INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2004; **45**:1187–1207 (DOI: 10.1002/fld.727)

High-order accurate numerical solutions of incompressible flows with the artificial compressibility method

John A. Ekaterinaris*,†

Foundation for Research and Technology Hellas (FORTH), Institute of Applied and Computational Mathematics (IACM), P.O. Box 1527, 71110 Heraklion Crete, Greece

SUMMARY

A high-order accurate, finite-difference method for the numerical solution of incompressible flows is presented. This method is based on the artificial compressibility formulation of the incompressible Navier–Stokes equations. Fourth- or sixth-order accurate discretizations of the metric terms and the convective fluxes are obtained using compact, centred schemes. The viscous terms are also discretized using fourth-order accurate, centred finite differences. Implicit time marching is performed for both steady-state and time-accurate numerical solutions. High-order, spectral-type, low-pass, compact filters are used to regularize the numerical solution and remove spurious modes arising from unresolved scales, non-linearities, and inaccuracies in the application of boundary conditions. The accuracy and efficiency of the proposed method is demonstrated for test problems. Copyright © 2004 John Wiley & Sons, Ltd.

KEY WORDS: numerical solutions; incompressible flow; high order methods; implicit schemes

1. INTRODUCTION

In many practical applications, such as flows over hydrofoils, wind-turbine blades, and aircraft wings during takeoff and landing the performance is dramatically affected by essentially incompressible separated flow. Furthermore, the development of affordable, high-speed trains and naval transportation with potential to commercial applications needs detailed flowfield information because the harsh environment of such endeavors imposes daunting mechanical, structural, and propulsive loads. For these applications, the flow is also incompressible, however, the difficulty of full-scale testing constrains efforts to evolve new concepts. Therefore application of advanced, efficient, and accurate methods for the numerical solution of the incompressible flow equations is required.

The incompressible-flow equations include pressure in a non-time-dependent form since the continuity equation has a non-evolutionary character. As a result, difficulties arise with the

*Correspondence to: John A. Ekaterinaris, Foundation for Research and Technology Hellas (FORTH), Institute of Applied and Computational Mathematics (IACM), P.O. Box 1527, 71110 Heraklion Crete, Greece.

†E-mail: ekaterin@iacm.forth.gr

Published online 14 May 2004
Received 20 October 2003
Revised 7 March 2004

numerical solution of incompressible-flow equations in coupling changes of the computed velocity field with changes in the pressure field while satisfying the continuity equation. The stream function/vorticity formulation, which is straightforward to apply for two-dimensional flows, does not present the same difficulties. Numerical solutions based on the primitive variable formulation are obtained either with fractional time-step methods [1–3] or by solving a Poisson equation for pressure [4]. High-order accurate numerical solutions of incompressible flows using primitive variable formulations were obtained in the past using fractional-step techniques either with finite-difference discretizations [5] or with spectral element discretizations [6, 7].

The present paper is focused on addressing the crucial issue of accurate and efficient incompressible flow computation by developing, evaluating, and demonstrating the efficiency of a new high-order algorithm for incompressible flows. The numerical solution of incompressible flows is based on the artificial compressibility or pseudocompressibility formulation of the incompressible Navier–Stokes equations [8]. The numerical algorithm is based on a high-order accurate in space, centred, finite-difference method. The non-linear convective fluxes are evaluated using fourth- or sixth-order accurate compact, central difference schemes. Steady simple flows may be computed with explicit schemes. For flows in complex domains and time-dependent problems, the numerical solution is advanced in time using implicit dual-time-stepping schemes. In order to eliminate spurious modes arising from aliasing errors, non-linear interactions, and inaccuracies from the application of boundary conditions, spectral-type, low-pass filtering is used for the computed flow variables.

The rest of the paper is organized as follows: The governing equations are summarized. The pseudocompressibility formulation and the time marching schemes are presented. Compact centred schemes used in combination with spectral-type filters are outlined. Finally, in the results section the accuracy and efficiency of the proposed method is demonstrated for test problems and flows of practical interest.

2. GOVERNING EQUATIONS

For finite-difference and finite-volume structured-grid techniques, the numerical treatment of flows in complex geometries is facilitated with the use of generalized non-orthogonal co-ordinate transformations. These transformations make possible the numerical solution with body fitted stretched meshes which greatly simplify the specification of the boundary conditions. The transformation of the Navier–Stokes equations from Cartesian (x, y, z) co-ordinates to generalized non-orthogonal co-ordinates (ξ, η, ζ) is

$$[\mathbf{D}]\frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial \mathbf{F}}{\partial \xi} + \frac{\partial \mathbf{G}}{\partial \eta} + \frac{\partial \mathbf{H}}{\partial \zeta} = \frac{1}{Re}\left(\frac{\partial \mathbf{F}_v}{\partial \xi} + \frac{\partial \mathbf{G}_v}{\partial \eta} + \frac{\partial \mathbf{H}_v}{\partial \zeta}\right) \tag{1}$$

where $\mathbf{D} = \mathrm{diag}[0, 1, 1, 1]$, $\mathbf{Q} = J\mathbf{q}$, J is the Jacobian of the transformation $J = \partial(x, y, z)/\partial(\xi, \eta, \zeta)$, $\mathbf{q} = [p, u, v, w]^{\mathrm{T}}$ is the unknown vector for the pressure, p, and u, v, w are the Cartesian velocity components. Furthermore, $\mathbf{F}$, $\mathbf{G}$, $\mathbf{H}$ are the inviscid flux vectors, Re is the Reynolds number, $Re = U_\infty L/\nu$, based on a characteristic length L and a characteristic velocity U_∞, and

$\mathbf{F}_v$, $\mathbf{G}_v$, $\mathbf{H}_v$ are the viscous fluxes given by

$$\mathbf{F} = (\mathbf{f}_\xi \xi_x + \mathbf{g}_\xi \xi_y + \mathbf{h}_\xi \xi_z)/J, \quad \mathbf{F}_v = (\mathbf{f}_{v_\xi} \xi_x + \mathbf{g}_{v_\xi} \xi_y + \mathbf{h}_{v_\xi} \xi_z)/J$$
$$\mathbf{G} = (\mathbf{f}_\eta \eta_x + \mathbf{g}_\eta \eta_y + \mathbf{h}_\eta \eta_z)/J, \quad \mathbf{G}_v = (\mathbf{f}_{v_\eta} \eta_x + \mathbf{g}_{v_\eta} \eta_y + \mathbf{h}_{v_\eta} \eta_z)/J$$
$$\mathbf{H} = (\mathbf{f}_\zeta \zeta_x + \mathbf{g}_\zeta \zeta_y + \mathbf{h}_\zeta \zeta_z)/J, \quad \mathbf{H}_v = (\mathbf{f}_{v_\zeta} \zeta_x + \mathbf{g}_{v_\zeta} \zeta_y + \mathbf{h}_{v_\zeta} \zeta_z)/J$$

where ξ_x, η_x, ζ_x etc. are the metrics and J the Jacobian of the geometrical transformation, and $\mathbf{f}$, $\mathbf{g}$, $\mathbf{h}$ are the convective fluxes in Cartesian co-ordinates,

$$\mathbf{f} = [u, u^2 + p, uv, uw]^{\mathrm{T}}, \quad \mathbf{g} = [v, vu, v^2 + p, vw]^{\mathrm{T}}, \quad \mathbf{h} = [w, wu, wv, w^2 + p]^{\mathrm{T}}$$

and $\mathbf{f}_v$, $\mathbf{g}_v$, $\mathbf{h}_v$ are the viscous flux terms in Cartesian co-ordinates

$$\mathbf{f}_v = \left[\frac{\partial u}{\partial x}, \frac{\partial v}{\partial x}, \frac{\partial w}{\partial x}\right]^{\mathrm{T}}, \quad \mathbf{g}_v = \left[\frac{\partial u}{\partial y}, \frac{\partial v}{\partial y}, \frac{\partial w}{\partial y}\right]^{\mathrm{T}}, \quad \mathbf{h}_v = \left[\frac{\partial u}{\partial z}, \frac{\partial v}{\partial z}, \frac{\partial w}{\partial z}\right]^{\mathrm{T}}$$

Using these definitions the non-linear inviscid terms $\mathbf{F}$, $\mathbf{G}$, $\mathbf{H}$ can be expressed as

$$\mathbf{F} = \frac{1}{J}\begin{bmatrix} 0 \\ \xi_x p + uU \\ \xi_y p + vU \\ \xi_z p + wU \end{bmatrix}, \quad \mathbf{G} = \frac{1}{J}\begin{bmatrix} 0 \\ \eta_x p + uV \\ \eta_y p + vV \\ \eta_z p + wV \end{bmatrix}, \quad \mathbf{H} = \frac{1}{J}\begin{bmatrix} 0 \\ \zeta_x p + uW \\ \zeta_y p + vW \\ \zeta_z p + wW \end{bmatrix}$$

where $U = \xi_x u + \xi_y v + \xi_z w$, $V = \eta_x u + \eta_y v + \eta_z w$, and $W = \zeta_x u + \zeta_y v + \zeta_z w$ are the contravariant velocity components. Analogous expressions are obtained for the viscous terms.

