INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2006; **50**:1269–1293
Published online 20 December 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1118

MPDATA error estimator for mesh adaptivity

Joanna Szmelter[1,*,†] and Piotr K. Smolarkiewicz[2,‡]

[1] *Cranfield University, Royal Military College of Science, Shrivenham, Swindon SN6 8LA, U.K.*
[2] *National Center for Atmospheric Research, Boulder, CO 80307, U.S.A.*

SUMMARY

In multidimensional positive definite advection transport algorithm (MPDATA) the leading error as well as the first- and second-order solutions are known explicitly by design. This property is employed to construct refinement indicators for mesh adaptivity. Recent progress with the edge-based formulation of MPDATA facilitates the use of the method in an unstructured-mesh environment. In particular, the edge-based data structure allows for flow solvers to operate on arbitrary hybrid meshes, thereby lending itself to implementations of various mesh adaptivity techniques. A novel unstructured-mesh nonoscillatory forward-in-time (NFT) solver for compressible Euler equations is used to illustrate the benefits of adaptive remeshing as well as mesh movement and enrichment for the efficacy of MPDATA-based flow solvers. Validation against benchmark test cases demonstrates robustness and accuracy of the approach. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: mesh refinement; nonoscillatory forward-in-time schemes; finite volume methods; unstructured meshes

1. INTRODUCTION

This paper examines properties of multidimensional positive definite advection transport algorithm (MPDATA) for the purpose of devising refinement indicators for mesh adaptivity. Adaptive techniques are a common tool used in engineering applications. In general, they improve the solution accuracy in problems where regions of steep gradients are embedded in large areas with slowly varying values of unknowns, when the steep-gradient regions either emerge or disappear, or change shape and location. For problems with time-dependent geometry—e.g. via movement of external/internal boundaries—mesh adaptivity may not be an option but rather a necessary device for enabling the simulation. Two primary aspects of

*Correspondence to: Joanna Szmelter, Cranfield University, Royal Military College of Science, Shrivenham, Swindon SN6 8LA, U.K.
†E-mail: j.m.szmelter@cranfield.ac.uk
‡E-mail: smolar@ncar.ucar.edu

Contract/grant sponsor: U.S. Department of Energy

Received 3 April 2005
Revised 29 July 2005
Accepted 11 August 2005

adaptivity are: (1) the mesh adaptation—a set of tools for providing the mesh with specified spatial discretization; and (2) the refinement indicator—a set of criteria for specifying desired spatial discretization. For fluid flow problems, several distinct mesh adaptation techniques, and their various combinations, have been used in practice: mesh movement; mesh enrichment (alias h-refinement); mesh regeneration; and p-refinement, i.e. rising the order of interpolating polynomials [1–7]. Our recent development of the edge-based formulation of MPDATA [8, 9] opens new possibilities for constructing MPDATA-based nonoscillatory forward-in-time (NFT) flow solvers on arbitrary hybrid meshes, with subsequent implementations of a variety of refinement techniques familiar from the unstructured-mesh framework.

Technically, MPDATA [10, 11] consists of a sequence of upwind iterations, where the initial iteration is the first-order-accurate upwind scheme, while the subsequent iterations are designed to compensate for the error of the preceding step. The corrective upwind iterations use a pseudo-velocity constructed from the leading (dissipative) truncation error of the preceding iteration. Thus, the scheme itself, in the way it refines the first-order to the second-order solution, may be interpreted as a form of p-refinement. By design, at the end of every upwind iteration the leading error of the iteration is explicitly known. This information can be readily applied for the solution error estimation and the refinement indicator design. Alternatively, the knowledge of the first- and second-order-accurate solutions can be used in refinement indicators following ideas akin to Richardson extrapolation [12]. Moreover, an *a posteriori* error estimation is feasible, for the sake of reducing simulation errors in specified objective functionals (e.g. drag and/or lift) in the spirit of the dual approach [13, 14].

Progress with the edge-based unstructured-mesh MPDATA formulation [9] has naturally lead to edge-based NFT flow solvers [15]. The NFT schemes [16, 17] for integrating fluid equations have been widely documented in the literature for structured grids. The term (nonoscillatory forward-in-time) 'NFT' was introduced in the late 1990s [18, 19] to label a class of second-order-accurate two-time-level algorithms for fluids built on modern nonlinear advection techniques that suppress/ reduce/control numerical oscillations characteristic of higher-order linear schemes. NFT is meant to distinguish these algorithms from classical centred-in-time-and-space linear methods. The interested reader is referred to Reference [20] for a comprehensive review of NFT solvers and discussion of their options; the theoretical foundations of the approach are briefly explained in Reference [21]. Since NFT solvers do not depend on the details of spatial discretization—but only assume a second-order-accurate nonoscillatory advection scheme as a building block, e.g. MPDATA—the numerical procedure is common to flux-form finite-difference and unstructured-mesh finite-volume discretization.[§] In general, the NFT approach admits the Eulerian (viz. control volume wise) [20] and semi-Lagrangian (viz. trajectory wise) [22] optional integration methods for both elastic and inelastic systems. In this paper, we focus on an Eulerian scheme suitable for high-Mach number (elastic) flow applications.

Central difference schemes using artificial dissipation models are popular in aeronautical applications for their efficiency. Their disadvantages include a tendency to smear shocks over several cells, and the dependence of the artificial dissipation on constants that have to be

[§]Although MPDATA is formally finite-volume by design, for distinction we refer to earlier formulations as 'finite-difference', because they were derived from finite differencing of flux-form continuous PDEs of generalized advective transport, while assuming structured grids; cf. Reference [21] in the same issue.

calibrated for each class of problems at hand. Moreover, designing the artificial dissipation on unstructured meshes, while avoiding overdiffusive solutions, is not straightforward. The alternatives include higher-order-accurate upwind algorithms with extensions to systems of conservation laws using flux–vector splitting, approximate Riemann solvers, and flux limiters that minimize spurious numerical oscillations. These schemes are less efficient as they require more mathematical operations per node than central difference approaches; see References [23, 24] and references therein for discussions. The NFT algorithm pursued in this paper is based on MPDATA. It evinces small implicit viscosity, and is robust throughout a broad range of flow regimes. Furthermore, it provides nonlinearly stable solutions without a need for tunable parameters characteristic of explicit artificial viscosity schemes.

For completeness, the edge-based unstructured-mesh high-speed flow solver for Euler equations is described in the next section. The solver's validation is discussed in Section 3, with some technical details addressed in Appendix. Together, Sections 2 and 3 lay the foundation for the design and application of adaptive indicators and meshing techniques introduced and compared to theoretical and numerical benchmark solutions in Section 4. Remarks in Section 5 conclude the paper.

2. COMPRESSIBLE FLOW EULER SOLVER

2.1. Governing equations

The NFT solution procedure presented below is particularly efficient when the compressible Euler equations are considered in the conservative form, modified relative to that commonly used for solution of high-speed problems on unstructured meshes. First, rather than combining advective fluxes of momentum with the homogeneous part of stress tensor under the divergence, the governing equation are separated into advection of dependent variables on the lhs, while designating pressure gradients as sources on the rhs. Incidentally, this facilitates designing implicit NFT algorithms for both elastic and inelastic systems [25]. Second, in lieu of the standard total-specific-energy equation, the conservation law for the potential temperature is employed. Since the potential temperature is invariant under adiabatic processes, it enters the system of the Euler equations as a homogeneous advection equation with zero rhs, similarly to the mass continuity equation. An immediate numerical benefit of this formulation is that updating the density and potential temperature prior to momenta, readily provides the updated pressure, thereby facilitating second-order-accurate integrals for kinematic variables. The Euler equations can be written as

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} + \frac{\partial \rho w}{\partial z} = 0 \tag{1a}$$

$$\frac{\partial \Theta}{\partial t} + \frac{\partial \Theta u}{\partial x} + \frac{\partial \Theta v}{\partial y} + \frac{\partial \Theta w}{\partial z} = 0 \tag{1b}$$

$$\frac{\partial q^I}{\partial t} + \frac{\partial q^I u}{\partial x} + \frac{\partial q^I v}{\partial y} + \frac{\partial q^I w}{\partial z} = -\frac{\partial p}{\partial x^I} \tag{1c}$$

where ρ and p denote fluid density and pressure, and the velocity components u, v, w are equivalent, respectively, to $u^I = q^I/\rho$ for $I = 1, 2, 3$ in (1c). $\Theta := \rho\theta$ is the density-weighted

potential temperature, with θ denoting the potential temperature defined as

$$\theta := T(p/p_0)^{-R/c_p} \tag{2}$$

where T is the temperature, R the gas constant, and c_p the specific heat at constant pressure. The constant p_0 in (2), is the reference (free stream) pressure.

The system of Equations (1) is supplemented with the ideal-gas law

$$p = \rho RT \equiv \mathscr{C}\Theta^{\gamma} \tag{3}$$

where $\gamma \equiv (1 - R/c_p)^{-1}$.

