INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2005; **49**:1261–1286
Published online 3 August 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1045

Finite element and implicit Runge–Kutta implementation of an acoustics–convection upstream resolution algorithm for the time-dependent two-dimensional Euler equations

Joe Iannelli*,†,‡

Centre for Aeronautics, School of Engineering and Mathematical Sciences, City University, Northampton Square, London EC1V 0HB, U.K.

SUMMARY

The second of a two-paper series, this paper details a solver for the characteristics-bias system from the acoustics–convection upstream resolution algorithm for the Euler and Navier–Stokes equations. An integral formulation leads to several surface integrals that allow effective enforcement of boundary conditions. Also presented is a new multi-dimensional procedure to enforce a pressure boundary condition at a subsonic outlet, a procedure that remains accurate and stable. A classical finite element Galerkin discretization of the integral formulation on any prescribed grid directly yields an optimal discretely conservative upstream approximation for the Euler and Navier–Stokes equations, an approximation that remains multi-dimensional independently of the orientation of the reference axes and computational cells. The time-dependent discrete equations are then integrated in time via an implicit Runge–Kutta procedure that in this paper is proven to remain absolutely non-linearly stable for the spatially-discrete Euler and Navier–Stokes equations and shown to converge rapidly to steady states, with maximum Courant number exceeding 100 for the linearized version. Even on relatively coarse grids, the acoustics–convection upstream resolution algorithm generates essentially non-oscillatory solutions for subsonic, transonic and supersonic flows, encompassing oblique- and interacting-shock fields that converge within 40 time steps and reflect reference exact solutions. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: Euler and Navier–Stokes equations; CFD; finite elements; characteristics; implicit Runge–Kutta; multi-dimensional upwind

1. INTRODUCTION

This paper provides the second part of a two-part investigation into the development of continuum, i.e. non-discrete, multi-dimensional and infinite-directional characteristics-bias

*Correspondence to: J. Iannelli, Centre for Aeronautics, School of Engineering and Mathematical Sciences, City University, Northampton Square, London EC1V 0HB, U.K.
†E-mail: g.iannelli@city.ac.uk
‡Director.

Received 9 January 2005
Revised 11 April 2005
Accepted 9 June 2005

approximations of the Euler and Navier–Stokes equations and subsequent computational implementation. The first part [1], also presented in this journal, synthesizes and analyses via characteristics the acoustic–convection upstream resolution algorithm. Detailing an integral formulation, a finite element discretization, and an implicit Runge–Kutta time integration, this paper presents computational results for inviscid two-dimensional subsonic, transonic, and supersonic flows with shock reflections and interactions. In particular, the algorithm allows reflected shocks to cross outflow boundaries unperturbed, without any spurious distortion.

As its guiding principle, the acoustics convection upstream resolution algorithm develops the upstream formulation in the continuum, directly at the partial differential equation level, before the approximation of the spatial partial derivatives, so that a traditional centred approximation of the associate integral formulation automatically generates an upstream discrete system for the Euler and Navier–Stokes equations. Associated with the Euler and Navier–Stokes equations, the non-discrete formulation results in a 'companion' characteristics-bias system that relies on a decomposition of the multi-dimensional Euler Jacobian into acoustics and convection components. For any Mach number, the characteristics-bias system induces consistent upwinding along every direction radiating from any flow-field point [1]. The formulation induces an anisotropic variable-strength upstream bias that directly correlates with the multi-dimensional spatial distribution of characteristic velocities. The magnitudes of the associated streamwise and crossflow dissipations remain different from and independent of each other, with crossflow dissipation that decreases for increasing Mach number. In this manner the developed formulation will not generate for increasing Mach number as much crosswind dissipation as induced by an isotropic or direction-split formulation.

This paper develops an integral formulation of the characteristics-bias system. This formulation directly leads to several surface integrals that allow effective enforcement of wall-tangency, pressure, surface-traction and heat-flux boundary conditions for arbitrary geometric shapes of the computational domain. In particular, these developments establish the stability of the pressure boundary-condition enforcement method, which is then confirmed by the computed solutions.

A Galerkin finite element is used to discretize in space the integral statement. Affording considerable geometric flexibility [2–4] and employing optimal metric data [5], this finite element implementation retains the ideal surface-integral venues in order accurately and efficiently to enforce several boundary conditions at solid walls and subsonic and supersonic outlets. On arbitrary grids, this finite element discretization of the integral formulation automatically and directly generates a consistent, discretely conservative and genuinely multi-dimensional upstream-bias approximation of the Euler and Navier–Stokes equations. The associated discrete upstream-bias remains independent of the direction of the coordinate axes as well as orientation of each computational-cell side, which obviates the need for rotated stencils. With an operation count comparable to that of a simple flux vector splitting algorithm, this approximation requires data only from the computational cells shared by each grid node and also reduces to a consistent upstream approximation of the acoustics equations, for vanishing Mach number, which addresses the challenging problem of calculating low-Mach-number flows. The developments in this study have employed basic Lagrange four-noded cells without any MUSCL-type local extrapolation of dependent variables, in order to implement the algorithm efficiently and determine the ultimate accuracy of bi-linear approximations of fluxes within four-noded cells.

The formulation, furthermore, supplies a stable and intrinsically infinite directional upstream-bias that induces minimal diffusion for crisp oblique-shock capturing. The upstream directions are continuously updated and high-rate convergence to machine zero achieved without additional shock-capturing terms, data filtering or loss of essential monotonicity. The magnitude of the induced upstream bias depends on local solution smoothness. Only at solution discontinuities, e.g. a shock wave, is the induced upstreaming commensurate with a fully upwind formulation. In regions of solution continuity, the upwind-bias reduces to a minimum, which corresponds to minimal induced dissipation [6].

To integrate the discrete finite element equations in time, this paper details an implicit Runge–Kutta algorithm with analytically determined Jacobians. Generalizing previous contributions [5, 7–9], this algorithm is proven in this paper absolutely non-linearly energy stable for the spatially discrete Euler and Navier–Stokes system and shown to determine steady-state flows rapidly, within 40 time steps, with maximum Courant number exceeding 100 for the linearized version. Based on this result, the stability of a CFD algorithm employing this implicit Runge–Kutta integration is thus chiefly determined by the upstream-bias formulation, spatial discretization, and boundary conditions, which should collectively generate a stable system of ordinary differential equations in continuum time.

Because of size constraints, this paper presents results for two-dimensional inviscid flows in order to demonstrate the computational performance of the algorithm in a setting that is familiar to a wide audience. A third paper is planned to present the three-dimensional formulation, detail a memory-efficient implementation of GMRES for solving the eventual numerical linear algebra problem, and discuss computational results for two- and three-dimensional viscous flows containing shock-wave boundary layer interactions and hypersonic bow shocks.

This paper is organized in seven sections. After the introductory remarks in Section 1, Section 2 summarizes the multi-dimensional non-discrete upstream-bias approximation for the Euler and Navier–Stokes equations. The integral formulation is described in Section 3, the finite element Galerkin spatial discretization is delineated in Section 4, and the Runge–Kutta time integration algorithm is discussed in Section 5. Section 6 details the computational results, with concluding remarks presented in Section 7.

2. NON-DISCRETE UPSTREAM-BIAS APPROXIMATION

As Reference [1] details, the non-discrete flux Jacobian decomposition (FJD) upstream-bias approximation is developed for the Euler and Navier–Stokes equations. With implied summation on repeated subscript indices, these equations can be abridged as the non-linear parabolic system

$$\frac{\partial q}{\partial t} + \frac{\partial f_j(q)}{\partial x_j} - \frac{\partial f_j^v}{\partial x_j} = 0 \tag{1}$$

which reduces to the Euler hyperbolic system when the fluid-viscosity flux f_j^v identically vanishes. For three-dimensional formulations, $1 \leqslant j \leqslant 3$, and with $\mathscr{R}$ denoting the real-number field, the independent variable $(\mathbf{x}, t)$, $\mathbf{x} \equiv (x_1, x_2, x_3)$, in (1) varies in the domain $D \equiv \Omega \times [t_o, t_f]$,

$[t_o, t_f] \subset \mathscr{R}^+$, $\Omega \subset \mathscr{R}^3$. Equivalent to this governing system is the integral statement

$$\int_{\hat{\Omega}} \hat{w} \left(\frac{\partial q}{\partial t} + \frac{\partial f_j(q)}{\partial x_j} - \frac{\partial f_j^v}{\partial x_j} \right) \mathrm{d}\Omega = 0 \tag{2}$$

when it holds for arbitrary subdomains $\hat{\Omega} \subseteq \Omega$ and test functions $\hat{w} \in \mathscr{H}^1(\hat{\Omega}) \subseteq \mathscr{H}^1(\Omega)$ with compact support in $\hat{\Omega}$, [10–13].

The non-discrete formulation induces a multi-dimensional upstream-bias directly in the continuum, at the partial-differential equation level, before the eventual discretization on a prescribed grid. This continuum upstream-bias formulation derives from a characteristics-bias integral statement associated with (1). With reference to (2), the characteristic-bias integral is then defined as

$$\int_{\hat{\Omega}} \hat{w} \left(\frac{\partial q}{\partial t} + \frac{\partial f_j^{\mathrm{C}}}{\partial x_j} - \frac{\partial f_j^v}{\partial x_j} \right) \mathrm{d}\Omega = 0 \tag{3}$$

where f_j^{C} corresponds to a characteristics flux that automatically induces within (3) a multi-dimensional and infinite directional upstream-bias approximation for the hyperbolic flux divergence $\partial f_j / \partial x_j$.

The acoustics–convection characteristics flux divergence for the Euler flux is expressed as [1]

$$\frac{\partial f_j^{\mathrm{C}}}{\partial x_j} = \frac{\partial f_j}{\partial x_j} - \frac{\partial}{\partial x_i} \left[\varepsilon \psi \left(c(\alpha a_i a_j + \alpha^N a_i^N a_j^N) \frac{\partial q}{\partial x_j} + a_i \frac{\partial f_j^q}{\partial x_j} + a_i \delta \frac{\partial f_j^p}{\partial x_j} \right) \right] \tag{4}$$

where f_j^q and f_j^p, respectively, denote the convection and pressure flux components. For 2-D flows, these components are defined as

$$f_j(q) = f_j^q(q) + f_j^p(q)$$

$$f_j^q(q) \equiv \begin{Bmatrix} m_j \\ \frac{m_j}{\rho} m_1 \\ \frac{m_j}{\rho} m_2 \\ \frac{m_j}{\rho} (E + p) \end{Bmatrix} = \frac{m_j}{\rho} q^H \equiv \frac{m_j}{\rho} \cdot \begin{Bmatrix} \rho \\ m_1 \\ m_2 \\ E + p \end{Bmatrix}, \quad f_j^p(q) \equiv \begin{Bmatrix} 0 \\ p\delta_1^j \\ p\delta_2^j \\ 0 \end{Bmatrix} \tag{5}$$

with straightforward generalization to 3-D flows. In the array q, the variables ρ, m_1, m_2, E, respectively, denote static density, volume-specific linear momentum components and total energy, with c, p, and T representing speed of sound, static pressure, and static temperature from the expressions

$$c^2 = p_\rho + p_E \left(\frac{E + p}{\rho} - \frac{1}{\rho^2}(m_1^2 + m_2^2) \right), \quad p = (\gamma - 1) \left(E - \frac{1}{2\rho}(m_1^2 + m_2^2) \right), \quad T = \frac{p}{\rho R} \tag{6}$$

where R denotes the gas constant. The Eulerian flow velocity $\mathbf{u}$, with Cartesian components u_j, $1 \leqslant j \leqslant 2$, is then defined as $\mathbf{u} \equiv \mathbf{m}/\rho$, with Mach number $M \equiv \|\mathbf{u}\|/\rho$. The non-negative variables ε, ψ, α, α^N and δ, each with magnitude not exceeding one, respectively, indicate a reference length, upstream-bias controller, streamwise and crossflow acoustic upstream functions and pressure-gradient upstream function. The direction cosines a_i, a_i^N, $1 \leqslant i \leqslant 2$, correspond to the components of unit vectors $\mathbf{a}$, $\mathbf{a}^N$, respectively, in the direction of and perpendicular to the local velocity vector. In result (4), the expressions $(c\alpha a_i a_j(\partial q/\partial x_j) + a_i(\partial f_j^q/\partial x_j) + a_i\delta(\partial f_j^p/\partial x_j))$ and $(c\alpha^N a_i^N a_j^N(\partial q/\partial x_j))$, determine the upstream biases within, respectively, the streamline and crossflow wave propagation regions.

