Available online at www.sciencedirect.com
SCIENCE DIRECT
Journal of Computational Physics 215 (2006) 589–613
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A staggered grid, high-order accurate method for the incompressible Navier–Stokes equations

Nikolaos A. Kampanis [a,*], John A. Ekaterinaris [a,b]

[a] *Foundation for Research and Technology-Hellas, Institute of Applied and Computational Mathematics, P.O. Box 1385, 71110 Heraklion, Crete, Greece*
[b] *School of Mechanical and Aerospace Engineering, University of Patras, 26110 Patras, Greece*

Received 23 October 2003; received in revised form 11 March 2005; accepted 7 November 2005
Available online 20 December 2005

Abstract

A high-order accurate, finite-difference method for the numerical solution of the incompressible Navier–Stokes equations is presented. Fourth-order accurate discretizations of the convective and viscous fluxes are obtained on staggered meshes using explicit or compact finite-difference formulas. High-order accuracy in time is obtained by marching the solution with Runge–Kutta methods. The incompressibility constraint is enforced for each Runge–Kutta stage iteratively either by local pressure correction or by a Poisson-equation based global pressure correction method. Local pressure correction is carried out on cell by cell basis using a local, fourth-order accurate discrete analog of the continuity equation. The global pressure correction is based on the numerical solution of a Poisson-type equation which is discretized to fourth-order accuracy, and solved using GMRES. In both cases, the updated pressure is used to recompute the velocities in order to satisfy the incompressibility constraint to fourth-order accuracy. The accuracy and efficiency of the proposed method is demonstrated in test problems.

1. Introduction

In many industrial applications, such as flows over hydrofoils, wind-turbine blades, and aircraft wings during takeoff and landing the performance is affected by separated flow occurring at low speeds $M_\infty < 0.1$. For these applications, the flow is practically incompressible. Incompressible flow is also encountered in biomedical and biomechanical applications, such as blood flow, bird and insect flight. Furthermore, the development of affordable, high-speed trains and naval transportation with potential to commercial applications needs detail flowfield information because the harsh environment of such endeavors imposes high structural, and propulsive loads. The difficulty of full-scale testing constrains efforts to evolve new concepts and requires application of advanced, efficient, and accurate methods for the numerical solution of the incompressible flow equations.

* Corresponding author. Tel.: +302810391780; fax: +302810391761.
E-mail addresses: kampanis@iacm.forth.gr (N.A. Kampanis), ekaterin@iacm.forth.gr (J.A. Ekaterinaris).

0021-9991/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2005.11.014

The primary problem with time-accurate solutions of the incompressible flow equations is the difficulty in coupling changes of the velocity field with changes of the pressure field while satisfying the continuity equation. The continuity equation of the incompressible flow equations has a non-evolutionary character because the pressure is included in a non-time-dependent form. Use of the alternative stream-function-velocity and vorticity–velocity formulations [36] are not straight forward in three dimensions. Numerical solutions of the incompressible flow equations are traditionally obtained either with pressure correction methods [11,13] or with fractional time-step methods [5,18,31,16]. Fractional time-step methods are often combined with a Poisson-type equation for pressure correction. Operator splitting and predictor–corrector methods that decouple the computation of velocity and pressure [14] are also used.

The Poisson equation utilized for pressure correction is obtained from the momentum equations, and yields the pressure field distribution in terms of the velocities. Pressure correction with Poisson equation is achieved by advancing the velocity field in time solving numerically the momentum equations, and subsequently solving the Poisson equation for pressure at the current time step so that continuity is enforced in the next time step. As a result, pressure correction using Poisson equation couples velocity and pressure indirectly. The fractional time-step method introduced by Chorin [5], on the other hand, first solves for an intermediate velocity field from the momentum equations and then obtains the pressure field that will map the intermediate velocity field to a divergence free velocity field. The operator-splitting method introduced by Issa [14], computes the pressure with a suitable Poisson equation at each time step. The SIMPLE method [27] also uses a Poisson-type equation to enforce incompressibility. This method was applied for curvilinear coordinates in the finite volume context in Ref. [28].

The artificial compressibility or pseudocompressibility method is often used as an alternative to the previous methods. This method was initially introduced by Chorin [4] for the solution of steady-state incompressible flows, and it was subsequently extended by Merkle and Athavale [23] to time-accurate incompressible flow solutions. Numerical solutions of complex, time-dependent flows were performed using the artificial compressibility method [30,6]. The main disadvantage of the above procedure, however, is that the numerical diffusion required to stabilize the numerical solution is rather high.

It has long been recognized that improved aerodynamic design requires detailed information of the near wall flowfield. For example, although the high drag caused by turbulent flow has significant impact on the operational cost of vehicles, mechanisms of receptivity and nonlinear growth of instabilities, which under various noise environments lead complex flows to transition and turbulence, are poorly understood. Even less understood is the impact of recently developed conventional and unconventional flow control techniques on performance and operational cost. Typical second-order accurate in space methods employed in computational fluid dynamics (CFD) algorithms of incompressible flow [11,8,13,29] require a large number of grid points to resolve the nonlinear growth of instability waves and fine turbulent structures. Furthermore, high-order upwind methods often introduce spurious (numerical) disturbances, which may contaminate the solution beyond acceptable limits, can lead to significant damping of turbulence fluctuations, and can mask the effects of the subgrid-scale (SGS) models used in large eddy simulations (LES). Spectral methods typically employed to overcome these problems are difficult if not impossible to use for complex configurations or even on canonical configurations with flow unsteadiness caused by wall motion. Robust, high fidelity, and accuracy methodologies are therefore required. In recent years, there has been an increased interest in developing methods based on high-order expansions of the discrete solution [3,17].

In parallel, more efficient computationally yet accurate finite-difference and finite-volume methods on staggered grids have been pursued for the numerical solution of incompressible flow equations [24,34,10,22]. A detailed analysis of several space discretization methods on regular or staggered meshes for different formulations of the incompressible flow equations was carried out by Morinishi et al. in [24]. The divergence, advective, skew-symmetric and rotational forms of the convective terms are considered in order to study the conservation properties of the resulting discrete formulations. It was concluded that momentum and kinetic energy are conserved at the discrete level only by certain high-order accurate spacial discretizations. The conservation properties of finite-difference methods for staggered grids were further investigated for non-uniform meshes in [34,10]. In [34], the schemes analyzed in [24] were extended to non-uniform meshes. It was concluded that on non-uniform meshes these schemes do not simultaneously conserve momentum and kinetic energy and the cause of the problem was identified. A second-order accurate scheme on non-uniform meshes was pro-

posed in [10], which simultaneously conserves momentum and kinetic energy. More recently, the conservative staggered grid approach was extended in the finite-volume context by Mahesh et al. in [22].

The present paper focuses on addressing the crucial issues of high-order accuracy and efficiency by developing, evaluating, and demonstrating a new high-order time–space algorithm for incompressible flow. This algorithm is based on novel, high-order accurate in space and time discretizations. High-order explicit finite-difference formulas or compact schemes, applied on a staggered-grid, are used to discretize the convective and viscous fluxes in the momentum equations. In the present paper, the divergence form of the convective terms is considered. The energy conservation properties of the explicit, fourth-order accurate finite-difference method incorporated in the proposed algorithm, have been analyzed by Morinishi et al. in [24]. The high-order accurate compact schemes employed as alternative to explicit finite differences, are based on the formulas given by Gaitonde and Visbal in [7] and analyzed in [12,19,21]. The conservation properties of compact schemes have been studied in the context of compressible flows [25,19]. The momentum equations are advanced in time using the explicit, fourth-order Runge–Kutta method. In the proposed algorithm, incompressibility is enforced using two different approaches, a local pressure correction technique analogous to that used in [13], or a global, Poisson-equation based pressure correction method. In both cases, fourth-order accurate space discretizations are used.

The rest of this paper is organized as follows: The governing equations are shown first. The staggered grid methodology is presented and the high-order discretization procedure is explained. The time marching scheme is defined. The two different approaches proposed for pressure update and incompressibility enforcement to high order are presented. Finally, in the results section the accuracy of the method is demonstrated.

2. Governing equations

For incompressible flow, and without loss of generality the high-order accurate method on staggered grids can be presented for the two-dimensional governing equations. The incompressible Navier–Stokes equations in Cartesian or cylindrical (x, y) coordinates can be expressed as

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\xi}{r}u=0, \tag{1}$$

$$\frac{\partial \mathbf{u}}{\partial t}+\frac{\partial \mathbf{F}}{\partial x}+\frac{\partial \mathbf{G}}{\partial y}+\frac{\xi}{r}\mathbf{S}=-\nabla p+\frac{1}{Re}\left(\frac{\partial \mathbf{F}_v}{\partial x}+\frac{\partial \mathbf{G}_v}{\partial y}+\frac{\xi}{r}\mathbf{S}_v\right), \tag{2}$$

where $\xi=0$ for Cartesian coordinates while $\xi=1$ and $r=x$ is the radial direction for cylindrical coordinates. In (1), (2), $\mathbf{u}=[u, v]^{\mathrm{T}}$ is the velocity vector and p is the pressure, $\mathbf{F}$ and $\mathbf{G}$ are the inviscid flux vectors, Re is the Reynolds number, $Re=UL/\nu$, where U is a characteristic velocity and L a characteristic length, and $\mathbf{F}_v$, and $\mathbf{G}_v$ are the viscous fluxes. The inviscid fluxes are given by

$$\mathbf{F}=[u^2, uv]^{\mathrm{T}}, \quad \mathbf{G}=[vu, v^2]^{\mathrm{T}},$$

and the viscous flux terms are

$$\mathbf{F}_v=\left[\frac{\partial u}{\partial x}, \frac{\partial v}{\partial x}\right]^{\mathrm{T}}, \quad \mathbf{G}_v=\left[\frac{\partial u}{\partial y}, \frac{\partial v}{\partial y}\right]^{\mathrm{T}}.$$

The additional source terms $\mathbf{S}$ and $\mathbf{S}_v$ appearing for cylindrical coordinates are $\mathbf{S}=\mathbf{F}$ and

$$\mathbf{S}_v=\left[\frac{\partial u}{\partial r}-\frac{u}{r}, \frac{\partial v}{\partial r}\right]^{\mathrm{T}}.$$

3. Numerical method

A second-order accurate method [13] was proposed for the numerical solution of incompressible flows on staggered grids with finite-difference discretizations for the convective and diffusive fluxes. The forward Euler method is used to advance in time the discrete equations. For stabilization, the donor cell, first-order upwind

scheme [8], is applied to the convective terms. The velocity field is computed using the momentum equations. Divergence free condition of the computed velocity field at a new time level is enforced iteratively by employing pressure correction. This correction is computed locally on each cell, instead of solving a global Poisson equation to determine simultaneously corrections for all cells. Once the pressure is corrected on each cell, the velocities are recomputed from the momentum equations. All cells are swept successively in a preselected order. After the velocities for one cell are adjusted, velocities of the adjacent cells are inevitably affected via the common nodes. An iterative process is then established for pressure corrections until all local divergences become zero within some tolerance.

In this paper, starting from the basic staggered grid discretization and pressure correction procedure of [13], higher accuracy in space and time is obtained. Fourth-order accuracy in space is obtained using explicit or compact centered schemes. Incompressibility is enforced using two different approaches, both based on the basic procedure of [13], that iteratively compute pressure corrections which introduced to the discrete momentum equations, produce perturbed velocities converging to a divergence-free limit. For the first approach a local, cell-based pressure correction is introduced to the momentum equations (2), and computed subsequently so that the perturbed velocities satisfy zero continuity via (1). For the second approach, globally defined, simultaneously on all cells, pressure corrections are introduced in (2) and computed subsequently via a Poisson-type equation occurring under the requirement that the perturbed velocities satisfy (1). In order to ensure high-order spatial accuracy, the derivatives in (1) or in the associated Poisson-type equation will be discretized to fourth-order accuracy using explicit formulas. The first-order, donor-cell upwind is maintained in the code. However, the computations in this paper were performed without upwinding. For time-stepping, the explicit fourth-order Runge–Kutta method [2] is used.