2.2. Solution algorithm

Our second-order-accurate NFT algorithm for the governing system (1) is a special case of a time-dependent curvilinear coordinate formulation discussed in Section 4.1 of Reference [21] (cf. Equation (32), therein). Consequently, all three equations in (1) are viewed as

$$\frac{\partial \Psi}{\partial t} + \nabla \cdot \Psi \mathbf{v} = F^{\Psi} \tag{4}$$

where Ψ denotes any of the dependent variables in (1), $\mathbf{v} \equiv [u, v, w]$ and F^{Ψ} is the associated right-hand side ($\equiv 0$ for ρ and Θ). The resulting NFT scheme takes the form

$$\Psi_{\mathbf{i}}^{n+1} = \mathscr{A}_{\mathbf{i}}(\widetilde{\Psi}, \mathbf{v}^{n+1/2}) + 0.5\Delta t F^{\Psi}|_{\mathbf{i}}^{n+1} \tag{5}$$

where, n and $\mathbf{i}$ symbolize the temporal and spatial location in a computational space, $\mathscr{A}$ is shorthand for a NFT advection operator (here MPDATA), Δt denotes the integration time step, $\widetilde{\Psi}$ is an auxiliary dependent variable

$$\widetilde{\Psi}_{\mathbf{i}} = \Psi_{\mathbf{i}}^{n} + 0.5\Delta t F^{\Psi}|_{\mathbf{i}}^{n} \tag{6}$$

and $\mathbf{v}^{n+1/2}$ is an $\mathscr{O}(\Delta t^2)$ estimate of the velocity field at $t + \Delta t/2$. The adopted analytical formulation of the governing Equations (1) results in a particularly simple solution procedure:

Step 1: Evaluate advective velocity $\mathbf{v}_{\mathbf{i}}^{n+1/2}$ by extrapolating linearly from the current n, and the preceding $n-1$ time level¶

$$\mathbf{v}_{\mathbf{i}}^{n+1/2} = 1.5\mathbf{v}_{\mathbf{i}}^{n} - 0.5\mathbf{v}_{\mathbf{i}}^{n-1}$$

and store afterwards $\mathbf{v}_{\mathbf{i}}^{n}$ for the future use as $\mathbf{v}_{\mathbf{i}}^{n-1}$.

Step 2: Evaluate auxiliary variables $\widetilde{\Psi}_{\mathbf{i}}$ according to (6). Since the mass continuity equation is seldom forced, and adiabatic motions are assumed, this step is only required for momenta, i.e.

$$\widetilde{q}_{\mathbf{i}}^{I} = q^{I}|_{\mathbf{i}}^{n} + 0.5\Delta t F^{q^I}|_{\mathbf{i}}^{n}, \quad I = 1, 2, 3$$

where $F^{q^I}|_{\mathbf{i}}^{n} \equiv -\partial p/\partial x^{I}|_{\mathbf{i}}^{n}$. Pressure gradients are not evaluated here, but are stored at the end of the preceding time step, see Step 4 below (note this requires initialization of the right-hand sides).

¶This requires storing an additional level of the velocity field; for alternate procedures consult [20] and references therein.

Step 3: Call MPDATA($\widetilde{\Psi}, \mathbf{v}^{n+1/2}$), for $\widetilde{\Psi}$ denoting $\widetilde{\rho} \equiv \rho$, $\widetilde{\Theta} \equiv \Theta$, and $\widetilde{q}^I$ for $I = 1,2,3$. The output (from the MPDATA advection routine) represents the first term on the rhs of (5). Since neither (1a) nor (1b) contain rhs, the values of ρ and Θ already represent the updated solutions at $n+1$.

Step 4: Using Θ^{n+1} updated in Step 3, compute p^{n+1} from (3)

$$p_{\mathbf{i}}^{n+1} = \mathscr{C}\Theta^{\gamma}|_{\mathbf{i}}^{n+1}$$

evaluate the pressure gradient force

$$F^{q^I}|_{\mathbf{i}}^{n+1} = -\left.\frac{\partial p}{\partial x^I}\right|_{\mathbf{i}}^{n+1}, \quad I = 1,2,3$$

and store it for future use in Step 2 as $F^{q^I}|_{\mathbf{i}}^{n}$.

Step 5: Update momenta according to (5), using auxiliary solutions stored after advection calls in Step 3, and forcings evaluated in Step 4.

Step 6: Evaluate advective velocities at $n+1$ by dividing newly updated momenta by the updated density

$$v^I|_{\mathbf{i}}^{n+1} = \left.\frac{q^I}{\rho}\right|_{\mathbf{i}}^{n+1}$$

This completes the model algorithm (5) over Δt from level n to $n+1$. For advancing the solution from $n+1$ to $n+2$, rename variables accordingly and go back to Step 1.

2.3. Spatial discretization

The scheme summarized in the preceding section has been implemented on hybrid meshes with unstructured edge-based data, using the median dual finite-volume discretization (in the spirit of Reference [26]). The median-dual discretization requires faces S_j of the dual mesh that bound the control volume $\mathscr{V}_i$ containing the vertex i, to intersect the centres of the $j = 1, l(i)$ edges connecting vertex i with its $l(i)$ neighbours—see Figure 1 for a two-dimensional schematic. While allowing for a mean curvature of cell faces, this approach appears optimal for the MPDATA advection—the heart of our NFT flow solver—because it simplifies the scheme without a loss of the second-order accuracy. The complete derivation of the finite-volume MPDATA has been presented in Reference [8]. For the reader's convenience, we only summarize the basic MPDATA, i.e. we provide essential details of the Step 3 in the model algorithm from Section 2.2:

Here, the finite-volume advection employs surface-weighted velocity normal to the cell face S_j, calculated using advective velocities from Step 1

$$v_j^{\perp} = \mathbf{S}_{\mathbf{j}} \cdot 0.5[\mathbf{v}_i^{n+1/2} + \mathbf{v}_j^{n+1/2}] \tag{7}$$

where $\mathbf{S}_j \equiv S_j \mathbf{n}_j$ symbolizes the oriented surface element with $\mathbf{n}_j$ denoting the normal. The upwind fluxes normal to the cell face S_j are then computed from

$$F_j^{\perp}(\widetilde{\Psi}^n, v_j^{\perp}) = [v_j^{\perp}]^{+}\widetilde{\Psi}_i^n + [v_j^{\perp}]^{-}\widetilde{\Psi}_j^n \tag{8}$$

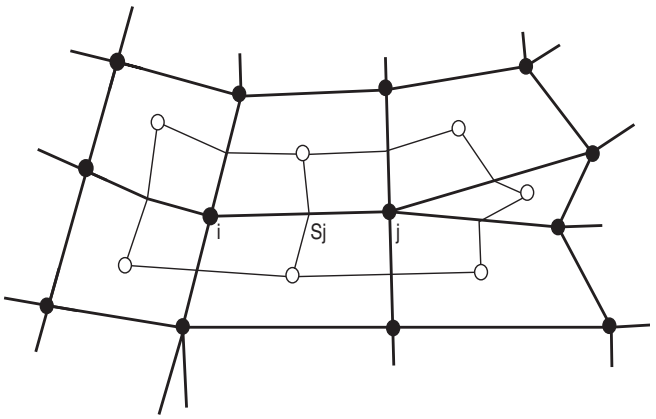

Figure 1. The edge-based median-dual approach: the edge connecting vertices i and j pierces the face S_j of the 2D computational cell surrounding vertex i.

where the nonnegative/nonpositive parts of $v_j^{\perp}$

$$[v]^+ := 0.5(v + |v|), \quad [v]^- := 0.5(v - |v|) \tag{9}$$

always coincide with outflow/inflow from/to the ith cell. With fluxes (8), the finite-volume advection of $\widetilde{\Psi}$

$$\Psi_i^* = \widetilde{\Psi}_i^n - \frac{\delta t}{\mathscr{V}_i} \sum_{j=1}^{l(i)} F_j^{\perp}(\widetilde{\Psi}^n, v_j^{\perp}) \tag{10}$$

results in the first-order-accurate sign-preserving upwind (alias donor cell) scheme.

The corrective upwind iteration employs the surface-weighted 'antidiffusive' pseudo-velocity $\hat{v}_j^{\perp}$ at the face using the field value updated in (10) and $v_j^{\perp}$ from (7)

$$\hat{v}_j^{\perp} = |v_j^{\perp}| \frac{|\Psi_j^*| - |\Psi_i^*|}{|\Psi_j^*| + |\Psi_i^*| + \varepsilon} - \frac{\delta t}{2} v_j^{\perp} \left(\mathbf{v}^{n+1/2} \cdot \frac{\nabla |\Psi^*|}{\overline{|\Psi^*|}} + \nabla \cdot \mathbf{v}^{n+1/2} \right)_{S_j} \tag{11}$$

where ε denotes a small constant, e.g. 10^{-10}, to assure that the denominator does not vanish whenever $\Psi_j^* = \Psi_i^* = 0$.[‖] The factor in brackets in the second term of the antidiffusive velocity has been written symbolically for conciseness. Its evaluation involves calculation of partial derivatives $\partial\Phi/\partial x^I$ that can be interpreted in terms of the Gauss theorem, by representing the derivative as the divergence of the augmented vector field $\Phi\nabla x^I$. Depending upon the specification of an auxiliary control volume which surrounds the edge midpoint at the face S_j, a number of approximations can be designed of various degrees of complexity and associated computational effort. Here, we take for an auxiliary control volume the sum of the control

[‖] In applications using the optional FCT enhancement, the antidiffusive velocity $\hat{v}_j^{\perp}$ is appropriately limited [9] to assure that the solution is locally bounded by the larger (in terms of the amplitude) of Ψ^* and Ψ^n.

volumes of the dual mesh surrounding vertices i and j, see Figure 1, so that

$$\left(\frac{\partial \Phi}{\partial x^I}\right)_j = \frac{1}{\overline{\mathscr{V}}_j}\left(\sum_{m=1}^{l(i)} \bar{\Phi}^{i,m} S_m^I + \sum_{m'=1}^{l(j)} \bar{\Phi}^{j,m'} S_{m'}^I\right), \quad \overline{\mathscr{V}}_j \equiv \mathscr{V}_i + \mathscr{V}_j \tag{12}$$

where, $\Phi \equiv |\Psi^*|$, $\bar{\Phi}^{i,m} \equiv 0.5(\Phi_i + \Phi_m)$, and S_m^I denotes the Ith area component of the (oriented) surface element at the mth edge. The associated $\overline{\Phi}$—the denominator in the first term in brackets on the rhs of (11)—is evaluated as a surface-area weighted average from the same auxiliary control volume

$$\overline{\Phi}_j = \frac{1}{\overline{\mathscr{S}}_j}\left(\sum_{m=1}^{l(i)} \bar{\Phi}^{i,m} |S_m^I| + \sum_{m'=1}^{l(j)} \bar{\Phi}^{j,m'} |S_{m'}^I| + \varepsilon\right) \tag{13a}$$

$$\overline{\mathscr{S}}_j \equiv \sum_{m=1}^{l(i)} |S_m^I| + \sum_{m'=1}^{l(j)} |S_{m'}^I| \tag{13b}$$

Similarly, the flow divergence appearing inside the bracket is evaluated as

$$(\nabla \cdot \mathbf{v})_j = \frac{1}{\overline{\mathscr{V}}_j}\left(\sum_{m=1}^{l(i)} v_m^{\perp} + \sum_{m'=1}^{l(j)} v_{m'}^{\perp}\right) \tag{14}$$

Having determined the antidiffusive pseudo-velocity, normal (to the cell face) corrective-upwind fluxes are evaluated as in (8), but using Ψ^* and $\hat{v}_j^{\perp}$ from (10) and (11), respectively. Updating the field by reusing the upwind scheme

$$\widetilde{\Psi}_i^{n+1} = \widetilde{\Psi}_i^* - \frac{\delta t}{\mathscr{V}_i}\sum_{j=1}^{l(i)} F_j^{\perp}(\widetilde{\Psi}^*, \hat{v}_j^{\perp}) \tag{15}$$

completes the basic MPDATA.