The first paper of this series [1] shows that the upstream-bias functions $\alpha = \alpha(M)$, $\delta = \delta(M)$, and $\alpha^N = \alpha^N(M)$ are then directly and exactly determined as

$$\alpha(M) = \lambda_1(M) - M, \quad \delta(M) = \frac{(\lambda_1(M) - \lambda_4(M))(\lambda_1(M) - \lambda_4(M) + p_E M)}{1 + p_E M(\lambda_1(M) - \lambda_4(M))} \tag{7}$$

$$\alpha^N(M) \equiv \begin{cases} (\alpha_M^{N'} M_M - 2\alpha_M^N + 2)\left(\dfrac{M}{M_M}\right)^3 - (\alpha_M^{N'} M_M - 3\alpha_M^N + 3)\left(\dfrac{M}{M_M}\right)^2 + 1, & 0 \leqslant M < M_M \\ \dfrac{1}{2}\left(1 + \dfrac{\varepsilon_M}{M_M - \sqrt{M_M^2 - 1}}\right)(M - \sqrt{M^2 - 1}), & M_M \leqslant M \end{cases} \tag{8}$$

where superscript prime ‘′’ denotes differentiation with respect to M and subscripts ‘M’ in both $\alpha_M^{N'}$ and α_M^N indicate their respective magnitudes from the second expression in (8) at $M = M_M \equiv 1 + \varepsilon_M$, $\varepsilon_M = \frac{1}{5}$. The streamline upstream-bias eigenvalues are expressed as

$$\lambda_1(M) \equiv \begin{cases} 1 - M + \dfrac{\varepsilon_M}{2}(2M)^{1/\varepsilon_M}, & 0 \leqslant M < \dfrac{1}{2} \\ \dfrac{(M - \frac{1}{2})^2}{2\varepsilon_M} + \dfrac{1 + \varepsilon_M}{2}, & \dfrac{1}{2} \leqslant M < \dfrac{1}{2} + \varepsilon_M \\ M, & \dfrac{1}{2} + \varepsilon_M \leqslant M \end{cases} \tag{9}$$

$$\lambda_4(M) \equiv \begin{cases} 1 - M, & 0 \leqslant M \leqslant 1 - \varepsilon_M \\ \dfrac{(M-1)^2}{2\varepsilon_M} + \dfrac{\varepsilon_M}{2}, & 1 - \varepsilon_M < M < 1 + \varepsilon_M \\ M - 1, & 1 + \varepsilon_M \leqslant M \end{cases} \tag{10}$$

The acoustics–convection upstream-resolution algorithm is implemented via the following steps. After calculating at each grid node the speed of sound c, Mach number M, and unit vectors $\mathbf{a}$, $\mathbf{a}^N$, the implementation computes the streamline eigenvalues λ_1 as well as λ_4 and the upstream bias functions α, δ, and α^N. All of these terms then allow formulating the discrete analogue of the characteristics-bias Euler flux divergence (4).

3. INTEGRAL FORMULATION AND BOUNDARY CONDITIONS

With the divergence (4) of the characteristics flux f_j^{C} established, the integral formulation for system (3), [10–13], seeks a solution $q \in \mathscr{H}^1(\Omega)$, subject to prescribed boundary conditions on $\partial\Omega \equiv \bar{\Omega}\backslash\Omega$, such that for all test functions $w \in \mathscr{H}^1(\Omega)$

$$\int_\Omega \left[w\left(\frac{\partial q}{\partial t} + \frac{\partial f_j}{\partial x_j}\right) + \frac{\partial w}{\partial x_i}\varepsilon\psi\left(c(\alpha a_i a_j + \alpha^N a_i^N a_j^N)\frac{\partial q}{\partial x_j} + a_i\frac{\partial f_j^q}{\partial x_j} + a_i\delta\frac{\partial f_j^p}{\partial x_j}\right)\right] \mathrm{d}\Omega + \int_\Omega \frac{\partial w}{\partial x_j} f_j^v \,\mathrm{d}\Omega - \oint_{\partial\Omega} w f_j^v n_j \,\mathrm{d}\Gamma = 0 \tag{11}$$

The surface integral on $\partial\Omega$ corresponding to the characteristics-bias expression vanishes because of the boundary condition $\psi(\mathbf{x}_{\partial\Omega}) = 0$, imposed to eliminate unnecessary boundary upstream bias. In (11), the term n_j denotes the jth component of the outward pointing unit vector perpendicular to the boundary, as depicted in Figure 1. As well known, the continuity equation within (11) remains the same for both inviscid and viscous flows. For reference, this equation becomes

$$\int_\Omega \left[w\left(\frac{\partial \rho}{\partial t} + \frac{\partial m_j}{\partial x_j}\right) + \frac{\partial w}{\partial x_i}\varepsilon\psi\left(c(\alpha a_i a_j + \alpha^N a_i^N a_j^N)\frac{\partial \rho}{\partial x_j} + a_i\frac{\partial m_j}{\partial x_j}\right)\right] \mathrm{d}\Omega = 0 \tag{12}$$

At an inlet, Dirichlet boundary conditions on density ρ and total energy E are enforced. Where the inlet flow is subsonic, a Dirichlet boundary condition is enforced on the transverse linear momentum component m_2; for an inviscid flow, no boundary condition is required on the axial linear-momentum component m_1; for a viscous flow, an inlet boundary condition for the axial linear-momentum equation that can be prescribed is a vanishing surface traction $\tau_{1j}n_j$, with τ_{1j} the components of the deviatoric Navier–Stokes stress tensor, a condition effectively enforced by deleting the corresponding surface integral in the statement

$$\int_\Omega w\left(\frac{\partial m_1}{\partial t} + \frac{\partial}{\partial x_j}\left(\frac{m_1 m_j}{\rho}\right) + \frac{\partial p}{\partial x_1}\right)\mathrm{d}\Omega + \int_\Omega \frac{\partial w}{\partial x_j}\tau_{1j}\,\mathrm{d}\Omega - \oint_{\partial\Omega} w\tau_{1j}n_j\,\mathrm{d}\Gamma + \int_\Omega \frac{\partial w}{\partial x_i}\varepsilon\psi\left(c(\alpha a_i a_j + \alpha^N a_i^N a_j^N)\frac{\partial m_1}{\partial x_j} + a_i\frac{\partial}{\partial x_j}\left(\frac{m_1 m_j}{\rho}\right) + a_i\delta\frac{\partial p}{\partial x_1}\right)\mathrm{d}\Omega = 0 \tag{13}$$

Where the inlet flow is supersonic, Dirichlet boundary conditions are enforced on all the components m_1 and m_2 of linear momentum.

With reference to decomposition (5) and Figure 1(a), the divergence of the convection flux with components f_j^q within (11) is integrated by parts at a wall region, an operation that

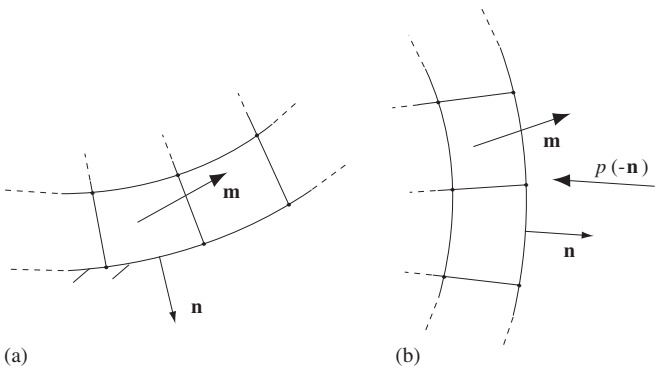

Figure 1. Boundary regions: (a) wall; and (b) outlet.

yields the wall boundary statement

$$\int_\Omega \left[w \left(\frac{\partial q}{\partial t} + \frac{\partial f_j^p}{\partial x_j} \right) + \frac{\partial w}{\partial x_i} \varepsilon\psi \left(c(\alpha a_i a_j + \alpha^N a_i^N a_j^N) \frac{\partial q}{\partial x_j} + a_i \frac{\partial f_j^q}{\partial x_j} + a_i \delta \frac{\partial f_j^p}{\partial x_j} \right) \right] \mathrm{d}\Omega$$

$$- \int_\Omega \frac{\partial w}{\partial x_j} f_j^q \, \mathrm{d}\Omega + \int_\Omega \frac{\partial w}{\partial x_j} f_j^v \, \mathrm{d}\Omega = - \oint_{\partial\Omega} w q^H \frac{\mathbf{m}}{\rho} \cdot \mathbf{n} \, \mathrm{d}\Gamma + \oint_{\partial\Omega} w f_j^v n_j \, \mathrm{d}\Gamma \tag{14}$$

This statement features a wall surface integral that depends on the mass flux $(\mathbf{m}/\rho)\cdot\mathbf{n}$. A wall mass-flux boundary condition is therefore directly enforced within this surface integral, with a wall-tangency boundary condition obtained by setting the entire integral to nought. For an inviscid flow, this is the only boundary condition that is required at a solid wall.

For a viscous flow, the two linear-momentum components in statement (14) are replaced by the no-slip boundary conditions on m_1 and m_2. The wall mass-flux surface integrals in the continuity and total-energy components in this statement are then eliminated; the wall heat-flux integral in the energy equation

$$\int_\Omega \left[w \frac{\partial E}{\partial t} - \frac{\partial w}{\partial x_j} \left(m_j \frac{E+p}{\rho} - u_i \tau_{ij} - k \frac{\partial T}{\partial x_j} \right) \right] \mathrm{d}\Omega + \oint_{\partial\Omega} w \left(m_j n_j \frac{E+p}{\rho} - u_i \tau_{ij} n_j - k \frac{\partial T}{\partial n} \right) \mathrm{d}\Gamma$$

$$+ \int_\Omega \frac{\partial w}{\partial x_i} \varepsilon\psi \left(c(\alpha a_i a_j + \alpha^N a_i^N a_j^N) \frac{\partial E}{\partial x_j} + a_i \frac{\partial}{\partial x_j} \left(m_j \frac{E+p}{\rho} \right) \right) \mathrm{d}\Omega = 0 \tag{15}$$

with k the coefficient of thermal conductivity, is then used to specify a heat-flux boundary condition, by inserting the prescribed heat-flux $\partial T/\partial n$ in this integral. For a cold- or hot-wall boundary condition, the entire energy equation (15) is then replaced at each hot- or cold-wall

arc by the wall surface integral

$$\oint_{\partial\Omega} w \left(\frac{p(q)}{\rho R} - T_w \right) \mathrm{d}\Gamma = 0 \tag{16}$$

where T_w denotes a prescribed wall temperature.