3.1. Computational grid

Rectangular physical domains in cartesian or cylindrical[1] coordinates are subdivided into rectangular cells $C_{i,j}$. These cells have width Δx and height Δy and vertices defined by the points $x_i = i\Delta x$, $i = 0, \ldots, I$, and $y_j = j\Delta y$, $j = 0, \ldots, J$, on the x- and y-axes, respectively. Midpoints on the vertical and horizontal edges, and the centers of the cells are defined by the associated midpoints on the axis denoted by $x_{i-1/2} = (i - 1/2)\Delta x$ and $y_{j-1/2} = (j - 1/2)\Delta y$. An exterior fictitious layer of three cells, adjacent on each side of the physical domain is added to facilitate the imposition of fourth-order accurate numerical boundary conditions. The additional points that are used to define the cells in the fictitious layer are numbered with $i = -3, -2, -1$ on the left and $i = I + 1, I + 2, I + 3$ on the right on the x-axis, and with $j = -3, -2, -1$ at the bottom and $j = J + 1, J + 2, J + 3$ at the top on the y-axis. The physical domain in unison with the fictitious layer form the computational domain.

The staggered grid is shown in Fig. 1. The dependent variables are discretized at the following locations on the cell $C_{i,j}$: the pressure p denoted by $p_{i,j}$ at the cell center with coordinates $(x_{i-1/2}, y_{j-1/2})$; the u velocity component denoted by $u_{i+1/2,j}$ at the vertical edges midpoints with coordinates $(x_i, y_{j-1/2})$; and the v velocity component denoted by $v_{i,j+1/2}$ at the horizontal edges midpoints with coordinates $(x_{i-1/2}, y_j)$.

3.2. High-order discretization in space

The staggered grid discretization requires that the u-momentum and the v-momentum equations (2) are discretized at the midpoints of the vertical and horizontal edges of the cells, respectively. The continuity equation (1) is discretized at the cell centers. Approximations for the first-order derivatives in the convective flux and source terms, and the second-order derivatives in the viscous flux terms are performed at appropriate locations on the staggered grid. Finite-difference, explicit or compact schemes are employed for the discretization of the derivatives at midpoints $x_{i-1/2}$ (or $y_{j-1/2}$) by using functional values of the variable at integer points x_i (or y_j). The staggered grid implementation of the explicit and compact, fourth-order finite-difference schemes is summarized in the following section.

[1] In cylindrical coordinates the radial direction coincides with the x-axis.

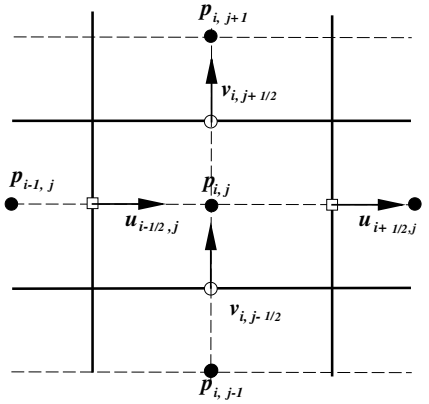

Fig. 1. Schematic of staggered grid.

3.2.1. Explicit fourth-order finite differences

The fourth-order accurate, first-order derivatives at midpoints $x_{i-1/2}$, using standard collocated formulas, are evaluated as

$$\left(\frac{\mathrm{d}f}{\mathrm{d}x}\right)_{i-1/2} \approx \frac{-f_{i+3/2} + 8f_{i+1/2} - 8f_{i-3/2} + f_{i-5/2}}{12\Delta x}, \tag{3}$$

and using functional values at integer points x_i as

$$\left(\frac{\mathrm{d}f}{\mathrm{d}x}\right)_{i-1/2} \approx \frac{-f_{i+1} + 27f_i - 27f_{i-1} + f_{i-2}}{24\Delta x}. \tag{4}$$

The fourth-order accurate, second-order derivatives with standard collocated formulas are evaluated as

$$\left(\frac{\mathrm{d}^2 f}{\mathrm{d}x^2}\right)_{i-1/2} \approx \frac{-f_{i+3/2} + 16f_{i+1/2} - 30f_{i-1/2} + 16f_{i-3/2} - f_{i-5/2}}{12(\Delta x)^2}. \tag{5}$$

Functional values at midpoints are computed as fourth-order averages of functional values of integer points by

$$f_{i-1/2} \approx \frac{-f_{i+1} + 9f_i + 9f_{i-1} - f_{i-2}}{16}. \tag{6}$$

Discretization of the nonlinear terms appearing in the convective fluxes is then obtained as follows:

$$\left(\frac{\partial(u^2)}{\partial x}\right)_{i+1/2,j} \approx \frac{-(u^2)_{i+2,j} + 27(u^2)_{i+1,j} - 27(u^2)_{i,j} + (u^2)_{i-1,j}}{24\Delta x}, \tag{7}$$

and

$$\left(\frac{\partial(uv)}{\partial y}\right)_{i+1/2,j} \approx \frac{-(uv)_{i+\frac{1}{2},j+\frac{3}{2}} + 27(uv)_{i+\frac{1}{2},j+\frac{1}{2}} - 27(uv)_{i+\frac{1}{2},j-\frac{1}{2}} + (uv)_{i+\frac{1}{2},j-\frac{3}{2}}}{24\Delta y}. \tag{8}$$

Functional values of the products u^2 and uv appearing in the above expressions are computed by

$$(u^2)_{i,j} \approx \left(\frac{-u_{i+3/2,j} + 9u_{i+1/2,j} + 9u_{i-1/2,j} - u_{i-3/2,j}}{16}\right)^2, \tag{9}$$

and

$$(uv)_{i+\frac{1}{2},j+\frac{1}{2}} \approx \left(\frac{-u_{i+1/2,j+2}+9u_{i+1/2,j+1}+9u_{i+1/2,j}-u_{i+1/2,j-1}}{16}\right) \cdot \left(\frac{-v_{i+2,j+1/2}+9v_{i+1,j+1/2}+9v_{i,j+1/2}-v_{i-1,j+1/2}}{16}\right). \tag{10}$$

The diffusive fluxes $(\partial^2 u/\partial x^2)_{i+1/2,j}$ and $(\partial^2 u/\partial y^2)_{i+1/2,j}$ are discretized using (5) at $i+1/2$ (for all j) and at j (for all $i+1/2$), respectively. For the source term appearing in cylindrical coordinates, application of (4) yields

$$\left(\frac{1}{r}\frac{\partial u}{\partial r}\right)_{i+1/2,j} \approx \frac{1}{12 r_i \Delta r}\left(-u_{i+\frac{5}{2},j}+8u_{i+\frac{3}{2},j}-8u_{i-\frac{1}{2},j}+u_{i-\frac{3}{2},j}\right),$$

while $(u^2/r)_{i+1/2,j} = (u^2)_{i+1/2,j}/r_i$ and $(u^2)_{i+1/2,j}$ is computed as in (9).

In the resulting semidiscrete momentum equations a first-order upwind term, obtained by the upstream, donor-cell differencing scheme of [8], can be introduced. Upwinding may be used in certain cases where numerical stability should be enforced. However, the numerical simulations presented in the subsequent sections were performed without upwinding.

Using the same notation as in Morinishi et al. [24] the staggered grid fourth-order accurate finite-difference formula (4) can be written as

$$D_x^4 f = \frac{9}{8}\frac{\delta_1 f}{\delta_1 x} - \frac{1}{8}\frac{\delta_3 f}{\delta_3 x}, \tag{11}$$

where

$$\frac{\delta_n f}{\delta_n x} = \frac{f(x+nh/2)-f(x-nh/2)}{nh},$$

and the staggered grid fourth-order accurate average (6) can be written as

$$M_x^4 f = \frac{9}{8}\bar{f}^{1x} - \frac{1}{8}\bar{f}^{3x}, \tag{12}$$

where

$$\bar{f}^{nx} = \frac{f(x+nh/2)+f(x-nh/2)}{2}.$$

The discretization of the nonlinear convective terms in the momentum equations suggested by (8)–(10), can be written according to (11) and (12), with $x_1 = x$, $x_2 = y$, $u_1 = u$, and $u_2 = v$, as

$$D_{x_j}^4\left((M_{x_j}^4 u_i)(M_{x_i}^4 u_j)\right), \tag{13}$$

which is analogous to Eq. (98) of Ref. [24]. Due to divergence form of (2), momentum is conserved by (13). At the discrete level, however, momentum is conserved within the discrete error in continuity. This error is negligible because, as is shown in the following sections, pressure is corrected up to fourth-order of accuracy and continuity is satisfied to machine accuracy (10^{-15} at least). However, Eq. (13) yields a conservation error for the kinetic energy as is pointed out in [24]. The discrete formula (13) can be substituted (see Eq. (101) of [24]) by

$$\frac{9}{8}\frac{\delta_1}{\delta_1 x_j}\left((M_{x_i}^4 u_j)\bar{u}_i^{1x_i}\right) - \frac{1}{8}\frac{\delta_3}{\delta_3 x_j}\left((M_{x_i}^4 u_j)\bar{u}_j^{3x_j}\right) \tag{14}$$

to achieve better conservation for the kinetic energy at the discrete level.

The associated pressure discretization also conserves momentum and kinetic energy at the discrete level within the discrete error in continuity, introduced by the pressure correction.

3.2.2. Compact fourth-order schemes

Fourth-order accuracy can also be obtained for the first and second derivatives, by using compact, Padé-type finite-difference schemes [7,12]. Compact schemes evaluate the derivatives in a coupled fashion by performing tridiagonal matrix inversions. Compact schemes have improved resolution in wavespace [19,21] in the sense that when applied to problems with a wide range of spatial scales they provide better representation

of the shorter length scales, compared with the wider stensil, but more efficient computationally explicit, finite-difference schemes of the same order. The conservation properties of compact schemes have been studied [25,19] in the context of compressible flows. Compact schemes which are suitably modified for the near boundary nodes satisfy a discrete type of energy conservation when applied to the conservative form of the governing equations [19]. Numerical experiments demonstrated (see Section 6) that using compact schemes for space discretization in the proposed algorithm also preserve energy for incompressible flows.

The first-order derivatives at integer points x_i are computed in a coupled fashion using functional values at integer points, by the simultaneous solution of the following system of linear equations:

$$h\left(\alpha\phi'_{i-1}+\phi'_i+\alpha\phi'_{i+1}\right)=\frac{a}{2}(\phi_{i+1}-\phi_{i-1})+\frac{b}{4}(\phi_{i+2}-\phi_{i-2}). \tag{15}$$

For $\alpha=1/4$, $a=3/2$ and $b=0$ the fourth-order accurate three-point stencil, compact scheme that requires the solution of a tridiagonal linear system is obtained.

In (15), the following boundary closures are used:

$$h(\phi'_1+3\phi'_2)=-\frac{17}{6}\phi_1+\frac{3}{2}(\phi_2+\phi_3)-\frac{1}{6}\phi_4,$$
$$h(3\phi'_{N-1}+\phi'_N)=\frac{1}{6}\phi_{N-3}-\frac{3}{2}(\phi_{N-2}+\phi_{N-1})+\frac{17}{6}\phi_N.$$

The first-order derivatives at integer points are computed using functional values at midpoints by solving the following linear system:

$$h\left(\alpha\phi'_{i-1}+\phi'_i+\alpha\phi'_{i+1}\right)=a\left(\phi_{i+\frac{1}{2}}-\phi_{i-\frac{1}{2}}\right)+\frac{b}{3}\left(\phi_{i+\frac{3}{2}}-\phi_{i-\frac{3}{2}}\right). \tag{16}$$

Here $a=(9-6\alpha)/8$ and $b=(-1+22\alpha)/8$. For $\alpha=1/22$, $b=0$ and a three-point stencil is obtained.

In (16), the following boundary closures are used:

$$h(\phi'_1+22\phi'_2)=-\frac{577}{24}\phi_{\frac{3}{2}}+\frac{603}{24}\phi_{\frac{5}{2}}-\frac{27}{24}\phi_{\frac{7}{2}}+\frac{1}{24}\phi_{\frac{9}{2}},$$
$$h\left(22\phi'_{N-1}+\phi'_N\right)=-\frac{1}{24}\phi_{N-\frac{7}{2}}+\frac{27}{24}\phi_{N-\frac{5}{2}}-\frac{603}{24}\phi_{N-\frac{3}{2}}+\frac{577}{24}\phi_{N-\frac{1}{2}}.$$

The required functional values at midpoints are obtained by compact interpolation of functional values at integer points as follows:

$$\alpha\phi_{i-\frac{1}{2}}+\phi_{i+\frac{1}{2}}+\alpha\phi_{i+\frac{3}{2}}=\frac{a}{2}(\phi_{i+1}+\phi_i)+\frac{b}{2}(\phi_{i+2}+\phi_{i-1}), \tag{17}$$

where $a=(9+10\alpha)/8$ and $b=(-1+6\alpha)/8$. For $\alpha=1/6$, $b=0$ and a three-point stencil is obtained.

