ACADEMIC PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Computational Physics 187 (2003) 168–196

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

A modified nodal scheme for the time-dependent, incompressible Navier–Stokes equations

Fei Wang, Rizwan-uddin *

Department of Nuclear, Plasma and Radiological Engineering, University of Illinois at Urbana-Champaign, 103 S. Goodwin Ave, Urbana, IL 61801, USA

Received 17 June 2002; received in revised form 9 January 2003; accepted 28 January 2003

Abstract

Based on Poisson equation for pressure, a nodal numerical scheme is developed for the time-dependent, incompressible Navier–Stokes equations. Derivation is based on local transverse-integrations over finite size brick-like cells that transform each partial differential equation to a set of ordinary differential equations (ODEs). Solutions of these ODEs for the transverse-averaged dependent variables are then utilized to develop the difference scheme. The discrete variables are scalar velocities and pressure, averaged over the faces of brick-like cells in the (x, y, t) space. Cell-interior variation of transverse-averaged pressure in each spatial direction is quadratic. Cell-interior variation of transverse-averaged velocity in each spatial direction is a sum of a constant, a linear and an exponential term. Due to the introduction of delayed coefficients, the exponential functions are to be evaluated only once at each time step. The semi-implicit scheme has inherent upwinding. Results of applications to several test problems show that the scheme is very robust and leads to a second-order error. As expected in such coarse-mesh schemes, even relatively large size cells lead to small errors. Extension to three dimensions is straightforward.

Keywords: Navier–Stokes; Difference equations; Nodal analytical; Coarse-mesh; Inherent upwinding

1. Introduction

A number of *coarse-mesh* numerical schemes, specifically targeted to solve problems over large computational domains, have been developed over the last three decades [1–18]. Characterized by an initial investment of human effort – now greatly reduced due to the availability of software for algebraic manipulations – these schemes yield numerical solutions with comparable accuracy in less CPU time than those obtained with other, more conventional, approaches. Typically, this efficiency is achieved by using a coarser mesh size than those required by other schemes. Hence, for given mesh size, a second-order coarse-mesh scheme is likely to lead to smaller error than a second-order finite-difference scheme. Coarse-mesh

* Corresponding author. Tel.: +217-244-4944; fax: +217-333-2906.
E-mail address: rizwan@uiuc.edu (Rizwan-uddin).

0021-9991/03/$ - see front matter
doi:10.1016/S0021-9991(03)00093-7

schemes do have some limitations. For example, those that rely on the transverse-integration procedure (explained in Section 2) are restricted to physical domains with boundaries parallel to one of the axis, i.e, to geometries that can be filled with brick-like cells. However, as shown by the experience of the nuclear industry, these schemes provide enough savings in CPU time to justify their development, even if they are applicable to only a limited set of problems. Moreover, efforts are also underway to relax these restrictions and hence make the coarse-mesh methods applicable to even larger set of problems.

Nodal methods [19] are a subset of coarse-mesh methods. A *nodal* scheme is developed by approximately satisfying the governing differential equations on finite size brick-like elements that are obtained by discretizing the space of independent variables. In the early development of nodal schemes these brick-like elements were referred to as *nodes* – hence the schemes were called *nodal*. Nodes in nodal methods are however similar to the *elements* of the finite element approach, i.e., they are *finite volumes* – and not points – in the space of independent variables. This is often a source of confusion since "node" is already used in the finite difference and finite volume methods to refer to a "point" in space. To avoid this confusion we will refer to the finite size brick-like volume in the space–time domain as a *cell*. (Consequently, nodal integral approach has also been called the "cell-analytic" approach [13].)

As in the space–time finite element method (FEM), time in the nodal approach may be treated in the same manner as any spatial direction. Nuclear industry has taken full advantage of developments in coarse-mesh methods, and consequently, they are the workhorse of the nuclear industry's neutron diffusion and neutron transport codes [1–5,8]. A review of nodal methods, developed and used by the nuclear industry, is given by Lawrence [11]. Coarse-mesh schemes have also been developed for fluid flow and heat transfer problems [7,9,12,16,17]. Other branches of science and engineering have also taken advantage of similar approaches to develop efficient schemes [15,19]. Nodal methods, as a general class of computational schemes, are discussed by Hennert [19]. A comparison of nodal schemes and *exact finite difference schemes* has appeared recently [20]. A brief survey of coarse-mesh methods is given below.

Partial current balance method (PCBM) [1,2], developed for multi-group neutron diffusion equations, utilizes multidimensional Green's function with nearest neighbor coupling to arrive at a set of discrete equations. PCBM results in a large number of discrete unknowns per cell. Later, because of the simplicity in development, *transverse integration procedure* (TIP) became the primary step in the development of new coarse mesh methods. This procedure leads to numerical schemes with smaller number of discrete unknowns per cell when compared with PCBM. Built upon the TIP, a *nodal Green's function method* (NGFM) was developed to solve the multi-group neutron diffusion equations [3,4]. The locally defined Green's functions were first applied to fluid flow problem in the *nodal Green's tensor method* (NGTM) [6,7]. Later, the nodal integral method (NIM) was developed for the steady-state [9] and time-dependent [12] Navier–Stokes equations. Application of the NIM – also known as *nodal analytical* or *cell analytic* method – to the neutron diffusion problem [8] and fluid flow problems [9] is mathematically equivalent to the NGTM. Since Green's function is not needed in the NIM, it is simpler to develop and implement than NGTM. NIM was applied to the steady-state Boussinesq equations for natural convection, and to several steady-state incompressible flow problems [8–10]. Esser and Witt [16] developed a nodal scheme for the two-dimensional, vorticity-stream function formulation of the Navier–Stokes equations. This development – that leads to inherent upwinding in the numerical scheme – however cannot be easily extended to three dimensions. NIM was also developed and applied to the time-dependent heat conduction problem [12]. Michael et al. [17] developed a second- and a third-order NIM for the convection–diffusion equation, and compared the results with those obtained using the LECUSSO scheme [21]. They showed that the nodal integral method achieved the same level of accuracy with significantly less CPU time than the very efficient LECUSSO scheme [17].

As mentioned above, nodal schemes have been developed for the Navier–Stokes equations. Though highly innovative, those early applications did not take full advantage of the potential that the nodal approach offers. Consequently, these schemes for the Navier–Stokes equations can be further improved. To

lay down the groundwork for the scheme developed in Section 3, pertinent features of the nodal integral scheme are outlined in the following section. Past applications to the Navier–Stokes equations are also discussed, leading to suggestions for improvements.

2. Background

Steps essential to the NIM are outlined briefly in Section 2.1. Issues relevant to the treatment of the nonlinear terms, and specifically those relevant to the Navier–Stokes equations, are separately discussed in Section 2.2, leading to the modified scheme developed in Section 3.

2.1. Nodal integral method and transverse integration procedure

In general, development of a nodal integral method can be split into the following four steps:

(a) After discretizing the space–time domain into brick-like cells, each PDE is reduced to a set of ODEs by applying the *transverse integration procedure* (TIP) over a cell. The dependent variables in these ODEs are referred to as *transverse-averaged variables*.
(b) These ODEs are split into homogeneous and inhomogeneous (also called, pseudo-source) terms. After making certain assumptions about the homogeneous and inhomogeneous terms, the ODEs are solved analytically for *local* solution within each cell using the discrete values of the transverse-averaged variables at the cell surfaces as boundary conditions. The transverse-averaged variables evaluated at the cell surfaces are the discrete variables of the nodal scheme.
(c) Continuity of these transverse-averaged variables (and their derivatives for second-order ODEs) is imposed on cell boundaries to obtain a set of discrete equations.
(d) Constraint conditions are next used to eliminate the coefficients of expansion of the pseudo-source terms (identified in step (b)) to obtain a set of discrete equations with number equal to the number of discrete unknowns per cell.

Steps (a) and (b) are further explained below.

In the TIP, after discretizing the space–time domain of independent variables (X, Y, T) into finite size computational cells of size $(\Delta x \times \Delta y \times \Delta t)$, cell specific local coordinates (x, y, t), with origin at the center of the cell, are introduced. Hence, with $\Delta x = 2a,\ \Delta y = 2b$ and $\Delta t = 2\tau$, the cell is given by $-a_i \leqslant x \leqslant +a_i$, $-b_j \leqslant y \leqslant +b_j,\ -\tau_k \leqslant t \leqslant +\tau_k$. See Fig. 1. Each governing PDE is then integrated locally over the space–time cell over all independent variables except one, leading to an ODE. Repeating this process with different combinations of independent variables leads to a set of ODEs for each PDE. The ODEs are for transverse-averaged variables. For example, an ODE will be obtained for $\bar{u}^{xt}_{i,j,k}(y)$ – the u velocity, $u(x, y, t)$, transverse-averaged *locally* over the cell in x and t directions – defined as

$$\bar{u}^{xt}_{i,j,k}(y) \equiv \frac{1}{4ab}\int_{-a}^{a}\int_{-\tau}^{\tau} u_{i,j,k}(x, y, t)\,\mathrm{d}x\,\mathrm{d}t, \tag{2.1}$$

where the overbar and the symbols that follow $(^{-xt})$ indicate the independent variables over which the local averaging has been carried out (extension to three dimensions is straightforward). The discrete unknowns of the nodal approach are the transverse-averaged variables evaluated at the cell-surfaces. Or, in other words the discrete unknowns are the unknowns $(u(x, y, t),\ v(x, y, t)$ and $p(x, y, t)$ for the Navier–Stokes equations) averaged over surfaces of the space–time cells. For example, one of the discrete unknown is the transverse-averaged variable $\bar{u}^{xt}_{i,j,k}(y)$ evaluated at $y = b$, i.e., $\bar{u}^{xt}_{i,j,k}(y = b) \equiv \bar{u}^{xt}_{i,j,k}$ (see Fig. 1).

While step (a) is common to almost all nodal integral methods, step (b) is crucial in understanding the difference between different nodal integral approaches, and is further elaborated here. In general, ODEs obtained after the TIP do not have analytical solutions. The basic idea behind NIM is to analytically solve in

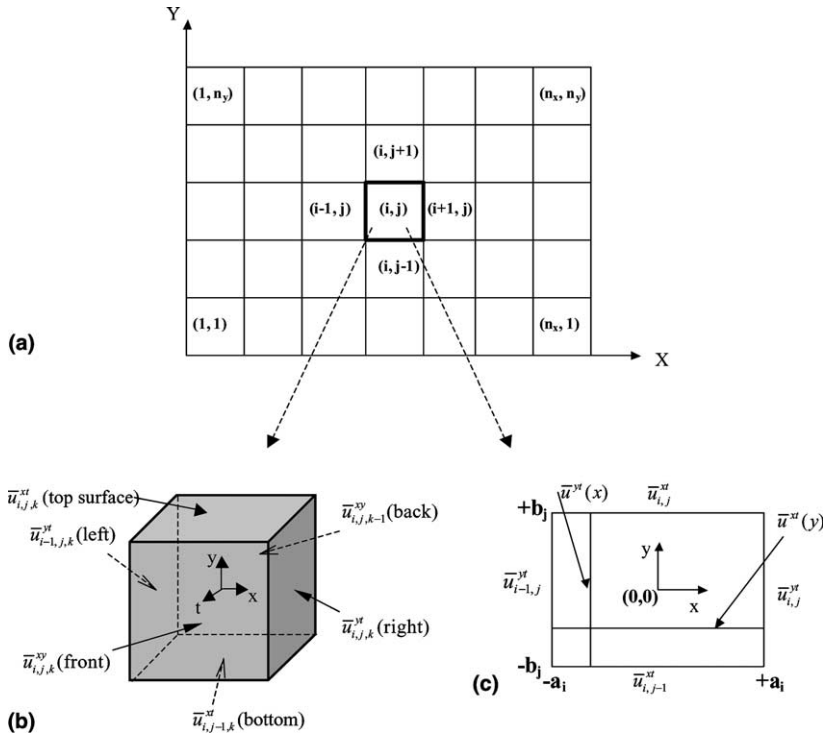


Fig. 1. Domain discretization for the nodal integral method. (a) Discretization of the spatial domain into $n_x \times n_y$ cells. (b) Space–time cell (i, j, k) and local coordinate system. (c) Details of the local coordinates in cell (i, j) in x–y plane.