At a supersonic outlet, no boundary conditions are enforced for an inviscid flow; for a viscous flow, the minimal-perturbation boundary conditions that are prescribed involve an adiabatic outlet stream and vanishing deviatoric surface tractions $\tau_{ij} n_j$. The adiabatic-stream condition is enforced by deleting the corresponding surface integral in (15); the vanishing-traction condition is enforced by deleting the corresponding surface integral from the linear-momentum statement

$$\int_\Omega w \left(\frac{\partial m_i}{\partial t} + \frac{\partial}{\partial x_j} \left(\frac{m_i m_j}{\rho} \right) + \frac{\partial p}{\partial x_i} \right) \mathrm{d}\Omega + \int_\Omega \frac{\partial w}{\partial x_j} \tau_{ij} \, \mathrm{d}\Omega - \oint_{\partial\Omega} w \tau_{ij} n_j \, \mathrm{d}\Gamma$$

$$+ \int_\Omega \frac{\partial w}{\partial x_\ell} \varepsilon\psi \left(c(\alpha a_\ell a_j + \alpha^N a_\ell^N a_j^N) \frac{\partial m_i}{\partial x_j} + a_\ell \frac{\partial}{\partial x_j} \left(\frac{m_i m_j}{\rho} \right) + a_\ell \delta \frac{\partial p}{\partial x_i} \right) \mathrm{d}\Omega = 0 \tag{17}$$

A similar surface-integral procedure is employed at a subsonic outlet region to enforce specified boundary conditions on deviatoric surface tractions $\tau_{ij} n_j$ and pressure p. With reference to decomposition (5) and Figure 1(b), the divergence of the pressure flux with components f_j^p within the momentum equations (17) is integrated by parts at an outlet region, an operation that yields the outflow boundary statement

$$\int_\Omega w \left(\frac{\partial m_i}{\partial t} + \frac{\partial}{\partial x_j} \left(\frac{m_j}{\rho} m_i \right) \right) \mathrm{d}\Omega - \int_\Omega \left(\frac{\partial w}{\partial x_i} p - \frac{\partial w}{\partial x_j} \tau_{ij} \right) \mathrm{d}\Omega + \oint_{\partial\Omega} w (p n_i - \tau_{ij} n_j) \, \mathrm{d}\Gamma$$

$$+ \int_\Omega \frac{\partial w}{\partial x_\ell} \varepsilon\psi \left(c(\alpha a_\ell a_j + \alpha^N a_\ell^N a_j^N) \frac{\partial m_i}{\partial x_j} + a_\ell \frac{\partial}{\partial x_j} \left(\frac{m_j}{\rho} m_i \right) + a_\ell \delta \frac{\partial p}{\partial x_i} \right) \mathrm{d}\Omega = 0 \tag{18}$$

This statement features an outflow surface integral that depends on the outlet-pressure specific-force components $p n_i$, $1 \leqslant i \leqslant 2$. The outlet pressure boundary condition is therefore directly enforced by specifying the prescribed outlet pressure for p within this surface integral. This strategy for imposing an outlet pressure boundary condition remains intrinsically stable as the following basic considerations on the linear-momentum equation for m_1 indicate; similar conclusions apply to the linear-momentum equation for m_2. Consider first the case of an outlet with $n_1 < 0$, which implies $\partial w / \partial x_1 < 0$ and $m_1 < 0$. If some computational perturbation induced a decrease in m_1 at the boundary, then $\| m_1 \|$ would increase hence p from (6) would decrease, which through the boundary domain integral of pressure in (18) would induce a restoring increase in m_1; similar stability conclusions would result by considering a perturbation increase of m_1 at the boundary. Consider the other case of an outlet with $n_1 > 0$, which implies $\partial w / \partial x_1 > 0$ and $m_1 > 0$. If some computational perturbation induced a decrease in m_1 at the boundary, then p from (6) would increase, which through the boundary domain integral of pressure in (18) would induce a restoring increase in m_1; similar stability conclusions result

by considering a perturbation increase of m_1 at the boundary. The results in Section 6 confirm the accuracy and stability of this pressure boundary-condition enforcement procedure.

4. FINITE ELEMENT GALERKIN WEAK STATEMENT

Since the characteristics flux divergence (4) is developed independently and before any discretization, a genuinely multi-dimensional upstream-bias approximation for the governing equations (1) on arbitrary grids directly results from a classic centred discretization of the characteristics-bias flux on the prescribed grid. To this end a Galerkin finite element method, [2–4], is employed to discretize in space the integral statement (11). This method not only accommodates arbitrary geometries or generates consistent non-extrapolation boundary equations for q, but also retains the ideal surface-integral venues of the integral statement to enforce the boundary conditions described in the previous section.

The discrete solution q^h, subject to prescribed boundary conditions, is sought within a finite dimensional subspace $\mathscr{H}^{h1}(\Omega^h) \subseteq \mathscr{H}^1(\Omega)$ of dimension N, for all $w^h \in \mathscr{H}^{h1}(\Omega^h)$, where superscript '$h$' signifies spatial discrete approximation, Ω^h denotes the discrete computational domain, and N indicates the number of computational nodes. The corresponding finite element statement associated with (11) is

$$\int_{\Omega^h} w^h \left(\frac{\partial q^h}{\partial t} + \frac{\partial f_j^h}{\partial x_j} \right) \mathrm{d}\Omega + \int_{\Omega^h} \frac{\partial w^h}{\partial x_j} f_j^{v^h} \, \mathrm{d}\Omega - \oint_{\partial\Omega^h} w^h f_j^{v^h} n_j \, \mathrm{d}\Gamma$$

$$+ \int_{\Omega^h} \frac{\partial w^h}{\partial x_i} \varepsilon^h \psi^h \left(c^h (\alpha^h a_i^h a_j^h + \alpha^{N^h} a_i^{N^h} a_j^{N^h}) \frac{\partial q^h}{\partial x_j} + a_i^h \frac{\partial f_j^{q^h}}{\partial x_j} + a_i^h \delta^h \frac{\partial f_j^{p^h}}{\partial x_j} \right) \mathrm{d}\Omega = 0 \quad (19)$$

with similar expressions for statements (12)–(18). The approximation q^h exists on a partition Ω^h, $\Omega^h \subseteq \Omega$, of Ω. Having its boundary nodes on the boundary $\partial\Omega$ of Ω, this partition Ω^h results from the union of N_e non-overlapping elements Ω_e, $\Omega^h = \bigcup_{e=1}^{N_e} \Omega_e$. Within Ω^h, there exist clusters of 'master' elements Ω_k^M, each cluster comprising only those adjacent elements that share a mesh node $\mathbf{x}_k$, with $1 \leqslant k \leqslant N$, where N denotes the total number of not only mesh nodes, but also master elements.

As Figure 2 shows, the discrete test function w^h within each master element Ω_k^M will coincide with the 'pyramid' basis function $w_k = w_k(\mathbf{x})$, $1 \leqslant k \leqslant N$, with compact support on Ω_k^M. Such a function equals one at node $\mathbf{x}_k$, zero at all other mesh nodes and also identically vanishes both on the boundary segments of Ω_k^M not containing $\mathbf{x}_k$ and on the computational domain outside Ω_k^M.

The discrete solution q^h and flux f_j^h at each time t assume the form of the following group linear combinations

$$q^h(\mathbf{x}, t) \equiv \sum_{\ell=1}^{N} w_\ell(\mathbf{x}) \cdot q^h(\mathbf{x}_\ell, t), \quad f_j^h(\mathbf{x}, t) \equiv \sum_{\ell=1}^{N} w_\ell(\mathbf{x}) \cdot f_j(q^h(\mathbf{x}_\ell, t)) \quad (20)$$

of time-dependent nodal solution values $q^h(\mathbf{x}_\ell, t)$, to be determined, and trial functions, which coincide with the test functions $w_\ell(\mathbf{x})$ for a Galerkin formulation; an analogous expression

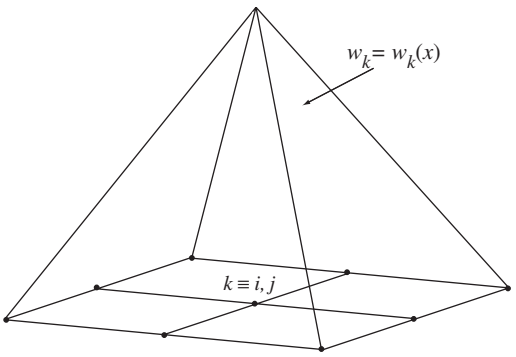

Figure 2. Pyramid test function for Ω_k^M.

applies for $f_j^{v^h}$. Similarly, the fluxes $f_j^q = f_j^q(q(\mathbf{x},t))$ and $f_j^p = f_j^p(q(\mathbf{x},t))$ are discretized through the group expressions

$$f_j^{q^h}(\mathbf{x},t) \equiv \sum_{\ell=1}^{N} w_\ell(\mathbf{x}) \cdot f_j^q(q^h(\mathbf{x}_\ell,t)), \quad f_j^{p^h}(\mathbf{x},t) \equiv \sum_{\ell=1}^{N} w_\ell(\mathbf{x}) \cdot f_j^p(q^h(\mathbf{x}_\ell,t)) \tag{21}$$

The notation for the discrete nodal variable and fluxes is then simplified as $q_\ell(t) \equiv q(\mathbf{x}_\ell,t)$, $f_{j_\ell} \equiv f_j(q^h(\mathbf{x}_\ell,t))$, $f_{j_\ell}^q \equiv f_j^q(q^h(\mathbf{x}_\ell,t))$, $f_{j_\ell}^p \equiv f_j^p(q^h(\mathbf{x}_\ell,t))$ and expansions (20) and (21) are then inserted into (19), which yields the discrete finite element weak statement

$$\int_{\Omega^h} w_k \left(w_\ell \frac{\mathrm{d}q_\ell}{\mathrm{d}t} + \frac{\partial w_\ell}{\partial x_j} f_{j_\ell} \right) \mathrm{d}\Omega + \int_{\Omega^h} \frac{\partial w_k}{\partial x_j} f_j^{v^h} \, \mathrm{d}\Omega - \oint_{\partial\Omega^h} w_k f_j^{v^h} n_j \, \mathrm{d}\Gamma$$

$$+ \int_{\Omega^h} \frac{\partial w_k}{\partial x_i} \frac{\partial w_\ell}{\partial x_j} \varepsilon^h \psi^h [c^h(\alpha^h a_i^h a_j^h + \alpha^{N^h} a_i^{N^h} a_j^{N^h}) q_\ell + a_i^h f_{j_\ell}^q + a_i^h \delta^h f_{j_\ell}^q] \, \mathrm{d}\Omega = 0 \tag{22}$$

for $1 \leqslant k \leqslant N$. There are three implied summations with respect to the subscript indices i, j, ℓ. The subscript indices i, j in this expression denote Cartesian-axis directions, hence $1 \leqslant i, j \leqslant 2$, whereas subscript ℓ indicates a mesh node, hence $1 \leqslant \ell \leqslant N$, although a sum like $\sum_{\ell=1}^{N} w_\ell \frac{\mathrm{d}q_\ell}{\mathrm{d}t}$ only involves a few neighbouring terms because the compact-support test function w_ℓ is only non-zero within a cluster of few neighbouring elements.