2.1. Pseudocompressibility formulation

The primary problem with both steady-state and time-accurate solutions of the incompressible flow equations, Equation (1), is the difficulty of coupling changes in the velocity field with changes in the pressure field while satisfying the continuity equation. The artificial compressibility or pseudocompressibility method is often used to overcome these difficulties. This method was initially introduced by Chorin [8] for the solution of steady-state incompressible flows, and it was subsequently extended [9] to time-accurate numerical solutions of incompressible flows. The artificial compressibility formulation can be utilized for the solution of incompressible flows when a pseudo-time derivative of pressure is added to the continuity equation. This term directly couples the pressure with velocity and allows the equations to advance in time by iterating until a divergence-free velocity field is obtained at the new time level. For completeness, the pseudocompressibility formulation and the technique used for numerical solution are described next.

The artificial compressibility or pseudocompressibility formulation is obtained from the original incompressible flow equations, Equation (1), by introducing an additional time derivative of pressure to the continuity equation as

$$\frac{\partial p}{\partial \tau} = -\beta \nabla \cdot \mathbf{U} = -\beta \left[\frac{\partial}{\partial \xi}\left(\frac{U}{J}\right) + \frac{\partial}{\partial \eta}\left(\frac{V}{J}\right) + \frac{\partial}{\partial \zeta}\left(\frac{W}{J}\right)\right] \tag{2}$$

Addition of this fictitious pressure derivative enables full coupling of the continuity with the momentum equations and significantly facilitates the numerical solution. In Equation (2), τ does not represent physical time; therefore in the momentum equation t is replaced by τ, and the pseudocompressible form of the governing equations is

$$\frac{\partial \mathbf{Q}}{\partial \tau} + \frac{\partial \mathbf{F}}{\partial \xi} + \frac{\partial \mathbf{G}}{\partial \eta} + \frac{\partial \mathbf{H}}{\partial \zeta} = \frac{1}{Re}\left(\frac{\partial \mathbf{F}_v}{\partial \xi} + \frac{\partial \mathbf{G}_v}{\partial \eta} + \frac{\partial \mathbf{H}_v}{\partial \zeta}\right) \tag{3}$$

In Equation (3), the matrix $[D]$ becomes the identity matrix and as in Equation (1) $\mathbf{Q} = [p, u, v, w]^{\mathrm{T}}/J$ is the solution variable vector and $\mathbf{F}$, $\mathbf{G}$, $\mathbf{H}$ and $\mathbf{F}_v$, $\mathbf{G}_v$, $\mathbf{H}_v$ are the same inviscid and viscous flux vectors, respectively.

In these equations, τ is referred to as pseudotime which can be considered as a time iteration parameter. Steady-state incompressible solutions are obtained with the artificial compressibility method by time marching as in the compressible flow case. The numerical methods for the solution of the pseudocompressible equations are very similar to the methods used for the solutions of the compressible flow equations [10]. At convergence, however, the time derivative of pressure, and consequently the divergence of the velocity, approach zero. The parameter β, which is referred to as the artificial compressibility or pseudocompressibility parameter, usually takes a value between 1 and 10, but larger values may be required for solutions on highly stretched grids. Time-accurate solutions of unsteady flows with the pseudocompressibility formulation with implicit schemes are obtained using dual time-stepping [11, 12] as shown in the next section.

2.2. Time marching schemes

Time marching of incompressible flows with the pseudocompressibility formulation becomes very similar to time-marching used for the compressible flow calculations. For time-accurate numerical solutions, the additional constraint of incompressibility must be satisfied. Therefore, dual-time stepping schemes introduced in Reference [9] must be combined with fast solvers such as multigrid and implicit methods.

2.2.1. Explicit methods. Time advancement to a steady-state of inviscid flows or viscous flows in simple domains with isotropic meshes may be obtained with the third- [13] or fourth-order [14] explicit Runge–Kutta methods. For example, the following third-order accurate Runge–Kutta method [13] can be used:

$$\begin{aligned} Q^{(1)} &= Q^n + \Delta t R(Q^n) \\ Q^{(2)} &= \tfrac{3}{4} Q^n + \tfrac{1}{4} Q^{(1)} + \tfrac{1}{4} \Delta t R(Q^{(1)}) \\ Q^{n+1} &= \tfrac{1}{3} Q^n + \tfrac{2}{3} Q^{(2)} + \tfrac{2}{3} \Delta t R(Q^{(2)}) \end{aligned} \tag{4}$$

where R represents the right-hand side terms $R(Q) = (F - F_v)_\xi + (G - G_v)_\eta + (H - H_v)_\zeta$. For time accurate solutions with explicit time-marching methods, an iterative, dual-time-stepping scheme with or without multigrid acceleration must be used [15]. Numerical solutions of incompressible flows with the pseudocompressibility formulations in irregular domains with highly stretched meshes can be obtained using multigrid acceleration with explicit time marching methods [16].

2.2.2. Implicit methods. Steady-state and time accurate solutions of complex flows can be obtained by using the following second-order accurate, three-time level formula to evaluate the time derivatives in the momentum equations:

$$\frac{3\boldsymbol{u}^{(n+1)} - 4\boldsymbol{u}^{(n)} + \boldsymbol{u}^{(n-1)}}{2\Delta t} = -R^{(n+1)} \tag{5}$$

where R represents the right-hand side residual term $R^{(n+1)} = (F - F_v)_\xi^{(n+1)} + (G - G_v)_\eta^{(n+1)} + (H - H_v)_\zeta^{(n+1)}$.

The discretized Equations (5) are solved for a divergence free velocity field at the $(n+1)$ time level by introducing a pseudotime level which is denoted by the superscript (m) in the following artificial compressibility relation:

$$\frac{\partial p}{\partial \tau} = -\beta \nabla \cdot \boldsymbol{u}^{(n+1,m+1)} \tag{6}$$

Iterative solution of this equation is performed so that the velocity vector $\boldsymbol{u}^{(n+1,m+1)}$ approaches $\boldsymbol{u}^{(n+1)}$ as the divergence $\nabla \cdot \boldsymbol{u}^{(n+1,m+1)}$ approaches zero.

The delta form of the linearized, unfactored, implicit algorithm, for both steady-state and time-accurate solutions is given by

$$\begin{aligned}\left[\frac{\mathbf{I}_{\mathrm{tr}}}{J} + \left(\frac{\partial R}{\partial Q}\right)^{(n+1,m)}\right] &\times (Q^{(n+1,m+1)} - Q^{(n+1,m)}) \\ &= -R^{(n+1,m)} - \frac{\mathbf{I}_{\mathrm{m}}}{\Delta t}(1.5Q^{(n+1,m)} - 2Q^{(n)} + 0.5Q^{(n-1)}) = \hat{R}\end{aligned} \tag{7}$$

where $\mathbf{I}_{\mathrm{tr}} = \mathrm{diag}[(\Delta t/\Delta\tau), 1.5, 1.5, 1.5]/\Delta t$ and $\mathbf{I}_{\mathrm{m}} = \mathrm{diag}[0, 1, 1, 1]$. The algorithm for steady-state solutions is obtained from Equation (7) when the internal iteration index (m) is dropped and only the first term, $R^{(n+1)}$, is retained on the right-hand side.

Exact evaluation of the residual term Jacobian, $\mathscr{J} = \partial R/\partial Q$, in the right-hand side of Equation (7) results in a banded matrix. Each entry of this banded matrix represents a vector of 4×4 blocks which is aligned along a diagonal of the matrix. Exact evaluation of the Jacobian $\mathscr{J}$ is very costly computationally, therefore, an approximate Jacobian of the flux differences is used. The resulting matrix system of linear equations is solved using a Gauss–Seidel-relaxation method [11]. The implementation of the solution procedure is as follows. First, the entire matrix in Equation (7) is formed and stored as the following banded matrix system:

$$\mathscr{B}[U_{\mathrm{L}}, 0, \ldots, 0, V_{\mathrm{L}}, 0, \ldots, 0, D_{\mathrm{L}}, D, D_{\mathrm{U}}, 0, \ldots, 0, V_{\mathrm{U}}, 0, \ldots, 0, U_{\mathrm{U}}]\Delta Q = \hat{R} \tag{8}$$

where $\Delta Q = Q^{(n+1,m+1)} - Q^{(n+1,m)}$ and $U_{\mathrm{L}}, V_{\mathrm{L}}, D_{\mathrm{L}}, D, D_{\mathrm{U}}, V_{\mathrm{U}}, U_{\mathrm{U}}$ are vectors of 4×4 blocks that lie on the diagonals of matrix $\mathscr{B}$, with the D vector on the main diagonal. This matrix equation is solved using an iterative approach where only one family of lines is used as the sweep direction. The resulting tridiagonal band matrix system is solved by performing and storing the lower–upper (LU) decomposition.

The left-hand side operator $[(\mathbf{I}_{\mathrm{tr}}/J) + (\partial R/\partial Q)^{(n+1,m)}]$ of Equation (7) can be approximately factorized to obtain the Beam–Warming algorithm as follows:

$$\begin{aligned}&\left[\frac{\mathbf{I}_{\mathrm{tr}}}{J} + \left(\frac{2}{3}\right)\delta_{\xi}\left(\frac{\partial \mathbf{F}}{\partial Q}\right)\right]^{m}\\ &\quad\times\left[\frac{\mathbf{I}_{\mathrm{tr}}}{J} + \left(\frac{2}{3}\right)\delta_{\eta}\left(\frac{\partial \mathbf{G}}{\partial Q}\right)\right]^{m}\\ &\quad\times\left[\frac{\mathbf{I}_{\mathrm{tr}}}{J} + \left(\frac{2}{3}\right)\delta_{\zeta}\left(\frac{\partial \mathbf{H}}{\partial Q}\right)\right]^{m}\times(Q^{(n+1,m+1)} - Q^{(n+1,m)})\\ &\quad= -R^{(n+1,m)} - \frac{\mathbf{I}_{\mathrm{m}}}{\Delta t}(1.5Q^{(n+1,m)} - 2Q^{(n)} + 0.5Q^{(n-1)}) = \hat{R}\end{aligned} \tag{9}$$

Among the time stepping methods of Equations (4), (7), and (9) the unfactored algorithm of Equation (7) yielded the best performance for both steady-state and time-accurate numerical solutions.