each cell as much of the transverse-integrated ODEs as possible [9,14] for a homogeneous solution, and obtain (approximate) particular solutions corresponding to the remaining terms. Hence, each ODE is split into two parts: a group of terms that are retained on the left-hand side, and remaining terms that are written on the right-hand side. Splitting the ODEs into terms retained on the left-hand side and those kept on the right-hand side is not arbitrary. In general, only the terms of the ODEs that are linear in the dependent variable to be solved using that ODE, are retained on the left-hand side. The nonlinear terms, as well as linear terms that involve other dependent variables, are lumped together on the right-hand side of the equation as *inhomogeneous* terms (traditionally called the *pseudo-source term*). Solutions to these ODEs are then written as the sum of homogeneous and particular solutions. The homogeneous part of the solution of these ODEs then consists of polynomial, trigonometric, exponential or other functions. Since this (homogeneous) component of the solution is obtained by analytically solving a part of the transverse-averaged ODEs, it is likely to capture characteristics that are directly *relevant* to the problem. The homogeneous solution can thus be considered to be a "finite set of natural basis functions" specific to the problem – or at least to a part of the problem. This feature makes nodal analytic method distinct from other numerical methods – such as Fourier, collocation and spectral etc. – in which "basis functions" independent of the problem at hand are usually employed. Particular solutions, corresponding to the terms that are lumped on the right-hand side in the pseudo-source term, are obtained after expanding the pseudo-source terms in a set of complete basis functions and truncating at a desired level. Hence, the terms lumped in the pseudo-source terms (and the physical process that these terms represent) are less accurately captured by the numerical scheme than those that contribute to the homogeneous part of the solution. Consequently, it is desirable to retain as many terms on the left-hand side in the transverse-averaged ODEs as possible. This point is further elaborated in [14].

In step (c), the general solution within each cell – consisting of the homogeneous and particular parts – is used to obtain the set of discretized equations. The coefficients of expansion of the pseudo-source terms, that appear in the particular solutions, are initially unknown. They are eliminated in step (d), leading to a set of discrete algebraic equations.

2.2. NIM for nonlinear equations and past applications to the Navier–Stokes equations

In early applications of the NIM, the nonlinear terms were treated as part of the pseudo-source terms [9,12]. For example, in the NIM developed for the time-dependent, two-dimensional Navier–Stokes (N–S) equations, the nonlinear convection terms as well as the pressure gradient term were lumped into the pseudo-source term [12]. In addition, dogged by the absence of pressure in the continuity equation, normal stress, instead of pressure, was used as an independent variable. Consequently, the standard continuity and the momentum equations for the two-dimensional, time-dependent, incompressible flow, after the transverse integration were transformed into the following set of ODEs [9,12]:

$$\frac{\mathrm{d}\bar{u}^{yt}(x)}{\mathrm{d}x} = \bar{\varphi}_3^{yt} \quad \text{(continuity equation integrated over } y \text{ and } t\text{)}, \tag{2.2}$$

$$\frac{\mathrm{d}\bar{v}^{xt}(y)}{\mathrm{d}y} = \bar{\varphi}_3^{xt} \quad \text{(continuity equation integrated over } x \text{ and } t\text{)}, \tag{2.3}$$

$$\frac{\mathrm{d}\bar{u}^{xy}(t)}{\mathrm{d}t} = \bar{\varphi}_1^{xy} \quad (x \text{ momentum equation integrated over } x \text{ and } y), \tag{2.4}$$

$$\frac{\mathrm{d}^2\bar{u}^{xt}(y)}{\mathrm{d}y^2} = \bar{\varphi}_1^{xt} \quad (x \text{ momentum equation integrated over } x \text{ and } t), \tag{2.5}$$

$$\frac{\mathrm{d}\bar{\tau}_x^{yt}(x)}{\mathrm{d}x} = \bar{\varphi}_4^{yt} \quad (x \text{ momentum equation integrated over } y \text{ and } t), \tag{2.6}$$

$$\frac{\mathrm{d}\bar{v}^{xy}(t)}{\mathrm{d}t} = \bar{\varphi}_2^{xy} \quad (y \text{ momentum equation integrated over } x \text{ and } y), \tag{2.7}$$

$$\frac{\mathrm{d}^2\bar{v}^{yt}(x)}{\mathrm{d}x^2} = \bar{\varphi}_2^{yt} \quad (y \text{ momentum equation integrated over } y \text{ and } t), \tag{2.8}$$

$$\frac{\mathrm{d}\bar{\tau}_y^{xt}(y)}{\mathrm{d}y} = \bar{\varphi}_4^{xt} \quad (y \text{ momentum equation integrated over } x \text{ and } t), \tag{2.9}$$

where the normal stresses are defined as

$$\tau_x \equiv P - \mu\frac{\partial u}{\partial x}, \tag{2.10}$$

$$\tau_y \equiv P - \mu\frac{\partial v}{\partial y}. \tag{2.11}$$

Terms not explicit on the left-hand side of the transverse-averaged equations [(2.2)–(2.9)] were lumped in the pseudo-source (φ) terms on the right-hand side.

These ODEs were solved to obtain cell-interior solutions for the transverse-averaged variables. For example, Eq. (2.5) led to a quadratic (local) variation in y for the transverse-averaged u velocity, $\bar{u}^{xt}(y)$; and Eq. (2.2) led to a linear variation in x for the transverse-averaged u velocity, $\bar{u}^{yt}(x)$. The formulation consequently led to asymmetries in the local solutions of transverse-averaged u (and v) velocities in the x and y directions. Moreover, lumping all the convection terms, when solving the momentum equation, into the right-hand side, also meant that the homogeneous part of the analytical solution captured only the diffusion process – and not convection.

Hence, there are three desirable features in any new coarse-mesh nodal numerical scheme for the N–S equations: (1) local analytical solutions that are more representative of the N–S equations and the physical processes they represent; (2) formulation in terms of only the primitive variables; and (3) a numerical scheme that is symmetric in all spatial directions. A recipe for the first of these is given in the following paragraph, followed by suggestions on how to include the second and third features.

Motivated by the desire to "exactly" solve *more* of the ODEs, i.e., to obtain homogeneous solution to a larger fraction of the ODEs, a modified nodal integral method (MNIM) was proposed by Rizwan-uddin [14]. The method proposed was successfully applied to Burgers' equation [14], and led to lower CPU time when compared with the conventional NIM. The new approach was further modified later and applied to the 2D Burgers' equations [15]. The ideas introduced are similar to the concept of "delayed coefficients" in which part of the nonlinear convection term is evaluated in terms of the u and v velocities at the *previous* time step [22]. Thus, in the MNIM for the 2D Burgers' equations, the nonlinear convection term, in its approximated form, is retained on the left-hand side of the ODE, and the homogeneous part of the solution is written for the diffusion as well as the convection term. That is, instead of Eq. (2.2) for $\bar{u}^{yt}(x)$ and Eq. (2.5) for $\bar{u}^{xt}(y)$ – which would, respectively, lead to linear and quadratic local transverse-averaged velocities – two ODEs are, respectively, obtained by locally transverse-averaging the x-momentum equation over x and t, and over y and t. These equations are of the form

$$\nu \frac{\mathrm{d}^2 \bar{u}^{\eta t}(\mu)}{\mathrm{d}\mu^2} - \sigma \frac{\mathrm{d}\bar{u}^{\eta t}(\mu)}{\mathrm{d}\mu} = \bar{\varphi}_1^{\eta t}, \tag{2.12}$$

where $\eta = y, x$, $\mu = x, y$ and $\sigma = u^0, v^0$. Consequently, a larger part of the transverse-integrated ODE is analytically inverted when these equations are solved for $\bar{u}^{yt}(x)$ and $\bar{u}^{xt}(y)$ leading to local cell-interior solutions of the *constant + linear + exponential* form

$$\bar{u}^{\eta t}(\mu) = C_3 \, \mathrm{e}^{\sigma\mu/\nu} + \bar{\varphi}_1^{\eta t}\mu + C_4. \tag{2.13}$$

These local, cell-interior solutions capture the effect of diffusion as well as convection, and are more representative of the physics than the linear or quadratic local variations for the transverse-averaged velocities used in earlier development of the NIM. Solution of the 1-D [14] and 2-D [15] Burgers' equations with the modified nodal approach – in which the convection term is retained on the left-hand side and contributes to the homogeneous part of the analytical solution – showed that the resulting analytical solution for the cell interior variation is capable of capturing steep variations within large size spatial cells. Moreover, the numerical scheme that results also has *inherent* upwinding.

A nodal scheme for the Navier–Stokes equations only in terms of primitive variables can be developed by using the Poisson equation for pressure [23]. Solving the (Poisson) continuity equation for pressure leaves the two momentum equations to be solved for the velocities. Consequently, this also eliminates the asymmetries between different spatial directions. (The asymmetry between $\bar{u}^{yt}(x)$ and $\bar{u}^{xt}(y)$ in the original development [9] resulted from the fact that continuity equation in its primitive form was used to solve for $\bar{u}^{yt}(x)$, while the x-momentum equation was solved to determine $\bar{u}^{xt}(y)$.) Michael and Dorning [24,25] have developed a nodal scheme for the Navier–Stokes equations in primitive variables recently. This scheme is similar in its treatment of the transverse-averaged velocities to the scheme developed below. However,

motivated by the desire to develop scheme that could be back-fitted in some existing production level codes, approximations were introduced to develop discrete equations for a single, cell-averaged pressure. The approach in the current work does not rely on similar approximations, and two discrete equations are retained for the two transverse-averaged pressures for each cell.

The modified nodal scheme for the Navier–Stokes equations – in which the continuity equation is replaced by the Poisson equation for pressure – is developed in Section 3. Corresponding boundary conditions are developed in Section 4. Numerical results are presented in Section 5. Concluding remarks are in Section 6.