In (17), the following boundary closures are used:

$$\phi_{\frac{3}{2}}+\phi_{\frac{5}{2}}=\frac{1}{4}(\phi_1+\phi_3)+\frac{3}{2}\phi_2,$$
$$\phi_{N-\frac{3}{2}}+\phi_{N-\frac{1}{2}}=\frac{1}{4}(\phi_{N-2}+\phi_N)+\frac{3}{2}\phi_{N-1}.$$

The convective fluxes **F** and **G** in (1)–(2) are discretized on the staggered grid using (15)–(17) compact schemes. Discretization of the second-order derivatives in the viscous fluxes $\mathbf{F}_v$ and $\mathbf{G}_v$ on staggered grids requires special treatment. Successive application of (15) leads to numerical instabilities [7]. Fourth-order compact schemes for the second derivative [19], on the other hand, require pentadiagonal matrix inversion. Here the second-order derivatives are computed with successive application of (16). Numerical experiments demonstrate that this approach is stable.

4. Time marching

The semidiscrete, i.e. the continuous in time momentum equations resulting from (2) after space discretization is applied, are written in compact form as follows:

$$\frac{d\mathbf{U}(t)}{dt} = \mathbf{R}(\mathbf{U}, P; t), \tag{18}$$

where $\mathbf{R}(\mathbf{U}, P; t) = -\nabla P + \mathbf{A}(\mathbf{U}; t)$. The derivatives $(\partial_x P)_{i+1/2,j}$ and $(\partial_y P)_{i,j+1/2}$, appearing in the pressure gradient ∇P, are evaluated to fourth-order accuracy using (4). The term $\mathbf{A} = (A_u, A_v)$, with $A_u(\mathbf{U}; t)$ and $A_v(\mathbf{U}; t)$, denotes the space discretization of the remaining terms in (2).

A q-stage Runge–Kutta method is defined by a set of $q(q+2)$ parameters, given as the entries of a matrix $A = (a_{ij}) \in \mathbf{R}^{q\times q}$, and two vectors $b = (b_1, \ldots, b_q)^{\mathrm{T}} \in \mathbf{R}^q$, $\tau = (\tau_1, \ldots, \tau_q)^{\mathrm{T}} \in \mathbf{R}^q$. The application of these methods to the numerical solution of ordinary differential equations has been extensively studied, [2]. Using the semidiscrete approach or method of lines, time marching is obtained by solving a system of ordinary differential equations given by (18). The four-stages explicit Runge–Kutta method is obtained with $a_{21} = a_{32} = 1/2$ and $a_{43} = 1$, and $a_{ij} = 0$, $b_1 = b_4 = 1/6$, $b_2 = b_3 = 1/3$, $\tau_1 = 0$, $\tau_2 = \tau_3 = 1/2$, and $\tau_4 = 1$ as follows:

$$\mathbf{U}^{n,1} = \mathbf{U}^n, \quad P^{n,1} = P^n, \tag{19}$$

$$\mathbf{U}^{n,2} = \mathbf{U}^n + \frac{\Delta t}{2}\mathbf{R}^{n,1}, \tag{20}$$

$$\mathbf{U}^{n,3} = \mathbf{U}^n + \frac{\Delta t}{2}\mathbf{R}^{n,2}, \tag{21}$$

$$\mathbf{U}^{n,4} = \mathbf{U}^n + \Delta t\mathbf{R}^{n,3}, \tag{22}$$

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{6}(\mathbf{R}^{n,1} + 2\mathbf{R}^{n,2} + 2\mathbf{R}^{n,3} + \mathbf{R}^{n,4}), \tag{23}$$

where $t^n = n\Delta t$, $t^{n,1} = t^n$, $t^{n,2} = t^{n,3} = t^n + \Delta t/2$, $t^{n,4} = t^n + \Delta t$, and $\mathbf{R}^{n,\ell} = \mathbf{R}(\mathbf{U}^{n,\ell}, P^{n,\ell}; t^{n,\ell})$ for $\ell = 2, 3, 4$.

The quantities $P^{n,\ell}$, $\ell = 2, 3, 4$, and P^{n+1}, appearing in (19)–(23), are determined by enforcing the incompressibility on the velocity vectors $\mathbf{U}^{n,\ell}$, $\ell = 2, 3, 4$, and $\mathbf{U}^{n+1}$. Incompressibility is enforced on the velocity vectors $\mathbf{U}^{n,\ell}$, $\ell = 2, 3, 4$, and $\mathbf{U}^{n+1}$ by the pressure update procedures described in the next section.

The fully explicit time-marching scheme was chosen because implementation of semi-implicit schemes in 3D is computationally intensive. In addition, CFL stability limitations of explicit methods can be alleviated with multigrid acceleration techniques.

5. Incompressibility condition

The velocities vectors $\mathbf{U}^{n,\ell}$, $\ell = 2, 3, 4$, and $\mathbf{U}^{n+1}$ obtained from (20)–(23) do not necessarily satisfy the incompressibility constraint of (1). Incompressibility is enforced to the computed velocity field at each time step and during the intermediate stages iteratively through pressure updates. Pressure updates are obtained with two approaches. The first approach applies pressure corrections locally on each cell by sweeping them in a predefined order and recomputing the velocities on each cell. The second approach computes pressures corrections for all cells simultaneously by solving a discrete Poisson-type equation globally and subsequently recomputing the velocities for all cells simultaneously. Both procedures must be applied iteratively after the pressure update is completed because the corrected velocities on adjacent cells are correlated via common staggered grid points and their values violate incompressibility locally. Therefore, iterations of pressure updates are carried out within each time step, until the divergence of the velocity field becomes smaller than a given machine tolerance. The spatial derivatives appearing in the continuity equation during the pressure update procedures, are computed using fourth-order accurate formulas.

A single iteration of the local pressure correction method is simpler to apply and computationally less demanding than the correction through the numerical solution of the Poisson equation, which is a global pressure correction method. For low Reynolds number flows the number of iterations required per time step by the Poisson-equation global pressure correction is slightly less than that for the local pressure correction. As a result, the local pressure correction method appears more attractive regarding the total computational time required. However, the advantage offered by the local pressure correction method per iteration does not characterizes the method as the most effective for all flow situations. Numerical experiments showed that by increasing the Reynolds number, the iterations required by the local pressure correction method per time

stage/step of the Runge–Kutta, time marching method, progressively increases compared to those required by the Poisson-equation global pressure correction method. Therefore, for high Reynolds number flows, the Poisson-equation global pressure correction method appears to become a more effective. Further comments on the efficiency of both methods can be found in the numerical results section.

5.1. Local pressure correction method

In this subsection, the local pressure correction procedure is described. This procedure is actually a high-order analog of the pressure correction technique used in [13]. The updated velocity vectors $\mathbf{U}^{n,\ell}$, $\ell = 2, 3, 4$, resulting after pressure updates during the intermediate Runge–Kutta stages and the velocity vector $\mathbf{U}^{n+1}$ of the next time step, then satisfy the incompressibility condition to fourth-order accuracy.

For the first stage, no correction is required because $\mathbf{U}^n$ is the approximate solution of the previous time step that already satisfies the incompressibility condition. For stages $\ell = 2, 3, 4$, the incompressibility constrain is enforced as follows:

5.1.1. Correction for the ℓth intermediate update

Let $\mathbf{U}^{n,\ell}_{\text{old}}$ be the discrete, intermediate velocity vector obtained from the ℓth stage of (19)–(22), and let $P^{n,\ell}_{\text{old}} = P^{n,\ell-1}$ be the current pressure distribution. The corrected pressure for the cell $C_{i,j}$, with $1 \leqslant i \leqslant I$, $1 \leqslant j \leqslant J$, is

$$(p_{\text{new}})^{n,\ell}_{i,j} = (p_{\text{old}})^{n,\ell}_{i,j} + \Delta p, \tag{24}$$

with Δp to be determined. The update of the velocity vector $\mathbf{U}$, resulting from this pressure correction is

$$\mathbf{U}^{n,\ell}_{\text{new}} = \mathbf{U}^n + a_{\ell,\ell-1}\Delta t \widetilde{\mathbf{R}}^{n,\ell-1}, \tag{25}$$

where $\widetilde{\mathbf{R}}^{n,\ell-1} = \mathbf{R}(\mathbf{U}^{n,\ell-1}, P^{n,\ell}_{\text{new}}; t^{n,\ell-1})$. Eq. (25) in combination with (24) implies that the only components of $\mathbf{U}^{n,\ell}_{\text{new}}$ different from those of $\mathbf{U}^{n,\ell}_{\text{old}}$ are the following:

$$(u_{\text{new}})^{n,\ell}_{i\pm1/2,j} = (u_{\text{old}})^{n,\ell}_{i\pm1/2,j} \pm a_{\ell,\ell-1}\Delta t \frac{27\Delta p}{24\Delta x}, \tag{26}$$

$$(u_{\text{new}})^{n,\ell}_{i\pm3/2,j} = (u_{\text{old}})^{n,\ell}_{i\pm3/2,j} \mp a_{\ell,\ell-1}\Delta t \frac{\Delta p}{24\Delta x}, \tag{27}$$

$$(v_{\text{new}})^{n,\ell}_{i,j\pm1/2} = (v_{\text{old}})^{n,\ell}_{i,j\pm1/2} \pm a_{\ell,\ell-1}\Delta t \frac{27\Delta p}{24\Delta y}, \tag{28}$$

$$(v_{\text{new}})^{n,\ell}_{i,j\pm3/2} = (v_{\text{old}})^{n,\ell}_{i,j\pm3/2} \mp a_{\ell,\ell-1}\Delta t \frac{\Delta p}{24\Delta y}. \tag{29}$$

The discrete, fourth-order accurate divergence, $D_{i,j}$, of the velocity vector $\mathbf{U}$ for the cell $C_{i,j}$, obtained by applying (4) and (6) to the left–hand side of (1), is

$$D_{i,j} = \frac{-u_{i+\frac{3}{2},j} + 27u_{i+\frac{1}{2},j} - 27u_{i-\frac{1}{2},j} + u_{i-\frac{3}{2},j}}{24\Delta x} + \frac{-v_{i,j+\frac{3}{2}} + 27v_{i,j+\frac{1}{2}} - 27v_{i,j-\frac{1}{2}} + v_{i,j-\frac{3}{2}}}{24\Delta y} + \left(\frac{\xi}{r_{i-\frac{1}{2}}}\right)\frac{-u_{i+\frac{3}{2},j} + 9u_{i+\frac{1}{2},j} + 9u_{i-\frac{1}{2},j} - u_{i-\frac{3}{2},j}}{16}. \tag{30}$$

The pressure correction Δp is determined by setting the discrete divergence of Eq. (30) equal to zero using the updated values $\mathbf{U}^{n,\ell}_{\text{new}}$ of (26)–(29). The zero divergence condition $(D_{\text{new}})^{n,\ell}_{i,j} = 0$, with (26)–(29) yields

$$(D_{\text{old}})^{n,\ell}_{i,j} + \frac{365}{144}\left(\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2}\right)a_{\ell,\ell-1}\Delta t\Delta p = 0,$$

and the resulting pressure correction is

$$\Delta p = -\frac{144(D_{\text{old}})^{n,\ell}_{i,j}}{365\left(\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2}\right)a_{\ell,\ell-1}\Delta t}. \tag{31}$$

The pressure correction given by (31) is used in (26)–(29) (or equivalently in (25)) to determine $\mathbf{U}_{\text{new}}^{n,\ell}$.