The numerical solution with the delta form of the implicit time integration algorithms, Equations (7) and (9), essentially operates as $[\mathrm{NUMERICS}]\Delta Q = \mathrm{PHYSICS}$. The term NUMERICS denotes the unfactored or factorized left-hand side implicit operators of Equations (7) and (9), while PHYSICS constitutes the evaluation of the residual term $\hat{R}$. Spatial differencing in the latter is performed either with fourth- or sixth-order accurate compact, centred schemes that are summarized in the next section.

3. HIGH-ORDER SPACE DISCRETIZATION

Compact, centred high-order accurate schemes are used to evaluate the derivatives of metric terms and the convective fluxes in the finite-difference framework. The spatial derivative $\mathscr{F}'$ of any scalar quantity, $\mathscr{F}$, such as flow variable, metric, or flux component in the equally spaced transformed plane (ξ, η, ζ) is obtained by solving the following tridiagonal system:

$$a\mathscr{F}'_{j-1} + \mathscr{F}'_{j} + a\mathscr{F}'_{j+1} = A\,\frac{\mathscr{F}_{j+1} - \mathscr{F}_{j-1}}{4\Delta\xi} + B\,\frac{\mathscr{F}_{j+2} - \mathscr{F}_{j-2}}{4\Delta\xi} \tag{10}$$

where the constants A, B, and a determine the spatial order of accuracy of the scheme. The formula of Equation (10) encompasses a family of schemes ranging in accuracy from the standard three-point, second-order accurate explicit method (E2) to the compact five-point, sixth-order accurate compact scheme (C6). The values of the coefficients a, A, B in Equation (10) for various schemes are given in Table I.

An extensive discussion of the resolving ability of these schemes in the wave domain can be found in References [17, 18]. For completeness, the resolving ability of the second-order accurate scheme (E2), the explicit fourth-order accurate scheme (E4), and the fourth- and sixth-order accurate compact schemes (C4) and (C6), respectively, is shown in Figure 1. The improvement in performance with compact schemes, demonstrated in Figure 1, is obtained at a small increase in the computing cost needed for tridiagonal matrix inversion. High-resolution computational methods based on compact schemes impacted other applications, such as direct

Table I. Explicit and compact schemes of Equation (10).

Scheme	a	A	B	Order
E2	0	1	0	2
E4	0	$\frac{4}{3}$	$-\frac{1}{3}$	4
C4	$\frac{1}{4}$	$\frac{3}{2}$	0	4
C6	$\frac{1}{3}$	$\frac{14}{9}$	$\frac{1}{9}$	6

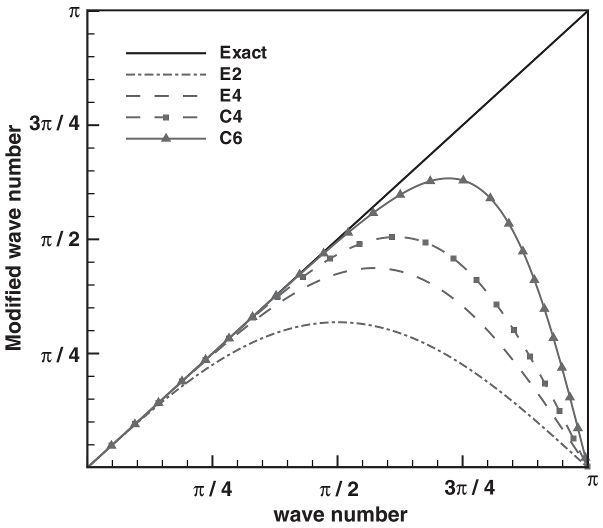

Figure 1. Wave space resolution of compact and explicit central-difference.

numerical simulations (DNS) [19], large eddy simulations (LES) [20] of compressible flow, and non-linear computational aeroacoustics (CAA).

For periodic domains, a periodic tridiagonal solver is used with Equation (10). For non-periodic domains, one-sided compact formulas at the edges of the domain are used. For example, the fourth-order accurate scheme at $i = 1$ evaluates the derivatives as

$$6(\mathscr{F}')_1 + 8(\mathscr{F}')_2 = (-17\mathscr{F}_1 + 9\mathscr{F}_2 + 9\mathscr{F}_3 - \mathscr{F}_4)/(\Delta\xi) \tag{11}$$

These stable boundary schemes for the evaluation of the derivatives with Equation (11) in arbitrary domains have been developed and analysed in Reference [21].

3.1. Viscous term evaluation

Evaluation of the inviscid flux terms with low-resolution spatial discretizations is the most important source of error. However, high-resolution computations require accurate evaluation of the viscous terms. Modified wavenumber analysis, similar to the analysis presented in Figure 1, shows that the second-order central difference approximation to a second derivative, ∂_{xx}, is almost twice as accurate as a second-order central difference approximation of the first derivative ∂_x. Therefore, many CFD codes [11], combine a third- or higher-order, upwind-biased spatial discretization of the inviscid fluxes with a second-order accurate central-difference discretization of the second derivative for the viscous flux terms. Computations of realistic high Reynolds number turbulent flows require low numerical errors in the boundary layer for the accurate prediction of drag. This is because the drag is affected by the balance between the inviscid and viscous flux terms. Furthermore, for separated and vortex dominated flows as well as for detached eddy simulation (DES) applications the accuracy for the viscous terms may have a significant influence on the fidelity of the computed results.

Recently, high-order compact schemes were developed for the simultaneous calculation of the first and second derivative [22]. However, for turbulent flow calculations with the Reynolds averaged Navier–Stokes (RANS) equations where the eddy viscosity is not constant the viscous terms have the following general form:

$$\partial_\zeta(v_{\text{tur}}\partial_\zeta u) = \partial_\zeta[a_{i,j,k}\partial_\zeta b_{i,j,k}] \tag{12}$$

Evaluation of the derivatives in Equation (12) with repeated application of compact schemes is possible. However, this procedure is very costly computationally. Evaluation of the derivatives in Equation (12) with explicit fourth-order accurate schemes without dropping the accuracy at the regions near the boundaries of the domain is not possible. Therefore, half point, fourth-order accurate explicit formulas are used for the evaluation of the inner term $a_{i,j,k}\partial_\zeta b_{i,j,k}$ in Equation (12). The derivative $\partial_\zeta b_{i,j,k}$ of the inner term is computed at half-node points, $k+\frac{1}{2}$, using available nodal values at k points. The value $a_{i,j,k+1/2}$ is obtained using a fourth-order accurate interpolation formula. Finally, the entire term of Equation (12) is evaluated at integer nodal points k using the half-node points values of the term in square brackets $[a_{i,j,k+1/2}(b_\zeta)_{i,j,k+1/2}]$. In this formulation, the metric terms are also evaluated at half-node points either by averaging, for example $(\xi_x)_{i,j,k} = [(\xi_x)_{i,j,k} + (\xi_x)_{i,j,k-1}]/2$, or by recomputing the metric at half points.

4. SPECTRAL-TYPE LOW-PASS FILTERS

Centred compact discretizations with Equation (10) are non-dissipative and therefore susceptible to numerical instabilities resulting from the growth of high-frequency spurious modes. These modes originate from non-linear flow features, mesh non-uniformities, and application of numerical boundary conditions. For well resolved, weakly non-linear flow problems straightforward application of compact schemes is possible. However, numerical solutions of practical complex flow problems in non-trivial geometries or solutions of flow problems with strong non-linearities eventually develop spurious modes that destroy the solution. In order to make the high-order accurate centred discretizations suitable to practical applications, while retaining the improved accuracy of high-order centred schemes, implicit high-order, spectral-type, low-pass filters are used. Spectral-type filters are simple and they are applied as a postprocessing stage after each time step integration to each component p, u, v, w of the solution vector Q separately. If a component of the solution vector $Q = (p,u,v,w)$ is denoted by $\mathscr{Q}$ the filtered values $\hat{\mathscr{Q}}$ are obtained by

$$a_{\text{f}}\hat{\mathscr{Q}}_{j-1} + \hat{\mathscr{Q}}_j + a_{\text{f}}\hat{\mathscr{Q}}_{j+1} = \sum_{n=0}^{N} \frac{a_n}{2}(\mathscr{Q}_{j+n} + \mathscr{Q}_{j+n}) \tag{13}$$

Table II. Coefficient of the explicit filters of Equation (13).