While an expansion like the ones in (20) for ψ^h, α^h, c^h, $\mathbf{a}^h$, $\mathbf{a}^{N^h}$ and δ^h can be directly accommodated within (22), each of these variables in this study has been set equal to a piece wise constant for computational simplicity, one centroidal constant value per element. In this study, the upstream-bias controller ψ^h within each domain element has been set equal to 0.5, which induces 50% less dissipation than a fully upwind scheme [6]. The term ε^h is set equal to a reference length within each element, typically a measure of the element size. In this study, $\varepsilon^h = (\Delta\ell)_e/2$ within each element 'e'. Since the streamline is a characteristic principal direction, [1], the term $(\Delta\ell)_e$ is set equal to the length of the streamline diameter of the generalized ellipse inscribed within the element.

Since the test and trial functions w_ℓ are prescribed functions of $\mathbf{x}$, the spatial integrations in (19) are directly carried out. For arbitrarily shaped elements, these integrations take place via

the usual finite element local-coordinate transformation that maps a quadrilateral into a square [2–4]. In this study, the resulting coordinate-transformation metrics within each element are set to the element-wise constant optimal metric data delineated in Reference [5]. This simplification allows the exact integration of the remaining integrals, which are then evaluated only once for each computation. Concerning the boundary variables, no extrapolation of variables is needed in this algorithm on a variable that is not constrained via a Dirichlet boundary condition. In this case, instead, the finite element algorithm (22) naturally generates for each unconstrained boundary variable a boundary-node ordinary differential equation. The complete integration with respect to $\mathbf{x}$ transforms (22) into a system of continuum-time ordinary differential equations (ODE) for determining at each time level t the unknown nodal values $q^h(\mathbf{x}_\ell, t)$, $1 \leqslant \ell \leqslant N$.

5. IMPLICIT RUNGE–KUTTA TIME INTEGRATION

The finite element equations (22), along with appropriate boundary equations and conditions, can be abridged as the non-linear ODE system

$$\mathscr{M} \frac{\mathrm{d}Q(t)}{\mathrm{d}t} = \mathscr{F}(t, Q(t)) \tag{23}$$

where $\mathscr{M} \equiv \{w_k w_\ell\}$ denotes the mass matrix, $\mathscr{M}(\mathrm{d}Q(t)/\mathrm{d}t)$ indicates the corresponding coupling of time derivatives in (22), and $\mathscr{F}(t, Q(t))$ represents the remaining terms in (22). The numerical time integration of (23) in this study takes place through a generalization of results in References [5, 7–9] in the form of an alternative class of two-stage diagonally implicit Runge–Kutta algorithms (IRK2) expressed as

$$\begin{aligned} Q_{n+1} - Q_n &= b_1 K_1 + b_2 K_2 \\ \mathscr{M} K_1 &= \Delta t \cdot \mathscr{F}(t_n + c_1 \Delta t, Q_n + a_{11} K_1) \\ \mathscr{M} K_2 &= \Delta t \cdot \mathscr{F}(t_n + c_2 \Delta t, Q_n + a_{21} K_1 + a_{22} K_2) \end{aligned} \tag{24}$$

where n now denotes a discrete time station. The algorithm is implicit because the determination of the Runge–Kutta arrays K_1 and K_2 requires solving systems of equations; the algorithm is diagonally implicit because the determination of K_1 does not couple K_2. Thus, given the solution Q_n at time t_n, K_1 is computed first, followed by K_2. The solution Q_{n+1} is then determined by way of the first expression in (24). The terms $b_1, b_2, c_1, c_2, a_{11}, a_{21}$, and a_{22} indicate constant Runge–Kutta coefficients, subject to the constraints $c_i = \sum_{j=1}^{2} a_{ij}$ and $\sum_{i=1}^{2} b_i = 1$. It is possible to select such coefficients to achieve both high-order temporal accuracy and non-linear absolute stability for arbitrary stiff non-linear ODE systems. In respect of this integration, the innovative element in this paper involves the concept of non-linear discrete-energy stability, the proof that algorithm (24) is absolutely non-linearly stable in the discrete energy for the Euler and Navier–Stokes equations, and the verification that this author's specific Runge–Kutta coefficient set generates an absolutely energy stable algorithm.

5.1. Non-linear stability and accuracy

The concept of non-linear stability of an implicit Runge–Kutta ODE solver fundamentally refers to the capability of the Runge–Kutta equations (24) to generate a solution Q_n that emulates the time evolution of the time-continuum solution $Q \equiv q^h$ for the time interval of interest and general non-linear rhs $\mathscr{F}(\cdot,\cdot)$. Algorithm (24) satisfies the non-linear energy stability condition that the time-discretization does not contribute an increase in an energy rate of change over an average energy rate of change of the time-continuum system. This condition is expressed as

$$\frac{\|Q_{n+1}\|^2 - \|Q_n\|^2}{\Delta t} \leqslant \overline{\frac{\mathrm{d}\|Q\|^2}{\mathrm{d}t}} \tag{25}$$

where $\|.\|$ denotes an inner-product norm, and overbar signifies an average over the time interval $(t, t+\Delta t)$. An energy rate of change of the discrete system, therefore, does not exceed an associated average energy rate of change of the time continuum system. Algorithm (24) can satisfy (25) regardless of the size of the time step Δt, which implies unconditional energy stability.

To describe how (24) satisfies (25), express the energy associated with the time continuum solution Q of (23) as

$$\|Q\|^2 = (Q, Q) \tag{26}$$

With $F \equiv \mathscr{M}^{-1}\mathscr{F}$, the rate of change of this solution energy becomes a function of Q and the rhs of (23) as

$$\frac{\mathrm{d}\|Q\|^2}{\mathrm{d}t} = 2\left(Q, \frac{\mathrm{d}Q}{\mathrm{d}t}\right) = 2(Q, F) \tag{27}$$

An average energy rate can then be expressed as

$$\overline{\frac{\mathrm{d}\|Q\|^2}{\mathrm{d}t}} \equiv \sum_{i=1}^{s} b_i \left.\frac{\mathrm{d}\|Q\|^2}{\mathrm{d}t}\right|_{Q=Q^i} = 2\sum_{i=1}^{s} b_i (Q^i, F(t^i, Q^i)) \tag{28}$$

Since (28) has to represent a meaningful average, the weights b_i, $1 \leqslant i \leqslant s$ must be non-negative, which leads to the constraints

$$b_i \geqslant 0, \quad 1 \leqslant i \leqslant s \tag{29}$$

Based on (24), the array Q^i in (28) is cast as

$$Q^i \equiv Q_n + \sum_{j=1}^{s} \alpha_{ij} K_j \tag{30}$$

which corresponds to a field at the intermediate time level $t^i \equiv t + c_i \Delta t \in (t, t+\Delta t)$. Together with (24) and $F \equiv \mathscr{M}^{-1}\mathscr{F}$, this form for Q^i leads to the expressions

$$K_i = \Delta t F(t^i, Q^i) \equiv \Delta t F_i, \quad Q_n - Q^i = -\sum_{j=1}^{s} \alpha_{ij} K_j = -\Delta t \sum_{j=1}^{s} \alpha_{ij} F_j \tag{31}$$

The Runge–Kutta algorithm (24) generates for the lhs of (25) the expression

$$\frac{\|Q_{n+1}\|^2 - \|Q_n\|^2}{\Delta t} = 2\sum_{i=1}^{s} b_i(Q^i, F(t^i, Q^i)) - \Delta t \sum_{i,j=1}^{s} (b_i\alpha_{ij} + b_j\alpha_{ji} - b_i b_j)(F(t^i, Q^i), F(t^j, Q^j)) \tag{32}$$

which applies to non-linear expressions $F(\cdot,\cdot)$. This result features a discrete energy rate at the lhs and an average time-continuum energy rate at the rhs. Capitalizing on this result, the energy stability condition (25) then translates into the quadratic-form inequality

$$-\Delta t \sum_{i,j=1}^{s} (b_i\alpha_{ij} + b_j\alpha_{ji} - b_i b_j)(F(t^i, Q^i), F(t^j, Q^j)) \leqslant 0 \tag{33}$$

With $\{\widetilde{b}_{ik}\}$ denoting the diagonal matrix with diagonal entries $\{b_i\}$, this inequality can be unconditionally satisfied for any positive time step Δt when the symmetric energy matrix

$$E \equiv \{\widetilde{b}_{ik}\alpha_{kj} + \alpha_{ki}\widetilde{b}_{kj} - b_i b_j\} \tag{34}$$

is non-negative definite. Employing expressions (30) and (31), the energy result (32) emerges from the square of the norm $\|Q_{n+1}\|$ as

$$\begin{aligned}
\|Q_{n+1}\|^2 &= \left\| Q_n + \sum_{i=1}^{s} b_i K_i \right\|^2 = \left(Q_n + \sum_{i=1}^{s} b_i K_i, Q_n + \sum_{j=1}^{s} b_j K_j \right) \\
&= (Q_n, Q_n) + \sum_{i=1}^{s} b_i (K_i, Q_n) + \sum_{j=1}^{s} b_j (Q_n, K_j) + \sum_{i,j=1}^{s} b_i b_j (K_i, K_j) \\
&= \|Q_n\|^2 + \Delta t \sum_{i=1}^{s} b_i (F_i, Q^i) + \Delta t \sum_{i=1}^{s} b_i (F_i, Q_n - Q^i) \\
&\quad + \Delta t \sum_{j=1}^{s} b_j (Q^j, F_j) + \Delta t \sum_{j=1}^{s} b_j (Q_n - Q^j, F_j) + \Delta t^2 \sum_{i,j=1}^{s} b_i b_j (F_i, F_j) \\
&= \|Q_n\|^2 + 2\Delta t \sum_{i=1}^{s} b_i (Q^i, F_i) \\
&\quad - \Delta t^2 \sum_{i,j=1}^{s} b_i \alpha_{ij} (F_i, F_j) - \Delta t^2 \sum_{i,j=1}^{s} b_j \alpha_{ji} (F_i, F_j) + \Delta t^2 \sum_{i,j=1}^{s} b_i b_j (F_i, F_j) \\
&= \|Q_n\|^2 + 2\Delta t \sum_{i=1}^{s} b_i (Q^i, F_i) - \Delta t^2 \sum_{i,j=1}^{s} (b_i\alpha_{ij} + b_j\alpha_{ji} - b_i b_j)(F_i, F_j)
\end{aligned} \tag{35}$$

Table I. Runge–Kutta coefficients.

	b_1	b_2	a_{11}	a_{21}	a_{22}
IRK2	$\frac{3-\sqrt{3}}{4}$	$\frac{1+\sqrt{3}}{4}$	$\frac{3-\sqrt{3}}{6}$	$2-\sqrt{3}$	$\frac{\sqrt{3}-1}{2}$

Crucially, the derivation of results (32) and (35) places no restriction on either the structure of the residual $F(\cdot,\cdot)$ or the semi-discretization employed. As a result, the diagonally implicit algorithm (24) becomes unconditionally energy stable not just for a traditional linear model problem, but for the full semi-discrete Euler and Navier–Stokes equations on arbitrary structured/unstructured meshes. Based on this result, the stability of a CFD algorithm employing this implicit Runge–Kutta integration is chiefly determined by the upstream-bias formulation, spatial discretization, and boundary conditions, which should collectively generate a stable system (23) of ordinary differential equations in continuum time.

