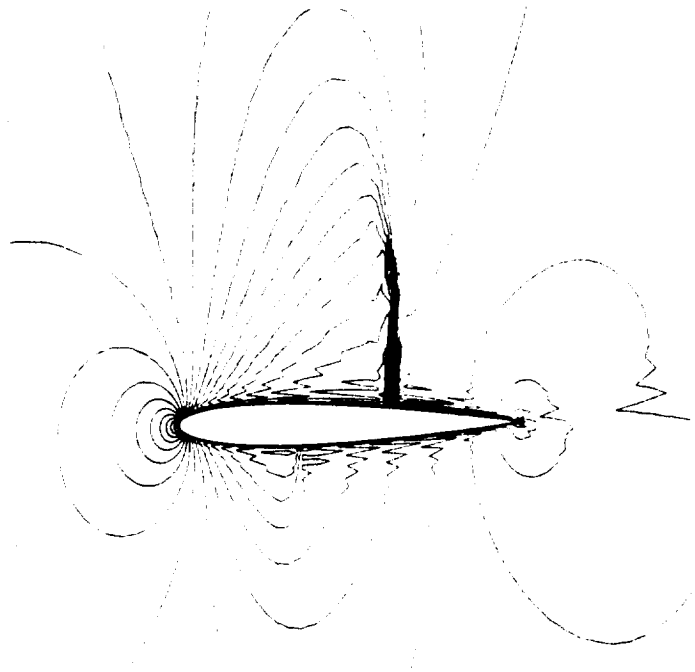

Figure 6: Constant density contours, basic UCV scheme

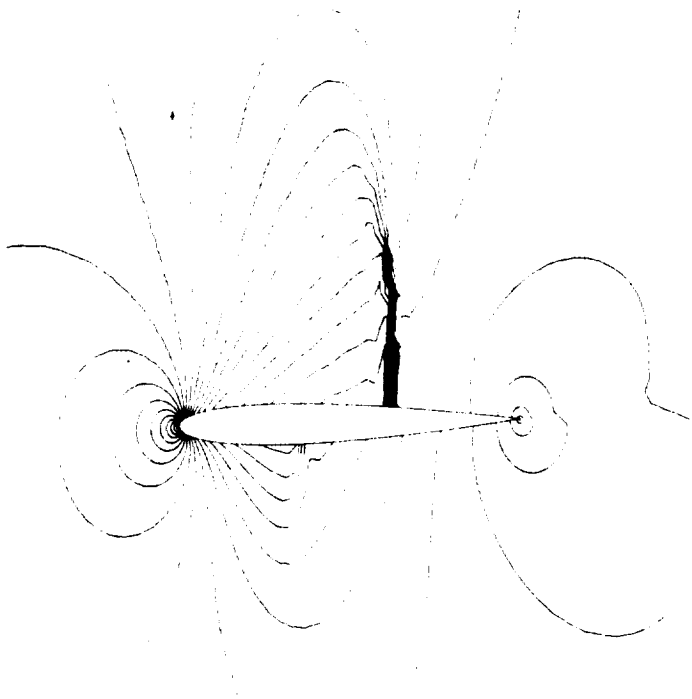

Figure 7: Constant density contours, entropy and vorticity smoothing

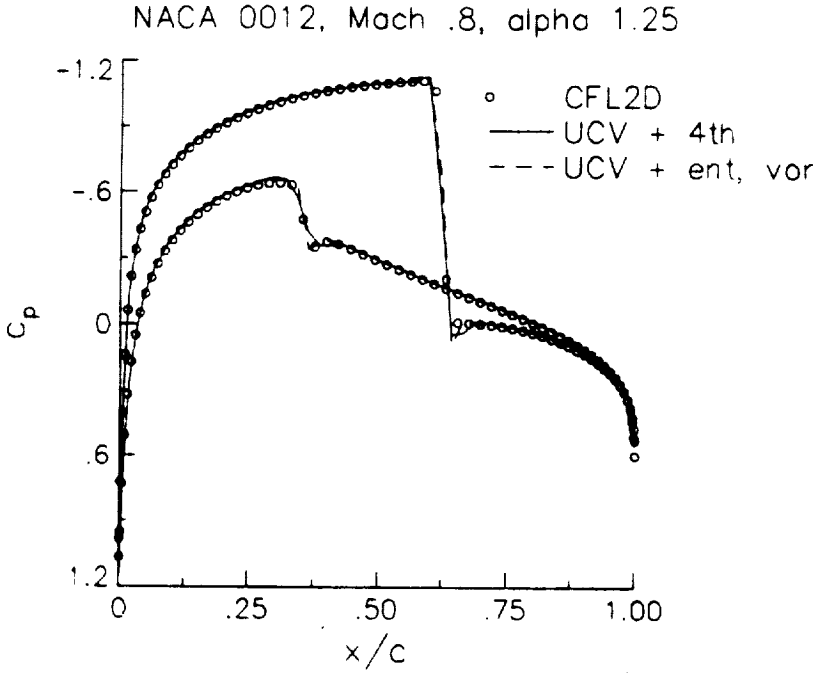

Figure 8: Surface c_p comparison, different smoothing

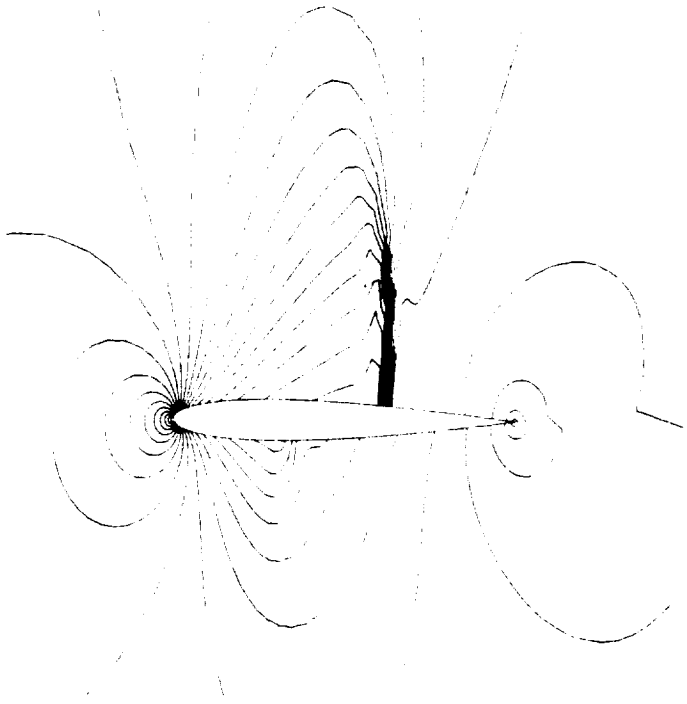

Figure 9: Constant density contours, fourth-difference smoothing

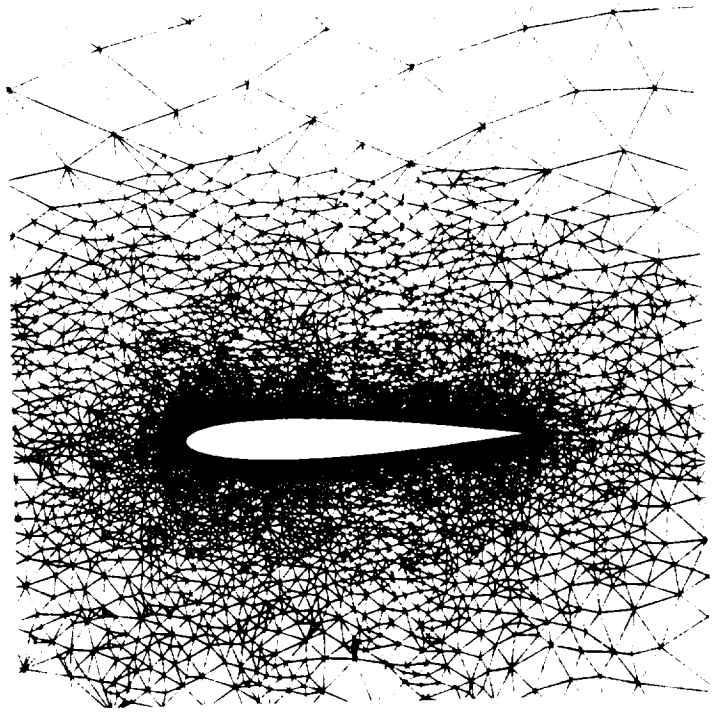

Figure 10: Irregular grid

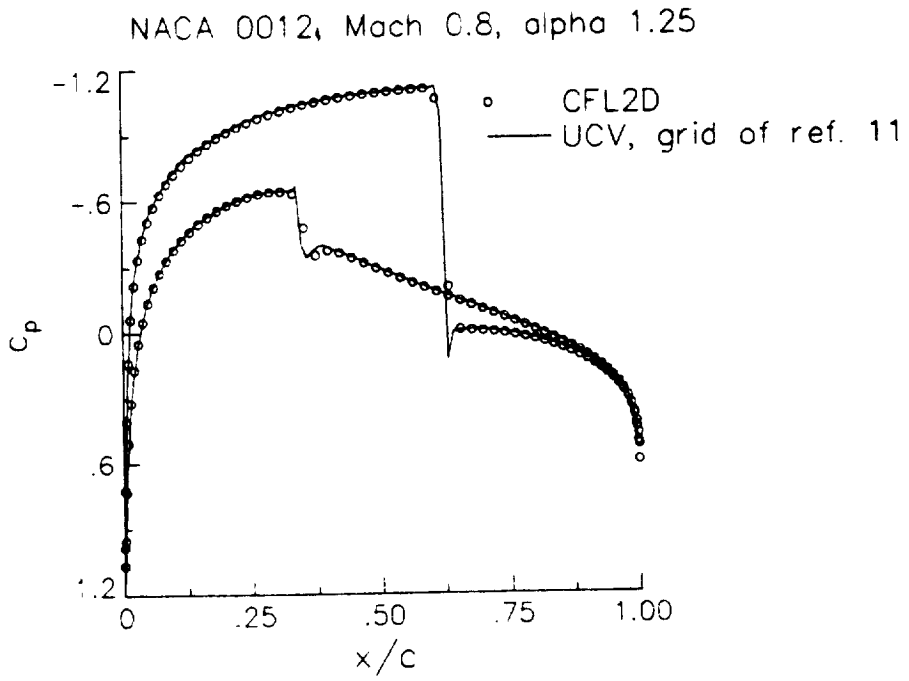

Figure 11: Surface c_p comparison, irregular grid

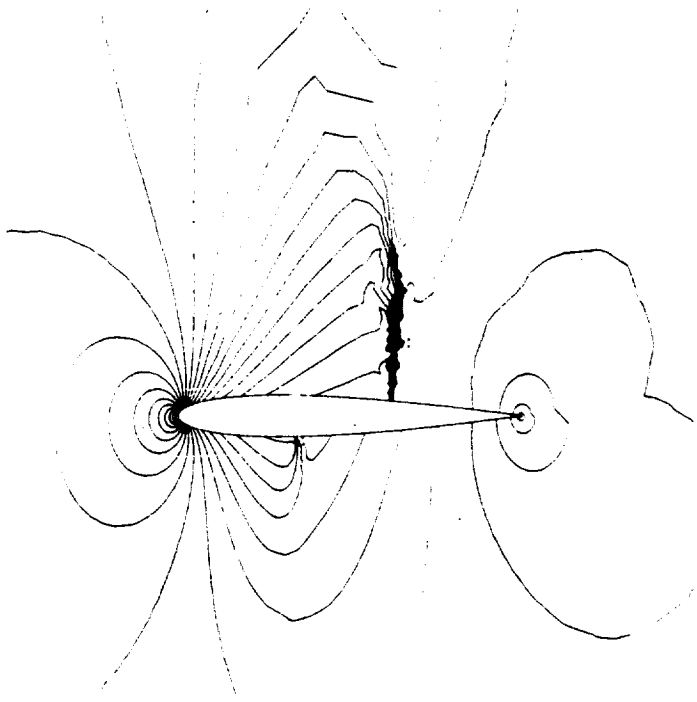

Figure 12: Constant density contours, irregular grid

NASA National Aeronautics and Space Administration

Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM-101664		

4. Title and Subtitle
The Upwind Control Volume Scheme for Unstructured Triangular Grids