3. Modified nodal integral method for Navier–Stokes equations

Two-dimensional, time-dependent, incompressible, isothermal Navier–Stokes equations in primitive variable form are

$$\frac{\partial u}{\partial X}+\frac{\partial v}{\partial Y}=0, \tag{3.1}$$

$$\frac{\partial u}{\partial T}+u\frac{\partial u}{\partial X}+v\frac{\partial u}{\partial Y}-\nu\left[\frac{\partial^2 u}{\partial X^2}+\frac{\partial^2 u}{\partial Y^2}\right]+\frac{1}{\rho}\frac{\partial p}{\partial X}+b_X(X,Y,T)=0, \tag{3.2}$$

$$\frac{\partial v}{\partial T}+u\frac{\partial v}{\partial X}+v\frac{\partial v}{\partial Y}-\nu\left[\frac{\partial^2 v}{\partial X^2}+\frac{\partial^2 v}{\partial Y^2}\right]+\frac{1}{\rho}\frac{\partial p}{\partial Y}+b_Y(X,Y,T)=0, \tag{3.3}$$

where $b_X(X,Y,T)$ and $b_Y(X,Y,T)$ represent volumetric sources such as gravity. To develop a numerical scheme, for well-known reasons [26], very often a Poisson equation for pressure is used instead of the continuity equation, Eq. (3.1). Manipulating Eqs. (3.2) and (3.3), the Poisson equation for pressure is given by

$$\frac{1}{\rho}\left[\frac{\partial^2 p}{\partial X^2}+\frac{\partial^2 p}{\partial Y^2}\right]=-\left(\frac{\partial u}{\partial X}\right)^2-2\frac{\partial u}{\partial Y}\frac{\partial v}{\partial X}-\left(\frac{\partial v}{\partial Y}\right)^2-\frac{\partial b_X}{\partial X}-\frac{\partial b_Y}{\partial Y}-\left[\frac{\partial D}{\partial T}+u\frac{\partial D}{\partial X}+v\frac{\partial D}{\partial Y}-\nu\frac{\partial^2 D}{\partial X^2}-\nu\frac{\partial^2 D}{\partial Y^2}\right], \tag{3.4}$$

where the dilatation term D is given by

$$D\equiv\frac{\partial u}{\partial X}+\frac{\partial v}{\partial Y}. \tag{3.5}$$

Note that Eq. (3.4) is derived from the two scalar momentum equations, and must be combined with the continuity equation before it is used to solve for pressure. Since the continuity equation is simply $D=0$, setting the square bracket on the RHS of Eq. (3.4) to zero leads to an equation that can be used to solve for pressure. However, several authors have pointed out that setting D in Eq. (3.4) identically to zero may lead to an unstable numerical scheme [26,27]. Hence, while solving the Poisson equation for pressure, retention of, for example, the temporal derivative of the local dilatation is considered essential for the convergence of a numerical scheme. Moreover, a discretization of the dilatation term D consistent with the continuity equation is believed to be important to ensure the convergence of the numerical scheme. Imposing continuity equation requires the entire square bracket on the RHS of Eq. (3.4) to be set equal to zero. However, to simplify and simultaneously retain the stability of the numerical scheme, we replace the square bracket on the RHS of Eq. (3.4) with $(D/2\tau)$ evaluated at the current time step.

3.1. Transverse integration procedure and the set of ODEs

In the nodal method, the space–time domain (X, Y, T) is first discretized into cells (i, j, k) of size $(2a_i \times 2b_j \times 2\tau_k)$ with cell-centered *local coordinates* $(-a_i \leqslant x \leqslant a_i,\ -b_j \leqslant y \leqslant b_j,\ -\tau_k \leqslant t \leqslant \tau_k)$. Fig. 1 shows the discretized spatial domain, a space–time cell, and the local coordinates in a cell with origin located at the center of the cell. As a prelude to the development of the numerical scheme, the momentum equations are re-written in terms of the local coordinate system in the following form:

$$\frac{\partial u}{\partial t} + u_{\mathrm{p}}\frac{\partial u}{\partial x} + v_{\mathrm{p}}\frac{\partial u}{\partial y} - \nu\left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right] = -\frac{1}{\rho}\frac{\partial p}{\partial x} - b_x(x,y,t) - (u - u_{\mathrm{p}})\frac{\partial u}{\partial x} - (v - v_{\mathrm{p}})\frac{\partial u}{\partial y}, \tag{3.6}$$

$$\frac{\partial v}{\partial t} + u_{\mathrm{p}}\frac{\partial v}{\partial x} + v_{\mathrm{p}}\frac{\partial v}{\partial y} - \nu\left[\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right] = -\frac{1}{\rho}\frac{\partial p}{\partial y} - b_y(x,y,t) - (u - u_{\mathrm{p}})\frac{\partial v}{\partial x} - (v - v_{\mathrm{p}})\frac{\partial v}{\partial y}, \tag{3.7}$$

where u_{p} and v_{p} are, respectively, the cell-averaged u and v velocities at the previous time step. Convection terms based on cell-averaged velocities at the previous time step have been added to both sides of Eqs. (3.2) and (3.3) to obtain Eqs. (3.6) and (3.7). The reason for writing the momentum equations in this form was alluded to in the previous section (delayed coefficients), and it will become further obvious in the following sections. By applying the local transverse integration procedure to Eqs. (3.4), (3.6) and (3.7), eight transverse-integrated ordinary differential equations are obtained below.

Applying the transverse-integration operator

$$\frac{1}{4a_i\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-a_i}^{a_i} \mathrm{d}x\,\mathrm{d}t$$

to Eqs. (3.4), (3.6) and (3.7), respectively, yields

$$\frac{\mathrm{d}^2\bar{p}^{xt}(y)}{\mathrm{d}y^2} = \bar{S}_1^{xt}(y), \tag{3.8}$$

$$v_{\mathrm{p}}\frac{\mathrm{d}\bar{u}^{xt}(y)}{\mathrm{d}y} - \nu\frac{\mathrm{d}^2\bar{u}^{xt}(y)}{\mathrm{d}y^2} = \bar{S}_2^{xt}(y), \tag{3.9}$$

$$v_{\mathrm{p}}\frac{\mathrm{d}\bar{v}^{xt}(y)}{\mathrm{d}y} - \nu\frac{\mathrm{d}^2\bar{v}^{xt}(y)}{\mathrm{d}y^2} = \bar{S}_3^{xt}(y), \tag{3.10}$$

where the cell-specific subscripts (i, j, k) on independent variables have been omitted, and

$$\bar{\phi}_{i,j,k}^{xt}(y) \equiv \frac{1}{4a_i\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-a_i}^{a_i}\phi_{i,j,k}(x,y,t)\,\mathrm{d}x\,\mathrm{d}t, \quad \phi = u, v, p. \tag{3.11}$$

$\bar{\phi}_{i,j,k}^{yt}(x)$ and $\bar{\phi}_{i,j,k}^{xy}(t)$ are similarly defined. Terms not explicit in Eqs. (3.8)–(3.10) are lumped into the right hand as *pseudo-source* terms:

$$\begin{aligned}\bar{S}_1^{xt}(y) \equiv -\frac{1}{4a_i\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-a_i}^{a_i}\mathrm{d}x\,\mathrm{d}t\Bigg(&\frac{\partial^2 p}{\partial x^2} + \rho\left(\frac{\partial u}{\partial x}\right)^2 + 2\rho\frac{\partial u}{\partial y}\frac{\partial v}{\partial x} + \rho\left(\frac{\partial v}{\partial y}\right)^2 + \rho\frac{\partial b_x}{\partial x} + \rho\frac{\partial b_y}{\partial y}\\ &+ \rho\left(\frac{\partial D}{\partial t} + u\frac{\partial D}{\partial x} + v\frac{\partial D}{\partial y} - \nu\frac{\partial^2 D}{\partial x^2} - \nu\frac{\partial^2 D}{\partial y^2}\right)\Bigg),\end{aligned} \tag{3.12}$$

$$\bar{S}_2^{xt}(y) \equiv -\frac{1}{4a_i\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-a_i}^{a_i} \mathrm{d}x\,\mathrm{d}t\left(\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+(v-v_\mathrm{p})\frac{\partial u}{\partial y}-v\frac{\partial^2 u}{\partial x^2}+\frac{1}{\rho}\frac{\partial p}{\partial x}+b_x\right) \tag{3.13}$$

and

$$\bar{S}_3^{xt}(y) \equiv -\frac{1}{4a_i\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-a_i}^{a_i} \mathrm{d}x\,\mathrm{d}t\left(\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+(v-v_\mathrm{p})\frac{\partial v}{\partial y}-v\frac{\partial^2 v}{\partial x^2}+\frac{1}{\rho}\frac{\partial p}{\partial y}+b_y\right). \tag{3.14}$$

There are no approximations introduced up to this stage of the development.

The inhomogeneous pseudo-source terms in Eqs. (3.8)–(3.10) are then expanded in Legendre polynomials. Here, the expansion is truncated at the zeroth order, which is consistent with the goal of a second-order scheme [9]. (In general, truncating at higher order, in conjunction with other consistent approximations, leads to numerical scheme of order higher than second.) The above process yields

$$\frac{\mathrm{d}^2\bar{p}^{xt}(y)}{\mathrm{d}y^2}=\bar{S}_1^{xt}, \tag{3.15}$$

$$v_\mathrm{p}\frac{\mathrm{d}\bar{u}^{xt}(y)}{\mathrm{d}y}-v\frac{\mathrm{d}^2\bar{u}^{xt}(y)}{\mathrm{d}y^2}=\bar{S}_2^{xt} \tag{3.16}$$

and

$$v_\mathrm{p}\frac{\mathrm{d}\bar{v}^{xt}(y)}{\mathrm{d}y}-v\frac{\mathrm{d}^2\bar{v}^{xt}(y)}{\mathrm{d}y^2}=\bar{S}_3^{xt}. \tag{3.17}$$

Note that it is only the absence of the argument that differentiates $\bar{S}_1^{xt}(y)$ in Eq. (3.8) from $\bar{S}_1^{xt}$ in Eq. (3.15). Latter is the zeroth-order Legendre expansion of the former.

Similarly, applying the transverse-integration operator

$$\frac{1}{4b_j\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-b_j}^{b_j}\mathrm{d}y\,\mathrm{d}t,$$

to Eqs. (3.4), (3.6) and (3.7), and approximating the pseudo-source terms by constants, result in

$$\frac{\mathrm{d}^2\bar{p}^{yt}(x)}{\mathrm{d}x^2}=\bar{S}_1^{yt}, \tag{3.18}$$

$$u_\mathrm{p}\frac{\mathrm{d}\bar{u}^{yt}(x)}{\mathrm{d}x}-v\frac{\mathrm{d}^2\bar{u}^{yt}(x)}{\mathrm{d}x^2}=\bar{S}_2^{yt} \tag{3.19}$$

and

$$u_\mathrm{p}\frac{\mathrm{d}\bar{v}^{yt}(x)}{\mathrm{d}x}-v\frac{\mathrm{d}^2\bar{v}^{yt}(x)}{\mathrm{d}x^2}=\bar{S}_3^{yt}, \tag{3.20}$$

where the definitions of the pre-truncated pseudo-source terms are

$$\begin{aligned}\bar{S}_1^{yt}(x) \equiv -\frac{1}{4b_j\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-b_j}^{b_j}\mathrm{d}y\,\mathrm{d}t\Bigg(&\frac{\partial^2 p}{\partial y^2}+\rho\left(\frac{\partial u}{\partial x}\right)^2+2\rho\frac{\partial u}{\partial y}\frac{\partial v}{\partial x}+\rho\left(\frac{\partial v}{\partial y}\right)^2+\rho\frac{\partial b_x}{\partial x}+\rho\frac{\partial b_y}{\partial y}\\&+\rho\left(\frac{\partial D}{\partial t}+u\frac{\partial D}{\partial x}+v\frac{\partial D}{\partial y}-v\frac{\partial^2 D}{\partial x^2}-v\frac{\partial^2 D}{\partial y^2}\right)\Bigg),\end{aligned} \tag{3.21}$$

$$\bar{S}_2^{yt}(x) \equiv -\frac{1}{4b_j\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-b_j}^{b_j} \mathrm{d}y\,\mathrm{d}t\left(\frac{\partial u}{\partial t}+v\frac{\partial u}{\partial y}+(u-u_\mathrm{p})\frac{\partial u}{\partial x}-\nu\frac{\partial^2 u}{\partial y^2}+\frac{1}{\rho}\frac{\partial p}{\partial x}+b_x\right) \tag{3.22}$$

and

$$\bar{S}_3^{yt}(x) \equiv -\frac{1}{4b_j\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-b_j}^{b_j} \mathrm{d}y\,\mathrm{d}t\left(\frac{\partial v}{\partial t}+v\frac{\partial v}{\partial y}+(u-u_\mathrm{p})\frac{\partial v}{\partial y}-\nu\frac{\partial^2 v}{\partial y^2}+\frac{1}{\rho}\frac{\partial p}{\partial y}+b_y\right). \tag{3.23}$$

Next, applying the operator,

$$\frac{1}{4a_ib_j}\int_{-a_i}^{a_i}\int_{-b_j}^{b_j}\mathrm{d}x\,\mathrm{d}y,$$

to Eqs. (3.5) and (3.6), and expanding and truncating the pseudo-source terms yield,

$$\frac{\mathrm{d}\bar{u}^{xy}(t)}{\mathrm{d}t}=\bar{S}_2^{xy}, \tag{3.24}$$

$$\frac{\mathrm{d}\bar{v}^{xy}(t)}{\mathrm{d}t}=\bar{S}_3^{xy}, \tag{3.25}$$

where the pre-truncated pseudo-source terms – $\bar{S}_2^{xy}(t)$ and $\bar{S}_3^{xy}(t)$ – are given by

$$\bar{S}_2^{xy}(t) \equiv -\frac{1}{4a_ib_j}\int_{-a_i}^{a_i}\int_{-b_j}^{b_j}\mathrm{d}x\,\mathrm{d}y\left(u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}-\nu\frac{\partial^2 u}{\partial x^2}-\nu\frac{\partial^2 u}{\partial y^2}+\frac{1}{\rho}\frac{\partial p}{\partial x}+b_x\right), \tag{3.26}$$

$$\bar{S}_3^{xy}(t) \equiv -\frac{1}{4a_ib_j}\int_{-a_i}^{a_i}\int_{-b_j}^{b_j}\mathrm{d}x\,\mathrm{d}y\left(u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}-\nu\frac{\partial^2 v}{\partial x^2}-\nu\frac{\partial^2 v}{\partial y^2}+\frac{1}{\rho}\frac{\partial p}{\partial y}+b_y\right). \tag{3.27}$$

Note that due to the absence of a time derivative, the pressure equation leads to only two ODEs. Eqs. (3.15)–(3.20), (3.24) and (3.25) form the set of eight ODEs that will be solved to develop the set of discrete equations.