Owing to its symmetry, the matrix E in (34) is non-negative definite when all of its principal minors remain non-negative. For an implicit algorithm it is possible to determine Runge–Kutta coefficients b_i and α_{ij} to satisfy this non-linear energy stability condition independently of Δt. The author's specific Runge–Kutta coefficients satisfy this stability condition and provide second-order time accuracy (Table I). As documented in the Computational Results section, the non-linear stability of this algorithm allows computations to progress stably for Courant numbers in excess of 100.

5.2. Numerical solution

The terminal numerical solution is then determined using Newton's method, which for the implicit fully-coupled computation of the IRK2 arrays K_i, $1 \leqslant i \leqslant 2$, is cast as

$$\left[\mathscr{M} - a_{\underline{ii}}\Delta t \left(\frac{\partial \mathscr{F}}{\partial Q}\right)^p_{Q_i^p}\right](K_i^{p+1} - K_i^p) = \Delta t \mathscr{F}(t_n + c_i\Delta t, Q_i^p) - \mathscr{M}K_i^p$$
$$Q_i^p \equiv Q_n + a_{i1}K_1^p + a_{i2}K_2^p \tag{36}$$

where $a_{ij} = 0$ for $j > i$, p is the iteration index, and $K_1^p \equiv K_1$ for $i = 2$; the Jacobian

$$J_i(Q) \equiv \mathscr{M} - a_{\underline{ii}}\Delta t \left(\frac{\partial \mathscr{F}}{\partial Q}\right)^p_{Q_i^p} \tag{37}$$

has been analytically determined and implemented, leading to a block sparse matrix. For all the results documented in the next section, the initial estimate K_i^0 is set equal to the zero array, while a Gaussian elimination is used with only one iteration executed for (36) within each time interval. In this mode, Newton's iteration becomes akin to a classical direct linearized implicit solver.

6. COMPUTATIONAL RESULTS

The acoustics–convection upstream resolution algorithm has generated accurate essentially non-oscillatory results for subsonic, transonic and supersonic flows, encompassing oblique

and interacting shocks that reflect exact solutions. The benchmarks discussed in this section include four flows: a subsonic flow and a transonic flow about a symmetrical aerofoil and two supersonic intake flows. In order to determine the coarse-grid performance of the algorithm and ultimate accuracy of quadrilateral elements, each benchmark has employed a finite element discretization of Lagrange bilinear elements without any MUSCL-type local extrapolation of variables; each resulting body-fitted grid consists of 40 bilinear elements in the transverse and longitudinal directions, for a total of 1600 elements, 1681 nodes and 6724 degrees of freedom. The computational efficiency of the procedure has remained comparable to that of a conventional centred algorithm for the characteristics-bias system, with upstream directions continuously updated without any filtering or freezing, with high-rate convergence of the residual norm to machine zero.

For each benchmark, the calculations proceeded with a prescribed constant maximum Courant number $C_{\max} = 110$. Considering the definition of the Courant number,

$$C_{\max} \equiv \max\{\|\mathbf{u}\| + c, \|\mathbf{u}\| - c, c\} \frac{\Delta t}{(\Delta \ell)_e} \tag{38}$$

for prescribed $(\Delta \ell)_e$ and $C_{\max}$ for each benchmark, the corresponding Δt is determined as

$$\Delta t = \frac{C_{\max}(\Delta \ell)_e}{\max\{\|\mathbf{u}\| + c, \|\mathbf{u}\| - c, c\}} \tag{39}$$

All the solutions in these validations are presented in non-dimensional form, with pressure p made dimensionless through the corresponding inlet stagnation (total) pressure.

The first computational test corresponds to a subsonic critical flow about a 3% thick symmetrical aerofoil, with stretched grid illustrated in Figure 3.

The subsonic inlet corresponds to a subsonic free-stream Mach number $M_\infty = 0.87$, hence the inlet boundary conditions only constrain density ρ, transverse linear momentum component m_2 and total energy E. The outlet, consisting of both the downstream exit and upper side, remains subsonic, hence static pressure is constrained at this boundary. At the lower surface, the inviscid wall-tangency boundary condition is enforced according to the method delineated

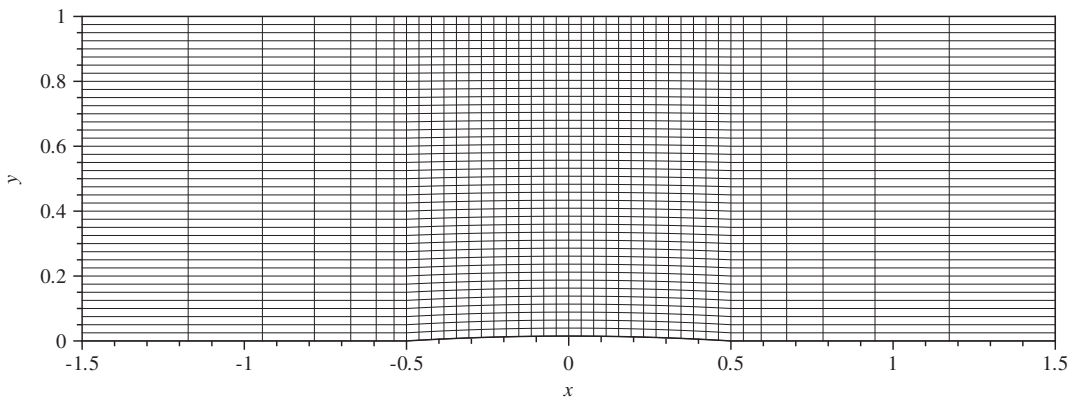

Figure 3. Aerofoil flows: computational grid.

in Section 3. A pressure drop is imposed upon an initially quiescent field and the final steady state is computationally achieved by advancing the solution in time.

The Mach-number distribution and flooded contours in Figures 4 and 5 portray a non-oscillatory solution with sharply resolved drops in Mach number at the aerofoil leading and trailing edges and undistorted capturing of a vanishingly small supersonic region over the aerofoil surface, towards the trailing edge.

Despite the stretched grid, a similar essentially non-oscillatory field is portrayed in the pressure distribution and flooded contours in Figures 6 and 7. Although this distribution indicates increased accuracy would result from a locally refined grid, the pressure peaks at the aerofoil leading and trailing edges remain undistorted. In particular the calculated pressure

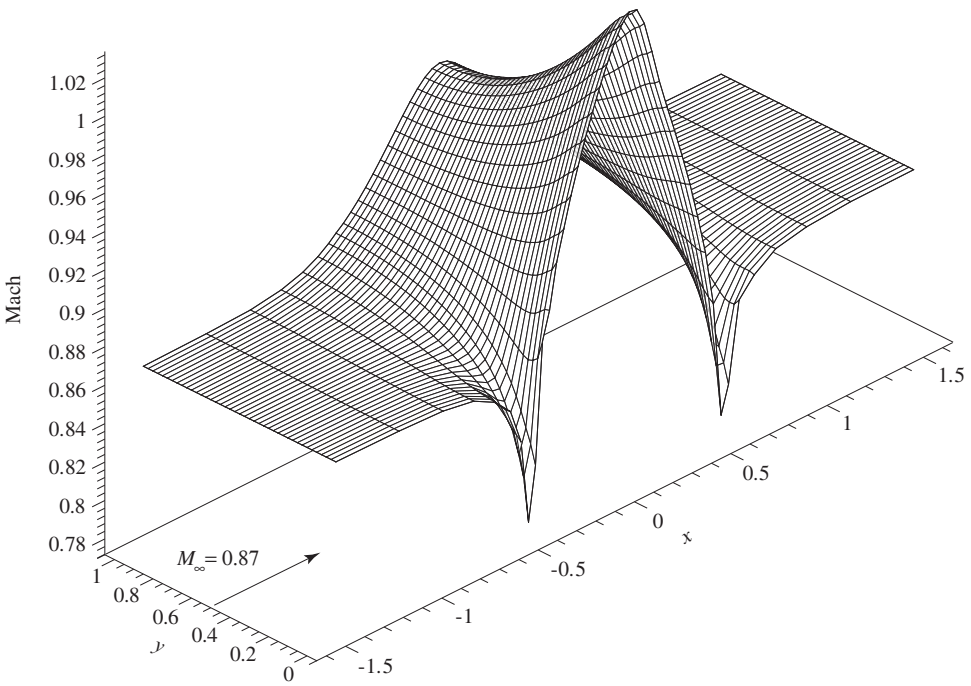

Figure 4. $M_\infty = 0.87$ aerofoil critical subsonic flow, Mach number contours.

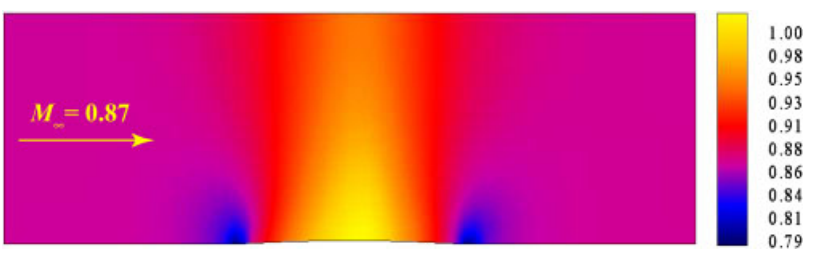

Figure 5. $M_\infty = 0.87$ aerofoil critical subsonic flow, Mach number distribution.

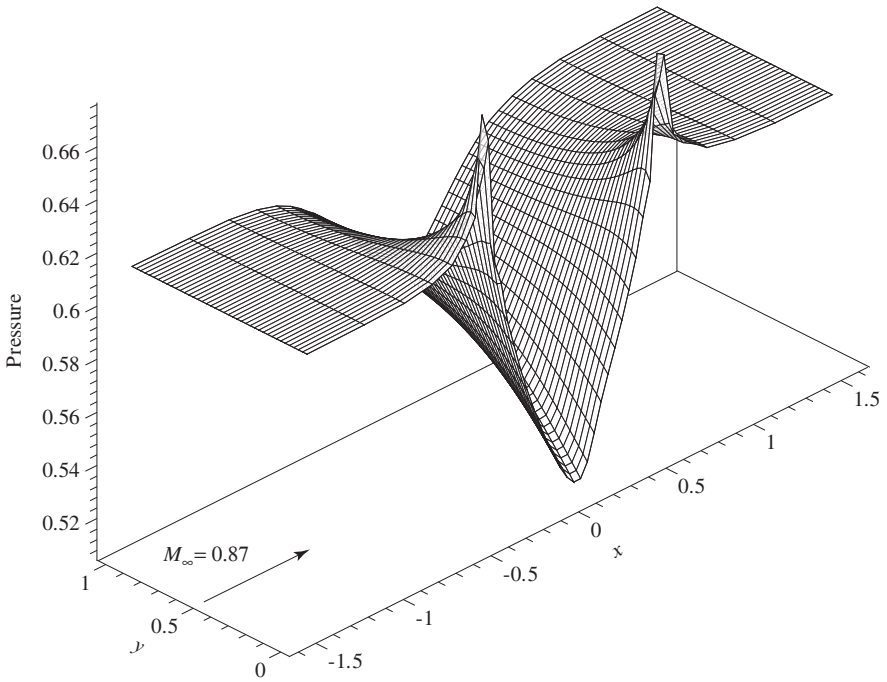

Figure 6. $M_\infty = 0.87$ aerofoil critical subsonic flow, pressure distribution.