Filter	a_0	a_1	a_2	a_3	a_4	a_5
F2	$\frac{(1+2a_{\rm f})}{2}$	$\frac{(1+2a_{\rm f})}{2}$	0	0	0	0
F4	$\frac{(5+6a_{\rm f})}{8}$	$\frac{(1+1a_{\rm f})}{2}$	$\frac{(-1+2a_{\rm f})}{8}$	0	0	0
F6	$\frac{(11+10a_{\rm f})}{16}$	$\frac{(15+34a_{\rm f})}{32}$	$\frac{(-3+6a_{\rm f})}{16}$	$\frac{(1+2a_{\rm f})}{32}$	0	0
F8	$\frac{(93+70a_{\rm f})}{128}$	$\frac{(7+18a_{\rm f})}{16}$	$\frac{(-7+14a_{\rm f})}{32}$	$\frac{(-7+14a_{\rm f})}{16}$	$\frac{(-1+2a_{\rm f})}{128}$	0
F10	$\frac{(193+126a_{\rm f})}{256}$	$\frac{(105+302a_{\rm f})}{256}$	$\frac{(-15+30a_{\rm f})}{64}$	$\frac{(45+90a_{\rm f})}{512}$	$\frac{(-5+10a_{\rm f})}{256}$	$\frac{(1-2a_{\rm f})}{512}$

The filter of Equation (13) was introduced in Reference [23] and it is based on the family of filters proposed in Reference [17]. Proper choice of coefficients suggested in References [23, 24] provides a $2N$th-order accurate formula on a $2N+1$ point stencil. Note that the computed values at the end points of the domain cannot be filtered even with the second-order filter which has a three point wide stencil. The $N+1$ coefficients, $a_0, a_1, \ldots, a_N$, of the filter in Equation (13) are given for different order filters in Table II where the first entry denotes the order of the filter.

The transfer function of the filters in Equation (13) that determines the resolution of the filter in wave space is given by

$$S(\omega) = \frac{\sum_{n=0}^{N} a_n \cos(n\omega)}{1 + 2a_{\rm f}\cos(\omega)} \tag{14}$$

The parameter $a_{\rm f}$ which is in the range $-0.5 < a_{\rm f} < 0.5$ determines the filtering properties. High values of the parameter $a_{\rm f}$ yield less dissipative filters. The function of Equation (14) for filters of second up to eighth-order for the same value of the filtering parameter $a_{\rm f} = 0.45$ is plotted in Figure 2. The function of Equation (14) for the second-, and eighth-order filter is plotted in Figure 3 for different values of the filtering parameter $a_{\rm f} = 0.45$–0.495. It is evident that as the value of the filtering parameter $a_{\rm f}$ becomes larger low-pass filtering is obtained even with the second-order filter. Numerical experiments in Reference [24] suggest values of the parameter $a_{\rm f}$ between 0.3 and 0.5 and filter order of accuracy at least two orders higher than the order of the basic difference scheme.

The filter stencil of an eight- or tenth-order filter, commonly used with fourth- and sixth-order accurate compact schemes, is very large. For the points of the domain where Equation (13) cannot be applied, two approaches are suitable. The first approach reduces the filter order of accuracy upon approaching the boundary to a level for which the filtering scheme of Equation (13) is applicable. For example, filtering of the points close to the boundary is ac-

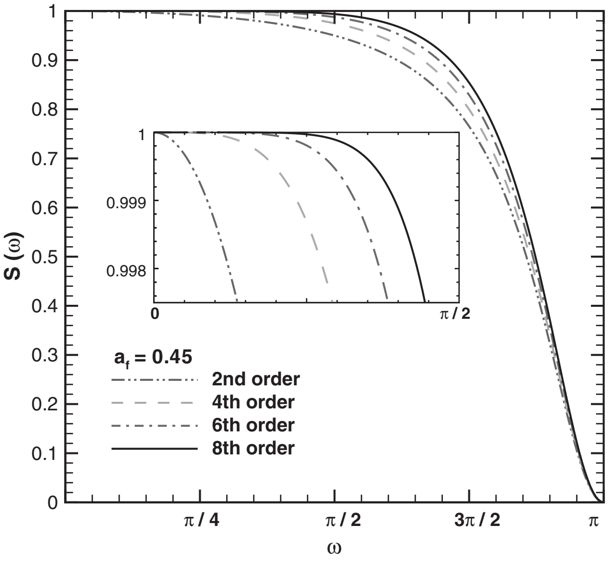

Figure 2. Effect of filter order of accuracy.

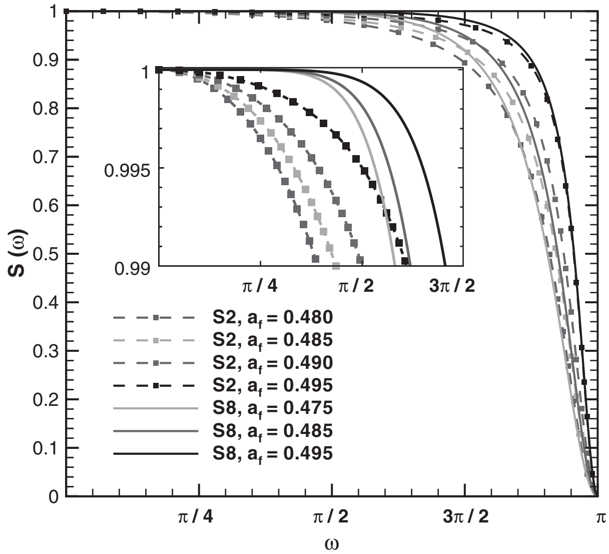

Figure 3. Effect of filtering parameter.

complished with the second-order filter and high values of a_f, while the higher-order (sixth-, eighth-, or tenth-order) filter is used for the interior points. The second method, described in Reference [24], employs higher-order one-sided formulas for points close to the computational boundaries. These formulas retain the tridiagonal form of the scheme and they are given by

$$a_{\mathrm{f}}\hat{\mathscr{Q}}_{j-1} + \hat{\mathscr{Q}}_j + a_{\mathrm{f}}\hat{\mathscr{Q}}_{j+1} = \sum_{n=1}^{11} a_{n,j}\mathscr{Q}_n \quad (2 \leqslant j \leqslant 5) \tag{15}$$

$$a_{\mathrm{f}}\hat{\mathscr{Q}}_{j-1} + \hat{\mathscr{Q}}_j + a_{\mathrm{f}}\hat{\mathscr{Q}}_{j+1} = \sum_{n=0}^{10} a_{n,j}\mathscr{Q}_n \quad (J\max -4 \leqslant j \leqslant J\max -1) \tag{16}$$

An extensive list of the coefficients for the higher-order, one-sided filters at the left and right boundary is given in Reference [25]. For the applications of this work, no significant difference was found between the first or the second approach. The first approach was used in the computed solutions.

5. RESULTS

Preliminary tests of the proposed method were carried out for simple, steady flow problems with exact solutions, such as pipe, channel flow, and flat plate boundary layer. Very good agreement of the computed solution with the analytic results was obtained. Furthermore, the reduction of error with grid refinement demonstrated that high order of accuracy is obtained.

5.1. Cavity flow

A classical steady-state flow problem with well-defined boundary conditions is the unit square driven-cavity flow. This problem is often used [5, 16] to demonstrate the accuracy and efficiency of numerical methods for incompressible flows. Numerical solutions for the driven-cavity flow were obtained at different Reynolds numbers using periodic boundary conditions in the spanwise direction. A comparison of the computed solution at $Re = 100$ with the computations of Reference [26] is shown in Figure 4. The agreement with the computed solution of Ghia *et al.* [26], widely used for comparisons in the literature (see for example Reference [5]), is good. The computed vorticity for the numerical solution obtained with a stretched, 101×101 point grid, the fourth-order accurate scheme, and the eighth-order filter are shown in Plate 1. The computed solutions were not sensitive to the value of the filtering parameter a_{f}. The computed solutions for $0.40 < a_{\mathrm{f}} < 0.49$ converged to the same steady-state.

The fourth-order accurate numerical solution converged to machine accuracy in approximately 150 iterations. Analogous convergence was obtained for the solutions computed on the same mesh for Reynolds numbers up to $Re = 500$ with both the second-order explicit scheme plus the fourth- or sixth-order filter and the sixth-order accurate compact scheme plus the eighth-order filter. Convergence to machine zero was achieved in all cases. However, before machine zero was reached, when the L_2 norm converged at least five orders of magnitude and the velocity divergence was less than 10^{-4} the numerical solution was practically converged to a steady-state. The overall convergence was practically unaffected by the values of the pseudocompressibility parameter in the range $1 < \beta < 5$. Larger values of the pseudocompressibility parameter, $\beta \approx 5$, yielded a slightly faster convergence to a steady state. In general, the numerical solutions of the driven-cavity flow essentially converged in approximately 120 iterations for $100 < Re < 500$ independently of the value of the pseudocompressibility parameter, the value of the filtering parameter, or the order of the scheme.

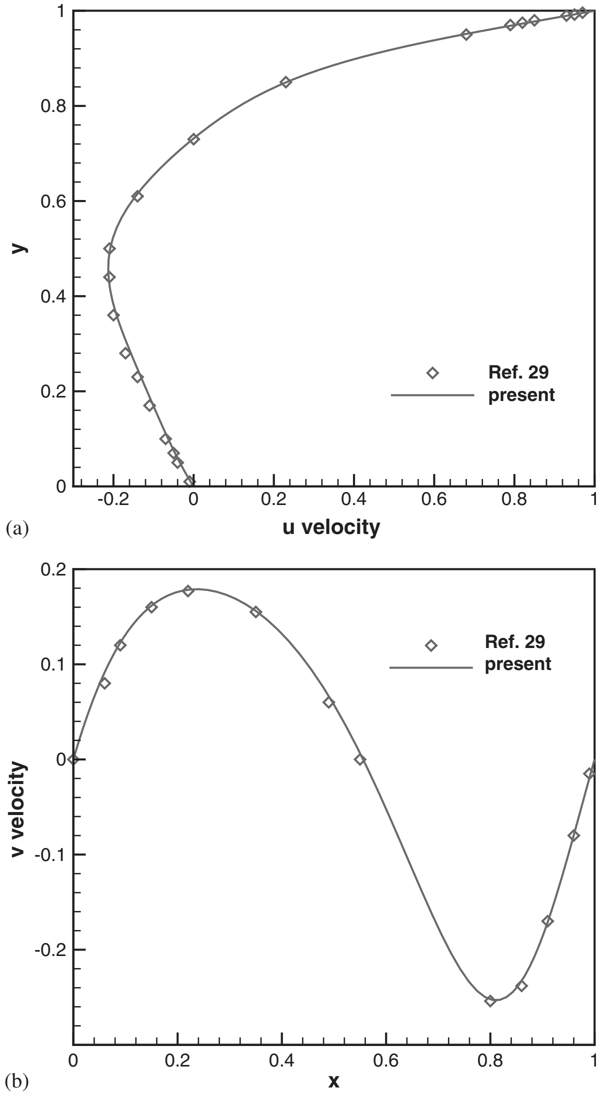

Figure 4. Comparison of the present solution with the results of Ghia *et al.* [26].