3. SOLVER VALIDATION

3.1. Convergence study

The overall algorithm in (5), and its particular realization in steps 1–6 of Section 2.2, warrants fully second-order-accurate solutions $\Psi_{\mathbf{i}}^{n+1}$, given a second-order accurate advection scheme $\mathscr{A}$ for an arbitrary time-independent flow $\mathbf{v}$, and an $\mathscr{O}(\Delta t^2)$ accurate estimation of the forcing $F^{\Psi}|_{\mathbf{i}}^{n+1}$. This follows the established theory of NFT schemes (cf. Reference [21] and references therein). Formal second- and third-order asymptotic convergence rates of various MPDATA options have been widely documented for the finite-difference formulation of advection problems [11, 20, 27, 28]. Recently [9], we have shown second-order accuracy of our edge-based MPDATA formulation (employed in this study) on unstructured median-dual meshes in the standard L_1, L_2 and L_∞ norms; remarkably, the second-order convergence rate held not only on quality meshes but also on skewed meshes. Here, we estimate the asymptotic convergence rate of the finite-volume edge-based NFT solver for the Euler equations, using an isentropic flow problem inspired by Nastase and Mavriplis [29].

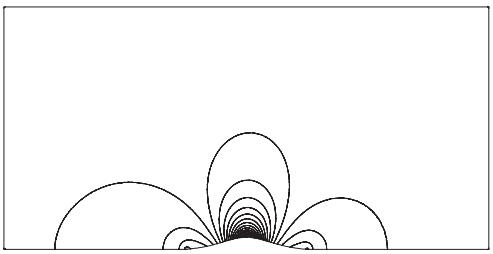

Figure 2. NFT solution (pressure contours) for a compressible flow $M = 0.5$ past a hill, using unstructured triangular mesh with $N = 280\,040$ elements.

We simulate a compressible, Mach number $M = 0.5$, channel flow past a bell-shaped hill

$$y_h(x) = h_o\{1 + [(x - x_o)/\sigma]^2\}^{-3} \tag{16a}$$

(familiar from theoretical studies of stratified orographic flows [30]) over a fixed period of time, long enough to reach steady state. The dimensionless amplitude and half-width are, respectively, $h_o = 0.1$ and $\sigma = 0.5$, and $x_o = 0$ is at the centre of the bottom wall of the $[-2, 2] \times [0, 2]$ domain. Free-slip boundary conditions are assumed at the top and bottom walls of the channel, and the characteristic conditions are specified at the inflow and outflow boundaries, cf. Reference [31]. Solutions were calculated on a sequence of ten fairly regular triangular meshes. The first mesh, consisting of $N = 501$ elements, was generated using uniform background spacing of 0.182. Each subsequent mesh was generated by decreasing the background spacing in the x and y directions by the factor $\sqrt{2}$, thus effectively doubling the number of elements. For each subsequent mesh, the time step was reduced accordingly, and the number of time steps increased correspondingly, to keep calculations stable and the simulated time fixed.

We evaluate the convergence using density norms, while approximating the exact result by the numerical solution from the finest mesh consisting of $N = 280\,040$ elements;** see Figure 2 for illustration. The subsequent-mesh solutions' departures from the 'exact' result are calculated after linearly interpolating the coarse-mesh results to the finest mesh. The standard L_2 and L_1 norms of the solutions' departures are normalized to reflect the rms and the absolute value of the truncation error per cell and the unit of time [9, 27, 28]. Table I lists the results and shows that the error reduction, with decreasing cell size and time step, approaches quadratic convergence as the resolution increases. For comparison, corresponding norms using the first-order upwind advection scheme in lieu of MPDATA are shown in parentheses. Although the second-order accuracy of NFT schemes is ensured from derivation [16, 17, 32], to our knowledge this is the first explicit illustration of the second-order asymptotic convergence for a complete NFT fluid model.

**In Reference [29] the L_2 entropy norm is analysed using a free stream entropy as a reference. When using system (1), for isentropic flows such a norm is not indicative of the truncation error, as it is always at the round-off error level.

Table I. Convergence of finite-volume MPDATA-based NFT solver for Euler equations at $M = 0.5$.

δ_E	L_2	L_1
4.47×10^{-2}	4.87×10^{-3} (5.47×10^{-3})	1.48×10^{-3} (1.97×10^{-3})
3.10×10^{-2}	3.58×10^{-3} (4.18×10^{-3})	1.19×10^{-3} (1.73×10^{-3})
2.19×10^{-2}	1.64×10^{-3} (2.83×10^{-3})	7.04×10^{-4} (1.51×10^{-3})
1.54×10^{-2}	1.01×10^{-3} (2.17×10^{-3})	4.36×10^{-4} (1.21×10^{-3})
1.07×10^{-2}	6.24×10^{-4} (1.58×10^{-3})	2.38×10^{-4} (9.01×10^{-4})
7.62×10^{-3}	3.38×10^{-4} (1.19×10^{-3})	1.25×10^{-4} (6.90×10^{-4})
5.35×10^{-3}	1.77×10^{-4} (8.57×10^{-4})	6.61×10^{-5} (5.10×10^{-4})
3.78×10^{-3}	8.50×10^{-5} (6.06×10^{-4})	2.92×10^{-5} (3.65×10^{-4})
2.67×10^{-3}	4.66×10^{-5} (4.34×10^{-4})	1.79×10^{-5} (2.64×10^{-4})

The first column lists the approximate element size $\delta_E := 1/(\sqrt{N})$. The two remaining columns provide norms of the solutions' departures from the estimated 'exact' result. The corresponding norms for the first-order-accurate UPWIND-based solver are included in parentheses.

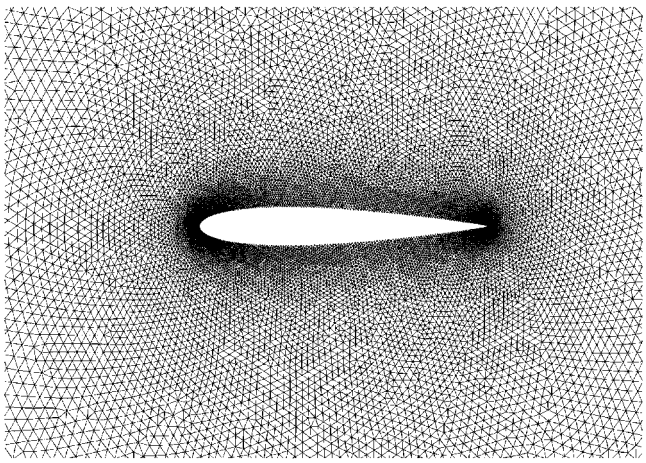

Figure 3. Computational mesh for the NACA0012 test cases; ≈ 320 points on the aerofoil.

3.2. *AGARD NACA0012 aerofoil test*

The performance of the NFT Euler solver has been studied in detail for transonic flow, AGARD test case 04 for the NACA0012 aerofoil at Mach number $M = 0.8$ and incidence angle $\alpha = 1.25°$ [33]. We generated the triangular mesh (Figure 3) consisting of 16 101 computational points, with 323 points along the aerofoil, to benchmark our solution against the solution 9 from the AGARD report, contributed by Schmidt and Jameson [33, pp. 6.21–6.23] and available in digital form. The solution 9 has been obtained by a Runge–Kutta central difference code with a blend of the second- and fourth-order artificial dissipation terms [34], while using a 320×64 structured mesh. In our calculations, in the far field placed at the distance of 20 chords, boundary conditions are specified from the free-stream values or

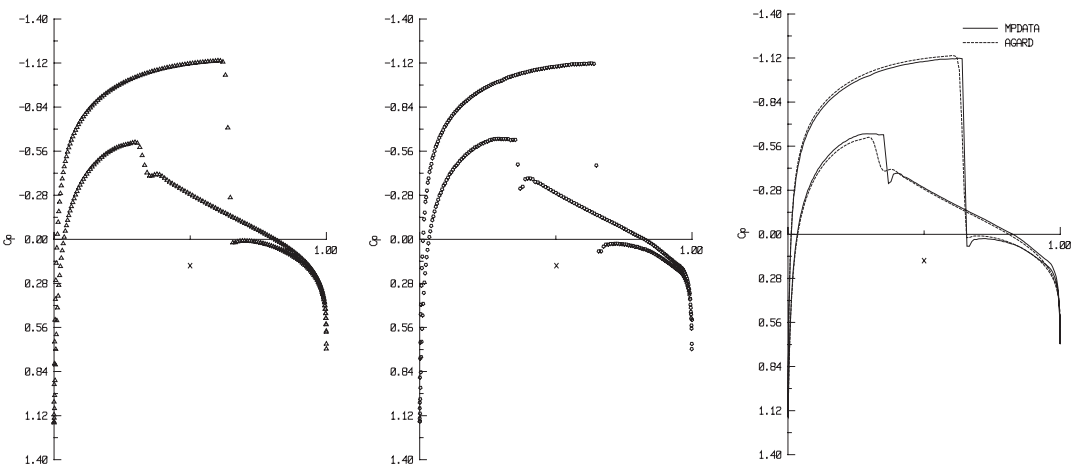

Figure 4. Surface pressure coefficient for NACA0012 aerofoil, $M = 0.8$, $\alpha = 1.25^\circ$: left, for AGARD results; centre, for MPDATA results; right, comparison; ≈ 320 points on the aerofoil.

extrapolated from the Riemann invariants, depending on the character of flow super- or subsonic, and inflow versus outflow. At the aerofoil, the free-slip condition is assumed.