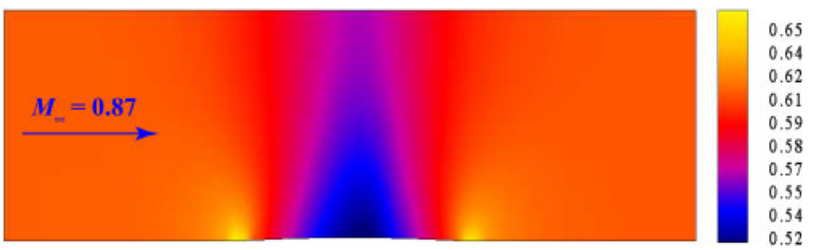

Figure 7. $M_\infty = 0.87$ aerofoil critical subsonic flow, pressure contours.

in the outlet region remains smooth and the calculated outlet pressure at the $x = 1.5$ outlet coincides with the imposed pressure boundary conditions, which reflects favourably on the surface-integral pressure enforcement strategy delineated in Section 3.

The second computational test corresponds to a transonic flow about the same aerofoil. The subsonic inlet corresponds to a subsonic free-stream Mach number $M_\infty = 0.87$, hence the inlet boundary conditions only constrain density ρ, transverse linear momentum component m_2 and total energy E. As in the first test, the outlet, consisting of both the downstream exit and portions of the upper side of the grid, remain subsonic, hence static pressure is constrained at these boundary segments. When the local Mach number exceeds one, the algorithm no longer enforces an outlet pressure boundary condition. At the lower surface, the inviscid

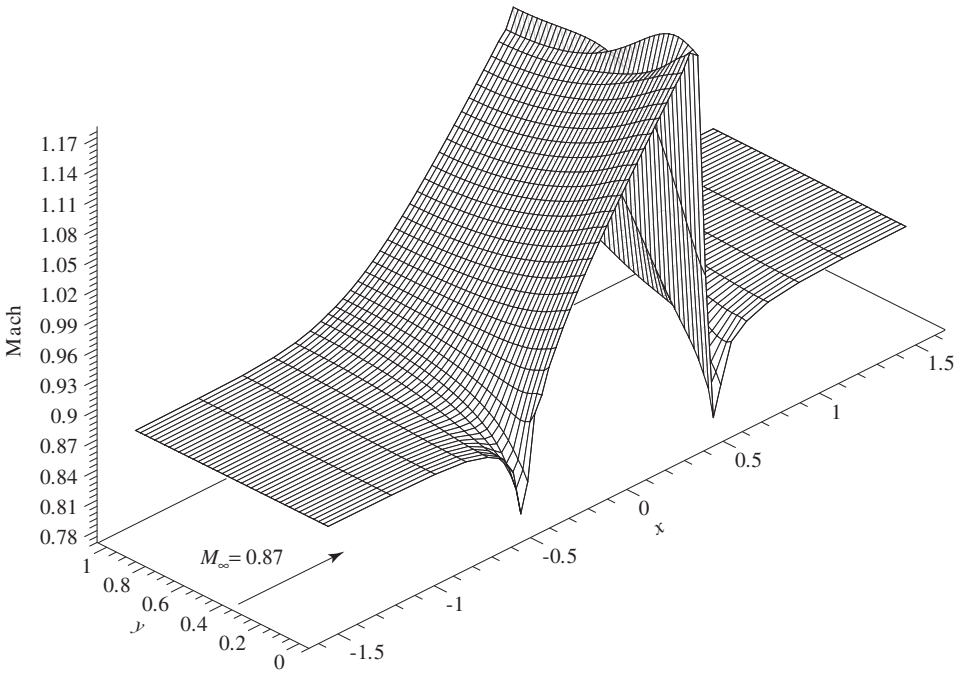

Figure 8. $M_\infty = 0.87$ aerofoil transonic flow, Mach number distribution.

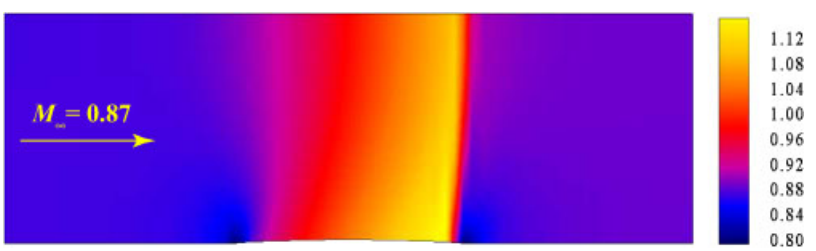

Figure 9. $M_\infty = 0.87$ aerofoil transonic flow, Mach number contours.

wall-tangency boundary condition is enforced according to the method delineated in Section 3. A pressure drop is imposed upon an initial field corresponding to the steady-state flow of the previous test and the final steady state is computationally achieved by advancing the solution in time. The imposed pressure drop was selected to induce a supersonic region that extends past the upper boundary, in order to test the capability of the algorithm to allow a captured shock to cross a boundary with minimal reflection.

The Mach-number distribution and flooded contours in Figures 8 and 9 present an essentially non-oscillatory solution with crisply calculated compressions and expansions, at the aerofoil leading and trailing edges, and with a supersonic pocket terminated by a slightly curved sharp

shock adjacent to the aerofoil trailing edge. In particular this shock is captured within two to three nodes.

Despite the stretched grid, a similar essentially non-oscillatory field is displayed in the pressure distribution and flooded contours in Figures 10 and 11.

This distribution indicates increased accuracy would result from a locally refined grid. Nevertheless, the pressure peaks at the aerofoil leading and trailing edges remain undistorted and the slightly curved shock is sharply captured within two to three nodes. The calculated subsonic distribution downstream of the shock remains smooth. In particular, the calculated outlet pressure at the $x = 1.5$ outlet coincides with the imposed pressure boundary conditions,

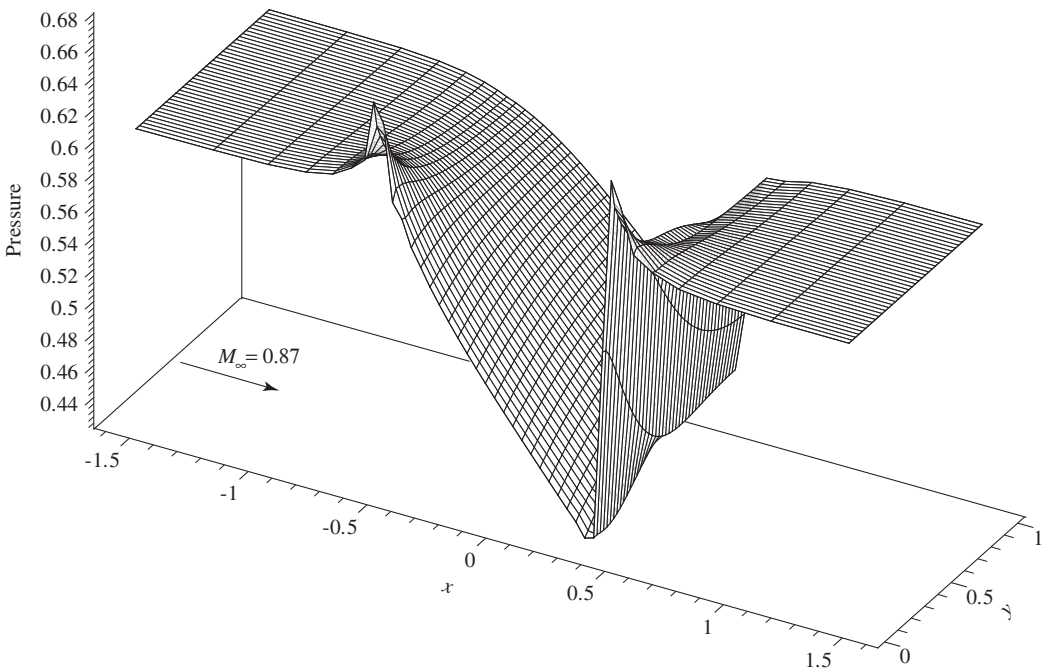

Figure 10. $M_\infty = 0.87$ aerofoil transonic flow, pressure distribution.

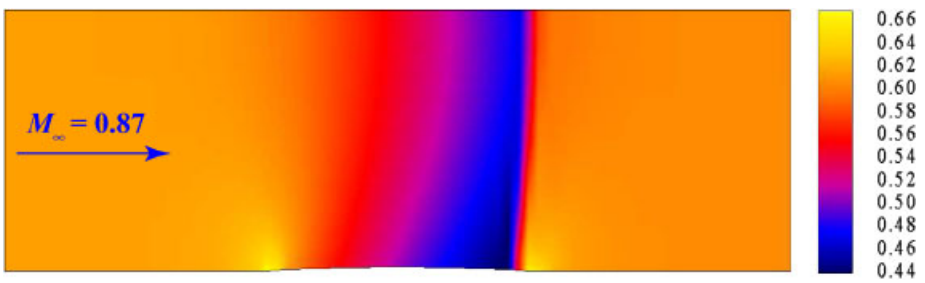

Figure 11. $M_\infty = 0.87$ aerofoil transonic flow, pressure contours.

which again reflects favourably on the surface-integral pressure enforcement strategy delineated in Section 3.

The next computational tests encompass two supersonic intake flows. One of these flows involves an oblique shock reflection case, with grid illustrated in Figure 12.

The supersonic intake corresponds to a free-stream Mach number $M_\infty = 2.40$, hence the inlet boundary conditions constrain density ρ, longitudinal and transversal linear momentum components m_1 and m_2 and total energy E. The outlet remains supersonic, hence no boundary conditions are enforced at this boundary. At the solid upper and lower walls, the inviscid wall-tangency boundary condition is enforced using the method in Section 3. An initially uniform supersonic $M = 2.40$ shockless flow is subject to a 5° deflection by the lower wall and the final steady state is computationally achieved by advancing the solution in time.

It is the wall-tangency boundary condition on the whole upper and lower boundary walls that induces emergence of an oblique shock at the lower ramp initiation point, a shock that propagates toward the upper wall and is then reflected downward towards the outlet. Apart from a minimal ripple confined to the ramp initiation point, a ripple in the Mach number and pressure distributions presumably caused by the need for localized grid refinement, the Mach-number distribution and flooded contours in Figures 13 and 14 present an essentially non-oscillatory solution with crisply calculated incident and reflected shocks. In particular, the algorithm allows the reflected shock to cross the outflow boundary unperturbed, without any spurious distortion. Significantly, this computational solution mirrors the available exact solution, with three juxtaposed plateaus connected by two oblique shocks. The calculated Mach numbers in the plateaus downstream of the two shocks are $M_2 = 2.20$ and $M_3 = 2.01$; the shock inclination angles are $\theta_2 = 28.53°$ and $\theta_3 = -31.12°$. Not only for Mach number and shock angles but also for pressure do these computed results coincide with the corresponding exact values.