The reason behind the introduction of the convection term based on the (known) cell-averaged velocity at the previous time step should now be obvious. These terms are linear, and hence allow the convection term – albeit a linear one – to contribute to the homogeneous solution of the transverse averaged momentum equations. A brief discussion of the treatment of the nonlinear term in nodal analytical schemes is given in the following section. This section can be skipped without any loss of continuity.

3.2. Discussion of the treatment of the nonlinear terms

First nodal integral scheme for the Navier–Stokes equations [9] was developed with only the diffusion terms contributing to the homogeneous solutions of the transverse-averaged differential equations. Hence, except for the diffusion term, all other terms in the momentum equations were lumped in the pseudo-source terms. See, for example, Eqs. (2.5) and (2.8). Cognizant of the advantages in obtaining homogeneous solution of the transverse-integrated momentum equations that locally capture the diffusion as well as the convection process, it is desirable, when the transverse averaged equations are split for the homogenous and particular components of the solution, to retain the convection terms on the left-hand side. However, convection terms, being nonlinear, do not lend themselves easily to analytical solutions. Hence, following the procedure for the convection–diffusion equation – in which the velocity field is assumed known, and

therefore the convection term, being linear, can be retained on the left-hand side [13] – a modified nodal scheme for the 1-D Burgers' equation was developed by approximating the nonlinear convection term $u\partial u/\partial x$ by $u_0\partial u/\partial x$, where the velocity u_0 is the (unknown) cell-averaged u velocity at the current time step. The nonlinearity was resolved through an iterative process. However, this approach was computationally expensive since the unknown cell-averaged velocities, u_0 and v_0, appear as argument of exponential functions that must be repeatedly evaluated during the iteration process.

To avoid this computational overhead, the scheme was further modified, and also applied to the 2-D Burgers' equation [15]. To reduce the computational burden, convection terms based on cell-averaged velocities at the *previous* time step are added to both sides of the transverse-integrated momentum equations [15]. For example, the term $v_p\partial u/\partial y$ is added to both sides of the u momentum equation before it is transverse-integrated in the x and t directions, where v_p is the cell-averaged v velocity at the *previous* time step. The nonlinear term, $\int\int v\partial u/\partial y\,dx\,dt$, is moved to the right-hand side and lumped into a modified pseudo-source term. This is the procedure followed for the Navier–Stokes equations in the previous section. It led to Eqs. (3.16), (3.17), (3.19) and (3.20), which can be solved analytically within each cell. Solutions of these equations for the cell-interior variations of the velocity are of *constant + linear + exponential* form. Clearly, this functional dependence can more accurately capture a wider range of cell-interior variations than the quadratic variation that results when all the convection terms are lumped into the pseudo-source term. The coefficients in the resulting scheme depend on the velocities u_p and v_p, and thus the scheme possesses inherent upwinding. Upwinding, however, is based on velocities at the *previous* time step. Thus, by introducing the cell-averaged velocities at the previous time step, the exponentials need to be evaluated only once for each time step rather than once every iteration, which significantly reduces the computational burden [15].

3.3. Transverse-averaged ODEs

The final set of eight transverse-integrated ordinary differential equations is:

$$\frac{d^2\bar{p}^{xt}(y)}{dy^2} = \bar{S}_1^{xt}, \tag{3.28}$$

$$\frac{d^2\bar{p}^{yt}(x)}{dx^2} = \bar{S}_1^{yt}, \tag{3.29}$$

$$v_p\frac{d\bar{u}^{xt}(y)}{dy} - v\frac{d^2\bar{u}^{xt}(y)}{dy^2} = \bar{S}_2^{xt}, \tag{3.30}$$

$$v_p\frac{d\bar{v}^{xt}(y)}{dy} - v\frac{d^2\bar{v}^{xt}(y)}{dy^2} = \bar{S}_3^{xt}, \tag{3.31}$$

$$u_p\frac{d\bar{u}^{yt}(x)}{dx} - v\frac{d^2\bar{u}^{yt}(x)}{dx^2} = \bar{S}_2^{yt}, \tag{3.32}$$

$$u_p\frac{d\bar{v}^{yt}(x)}{dx} - v\frac{d^2\bar{v}^{yt}(x)}{dx^2} = \bar{S}_3^{yt}, \tag{3.33}$$

$$\frac{d\bar{u}^{xy}(t)}{dt} = \bar{S}_2^{xy} \tag{3.34}$$

and

$$\frac{\mathrm{d}\bar{v}^{xy}(t)}{\mathrm{d}t} = \bar{S}_3^{xy}, \tag{3.35}$$

where $-a_i \leqslant x \leqslant a_i,\ -b_j \leqslant y \leqslant b_j$ and $-\tau_k \leqslant t \leqslant \tau_k$; cell-specific subscripts (i, j, k) have been omitted; and the right-hand sides represent the truncated expansions of the pseudo-source terms. Complete symmetry exists between u and v velocities, and between x and y directions in this formulation.

3.4. Local solutions

These ODEs are solved locally within each cell. The local solution of the ODEs for transverse-integrated pressure is quadratic. For example, the solution of Eq. (3.28) is

$$\bar{p}^{xt}(y) = \frac{\bar{S}_1^{xt}}{2} y^2 + C_1 y + C_2. \tag{3.36}$$

A similar solution can be written for $\bar{p}^{yt}(x)$. The local solution for $\bar{u}^{xt}(y)$ (and other transverse-integrated velocities $(\bar{u}^{yt}(x),\ \bar{v}^{xt}(y),\ \bar{v}^{yt}(x))$ is of the form

$$\bar{u}^{xt}(y) = C_3\, \mathrm{e}^{\{v_{\mathrm{p}} y\}/\nu} + \bar{S}_2^{xt} y + C_4. \tag{3.37}$$

The solutions for $\bar{u}^{xy}(t)$ and $\bar{v}^{xy}(t)$ are linear in time. For example,

$$\bar{u}^{xy}(t) = \bar{S}_2^{xy} t + C_5. \tag{3.38}$$

Recognizing that the discrete unknowns associated with the cell (i, j, k) are going to be the surface-averaged variables on cell surfaces, the constants C_i $(i = 1, 2, \ldots)$ in the above equations are eliminated in favor of these discrete unknowns by imposing boundary conditions, or initial conditions, on cell surfaces normal to the independent variable. For example, boundary conditions for Eq. (3.36) are

$$\bar{p}_{i,j,k}^{xt}(y = +b_j) = \bar{p}_{i,j,k}^{xt}, \quad \bar{p}_{i,j,k}^{xt}(y = -b_j) = \bar{p}_{i,j-1,k}^{xt}. \tag{3.39a}$$

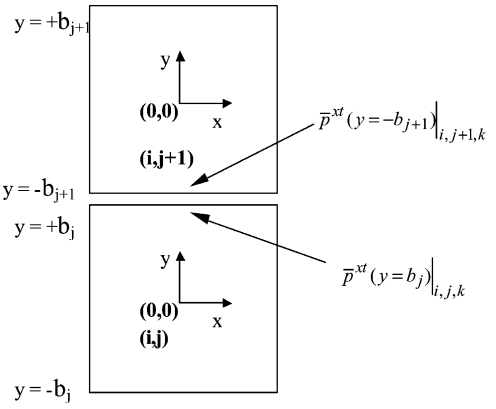
See Fig. 2. The resulting expressions for $\bar{p}_{i,j}^{xt}(y),\ \bar{u}_{i,j}^{xt}(y),\ \bar{v}_{i,j}^{xt}(y),\ \bar{u}_{i,j}^{xy}(t)$ and $\bar{v}_{i,j}^{xy}(t)$ are

$$\bar{p}_{i,j}^{xt}(y) = \frac{\bar{S}_{1i,j}^{xt}}{2}\left(y^2 - b_j^2\right) + \frac{1}{2b_j}(\bar{p}_{i,j}^{xt} - \bar{p}_{i,j-1}^{xt}) y + \frac{1}{2}(\bar{p}_{i,j}^{xt} + \bar{p}_{i,j-1}^{xt}), \tag{3.39b}$$

$$\begin{aligned}\bar{u}_{i,j}^{xt}(y) = {} & \frac{\mathrm{e}^{(1/2)Rev_{i,j}}(-2b_j\bar{S}_{2i,j}^{xt} + \bar{u}_{i,j}^{xt}v_{pi,j} - \bar{u}_{i,j-1}^{xt}v_{pi,j})}{v_{pi,j}(-1 + \mathrm{e}^{Rev_{i,j}})} \exp\left\{\frac{v_{pi,j}y}{\nu}\right\} + \frac{1}{v_{pi,j}}\bar{S}_{2i,j}^{xt} y \\ & + \frac{b_j\bar{S}_{2i,j}^{xt}(1 + \mathrm{e}^{Rev_{i,j}}) - \bar{u}_{i,j}^{xt}v_{pi,j} + \bar{u}_{i,j-1}^{xt}v_{pi,j}\mathrm{e}^{Rev_{i,j}}}{v_{pi,j}(-1 + \mathrm{e}^{Rev_{i,j}})},\end{aligned} \tag{3.40}$$

$$\begin{aligned}\bar{v}_{i,j}^{xt}(y) = {} & \frac{\mathrm{e}^{(1/2)Rev_{i,j}}(-2b_j\bar{S}_{3i,j}^{xt} + \bar{v}_{i,j}^{xt}v_{pi,j} - \bar{v}_{i,j-1}^{xt}v_{pi,j})}{v_{pi,j}(-1 + \mathrm{e}^{Rev_{i,j}})} \exp\left\{\frac{v_{pi,j}y}{\nu}\right\} + \frac{1}{v_{pi,j}}\bar{S}_{3i,j}^{xt} y \\ & + \frac{b_j\bar{S}_{3i,j}^{xt}(1 + \mathrm{e}^{Rev_{i,j}}) - \bar{v}_{i,j}^{xt}v_{pi,j} + \bar{v}_{i,j-1}^{xt}v_{pi,j}\mathrm{e}^{Rev_{i,j}}}{v_{pi,j}(-1 + \mathrm{e}^{Rev_{i,j}})},\end{aligned} \tag{3.41}$$

$$\bar{u}^{xy}(t) = \bar{S}_2^{xy}(t + \tau) + \bar{u}_{k-1}^{xy}, \tag{3.42}$$



$$\bar{p}^{xt}(y=b_j)\big|_{i,j,k} = \bar{p}^{xt}(y=-b_{j+1})\big|_{i,j+1,k} \equiv \bar{p}^{xt}_{i,j,k}$$

Fig. 2. Continuity of transverse averaged pressure between cell (i,j) and $(i,j+1)$.

and

$$\bar{v}^{xy}(t) = \bar{S}_3^{xy}(t+\tau) + \bar{v}^{xy}_{k-1}, \tag{3.43}$$

where the local Reynolds number in the y direction is defined as

$$Rev_{i,j} \equiv \frac{2b_j v_{pi,j}}{v}, \tag{3.44}$$

and the subscript k for current time step variables has been omitted. Solution for $\bar{p}^{yt}_{i,j}(x)$, $\bar{u}^{yt}_{i,j}(x)$ and $\bar{v}^{yt}_{i,j}(x)$ can be obtained by interchanging x and y, replacing b_j by a_i, $v_{pi,j}$ by $u_{pi,j}$, $Rev_{i,j}$ by $Reu_{i,j}$ and $(i,j-1)$ by $(i-1,j)$ in Eqs. (3.39b), (3.40) and (3.41), respectively. Similar to $Rev_{i,j}$, $Reu_{i,j}$ is defined as

$$Reu_{i,j} \equiv \frac{2a_i u_{pi,j}}{v}. \tag{3.45}$$

This completes steps (a) and (b) discussed in Section 2.1. Local solutions obtained above are used in the following section to derive the set of discrete equations (step c).

