J. Fluid Mech. (1984), *vol.* 148, *pp.* 45–71
Printed in Great Britain

Computation of flow around wings based on the Euler equations

By ARTHUR RIZZI AND LARS-ERIK ERIKSSON
FFA, The Aeronautical Research Institute of Sweden, S-161 11 Bromma, Sweden

(Received 20 October 1983 and in revised form 21 March 1984)

Inviscid transonic flows containing either strong shock waves or complex vortex structure call for the Euler equations as a realistic model. We present here a computational procedure, termed WINGA2, for solving the Euler equations for transonic flow around aircraft upon a 0–0 mesh generated by transfinite interpolation. An explicit time-marching finite-volume technique solves the flow equations and features a non-reflecting far-field boundary condition and an internal mechanism for temporal damping together with a model for artificial viscosity. The method's convergence to a steady state is studied, and results computed on the CYBER 205 vector processor are presented. The Euler equation model is found to predict the existence of a tip vortex created by flow separating from the downstream region of the tip of the ONERA M6 wing where the radius of curvature approaches zero.

1. Introduction

The great majority of inviscid compressible flows being simulated by computational methods for aerodynamic applications are steady. The standard model is the full potential equation and has proved to be a helpful tool in the design of aircraft. Difficulties arise, however, when discontinuities appear in the flow. Shock waves captured in this model agree with the Rankine–Hugoniot relations only if their strength is relatively weak, and vortex sheets cannot be captured at all but must be fitted, which in three dimensions is cumbersome except for simple situations. In these cases the Euler equations are a better model, because Rankine–Hugoniot shocks are captured correctly regardless of their strength. But, more importantly, vortex sheets and vorticity can also be captured as weak and genuine solutions. However, in the absence of any extra procedure to introduce it explicitly, the way vorticity is generated by this inviscid model remains to be explained conclusively. A number of hypotheses have been advanced so far, and we discuss two of them here which focus on an irreversible process, either physical or numerical, localized at the trailing edge. The development of numerical methods to solve the Euler equations is an active field right now; and some three-dimensional results have already been reported (Jameson & Baker 1984; Koeck & Neron 1984; Rizzi 1982). The applications of these methods range from the study of flow fields around military aircraft and missiles where shock waves are strong, to more complex non-uniform shear flows past wings, for example flow containing the slipstream of a propeller over a rectangular wing or the vorticity shed from a canard ahead of a delta wing (Erikson & Rizzi 1984).

In order to adapt readily to a general body-fitted grid we formulate the numerical method to solve the Euler equations in the integral-equation form or so-called finite-volume approach using centred second-order-accurate space differences. Topology of the boundaries is discussed, numerical implementations of boundary

conditions are developed, and a model for artificial viscosity is introduced. We are interested here only in steady flow, and true time accuracy is of no concern. But it is convenient to use the time variable as an iteration path to the steady state, sometimes referred to as a pseudo-unsteady approach. This then makes it natural to study the convergence of the iteration to steady state by the so-called semidiscrete analysis, namely investigation of the system of ordinary differential equations that results when all the spatial derivatives are replaced by differences but the time variable remains continuous. This system of course has to be solved for discrete time – the fully discrete problem – and we present a scheme for the time integration that is temporally dissipative and whose solution, once it reaches a steady state, satisfies the original steady difference operator exactly. The method is demonstrated with the computation of transonic flow past the M6 wing.

2. Euler equations for compressible flow

Continuum equations

Representing the conservation of mass and momentum in any arbitrary volume Ω of space, the Euler equations of motion in integral form are

$$\frac{\partial}{\partial t}\int_{\Omega} \boldsymbol{q}\,\mathrm{d}\,\mathrm{vol}+\iint_{\partial\Omega} \boldsymbol{H}\cdot\boldsymbol{n}\,\mathrm{d}s = 0, \tag{1}$$

where the dependent variables $\boldsymbol{q} = [\rho, \rho u, \rho v, \rho w]$ comprise a column vector containing as elements the density and rectangular components of momentum referred to a Cartesian system (x, y, z) fixed in space. The velocity of the fluid is

$$\boldsymbol{V} = u\boldsymbol{e}_x + v\boldsymbol{e}_y + w\boldsymbol{e}_z.$$

The quantity $\boldsymbol{H}(\boldsymbol{q})\cdot\boldsymbol{n} = [\boldsymbol{q}\boldsymbol{V} + (o, \boldsymbol{e}_x, \boldsymbol{e}_y, \boldsymbol{e}_z)\,p]\cdot\boldsymbol{n}$ represents the net flux of $\boldsymbol{q}$ transported across, plus the pressure p acting on, the closed surface $\partial\Omega$ that bounds the volume Ω with unit normal $\boldsymbol{n}$. We only treat flows with a globally constant stagnation enthalpy h_0, so that for a perfect gas the Bernoulli equation $p = \frac{1}{2}\kappa\rho(2h_0 - u^2 - v^2 - w^2)$, where $\kappa = (\gamma - 1)/\gamma$, completes system (1).

Dividing by Ω and then shrinking Ω to a point leads to the differential conservation law valid at that point if the partial derivatives are continuous there. Conceptually, however, we find it more appealing to discretize the finite-domain integral system (1) directly, the so-called finite-volume approach (Rizzi 1981), since the integral law formally does not exclude discontinuities from the interior of Ω. Our method therefore is a cell concept rather than a grid-point concept. The integral approach may be important for the correct capturing of discontinuities in the flow. It also lends itself to an obvious geometrical interpretation between the dependent and independent variables in the physical space and their counterparts in the computational space which makes the use of any arbitrary coordinate system more readily comprehensible.

Coordinate geometry

Perhaps the most attractive feature of the integral approach is its readiness to accommodate any type of coordinate system. Any convenient grid-generation technique can be used simply to pack a grid of cells in an orderly fashion so that they discretize the entire flow field. Although any arbitrary mesh can be used, the one we found most practical is hexahedral cells orchestrated by a three-dimensional coordinate system for which the body is aligned with one of the three coordinate surfaces. Such an arrangement facilitates the enforcement of the solid-wall boundary conditions. A

variety of body-aligned coordinate topologies can be formulated, but we prefer the so-called 0–0 mapping (a generalization of the classical conical system) because it focusses grid points along all edges of the wing. Figure 1 illustrates this type of mesh and shows our placement of the singular lines which are unavoidable in a three-dimensional body-aligned mapping. We work with the transfinite interpolation procedure to construct our 0–0 type mesh based on the curvilinear coordinates

$$X_I = X_I(x, y, z), \quad X_J = X_J(x, y, z), \quad X_K = X_K(x, y, z),$$

where the surface X_J = constant aligns with the wing. The non-orthogonal coordinates (X_I, X_J, X_K) define in physical space the edges of the mesh cells (figure 1*a*), the integers I, J, K are the corresponding directions in the computational space, and in the physical space the unit vectors $\boldsymbol{n}_I$, $\boldsymbol{n}_J$, $\boldsymbol{n}_K$ are normal respectively to the cell surfaces X_I, X_J, X_K equal to constants. Complete details on the construction of such a mesh as well as a discussion of the relative economy of resolution afforded by the 0–0 mapping is found in Eriksson (1982).

For the finite-volume method no global coordinate transformation needs to be specified. In fact the only details about the mesh that we transmit to the method are the three Cartesian coordinates of the eight vertices of every cell in the mesh. With this information it is not even necessary, as it is for a grid-point method, to formulate the local curvilinear coordinate system (X_I, X_J, X_K) in order to calculate the metric coefficients of the coordinate transformation. Instead the equivalent of these terms can be determined strictly by the principles of geometry. For example, altogether ten metric quantities are needed – the three components of each of the three surface areas $\boldsymbol{S}_I$, $\boldsymbol{S}_J$, $\boldsymbol{S}_K$ of a cell together with its volume Ω. If the four vertices defining a surface are coplanar its area is given exactly by one half the cross product of its diagonal line segments $\boldsymbol{S} = \frac{1}{2}\boldsymbol{l}_{31} \times \boldsymbol{l}_{42}$, and this is a good approximation even if it is non-planar.

The volume Ω is computed in the following way. Without restriction, a general hexahedron is composed of five tetrahedra (figure 2), each of whose volume is determined exactly by

$$T_{1236} = \tfrac{1}{6} \begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_6 & y_6 & z_6 & 1 \end{vmatrix},$$

where the integer subscripts on T_{1236} refer to the four vertices that define the tetrahedron. The volume of the hexahedron is then the sum of the volumes of these five tetrahedra.

Spatial finite-volume discretization

Since (1) is valid for any arbitrary volume it also holds locally for each individual cell ijk in the mesh, where the bounding surface $\partial\Omega_{ijk}$ now consists of the family of the three coordinate surface $\boldsymbol{S} = \{\boldsymbol{S}_I, \boldsymbol{S}_J, \boldsymbol{S}_K\}$ that delineate the hexahedral mesh cell (figure 1). In order to solve this continuum equation we must evaluate the integrals by some discrete approximation which then characterizes the class of the cell method. Here we consider only a single-point evaluation per cell for the dependent variables $\boldsymbol{q}$, so that by the mean-value theorem (1) becomes

$$\Omega_{ijk} \frac{\mathrm{d}\boldsymbol{q}_{ijk}}{\mathrm{d}t} + \delta[\boldsymbol{H}(\boldsymbol{q}) \cdot \boldsymbol{S}]_{ijk} = 0, \tag{2}$$

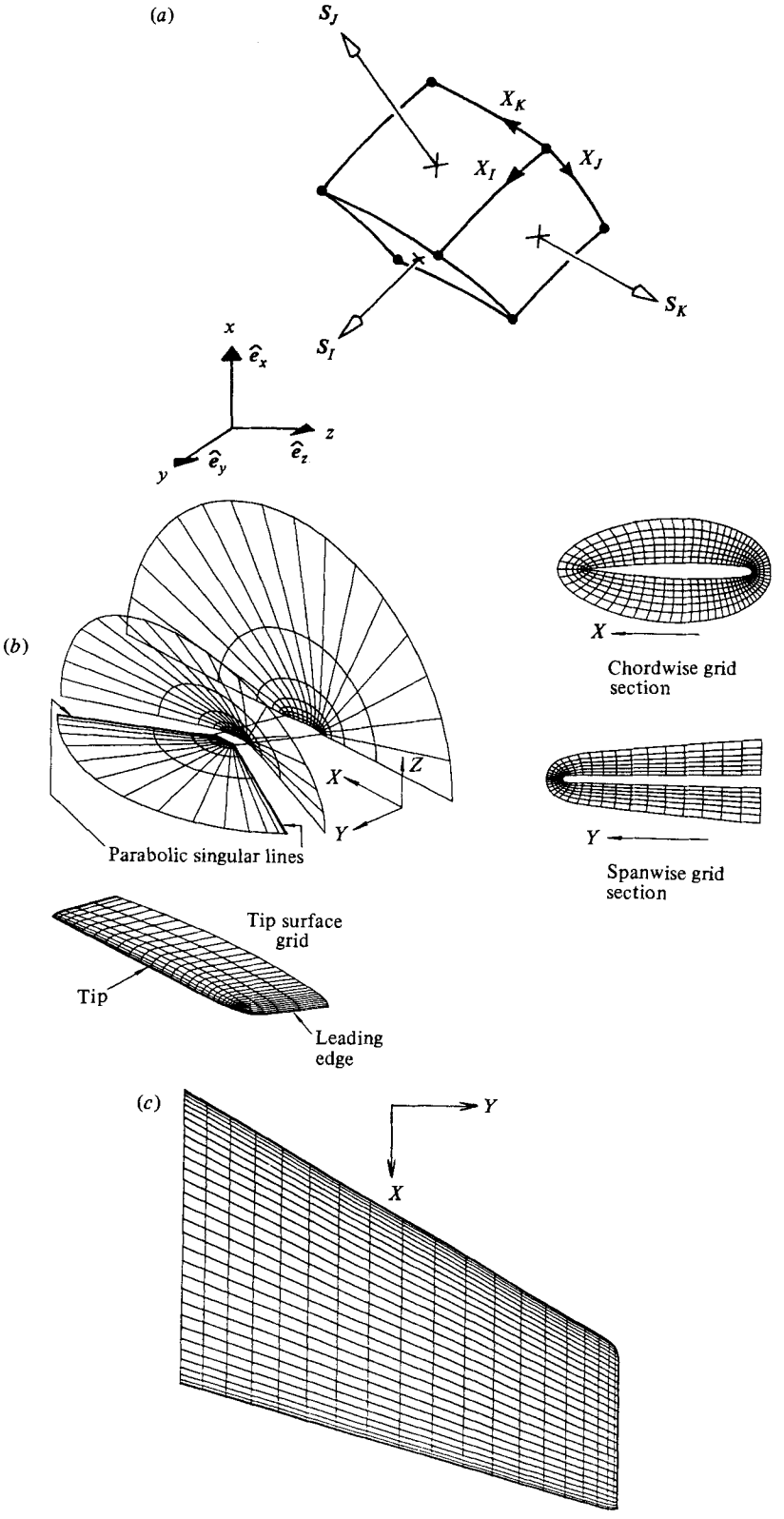

FIGURE 1 (*a–c*). For caption see facing page.