This procedure corrects the approximate intermediate velocity vector $\mathbf{U}_{\text{old}}^{n,\ell}$ to a new velocity vector $\mathbf{U}_{\text{new}}^{n,\ell}$ which satisfies the discrete incompressibility condition (30) in the $C_{i,j}$ cell. However, the common points of the staggered grid cause violation of the incompressibility condition in the adjacent cells $C_{i-2,j}$, $C_{i-1,j}$, $C_{i+1,j}$, $C_{i,j-2}$, $C_{i,j-1}$ and $C_{i,j+1}$. As a result, in order to enforce global incompressibility the above procedure should be repeated for the pressure updates. The pressure correction procedure performs a complete sweep of all cells. This sweep can be performed row by row from left to right. Initially, $\mathbf{U}_{\text{new}}^{n,\ell}$ is corrected using (26)–(29) for each cell, then $\mathbf{U}_{\text{old}}^{n,\ell} = \mathbf{U}_{\text{new}}^{n,\ell}$, $P_{\text{old}}^{n,\ell} = P_{\text{new}}^{n,\ell}$ are substituted before proceeding to the next cell. Application of the full pressure correction procedure sweep is repeated until the discrete divergence $(D_{\text{old}})_{i,j}^{n,\ell}$ approaches zero for all cells simultaneously within some preset tolerance. The velocity vector and the pressure are updated as $\mathbf{U}^{n,\ell} = \mathbf{U}_{\text{new}}^{n,\ell}$ and $P^{n,\ell} = P_{\text{new}}^{n,\ell}$, once the discrete divergence-free condition (30) is globally satisfied. After global discrete incompressibility is reached iteratively the solution proceeds to the next stage of the Runge–Kutta method.

5.1.2. Correction for the n + 1 update

For the last stage (23), the Runge–Kutta method obtains the velocity vector for the next time level by

$$\mathbf{U}_{\text{old}}^{n+1} = \mathbf{U}^{n} + \Delta t \sum_{\ell=1}^{4} b_{\ell}(-\nabla P^{n,\ell} + \mathbf{A}^{n,\ell}),$$

where $P_{\text{old}}^{n+1} = \sum_{\ell=1}^{4} b_{\ell} P^{n,\ell}$. In addition, for the cell $C_{i,j}$ let

$$(p_{\text{new}})_{i,j}^{n+1} = (p_{\text{old}})_{i,j}^{n+1} + \Delta p,$$

and set

$$\mathbf{U}_{\text{new}}^{n+1} = \mathbf{U}^{n} + \Delta t \left(-\nabla P_{\text{new}}^{n+1} + \sum_{\ell=1}^{4} b_{\ell} \mathbf{A}^{n,\ell} \right).$$

The components of $\mathbf{U}_{\text{old}}^{n+1}$ and $\mathbf{U}_{\text{new}}^{n+1}$ velocity vectors are related by a set of equations analogous to (26)–(29), only the superscripts $n, \ell - 1$ should be replaced by $n + 1$, and $a_{\ell,\ell-1}$ by one. Setting the discrete divergence (cf. (30)) of $\mathbf{U}_{\text{new}}^{n+1}$ equal to zero, i.e. $D_{\text{new}}^{n+1} = 0$, obtain

$$\Delta p = -\frac{144 (D_{\text{old}})_{i,j}^{n}}{365 \left(\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2} \right) \Delta t}.$$

An iteration analogous to that used for the intermediate stages is applied to impose the divergence-free constraint to the discrete velocity vector at time step $n + 1$, $\mathbf{U}^{n+1}$.

5.2. Poisson equation pressure correction method

In this section a procedure to enforce incompressibility iteratively by solving a global Poisson-type equation to compute pressure updates is presented. This discrete Poisson-type equation results by first applying a fourth-order accurate stencil to discretize the pressure gradient in (2) which is then substituted in the fourth-order accurate discrete divergence given by (30). Two different approaches may be used for the derivation of the discrete Poisson-type equation. The resulting stencils have different width. Discrete velocities at the intermediate stages and the current time step are then updated to satisfy the incompressibility condition up to fourth-order accuracy.

5.2.1. Wide stencil, fourth-order accurate Poisson update

The first approach for global pressure correction initiates by discretizing the momentum equations (2) with fourth-order accurate stencils. Then considering a pressure correction along with a fourth-order accurate approximation of the continuity equation arrive to a discrete Poisson-type equation for the pressure correction. As a result, a wide, fourth-order accurate stencil is obtained.

The approximate velocity at the first stage satisfies the incompressibility condition. For an intermediate stage ℓ, let $\mathbf{U}^{n,\ell}_{\text{old}}$ denote the discrete intermediate velocity vector obtained from the ℓth stage of (19)–(22), and set $P^{n,\ell}_{\text{old}} = P^{n,\ell-1}$. A modified pressure $P^{n,\ell}_{\text{new}}$ is defined as follows:

$$P^{n,\ell}_{\text{new}} = P^{n,\ell}_{\text{old}} + \Delta P,$$

and the velocity correction as in (25) defines a new velocity vector $\mathbf{U}^{n,\ell}_{\text{new}}$ that satisfies for $i = 1, \ldots, I$, $j = 1, \ldots, J$,

$$(u_{\text{new}})^{n,\ell}_{i+1/2,j} = (u_{\text{old}})^{n,\ell}_{i+1/2,j} - a_{\ell,\ell-1}\Delta t(\partial_x \Delta P)_{i+1/2,j}, \tag{32}$$

$$(v_{\text{new}})^{n,\ell}_{i,j+1/2} = (v_{\text{old}})^{n,\ell}_{i,j+1/2} - a_{\ell,\ell-1}\Delta t(\partial_y \Delta P)_{i,j+1/2}. \tag{33}$$

Here $(\partial_x \Delta P)_{i+1/2,j}$ and $(\partial_y \Delta P)_{i,j+1/2}$ approximate the components of the gradient of the pressure correction ΔP on the staggered grid. These approximations are obtained using the fourth-order accurate formula (4). Setting the discrete divergence equal to zero, $D^{n,\ell}_{\text{new}} = 0$, and using (32), (33) obtain for $i = 1, \ldots, I$, $j = 1, \ldots, J$,

$$\begin{aligned}
&\frac{-(\partial_x \Delta P)_{i+\frac{3}{2},j} + 27\left[(\partial_x \Delta P)_{i+\frac{1}{2},j} - (\partial_x \Delta P)_{i-\frac{1}{2},j}\right] + (\partial_x \Delta P)_{i-\frac{3}{2},j}}{24\Delta x} \\
&\quad + \frac{-(\partial_y \Delta P)_{i,j+\frac{3}{2}} + 27\left[(\partial_y \Delta P)_{i,j+\frac{1}{2}} - (\partial_y \Delta P)_{i,j-\frac{1}{2}}\right] + (\partial_y \Delta P)_{i,j-\frac{3}{2}}}{24\Delta y} \\
&\quad + \left(\frac{\xi}{r_{i-\frac{1}{2}}}\right) \frac{-(\partial_x \Delta P)_{i+\frac{3}{2},j} + 9\left[(\partial_x \Delta P)_{i+\frac{1}{2},j} + (\partial_x \Delta P)_{i-\frac{1}{2},j}\right] - (\partial_x \Delta P)_{i-\frac{3}{2},j}}{16} \\
&\quad = \frac{1}{a_{\ell,\ell-1}\Delta t}(D_{\text{old}})^{n,\ell}_{i,j}.
\end{aligned}$$

Evaluation of the derivatives using (4) yields the following linear system that represents a discrete Poisson-type equation for the evaluation of pressure correction ΔP:

$$\left(\partial_x^2 + \partial_y^2 + \frac{\xi}{r_{i-\frac{1}{2}}}\delta_x\right)\Delta p_{i,j} = \frac{(D_{\text{old}})^{n,\ell}_{i,j}}{a_{\ell,\ell-1}\Delta t}. \tag{34}$$

Here $\partial_x^2 \Delta p_{i,j}$ and $\partial_y^2 \Delta p_{i,j}$ represent approximations of the second-order derivatives. These derivatives are evaluated from the following formula:

$$\left(\frac{\mathrm{d}^2 f}{\mathrm{d}x^2}\right)_i \approx \frac{f_{i+3} - 54f_{i+2} + 783f_{i+1} - 1460f_i + 783f_{i-1} - 54f_{i-2} + f_{i-3}}{576(\Delta x)^2}. \tag{35}$$

As a result, the left-hand-side stencil in (34) involves thirteen points, the pressure point plus six points in each direction. Taylor series analysis shows that (35) is a fourth-order accurate, collocated approximation of the second-derivative. Similarly, $\delta_x \Delta p_{i,j}$ represents an approximation of the first-order derivative that appears in cylindrical coordinates. The derivative is evaluated from the following formula:

$$\left(\frac{\mathrm{d}f}{\mathrm{d}x}\right)_i \approx \frac{f_{i+3} - 36f_{i+2} + 261f_{i+1} - 261f_{i-1} + 36f_{i-2} - f_{i-3}}{384\Delta x}. \tag{36}$$

Taylor series analysis shows again that (36) is a fourth-order accurate, collocated approximation of the first-order derivative.

Similarly to the pressure correction method of Section 5.1, an analogous iteration for the Poisson-type equation defining the pressure updates is established in order to enforce global incompressibility. The update of the velocity vector $\mathbf{U}^{n,\ell}_{\text{new}}$, computed by (32) and (33) using the pressure correction $\Delta P^{n,\ell}$, again violates local incompressibility. Therefore, after $\mathbf{U}^{n,\ell}_{\text{new}}$ is computed, the velocity vector and the pressure are updated as $\mathbf{U}^{n,\ell}_{\text{old}} = \mathbf{U}^{n,\ell}_{\text{new}}$ and $P^{n,\ell}_{\text{old}} = P^{n,\ell}_{\text{new}}$, and the iteration is repeated until $(D_{\text{old}})^{n,\ell}_{i,j}$ approaches zero for all cells simultaneously within some preset tolerance. The imposition of the incompressibility condition to $\mathbf{U}^{n+1}$ is straightforward.

5.2.2. Boundary conditions

The finite-difference scheme resulting from (34) using (35) and (36), represents a fourth-order accurate finite-difference approximation of a Poisson-type equation. Appropriate boundary conditions for ΔP on the boundary of the physical domain are needed to define properly a solution of (34). The boundary conditions for the pressure correction ΔP and pressure are directly associated. For a Dirichlet boundary condition, $p = \text{const}$, therefore $\Delta P = 0$. For example if the pressure is constant along the left (vertical) boundary, set $(\Delta p)_{3+1/2,j} = 0$, for $j = 1, \ldots, J$, and obtain the pressure corrections at the cell centers as follows:

$$(\Delta p)_{6,j} = -(\Delta p)_{1,j}, \quad (\Delta p)_{5,j} = -(\Delta p)_{2,j}, \quad (\Delta p)_{4,j} = -(\Delta p)_{3,j}. \tag{37}$$

The pressure correction ΔP^{p} of the previous iteration is assumed to satisfy (37). Therefore the appropriate components of the pressure correction ΔP^{c} of the current iteration, are defined as follows:

$$(\Delta p^{\mathrm{c}})_{6,j} = -(\Delta p^{\mathrm{p}})_{6,j}, \quad (\Delta p^{\mathrm{c}})_{5,j} = -(\Delta p^{\mathrm{p}})_{5,j}, \quad (\Delta p^{\mathrm{c}})_{4,j} = -(\Delta p^{\mathrm{p}})_{4,j}. \tag{38}$$

For a Neumann boundary condition, $\partial p/\partial n = 0$, that implies $\partial(\Delta P)/\partial n = 0$, with n the outward normal on the boundary. Implementation of Neumann boundary conditions require significant bookkeeping. It was found, however, that application of modified Dirichlet–type boundary conditions for the iterations with the Poisson equation is sufficiently accurate. The unknown values occurring from the discretization of the directional derivative, are obtained, as for the Dirichlet case, from the associated values computed by the previous iteration. For example a Neumann condition $(\partial(\Delta p)/\partial x)_{3+1/2,j} = 0$, for $j = 1, \ldots, J$, is imposed numerically by setting

$$(\Delta p)_{6,j} = (\Delta p)_{1,j}, \quad (\Delta p)_{5,j} = (\Delta p)_{2,j}, \quad (\Delta p)_{4,j} = (\Delta p)_{3,j}. \tag{39}$$

The pressure correction ΔP^{p} of the previous iteration now satisfies (39), and the pressure correction ΔP^{c} of the current iteration is obtained by the following Dirichlet condition

$$(\Delta p^{\mathrm{c}})_{6,j} = (\Delta p^{\mathrm{p}})_{6,j}, \quad (\Delta p^{\mathrm{c}})_{5,j} = (\Delta p^{\mathrm{p}})_{5,j}, \quad (\Delta p^{\mathrm{c}})_{4,j} = (\Delta p^{\mathrm{p}})_{4,j}. \tag{40}$$

5.2.3. Narrow stencil, fourth-order accurate Poisson update

In this section an alternative approach is used to derive the discrete Poisson-type equation. Considering the semidiscrete (discrete only in time) form of the momentum equations (2) and using the divergence free condition on the pressure points of the staggered grid obtain an equation at pressure points. Using the continuity equation, this equation is reduced to a Poisson-type equation for pressure correction. This Poisson-type equation is then discretized with a fourth-order accurate method. This approach results into a narrower stencil for the discrete Poisson-type equation.