The left panel in Figure 4 shows the standard display of the surface pressure coefficient C_p obtained for the AGARD reference solution 9. The corresponding MPDATA result (with a synchronized FCT enhancement, see Appendix) is shown in the central panel, and the overall comparison of the two solutions is included in the right panel. The figure demonstrates that our MPDATA solution compares favourably with the reference result. Note that in the MPDATA solution both the upper- and lower-surface shocks are captured with one point only, whereas in the AGARD result the two shocks are captured with 3 and 4 points, respectively. Furthermore, MPDATA gives much sharper pressure jump on the weaker lower-surface shock (located in the region of higher wall curvature) that, in contrast, is heavily diffused in the AGARD result. For the MPDATA solution, the drag and lift coefficient are $C_\mathrm{d} = 0.0223$ and $C_\mathrm{l} = 0.3513$, respectively; while in the reference AGARD solution $C_\mathrm{d} = 0.0230$, and $C_\mathrm{l} = 0.3632$.

In order to separate purely algorithmic issues from the discretization *per se*, in Figure 5, MPDATA is compared with the numerical scheme akin to that used in the structured-mesh AGARD reference run. Our solution labelled R–K employs three stage Runge–Kutta central-difference scheme and the Jameson artificial dissipation [34] customized for unstructured meshes. Both results in Figure 5 were obtained on the same mesh, and both codes use the same edge-based data structure. In all aspects the R–K results are overdiffusive in comparison with MPDATA. Comparing Figures 5 and 4 shows that R–K captures the weaker shock somewhat better than the AGARD solution, however, the prediction of the stronger upper-surface shock is worse than in AGARD. Clearly, for both shocks, the MPDATA predictions are the most accurate among the three solutions.

There are other algorithms designed to support single-point shock waves. Because we are unaware of published digital data generated with such schemes for the AGARD NACA0012 aerofoil test, we refer to pressure coefficients plots with available distribution of computational points, for further accuracy assessment.

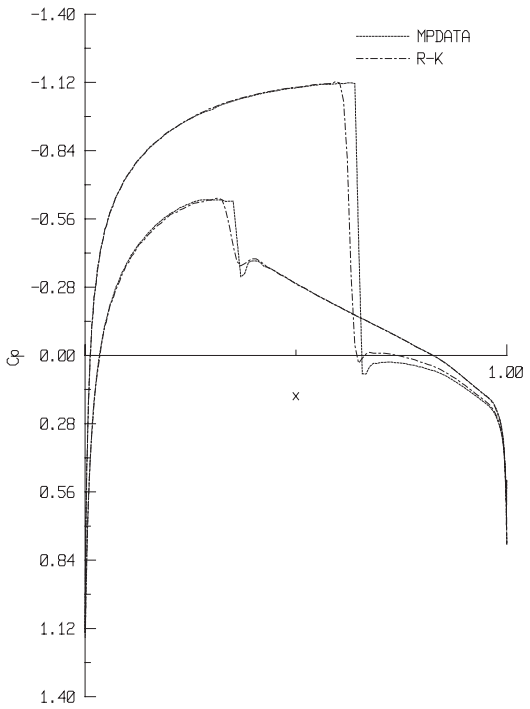

Figure 5. As in Figure 4 but comparing MPDATA with the R–K code on the unstructured mesh; ≈ 320 points on the aerofoil.

First, we consider results reported for structured meshes. The result similar to the MPDATA solution, with both strong and weaker shocks captured with one point only ($C_d = 0.0225$, $C_l = 0.3536$), is shown in Reference [35]. It was obtained on 320×64 structured 'O' mesh using Runge–Kutta solver with Block Jacobi matrix dissipation. The Reference [36] presents a result that also captures each shock with a single point ($C_d = 0.02310$, $C_l = 0.3610$). It is obtained from HCUSP solver (Modified Convective Upwind and Split Procedure, allowing constant enthalpy solutions for steady flows) using 384×64 structured 'C' mesh. In Reference [37], the solution is included for a coarser 160×32 'O' mesh with two points on the strong shock and heavily dissipated weaker shock, like in the AGARD solution. The procedure in Reference [37] is based on the reconstruction of upwind fluxes of a flux splitting using biased averaging. The reference also provides a favourable comparison of this procedure to the third-order ENO-Roe scheme for a slowly moving one-dimensional shock problem. In contrast, Figure 6 shows MPDATA solution for a triangular mesh with 162 points on the aerofoil, with a single point capture of both shock waves. By comparison with the MPDATA fine-mesh result, the lower shock is diffused at the foot and does not capture the Zierep singularity. Furthermore, the absolute values of C_p on both upper and lower surfaces are somewhat lower than in the fine-mesh result. This is particularly transparent upstream, in the vicinity of the weaker shock.

Unstructured meshes were used to solve the NACA0012 test case in References [38, 39], using Quasi-ENO, and the higher-order gradient reconstruction with multidimensional limiter

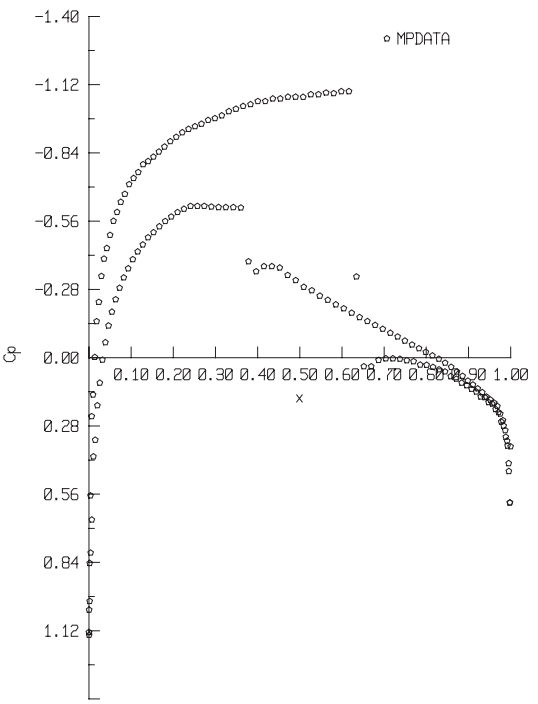

Figure 6. Surface pressure coefficient for the NACA0012 aerofoil; $M = 0.8$, $\alpha = 1.25^\circ$. MPDATA solution on a coarse unstructured mesh, 162 points on the aerofoil.

schemes, respectively. In Reference [38] the solution obtained using a triangular mesh with 320 points on the airfoil is compared and it is very similar to the AGARD data with two points needed to capture the upper shock and diffused lower shock. In Reference [39] solutions for triangular meshes equivalent to 128×32 '0' grid show between one to two points on the upper shock and diffused lower shock. When a finer mesh is used (with an unspecified number of nodes on the airfoil) both shocks are captured with one point only.

4. ADAPTIVITY

Notwithstanding a substantial portfolio of successful applications in various areas of computational physics, until recently [40] all documented MPDATA theory and implementations were carried out in the finite-difference framework. Development of unstructured mesh MPDATA flow solvers provides an opportunity for exploiting the potential of adaptive methods for new applications as well as for geospherical flows previously simulated with finite-difference MPDATA. In this section, we discuss a class of refinement indicators that naturally arise from the MPDATA approach, and demonstrate their efficacy. Furthermore, the numerical examples illustrate various mesh adaptive techniques incorporated in the NFT solver discussed in Sections 2 and 3.

4.1. Refinement indicators

The aim of adaptivity is to optimize the required mesh size to improve the accuracy of calculated results. The basic strategy is to reduce the size of the mesh locally, in regions most adversely affecting the solution accuracy, while coarsening the mesh elsewhere to improve the computational economy. For hyperbolic problems the most straightforward approach is to adapt the mesh to physical features of the flow, by employing indicators based on the gradient of a selected dependent variable. However, such a procedure does not necessarily guarantee the desired reduction of global error measures. A relatively simple alternative (formal for steady-state problems) is to employ an error estimation based on Richardson extrapolation, e.g. see Reference [12], where the estimation proceeds by differencing residuals of the solutions obtained from coarse and fine grids. Typically, the solution generated on the 'fine mesh' with halved local mesh spacing compared to the coarse mesh, is designated as an 'accurate' solution. This strategy is easiest to implement for structured meshes, but it can be ineffective in regions where the flow is discontinuous [12]. Recent efforts in devising adaptive indicators concentrate on an *a posteriori* error estimation of the user-prescribed functionals. While our preliminary investigations indicate (see REMARKS) that the adopted form (1) of the Euler equations is convenient for deriving an indicator based on objective functionals using duality arguments [13, 14], here we propose a more straightforward approach founded directly on MPDATA.

In MPDATA the first- and second-order solutions (Equations (10) and (15), respectively) as well as the leading error ($\propto \nabla\hat{v}^{\perp}\Psi$) are known explicitly. The analytic forms of the truncation errors are reviewed in Reference [21], whereas a comprehensive discussion of the exact form of the error for the edge-based formulation of MPDATA can be found in Reference [9]. Having error forms and measures available, naturally dictates refinement indicators akin to the Richardson extrapolation. There is no unique way to devise a practical indicator, and the error knowledge can be utilized in various manners. For example various normalizations and representations of the truncation error may be emphasized. Here we consider a class of indicators valid for arbitrary meshes that takes advantage of the MPDATA property that a more accurate solution is obtained from every subsequent upwind iteration.