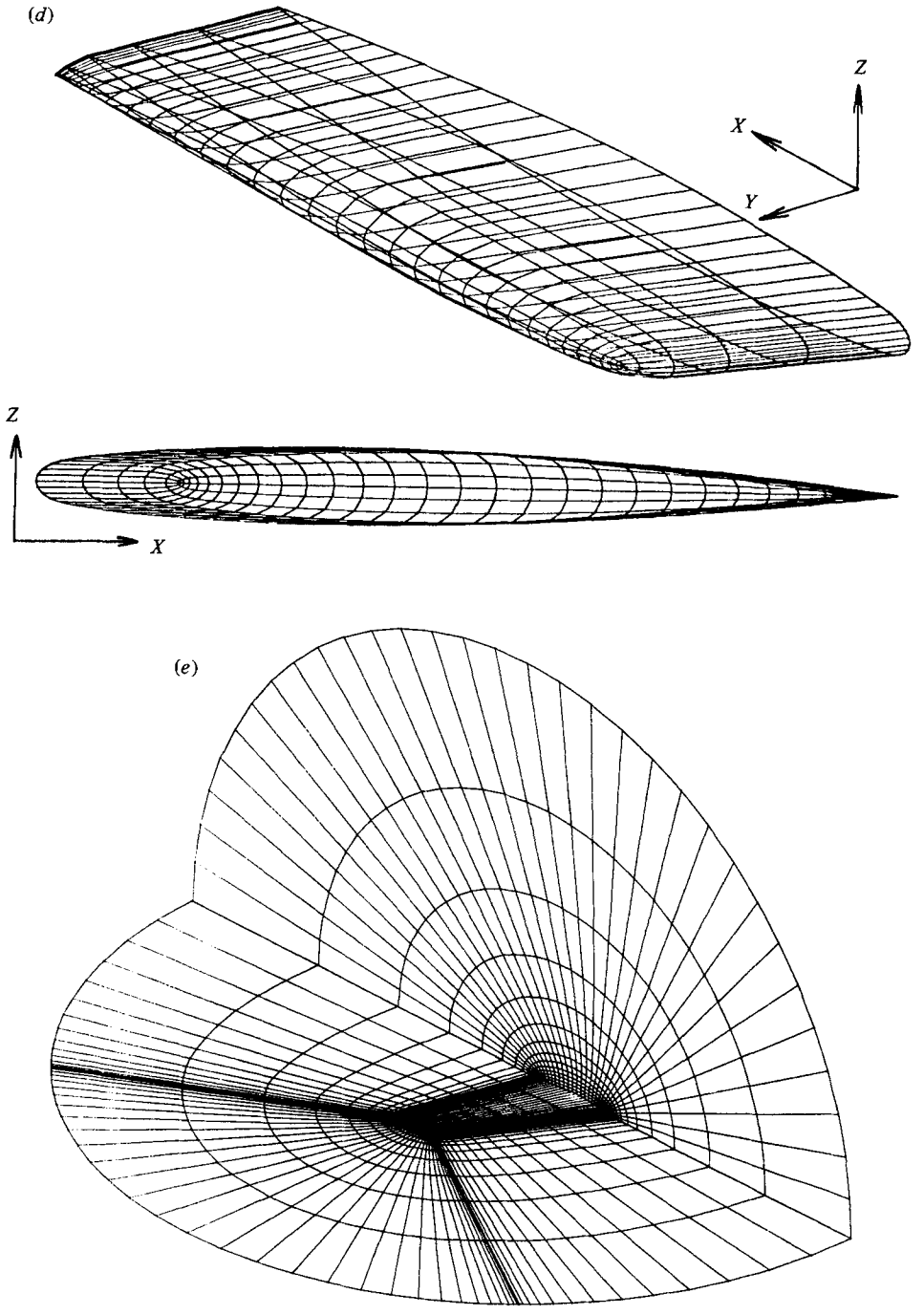

FIGURE 1. The 0–0 mesh topology wraps cells around all the edges of a large-aspect-ratio wing and offers good resolution near the wing. (*a*) The hexahedral cells of the mesh are defined by their eight vertices expressed in Cartesian coordinates (x, y, z). (*b*) Basic features of the 0–0 mesh: oval surfaces encircling all edges and two parabolic singular lines starting at the tip. (*c*) Vertical view of wing-surface discretization with a rounded leading corner at the tip. (*d*) Two different views of the tip region of the wing-surface mesh. The parabolic singular point at the leading-edge/tip corner is clearly visible. (*e*) Cut-away 3-dimensional view of the 0–0 mesh around the M6 wing shows the focusing of cells near the surface.

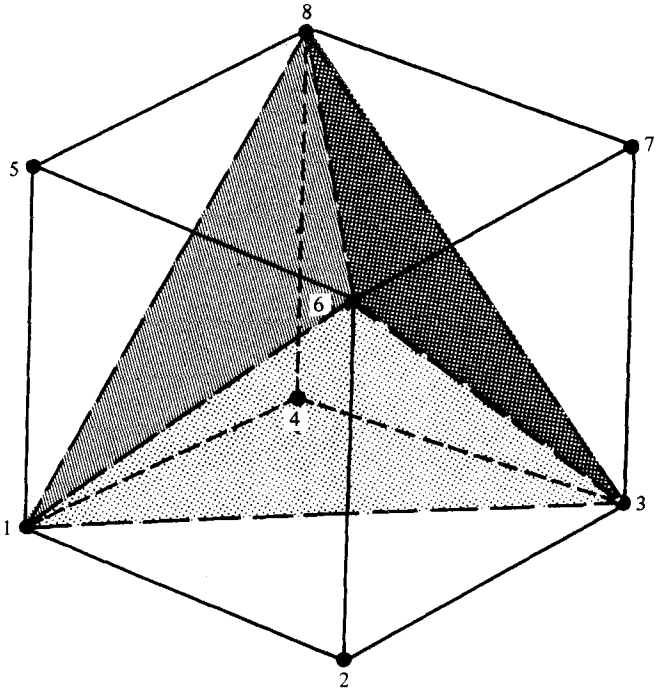

FIGURE 2. The volume of a hexahedral cell is the sum of the five constituent tetrahedra: $T_{1236}+T_{3867}+T_{3816}+T_{1685}+T_{1348}$.

where $\boldsymbol{q}_{ijk}$ is now interpreted as a volumetric average located at the centre of the cell, and $\boldsymbol{H}(\boldsymbol{q})\cdot\boldsymbol{S}$ is the corresponding flux evaluated at the surfaces $\boldsymbol{S}$. That $\boldsymbol{q}$ and $\boldsymbol{H}\cdot\boldsymbol{S}$ reside at different spatial positions is a central feature of the finite-volume concept. The three-dimensional undivided central-difference operator

$$\delta\psi_{ijk} = (\delta_I+\delta_J+\delta_K)\,\omega_{ijk} = (\psi_{i+\frac{1}{2},j,k}-\psi_{i-\frac{1}{2},j,k}) + (\psi_{i,j+\frac{1}{2},k}-\psi_{i,j-\frac{1}{2},k})+(\psi_{i,j,k+\frac{1}{2}}-\psi_{i,j,k-\frac{1}{2}})$$

expresses the net gain of flux into the cell and is fundamental to the conservation property and independent of any particular choice of spatial differencing.

Since $\boldsymbol{q}_{ijk}$ is located in the centre of the cell but $\boldsymbol{H}(\boldsymbol{q})$ must be expressed at its surfaces, some form of local interpolation of the neighbouring discrete values $\boldsymbol{q}$ must be devised and a numerical quadrature of the surface integrals performed in order to carry out the discrete solution of (2). It is the particular type of interpolating function and quadrature that defines the specific spatial-difference scheme of the method. In our case the simplest, and perhaps most natural, function is

$$[\boldsymbol{H}\cdot\boldsymbol{S}]_{ijk} = [\boldsymbol{H}(\mu_I\,\boldsymbol{q}_{ijk})\cdot\boldsymbol{S}_I+\boldsymbol{H}(\mu_J\,\boldsymbol{q}_{ijk})\cdot\boldsymbol{S}_j+\boldsymbol{H}(\mu_K\,\boldsymbol{q}_{ijk})\cdot\boldsymbol{S}_K], \tag{3a}$$

where μ is the averaging operator

$$\mu_I\,\psi_{ijk} = \tfrac{1}{2}(\psi_{i+\frac{1}{2},j,k}+\psi_{i-\frac{1}{2},j,k}).$$

An alternative to this, since each face of the cell ijk lies between two dependent variables, is to compute the flux separately for each of the two neighbouring dependent variables and then average the two results, i.e.

$$[\boldsymbol{H}\cdot\boldsymbol{S}]_{ijk} = [\mu_I\boldsymbol{H}(\boldsymbol{q}_{ijk})]\cdot\boldsymbol{S}_I+[\mu_J\boldsymbol{H}(\boldsymbol{q}_{ijk})]\,\boldsymbol{S}_J+[\mu_K\boldsymbol{H}(\boldsymbol{q}_{ijk})]\cdot\boldsymbol{S}_K. \tag{3b}$$

If the flux function $\boldsymbol{H}$ were linear, alternatives (3*a*) and (3*b*) would obviously be equivalent, but $\boldsymbol{H}$ is quadratic. We have tested both forms and found that in smooth regions of the flow the differences in the two results are imperceptible. They are larger at shock waves, however. We choose to work with (3*b*) because for the idealized case of one constant flow field ahead of and another behind a shock wave, and a cell face aligned to it, only scheme (3*b*) provides the correct jump in $\boldsymbol{q}$ across the shock. Equation (2) together with (3*b*) leads to a spatial-difference operator completely centred in all three coordinate directions, which is second-order-accurate in space if the variation in mesh size is reasonably smooth. These three-point differences lead to a simple program structure requiring no logic to decide whether to skew the differencing to one side or the other, which makes it amenable to a large degree of computer vectorization.

This finite-volume discretization bears some similarity to both the conventional finite-difference and finite-element discretizations. Its difference stencil is that of a finite-difference scheme, but it differs in that cell-averaged instead of point quantities are differenced, and as we shall see below this gives a significant distinction near a mesh singularity. Like the finite-element procedure, its formulation begins with the integral equation, and in fact we could present it in the context of a finite-element technique, but the resulting shape function is so peculiar that we think it warrants a presentation and name of its own.

Stability and accuracy at mesh singularities

Notice that even if the underlying mesh transformation is singular so that an edge of a cell contracts to a point, a surface collapses to a line or a point, or seven of its eight vertices become coplanar, this geometrical procedure still returns meaningful and accurate values for the areas and volume. The flux quantities can therefore be defined, and, since (2) is balanced in the interior of the cell where no coordinates are used, it remains finite even in the presence of these mesh singularities. And this is accomplished automatically without any special programming considerations. The same may not be true for the usual grid-point methods. Eriksson (1984) analysed the further question of whether mesh singularities destroy the spatial accuracy of finite-volume and finite-difference schemes or their stability as they step forward in time. He found that without any modification the finite-volume technique remains stable in the presence of a singularity, but its accuracy decreases to somewhere between first and second order in space. Without alteration the finite-difference scheme is unstable even if the singularity is straddled. Stability can be restored, however, if a limiting form of the difference scheme is derived at the singular point and implemented in the computer code.

Artificial-viscosity model

The purely convective difference operator $\mathrm{F_C}(\boldsymbol{q}_{ijk}) = -\delta(\boldsymbol{H}\cdot\boldsymbol{S})_{ijk}/\Omega_{ijk}$ in (2) suffers a number of drawbacks. It is well known, even for linear problems when the boundary conditions are unable to prevent it, that centred-differences admit as a solution so-called sawtooth or plus–minus waves, i.e. waves with the shortest wavelength $L \sim 2\Delta X$ that the mesh can support. This just reflects the fact that the truncation error of these schemes is entirely dispersive and not dissipative. When the problem is nonlinear there arises an aliasing phenomenon whereby short waves interact with each other, vanish, and reappear again as distorted long waves. But these defects in general could be dealt with satisfactorily by digital filtering techniques if it were not for further deficiencies in the differential Euler equations themselves. When

shocks are to be captured in a nonlinear flow field, the conservation equations admit non-unique weak solutions, and an entropy condition has to be supplied in order to obtain the physically correct weak solution (MacCormack & Paullay 1974). A standard way to invoke an entropy condition is to model the true physical process inside a shock by the addition of a small dissipation term to the convective differences. But even if the flow is smooth the fundamental question of the existence and uniqueness of a steady-state solution to these equations is not fully answered. In nonlinear transport there is a mechanism by which energy migrates from long-wavelength motion to progressively shorter and shorter scales until in reality it is removed from the flow by molecular viscosity. The differential Euler equations possess no such viscosity, so that this energy probably just piles up in the small scales. In the discrete representation this energy would migrate to the smallest scale resolvable on the mesh and then return transformed to large-scale motion via aliasing, which is clearly non-physical and would appear to make a steady state unattainable (Lomax 1982). Within the context of the inviscid-flow equations, our best recourse against all of these deficiencies, albeit crude, is to attenuate waves more and more severely as their wavelength decreases, so that none migrate out and alias back, but in such a way as not to alter completely the inviscid character of the solution. This is the so-called artificial-viscosity model. The idea of course is to mimic the short-wave dissipation by the real physical viscosity, and its justification is simply that in inviscid flow short-wave motion is of such low amplitude that whether removed or not it has no important effect on the overall flow character. In actual flow simulations this model is judged with a view to the crispness of shock profiles and the thinness of vortex sheets in weak solutions, and the amount of entropy produced or equivalently the variation in total pressure through regions of smooth flow.