3.5. Set of discrete equations in terms of the pseudo-source terms

A set of discrete equations is obtained by imposing continuity at cell interfaces – C^0 for first order, and C^1 for second-order ODEs. For the first-order ODEs for $\bar{u}^{xy}_{i,j,k}$ and $\bar{v}^{xy}_{i,j,k}$, the algebraic equations are obtained by simply evaluating the local solutions for $\bar{u}^{xy}(t)$ and $\bar{v}^{xy}(t)$ – Eqs. (3.42) and (3.43) – at $t=\tau$. For the second-order ODEs, C^0 continuity of the transverse averaged variable is (automatically) imposed by simply using the same notation to identify the discrete variable at the interface between the two neighboring cells. For example, for transverse averaged pressure between cells (i,j,k) and $(i,j+1,k)$ this means

$$\bar{p}^{xt}_{i,j,k}(y=b_j)\Big|_{i,j,k} = \bar{p}^{xt}_{i,j+1,k}(y=-b_{j+1})\Big|_{i,j+1,k} \equiv \bar{p}^{xt}_{i,j,k}. \tag{3.46}$$

See Fig. 2 for details. Then, imposing the continuity of the derivative (C^1) at the cell interfaces yields a three-point scheme. Again, for transverse averaged pressure between cells (i,j,k) and $(i,j+1,k)$, this leads to

$$\left.\frac{\mathrm{d}\bar{p}^{xt}_{i,j,k}}{\mathrm{d}y}(y=b_j)\right|_{i,j,k} = \left.\frac{\mathrm{d}\bar{p}^{xt}_{i,j+1,k}}{\mathrm{d}y}(y=-b_{j+1})\right|_{i,j+1,k}. \tag{3.47}$$

Eq. (3.47) is the discrete equation for pressure $\bar{p}^{xt}_{i,j,k}$. (See Eq. (3.48) below). Repeating the same process for the other variables, a total of six coupled, algebraic equations per cell for $\bar{u}^{xt}_{i,j,k}$, $\bar{v}^{xt}_{i,j,k}$, $\bar{p}^{xt}_{i,j,k}$, $\bar{u}^{yt}_{i,j,k}$, $\bar{v}^{yt}_{i,j,k}$, $\bar{p}^{yt}_{i,j,k}$ are derived in terms of the pseudo-source terms, S's.

Eight discrete algebraic equations thus obtained are

$$\frac{(b_j+b_{j+1})}{2b_jb_{j+1}}\bar{p}^{xt}_{i,j} - \frac{1}{2b_j}\bar{p}^{xt}_{i,j-1} - \frac{1}{2b_{j+1}}\bar{p}^{xt}_{i,j+1} + b_j\bar{S}^{xt}_{1i,j} + b_{j+1}\bar{S}^{xt}_{1i,j+1} = 0, \tag{3.48}$$

$$\frac{(a_i+a_{i+1})}{2a_ia_{i+1}}\bar{p}^{yt}_{i,j} - \frac{1}{2a_i}\bar{p}^{yt}_{i-1,j} - \frac{1}{2a_{i+1}}\bar{p}^{yt}_{i+1,j} + a_i\bar{S}^{yt}_{1i,j} + a_{i+1}\bar{S}^{yt}_{1i+1,j} = 0, \tag{3.49}$$

$$A_{21}\bar{u}^{xt}_{i,j-1} + (A_{21}+A_{22})\bar{u}^{xt}_{i,j} + A_{22}\bar{u}^{xt}_{i,j+1} + A_{23}\bar{S}^{xt}_{2i,j} + A_{24}\bar{S}^{xt}_{2i,j+1} = 0, \tag{3.50}$$

$$A_{21}\bar{v}^{xt}_{i,j-1} + (A_{21}+A_{22})\bar{v}^{xt}_{i,j} + A_{22}\bar{v}^{xt}_{i,j+1} + A_{23}\bar{S}^{xt}_{3i,j} + A_{24}\bar{S}^{xt}_{3i,j+1} = 0, \tag{3.51}$$

$$A_{51}\bar{u}^{yt}_{i-1,j} + (A_{51}+A_{52})\bar{u}^{yt}_{i,j} + A_{52}\bar{u}^{yt}_{i+1,j} + A_{53}\bar{S}^{yt}_{2i,j} + A_{54}\bar{S}^{yt}_{2i+1,j} = 0, \tag{3.52}$$

$$A_{51}\bar{v}^{yt}_{i-1,j} + (A_{51}+A_{52})\bar{v}^{yt}_{i,j} + A_{52}\bar{v}^{yt}_{i+1,j} + A_{53}\bar{S}^{yt}_{3i,j} + A_{54}\bar{S}^{yt}_{3i+1,j} = 0, \tag{3.53}$$

$$\bar{u}^{xy}_{i,j} - \bar{u}^{xy}_{i,j,k-1} - 2\tau\bar{S}^{xy}_{2i,j} = 0, \tag{3.54}$$

$$\bar{v}^{xy}_{i,j} - \bar{v}^{xy}_{i,j,k-1} - 2\tau\bar{S}^{xy}_{3i,j} = 0, \tag{3.55}$$

where, once again, the subscript k for current time step variables has been omitted, $k-1$ denotes the previous time-step values, and $A_{21}, A_{22}, \ldots$, and $A_{51}, A_{52}, \ldots$, are coefficients which are functions of a_i, b_j, v, $Reu_{i,j}$ and $Rev_{i,j}$. For example,

$$A_{21} = \frac{v_{pi,j}}{v(\mathrm{e}^{-Rev_{i,j}}-1)}; \quad A_{23} = \frac{2b_j\,\mathrm{e}^{Rev_{i,j}}}{v(1-\mathrm{e}^{Rev_{i,j}})} + \frac{1}{v_{pi,j}}. \tag{3.56}$$

Three characteristics of the numerical scheme being developed can be identified at this stage. First, the local solution of transverse averaged velocities has a component that varies exponentially in space. These exponential terms can capture steep spatial variation of velocities within each cell, thus, allowing the use of coarse meshes. Second, because of the appearance of the local Reynolds number in the exponential terms, the scheme being developed has inherent upwinding [15]. Third, local Reynolds number based only on cell-averaged velocities at the *previous* time step appear as argument of the exponential terms. Hence, these terms can be evaluated at the beginning of each time step outside the iteration loop, which significantly reduces the computation time.

3.6. Constraint equations

Eight discrete algebraic equations (3.48)–(3.55) per cell, given in the last section, are in terms of 16 unknowns: $\bar{u}^{xt}_{i,j}$, $\bar{v}^{xt}_{i,j}$, $\bar{p}^{xt}_{i,j}$, $\bar{u}^{yt}_{i,j}$, $\bar{v}^{yt}_{i,j}$, $\bar{p}^{yt}_{i,j}$, $\bar{u}^{xy}_{i,j}$, $\bar{v}^{xy}_{i,j}$, $\bar{S}^{xt}_{1i,j}$, $\bar{S}^{yt}_{1i,j}$, $\bar{S}^{xt}_{2i,j}$, $\bar{S}^{yt}_{2i,j}$, $\bar{S}^{xy}_{2i,j}$, $\bar{S}^{xt}_{3i,j}$, $\bar{S}^{yt}_{3i,j}$, $\bar{S}^{xy}_{3i,j}$. Thus, eight more

equations are needed to close the set of equations (step d). Following the *nodal approach* [9], these are developed next using eight *constraint equations*. Three constraint equations are obtained by ensuring that the continuity and the momentum equations are satisfied over each cell in an integral sense. Applying the cell-averaging operator,

$$\frac{1}{8a_ib_j\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-b_j}^{b_j}\int_{-a_i}^{a_i}\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}t,$$

on Eqs. (3.4), (3.6) and (3.7), respectively, yields

$$\bar{S}_1^{xt}+\bar{S}_1^{yt}+f_1=0, \tag{3.57}$$

$$\bar{S}_2^{xt}+\bar{S}_2^{yt}+\bar{S}_2^{xy}+f_2=0 \tag{3.58}$$

and

$$\bar{S}_3^{xt}+\bar{S}_3^{yt}+\bar{S}_3^{xy}+f_3=0, \tag{3.59}$$

where

$$f_1\equiv\rho\left(\frac{\bar{u}_{i,j}^{yt}-\bar{u}_{i-1,j}^{yt}}{2a_i}\right)^2+2\rho\left(\frac{\bar{v}_{i,j}^{yt}-\bar{v}_{i-1,j}^{yt}}{2a_i}\right)\left(\frac{\bar{u}_{i,j}^{xt}-\bar{u}_{i,j-1}^{xt}}{2b_j}\right)+\rho\left(\frac{\bar{v}_{i,j}^{xt}-\bar{v}_{i,j-1}^{xt}}{2b_j}\right)^2$$
$$+\rho\left(\frac{\bar{b}_{yi,j}^{xt}-\bar{b}_{yi,j-1}^{xt}}{2b_j}\right)+\rho\left(\frac{\bar{b}_{xi,j}^{yt}-\bar{b}_{xi-1,j}^{yt}}{2a_i}\right), \tag{3.60}$$

$$f_2\equiv(u_0-u_\mathrm{p})\frac{1}{2a_i}(\bar{u}_{i,j}^{yt}-\bar{u}_{i-1,j}^{yt})+(v_0-v_\mathrm{p})\frac{1}{2b_j}(\bar{u}_{i,j}^{xt}-\bar{u}_{i,j-1}^{xt})+\frac{1}{\rho}\frac{1}{2a_i}(\bar{p}_{i,j}^{yt}-\bar{p}_{i-1,j}^{yt})+b_x^{xyt} \tag{3.61}$$

and

$$f_3\equiv(u_0-u_\mathrm{p})\frac{1}{2a_i}(\bar{v}_{i,j}^{yt}-\bar{v}_{i-1,j}^{yt})+(v_0-v_\mathrm{p})\frac{1}{2b_j}(\bar{v}_{i,j}^{xt}-\bar{v}_{i,j-1}^{xt})+\frac{1}{\rho}\frac{1}{2b_j}(\bar{p}_{i,j}^{xt}-\bar{p}_{i,j-1}^{xt})+b_y^{xyt} \tag{3.62}$$