The subsequent solution updates take the form common to (10) or (15), and can be written in brief as

$$\Psi_i^p = \Psi_i^{p-1} - \frac{\delta t}{\mathscr{V}_i}\mathrm{RHS}^{p-1}, \quad p = 1, \ldots, \mathrm{IORD} \tag{16b}$$

Here p numbers upwind iterations in MPDATA, initiated with values from the previous time step n, $\Psi^0 \equiv \Psi^n$. At the end of the upwind iterations $\Psi^{\mathrm{IORD}} = \Psi^{n+1}$. Thus, Ψ^1 is the first-order accurate upwind solution. While $(\delta t/\mathscr{V})\mathrm{RHS}^0$ represents the integral (first-order accurate) of the divergence of the convective flux, $(\delta t/\mathscr{V})\mathrm{RHS}^1$ is a first-order approximation to the leading truncation error of the solution integral. For $p \geqslant 3$ the term becomes a measure of, predominantly, the second-order error [11, 41]. In principle IORD can be arbitrary (see Reference [28] for IORD $\nearrow \infty$ limit). In practice, however, IORD > 3 is rarely used as the solution improvement becomes small and does not justify the additional effort [11, 41]. Noteworthy, for an arbitrary p and a steady flow problem, RHS acquires the sense of the residual error.

In order to capture the solution error measure, rather than its auxiliary component $\Psi + 0.5\delta t F^{\Psi}$ in (6), we set $\Psi \equiv \rho$ in our implementation (recall $F^{\rho} \equiv 0$). The basic refinement

indicator can be defined for all edges associated with ith node

$$\eta_{ij}^{p} = \frac{|\mathrm{RHS}_i^{p-1} - \mathrm{RHS}_j^{p-1}|}{\underset{i=1,\mathcal{N}}{\mathrm{Max}} |\mathrm{RHS}_i^{p-1}|}, \quad j = 1, l(i) \tag{17}$$

where $\mathcal{N}$ denotes total number of nodes. Thinking in terms of residual errors, for $p = 2$ the residual error is based on comparing the first- and second-accurate solutions, while for $p = 3$ it is constructed from the second- and (nearly) third-order solutions [11, 41]. Therefore, it appears that $p = 3$ may be a better option. We find, however, that in practice the indicator (17) acts similarly for either two or three upwind passes in MPDATA, capturing equally well both strong and weak local flow structures.†† This is consistent with observation that the phase errors of MPDATA solutions are insensitive to p [27]. In general, however, this does not preclude the utility of the $p = 3$ option. Inasmuch as the $p = 2$ indicator will tend to respond to local extrema, $p = 3$ focus on third-order derivatives, whereupon it may be a more effective for detecting saddle points in the flow, frequent precursors of shear instabilities. In the following we show the refinement indicator (17) at work.

4.2. Adaptive mesh techniques

Next, we present examples of mesh adaptivity using remeshing, mesh movement and enrichment. These techniques are in general independent of the solution algorithm and are routinely applied in unstructured mesh codes. The best choice of adaptive meshing will depend on the problem at hand. Each of these techniques has drawbacks. Remeshing, particularly in three dimensions, is expensive and introduces the complexities and inaccuracies in mapping of variables from one mesh to another. Mesh enrichment is constrained by the quality of the initial mesh and becomes excessively expensive with an increasing number of points. In some cases, this cost can be reduced by using mesh coarsening. However, an advanced point removal procedure allowing for alteration of the initial mesh may be complex, expensive, and may pose difficulties for data structures used in a flow solver. Although the mesh movement can be performed in a simple and efficient manner, it is well known that it can result in the adapted mesh having very skewed cells. In general this limits the magnitude of the movement. Sometimes a straightforward combination of any of the refinement strategies is used to avoid a particular problem. Apart from the system employing mesh enrichment described in Reference [40], we are unaware of documented mesh adaptive techniques used in combination with MPDATA. Consequently, the selection of examples that follow is intended not only as substantiation of the postulated refinement indicator (17), but also as an illustration of the mesh-adaptivity strategies, potentially beneficial for a variety of current MPDATA applications.

4.2.1. Remeshing. Throughout this paper, the advancing front [6] triangular mesh generator is used in all applications. The adaptive strategy for steady-state problems can be outlined as follows:

(a) generate an initial mesh, using the initial background mesh, ensuring quality sufficient to capture characteristic features of the flow;

†† All examples with mesh adaptivity shown in the remainder of this paper were tested for both $p = 2$ and 3.

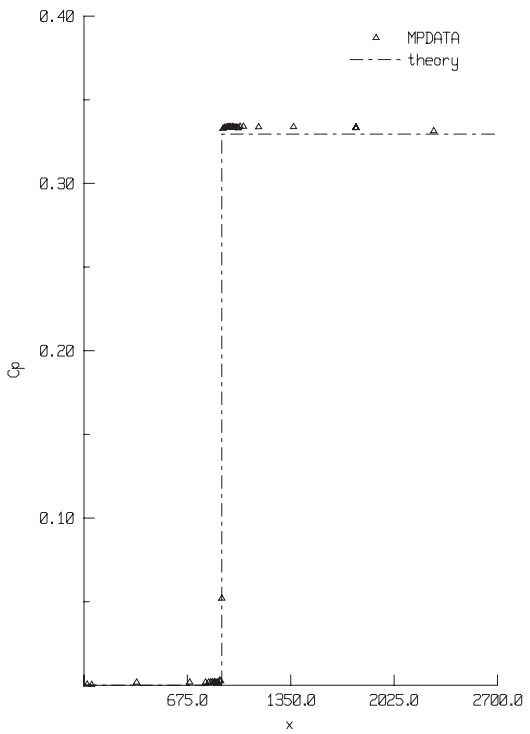

Figure 7. Comparison of theoretical and MPDATA results for 15° wedge at $M = 2.5$.

(b) obtain a solution;
(c) calculate values of the refinement indicator for every edge and use them to change spacing in the background mesh;
(d) generate new computational mesh using the updated background mesh.

For transient problems, repeat steps b–d and interpolate flow variables to the new mesh. For computational efficiency perform remeshing locally.

An example of supersonic flow over the 15° wedge using remeshing and the refinement indicator (17) with $p = 2$ is presented. Figures 7 and 8 show the pressure coefficient C_p, an adaptive mesh, and contour plots for the solution at Mach number $M = 2.5$. For this problem, $C_p = 0.329$ downstream of the shock is estimated from theoretical formulas, Section 9 in Reference [42], rounded to the nearest entry in the reference table, in Reference [42, pp. 754–757]. The corresponding computed values for all wall boundary points placed downstream of the shock $C_p \in [0.3334, 0.3339]$ are in excellent agreement. An angle β between the shock and the horizontal can be theoretically evaluated using Equation (9.22) in Reference [42] (best solved numerically), i.e.

$$\frac{\tan(\beta - \beta_o)}{\tan \beta} = \frac{2 + (\gamma - 1)M^2 \sin^2 \beta}{(\gamma + 1)M^2 \sin^2 \beta}$$

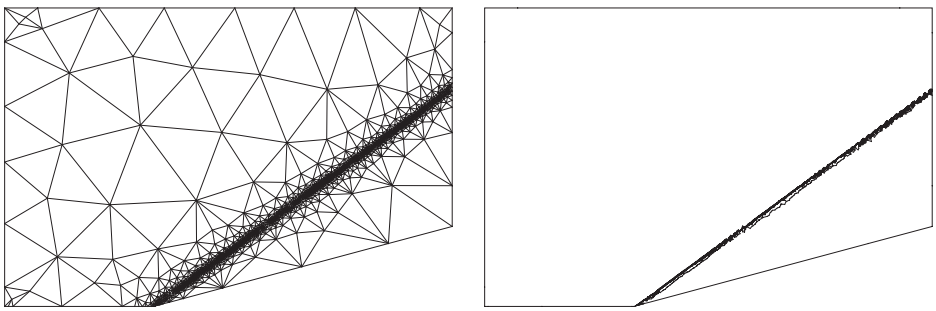

Figure 8. Remeshing (left) and pressure contours MPDATA results (right) for 15° wedge at $M = 2.5$.

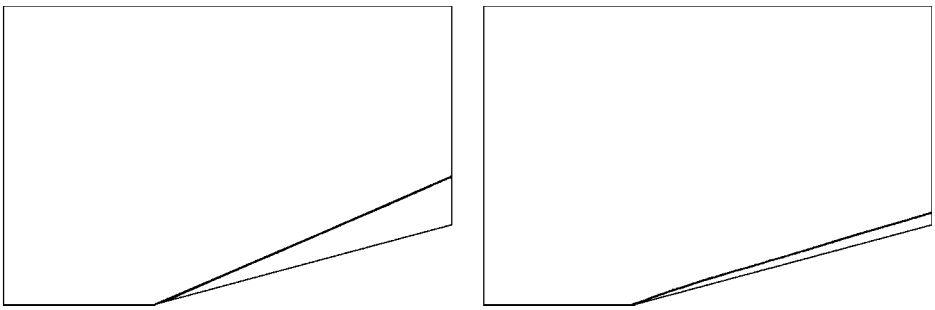

Figure 9. Pressure contours MPDATA results for 15° wedge at $M = 5$ (left) and $M = 15$ (right).

Table II. Theoretical and MPDATA results for supersonic flows past 15° wedge.

Mach number	β (theoretical)	β (computed)
2.5	36.95	36.2
5.0	24.30	23.9
15.0	18.96	18.2

where β_o is the wedge angle, $\beta_o = 15°$. Figure 9 complements Figure 8 with contour plots for further cases of Mach number $M = 5$ and $M = 15$. Table II shows, for all cases, a close match between the theoretical and numerical values of β, an excellent result.