We prefer to introduce dissipation into our system at the same time level as the transport process by adding to it damping terms whose magnitude lies in or below the range of the truncation error of the discrete approximation. Our total difference operator $\mathrm{F}(\boldsymbol{q})$ therefore consists of: (i) the convective part $\mathrm{F_C}(\boldsymbol{q})$ that results from discretizing the Euler equations in space by the centred finite-volume scheme, and (ii) the dissipative part $\mathrm{F_D}(\boldsymbol{q})$. The semidiscrete approximation (2) can then be written

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{q}_{ijk} = \mathrm{F_C}(\boldsymbol{q}_{ijk}) + \mathrm{F_D}(\boldsymbol{q}_{ijk}) = \mathrm{F}(\boldsymbol{q}_{ijk}). \tag{4}$$

The total discrete dissipative operator $\mathrm{F_D}(\boldsymbol{q}_{ijk})$ includes its own artificial boundary conditions, described below, and comprises both linear and nonlinear terms according to $\mathrm{F_D}(\boldsymbol{q}_{ijk}) = \mathrm{g}(\boldsymbol{q}_{ijk}) + \boldsymbol{D}\boldsymbol{q}_{ijk}$, where $\boldsymbol{D}$ is a constant matrix. The nonlinear expression $\mathrm{g}(\boldsymbol{q}_{ijk})$ is designed to provide dissipation at discontinuities, whereas the linear one is formulated to suppress spurious solutions (sawtooth waves) and to control the migration of energy from large to subgrid scales.

Nonlinear artificial-viscosity term

For all cells in the interior of the domain the nonlinear artificial viscosity is expressed by

$$g_{ijk} = \chi\{\delta_I[s_I(\boldsymbol{q}_{ijk})\,\delta_I] + \delta_J[s_J(\boldsymbol{q}_{ijk})\,\delta_J] + \delta_K[s_K(\boldsymbol{q}_{ijk})\,\delta_K]\}\,\boldsymbol{q}_{ijk}, \tag{5}$$

where χ is a constant in the range 0–0.1, and s_I, s_J and s_K are coefficients that depend on the solution field through the pressure p according to $s_I \propto |\delta_I^2 p_{ijk}|$, $s_J \propto |\delta_J^2 p_{ijk}|$ and $s_K \propto |\delta_K^2 p_{ijk}|$. These coefficients are normalized by their maximum value so that their magnitudes lie between 0 and 1. Their purpose is to sense non-smooth flow and

increase the filtering of large gradients so that in effect an entropy condition is enacted. This much is standard and is used in many methods. However, at cells adjacent to boundaries outside of which $\boldsymbol{q}_{ijk}$ cannot be defined naturally, (5) must be modified by what we call artificial boundary conditions, and here enters a degree of arbitrariness that causes the results of one method to differ markedly from another. We believe that the quadratic form $\boldsymbol{q}^{\mathrm{T}}\mathrm{H}\boldsymbol{q}$, where $\mathrm{H}\boldsymbol{q} = \mathrm{g}(\boldsymbol{q})$ for fixed s_I, s_J and s_K, provides a useful guideline for the appropriate conditions at such boundaries.

The purpose of the total dissipative operator is to drain off energy as time increases. How this is accomplished can be shown best by considering this locally linearized dissipative term $\mathrm{H}\boldsymbol{q}$ separate from the convective term. In the absence of $\mathrm{F_C}$, system (4) behaves as $\mathrm{d}\boldsymbol{q}/\mathrm{d}t = \mathrm{H}\boldsymbol{q}$. Let us define the 'energy' quantity $\boldsymbol{q}^2 = \boldsymbol{q}^{\mathrm{T}}\boldsymbol{q}$ of the discrete dependent variables. Since the time derivative of this energy should be negative, i.e. $\mathrm{d}\boldsymbol{q}^2/\mathrm{d}t = 2\boldsymbol{q}^{\mathrm{T}}(\mathrm{d}\boldsymbol{q}/\mathrm{d}t) \leqslant 0$, we determine the condition $\boldsymbol{q}^{\mathrm{T}}\mathrm{H}\boldsymbol{q} \leqslant 0$ for the quadratic form. Now at boundaries if, as an artificial boundary condition, we simply set the corresponding sensors s_I, s_J, s_K in (5) to zero, we find that the quadratic form

$$\begin{aligned}\boldsymbol{q}^{\mathrm{T}}\mathrm{H}\boldsymbol{q} &= \sum_{i=1}^{NI}\sum_{j=1}^{NJ}\sum_{k=1}^{NK} \boldsymbol{q}_{ijk}\mathrm{H}\boldsymbol{q}_{ijk} \\ &= -\chi \sum_{i=1}^{NI-1}\sum_{j=1}^{NJ}\sum_{k=1}^{NK} s_{I_{i+\frac{1}{2},jk}}(\boldsymbol{q}_{i+1,j,k}-\boldsymbol{q}_{i,j,k})^2 \\ &\quad -\chi \sum_{i=1}^{NI}\sum_{j=1}^{NJ-1}\sum_{k=1}^{NK} s_{J_{i,j+\frac{1}{2},k}}(\boldsymbol{q}_{i,j+1,k}-\boldsymbol{q}_{i,j,k})^2 \\ &\quad -\chi \sum_{i=1}^{NI}\sum_{j=1}^{NJ}\sum_{k=1}^{NK-1} s_{K_{i,j,k+\frac{1}{2}}}(\boldsymbol{q}_{i,j,k+1}-\boldsymbol{q}_{i,j,k})^2\end{aligned}$$

is always negative, and hence energy dissipates even in the boundary cells. We therefore believe this to be a good choice of artificial boundary condition for the nonlinear dissipation.

Linear artificial-viscosity term

Our model for linear artificial viscosity uses the fourth-difference operator

$$\mathrm{D}\boldsymbol{q}_{ijk} = -\Upsilon(\delta_I^4+\delta_J^4+\delta_K^4)\boldsymbol{q}_{ijk} \tag{6}$$

at all interior cells where Υ is a constant in the range 0–0.02. But at boundaries we also must alter this expression by some suitable boundary procedure, and we seek to do so in a way that guarantees positive dissipation in all cells. Guided again by its quadratic form, we use no data outside the computational domain, but instead incorporate non-centred differences for the boundary cells together with scheme (6) at the interior ones in order to obtain the total discrete linear dissipative operator D with the property

$$\begin{aligned}\boldsymbol{q}^{\mathrm{T}}\mathrm{D}\boldsymbol{q} &= \sum_{i=1}^{NI}\sum_{j=1}^{NJ}\sum_{k=1}^{NK} \boldsymbol{q}_{ijk}\mathrm{D}\boldsymbol{q}_{ijk} \\ &\quad -\Upsilon \sum_{i=2}^{NI-1}\sum_{j=1}^{NJ}\sum_{k=1}^{NK} (\boldsymbol{q}_{i+1,j,k}-2\boldsymbol{q}_{i,j,k}+\boldsymbol{q}_{i-1,j,k})^2 \\ &\quad -\Upsilon \sum_{i=1}^{NI}\sum_{j=2}^{NJ-1}\sum_{k=1}^{NK} (\boldsymbol{q}_{i,j+1,k}-2\boldsymbol{q}_{i,j,k}+\boldsymbol{q}_{i,j-1,k})^2 \\ &\quad -\Upsilon \sum_{i=1}^{NI}\sum_{j=1}^{NJ}\sum_{k=2}^{NK-1} (\boldsymbol{q}_{i,j,k+1}-2\boldsymbol{q}_{i,j,k}+\boldsymbol{q}_{i,j,k-1})^2,\end{aligned} \tag{7}$$

which automatically ensures that the dissipation is greater than or equal to zero (for $\Upsilon > 0$). The attractive feature of this artificial viscosity model, easily seen in (7), is that, if $\boldsymbol{q}_{ijk}$ is bilinear in i,j and k, the total operator D acting on $\boldsymbol{q}_{ijk}$ always returns zero, even at the boundaries. Compare this to the second-difference operator H that returns zero only if $\boldsymbol{q}_{ijk}$ is a constant. The details of the construction of D will be presented in a forthcoming paper.

Boundary conditions

A particular steady flow field is determined by the conditions imposed upon it at its boundaries, and usually the stability and accuracy of the discrete conditions are more difficult to analyse than the difference scheme itself. This means that in general the theory of boundary conditions for numerical computations is more empirical. In our case of a 0–0 mesh conforming to a wing/body combination, boundary conditions are enforced at the six outer surfaces of the computational space (figure 3). There are three distinct types: flow into or out of the far field, periodic conditions across coordinate cuts, and conditions on solid walls.

Inflow/outflow boundary

With a 0–0 mesh, flow in the far field enters and leaves through the outermost J-surface. This is an artificial boundary in the sense that the actual flow in the physical domain is open, whereas the computational space must for practical reasons be closed. The numerical conditions, therefore, ideally should allow phenomena generated in the computational domain to pass through the boundary without undergoing significant distortion and without influencing the interior solution. In this way the maximum amount of transient energy escapes from the field, and the time-dependent solution can converge to the steady state. Engqvist & Majda (1977) present a mathematical theory for the practical application of local absorbing boundary conditions at artificial boundaries.

One of the six faces of cell i, JL, k located in the mesh layer JL farthest from the body is coincident with the outer boundary; call the surface area of that face $\boldsymbol{S}_{JL}$. The edges of the cell define the local curvilinear coordinate system (X_I, X_J, X_K), where the positive X_J direction points from the outer boundary surface into the domain, the other two being tangent to the surface. In this system the differential conservation equations equivalent to (1) are

$$\frac{\partial}{\partial t}\boldsymbol{q} + \boldsymbol{A}\frac{\partial \boldsymbol{q}}{\partial X_J} + \boldsymbol{B}\frac{\partial \boldsymbol{q}}{\partial X_I} + \boldsymbol{C}\frac{\partial \boldsymbol{q}}{\partial X_K} = 0,$$

where

$$\boldsymbol{A} = \frac{\partial(\boldsymbol{H}\cdot\boldsymbol{S}_{JL})}{\partial \boldsymbol{q}}, \quad \boldsymbol{B} = \frac{\partial(\boldsymbol{H}\cdot\boldsymbol{S}_{I})}{\partial \boldsymbol{q}}, \quad \boldsymbol{C} = \frac{\partial(\boldsymbol{H}\cdot\boldsymbol{S}_{K})}{\partial \boldsymbol{q}}$$

are the Jacobian matrices of the flux function. The matrix $\boldsymbol{A}$ works out to be

$$\boldsymbol{A} = \begin{bmatrix} 0 & \alpha & \beta & \epsilon \\ \alpha c^2 - uU + \kappa\alpha V^2 & \alpha(1-\kappa)u + U & \beta u - \kappa\alpha v & \epsilon u - \kappa\alpha w \\ \beta c^2 - vU + \kappa\beta V^2 & \alpha v - \kappa\beta u & \beta(1-\kappa)v + U & \epsilon v - \kappa\beta w \\ \epsilon c^2 - wU + \kappa\epsilon V^2 & \alpha w - \kappa\epsilon u & \beta w - \kappa\epsilon v & \epsilon(1-\kappa)w + U \end{bmatrix},$$

where

$$\alpha = \boldsymbol{S}_{JL}\cdot\boldsymbol{e}_x; \quad \beta = \boldsymbol{S}_{JL}\cdot\boldsymbol{e}_y; \quad \epsilon = \boldsymbol{S}_{JL}\cdot\boldsymbol{e}_z,$$

$$U = \alpha u + \beta v + \epsilon w; \quad V^2 = \boldsymbol{V}\cdot\boldsymbol{V},$$

and c is the local speed of sound.

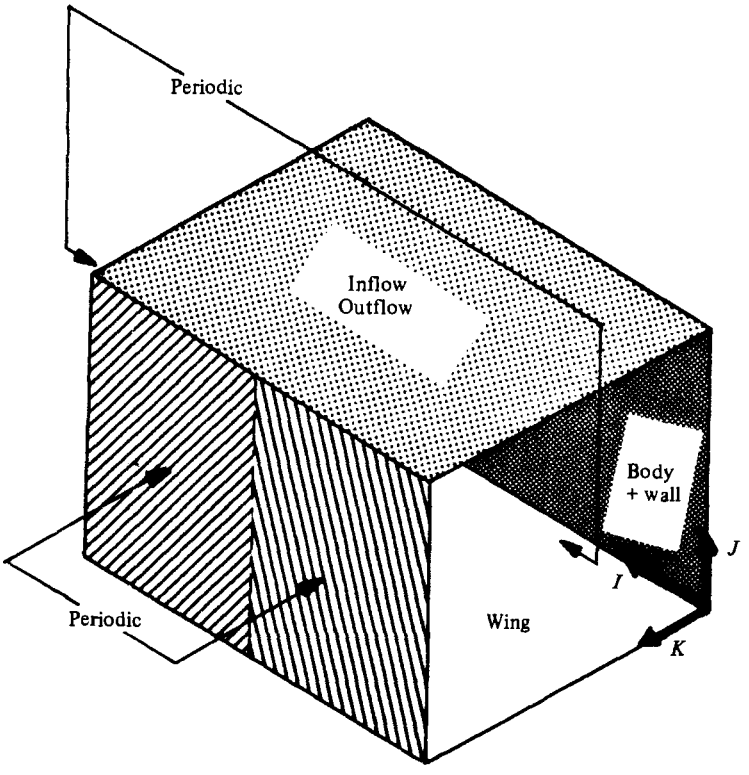

FIGURE 3. Types of boundary conditions on the six outer surfaces of the computational space IJK resulting from the 0–0 grid mapping.