and the following approximation

$$\frac{1}{8a_ib_j\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-b_j}^{b_j}\int_{-a_i}^{a_i}\phi(x,y,t)\psi(x,y,t)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}t$$
$$\approx\left(\frac{1}{8a_ib_j\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-b_j}^{b_j}\int_{-a_i}^{a_i}\phi(x,y,t)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}t\right)\left(\frac{1}{8a_ib_j\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-b_j}^{b_j}\int_{-a_i}^{a_i}\psi(x,y,t)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}t\right) \tag{3.63}$$

has been used to arrive at Eqs. (3.60)–(3.62). Approximating the average of the product by product of the averages, as above, is known to lead to a second-order error in the numerical scheme [9]. For example, $v(x,y,t)(\partial u(x,y,t)/\partial y)$ is locally averaged over x and t as

$$\frac{1}{4a_i\tau_k}\int_{-\tau_k}^{\tau_k}\int_{-a_i}^{a_i}v(x,y,t)\frac{\partial u(x,y,t)}{\partial y}\,\mathrm{d}x\,\mathrm{d}t\approx\bar{v}^{xt}(y)\frac{\mathrm{d}\bar{u}^{xt}(y)}{\mathrm{d}y}\approx v_0\frac{\mathrm{d}\bar{u}^{xt}(y)}{\mathrm{d}y}, \tag{3.64}$$

where v_0 is the *current* time step cell-averaged velocity. The other five constraint equations are obtained by imposing the condition that the cell-averaged variables be unique, independent of the order of integration, i.e.,

$$\bar{\bar{u}}^{xty} \equiv \frac{1}{2b_j}\int_{-b_j}^{b_j} \bar{u}^{xt}(y)\,\mathrm{d}y = \frac{1}{2\tau_k}\int_{-\tau_k}^{\tau_k} \bar{u}^{xy}(t)\,\mathrm{d}t \equiv \bar{u}^{xyt}, \tag{3.65}$$

$$\bar{\bar{u}}^{ytx} \equiv \frac{1}{2a_i}\int_{-a_i}^{a_i} \bar{u}^{yt}(x)\,\mathrm{d}x = \frac{1}{2\tau_k}\int_{-\tau_k}^{\tau_k} \bar{u}^{xy}(t)\,\mathrm{d}t \equiv \bar{u}^{xyt}, \tag{3.66}$$

$$\bar{\bar{v}}^{xty} \equiv \frac{1}{2b_j}\int_{-b_j}^{b_j} \bar{v}^{xt}(y)\,\mathrm{d}y = \frac{1}{2\tau_k}\int_{-\tau_k}^{\tau_k} \bar{v}^{xy}(t)\,\mathrm{d}t \equiv \bar{v}^{xyt}, \tag{3.67}$$

$$\bar{\bar{v}}^{ytx} \equiv \frac{1}{2a_i}\int_{-a_i}^{a_i} \bar{v}^{yt}(x)\,\mathrm{d}x = \frac{1}{2\tau_k}\int_{-\tau_k}^{\tau_k} \bar{v}^{xy}(t)\,\mathrm{d}t \equiv \bar{v}^{xyt}, \tag{3.68}$$

$$\bar{\bar{p}}^{ytx} \equiv \frac{1}{2a_i}\int_{-a_i}^{a_i} \bar{p}^{yt}(x)\,\mathrm{d}x = \frac{1}{2b_j}\int_{-b_j}^{b_j} \bar{p}^{xt}(y)\,\mathrm{d}y \equiv \bar{\bar{p}}^{xty}. \tag{3.69}$$

These constraint conditions are simplified after substituting the local solutions given by Eqs. (3.39a), (3.39b), (3.40)–(3.43), and the corresponding expressions for the other dependent variables. For example, Eq. (3.65) yields

$$\begin{aligned}\bar{S}^{xt}_{2i,j}\left(\frac{b_j}{v_{pi,j}}\frac{1+\mathrm{e}^{Rev_{i,j}}}{\mathrm{e}^{Rev_{i,j}}-1}-\frac{1}{v^2_{pi,j}}\right)-\bar{S}^{xy}_{2i,j}\tau-\bar{u}^{xy}_{i,j,k-1}+\bar{u}^{xt}_{i,j-1}\left(\frac{\mathrm{e}^{Rev_{i,j}}}{-1+\mathrm{e}^{Rev_{i,j}}}-\frac{1}{Rev_{i,j}}\right)\\ +\bar{u}^{xt}_{i,j}\left(1-\left(\frac{\mathrm{e}^{Rev_{i,j}}}{-1+\mathrm{e}^{Rev_{i,j}}}-\frac{1}{Rev_{i,j}}\right)\right)=0.\end{aligned} \tag{3.70}$$

This can be rewritten as

$$A_{91}\bar{S}^{xt}_{2i,j}-\bar{S}^{xy}_{2i,j}\tau-\bar{u}^{xy}_{i,j,k-1}+A_{92}\bar{u}^{xt}_{i,j-1}+(1-A_{92})\bar{u}^{xt}_{i,j}=0, \tag{3.71}$$

where the definitions of A_{91} and A_{92} are obvious from the comparison of Eqs. (3.70) and (3.71). Similarly, Eqs. (3.66)–(3.68) yield the following equations:

$$A_{a1}\bar{S}^{yt}_{2i,j}-\bar{S}^{xy}_{2i,j}\tau-\bar{u}^{xy}_{i,j,k-1}+A_{a2}\bar{u}^{yt}_{i-1,j}+(1-A_{a2})\bar{u}^{yt}_{i,j}=0, \tag{3.72}$$

$$A_{91}\bar{S}^{xt}_{3i,j}-\bar{S}^{xy}_{3i,j}\tau-\bar{v}^{xy}_{i,j,k-1}+A_{92}\bar{v}^{xt}_{i,j-1}+(1-A_{92})\bar{v}^{xt}_{i,j}=0, \tag{3.73}$$

$$A_{a1}\bar{S}^{yt}_{2i,j}-\bar{S}^{xy}_{2i,j}\tau-\bar{v}^{xy}_{i,j,k-1}+A_{a2}\bar{v}^{yt}_{i-1,j}+(1-A_{a2})\bar{v}^{yt}_{i,j}=0, \tag{3.74}$$

where A_{a1}, A_{a2} have definitions similar to A_{91} and A_{92}. The uniqueness of pressure, Eq. (3.69), has the following form:

$$\frac{1}{2}\bar{p}^{xt}_{i,j}+\frac{1}{2}\bar{p}^{xt}_{i,j-1}-\frac{1}{2}\bar{p}^{yt}_{i,j}-\frac{1}{2}\bar{p}^{yt}_{i-1,j}-\frac{b_j^2}{3}\bar{S}^{xt}_{1i,j}+\frac{a_i^2}{3}\bar{S}^{yt}_{1i,j}=0. \tag{3.75}$$

Thus, a total of 16 algebraic equations are derived for 16 unknowns for each cell: eight from the continuity of transverse-integrated variables and their derivatives [Eqs. (3.48)–(3.55)]; three from the cell-averaged conservation equations [Eqs. (3.57)–(3.59)]; and five from the uniqueness conditions [Eqs. (3.71)–(3.75)]. The pseudo-source terms are eliminated next from this set, leaving only eight physically relevant unknowns and eight equations per cell.

3.7. Set of discrete equations

The final set of eight, discrete, algebraic equations are:

$$F_{17}\bar{p}^{xt}_{i,j} = F_{11}\bar{p}^{xt}_{i,j-1} + F_{12}\bar{p}^{xt}_{i,j+1} + F_{13}(\bar{p}^{yt}_{i,j} + \bar{p}^{yt}_{i-1,j}) + F_{14}(\bar{p}^{yt}_{i,j+1} + \bar{p}^{yt}_{i-1,j+1}) + F_{15}f_{1i,j} + F_{16}f_{1i,j+1}, \tag{3.76}$$

$$F_{27}\bar{p}^{yt}_{i,j} = F_{21}\bar{p}^{yt}_{i-1,j} + F_{22}\bar{p}^{yt}_{i+1,j} + F_{23}(\bar{p}^{xt}_{i,j} + \bar{p}^{xt}_{i,j-1}) + F_{24}(\bar{p}^{xt}_{i+1,j} + \bar{p}^{xt}_{i+1,j-1}) + F_{25}f_{1i,j} + F_{26}f_{1i+1,j}, \tag{3.77}$$

$$F_{37}\bar{u}^{xt}_{i,j} = F_{31}\bar{u}^{xt}_{i,j-1} + F_{32}\bar{u}^{xt}_{i,j+1} + F_{33}(\bar{u}^{xy}_{i,j} + \bar{u}^{xy}_{i,j,k-1}) + F_{34}(\bar{u}^{xy}_{i,j+1} + \bar{u}^{xy}_{i,j+1,k-1}), \tag{3.78}$$

$$F_{37}\bar{v}^{xt}_{i,j} = F_{31}\bar{v}^{xt}_{i,j-1} + F_{32}\bar{v}^{xt}_{i,j+1} + F_{33}(\bar{v}^{xy}_{i,j} + \bar{v}^{xy}_{i,j,k-1}) + F_{34}(\bar{v}^{xy}_{i,j+1} + \bar{v}^{xy}_{i,j+1,k-1}), \tag{3.79}$$

$$F_{57}\bar{u}^{yt}_{i,j} = F_{51}\bar{u}^{yt}_{i-1,j} + F_{52}\bar{u}^{yt}_{i+1,j} + F_{53}(\bar{u}^{xy}_{i,j} + \bar{u}^{xy}_{i,j,k-1}) + F_{54}(\bar{u}^{xy}_{i+1,j} + \bar{u}^{xy}_{i+1,j,k-1}), \tag{3.80}$$

$$F_{57}\bar{v}^{yt}_{i,j} = F_{51}\bar{v}^{yt}_{i-1,j} + F_{52}\bar{v}^{yt}_{i+1,j} + F_{53}(\bar{v}^{xy}_{i,j} + \bar{v}^{xy}_{i,j,k-1}) + F_{54}(\bar{v}^{xy}_{i+1,j} + \bar{v}^{xy}_{i+1,j,k-1}), \tag{3.81}$$

$$F_{77}\bar{u}^{xy}_{i,j} = F_{71}\bar{u}^{xt}_{i,j} + F_{72}\bar{u}^{xt}_{i,j-1} + F_{73}\bar{u}^{xy}_{i,j,k-1} + F_{74}\bar{u}^{yt}_{i,j} + F_{75}\bar{u}^{yt}_{i-1,j} + f_{2i,j}, \tag{3.82}$$

$$F_{77}\bar{v}^{xy}_{i,j} = F_{71}\bar{v}^{xt}_{i,j} + F_{72}\bar{v}^{xt}_{i,j-1} + F_{73}\bar{v}^{xy}_{i,j,k-1} + F_{74}\bar{v}^{yt}_{i,j} + F_{75}\bar{v}^{yt}_{i-1,j} + f_{3i,j}, \tag{3.83}$$

where F's are coefficients that are functions of $a_i,\ b_j,\ \nu,\ \tau,\ Reu_{i,j},\ Rev_{i,j}$ and $A_{i,j}$. For example,

$$F_{11} = \frac{1}{2b_j} + \frac{3b_j}{2(a_i^2 + b_j^2)}, \tag{3.84}$$

$$F_{31} = A_{21} - \frac{A_{23}A_{92}}{A_{91}} \tag{3.85}$$

and

$$F_{73} = \frac{1}{2}\left(\frac{1}{A_{91}} + \frac{1}{A_{a1}} - \frac{1}{\tau}\right). \tag{3.86}$$

The discrete unknowns for the cell (i,j,k) are the variables averaged on the cell surfaces: $\bar{u}^{xt}_{i,j},\ \bar{v}^{xt}_{i,j},\ \bar{p}^{xt}_{i,j},\ \bar{u}^{yt}_{i,j},\ \bar{v}^{yt}_{i,j},\ \bar{p}^{yt}_{i,j},\ \bar{u}^{xy}_{i,j},\ \bar{v}^{xy}_{i,j}$.