5.2. Oscillating plate

The ability of the proposed numerical method to retain high order of accuracy for unsteady incompressible flow problems with exact solutions is demonstrated next. First, the flow over an infinite oscillating plate is computed. The numerical solution initiated impulsively with zero velocity initial condition. For this computation, 81 points distributed with simple exponential stretching along the normal to the wall direction were used. The flow velocity at the upper edge of the domain, which was located approximately 20 units away from the solid wall, was

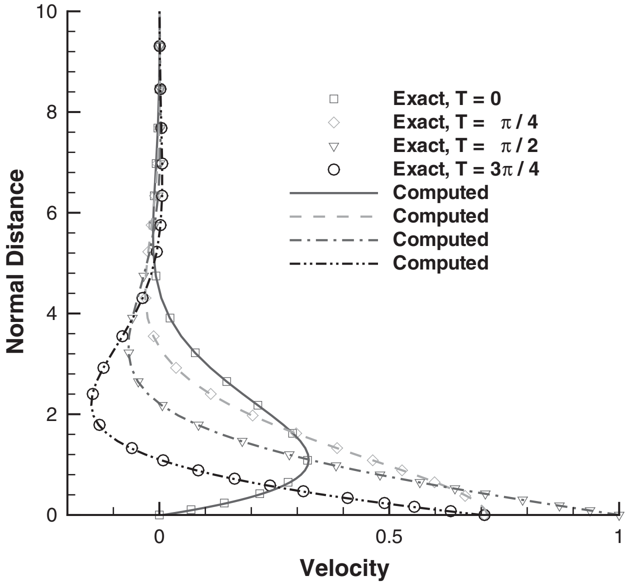

Figure 5. Comparison of the exact result (symbols) with the computed solution (lines) for the Stokes problem.

extrapolated from the interior. In the streamwise and spanwise directions, periodic boundary conditions were imposed. Along the streamwise direction sufficient number of grid points were used in order to make possible the application of the eighth-order filter without the second-order filter closure for the points near the boundaries of the domain. A sufficient number of flow cycles was completed until a time-periodic response was reached. A comparison of the computed velocity distribution with the exact solution at $t=0$, $T/4$, $T/2$, and $t=3T/4$ is shown in Figure 5. The agreement with the exact solution is very good. For this simple problem, a numerical solution was obtained using the explicit time marching scheme of Equation (4) and the dual-time stepping of Equation (5). Filtering was applied at the end of the Runge–Kutta cycle. The numerical solution obtained with the explicit method was in agreement with the solution obtained with the implicit scheme.

5.3. Oseen vortex decay

The second unsteady flow problem with an exact solution is the decay of an ideal vortex. This problem is of interest to numerical simulations of trailing vortices, (LES) and (DES) simulations. The initial condition for the flow is uniform pressure and velocity distribution of the Oseen vortex given by

$$v_\theta(r,t=0)=\frac{\Gamma}{2\pi r} \tag{17}$$

where Γ defines the strength of the vortex and r is the distance from the origin. This vortex decays under the action of viscous forces and the velocity distribution at time t is given by

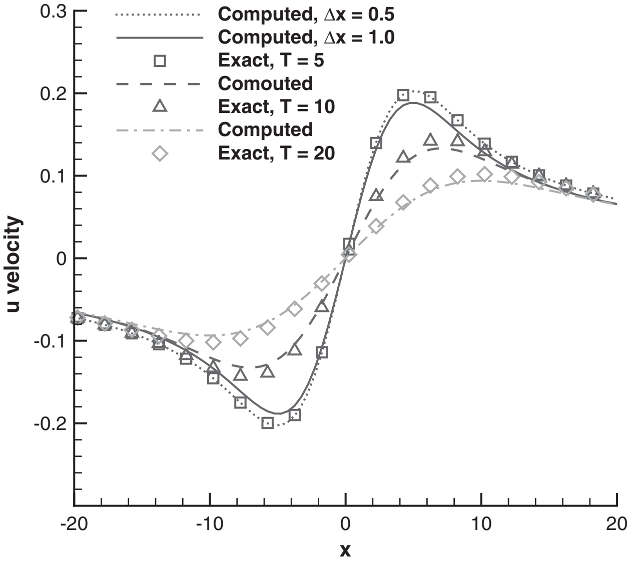

Figure 6. Comparison of the computed (lines) and exact (symbols) velocity for a decaying vortex.

the following exact solution:

$$v_\theta(r,t)=\frac{\Gamma}{2\pi r}\left[1-\exp\left(-\frac{r^2}{4\nu t}\right)\right] \tag{18}$$

The time-dependent flow with initial condition from Equation (17) was computed using an equally spaced, Cartesian grid with $\Delta x=\Delta y=1$ and $\Gamma=10$. A qualitative comparison of the computed solution obtained using the fourth-order accurate compact scheme and the eighth-order filter with the exact result of Equation (18) for time $T=5$ is shown in Plates 2a and 2b. The computed velocity magnitude v_θ and vorticity distribution are in good agreement with the exact result.

A comparison of the computed velocity distributions with the exact result along the grid line that passes close to the origin (at a distance $d=\Delta x/2=0.5$ since the numerical mesh does not pass through the origin where the initial velocity is singular) is shown in Figure 6. The differences between the computed solution and the exact result are large at the region where the gradients are steep. For the computation with $\Delta x=1$, the vortex core, which appears to be confined in the region $r<10$ (see Plate 2), is represented with 20 points. The computation with a refined grid in both directions where the vortex core is represented with 40 points in each direction yielded very good agreement with the exact solution. A comparison of the relative error at $T=5$ and 20 is shown in Figure 7. The error slightly increases with time and grid refinement in both directions reduces the error by an order of magnitude.

A sequence of meshes was used to investigate the decay of error between the exact result and the numerical solution as the mesh is refined and the order of accuracy of the scheme increases from second-order explicit to fourth- and sixth-order compact. The high-order accurate solutions were computed with the eighth-order filter. It was found that the solution is not

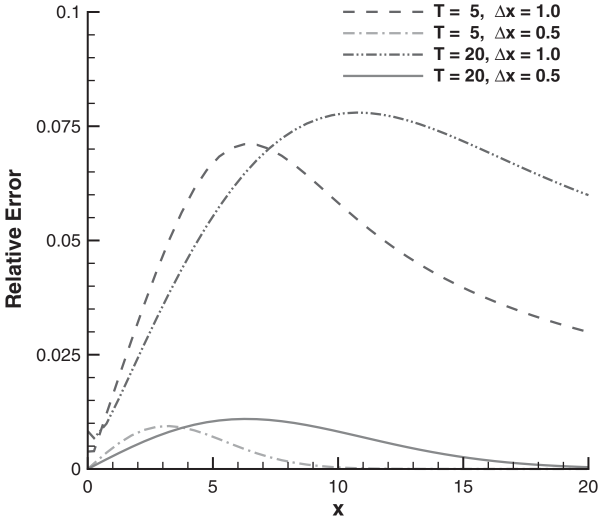

Figure 7. Error of the computed velocity for a decaying vortex.

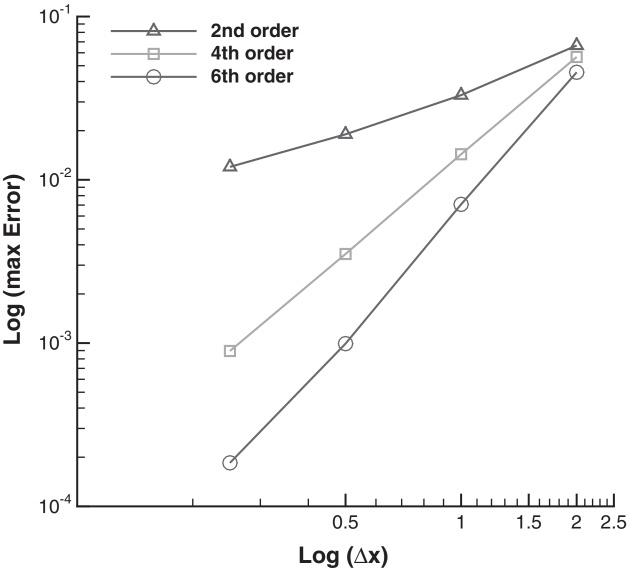

Figure 8. Grid convergence of the computed solution for a decaying vortex.

sensitive to values of the filtering parameter larger than 0.45. The filtering parameter was set to $a_f = 0.475$ for the second-order filter at the boundaries of the domain and $a_f = 0.465$ for the eighth-order filter. The error of the solution versus grid size is shown in Figure 8. It appears that satisfactory grid convergence is obtained for both compact schemes. Grid convergence is demonstrated in Figure 8 for the maximum error, or $L_{\max}$ norm of the velocity at $T = 5$. Analogous grid-convergence properties are obtained, however, for the L_2 norm. Grid convergence is not altered if the norm is based on velocity magnitude or vorticity. Furthermore, the same grid convergence is obtained for different times $T = 10$, 20 during the time-dependent simulation.