The First Approximation in the hierarchical theory of Engqvist & Majda amounts to specifying the characteristic variables of the corresponding one-dimensional problem which is well posed and maximally dissipative. Since we work only with their First Approximation, it is easier to grasp the idea if we present our boundary condition in a heuristic development based directly on the characteristic variables instead of the formal theory. The presentation for our 4×4 system mirrors the one given by Gottlieb & Gustafsson (1976) for the 3×3 system, but is more general because of our non-orthogonal coordinates.

We seek the characteristic variables of the corresponding one-dimensional problem local to a given cell

$$\frac{\partial}{\partial t}\boldsymbol{q} + \boldsymbol{A}\frac{\partial \boldsymbol{q}}{\partial X_J} = 0, \tag{8}$$

which means that we focus on a particular set of characteristic planes, those whose normals point along X_J and whose slopes in time are the eigenvalues λ of $\boldsymbol{A}$. Solving $\det(\boldsymbol{A} - \lambda \boldsymbol{I}) = 0$, we find

$$\lambda_1 = U, \quad \lambda_2 = U, \quad \lambda_3 = U - a_+, \quad \lambda_4 = U - a_-,$$

where

$$a_\pm = \tfrac{1}{2}\kappa U \pm [\tfrac{1}{4}\kappa^2 U^2 + c^2(\alpha^2 + \beta^2 + \epsilon^2)]^{\frac{1}{2}}.$$

The left and right eigenvectors associated with these four eigenvalues make up the rows and columns of the transformation matrices $\boldsymbol{T}^{-1}$ and $\boldsymbol{T}$ respectively that diagonalize (8):

$$\frac{\partial}{\partial t}\boldsymbol{\phi} + \boldsymbol{\Lambda}\frac{\partial \boldsymbol{\phi}}{\partial X_J} = 0,$$

where $\boldsymbol{\phi} = \boldsymbol{T}^{-1}\boldsymbol{q}, \quad \boldsymbol{\Lambda} = \boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{T} = \mathrm{diag}\{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$

After the intermediate variables

$$\tilde{U} = \beta u + \alpha v, \quad \tilde{V} = -\epsilon v + \beta w, \quad \tilde{W} = \epsilon u - \alpha w,$$

$$\xi = \alpha^2 + \beta^2 + \epsilon^2, \quad Q_\pm = \kappa U - a_\pm, \quad R_\pm = \epsilon Q_\pm - \zeta \kappa w, \quad P_\pm = \kappa w a_\pm + \epsilon c^2$$

have been defined for the sake of simplification, we find

$$\boldsymbol{T} = \begin{bmatrix} \kappa\tilde{U} & 0 & R_+ & R_- \\ \kappa v U - \beta(\kappa V^2 + c^2) & \tilde{V} & uR_+ + \alpha P_+ & uR_- + \alpha P_- \\ -\kappa u U + \alpha(\kappa V^2 + c^2) & \tilde{W} & vR_+ + \beta P_+ & vR_- + \beta P_- \\ 0 & \tilde{U} & wR_+ - \kappa(\alpha u + \beta v)\, a_+ - (\alpha^2 + \beta^2)\, c^2 & wR_- - \kappa(\alpha u + \beta v)\, a_- - (\alpha^2 + \beta^2)\, c^2 \end{bmatrix}$$

and

$$\boldsymbol{T}^{-1} = \begin{bmatrix} \dfrac{\xi V^2 - U^2}{d_1} & \dfrac{-\epsilon\tilde{W} + \beta\tilde{U}}{d_1} & \dfrac{\epsilon\tilde{V} - \alpha\tilde{U}}{d_1} & \dfrac{\alpha\tilde{W} - \beta\tilde{V}}{d_1} \\ [\kappa w(U^2 - \xi V^2) + c^2(\epsilon U - \xi w)]/d_2 & [+\kappa w(-\beta\tilde{U} + \epsilon\tilde{W}) - \alpha\epsilon c^2]/d_2 & [\kappa w(\alpha\tilde{U} - \epsilon\tilde{V}) - \beta\epsilon c^2]/d_2 & [-\kappa w(\alpha\tilde{W} - \beta\tilde{V}) + (\alpha^2 + \beta^2)\, c^2]/d_2 \\ \dfrac{R_+}{d_3} & \dfrac{\kappa\xi V^2 - (U + a_+)\, Q_+}{d_3} & \dfrac{-\kappa\xi u + \alpha Q_+}{d_3} & \dfrac{-\kappa\xi v + \beta Q_+}{d_3} \\ \dfrac{R_-}{d_4} & \dfrac{\kappa\xi V^2 - (U + a_-)\, Q_-}{d_4} & \dfrac{-\kappa\xi u + \alpha Q_-}{d_4} & \dfrac{-\kappa\xi v + \beta Q_-}{d_4} \end{bmatrix}$$

The factors d_1, d_2, d_3 and d_4 in the denominators are normalizing coefficients so that $\boldsymbol{T}^{-1}\boldsymbol{T}$ equals the unit matrix.

For the one-dimensional case it is well known that the number of conditions to be imposed in a cell at the outer boundary should equal the number of characteristic directions that enter the computational domain. Four typical cases are depicted in figure 4. With subsonic inflow our implementation is to set the three ingoing characteristic variables $\phi^{(1)}$, $\phi^{(2)}$ and $\phi^{(3)}$ to their free-stream values, linearly extrapolate the fourth $\phi^{(4)}$ from the computational field, and then solve for the original unknowns $\boldsymbol{q} = \boldsymbol{T}\boldsymbol{\phi}$. At outflow it is $\phi^{(3)}$ that is given the values of undisturbed flow, and $\phi^{(1)}$, $\phi^{(2)}$ and $\phi^{(4)}$ are extrapolated from the computational field.

Coordinate cuts

Conditions on these boundaries are the least troublesome since at a cut the physical space folds on to itself and the condition on the flow at the computational boundary is periodicity. We remark that in the 0–0 topology these boundaries occur conveniently at the trailing edge and tip of the wing (figure 3).

Solid walls

For inviscid flow the imposition of the boundary condition on the surface of the aerodynamic vehicles of interest in our work possesses two different but related aspects. The first is the usual one that no flow is allowed through a solid wall, and the second is the so-called Kutta condition, which dictates that the flow separates from a sharp trailing edge and can be a source of vorticity.

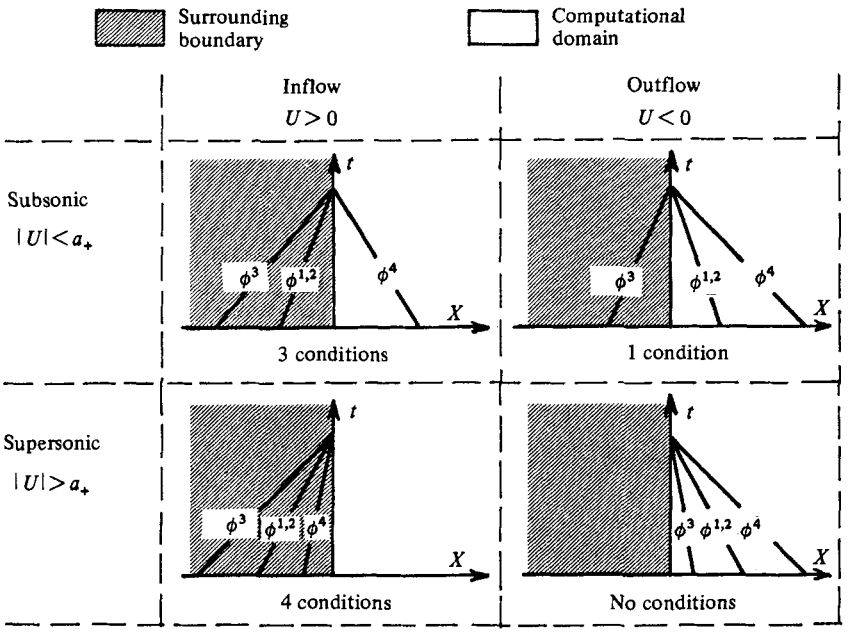

FIGURE 4. The number of boundary conditions at inflow and outflow based on the ingoing characteristic variables ϕ^m, $m = 1, \ldots, 4$ (cf. (8)).

Zero flux transport

The physical condition of zero transport $\boldsymbol{V}\cdot\boldsymbol{n} = 0$ applies to two surfaces, the wing $J = 1$ and the fuselage which merges together with the wall of symmetry $K = 1$ (see figure 3). Since the computational cells are aligned to both these surfaces, the physical condition reduces the dependence of $\boldsymbol{H}(\boldsymbol{q})_{\text{body}}$ to $\boldsymbol{H}(p)_{\text{body}}$, and we are forced to determine a value for p on the vehicle surface by numerical means, usually by differencing some auxiliary equation in order to relate values in the field to those on the surface. Our procedure to obtain an estimate of p_{body} from the interior solution has been described before in the general case (Rizzi 1978). It is valid at both the wing and the body-plus-wall surfaces, and we summarize it briefly. The basis of our auxiliary relation for p_{body} begins with the streamline differentiation of the physical condition $(\partial/\partial t + \boldsymbol{V}\cdot\mathbf{grad})\,(\boldsymbol{V}\cdot\boldsymbol{n}) \equiv 0$, where $\boldsymbol{n}$ is the unit vector normal to either the wing or fuselage. This expression, when combined with the inner product of the quasilinear momentum equation and $\boldsymbol{n}$ and rearranged, becomes, for a stationary body,

$$\rho\boldsymbol{V}\cdot(\boldsymbol{V}\cdot\mathbf{grad})\,\boldsymbol{n} = \boldsymbol{n}\cdot\mathbf{grad}\,p, \tag{9}$$

where

$$\mathbf{grad} \equiv \frac{\boldsymbol{S}_I}{|\boldsymbol{S}_I|}\frac{\partial}{\partial X_I} + \frac{\boldsymbol{S}_J}{|\boldsymbol{S}_J|}\frac{\partial}{\partial X_J} + \frac{\boldsymbol{S}_K}{|\boldsymbol{S}_K|}\frac{\partial}{\partial X_K}$$

and relates ρ, $\boldsymbol{V}$ and the geometry of the surface to the normal derivative of p. When it is differenced to formally first-order accuracy the pressure on the surfaces is deduced from the interior values.

Trailing-edge vorticity generation

Because a velocity potential satisfies $\mathbf{curl}\ \boldsymbol{V} = 0$ identically, the exact solution of potential flow past a wing as well as any finite-difference approximation to it cannot admit distributed vorticity in the field. For the same reason numerical solutions to

the potential equation, no matter how crude, must be loss-free (isentropic), i.e. the flow can undergo only reversible processes. Numerical methods based on a potential therefore introduce a vortex sheet (usually rigid and planar) in the flow field in order to model the vorticity generated by the wing so that non-zero values for lift and induced drag can be realized. However, at a workshop (Rizzi 1981) on transonic flow those methods which solved the Euler equations for flow past an airfoil section but which did *not* implement any explicit procedure or Kutta condition to introduce vorticity still obtained accurate solutions with the correct lift. It was argued that the Euler equations, unlike the potential equation, admit vorticity and vortex sheets as genuine and weak solutions respectively, so that, if somehow introduced into the flow, vortex sheets can be 'captured' in the same sense that shock waves can be captured, and a Kutta condition need not be enforced explicitly. Since then, further examples of computed three-dimensional flow past a finite wing have been reported for which a Kutta condition was not explicitly imposed but the correct lift was still obtained.

However, the question of how in the absence of a Kutta condition the separating vortex sheet is initially introduced into the flow field remains inconclusively answered. The creation of vorticity is usually associated with irreversible flow processes. In the context of the differential Euler equations such processes occur only across shock waves. And our first explanation for the source of vorticity focused on an unsteady mechanism in the Euler equations, with the nature of an unsteady Prandtl–Meyer expansion, which, if the flow were to remain attached around a sharp trailing edge, would generate a shock and entropy that ultimately force the separation point to the trailing edge (Rizzi 1982). But additional losses can enter a discrete solution to the Euler equations through numerical error, depending upon the mesh size, long before gradients in the flow have steepened into something we would call a shock. They can occur even for expansion gradients. A good example of the latter is the one or two per cent loss in total pressure at the leading edge of an airfoil that is commonly observed in numerical solutions to the Euler equations for transonic flow. Unlike the potential solution, which is *a priori* loss-free, only accurate solutions to the Euler equations conserve entropy along streamlines of smooth flow. But this does not necessarily mean that our postulated shock mechanism must be modelled accurately to generate vorticity at the trailing edge. The most convincing argument why it need not be has been recently advanced by E. H. Hirschel (1984 private communication). In studying the behaviour of potential flow past a thin ellipsoid at incidence, Hirschel & Fornasier (1984) point out that, right after a streamline divides into upper- and lower-surface components at the leading edge, substantial shear develops between them (see figure 2 of Hirschel & Fornasier 1984). For potential flow this shear of course cancels out again by the reverse process in a region near the trailing edge where the two components reunite. As its curvature increases and the trailing edge becomes sharper, this region shrinks. But in a numerical solution of the Euler equations, which must approximate this process accurately in order to remain loss-free, this shrinkingly smaller region eventually disappears from the resolution of the usual mesh. And the computed flow then leaves the trailing edge in shear. Since this shear originates from the forward portion of the wing, where we assume that the mesh does resolve the inviscid flow accurately, we can expect to obtain a realistic level of circulation in the solution. But, if the trailing edge is not sufficiently sharp, the numerical solution may begin to approximate potential flow more accurately at the trailing edge and contain less circulation. We leave this discussion for now, however sketchy it may be, but it can and should be tested by numerical experiments in order to explore under what conditions a solution to the Euler equations can be forced to depart from a potential solution.