A similar essentially non-oscillatory field is displayed in the pressure distribution and flooded contours in Figures 15 and 16. The incident and reflected pressure shocks are crisply calculated, with a reflected shock that can cross the outflow boundary without any spurious distortion.

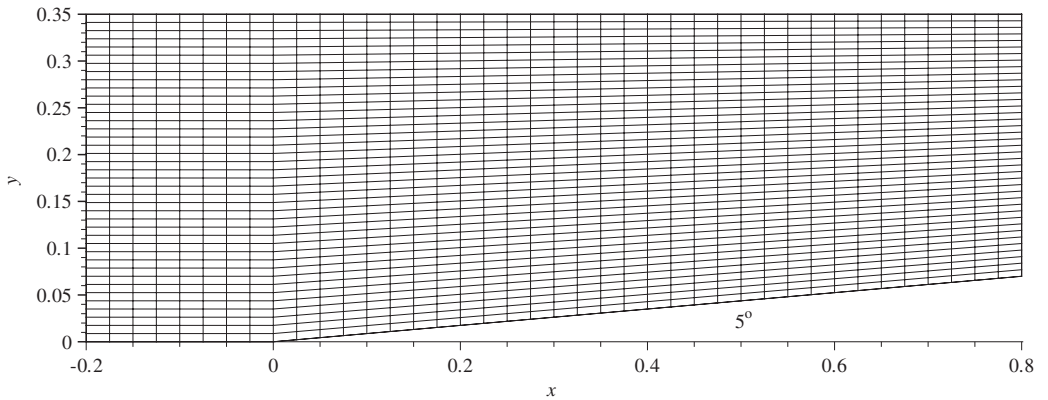

Figure 12. $M_\infty = 2.40$ shock reflection, computational grid.

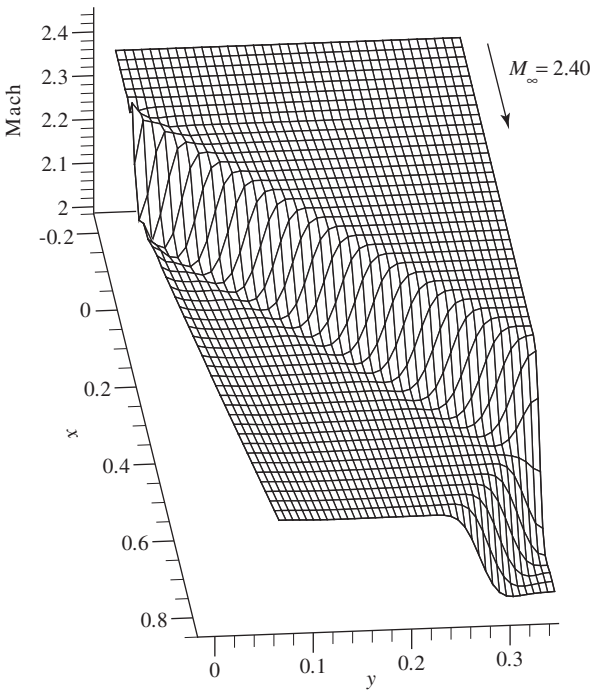

Figure 13. $M_\infty = 2.40$ shock reflection, Mach number distribution.

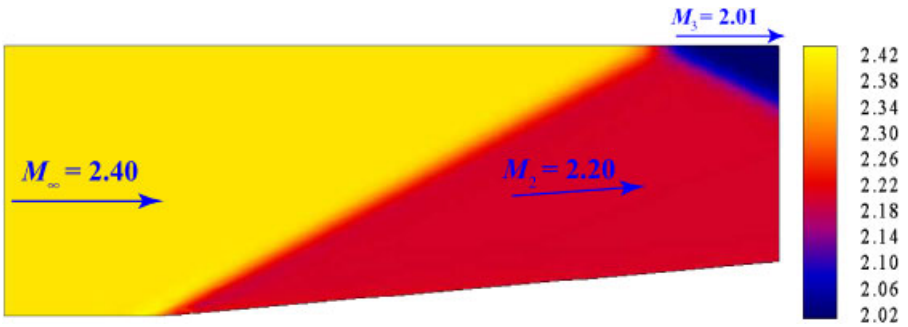

Figure 14. $M_\infty = 2.40$ shock reflection, Mach number contours.

The last test case in this study involves the asymmetric interaction of two oblique shocks, with grid illustrated in Figure 17. The supersonic intake corresponds to a free-stream Mach number $M_\infty = 2.40$, hence the inlet boundary conditions constrain density ρ, longitudinal and transversal linear momentum components m_1 and m_2 and total energy E. The solid wall specification only extends to $x = 0.4$; beyond this station, the upper and lower boundaries correspond to outlets, to allow the reflected shocks to cross the computational boundaries with minimal further reflection. The total outlet remains supersonic, hence no boundary conditions are enforced at this boundary. At the solid upper and lower walls, only extending to $x = 0.4$,

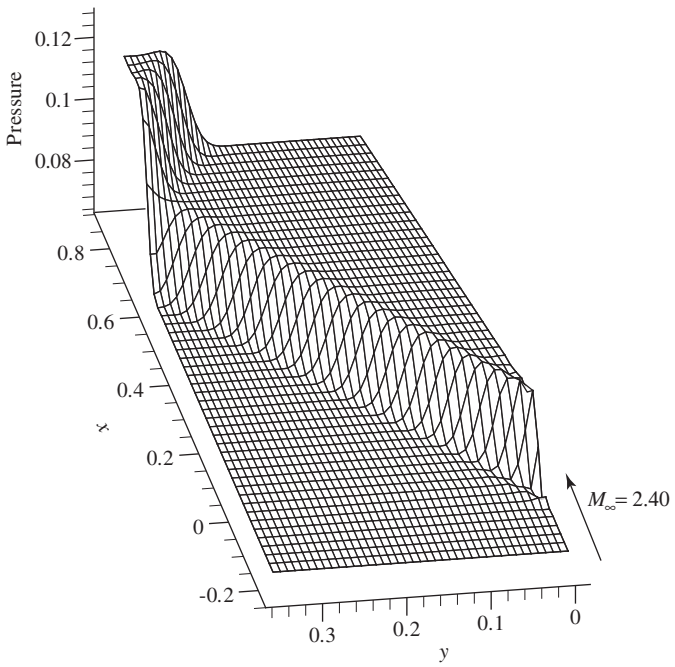

Figure 15. $M_\infty = 2.40$ shock reflection, pressure distribution.

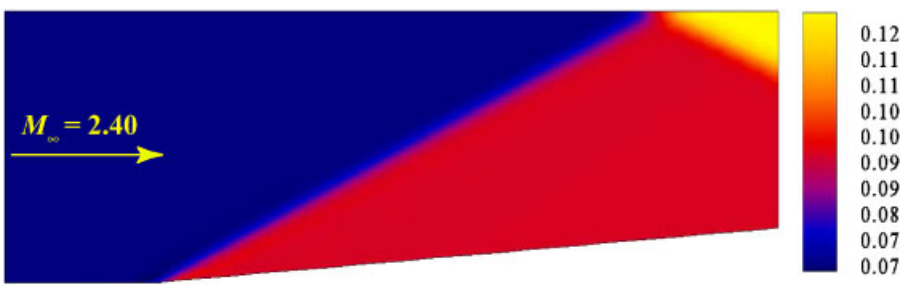

Figure 16. $M_\infty = 2.40$ shock reflection, pressure contours.

the inviscid wall-tangency boundary condition is enforced using the method in Section 3. An initially uniform supersonic $M = 2.40$ shockless flow is subject to a 5° deflection by the lower wall and a 3° deflection by the lower wall; the final steady state is computationally achieved by advancing the solution in time.

The wall-tangency boundary conditions on the upper and lower boundary walls induce emergence of two separate oblique shocks of different strengths, shocks that propagate toward each other, interact, and reflect away from each other, towards the outlet. The minimal ripples at the origin of the primary oblique shocks remain confined to the starting point of the two ramps, ripples in the Mach number and pressure distributions presumably caused by the need for localized grid refinement. Nevertheless, the Mach-number distribution and flooded contours in Figures 18 and 19 present an essentially non-oscillatory solution with crisply

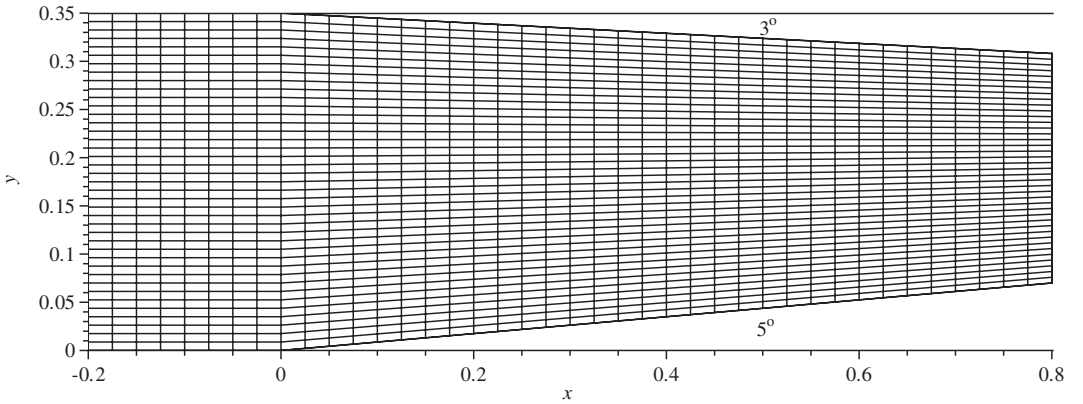

Figure 17. $M_\infty = 2.40$ shock-on-shock interaction, computational grid.

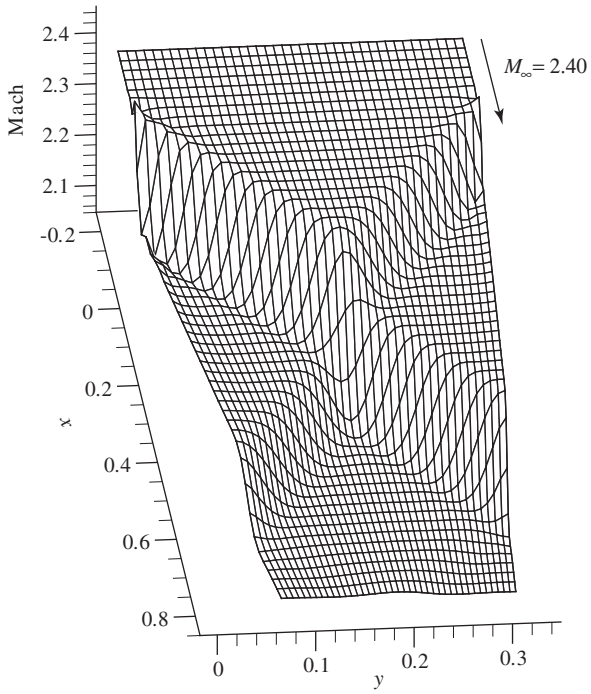

Figure 18. $M_\infty = 2.40$ shock-on-shock interaction, Mach number distribution.

calculated incident and reflected shocks. Significantly, this computational solution mirrors the available exact solution, with four juxtaposed plateaus connected by four oblique shocks. Since the ramp deflection angles remain somewhat moderate and close to each other, the velocity magnitudes and Mach numbers remain essentially constant across the slip line that originates at the reflected-shock intersection point. Nevertheless, the reflected shock interaction rotates the emerging velocity vector to the equilibrium angle of 2° with respect to the horizontal direction.