5.4. Shear layer

It has been documented in the literature [27, 28] that unresolved numerical simulations of incompressible flow can produce artifacts that are called 'spurious eddies'. In particular, the time-dependent flow with initial condition

$$\begin{aligned} u(y, t = 0) &= \tanh(\delta(y - 0.25)) \quad \text{for } y \leqslant 0.5 \\ u(y, t = 0) &= \tanh(\delta(0.75 - y)) \quad \text{for } y > 0.5 \end{aligned} \tag{19}$$

that represents a shear layer, where δ determines the thickness of the layer, evolves into a periodic eddy pattern when it is perturbed in the y direction. Using a sinusoidal perturbation of the form $v(x, t = 0) = v' \sin(2\pi x)$ as in Reference [28], where v' is perturbation amplitude, the initial shear layer plus the perturbation has a converged solution that takes the form of a regular periodic vortex street. This problem was simulated in the domain $[0, 1] \times [0, 1]$ using a 150×150 point grid. Periodic boundary conditions were imposed in all directions. Because of periodicity it was possible to use the eighth-order filter everywhere. The computed solutions with the second- and fourth-order accurate schemes are compared in Plate 3. In both simulations, the Reynolds number was $Re = 10\,000$, the thickness parameter was $\delta = 100$, and the perturbation amplitude $v' = 0.05$. The fourth-order accurate computation at the top of Plate 3 produced the regular periodic eddy pattern. The second-order accurate computation at the bottom of Plate 3 produced the spurious eddy in the middle of the domain.

5.5. Airfoil flows

Inviscid flow solutions over a NACA-0012 airfoil at 0° angle of incidence were computed on a series of numerical meshes. The far boundary of these meshes was placed 20 chord lengths away from the airfoil surface. The coarsest mesh had 101 points on the airfoil surface and an average grid spacing in the streamwise direction approximately 0.02. The finest grid had 401 points on the airfoil surface and average grid spacing 0.005. In the normal direction, a variable number of grid points was used in order to obtain cells with almost unit aspect ratio close to the surface. The computed drag coefficient converged towards the correct value of zero with grid refinement for all schemes and followed the trends demonstrated in Figure 9.

Turbulent flows over airfoils were computed using the one-equation Spaplart–Allmaras [29] turbulence model. The rapid convergence and efficiency of the numerical solution for attached and separated flow with the highly stretched meshes required for turbulent flow solutions is demonstrated. Numerical solutions at different angles of incidence were computed for a NACA-0012 airfoil. A baseline, 201×81 point, C-type grid with 140 points on the airfoil surface was used for the convergence tests. In addition, an intermediate 301×121 point grid and a fine 401×161 point grid were used to conduct grid independence studies. The convergence of numerical solutions based on the L_2 norm of the pressure residual is shown in Figure 10. Fully attached flows ($\alpha = 5.0°$ and $8.0°$) on the baseline grid converged to machine

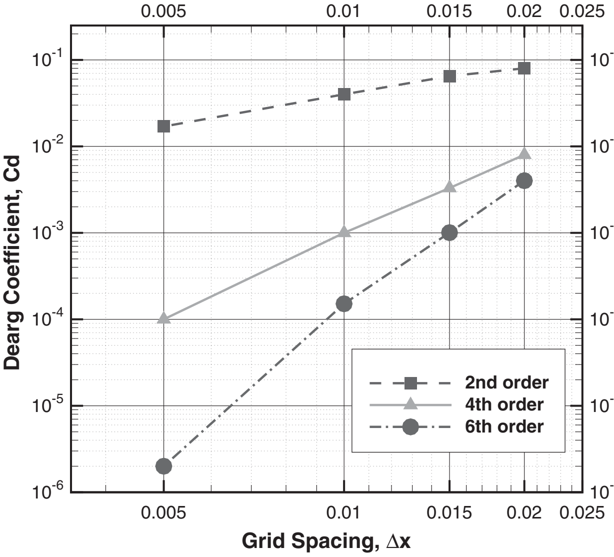

Figure 9. Grid convergence of the computed drag coefficient for inviscid flow at $\alpha = 0.0°$ over the NACA-0012 airfoil.

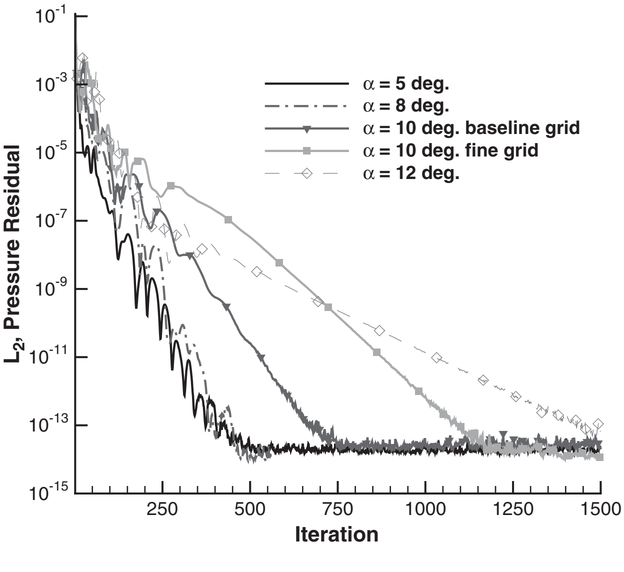

Figure 10. Effect of angle of incidence and grid resolution on the convergence rate of the computed turbulent flow solution over a NACA-0012 airfoil.

accuracy in approximately 500 iterations. The onset of flow separation at $\alpha = 10.0°$ delayed convergence. Convergence was seriously delayed for massively separated flow at $\alpha = 12.0°$ and with the use of fine grids. The computed flowfields at $\alpha = 10.0°$ and $\alpha = 12.0°$ are shown in

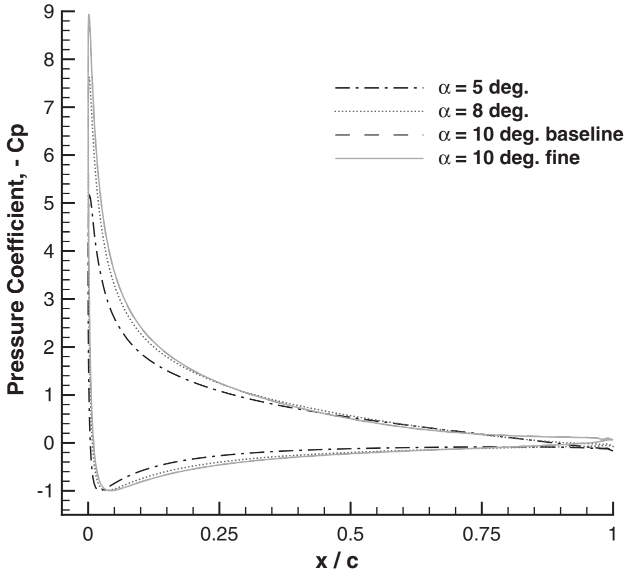

Figure 11. Computed surface pressure coefficients at $Re = 2.0 \times 10^6$.

Plates 4a and 4b. The computed surface pressure coefficients for different angles of incidence are plotted in Figure 11. Grid independent solution, in terms of the computed surface pressure distributions, velocity profiles and skin friction, was obtained for the solutions at $\alpha = 10.0°$ with the fine and the intermediate grid. The computed integrated loads also showed grid-convergence analogous to the convergence observed for the inviscid flow calculations.

5.6. Square duct flow

The flow through a square duct with a 90° bend was used as the final steady-state, three-dimensional test case. The Reynolds number, based on the duct side $D = 1$ and average inflow velocity was $Re_D = 790$. At the inflow a uniform flow velocity was prescribed and the pressure was extrapolated from the interior. The straight inflow section before the bend was set to a length of 40 units. The outflow after the bend was also 40 units long. The computational mesh and the computed flowfield is shown in Plate 5. The mesh is shown for the inner plane of the bend. The outer plane of the bend shows the pressure variation in the streamwise direction. The planes along the streamwise direction show the pressure variation in each section. The recirculation region in each section is indicated with the low pressure area. Numerical solutions were performed in a series of meshes with 21×21, 41×41, 61×61, and 81×81, points in each cross section. The velocities with the fourth-order accurate scheme for solutions obtained with 61×61 and 81×81 were identical. The computed velocities at the symmetry plane are compared with the experimental measurements of Humphrey *et al.* [30] in Figures 12a and 12b. The velocity distribution at $\theta = 60°$ (see Figure 12a) computed with the fourth-order accurate scheme is in better agreement with the experiment. Similar trends are observed in Figure 12b for the computed velocity distribution at $\theta = 90°$. The recirculating flow at $\phi = 30°$ and $\phi = 60°$ is shown in Plates 6a and 6b.