5. Report Date
September 1989

6. Performing Organization Code

7. Author(s)
Michael Giles, W. Kyle Anderson, and Thomas W. Roberts

8. Performing Organization Report No.

10. Work Unit No.
533-02-01-03

9. Performing Organization Name and Address
NASA Langley Research Center
Hampton, VA 23665-5225

11. Contract or Grant No.

13. Type of Report and Period Covered
Technical Memorandum

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, DC 20546-0001

14. Sponsoring Agency Code

15. Supplementary Notes
Michael Giles: Massachusetts Institute of Technology, Cambridge, Massachusetts.
W. Kyle Anderson: Langley Research Center, Hampton, Virginia.
Thomas W. Roberts: Vigyan Research Associates, Inc., Hampton, Virginia.

16. Abstract
A new algorithm for the numerical solution of the Euler equations is presented. This algorithm is particularly suited to the use of unstructured triangular meshes, allowing geometric flexibility. Solutions are second-order accurate in the steady state. Implementation of the algorithm requires minimal grid connectivity information, resulting in modest storage requirements, and should enhance the implementation of the scheme on massively parallel computers. A novel form of upwind differencing is developed, and is shown to yield sharp resolution of shocks. Two new artificial viscosity models are introduced that enhance the performance of the new scheme. Numerical results for transonic airfoil flows are presented, which demonstrate the performance of the algorithm.

17. Key Words (Suggested by Author(s))
Euler equations, numerical algorithms, unstructured grids, upwind differencing

18. Distribution Statement
Unclassified - Unlimited
Subject Category 02

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	21	A03