4. Time integration and convergence to steady state

With the above boundary conditions and artificial-viscosity model now included in $\mathrm{F} = \mathrm{F_C} + \mathrm{F_D}$, our complete difference operator, the problem (2) we want to solve becomes, with spatial indices ijk suppressed,

$$\frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}t} = \mathrm{F}(\boldsymbol{q}). \tag{10}$$

For a given mesh size one can look upon this as a large system of ordinary differential equations, the so-called semidiscrete representation. Ultimately of course the problem must be solved in discrete time, but it is instructive to look at this form (10) first. Our goal is to integrate (10) forward in time until a steady state is reached, but without concern for time accuracy. The central issue to be addressed is the stability of the integration, not in the strict formal sense but in the more limited one of convergence to steady state, that is to say we do not attempt to establish a uniform stability bound on the time integration in the limit of finer and finer meshes. Instead, what we ask of any candidate used to march (10) to a steady state $\boldsymbol{q}^*$ for a given grid are two criteria: that after many time steps the transients $\mathrm{d}\boldsymbol{q}^*/\mathrm{d}t$ become vanishingly small, and that $\boldsymbol{q}^*$ satisfy the steady discrete operator $\mathrm{F}(\boldsymbol{q}^*) = 0$ independently of the time step size.

Equation (10) is nonlinear, and it is possible to study these questions only after the complete operator F has been locally linearized. Think of $\boldsymbol{q}$ as some transient perturbation superposed upon the steady state $\boldsymbol{q}^*$. The linearization of (10) then leads to a homogeneous equation for the transients,

$$\frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}t} = \mathrm{A}\boldsymbol{q}, \quad \text{where} \quad \mathrm{A} = \frac{\partial \mathrm{F}}{\partial \boldsymbol{q}}, \tag{11}$$

and $\boldsymbol{q}$ decays if all of the eigenvalues λ of A lie to the left of the imaginary axis. In general the order of A is large, and we cannot compute its entire eigenvalue spectrum. It is possible, however, to get a reasonable picture of the spectrum in the light of three different approximations.

The first is the usual Neumann analysis, and tells us something about the character of the convective difference. Assume for the moment that F is linear, purely convective (the artificial viscosity coefficients χ and Υ are set to zero), and changes occur in only one direction X which is free of boundaries so that $\mathrm{F} = \mathrm{A} = -a(\boldsymbol{q}_{j+1} - \boldsymbol{q}_{j-1})/2\Delta X$. Since this analysis is only local, the mesh can be considered uniform, and the finite-volume discretization then reverts to the standard central-difference expression. If we ask what happens to a Fourier mode $\boldsymbol{q} = \boldsymbol{q}\,\mathrm{e}^{\mathrm{i}\phi}$ of wavelength $L(\phi = 2\pi X/L)$ obeying (11), we obtain $\mathrm{d}\boldsymbol{q}/\mathrm{d}t = -[\mathrm{i}(a \sin\phi)/\Delta X]\boldsymbol{q}$, so that under these assumptions the Fourier eigenvalues $\lambda = -\mathrm{i}(a \sin\phi)/\Delta X$ are purely imaginary, and continuous-time transients will oscillate indefinitely, indicating that the centred difference is non-dissipative. But, on integrating this equation by the following three-stage two-step scheme in parameter $\theta = \frac{1}{2}$ or 1,

$$\left.\begin{aligned} \boldsymbol{q}_0 &= \boldsymbol{q}^n, \\ \boldsymbol{q}' &= \boldsymbol{q}_0 + \Delta t\,\mathrm{F}(\boldsymbol{q}_0), \\ \boldsymbol{q}'' &= \boldsymbol{q}_0 + \Delta t[(1-\theta)\,\mathrm{F}(\boldsymbol{q}_0) + \theta\,\mathrm{F}(\boldsymbol{q}')], \\ \boldsymbol{q}^{n+1} &= \boldsymbol{q}_0 + \Delta t[(1-\theta)\,\mathrm{F}(\boldsymbol{q}_0) + \theta\,\mathrm{F}(\boldsymbol{q}'')], \end{aligned}\right\} \tag{12}$$

over the discrete time step from $\boldsymbol{q}^n$ to $\boldsymbol{q}^{n+1}$, we find that the solution

$$\boldsymbol{q}^{n+1} = \{I + \Delta t\,\mathrm{A} + \theta(\Delta t\,\mathrm{A})^2 + \theta^2(\Delta t\,\mathrm{A})^3\}\boldsymbol{q}^n = \mathrm{C}\boldsymbol{q}^n \tag{13}$$

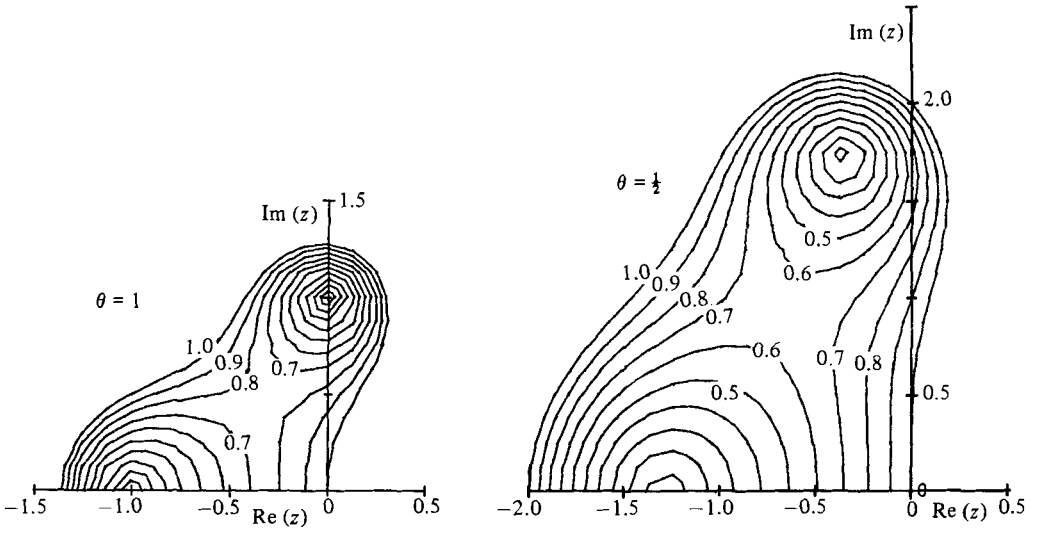

FIGURE 5. Contours of constant amplitude of the amplification factor $|\sigma|$ of scheme (12).

decays if the magnitude of the maximum eigenvalue

$$\sigma = 1 + z + \theta z^2 + \theta z^3, \quad z = \Delta t \lambda, \tag{14}$$

of C is less than one, the well-known CFL condition that determines the size of the time step Δt. Contour levels of $|\sigma|$ in intervals of 0.1 are plotted versus z in the complex plane of figure 5. Since all λ are imaginary we see that the CFL condition is encountered by the mode ($L = 4\Delta X$) associated with the eigenvalue of largest modulus, and is satisfied if $\Delta t\,|\lambda|_{\max}$ is less than the CFL number, 2 for $\theta = \frac{1}{2}$ and 1.2 for $\theta = 1$. But, except for very long waves or extremely short ones ($L \approx 2\Delta X$), all modes are temporally damped by the time integration, and to a greater degree as θ goes from $\frac{1}{2}$ to 1. Compare this to the corresponding plot (figure 6) for the fourth-order time-accurate Runge–Kutta scheme, which offers significantly less temporal damping but a larger stability bound (CFL = 2.8).

Under the same assumptions, but now in three dimensions, a similar Fourier analysis of scheme (12) indicates, and it is confirmed by actual numerical tests, that the stability limit on the step size is $\Delta t \leqslant \mathrm{CFL}\,\min_{ijk}(\Delta t_l)$, where

$$\Delta t_l = \frac{\Omega_{ijk}}{\frac{1}{2}\kappa(Q_I + Q_J + Q_K) + [\frac{1}{4}\kappa^2(Q_I^2 + Q_J^2 + Q_K^2) + c^2(S_I^2 + S_J^2 + S_K^2)]^{\frac{1}{2}}}, \tag{15}$$

with

$$Q_I = |\boldsymbol{V}\cdot\boldsymbol{S}_I|, \quad Q_J = |\boldsymbol{V}\cdot\boldsymbol{S}_J|, \quad Q_K = |\boldsymbol{V}\cdot\boldsymbol{S}_K|, \quad c^2 = \tfrac{1}{2}\kappa(2h_0 - u^2 - v^2 - w^2).$$

The preceding Fourier analysis tells us a good deal about the character of A under the simplifying assumptions of no boundaries and zero artificial viscosity, but we would like to know how it changes as these assumptions are gradually removed. Consider boundaries first, since the release of energy through them is known to play a crucial role in the convergence to steady state. Lomax, Pulliam & Jespersen (1981) have taken this simplified linear one-dimensional example, added boundary conditions at each end, and obtained the discrete operator A. Since in one dimension the order

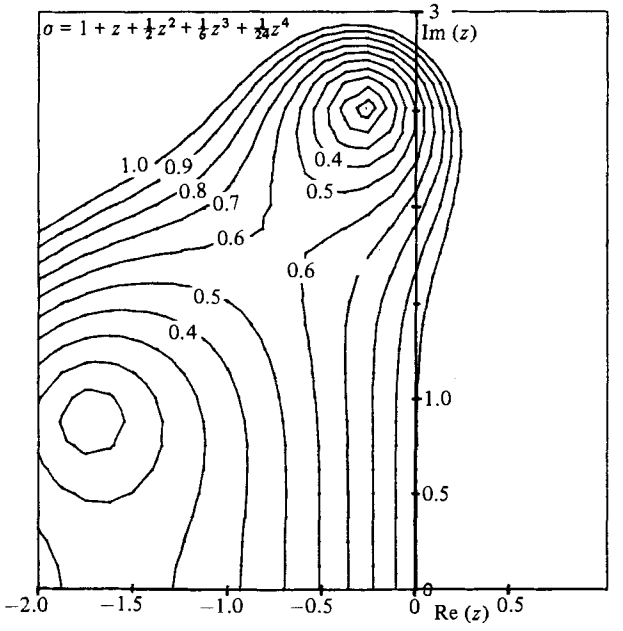

FIGURE 6. Contours of constant amplitude of the amplification factor $|\sigma|$ of the fourth-order Runge–Kutta scheme.

of A is low, they were able to calculate the entire eigenvalue spectrum of A, and found that the effect of introducing boundary conditions is to shift the eigenvalues a small distance horizontally to the left of the imaginary axis. The boundary conditions therefore provide a very important damping of transients.

For the third step of our study we consider a much more realistic problem, the two-dimensional counterpart of (11) where the complete discrete operator F includes the boundary conditions and artificial viscosity described above and is formed on a non-uniform 0-type mesh for transonic flow around an airfoil. The matrix $\boldsymbol{A}$, the local linearization of F around the state obtained by scheme (12) after 15 steps from the free stream, is too large now to determine the full eigenvalue spectrum exactly, but by a Krylov subspace method we have been able to compute a signature of the spectrum (Eriksson & Rizzi 1984). Figure 7 (*a*) presents the approximate spectrum that we computed for $\boldsymbol{A}$ and confirms the expected shift to the left by the boundary conditions and artificial viscosity. But what is disturbing is the eigenvalue very close to the origin, not only because it is damped very slightly but because it indicates that $\boldsymbol{A}$ is poorly conditioned. This situation is a direct result of the non-uniformity of the mesh, and can be alleviated by the well-known technique of advancing the solution with the local Δt_l instead of the minimum time step. To demonstrate its effect we scale the matrix $\boldsymbol{A}$ by multiplying its lth row by Δt_l and present the resulting spectrum in figure 7 (*b*). Notice that the smallest-modulus eigenvalue now is also shifted to the left, the overall condition is improved, and the discrete-time solution of (12) can be expected to decay to a steady state. In Eriksson & Rizzi (1984) we further show that without artificial viscosity some eigenvalues move to the right of the imaginary axis and the computation fails to converge. Artificial viscosity is an essential feature of the numerical model for inviscid transonic flow.