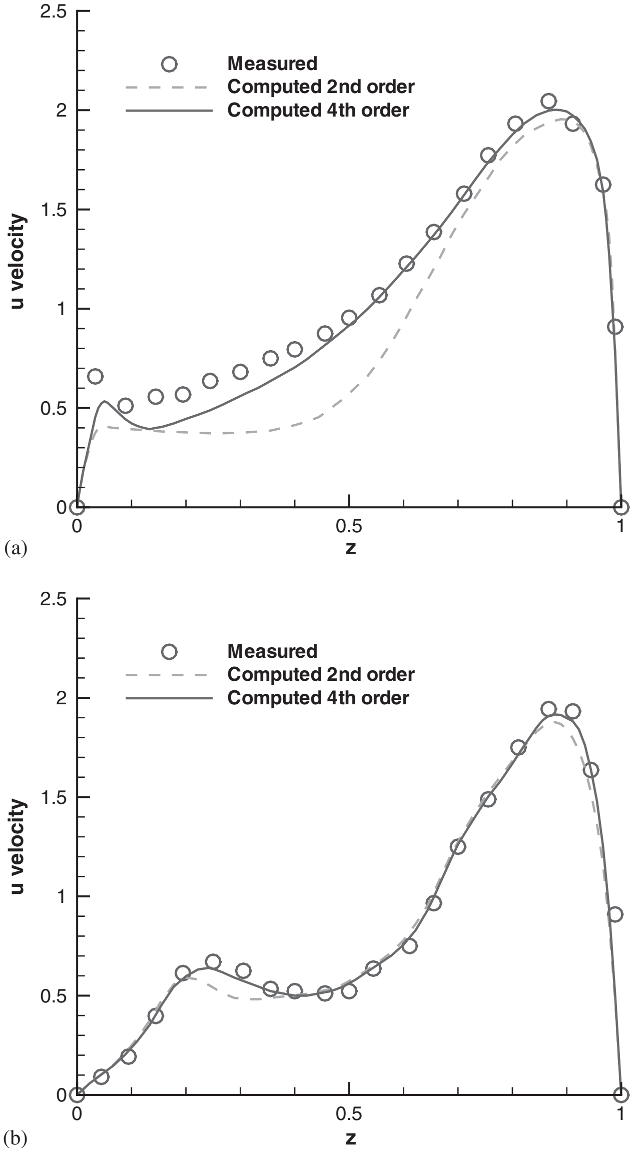

Figure 12. Comparison of the computed velocity with a 41 × 41 point grid with the measurements of Humphrey *et al.* [30].

6. CONCLUSIONS

A high-order-accurate method for computations of incompressible flows in complex domains was presented. This method is based on the artificial compressibility formulation of the incompressible Navier–Stokes equations. It performs centred discretization of the convective and viscous fluxes. For the non-linear convective fluxes, fourth- or sixth-order accuracy is obtained using compact schemes. The viscous fluxes are computed using explicit, centred fourth-order accurate finite differences. Time marching to steady-state or time-accurate numerical solutions are obtained with implicit, dual-time-step algorithms. Spectral-type, low-pass filters are used to postprocess the solution vector and remove unresolved spurious modes resulting from the dissipation free centred spatial discretization. The performance of the proposed method was evaluated for simple and more complex flow problems. For simple problems, high-order accuracy was demonstrated. High-order accuracy was retained for complex flow problems where fast convergence to steady-state was achieved and grid-independent solutions were obtained.

ACKNOWLEDGEMENTS

The author gratefully acknowledges the European Commission Research Directorate-General contract ENK6-CT-2001-00503 KNOW-BLADE, which supported this research.

REFERENCES

1. Chorin AJ. Numerical solution of the Navier–Stokes equations. *Mathematics of Computation* 1968; **22**: 745–762.
2. Kim J, Moin P. Application of a fractional time-step method to incompressible Navier–Stokes equations. *Journal of Computational Physics* 1985; **59**:308–323.
3. Rosenfeld M, Kwak D, Vinokur M. A fractional step solution method for the unsteady incompressible Navier–Stokes equations in generalized coordinate systems. *Journal of Computational Physics* 1991; **94**:102–137.
4. Harlow FH, Welch JE. Numerical calculation of time-dependent viscous incompressible flows with free surface. *Physics of Fluids* 1965; **8**:2182–2189.
5. Jordan SA. An efficient fractional-step technique for unsteady incompressible flows using a semi-staggered grid strategy. *Journal of Computational Physics* 1996; **127**:218–225.
6. Patera AT. A spectral element method for fluid dynamics: laminar flow in a channel expansion. *Journal of Computational Physics* 1984; **54**:468–488.
7. Karniadakis GE, Israeli M, Orszag SA. High-order splitting methods for the incompressible Navier–Stokes equations. *Journal of Computational Physics* 1996; **127**:218–225.
8. Chorin AJ. Numerical methods for solving incompressible, viscous flow problems. *Journal of Computational Physics* 1967; **2**:12–26.
9. Merkle CL, Athavale M. Time-accurate unsteady incompressible flow algorithm based on artificial compressibility. *AIAA Paper* 87-1137, 1987.
10. Ekaterinaris JA. Performance of high-order accurate low-diffusion numerical schemes for compressible flow. *AIAA Paper* 2003-3539 2004; **42**(3):493–500.
11. Ekaterinaris JA. Numerical simulation of incompressible two-blade rotor flowfields. *AIAA Journal of Propulsion and Power* 1998; **14**:367–374.
12. Rogers SE, Kwak D, Kiris C. Steady and unsteady solutions of the incompressible Navier–Stokes equations. *AIAA Journal* 1991; **29**:603–610.
13. Cockburn B, Shu C-W. The Runge–Kutta discontinuous Galerkin method for conservation laws. *Journal of Computational Physics* 1998; **141**(2):199–224.
14. Fyfe DJ. Economical evaluation of Runge–Kutta formulae. *Mathematics of Computation* 1966; **20**:392–398.
15. Venkatakrishnan V, Mavriplis DJ. Implicit method for the computation of unsteady flows on unstructured grids. *Journal of Computational Physics* 1996; **127**:380–397.
16. Drikakis D, Iliev OP, Vassileva DP. A nonlinear multigrid method for the three-dimensional incompressible Navier–Stokes equations. *Journal of Computational Physics* 1998; **146**:301–321.
17. Lele SK. Compact finite difference schemes with spectral-like resolution. *Journal of Computational Physics* 1992; **103**(1):16–42.
18. Ekaterinaris JA. Implicit high-order accurate in space algorithms for the Navier–Stokes equations. *AIAA Journal* 2000; **38**(6):1594–1602.
19. Rizzetta DP, Visbal MR. Application of large-eddy simulation to supersonic compression ramps. *AIAA Journal* 2002; **40**(8):1574–1581.
20. Garnier E, Sagaut M, Deville M. A class of explicit ENO filters with application to unsteady flows. *Journal of Computational Physics* 1993; **180**(1):184–204.

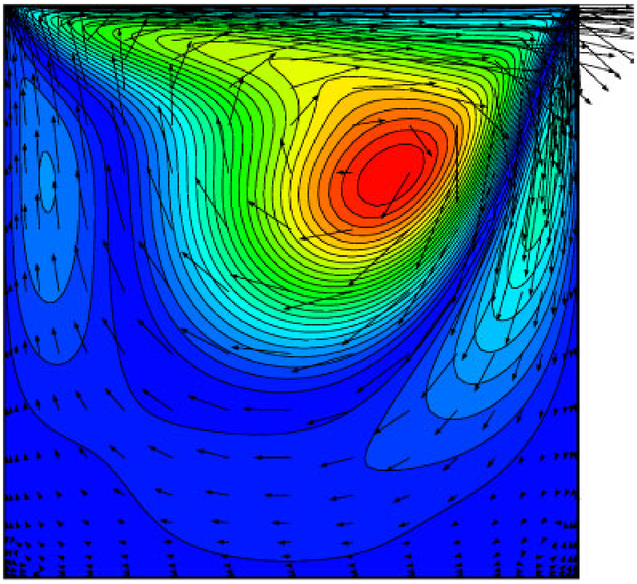

Plate 1. Computed flowfield in the unit square driven-cavity shown by vorticity contours.

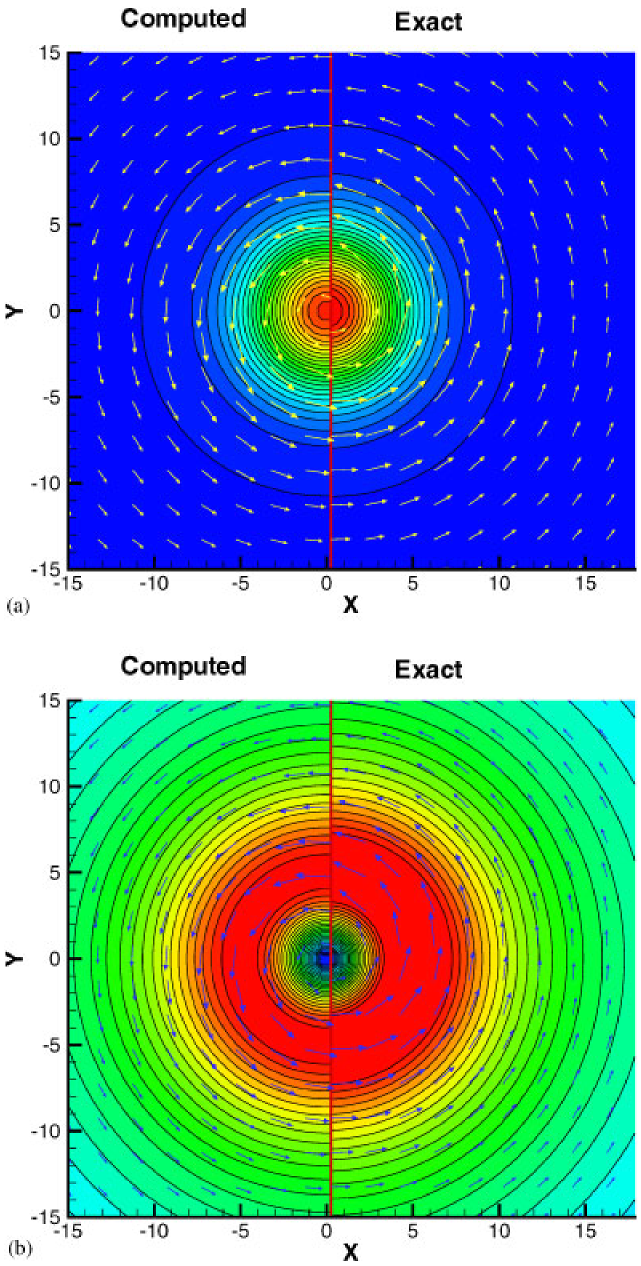

Plate 2. (a) Comparison of the computed and exact vorticity field for a decaying vortex. (b) Comparison of the computed and exact total velocity field for a decaying vortex.