Consider the momentum equations (2) written in a compact form, analogous to (18), as

$$\left(\frac{\mathrm{d}\mathbf{u}(t)}{\mathrm{d}t}\right) = -\nabla p + \mathbf{A}(\mathbf{u}; t). \tag{41}$$

Regarding the above equations as a system of ordinary differential equations, we discretize in time using the fourth-order Runge–Kutta method. Let $\mathbf{u}^{n,\ell}_{\mathrm{old}}$ be the discrete in time (but continuous in space) solution of (41) obtained from the ℓth stage of the Runge–Kutta method. Set also $p^{n,\ell}_{\mathrm{old}} = p^{n,\ell-1}$, where $p^{n,\ell-1}$ is the discrete in time pressure from the previous intermediate stage. The corrected pressure at the ℓth stage is

$$p^{n,\ell}_{\mathrm{new}} = p^{n,\ell}_{\mathrm{old}} + \Delta p.$$

An updated velocity $\mathbf{u}^{n,\ell}_{\mathrm{new}}$ is defined by

$$\mathbf{u}^{n,\ell}_{\mathrm{new}} = \mathbf{u}^{n} + a_{\ell,\ell-1}\Delta t(-\nabla p^{n,\ell}_{\mathrm{new}} + \mathbf{A}(\mathbf{u}^{n,\ell}_{\mathrm{old}}; t)), \tag{42}$$

which is equivalent to

$$\mathbf{u}^{n,\ell}_{\mathrm{new}} = \mathbf{u}^{n,\ell}_{\mathrm{old}} - a_{\ell,\ell-1}\Delta t\nabla(\Delta p).$$

Application of the continuity equation (1) to $\mathbf{u}^{n,\ell}_{\mathrm{new}}$ implies that at the centers of the cells $C_{i,j}$, $i = 1, \ldots, I$, $j = 1, \ldots, J$,

$$\nabla \cdot \mathbf{u}_{\text{new}}^{n,\ell} + \frac{\xi}{r}\mathbf{u}_{\text{new}}^{n,\ell} = 0. \tag{43}$$

Substituting (42) in (43) obtain the following Poisson-type equation, valid at the cell centers, for $i = 1, \ldots, I$, $j = 1, \ldots, J$:

$$\left(\frac{\partial^2(\Delta p)}{\partial x^2}\right)_{i,j} + \left(\frac{\partial^2(\Delta p)}{\partial y^2}\right)_{i,j} + \frac{\xi}{r_{i-\frac{1}{2}}}\left(\frac{\partial(\Delta p)}{\partial y}\right)_{i,j} = \frac{1}{a_{\ell,\ell-1}\Delta t}\left[(\nabla \cdot \mathbf{u}_{\text{old}}^{n,\ell})_{i,j} + \frac{\xi}{r_{i-\frac{1}{2}}}(\mathbf{u}_{\text{old}}^{n,\ell})_{i,j}\right]. \tag{44}$$

Second-order derivatives at the left-hand side of (44) are approximated to fourth-order accuracy using (5), which has a five-points wide stencil. The resulting discrete Poisson equation has a nine-points wide stencil, the pressure point plus four points from each direction. The values of $\mathbf{u}_{\text{old}}^{n,\ell}$ and its divergence at the cell centers appearing at the right-hand side of (44), are computed using (3) and (6), respectively. These values on the staggered grid are known from the previous iteration.

The Neumann boundary condition for this stencil, in analogy with (40), is applied as

$$(\Delta p^{\text{c}})_{2,j} = (\Delta p^{\text{p}})_{5,j}, \quad (\Delta p^{\text{c}})_{3,j} = (\Delta p^{\text{p}})_{4,j}.$$

The homogeneous Dirichlet boundary condition is as

$$(\Delta p^{\text{c}})_{2,j} = -(\Delta p^{\text{p}})_{5,j}, \quad (\Delta p^{\text{c}})_{3,j} = -(\Delta p^{\text{p}})_{4,j}.$$

The stencil constructed by the discretization of (44) is more efficient than that of (34), since it involves fewer points and requires fewer operations. It is also more robust and numerical experiments demonstrated that the same solution is obtained from (34) and (44).

5.3. The GMRES method

Let I and J be the maximum values attained by the indices i and j, respectively. Then, the linear systems defined by (34), or (44), have a block matrix structure, with $J \times J$ blocks, each of dimension $I \times I$.

The matrix associated with (34) has a seven diagonal block structure. The diagonal blocks have seven diagonals each, while the off-diagonal blocks are diagonal. The matrix associated with (44) has a five diagonal block structure, with diagonal blocks of five diagonals, and off-diagonal blocks are diagonal.

For the solution of these linear systems a preconditioned GMRES method [33] is used. The algorithm uses reverse communication for the definition of matrix–vector multiplications by the user according to the storage mode employed. Presently, the compressed sparse row (CSR) storage [32] is used. For the preconditioning, the ILUT preconditioner, based on an incomplete LU factorization with a dual truncation mechanism, is applied. Truncation is performed by dropping any elements in L and U, smaller in absolute value than some tolerance, relative to the absolute value of diagonal elements in U. A fill-in parameter control keeps the maximum number of elements in each column and row, to be kept according to their magnitude.

6. Numerical results

In this section tests of the proposed high-order method are carried out for simple, steady as well as time-dependent flow problems with exact solutions. These tests include characteristic flow cases such as cavity and pipe flow. The numerical solutions obtained compare very well with the analytic solution. Furthermore, high order of accuracy is achieved, as is demonstrated by the reduction of error with grid refinement. These results were obtained by both the second- and fourth-order methods without using any upwind. Fourth-order accuracy is obtained either by using the explicit or the compact fourth-order differencing scheme. The incompressibility constraint is enforced either by the local pressure correction or the Poisson-type equation method. In order to retain high-order accuracy in time and avoid nonlinear instabilities in the numerical experiments we perform the computations with CFL numbers below the stability limit of the Runge–Kutta method. It was found that the divergence free condition on the discrete level is enforced to machine accuracy (less than 10^{-16}) for all methods.

6.1. Circular pipe flow

The first example, used to validate the proposed fourth-order accurate method but not shown here, is the computation of the flow in a circular pipe driven by a constant pressure gradient. The pipe has a length $L = 10$ units and a radius $R = 1$. The radial direction r is taken along the x-axis. The y-axis, which is the axis of symmetry, is also the vertical left boundary where $u = \partial v/\partial n = \partial p/\partial n = 0$ holds. On the right vertical boundary placed at $r = 1$, $u = v = \partial p/\partial n = 0$ are used as boundary conditions. At the inflow and outflow extrapolated conditions for u and v are used while the pressure is taken as $p_{in} = 2$ and $p_{out} = 1$, respectively. The axial velocities obtained by the fourth- and second-order methods on a series of grids with 21×91, 31×136, 41×181 and 81×361 points, were compared with the exact solution $v(r) = Re(p_{in} - p_{out})(R^2 - r^2)/(4L)$. The steady state numerical solutions were obtained by iterating the numerical solution up to $T = 15$ time units. The results obtained from both methods practically coincided and their deviation from the exact solution was negligible. Numerical solutions obtained by all variants of the fourth-order method proposed, i.e. using compact schemes or explicit fourth-order finite-difference formulas to discretize the viscous and diffusive terms and Poisson-type pressure equations or cell-based pressure corrections for imposing incompressibility, also coincided.

6.2. Driven cavity flow

A classical steady-state flow problem with well defined boundary conditions is the unit square driven-cavity flow. The pressure for all walls is obtained by assuming zero normal pressure gradient $\partial p/\partial n = 0$. Non slip condition is imposed for all walls except for the top wall where $u = 1$ and $v = 0$. This problem is often used [15,1] 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. The boundary conditions were discretized to fourth-order accuracy, by appropriately matching the staggered grid functional values within the three-cells wide fictitious layer. Good asymptotic convergence was not achieved for very coarse meshes due to the poor resolution of the near wall flow. Well converged numerical solutions were produced by the high-order method at a Reynolds number $Re = 100$ on a series of meshes. These meshes are an 111×111, an 121×121, an 131×131, an 141×141, an 151×151, an 161×161, an 181×181, an 191×191 an 201×201, and an 221×221 point uniform grids. For comparison, numerical solutions were also obtained by the second-order accurate method of Ref. [13], on the same grids. In addition, a numerical solution was computed by the method of Ref. [13] on a very fine 301×301 point uniform grid. All the computations were performed using double precision.

It was assumed that the 301×301 point very fine grid solution obtained by the second-order accurate method of Ref. [13], achieves a negligible error to the exact solution in the L^2 norm. Therefore, this numerical solution was used to make a posteriori error estimates to establish the fourth-order accuracy of the proposed method. Computed solutions are compared with the numerical solution of Ref. [9] that is used widely for comparisons in the literature. For all grids, the divergence on the discrete level was enforced to machine zero (approximately 10^{-16}) and did not change with grid refinement.

A comparison of the u-velocities computed by the fourth-order accurate method on the 201×201 and 221×221 point grids with those computed by the second-order accurate method on the 221×221 and 301×301 point grids and the results of Ref. [9] is shown in Fig. 2. An analogous comparison for the computed v-velocities is shown in Fig. 3. Good agreement is obtained for the fourth-order accurate solution with the solution of Ref. [9]. Second-order accurate solutions on finer grids show larger deviations than the fourth-order accurate solutions on coarser grids. The velocity field and the contours of the velocity magnitude $\sqrt{u^2 + v^2}$ computed by the fourth-order accurate method on the 221×221 point grid are shown in Fig. 4.

The numerical solutions shown in Figs. 2–4 were obtained after integrating up to $T = 10$ time units. Since no difference was found by integrating up to $T = 15$ or $T = 20$ time units, the steady-state solution was safely reached for $T = 10$ time units. This is further confirmed by Fig. 5, where the L^2 norm of the residual $\mathbf{R}$ (see Eq. (18)) of the numerical solutions computed with the 221×221 point grid (time step equal to 1/2500) and the fourth- and second-order accurate methods, and the numerical solution computed with the 301×301 point grid (time step 1/4500) and the second-order accurate method are shown to level off (i.e. reduce to practically zero, that is less than 10^{-10}) after integrating up even $T = 5$ time units. The time step for the 221×221 point

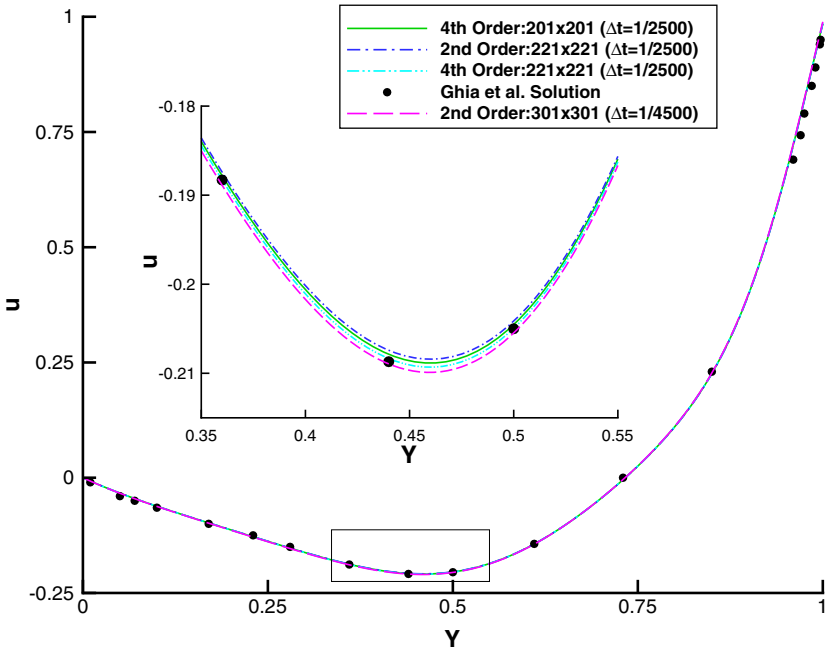

Fig. 2. Comparison of the u-velocity computed by the fourth- and second-order accurate methods with a reference solution by Ghia et al. [9]; driven cavity flow ($Re = 100$).