After the flow has been marched forward by scheme (12) to a state where all time

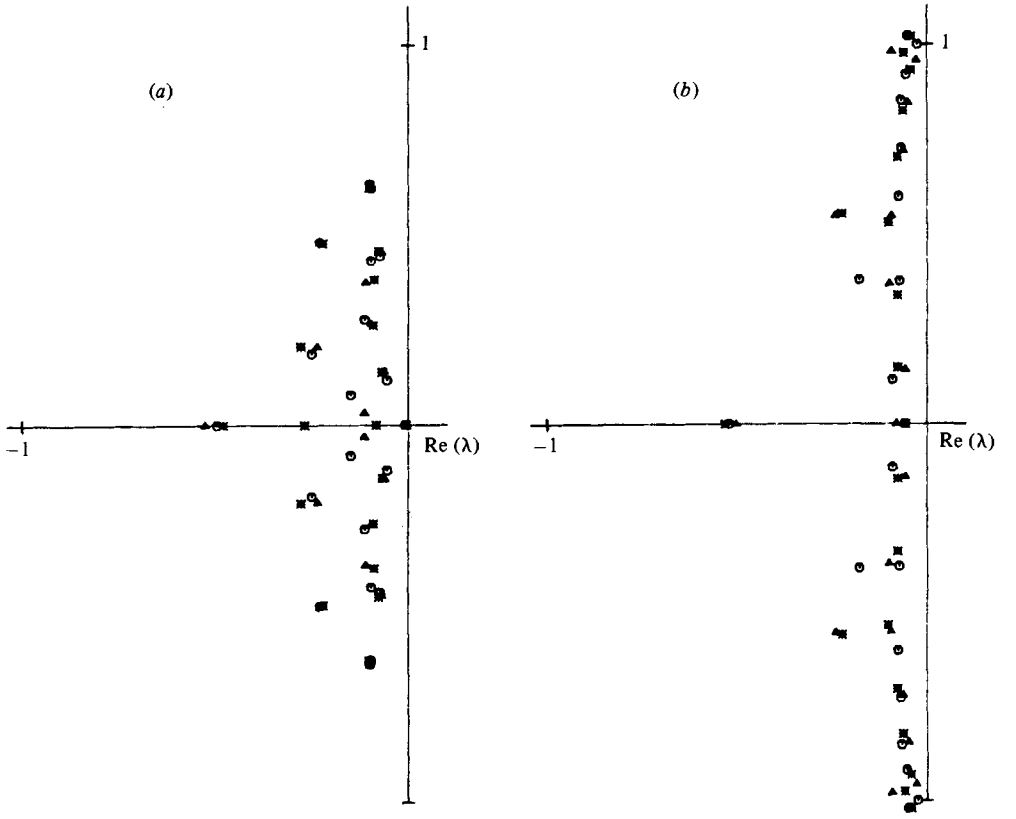

FIGURE 7. Effect of local time-step scaling on spectrum of linearized system. 32×7 grid around NACA 0012 airfoil, $M_\infty = 0.8$, $\alpha = 0°$, nonlinear dissipation added. System linearized after 15 time steps with free-stream initial conditions. (*a*) Unscaled system. (*b*) Scaled system.

perturbations $\boldsymbol{q}$ cease, the question that remains is: does this state $\boldsymbol{q} = \boldsymbol{q}^{n+1} = \boldsymbol{q}^n$ given by the integration scheme (12) satisfy the (linearized) steady operator $\boldsymbol{A}\boldsymbol{q} = 0$ identically? If all time variations are absent, the relation

$$\boldsymbol{q} = \{\boldsymbol{I} + \Delta t\boldsymbol{A} + \theta(\Delta t\boldsymbol{A})^2 + \theta^2(\Delta t\boldsymbol{A})^3\}\,\boldsymbol{q}$$

follows from (13), and hence $\Delta t[\boldsymbol{I} + \theta\,\Delta t\boldsymbol{A} + \theta^2(\Delta t\boldsymbol{A})^2]\,\boldsymbol{A}\boldsymbol{q} = 0$ holds. The answer then is yes, because the characteristic polynomial $1 + \theta z + \theta^2 z^2$ of the bracketed term cannot be zero for any $z = \Delta t\lambda$ within the scheme's stability region, since substituting its roots $z = (-1 \pm \mathrm{i}\sqrt{3})/2\theta$ in (14) yields $\sigma = 1$, which is just outside the stability bound. And we see that the solution to the discrete steady operator does not depend upon the time step used to reach it. This property of scheme (12) relies heavily on the fact that the same discrete operator is used in each of its stages. In schemes where this is not the case, for example the MacCormack scheme, the commutativity of the skewed forward and backward differences has to be considered, and the question becomes much more difficult to answer.

5. Computed example: M6 wing ($M_\infty = 0.84$, $\alpha = 3.06$)

We choose to present results computed for this wing of intermediate aspect ratio to demonstrate the overall applicability of the method, not only because it has become a standard test example of AGARD for computer-program evaluation, but primarily

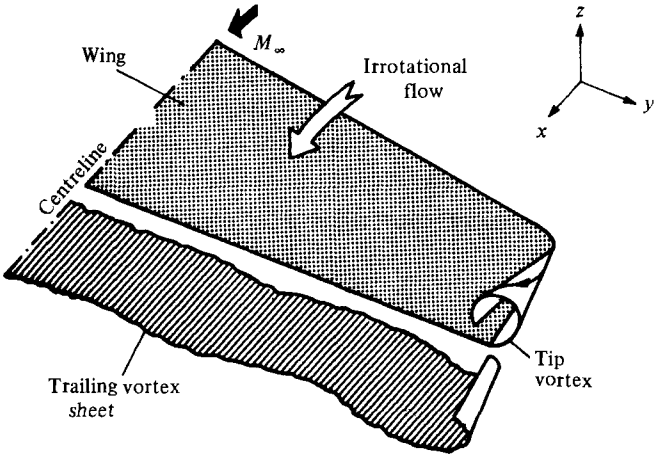

FIGURE 8. General features of the flow around a large-aspect-ratio wing. Inboard it is irrotational and highly two-dimensional. A free shear layer separates from the tip, interacts with the vortex sheet behind the trailing edge, and rolls up into a wake vortex.

because experiments have revealed the presence of a significant vortex above the wing tip. The rotational phenomenon we expect to see in our inviscid model is a captured non-uniform vortex sheet shed from the trailing edge. In order to satisfy the solenoidal requirement on vorticity, the corresponding vortex bound in the wing must exit from the wing tip and turn downstream. The vortex sheet then interacts with this tip vortex by rolling up and feeding its strength. The situation is pictured schematically in figure 8. Inboard over most of the span the flow is expected to be irrotational and nearly two-dimensional. This region of the flow has been simulated many times before and very accurately by potential-flow methods. Our interest here is to focus on the tip region in order to determine whether or not the Euler-equation model is able to capture the inviscid rotational flow we expect there with at least some degree of realism. We do not, however, compare our solution computed for this case with results from other Euler-equation methods, primarily because a critical comparison would turn into a full-length paper of its own. Readers interested in a thorough and up-to-date comparison of results from the currently best-known methods, ours included, are referred to the forthcoming report of AGARD Working Group 07(1984).

Containing a total of $96 \times 20 \times 20$ cells, our grid is a 0–0 type that wraps smoothly around the tip with a cell density sufficient to resolve the details of the flow there (refer to figure 1 for plots of the mesh). The solution we present is computed with the time-integration parameter $\theta = \frac{1}{2}$ and no Kutta condition or special procedure applied at either the sharp trailing edge or the smooth tip. In order to grasp the nature of the results, we must probe the computed solution by means of a series of contour maps, not only on the wing surface itself but out in the field as well. An overall impression of the pressure field around the wing from root to tip is illustrated in figure 9 by isobar maps in 4 selected chordwise grid surfaces (non-planar) that depict the familiar expansion/compression phenomenon near the leading and trailing edge as well as the formation of the lamda shock. The dotted line shows the sonic surface. These features are brought out even more distinctly by the isobar and Mach number contours on the upper wing surface (figure 10), together with the usual sectional chordwise plots of C_p in figure 11. At the tip figure 10 further indicates by the

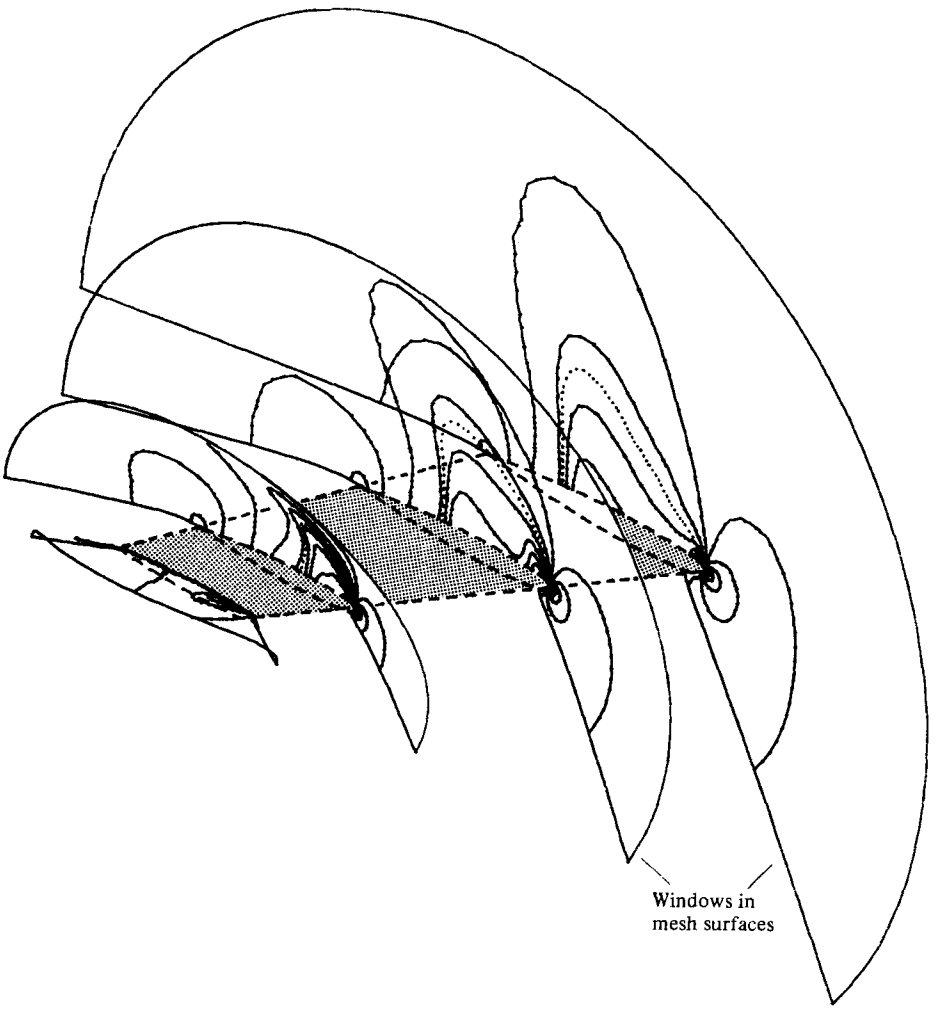

FIGURE 9. Isobar maps in 4 nonplanar chordwise grid surfaces depicting the pressure field around the M6 wing. The dotted line is the sonic surface. Contour interval 0.05. $M_\infty = 0.84$, $\alpha = 3.06$, $C_L = 0.286$, $C_D = 0.0116$.

coalescence of contour lines the stem of the lamda shock just aft of the leading edge. But what is most striking is the running together of contour lines at the trailing edge of the tip. Although novel for this type of wing, sectional spanwise plots of C_p are the best way to view this detail. Figure 12 contains three such sections (fore, mid and aft chord) together with the experimental values plotted in this uncommon manner. Inboard, reflecting the two-dimensional character of the flow, the pressure varies only slowly with span until the tip is approached. There locally in this intriguing region we see, and it is confirmed by the measurements, a small amount of negative lift being produced in section $x/c = 0.28$, which then changes to a positive contribution to lift in the next two sections $x/c = 0.68$ and 0.87. Similar to the suction peaks under the vortex generated by a delta wing, the computed pressure distribution at the tip leads one to suspect rotational flow. A probe of the vorticity field by 4 contour maps (figure 13) of the vorticity magnitude $|\Omega| = |\mathrm{curl}\, V|$ in the field around the wing tip confirms the existence of rotational flow local to that region. The trace