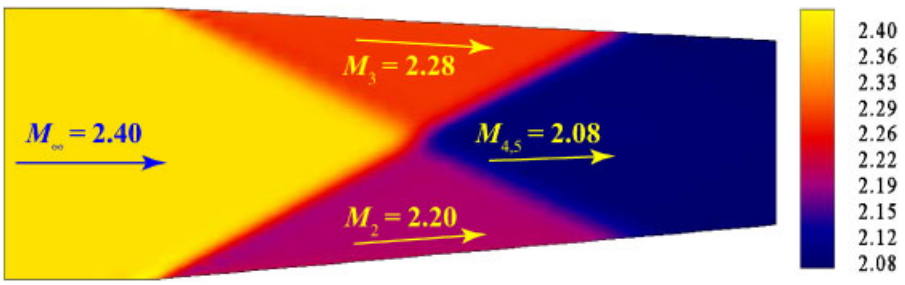

Figure 19. $M_\infty = 2.40$ shock-on-shock interaction, Mach number contours.

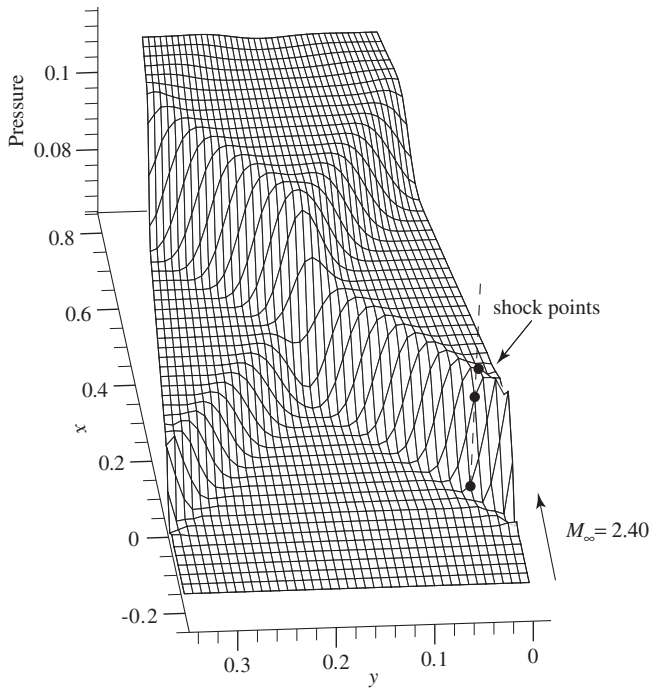

Figure 20. $M_\infty = 2.40$ shock-on-shock interaction, pressure distribution.

This angle corresponds to a deflection of 3°, across the lower reflected shock, and a deflection angle of −5°, across the upper reflected shock. These calculated velocity and deflection angles mirror the exact values. The calculated Mach numbers in the plateaus downstream of the four shocks are $M_2 = 2.20$, $M_3 = 2.28$, $M_4 \simeq M_5 = 2.08$; the shock inclination angles are $\theta_2 = 28.53°$, $\theta_3 = -26.90°$, $\theta_4 = 29.52$ and $\theta_5 = -30.15$. Not only for Mach number and shock angles but also for pressure do these computed results coincide with the corresponding exact values.

A similar essentially non-oscillatory field is displayed in the pressure distribution and flooded contours in Figures 20 and 21. The incident and reflected pressure shocks are crisply calculated, with reflected shocks that can cross the outflow boundary without any spurious dis-

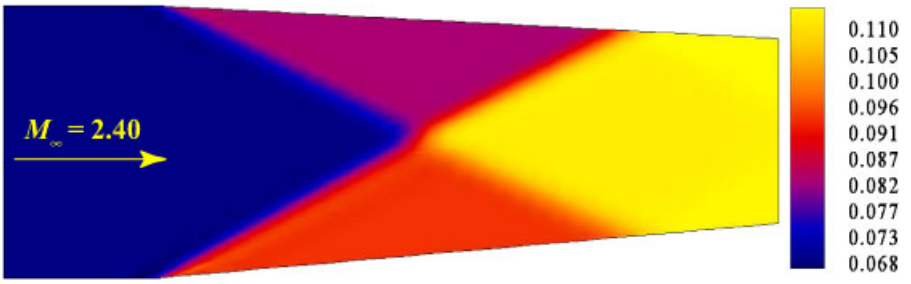

Figure 21. $M_\infty = 2.40$ shock-on-shock interaction, pressure contours.

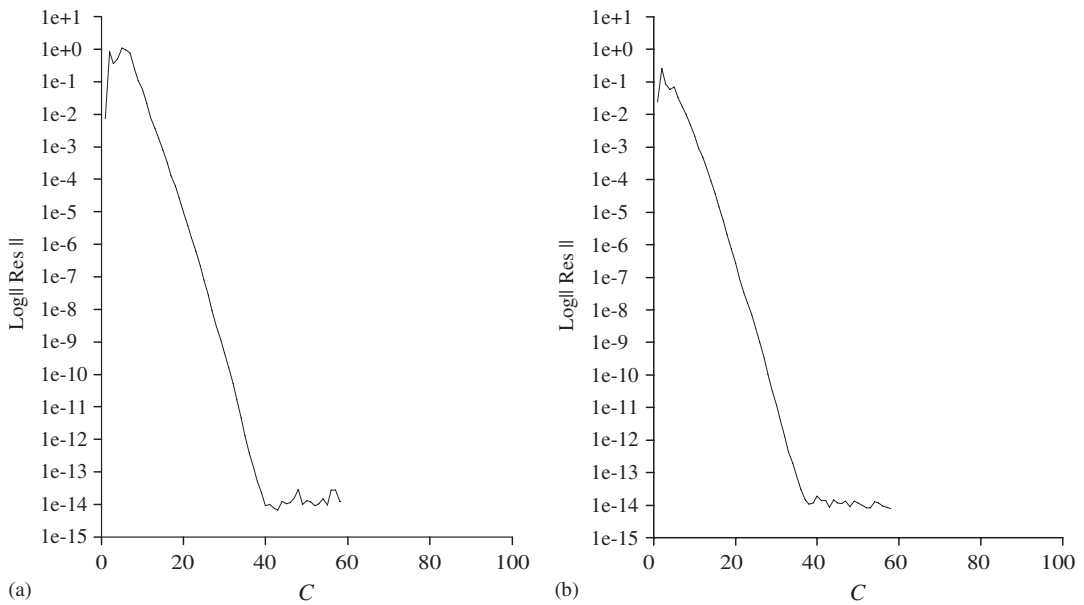

Figure 22. Convergence rate: (a) shock reflection; and (b) shock interaction.

tortion. In particular, the algorithm allows the reflected shocks to cross the outflow boundary essentially unperturbed, without any spurious distortion.

For all the flow cases discussed, the algorithm has rapidly calculated a steady state, as exemplified by Figure 22 for the shock reflection and interaction problems.

These curves document the high-rate convergence rate of the algorithm, with a reduction of the residual norm to 1×10^{-14}, hence machine zero, achieved in about 45 cycles 'C'at a constant maximum Courant number in excess of 100. The recorded essentially monotone decrease of the residual is seen to correspond to an exponential convergence rate.

7. CONCLUDING REMARKS

The characteristics-bias procedure detailed in this two-paper series generates the upstream bias at the differential equation level, before any discrete approximation, within a characteristics-bias system associated with the Euler and Navier–Stokes equations. Along all the infinite directions of wave propagation, the formulation induces anisotropic and variable-strength consistent upwinding that correlates with the spatial distribution of characteristic velocities.

A classical finite element Galerkin discretization of the characteristics-bias system is then used for the spatial approximation. The developments in this investigation have implemented the algorithm using a bi-linear approximation of fluxes within quadrilateral four-noded cells without any MUSCL-type local extrapolation of variables, to generate a computationally efficient algorithm and determine the ultimate accuracy of bi-linear resolutions.

This formulation has also developed an efficient strategy to enforce wall-tangency and static-pressure boundary conditions. This strategy has been validated by the computational solution fields, which remain undistorted. The time-dependent discrete equations are then integrated in time via an implicit Runge–Kutta procedure with analytically determined Jacobians. This procedure remains non-linearly stable for the Euler and Navier–Stokes equations and affords rapid convergence to steady states, with continuously updated upstream directions and maximum Courant number exceeding 100.

Relying upon the physics and mathematics of multi-dimensional characteristic acoustics and convection, the acoustics–convection upstream resolution algorithm generates an intrinsically multi-dimensional upstream approximation for the Euler and Navier–Stokes equations for general equations of state. Coupled with a classical finite element discretization and featuring a computational simplicity that parallels a traditional centred discretization, even on relatively coarse grids the algorithm rapidly generates solutions for subsonic, transonic and supersonic flows, solutions that both remain essentially non-oscillatory and reflect reference exact solutions.

REFERENCES

1. Iannelli J. Derivation and characteristics analysis of an acoustics convection upstream resolution algorithm for the two-dimensional Euler and Navier–Stokes equations. *International Journal for Numerical Methods in Fluids* 2005, this issue.
2. Baker AJ. *Finite Element Computational Fluid Dynamics*. Taylor & Francis: London, 1983.
3. Johnson C. *Numerical Solution of Partial Differential Equations by the Finite Element Method*. Cambridge, 1987.
4. Zienkiewicz OC, Taylor RL. *The Finite Element Method*, vols. 1, 3. Butterworth & Heinemann: Stoneham, MA, 2000.
5. Iannelli JS, Baker AJ. A non-linearly stable implicit finite element CFD algorithm for hypersonic aerodynamics. *International Journal for Numerical Methods in Engineering* 1992; **34**:419–441.
6. Iannelli J. A CFD Euler solver from a physical acoustics-convection flux Jacobian decomposition. *International Journal for Numerical Methods in Engineering* 1999; **31**:821–860.
7. Burrage K, Butcher JC. Non-linear stability for implicit Runge–Kutta methods. *SIAM Journal on Numerical Analysis* 1979; **16**:46–57.
8. Crouzeix M. Sur la B-Stabilite des methodes de Runge Kutta. *Numerische Mathematik* 1979; **32**:75–82.
9. Dekker K, Verwer JG. *Stability of Runge–Kutta methods for Stiff Non-linear Differential Equations*. Elsevier Publishers: Amsterdam, 1984.
10. Garabedian PR. *Partial Differential Equations*. Chelsea Publishing Company: New York, NY, 1986.
11. Zauderer E. *Partial Differential Equations of Applied Mathematics*. Wiley: New York, NY, 1989.
12. Oden JT, Reddy JN. *An Introduction to the Mathematical Theory of Finite Elements*. Wiley: New York, NY, 1976.
13. Hutson V, Pym JS. *Applications of Functional Analysis and Operator Theory*. Academic Press: New York, 1980.