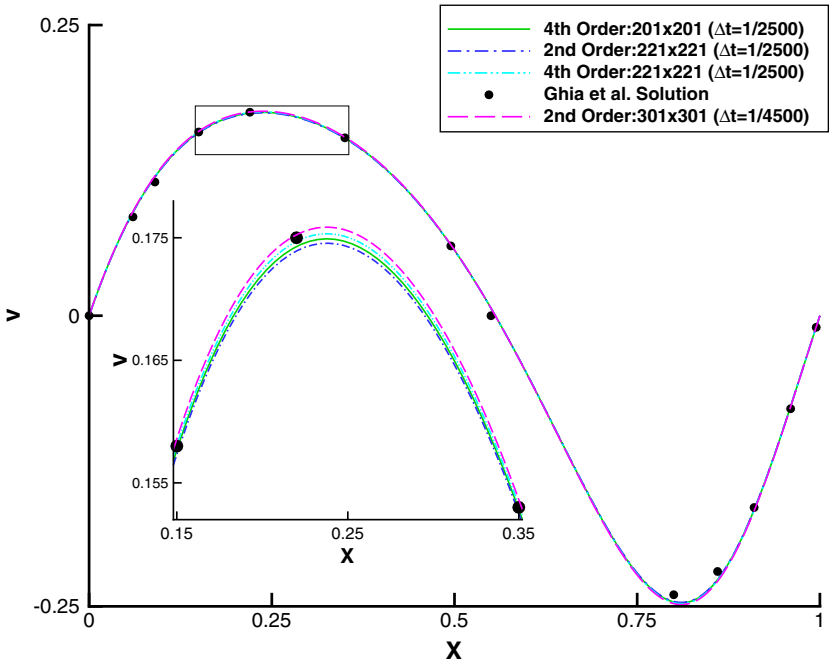

Fig. 3. Comparison of the v-velocity computed by the fourth- and second-order accurate methods with a reference solution by Ghia et al. [9]; driven cavity flow ($Re = 100$).

grid computations with the fourth and second was close to the CFL stability limit of the Runge–Kutta time marching method.

For the cavity flow, a posteriori estimations of the spatial discretization error using the L^2 norm of the error are shown in Fig. 6. A very fine grid solution computed on a 301×301 point grid by the established second-order

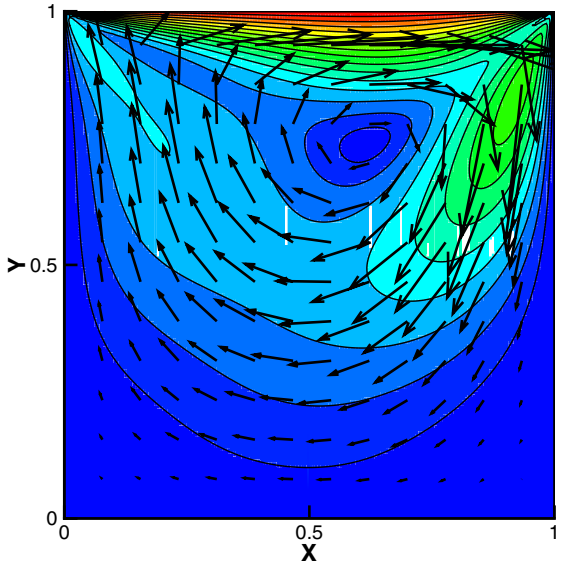

Fig. 4. Velocity field and velocity magnitude contours computed by the fourth-order method on the 221×221 point grid; driven cavity flow ($Re = 100$).

accurate method of Ref. [13] is used as reference solution u_R. It is assumed that this reference solution has negligible error compared to the 'exact' solution u_E in the L^2 norm, i.e. $\|u_R - u_E\|_{L^2}$ is very small. The L^2 error norms of Fig. 6 are defined as $\|u_N - u_R\|_{L^2}$, where u_N denotes the fourth-order accurate solutions obtained on the $N \times N$ point grids with $N = 111, 121, 131, 141, 151, 161, 181, 191, 201$, and 221. The triangular inequality $\|u_N - u_E\|_{L^2} \leqslant \|u_N - u_R\|_{L^2} + \|u_R - u_E\|_{L^2}$, shows that a good approximation of the error $\|u_N - u_E\|_{L^2}$ of the u_N solutions can be obtained from $\|u_N - u_R\|_{L^2}$. Therefore, grid convergence of the numerical solution of Fig. 6 is demonstrated with the decay of the L^2 norm $\|u_N - u_R\|_{L^2}$. It is evident that the observed order of accuracy of the proposed fourth-order method is fair.

The numerical solutions obtained by all variants of the proposed fourth-order accurate in space and time method coincide. For the Reynolds number $Re = 100$, the variants using the local pressure correction update required less computing time than those based on the solution of the Poisson-type equation for the pressure correction. Enforcement of the incompressibility condition to machine accuracy (i.e. $\nabla \cdot \mathbf{u} \approx 10^{-16}$) was achieved for all cases. The number of iterations of the local pressure correction per time step of the Runge–Kutta method, for the 121×121 point grid solution for example, varies as follows: For the initial time iterations with the Runge–Kutta method, where the transients are removed, the number of correction sweeps within each stage is different. For the second and third stages is about 400 and for the fourth stage is about 50. Once the solution passes through the transient stage the number of correction sweeps diminishes considerably. For the second and third stages of the Runge–Kutta method the iterations are about 100 and for the fourth stage about 15. Just before the steady-stage regime, the number of correction sweeps for the second and third stages of the Runge–Kutta method is between 10 and 20 and for the fourth stage to less than 10. When the steady-state is reached the number of iterations required drops to about 10 for all stages. On the other hand, the number of correction sweeps per time step, required by the Poisson-equation pressure correction method to obtain the 121×121 point grid solution, is smaller compared to that for the local pressure correction. However, since the computing time for solving numerically the Poisson-type equation is larger, the total computing time required by the local pressure correction method to enforce incompressibility is smaller compared to that of the Poisson-equation global pressure correction method. The same comparative behavior concerning the number of iterations per time step was observed for $Re = 100$, for finer or coarser grids.

The observation that the number of iterations for the Poisson-equation pressure correction method was in all cases less than the number of iterations required by the local pressure correction method prompted further experiments at higher Reynolds numbers. Therefore, numerical solutions were obtained for $Re = 500$, 1000,

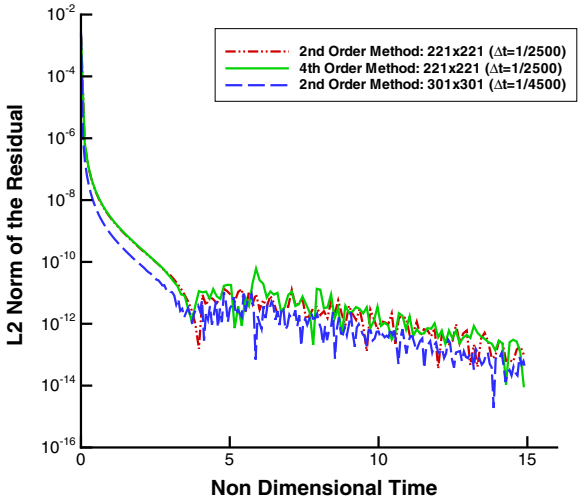

Fig. 5. Comparison of the L^2 norm of the residuals **R** of the numerical solution for the fourth- (on the 221×221 points grid) and the second-order (on the 221×221, 301×301 point grids) accurate methods; driven cavity flow ($Re = 100$).

2000 and 10,000 on a 101×101 and a 151×151 point grids. The important outcome of this study was that while the Reynolds number increases, the ratio of the number of iterations of the Poisson-equation pressure correction method to that of the local pressure correction method, required to enforce incompressibility at each time step of the Runge–Kutta method, decreases. As a result, it appears that with the progressive increase of the Reynolds number, use of the Poisson-equation global pressure correction method becomes more attractive alternative in terms of efficiency. This is an encouraging conclusion for applications to 3D high Reynolds number flows, where computing time becomes an important issue.

6.3. Oseen vortex decay

The decay of an ideal vortex is an unsteady flow problem with an exact solution [26]. This problem is of interest to numerical simulations of trailing vortices, (LES), and (DES) simulations. The initial velocity distribution of the Oseen vortex is given by

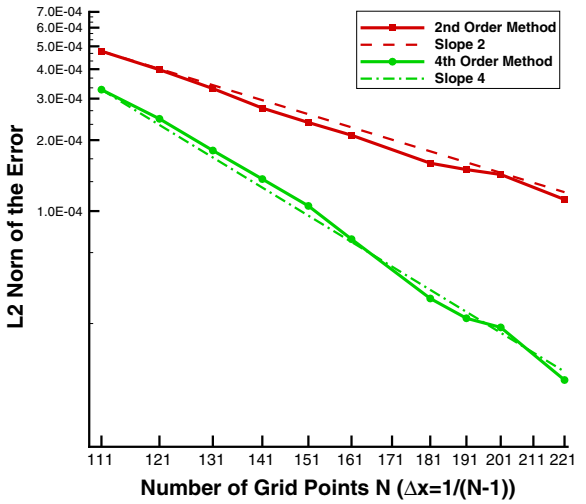

Fig. 6. Grid convergence shown by the L^2 norms of the error obtained by the fourth- and second-order accurate methods; driven cavity flow ($Re = 100$).

$$v_\theta(r, t=0) = \frac{\Gamma}{2\pi r}, \tag{45}$$

where Γ defines the strength of the vortex and r is the distance from the origin, and the initial pressure distribution is $p = \text{const}$. This vortex decays under the action of viscous forces and the velocity at time t is given by the following exact solution of Ref. [26]:

$$v_\theta(r, t) = \frac{\Gamma}{2\pi r}\left[1 - \exp\left(-\frac{r^2 Re}{4t}\right)\right]. \tag{46}$$

The time-dependent flow with initial condition given by (45) was computed for $\Gamma = 5$. In Figs. 7 and 8, the numerical solutions obtained on equally spaced, Cartesian grids with 51×51 and 101×101 points, respectively, using the fourth- and the second-order accurate methods were compared with the exact solution (46) for time $T = 4$. For both figures, a small time step of 1/1000, well below the CFL stability limit of the Runge–Kutta method, was used to ensure that time-accuracy errors are negligible compared to spatial discretization errors. Both figures show the superior performance of the fourth-order accurate method compared to that of the second-order accurate method.

A comparison of the L^2 norm of the spatial error at $T = 1$ obtained by the fourth- and second-order accurate methods is shown in Fig. 9. It is evident that the fourth-order accurate method yields the expected fourth-order spatial accuracy. A comparison of the L^2 norm of the temporal error at $T = 1$ obtained by the fourth- and second-order accurate methods is shown in Fig. 10. The fourth-order method again yields the expected fourth-order temporal accuracy. In Fig. 11 numerical results obtained by the fourth- and second-order accurate methods computed on a 1001×1001 point grid (and a time step of 1/1000) for a very large Reynolds number ($Re = 10^{10}$) are compared with the exact solution.