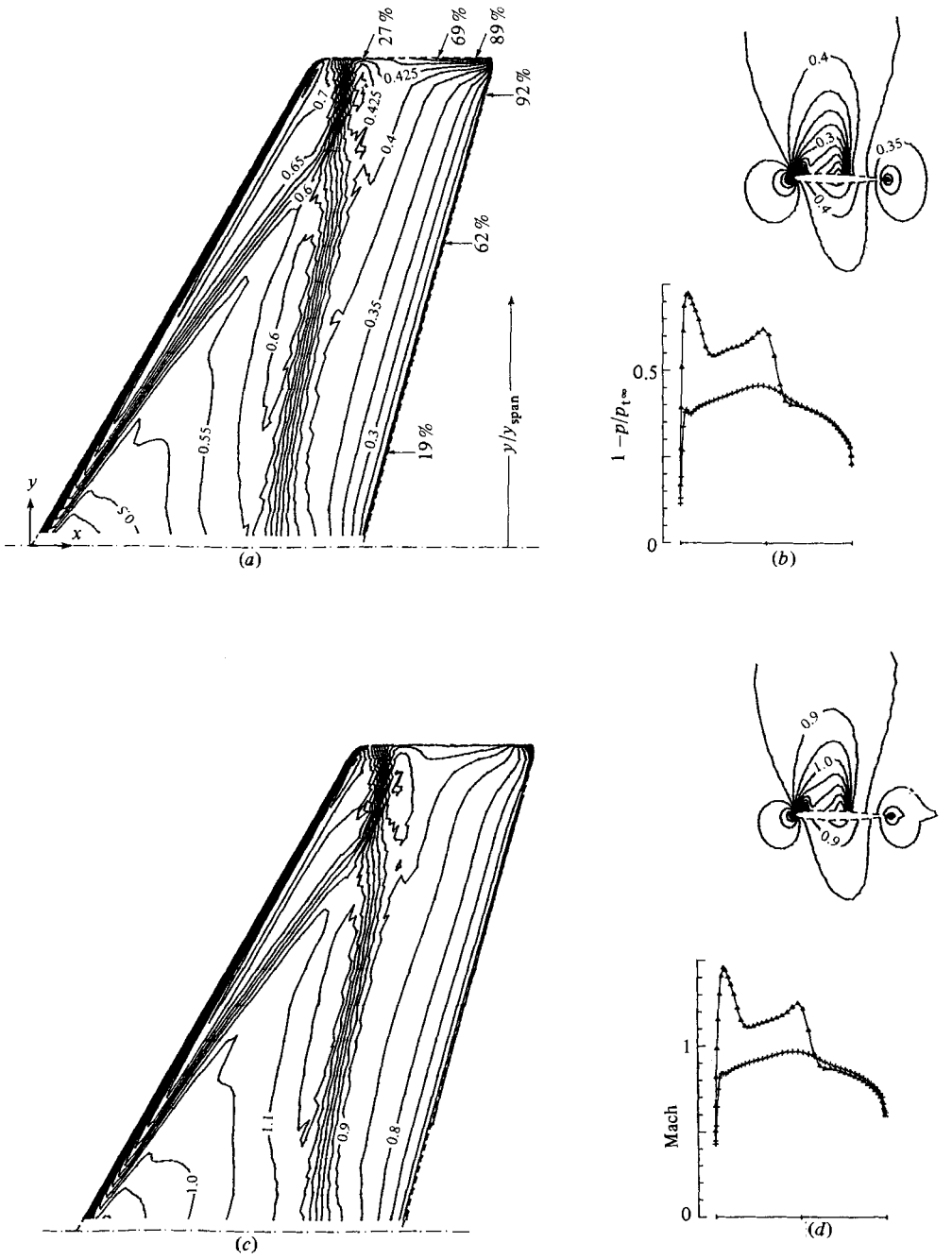

FIGURE 10. Contour maps of the solution computed around the M6 wing. (*a*) Lines of constant pressure $1-p/p_{t\infty}$ normalized by the freestream total pressure $p_{t\infty}$ on the upper surface of the M6 wing. Percent stations indicate the location of the 3 chordwise and spanwise sectional plots displayed in figures 11 and 12. $M_\infty = 0.84$, $\alpha = 3.06$. Contour interval 0.025. (*b*) Sectional graph and isobar map of $1-p/p_{t\infty}$ in the field around the wing in the 47 % span station. (*c*) Lines of constant Mach number on the upper surface. (*d*) Sectional graph and isobar map of Mach number in the field around the wing in the 47 % span station.

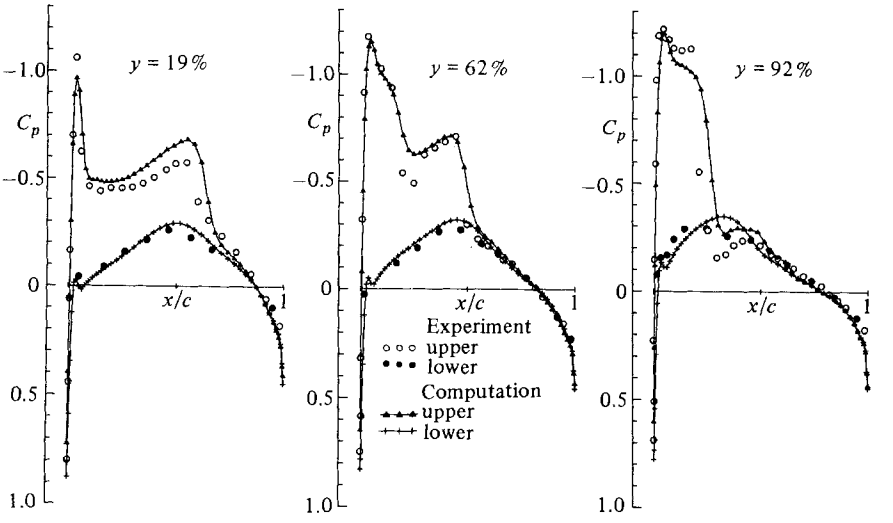

FIGURE 11. Comparison of measured and computed C_p in three chordwise sections of the M6 wing (see figure 10). $M_\infty = 0.84$, $\alpha = 3.06$.

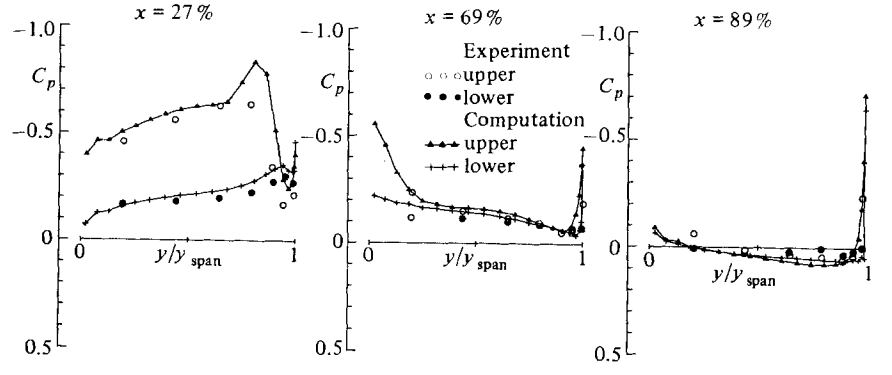

FIGURE 12. Comparison of measured and computed C_p in three spanwise sections (constant percent local chord) of the M6 wing (see figure 10). $M_\infty = 0.84$, $\alpha = 3.06$.

(figure 14) of these contours on the upper surface of the wing shows more clearly that the greatest amount and largest magnitude of vorticity is found at the trailing-tip corner, although there is a small amount along most of the leading as well as the trailing edge, the former produced, we suspect, by the leading-edge shock where the Mach number is greatest (over 1.4), and the latter probably diffused upstream from the trailing vortex sheet. The velocity field around the tip, given by vectors of the computed v and w velocity components (in figure 14), substantiates even more concretely the flow behaviour. In the upstream part of the tip region we see beyond doubt that the flow is attached, whereas further downstream it separates from the surface, creating a rotational flow above the upper surface of the wing. This inviscid separation phenomenon at the tip is, we believe, the emergence into the flow of the bound vortex, which in turn is the kinematic consequence of the vortex sheet shed along the entire trailing edge. Also shown in figure 14 is a velocity-vector plot of the transverse flow in a wake section that clearly shows the position of the tip vortex and the (smeared) vortex sheet emanating from the trailing edge. For a reference length, the trailing edge is projected on to this section and indicated by the solid line.

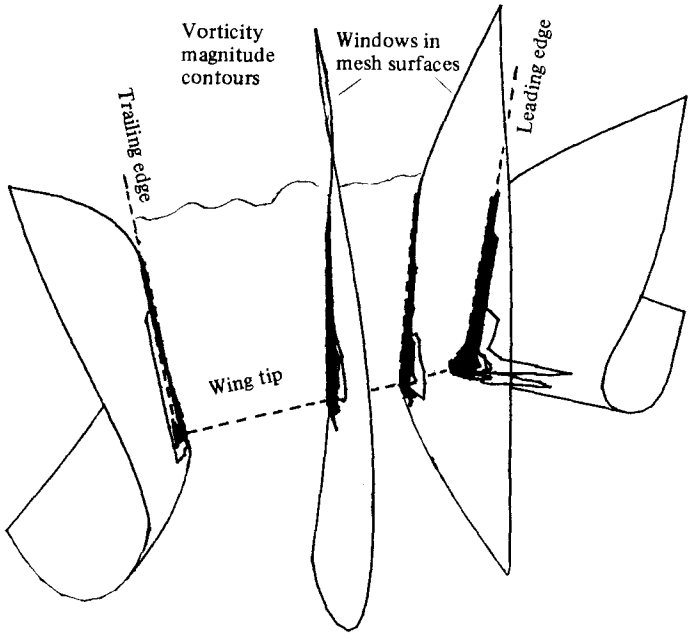

FIGURE 13. Lines of constant vorticity magnitude $|\Omega|$ in 4 grid surfaces around the wing tip show that vorticity at these surfaces does not extend very far out into the field. The maximum value, normalized to one, occurs at the upper-surface tip trailing edge. Contour interval = 0.001.

We further point out that the centre of the vortex sheet seems to be positioned slightly above the wing plane and that the centre of the tip vortex is displaced inboard. These are typical effects, qualitatively correct, and often observed in experiments as well as in the computed results of vortex-lattice methods with wake-vortex relaxation.

The dynamics of the computed wake are worth exploring further. The behaviour of the trailing vortex is shown in figure 15 by 4 contour maps of $|\Omega|$ in surfaces that cut spanwise through the wake, one just 2% of the root chord behind the trailing edge and the others further downstream. In the first we see a small but clearly developed tip vortex and a rather strong vortex sheet spanning the trailing edge (a portion of the map is cut away to reveal the trailing edge for a position reference). As it moves downstream the vortex grows in size, being fed by the decrease in vorticity across the trailing-edge sheet. Also, as already suggested by the velocity-vector plot, the vortex sheet rises above the wing plane and the tip vortex drifts inboard. At the 150% station behind the trailing edge it is fully developed. Such wake phenomena are a good test of the numerical method. We remark that our artificial-viscosity model does not unduly smear out the vortex sheet at the trailing edge. Although probably diffused on the rapidly expanding mesh, the vorticity contour is not at all distorted at the 450% station, which is the station nearest to the downstream boundary, and an indication that our non-reflecting far-field boundary conditions are able to handle even this case of a vortex leaving the field.

The free-stream flow is used as the initial conditions for the computation, which begins impulsively first on a coarse 0–0 grid of $48 \times 10 \times 10$ cells and then continues to the second and final (fine) grid of $96 \times 20 \times 20$ cells. After 800 iterations the coarse field is interpolated to the fine grid and advanced 200 iterations. The convergence history and the evolution of lift and drag for the entire computation is given in figure 16 by plots of the logarithm of the residual (defined as $\rho^{n+1}-\rho^n$), the number of

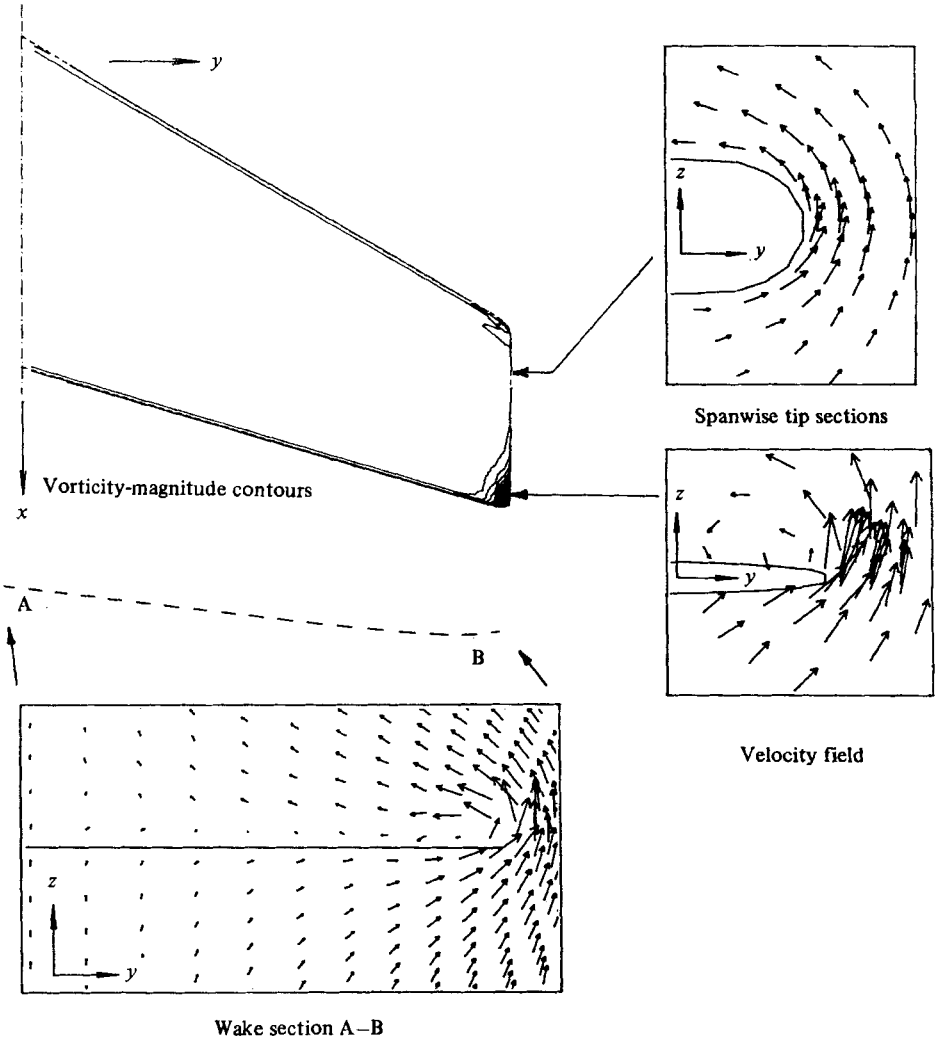

FIGURE 14. Lines of constant vorticity magnitude $|\Omega|$ on the upper wing surface. The maximum value, normalized to one, occurs at the upper surface tip trailing edge. Also drawn are vectors of the computed $v-w$ velocity components in two spanwise tip sections and one wake section to confirm the development of rotational flow.

supersonic points in the field, and the aerodynamic coefficients C_L and C_D all versus the number of time cycles and CPU time taken on the CYBER 205. The coarse-grid computation begins with the free-stream field, and after 800 cycles and 98 CPU seconds the residual is reduced by over 3 orders of magnitude, and the lift and drag is within about 10 % of their final values in the fine grid ($C_L = 0.286$ and $C_D = 0.0116$). The rate of convergence in the fine mesh is slower, the residual (normalized to one at the start) is reduced by less than two orders of magnitude to an absolute level of less than 10^{-4}, and the computing time is longer. But these are acceptable because the main function of the fine-mesh computation is just to improve upon the local accuracy of the coarse-mesh solution, which is reasonably correct in the large. The machine carried out this computation at an average rate of 57 million floating-point operations per second (m flops) sustained over the entire computation.