Apart from substantiating (17), this test case highlights the extent of MPDATA applicability. It shows that MPDATA effectively resolves supersonic and hypersonic flows. Other examples presented in this paper show the efficacy of the approach for transonic flows. Note that MPDATA already has a proven record of successful applications for low-speed flows [20, 32].

Although not directly related to *adaptive* refinement, for completeness, the next example of an oscillating aerofoil demonstrates dynamic remeshing employed for an important class of problems with changing boundaries—in this case, pitching of an aerofoil. For NACA64A010 (NASA AMES Model), the theoretical profile can be found in Reference [43] and

experimental model coordinates in Reference [44]. The ordinates are tabulated in Table II of Reference [44, p. 5]; the chord length = 0.50 m; and pitching is about 0.25 chord point. The angle of incidence $\alpha(t)$ is defined by the function

$$\alpha(t) = \alpha_0 + \alpha_m \sin(\omega t)$$

where α_0 and α_m are the pitch angle and mean angle amplitude of the harmonic oscillation, respectively. The reduced frequency of the oscillation is defined for aerofoils by

$$k = \frac{\omega c}{2U_\infty}$$

where c is the aerofoil chord, ω is the angular frequency of the oscillation, and U_∞ is the free-stream velocity. For the chosen test case, $M = 0.796$, $k = 0.2002$, $\alpha_0 = 0^\circ$ (with the pivot angle fixed, the pitch angle is equivalent to the aerofoil's incidence angle), and $\alpha_m = 1.01^\circ$. The experimental data are available in Reference [45]—computational test (CT) case no. 6, dynamic index 55. The pivot point of the harmonic oscillation in pitch was fixed at $x/c = 0.248$. The nonlinear unsteady characteristics of the surface pressure are manifested in the shock movement. At the maximum angle of attack a supersonic region is present on the upper surface. As the angle of attack decreases, the flow speed gradually decreases on the upper surface, with a corresponding increase in pressure. At the same time the shock strength decreases. On the lower surface the flow speed increases, resulting in lower surface pressure and the flow pattern is reversed. Figure 10 shows the key result—a history of lift coefficient C_l versus angle of attack α. Comparison with experimental data is very good, and consistent with inviscid calculations reported in References [46, 47]. In addition, in References [46, 47] only minor improvements were shown for viscous calculations. This is no longer the case for the moment coefficient that is sensitive to boundary layer-shock interaction, and for which inviscid calculations are inadequate.

4.2.2. Mesh movement. There are several techniques that can be implemented for adaptive mesh movement, cf. Reference [48] for a recent review. A simple and general procedure [49] is based on the equidistribution principle, stating that throughout the field a multiplication of adaptive indicator and the local spacing between the nodes should be constant. The relaxation procedure applied here is to alter the position $\mathbf{x}_i = (x_i^1, x_i^2)$ of the ith node at the relaxation level $k+1$ according to:

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k - w \left(\sum_{j=1}^{l(i)} C_{ij}(\mathbf{x}_j^k - \mathbf{x}_i^k) \right) \Bigg/ \sum_{j=1}^{l(i)} C_{ij}$$

where C_{ij} is an adaptive weight function between the nodes i and j, and w is the relaxation parameter. The weight function is taken as the adaptive indicator measure, e.g. η_{ij}^p in (17). In order to improve the quality of the moved mesh, edge swapping is performed for distorted elements.

The method is demonstrated for a transonic flow over RAE2822 aerofoil at $M = 0.75$ and angle of attack $\alpha = 3$. The mesh adapted using the refinement indicator (17) with $p = 2$ is displayed in Figure 11, and the corresponding pressure contours are displayed in the left panel of Figure 12. The right panel of Figure 12 shows favourable comparison with the benchmark AGARD test case 6, solution 9 [33]. The AGARD solution was obtained with 320×64 mesh

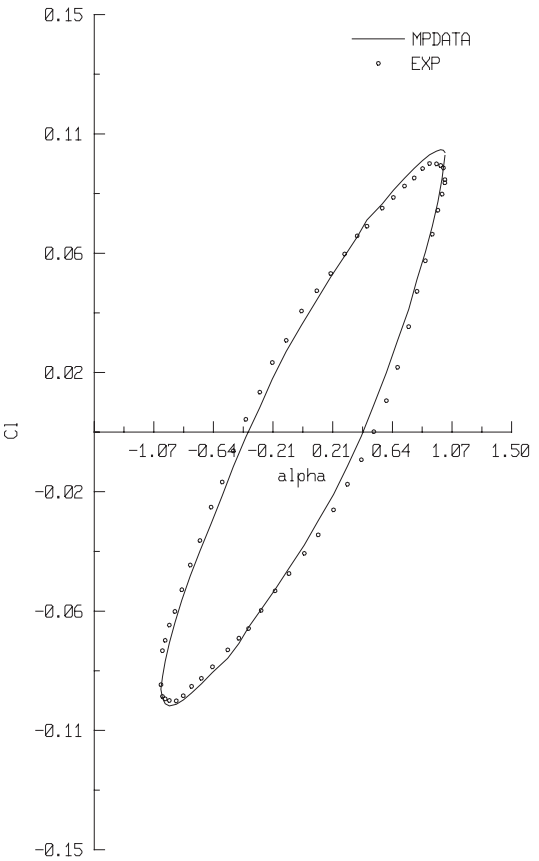

Figure 10. Hysteresis of lift coefficient as a function of the angle of attack.

points, while MPDATA solution provides very good prediction with substantially fewer points, i.e. 7523 mesh points and 205 points on the aerofoil.

4.2.3. Mesh enrichment. For a given edge, a refinement indicator is examined, and when its value exceeds the prescribed tolerance then a point is added in the centre of each of two elements associated with this edge. New edges connecting new points are generated and the data structure is updated. For distorted elements edge swapping is performed.

For demonstration, a NACA0012 aerofoil at $M = 0.8$ and $\alpha = 1.25$, described in Section 3, is considered again. An example of adapted mesh with three levels of enrichment is shown in Figure 13. Here, the refinement indicator (17) with $p = 3$ was used. Figure 14 displays the corresponding pressure contours as well as a comparison of pressure coefficient with the MPDATA solution obtained on the fine mesh analysed in Section 3. Noteworthy, both solutions are similar, however the mesh enrichment allows for accurate prediction using only 11 915 points, in contrast to 16 101 points used in the fine-mesh case. Moreover, the weaker

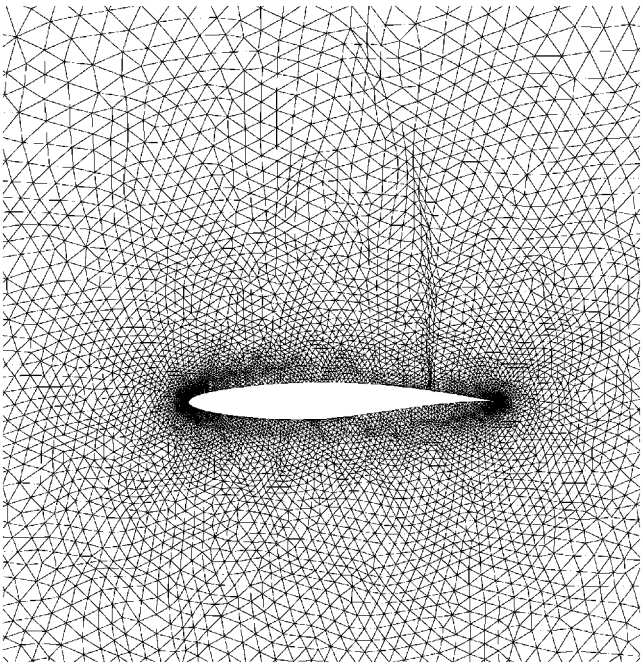

Figure 11. Mesh movement for RAE2822 $M = 0.75, \alpha = 3$.

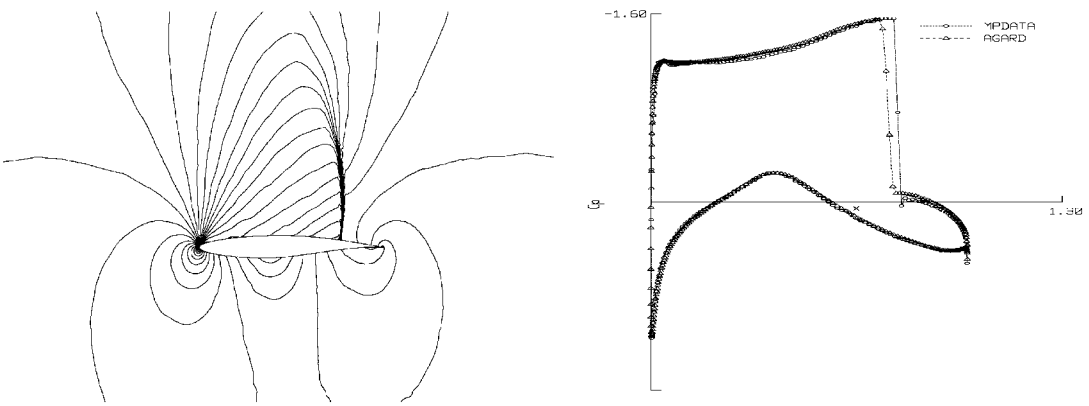

Figure 12. Pressure contours for RAE2822 $M = 0.75, \alpha = 3$ (left) and comparison of AGARD solution and MPDATA results (right) obtained on the adapted mesh shown in Figure 11.

shock is captured with zero points present on the shock. For this case, a quality solution was already obtained using the first level of enrichment (6849 points); however, the procedure was continued to demonstrate the potential of multilevel enrichment.