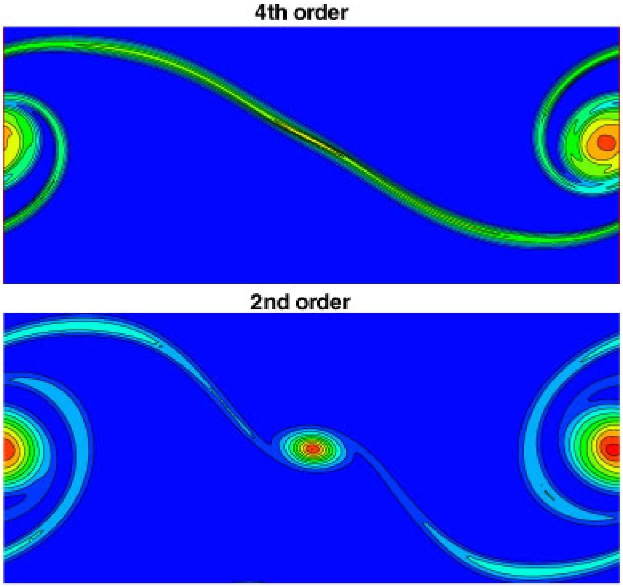

Plate 3. Comparison of the computed vorticity field for the solutions obtained with second- and fourth-order accurate schemes.

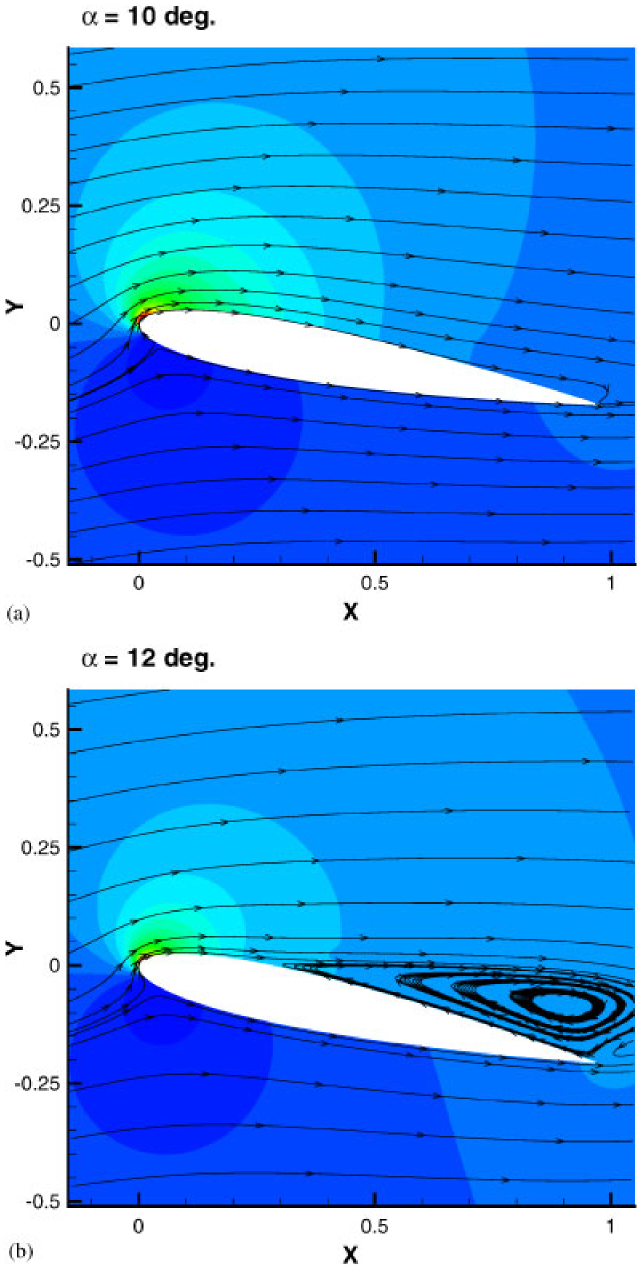

Plate 4. (a) Computed flowfield at $\alpha = 10^\circ$ and $Re = 2.0 \times 10^6$. (b) Computed flowfield at $\alpha = 12^\circ$ and $Re = 2.0 \times 10^6$.

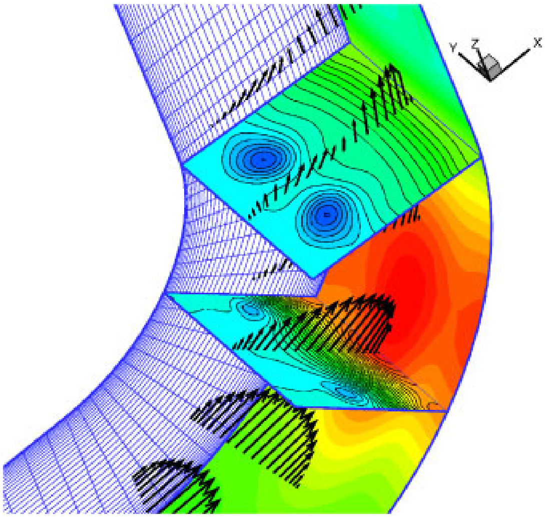

Plate 5. Computational mesh and pressure field at the bend.

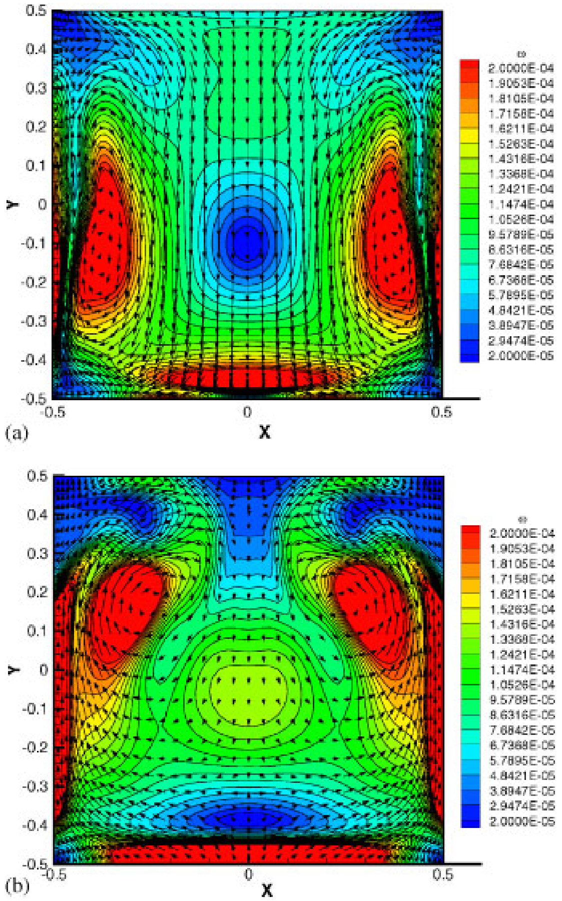

Plate 6. Computed flowfield structure at $\phi = 30°$ and $60°$ shown by vorticity field and velocity vectors.

21. Carpenter MH, Gottlieb D, Abarbanel S. The stability of numerical boundary treatments of compact high-order finite-difference schemes. *Journal of Computational Physics* 1993; **108**(2):272–295.
22. Mahesh K. A family of high order finite difference schemes with good spectral resolution. *Journal of Computational Physics* 1998; **145**:332–358.
23. Visbal MR, Gaitonde DV. High-order accurate methods for unsteady vortical flows on curvilinear meshes. *AIAA Journal* 1999; **37**(10):1231–1239.
24. Visbal MR, Gaitonde DV. On the use of high-order finite-difference schemes on curvilinear and deforming meshes. *Journal of Computational Physics* 2002; **181**:155–185.
25. Gaitonde DV, Visbal MR. High-order schemes for Navier–Stokes equations: algorithm and implementation into FDL3DI. *Airforce Research Laboratory Technical Report* AFRL-VA-WP-TR-1998-3060, 1998.
26. Ghia U, Ghia KN, Shin CT. High-resolutions for incompressible flow using the Navier–Stokes equations and a multigrid method. *Journal of Computational Physics* 1982; **48**:387–411.
27. Minion ML, Brown DL. Performance of under-resolved two-dimensional incompressible simulations. *Journal of Computational Physics* 1997; **122**:734–765.
28. Drikakis D, Smolarkiewicz PK. On spurious vortical structures. *Journal of Computational Physics* 2001; **172**: 309–325.
29. Spalart PR, Allmaras SR. A one-equation turbulence model for aerodynamic flows. *AIAA Paper* 92-0439, 1992.
30. Humphrey AC, Taylor AMK, Whitelaw JH. Laminar flow in a square duct of strong curvature. *Journal of Fluid Mechanics* 1977; **83**:509–527.