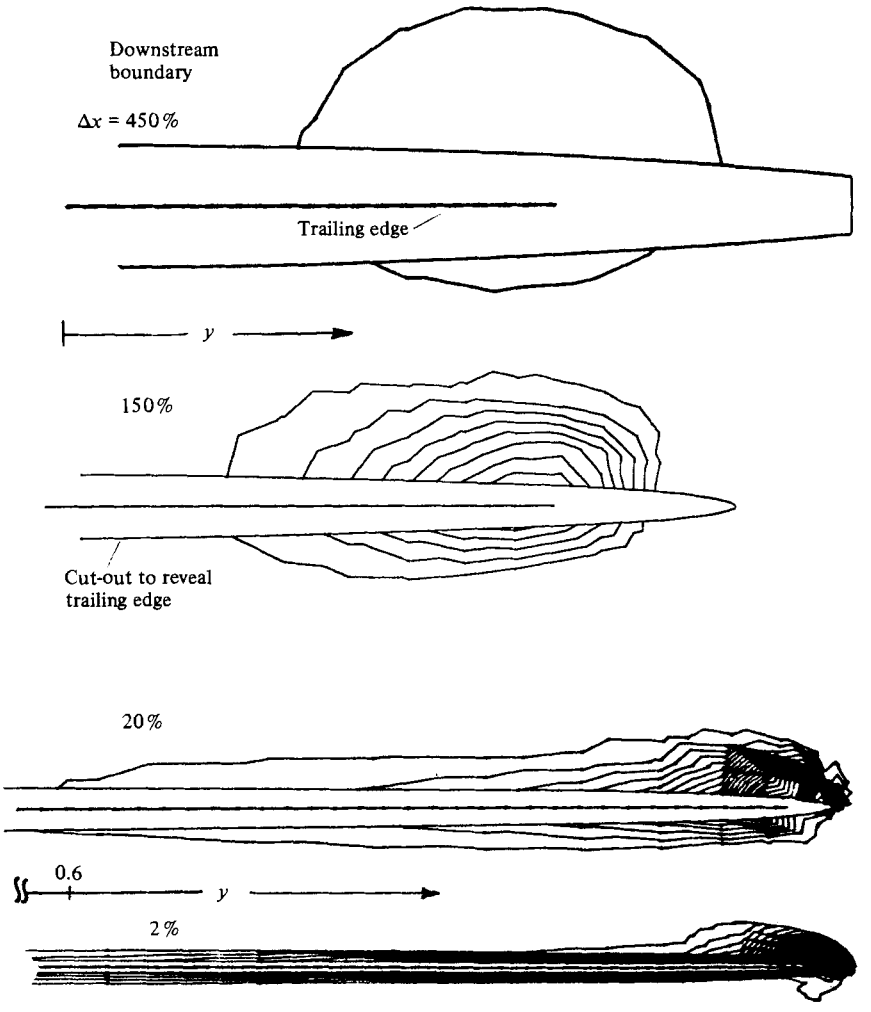

FIGURE 15. The wake region behind the M6 wing revealed by constant vorticity-magnitude contours plotted in 4 spanwise sections at distances in percent root chord behind the trailing edge. $M_\infty = 0.84$, $\alpha = 3.06$.

6. Concluding remarks

A large number of details enter in the design of a numerical procedure to simulate transonic flow. The most important ones have been touched upon here – mesh topology, coordinate singularities, flux conservation, boundary conditions, artificial viscosity, time-integration schemes, and ways to deal with an ill-conditioned system of equations. But perhaps the most difficult of all is the question of modelling. We want, after all, to simulate real flows. The Euler equations *per se* are only an approximation to reality, but by modelling techniques we can hope to make them a better one. In order to do so, the artificial-viscosity models will have to be improved, and we shall have to reach a quantitative understanding of the way the vortex sheet develops at the trailing edge.

What we have not addressed at all here is how the details of our numerical procedure are translated into an executable program intelligible to a computing machine. This consideration is taken up in Rizzi (1983), which spells out in detail how

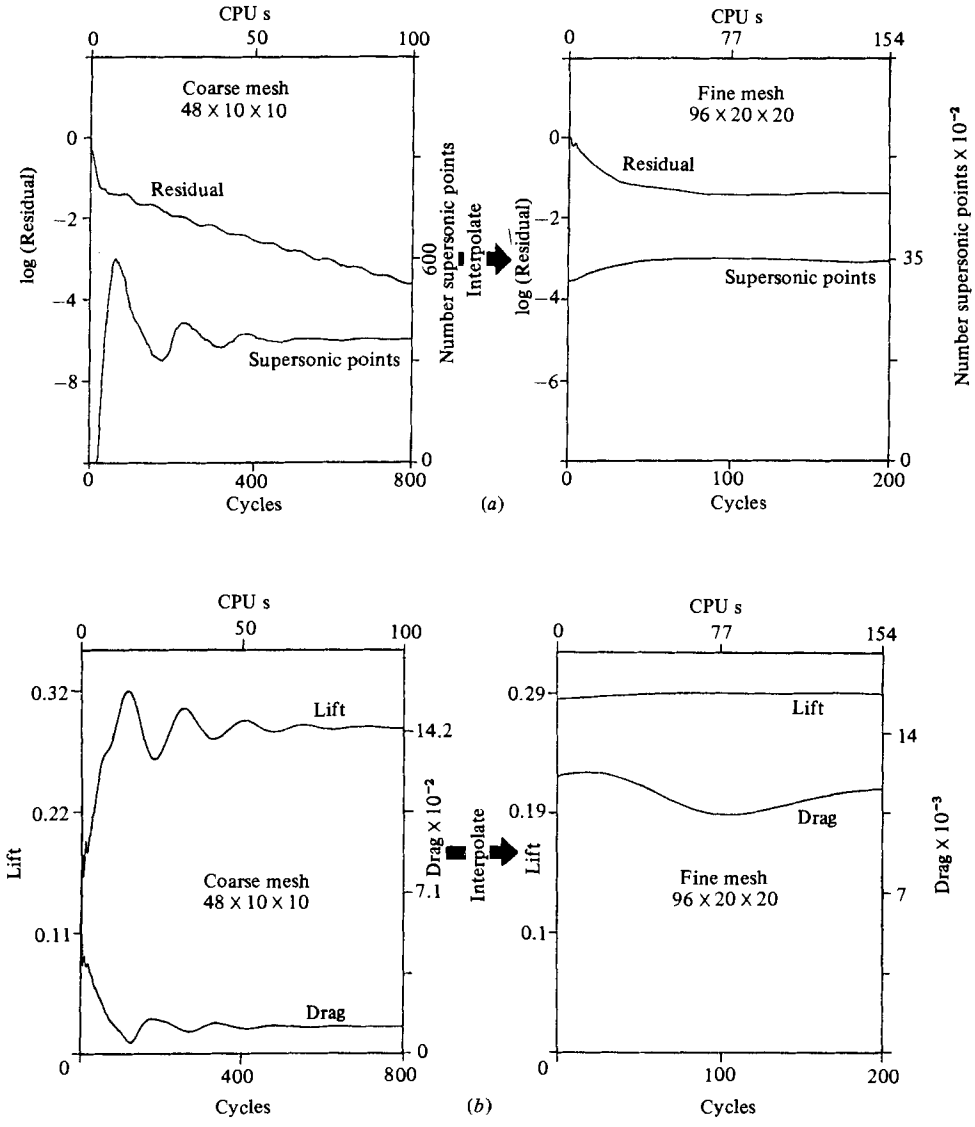

FIGURE 16. The convergence of the computation to steady state. The calculation begins with the free-stream field on the coarse mesh which is then interpolated to the fine mesh for final results. M6 wing. $M_\infty = 0.84, \alpha = 3.06$. (*a*) Transient-residual decay (normalized to one at start) and build-up of supersonic field versus iteration cycles and CPU seconds on the CYBER 205. (*b*) Evolution of lift and drag versus iteration cycles and CPU seconds on CYBER 205. Final values $C_L = 0.286$ and $C_D = 0.0116$.

to code our numerical procedure for efficient execution on the CYBER 205 vector computer. The software itself, which we call WINGA2,† is segmented along its functional lines of mesh generation as the preprocessor, the flow solver described here as the central processor, and the graphical surveying of results the post-processor (Rizzi & Eriksson 1984).

The results we have presented for flow around the M6 wing, especially those at the

† Readers interested in obtaining a copy of WINGA2 are invited to write to the authors for details.

tip, demonstrate the type of rotational flow field that can be simulated with at least qualitative realism by the Euler equations, and they justify the computational effort to obtain them. We do not believe this to be a fortuitous case. But we do seriously doubt that the same quality of results we obtained at the wing tip can be achieved without the resolving power of a 0–0 or C–0 mesh encircling a smooth rounded tip.

REFERENCES

AGARD WORKING GROUP 07 1984 Test cases for steady inviscid transonic and supersonic flows. *AGARD Publ.* (in preparation).

ENGQVIST, B. & MAJDA, A. 1977 Absorbing boundary conditions for the numerical simulation of waves. *Math. Comp.* **31**, 629–651.

ERIKSSON, L.-E. 1982 Generation of boundary-conforming grids around wing–body configurations using transfinite interpolation. *AIAA J.* **20**, 1313–1320.

ERIKSSON, L.-E. 1984 A study of mesh singularities and their effects on numerical errors. *FFA TN* 1984–10, *Stockholm.*

ERIKSSON, L.-E. & RIZZI, A. 1984 Computation of vortex flow around a canard–delta combination. *J. Aircraft* (in press).

ERIKSSON, L.-E. & RIZZI, A. 1984 Computer-aided analysis of the convergence to steady state of a discrete approximation to the Euler equations. *J. Phys.* (in press).

GOTTLIEB, D. & GUSTAFSSON, B. 1976 On the Navier–Stokes equations with constant total temperature. *Stud. Appl. Maths* **55**, 167–185.

HIRSCHEL, E. H. & FORNASIER, L. 1984 Flowfield and vorticity distribution near wing trailing edges. *AIAA Paper* 84–0421, *New York.*

JAMESON, A. & BAKER, T. J. 1984 Multigrid solution of the Euler equations for aircraft configurations. *AIAA Paper* 84-0093, *New York.*

KOECK, C. & NERON, M. 1984 Computations of three-dimensional transonic inviscid flows on a wing by pseudo-unsteady resolution of the Euler equations. In *5th GAMM Conf. Proc. Numer. Meth. Fluid Mech.* (ed. M. Pandolfi & R. Piva). Vieweg.

LOMAX, H. 1982 Some prospects for the future of computational fluid dynamics. *AIAA J.* **20**, 1033–1043.

LOMAX, H., PULLIAM, T. H. & JESPERSEN, D. C. 1981 Eigensystem analysis techniques for finite-difference equations. *AIAA Paper* 81-1027.

MACCORMACK, R. W. & PAULLAY, A. J. 1974 The influence of the computational mesh on accuracy for initial value problems with discontinuous or nonunique solutions. *Comp. Fluids* **2**, 339–361.

RIZZI, A. W. 1978 Numerical implementation of solid-body boundary conditions for the Euler equations. *Z. angew. Math. Mech.* **58**, T301–T304.

RIZZI, A. W. 1981 Computation of rotational transonic flow. In *Numerical Methods for the Computation of Inviscid Transonic Flow with Shocks, a GAMM Workshop* (ed. A. W. Rizzi & H. Viviand). Notes on Numerical Fluid Mechanics. Vieweg.

RIZZI, A. 1982 Damped Euler-equation method to compute transonic flow around wing–body combinations. *AIAA J.* **20**, 1321–1328.

RIZZI, A. 1983 Vector coding the finite-volume procedure for the CYBER 205. In *Lecture Series Notes 1983-04, von Kármán Inst., Brussels.*

RIZZI, A. & ERIKSSON, L.-E. 1984 The FFA aerodynamic flow code WINGA2 for CYBER 205: numerical software to solve the Euler equations. *Adv. Engng Software* (in press).