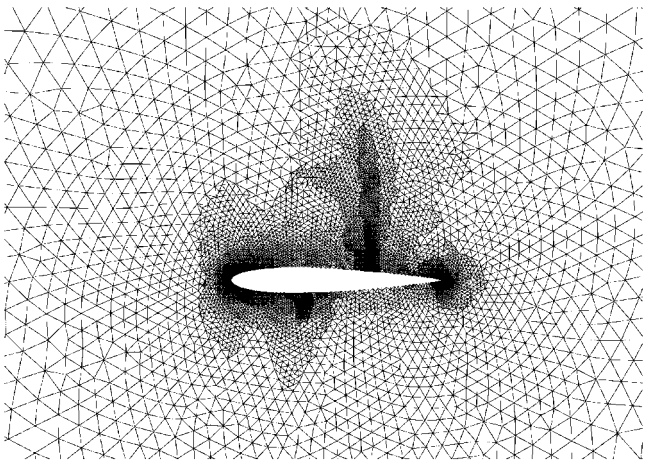

Figure 13. Enriched mesh for NACA0012 $M = 0.8, \alpha = 1.25$.

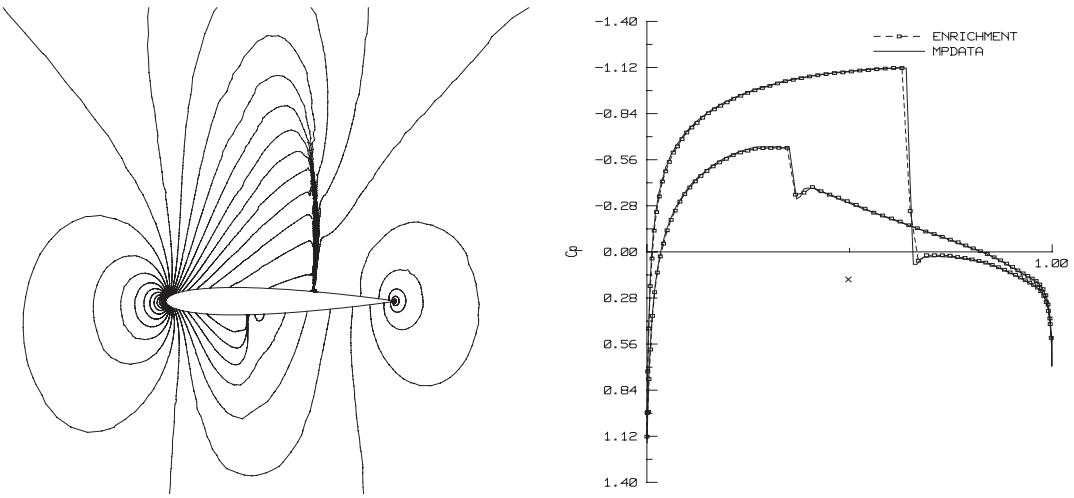

Figure 14. Pressure contours for the enriched mesh in Figure 13 (left), and comparison of MPDATA solutions obtained on fine (Figure 3) and enriched mesh (Figure 13).

5. REMARKS

We discussed properties of MPDATA that reveal advantages of the scheme for constructing refinement indicators, and documented a novel NFT edge-based solver for compressible flows. In particular, we demonstrated second-order asymptotic convergence of the solver (implied by the theory of NFT schemes) for a Mach number 0.5 flow past a smooth obstacle, and compared our MPDATA-based NFT solutions with published results for the AGARD NACA0012 aerofoil test. Further, we postulated refinement indicators arising naturally from the MPDATA

approach, and showed that the edge-based data structure enables the use of MPDATA in conjunction with adaptive meshing techniques established for unstructured meshes. The present work extends the utility of MPDATA to complex-geometry applications. The selected numerical examples, with and without adaptive techniques, show consistently that the NFT MPDATA edge-based Euler solver has a low-level implicit diffusion, and remains robust and accurate for a broad range of flow speeds. Indeed, the solutions of aeronautical problems discussed throughout the paper add another application area to the portfolio of accomplishments of MPDATA.

A significant amount of effort has been put into the development of schemes for hyperbolic conservation laws over the last decades. The choice of algorithm for a particular problem will eventually be guided by a trade-off between accuracy and computational economy. MPDATA is a family of schemes with many options. Here, we studied an unstructured-mesh, high-speed flow solver rooted in the basic second-order-accurate scheme. For unstructured meshes, the potential of more advanced versions such as the third-order-accurate algorithm, 'infinite-gauge', synchronized FCT, and a vector form still needs to be explored. Independent of the choice of the MPDATA scheme, there are several options available for designing NFT solvers, including nonlinear extrapolations of advective velocities to intermediate time levels, and alternate compensations of truncation errors due to the coupling of advection with forcings. The present NFT Euler solver is derived for a global time step only. Its efficiency can be greatly improved by adopting techniques standard in engineering applications such as local time stepping and multigrid acceleration.

The refinement indicators discussed rely on assessment of local truncation errors, readily available for MPDATA schemes. Nonetheless, our preliminary investigations indicate that the form (1) of the Euler equations considered here is also convenient for deriving an objective-functional-based indicator using duality arguments. In particular the adjoint equations will take a similar form to the flow Equations (1). The advection of adjoint variables can be solved using MPDATA NFT solver with right-hand side incorporating sources derived from an objective functional. For steady-state problems of interest to aerospace applications, when objective functional is formulated in terms of lift or drag, only one adjoint equation—related to Equation (1b)—need to be considered, since in the present form pressure is a unique function of Θ. This approach is currently under investigation.

APPENDIX

The NFT algorithm outlined in Section 2 allows for several choices of numerical realization. Some derive from the rules employed to approximate the effective trajectory integrals of the rhs in (4) [20, 32], but most depend on the options employed in the MPDATA itself. The basic scheme summarized in Section 2.3 is sign preserving (viz. nonoscillatory near zeros) and thus nonlinearly stable [20, 32]. When implemented in Step 3, it provides a meaningful solution without resorting to any explicit artificial viscosity, see solution MPDATA0 in Figure A1. In contrast, the reference scheme R–K requires explicit artificial viscosity for stability. Nonetheless this MPDATA solution is not oscillation free. The quality of the result can be greatly improved via the FCT enhancement (alluded to in the footnote in following Equation (11)); see solution MPDATA1 in Figure A1. In the latter case, the limiting of the antidiffusive velocities is still uncoupled for various transported depended variables, i.e. *asynchronous*.

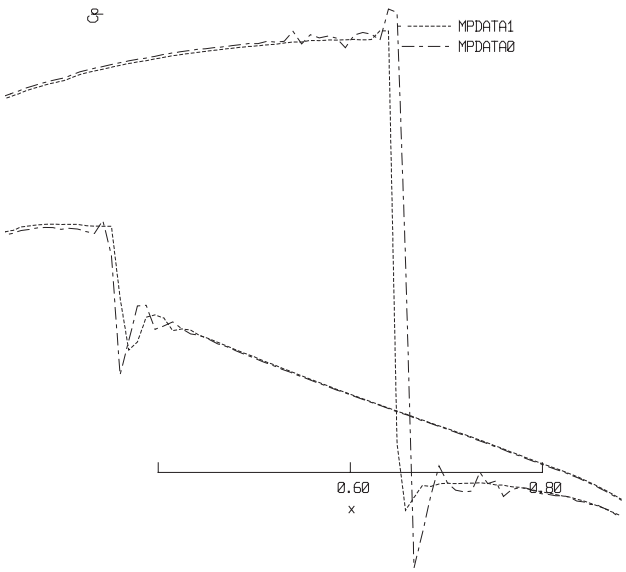

Figure A1. Surface pressure coefficient for the NACA0012 aerofoil; $M = 0.8$, $\alpha = 1.25^{o}$. A comparison of the basic (MPDATA0) and FCT-enhanced (MPDATA1) schemes.

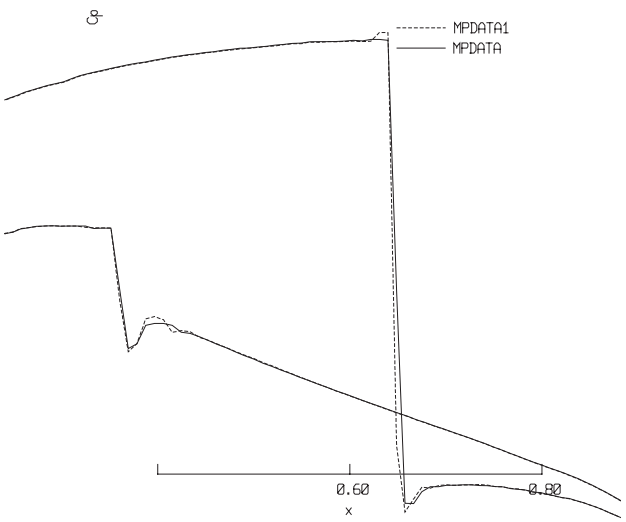

Figure A2. Surface pressure coefficient for the NACA0012 aerofoil; $M = 0.8$, $\alpha = 1.25^{o}$. A comparison of synchronized (MPDATA) and asynchronous (MPDATA1) FCT-enhanced schemes.

As a consequence, the solution MPDATA1 still evinces weak oscillations in the vicinity of steep gradients; e.g. note a spike in the pressure coefficient upstream of the upper-surface shock.

The concept of synchronized limiting has a long tradition in the literature [50, 51], including MPDATA itself [52, 53]. To assess the potential of the approach, we extended the standard (i.e. asynchronous) FCT enhancement by adopting the classical idea of the pressure-curvature-dependent switch, common in artificial dissipation models [34, 54]. The MPDATA structure is particularly convenient for controlling artificial viscosity without the necessity of constructing dissipative operators. Multiplying the antidiffusive fluxes on the rhs of Equation (15) by a switch $\mathscr{S} = 1 - \max(0, \min(1, s))$, where $s \sim p^{-1}|\partial^2 p/\partial n^2|$, assures full antidiffusive correction in regions of low-pressure gradient, while coupling the diffusion of all transported fields in the vicinity of a shock. The impact of such a shock-capturing enhancement is illustrated in Figure A2. The solution labelled MPDATA is virtually oscillation free.

ACKNOWLEDGEMENTS

PKS acknowledges partial supported from the U.S. Department of Energy 'Climate Change Prediction Program' (CCPP) research initiative.