In Fig. 12, the kinetic energy defined as $\mathbf{E}(t) = \int_D u^2(r, \theta, t) r \, dr \, d\theta$ is plotted for $Re = 1$, for the exact solution and the fourth-order accurate numerical solution computed on the 101×101 point grid using the explicit finite-differences of Section 3.2.1 to discretize in space the momentum equations. The kinetic energy is plotted in the same figure for the inviscid limit $Re = 10^{10}$, for the exact solution and the fourth-order accurate numerical solution on the 1001×1001 point grid. In both cases, the energy conservation of the proposed fourth-order accurate numerical method is evident. Fourth-order accurate numerical solutions computed on the same grid and the compact schemes of Section 3.2.2 for the space discretization, exhibited the same behavior. For the high Reynolds number solutions the numerical mesh is shifted to avoid having a grid point at the origin

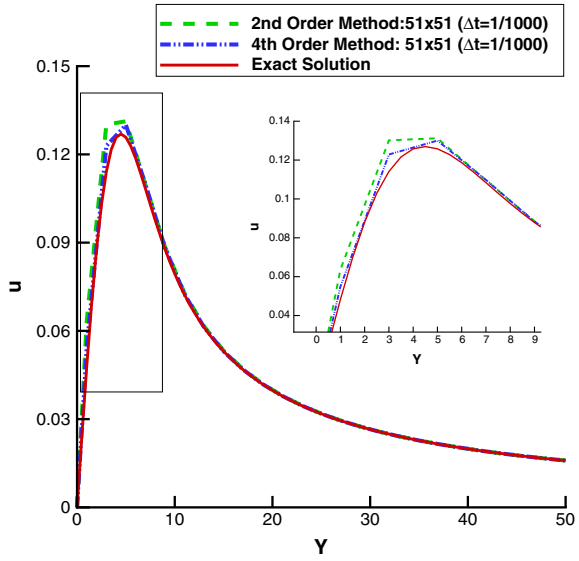

Fig. 7. Comparison of u-velocity computed by the fourth- and second-order accurate methods on the 51×51 point grid with the exact solution at $T = 4$; Oseen vortex decay.

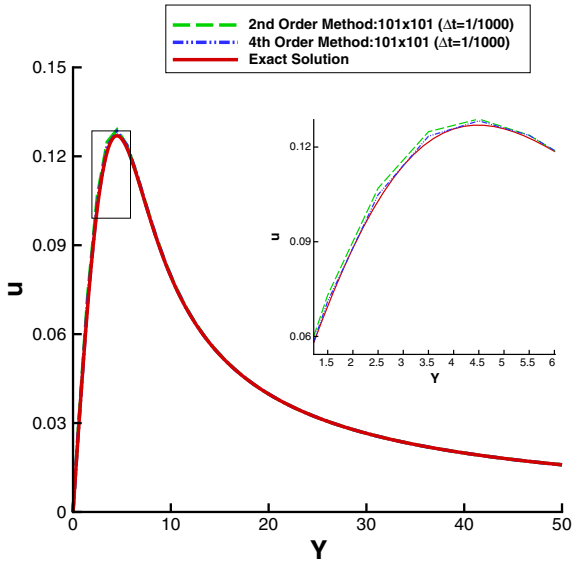

Fig. 8. Comparison of u-velocity computed by the fourth- and second-order accurate methods on the 101×101 point grid with the exact solution at $T = 4$; Oseen vortex decay.

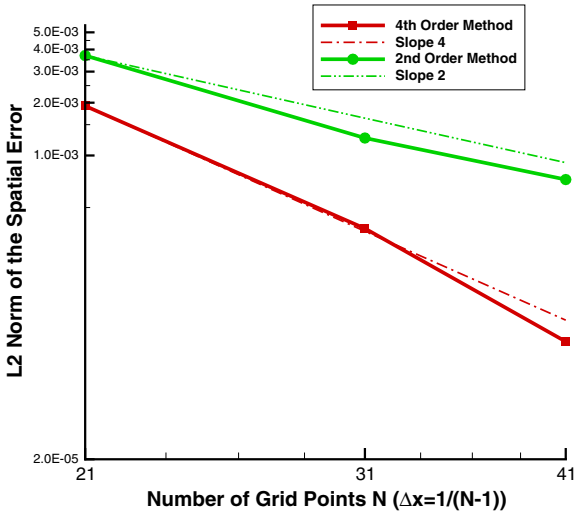

Fig. 9. Comparison of the L^2 norm of the spatial error obtained by the fourth- and second-order accurate methods at $T = 1$; Oseen vortex decay.

where the velocity is very large (see Eq. (46)). For this reason, the principal value of the integral is computed after excluding a circle with center the origin and radius the half mesh length.

All the results were computed using double precision. Note that all numerical solutions obtained by all variants of the fourth-order accurate method proposed coincided.

6.4. Stokes oscillating plate

The flow over an infinite oscillating plate (Stokes solution) is an unsteady incompressible flow problem with an exact solution [26]. The flow over the plate is established after the plate starts an oscillatory motion (on the plane $y = 0$) with speed

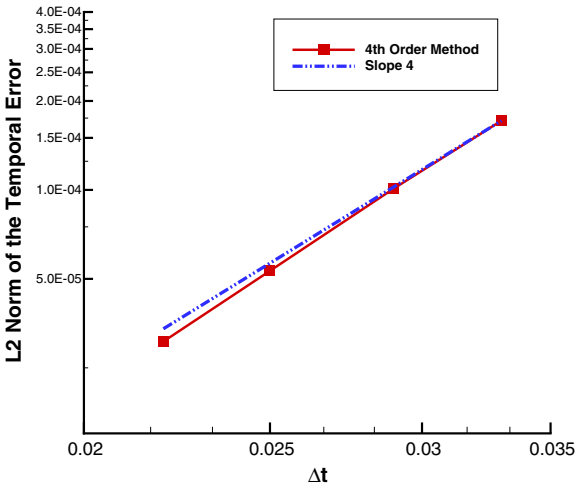

Fig. 10. Comparison of the L^2 norm of the temporal error obtained by the fourth-order accurate in space method at $T = 1$; Oseen vortex decay.

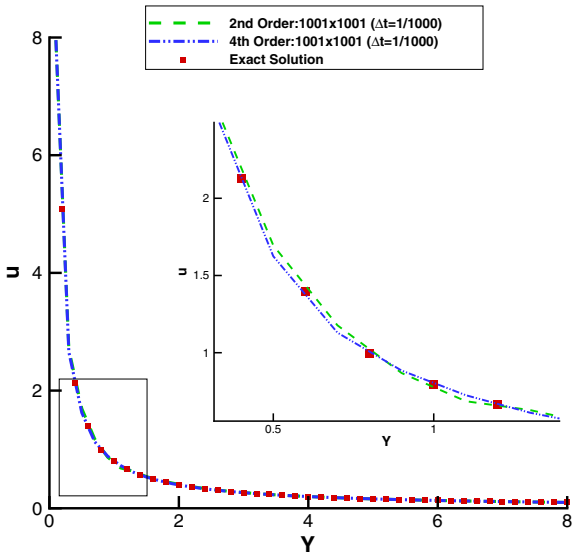

Fig. 11. Comparison of u-velocity computed by the fourth- and second-order accurate methods on the 1001×1001 point grid with the exact solution for $Re = 10^{10}$, at $T = 4$; Oseen vortex decay.

$$u(x, 0, t) = u_0 \sin \Omega t. \tag{47}$$

The exact solution gives the time varying velocity as

$$u(x, y, t) = \exp\left(-y\sqrt{\frac{\Omega Re}{2}}\right) \sin\left(\Omega t - y\sqrt{\frac{\Omega Re}{2}}\right). \tag{48}$$

For the computations, periodic boundary conditions were imposed in the streamwise direction, since the solution (48) is independent of x. On the plate $v = \partial p/\partial n = 0$ was considered. The following values of the parameters $u_0 = 1$ and $\Omega = 2\pi$ are used. The resulting oscillation period is $T = 1$. At the top boundary of the domain, which was located eight units away from the solid wall, the flow velocity and pressure were taken equal to zero. For this computation, a series of grids with 21, 31, 41, 51, 71, 91 and 101 points in the normal to the

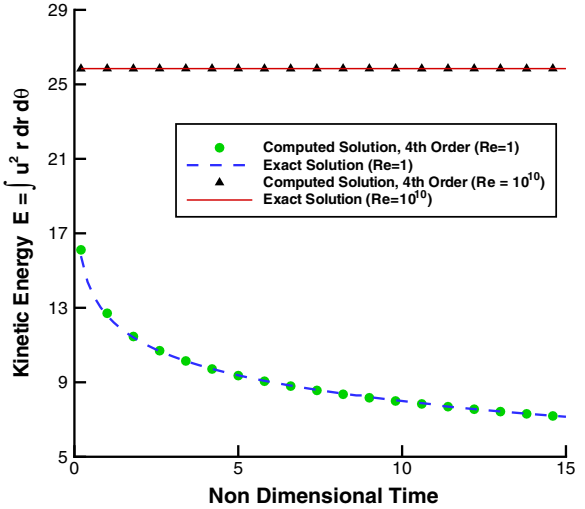

Fig. 12. Comparison of the kinetic energy **E** as a function of time obtained by the u-velocity computed by the fourth-order accurate method for $Re = 1$ (on the 101×101 point grid) and $Re = 10^{10}$ (on the 1001×1001 point grid), and the corresponding exact solution; Oseen vortex decay.

solid wall direction and 6 points in the streamwise direction were used. Numerical solutions were computed by the fourth- and second-order accurate methods after iterating for 30 periods until a time periodic solution was achieved.

Comparisons of the computed solution, from both numerical methods, (at $T = 1/4$ after a time periodic solution was achieved) with the exact solution (48), are shown in Fig. 13 for the 6×101 point grid. The agreement with the exact solution is very good and the better resolution of the fourth-order method is evident. The variation of the L^2 norm of the error obtained by both methods with grid refinement are plotted in Fig. 14. It is evident that grid convergence is achieved and the proposed fourth-order accurate method retains fourth order of accuracy. It was found that the variant of the fourth-order method with compact schemes for the

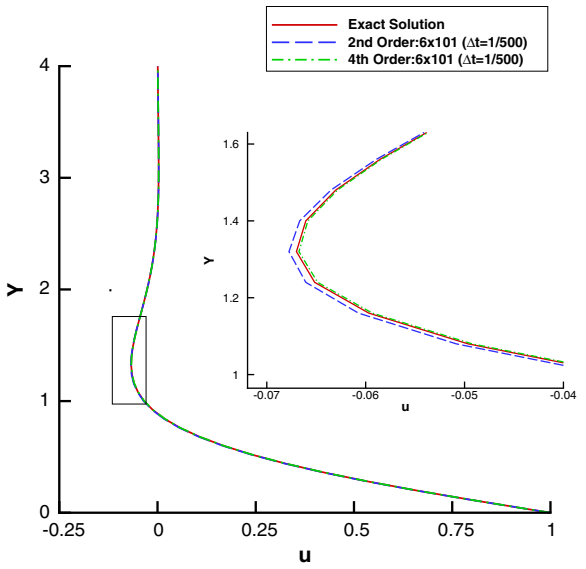

Fig. 13. Comparison of u-velocity computed by the fourth- and second-order accurate methods on the 6×51 point grid with the exact solution at $T = 1/4$; Stokes oscillating plate.

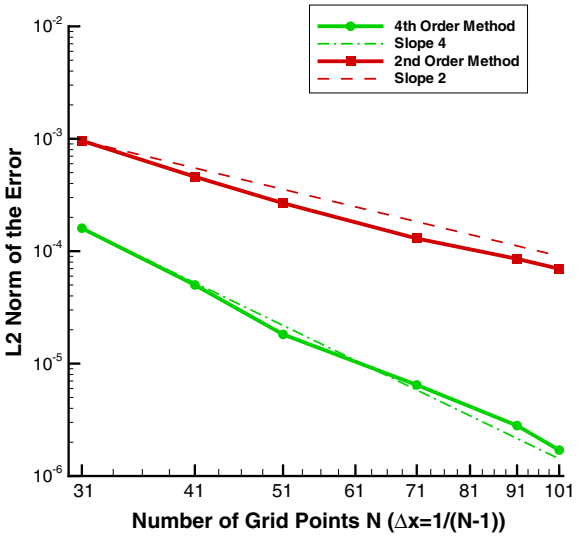

Fig. 14. Grid convergence shown by the L^2 norm of the errors obtained by the fourth- and second-order accurate methods at $T = 1/4$; Stokes oscillating plate.

evaluation of the derivatives, requires a slightly smaller time step to converge to the same numerical solution. All the results were obtained using double precision arithmetic.

6.5. Pulsatile pipe flow

Flow in a circular pipe established by an oscillating pressure gradient is an unsteady incompressible flow problem with an exact solution [35]. In this case $p(r, 0, t) = p_{\text{out}} + L\cos\omega t$, with p_{out} a given constant and L the length of the pipe. The exact solution [20] for the spatial and temporal variation of the axial velocity is given by

$$v(r,t) = \Re\left\{\frac{R^2}{\mathrm{i}Re\alpha^2}\left[1 - \frac{J_0(\alpha\zeta i^{3/2})}{J_0(\alpha i^{3/2})}\right]\mathrm{e}^{\mathrm{i}\omega t}\right\}, \tag{49}$$

where $\zeta = r/R$, $\alpha = R\sqrt{\omega Re}$ is the Womersley number, R is the radius of the circular pipe, $\Re$ denotes the real part, and J_0 is the Bessel function of the first kind of order zero and complex argument. Pulsatile flow in used in physiological flows to describe blood flow in arteries [20].