4. Boundary conditions

The discrete unknowns in the scheme developed in the previous section are the dependent variables averaged over the surfaces of the space–time cell (i,j,k) in X–Y–T space. Boundary conditions for velocities are relatively straightforward. No slip boundary conditions are imposed on solid surfaces. In addition, Dirichlet condition can also be specified, for example, on inlet surfaces. Nodal scheme developed in the previous section leads to a collocated discretization. Hence, along with boundary conditions on u and v velocities, pressure boundary conditions are also needed.

Boundary conditions for pressure on no-slip surfaces are derived using the x- and y-momentum equations [26–29]. For example, on vertical no-slip surfaces, $u = v = 0,\ \partial u/\partial t = \partial^2 u/\partial y^2 = 0$, and thus, the u-momentum equation, averaged locally over y and t, becomes [29]

$$-\frac{1}{\rho}\frac{\mathrm{d}\bar{p}^{vt}(x)}{\mathrm{d}x}+v\frac{\mathrm{d}^2\bar{u}^{vt}(x)}{\mathrm{d}x^2}+\bar{b}_x^{vt}=0. \tag{4.1}$$

One straightforward approach to derive the discrete form of this boundary condition is to satisfy Eq. (4.1) on the surface of the boundary cell. This can be achieved by substituting in Eq. (4.1) the expressions derived in Section 3.4 for the local solution of transverse-integrated pressure $\bar{p}^{vt}(x)$

$$\bar{p}^{vt}(x)=\frac{\bar{S}_1^{vt}}{2}x^2+B_1x+B_2$$

and transverse-integrated velocity $\bar{u}^{vt}(x)$

$$\bar{u}^{vt}(x)=B_3\mathrm{e}^{u_{\mathrm{P}}x/v}+\bar{S}_2^{vt}x+B_4.$$

Thus, algebraic equations for transverse-averaged pressure on vertical boundaries can be easily obtained. However, local solutions for $\bar{p}^{vt}(x)$ and $\bar{u}^{vt}(x)$ are second-order accurate. Consequently, the second derivative of $\bar{u}^{vt}(x)$ only has zeroth-order accuracy, and the discrete form of the boundary condition will also be only zeroth-order accurate. Hence, to derive a second-order accurate boundary condition for pressure, consistent with the second-order accuracy of the scheme, a fourth-order accurate finite difference expression for $\bar{u}^{vt}(x)$ on the boundaries is used. For a stationary right vertical surface the boundary condition is developed as follows. Let the discrete variable $\bar{u}^{vt}$ on the right surface of a boundary cell (i,j), and on the right surfaces of cells $(i-1,j)$, $(i-2,j)$ and $(i-3,j)$ be represented by $\bar{u}^{vt}(x_0)=\bar{u}_{i,j}^{vt}$, $\bar{u}^{vt}(x_0-h_1)=\bar{u}_{i-1,j}^{vt}$, $\bar{u}^{vt}(x_0-h_1-h_2)=\bar{u}_{i-2,j}^{vt}$ and $\bar{u}^{vt}(x_0-h_1-h_2-h_3)=\bar{u}_{i-3,j}^{vt}$ (see Fig. 3). A second-order accurate discrete approximation for the second derivative at $x=x_0$ is

$$\begin{aligned}\frac{\mathrm{d}^2\bar{u}^{vt}(x=x_0)}{dx^2}=&\frac{2(3h_1+2h_2+h_3)}{h_1(h_1+h_2)(h_1+h_2+h_3)}\bar{u}_{i,j}^{vt}-\frac{2(2h_1+2h_2+h_3)}{h_1h_2(h_2+h_3)}\bar{u}_{i-1,j}^{vt}+\frac{2(2h_1+h_2+h_3)}{h_2(h_1+h_2)h_3}\bar{u}_{i-2,j}^{vt}\\&-\frac{2(2h_1+h_2)}{h_3(h_2+h_3)(h_1+h_2+h_3)}\bar{u}_{i-3,j}^{vt}+\mathrm{O}(h^2).\end{aligned} \tag{4.2}$$

The large template for the second derivative can be reduced by imposing additional conditions. For example, the continuity equation

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0, \tag{4.3}$$

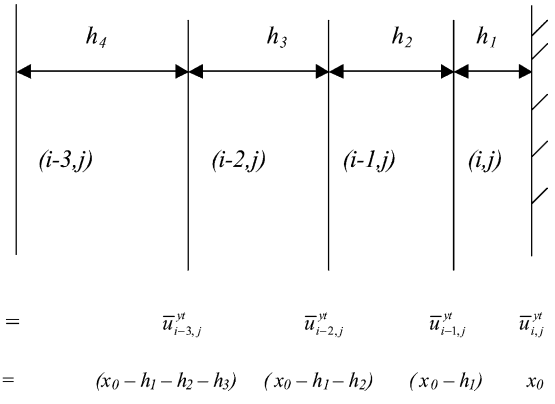


Fig. 3. Boundary condition for pressure at the right surface.

when imposed along a stationary, vertical, no-slip wall, requires

$$\frac{\partial u}{\partial x} = 0. \tag{4.4}$$

Transverse integrating Eq. (4.4) locally over y and t yields

$$\frac{\mathrm{d}\bar{u}^{yt}}{\mathrm{d}x} = 0. \tag{4.5}$$

Using a third-order accurate finite difference expression for the first derivative, Eq. (4.5) becomes

$$\begin{aligned}\frac{\mathrm{d}\bar{u}^{yt}(x=x_0)}{\mathrm{d}x} &= \left(\frac{1}{h_1}+\frac{1}{h_1+h_2}+\frac{1}{h_1+h_2+h_3}\right)\bar{u}^{yt}_{i,j} - \frac{(h_1+h_2)(h_1+h_2+h_3)}{h_1h_2(h_2+h_3)}\bar{u}^{yt}_{i-1,j} \\ &+ \frac{h_1(h_1+h_2+h_3)}{h_2(h_1+h_2)h_3}\bar{u}^{yt}_{i-2,j} - \frac{h_1(h_1+h_2)}{h_3(h_2+h_3)(h_1+h_2+h_3)}\bar{u}^{yt}_{i-3,j} + \mathrm{O}(h^3) = 0.\end{aligned} \tag{4.6}$$

Eq. (4.6) is used to eliminate the discrete variable farthest from the surface ($\bar{u}^{yt}_{i-3,j}$) from the four-point finite difference expression (Eq. (4.2)), resulting in a second-order accurate, three-point scheme for the second derivative at the wall, $\mathrm{d}^2\bar{u}^{yt}(x=x_0)/\mathrm{d}x^2$,

$$\begin{aligned}\left.\frac{\mathrm{d}^2\bar{u}^{yt}}{\mathrm{d}x^2}\right|_{\text{wall}} &= \frac{\mathrm{d}^2\bar{u}^{yt}(x=x_0)}{\mathrm{d}x^2} \\ &= -\frac{2(3h_1^2+3h_1h_2+h_2^2)}{h_1^2(h_1+h_2)^2}\bar{u}^{yt}_{i,j} + \frac{2(h_1+h_2)}{h_1^2h_2}\bar{u}^{yt}_{i-1,j} - \frac{2h_1}{h_2(h_1+h_2)^2}\bar{u}^{yt}_{i-2,j} + \mathrm{O}(h^2).\end{aligned} \tag{4.7}$$

A second-order accurate scheme for the first derivative of pressure at the wall is also developed in a similar manner. [However, the results obtained using the second-order scheme and those obtained using the cell interior expression for $\bar{p}^{yt}(x)$ to evaluate the derivative $\mathrm{d}\bar{p}^{yt}/\mathrm{d}x|_{\text{wall}}$, yield similar numerical results.] These expressions for $\mathrm{d}^2\bar{u}^{yt}/\mathrm{d}x^2$ and $\mathrm{d}\bar{p}^{yt}/\mathrm{d}x$ are substituted in Eq. (4.1) to obtain the discrete form of the pressure boundary condition for $\bar{p}^{yt}_{i,j}$ at the right wall. Pressure boundary conditions for the other walls are similarly derived.

This completes the development of the set of discrete equations for the time-dependent, incompressible Navier–Stokes equations. This set of equations has been implemented in a Fortran code, and tested on several two-dimensional, steady-state and time-dependent fluid flow problems. Steady-state problems are solved by marching in time. Several iterative approaches have been tested to solve the final set of algebraic equations at each time step. Results presented in the next section are based on a Gauss–Seidel iterative procedure in conjunction with a Simple-like algorithm that couples the field variables. That is, for fixed pressure field, velocities are evaluated from bottom left to the top right of the domain, row by row. Next, keeping velocities fixed, discrete pressure values are evaluated (row by row) from lower left of the domain to the top right of the domain. For given velocity field, around 15 pressure-sweeps yield near optimum convergence. (For pressure updates, ADI scheme was tested but was found to be less efficient.) As is the case with many other iterative approaches, for most examples studied here only a single sweep to update the velocity, for given pressure field, was found to be sufficient. Numerical results are reported in the following section.

5. Numerical results

Numerical scheme and the boundary conditions developed in Sections 3 and 4 were coded in Fortran. The code runs on a SUN Ultra II as well as on a PC running LINUX operating system. All

CPU times reported in this section are for simulations carried out on the PC. CPU times for the 1.5 GHz PC were lower by up to a factor of 8 when compared with those for the SUN Ultra II workstation.

Numerous problems, including the classical lid driven cavity problem, have been solved using this modified nodal scheme. Results are presented here for three problems: two, two-dimensional steady-state problems, and a two-dimensional, time-dependent problem. Two of these have exact analytical solutions, and all three have been used extensively by researchers to test numerical schemes developed for the Navier–Stokes equations.

5.1. Classical lid driven cavity problem

In this well-known problem [30], the flow in a square cavity of dimension one on each side is driven by the moving lid. The other three surfaces are at zero velocity. Numerical results obtained over a very fine mesh [31] have been widely used for comparison. For a Reynolds number of 1000, Fig. 4 shows the u velocity component along the vertical line passing through the center of the box. Fine-mesh results of [31] are also plotted. Fig. 5 shows the v velocity component along the horizontal line passing through the center of the box. Also shown are fine-mesh results from [31]. For nodal method, $\bar{u}^{yt}$ and $\bar{v}^{xt}$ values are plotted at the center of the cell. Results are presented for 12×12, 16×16 and 20×20 non-uniform mesh. Non-uniform meshes for this problem were generated using a geometric factor of 1.4 from the center of the cavity toward the wall. Even for as coarse as 12×12 mesh, results match fairly well with the reference data. Results obtained on the 16×16 mesh compare very well with those reported in [32] (30×30 mesh) obtained using a variable explicit/implicit method for unstructured meshes.

Figs. 6 and 7 show the u and v velocities along the vertical and horizontal lines passing through the center of the box for a Reynolds number of 100. Fine mesh results from [31] are also shown. Results presented in Figs. 6 and 7 are obtained using 4×4, 8×8 and 16×16 non-uniform meshes. The geometric factor used was 1.3. Fine mesh results of [31] are also shown. Even the results on a very coarse 4×4 mesh, for this relatively low Reynolds number problem, agree fairly well with the fine-mesh data.