REFERENCES

1. Peraire J, Vahdati M, Morgan K, Zienkiewicz OC. Adaptive remeshing for compressible flow computations. *Journal of Computational Physics* 1987; **72**:449–466.
2. Peraire J, Peiro J, Formaggia L, Morgan K, Zienkiewicz OC. Finite-element Euler computation in 3 dimensions. *International Journal for Numerical Methods in Engineering* 1988; **26**:2135–2159.
3. Zienkiewicz OC, Liu YC, Huang GC. Error estimation and adaptivity in flow formulation for forming problems. *International Journal for Numerical Methods in Engineering* 1988; **25**:23–42.
4. Oden JT. Progress in adaptive methods in computational fluid dynamics. In *Adaptive Methods for Partial Differential Equations*, Flaherty JE *et al.* (eds). SIAM: Philadelphia, PA, 1989; 205–252.
5. Baehmann P, Flaherty J, Guerinoni F, Ludwig R, Shephard MW. Solving compressible flow problems using adaptive finite quadtree and octree grids. In *Finite Element Analysis in Fluids*, Chueng TJ *et al.* (eds). UAH Press: New York, 1989; 87–92.
6. Löhner R, Morgan K, Zienkiewicz OC. An adaptive finite-element procedure for compressible high-speed flows. *Computer Methods in Applied Mechanics and Engineering* 1985; **51**:441–465.
7. Demkowicz I, Oden JT, Strouboilis T. An adaptive P-version finite element procedure for transient flow problems with moving boundaries. In *Finite Element in Fluids*, vol. 6, Gallagher RH *et al.* (eds). Wiley: New York, 1985; 291.
8. Smolarkiewicz PK, Szmelter J. Multidimensional positive definite advection transport algorithm (MPDATA): an edge-based unstructured-data formulation. *International Journal for Numerical Methods in Fluids* 2005; **47**:1293–1299.
9. Smolarkiewicz PK, Szmelter J. MPDATA: an edge-based unstructured-grid formulation. *Journal of Computational Physics* 2005; **206**:624–649.
10. Smolarkiewicz PK. A simple positive definite advection scheme with small implicit diffusion. *Monthly Weather Review* 1983; **111**:479–486.
11. Smolarkiewicz PK. A fully multidimensional positive definite advection transport algorithm with small implicit diffusion. *Journal of Computational Physics* 1984; **54**:325–362.
12. Berger MJ, Jameson A. Automatic adaptive grid refinement for the Euler equations. *AIAA Journal* 1985; **23**(4):561–568.
13. Pierce NA, Giles MB. Adjoint recovery of superconvergent functionals from PDE approximations. *SIAM Review* 2000; **42**:247–264.
14. Barth TJ. A posteriori error estimation and mesh adaptivity for finite volume and finite elements methods. Springer Series Lecture Notes in Computational Science and Engineering (LNCSE), 2004; **41**.
15. Szmelter J, Smolarkiewicz PK. A low-implicit-diffusion flow solver for unstructured meshes. *AIAA 2005-0321 Paper*.
16. Smolarkiewicz PK. On forward-in-time differencing for fluids. *Monthly Weather Review* 1991; **119**:2505–2510.
17. Smolarkiewicz PK, Margolin LG. On forward-in-time differencing for fluids: extension to a curvilinear framework. *Monthly Weather Review* 1993; **121**:1847–1859.
18. Smolarkiewicz PK, Grubišić V, Margolin LG. On forward-in-time differencing for fluids: stopping criteria for iterative solutions of an elastic pressure equations. *Monthly Weather Review* 1997; **125**:647–654.

19. Margolin LG, Smolarkiewicz PK, Sorbjan Z. Large-eddy simulations of convective boundary layers using nonoscillatory differencing. *Physica D* 1999; **133**:390–397.
20. Smolarkiewicz PK, Margolin LG. MPDATA: a finite-difference solver for geophysical flows. *Journal of Computational Physics* 1998; **140**:459–480.
21. Smolarkiewicz PK. Multidimensional positive definite advection transport algorithm: an overview. *International Journal for Numerical Methods in Fluids* 2005, ibidem.
22. Smolarkiewicz PK, Pudykiewicz JA. A class of semi-Lagrangian approximations for fluids. *Journal of the Atmospheric Sciences* 1992; **49**:2082–2096.
23. Swanson RC, Turkel E. On central-difference and upwind schemes. *Journal of Computational Physics* 1992; **101**:292–306.
24. Jameson A. Analysis and design of numerical schemes for gas-dynamics, 1: artificial diffusion, upwind biasing, limiters and their effect on accuracy and multigrid convergence. *International Journal of Computational Fluid Dynamics* 1995; **4**:171–218.
25. Smolarkiewicz PK, Margolin LG, Wyszogrodzki AA. A class of nonhydrostatic global models. *Journal of the Atmospheric Sciences* 2001; **58**:349–364.
26. Barth TJ. Aspects of unstructured grids and finite volume solvers for the Euler and Navier–Stokes equations. *Special Course on Unstructured Grid Methods for Advection Dominated Flows*, *AGARD. Report 787*, 1992; 6.1–6.61.
27. Smolarkiewicz PK, Grabowski WW. The multidimensional positive definite advection transport algorithm: nonoscillatory option. *Journal of Computational Physics* 1990; **86**:355–375.
28. Margolin LG, Smolarkiewicz PK. Antidiffusive velocities for multipass donor cell advection. *SIAM Journal on Scientific Computing* 1998; **20**(3):907–929.
29. Nastase CR, Mavriplis DJ. High-order discontinuous Galerkin methods using a spectral multigrid approach. *AIAA 2005-1268 Paper*.
30. Smith RB. Linear theory of stratified hydrostatic flow past an isolated mountain. *Tellus* 1980; **32**:348–364.
31. Lee S-H. Convergence characteristics of preconditioned Euler equations. *Journal of Computational Physics* 2005; **208**:266–288.
32. Smolarkiewicz PK, Prusa JM. Forward-in-time differencing for fluids: simulation of geophysical turbulence. In *Turbulent Flow Computation*, Drikakis D, Guertz BJ (eds). Kluwer Academic Publishers: Dordrecht, 2002; 279–312.
33. Test Cases for Inviscid Flow Field Methods. *AGARD Advisory Report*, *No 211 AGARD-AR-211*, 1985.
34. Jameson A, Schmidt W, Turkel E. Numerical solution of the Euler equations by finite volume methods using Runge–Kutta time-stepping schemes. *AIAA 81-1259 Paper*.
35. Pierce NA, Giles MB. Preconditioned multigrid methods for compressible flow calculations on stretched meshes. *Journal of Computational Physics* 1997; **136**:425–445.
36. Swanson RC, Radespiel R, Turkel E. On some numerical dissipation schemes. *Journal of Computational Physics* 1998; **147**:518–544.
37. Choi H, Liu JG. The reconstruction of upwind fluxes for conservation laws: its behaviour in dynamic and steady state calculations. *Journal of Computational Physics* 1998; **144**:237–256.
38. OllivierGooch CF. Quasi-ENO schemes for unstructured meshes based on unlimited data-dependent least-squares reconstruction. *Journal of Computational Physics* 1997; **133**:6–17.
39. Jawahar P, Kamath H. A high-resolution procedure for Euler and Navier–Stokes computations on unstructured grids. *Journal of Computational Physics* 2000; **164**:165–203.
40. Bacon DP *et al.* A dynamically adapting weather and dispersion model: the operational environment model with grid adaptivity (OMEGA). *Monthly Weather Review* 2000; **128**:2044–2076.
41. Margolin LG, Rider WJ, Grinstein FF. Modeling turbulent flow with implicit LES. *Journal of Turbulence* 2005, accepted.
42. Anderson JD. *Fundamentals of Aerodynamics*. McGraw-Hill Inc.: New York, 1991.
43. Abbott I, Von Doenhoff A. *Theory of Wing Sections*. Dover Publishers: New York, 1959.
44. Bland SR. AGARD Two-dimensional aeroelastic configurations. *AGARD-AR-156*, August 1979.
45. Davis SS. 82 Data set 2, NACA 64A010 (NASA AMES MODEL) oscillatory pitching. *AGARD-R-702 Advisory Report No. 702*. Compendium of Unsteady Aerodynamic Measurements, August 1982; 2-1, 2-22.
46. Chyu WJ, Davis SS, Chang KS. Calculation of unsteady transonic flow over an aerofoil. *AIAA Journal* 1981; **19**(6):684–690.
47. Liu F, Ji S. Unsteady flow calculations with a multigrid Navier–Stokes method. *AIAA Journal* 1996; **34**(10):2047–2053.
48. Yang Z, Mavriplis D. Unstructured dynamic meshes with higher-order time integration schemes for the unsteady Navier–Stokes equations. *AIAA 2005-1222 Paper*.
49. Szmelter J, Marchant MJ, Evans A, Weatherill NP. Two-dimensional Navier–Stokes equations—adaptivity on structured meshes. *Computer Methods in Applied Mechanics and Engineering* 1992; **101**:355–368.
50. Harten A, Zwas G. Self-adjusting hybrid schemes for shock computations. *Journal of Computational Physics* 1972; **9**:568–583.

51. Löhner R, Morgan K, Peraire J, Vahdati M. Finite element flux-corrected transport (FEM-FCT) for the Euler and Navier–Stokes equations. *International Journal for Numerical Methods in Fluids* 1987; **7**:1093–1109.
52. Grabowski WW, Smolarkiewicz PK. Monotone finite-difference approximations to the advection–condensation problem. *Monthly Weather Review* 1990; **118**:2082–2097.
53. Schär C, Smolarkiewicz PK. A synchronous and iterative flux-correction formalism for coupled transport equations. *Journal of Computational Physics* 1996; **128**:101–120.
54. MacCormack RW, Baldwin BS. A numerical method for solving the Navier–Stokes equations with applications to shock-boundary layer interaction. *AIAA 1975-751 Paper*.