For the computations $\omega = 4$ ($\alpha = 6.3245$) was selected. For this case the unsteady motion has a period $T = \pi/2$. Furthermore, values of $p_{\text{out}} = 1$, $R = 1$ and $L = 5$ were used. Time periodic response of the numerical solutions was obtained after iterating for 10 periods ($t = 5\pi$). Numerical solutions obtained by the fourth- and the second-order accurate methods on a series of grids with 21×51, 31×76 and 41×101 points, are compared at $T = \pi/4$, after time-periodic response was reached. A comparison of the fourth- and the second-order accurate solutions for the 41×101 point grid, with the exact solution of (49), is shown in Fig. 15. Fig. 16 shows that the amplitude and phase errors, over one period, of the fourth-order solution are negligible. In this figure only one half of the period is shown because the other half is a repetition.

For the pulsatile flow calculations the effect of the approximate inflow and outflow boundary data specification degrades the overall accuracy of the numerical solution. As a result grid convergence in the L^2 norm of the error was not achieved because the order of accuracy of the numerical solution degrades due to time lag of the extrapolation at the inflow and outflow from the interior. However, it was observed that the fourth-order method simply achieves smaller L^2 error levels compared to the second-order solution. All variants of the fourth-order method showed the same convergence behavior. All results were obtained using double precision arithmetic.

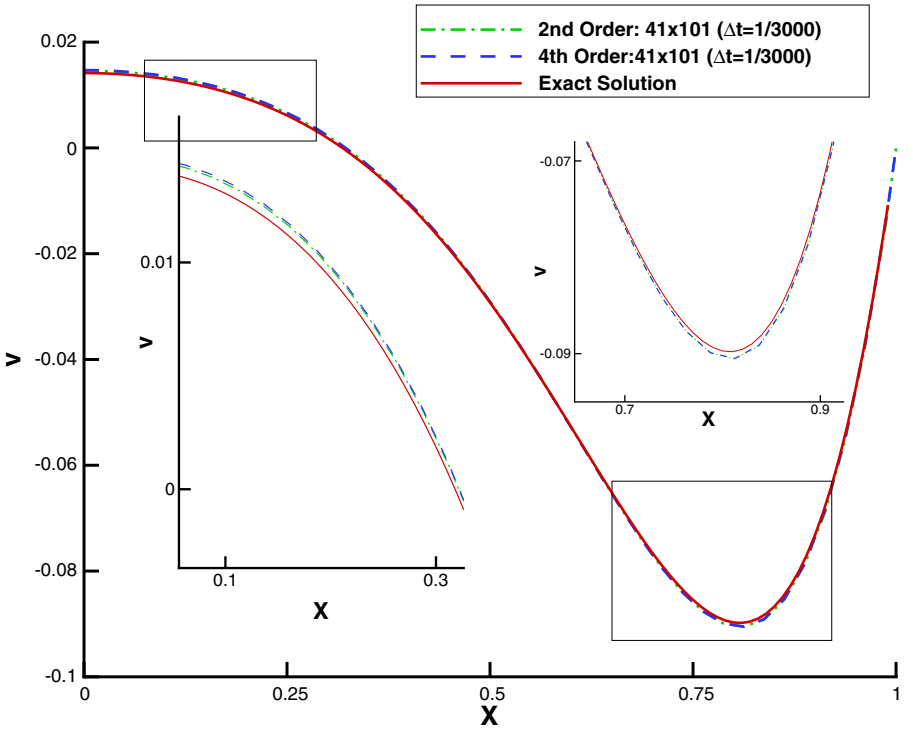

Fig. 15. Comparison of v-velocity component computed by the fourth- and second-order accurate methods on the 41×101 point grid with the exact solution at $T = \pi/4$; pulsatile flow with $\alpha = 6.3245$ and $\omega = 4$.

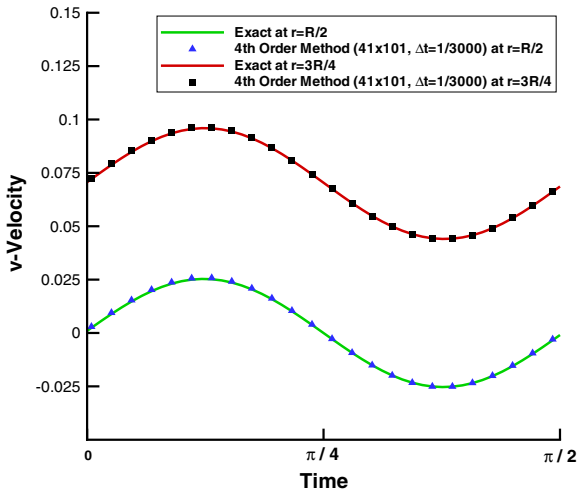

Fig. 16. Comparison of the computed and exact axial velocity variation over half period at $r = R/2$ and $r = 3R/4$; pulsatile flow with $\alpha = 6.3245$ and $\omega = 4$.

7. Conclusions

A fourth-order accurate method for the numerical solution of the incompressible Navier–Stokes equations has been developed. The proposed method uses explicit or compact fourth-order accurate schemes for the discretization of the convective and viscous fluxes. Incompressibility is imposed iteratively by two methods. The

first method computes a pressure update by consecutive pressure corrections for all cells followed by local velocity corrections. The second method solves a global Poisson-type equation to compute pressure updates for all cells simultaneously and recomputes the velocity globally. In both cases, the derivatives in the continuity equation that is used for pressure corrections, are discretized to fourth-order accuracy. The proposed fourth-order method is compared with exact solutions in several incompressible fluid flow problems. These problems include computations of steady-state flows such as circular pipe and a driven cavity flows. Furthermore, unsteady flow solutions for the decay of the Oseen vortex, the flow above an oscillating plate and the pulsatile flow in a pipe are computed. The comparisons show that the fourth-order accurate method performs better than the second-order accurate method in all cases. Furthermore, the proposed method achieves fourth-order accuracy in space and time. The local pressure correction method performs better (i.e. enforces the incompressibility condition to machine zero in less computational time) for low Reynolds number flows. However, it appears that the Poisson equation, global pressure correction method becomes an attractive alternative as the Reynolds number became larger.

References

[1] F. Auteri, N. Parolini, L. Quartapelle, Numerical investigation on the stability of singular driven cavity flow, J. Comput. Phys. 183 (2002) 1–25.
[2] J.C. Butcher, The Numerical Analysis of Ordinary Differential Equations: Runge–Kutta Methods and General Linear Methods, John Wiley, Chichester, 1987.
[3] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods in Fluid Dynamics, Springer-Verlag, New York, 1988.
[4] A.J. Chorin, A numerical method for solving incompressible, viscous flow problems, J. Comput. Phys. 2 (1) (1967) 12–26.
[5] A.J. Chorin, Numerical solution of the Navier–Stokes equations, Math. Comput. 22 (1968) 745–762.
[6] J.A. Ekaterinaris, Numerical simulation of incompressible two-blade rotor flow fields, AIAA J. Propuls. Power 14 (3) (1998) 367–374.
[7] D.V. Gaitonde, M.R. Visbal, high-order schemes for Navier–Stokes equations: algorithms and implementation into FDL3DI, Air Vehicles Directorate, Air Force Research Laboratory, Write-Patterson Air Force Base, Ohio 45433-7913, AFRL-VA-WP-TR-1998-3060, August 1998.
[8] R.A. Gentry, R.E. Martin, B.J. Daly, An Eulerian differencing method for unsteady compressible flow problems, J. Comput. Phys. 1 (1966) 87–118.
[9] U. Ghia, K.N. Ghia, C.T. Shin, High-*Re* solutions for incompressible flow using the Navier–Stokes equations and a multigrid method, J. Comput. Phys. 48 (1982) 387–411.
[10] F.E. Ham, F.S. Lien, A.B. Strong, A fully conservative second-order finite difference scheme for incompressible flow on nonuniform grids, J. Comput. Phys. 177 (2002) 117–133.
[11] F.H. Harlow, J.E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, Phys. Fluids 8 (12) (1965) 2182–2189.
[12] R.S. Hirsh, Higher order accurate difference solutions of fluid mechanics problems by a compact differencing technique, J. Comput. Phys. 19 (1975) 90–109.
[13] C.W. Hirt, B.D. Nichols, N.C. Romero, SOLA – a numerical solution algorithm for transient fluid flows, Los Alamos Report LA-5852, January 1975.
[14] R.I. Issa, Solution of the implicitly discretised fluid flow equations by operator-splitting, J. Comput. Phys. 62 (1985) 40–65.
[15] S.A. Jordan, An iterative scheme for numerical solution of steady incompressible viscous flows, Comput. Fluids 21 (4) (1992) 503–517.
[16] G.E. Karniadakis, M. Israeli, S. Orszag, High-order splitting methods for the incompressible Navier–Stokes equations, J. Comput. Phys. 97 (2) (1991) 414–443.
[17] G.E. Karniadakis, S.J. Sherwin, Spectral/hp Element Method for CFD, Oxford University Press, Oxford, 1999.
[18] J. Kim, P. Moin, Application of a fractional-step method to incompressible Navier–Stokes equations, J. Comput. Phys. 59 (1985) 308–323.
[19] S.K. Lele, Compact finite difference schemes with spectral-like resolution, J. Comput. Phys. 103 (1992) 16–42.
[20] D.A. McDonald, Blood Flow in Arteries, Edward Arnold Ltd., 1974.
[21] K. Mahesh, A family of high order finite difference schemes with good spectral resolution, J. Comput. Phys. 145 (1998) 332–358.
[22] K. Mahesh, G. Constantinescu, P. Moin, A numerical method for large-eddy simulation in complex geometries, J. Comput. Phys. 197 (2004) 215–240.
[23] C.L. Merkle, M. Athavale, Time-accurate unsteady incompressible flow algorithms based on artificial compressibility, AIAA Paper 87-1137, 1987.
[24] Y. Morinishi, T.S. Lund, O.V. Vasilyev, P. Moin, Fully conservative higher order finite difference schemes for incompressible flow, J. Comput. Phys. 143 (1998) 90–124.
[25] N. Santhanam, S.K. Lele, J.H. Ferziger, A robust high-order compact method for large eddy simulation, J. Comput. Phys. 191 (2003) 392–419.
[26] R.L. Panton, Incompressible Flow, John Wiley, Chichester, 1984.
[27] S.V. Patankar, Numerical Heat Transfer and Fluid Flow, Hemisphere Publishing Co., Washington, DC, 1980.

[28] C.M. Rhie, W.L. Chow, Numerical study of the turbulent flow past an airfoil with trailing edge separation, AIAA J. 21 (11) (1983) 1525–1532.
[29] S.E. Rogers, Numerical solution of the incompressible Navier–Stokes equations, NASA TM 102199, Ames Research Center, Moffett Field, CA, 1990.
[30] S.E. Rogers, D. Kwak, C. Kiris, Steady and unsteady solutions of the incompressible Navier–Stokes equations, AIAA J. 29 (4) (1991) 603–610.
[31] M. Rosenfeld, D. Kwak, M. Vinokur, A fractional step solution method for the unsteady incompressible Navier–Stokes equations in generalized curvilinear coordinate systems, J. Comput. Phys. 94 (1) (1991) 102–137.
[32] Y. Saad, SPARSKIT: A basic tool kit for sparse matrix computations, NASA Ames Research Center, 1994.
[33] Y. Saad, Iterative Methods for Sparse Linear Systems, PWS Publishing Co., Boston, 1996.
[34] O.V. Vasilyev, High order finite difference schemes on non-uniform meshes with good conservation properties, J. Comput. Phys. 157 (2000) 746–761.
[35] J.R. Womersley, The mathematical analysis of the arterial circulation in a state of oscillatory motion, Wright Air Development Center, WADC-TR56-614, 1957.
[36] J.C. Wu, Theory of aerodynamic force and moment in viscous flow, AIAA J. 19 (4) (1981) 432–441.