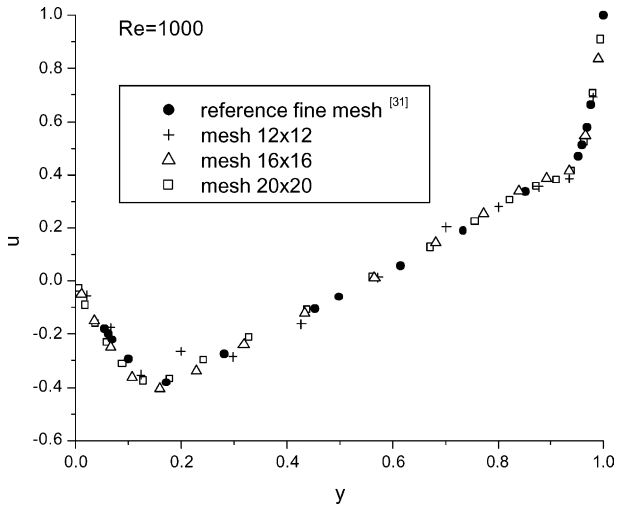


Fig. 4. u-Velocity along the vertical line through geometric center of the cavity for classical lid-driven cavity problem for $Re = 1000$. Fine mesh results are from [31]. Results of the nodal scheme are plotted at the center of the cell.

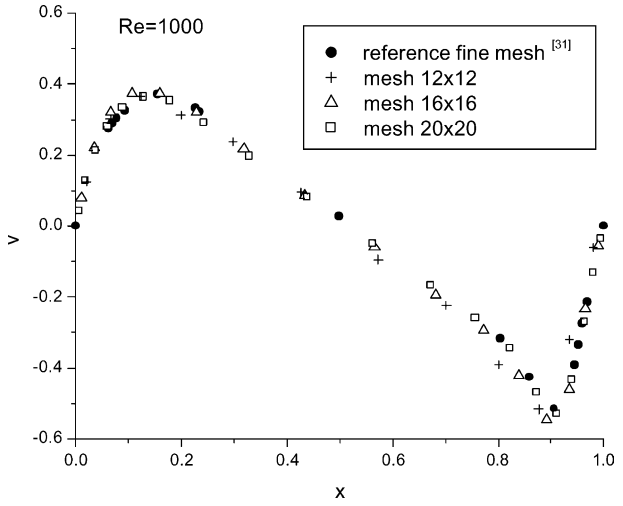


Fig. 5. v-Velocity along the horizontal line through geometric center of the cavity for classical lid-driven cavity problem for $Re = 1000$. Fine mesh results are from [31]. Results of the nodal scheme are plotted at the center of the cell.

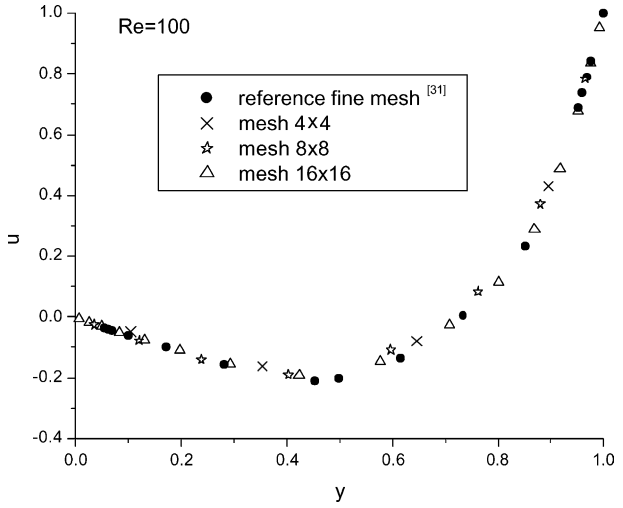


Fig. 6. u-Velocity along the vertical line through geometric center of the cavity for classical lid-driven cavity problem for $Re = 100$. Fine mesh results are from [31]. Results of the nodal scheme are plotted at the center of the cell.

5.2. Modified lid driven cavity problem

A variation of the classical lid driven cavity problem has been proposed by Shin et al. [33]. This problem – here referred to as the *modified lid driven cavity problem* – has an exact analytical solution. The modifications include a lid velocity that varies along the lid, i.e., $u_{\text{lid}} = u(x)$, and space-dependent body forces within the cavity. The fact that the lid velocity is equal to zero at the two corners eliminates the singularity that exists at those two points in the classical lid driven cavity problem. This problem was solved by Shin et al. to compare nine numerical schemes developed for the Navier–Stokes equations [33]. The exact solution is reproduced in Appendix A [33]. The velocity and pressure fields,

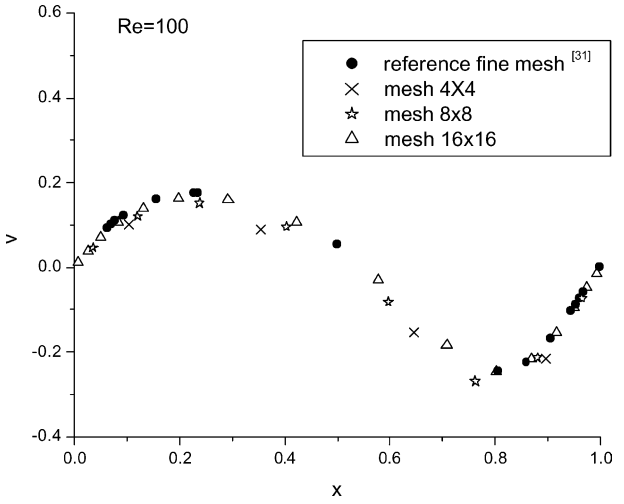


Fig. 7. v-Velocity along the horizontal line through geometric center of the cavity for classical lid-driven cavity problem for $Re = 100$. Fine mesh results are from [31]. Results of the nodal scheme are plotted at the center of the cell.

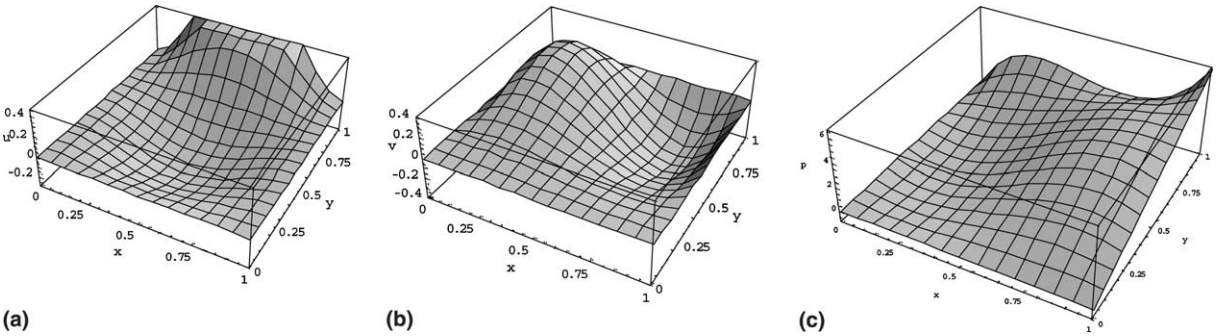


Fig. 8. Velocity and pressure fields of the modified lid driven cavity problem.

$u(x,y)$, $v(x,y)$ and $p(x,y)$, over $0 \leqslant x \leqslant 1$ and $0 \leqslant y \leqslant 1$, are shown in Fig. 8. A vector plot of the velocity field for $Re = 1$ is shown in Fig. 9.

This steady-state problem is also solved by starting from an arbitrary initial condition (zero uniform velocity and zero uniform pressure) and marching in time till steady state is reached. The results for Reynolds number $Re = 1$, 10 and 20 are given in Tables 1–3. These results were obtained with Dirichlet boundary conditions for all variables, including pressure, on all surfaces. RMS errors in $\bar{u}^{xy}$, $\bar{v}^{xy}$, $\bar{p}^{xt}$ and $\bar{p}^{yt}$ are reported for different mesh sizes. CPU times are also reported. For $Re = 1$, even for as coarse as 5×5 uniform mesh, the numerical scheme developed here yields a small RMS error of 0.006 and 0.001 for $\bar{u}^{xy}$ and $\bar{v}^{xy}$; and it takes only 1.2 s of CPU time on the PC. The CPU time is low despite the fact that very simple Gauss–Seidel sweeps are used repeatedly at each time step till convergence. For larger problems, significant savings in CPU time can be achieved by incorporating more efficient solvers.

The near second-order accuracy of the scheme can be seen from the tables, confirming the (at least) second-order nature of the approximations introduced in the development. As the Reynolds number is increased to 10, the RMS error for velocity and pressure (except for $\bar{v}^{xy}$) in general decrease, in some case by as much as a factor of 5. However, the error in cell-averaged v velocity increases. The error in $\bar{u}^{xy}$, $\bar{p}^{xt}$ and $\bar{p}^{yt}$

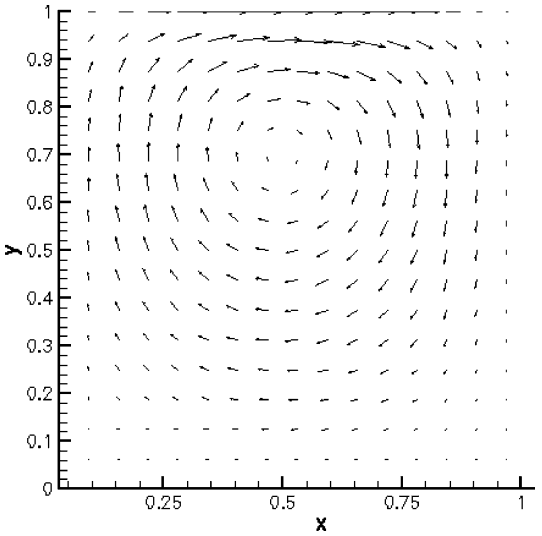


Fig. 9. Velocity vector plot of the modified lid driven cavity problem.

Table 1
RMS errors and CPU times for $Re = 1$ (Dirichlet boundary conditions)

Mesh	$\bar{u}^{xy}$	$\bar{v}^{xy}$	$\bar{p}^{xt}$	$\bar{p}^{yt}$	CPU time (s)
5×5	0.5816×10^{-2}	0.1058×10^{-2}	0.1849×10^{-1}	0.2211×10^{-1}	1.2
10×10	0.1313×10^{-2}	0.2595×10^{-3}	0.5448×10^{-2}	0.5539×10^{-2}	2.2
20×20	0.2779×10^{-3}	0.6216×10^{-4}	0.1703×10^{-2}	0.1709×10^{-2}	31.9

RMS errors and CPU times for three different mesh sizes for the modified lid driven cavity problem with $Re = 1$. Dirichlet boundary conditions for pressure were imposed on all surfaces.

Table 2
RMS errors and CPU times for $Re = 10$ (Dirichlet boundary conditions)

Mesh	$\bar{u}^{xy}$	$\bar{v}^{xy}$	$\bar{p}^{xt}$	$\bar{p}^{yt}$	CPU time (s)
5×5	0.4077×10^{-2}	0.3715×10^{-2}	0.8003×10^{-2}	0.8373×10^{-2}	1.3
10×10	0.9918×10^{-3}	0.7849×10^{-3}	0.1585×10^{-2}	0.1628×10^{-2}	3.3
20×20	0.2511×10^{-3}	0.1879×10^{-3}	0.3740×10^{-3}	0.3752×10^{-3}	37.5

RMS errors and CPU times for three different mesh sizes for the modified lid driven cavity problem with $Re = 10$. Dirichlet boundary conditions for pressure were imposed on all surfaces.

Table 3
RMS errors and CPU times for $Re = 20$ (Dirichlet boundary conditions)

Mesh	$\bar{u}^{xy}$	$\bar{v}^{xy}$	$\bar{p}^{xt}$	$\bar{p}^{yt}$	CPU time (s)
5×5	0.5398×10^{-2}	0.6616×10^{-2}	0.7904×10^{-2}	0.8506×10^{-2}	2.4
10×10	0.1173×10^{-2}	0.1333×10^{-2}	0.1623×10^{-2}	0.1736×10^{-2}	6.4
20×20	0.2946×10^{-3}	0.3119×10^{-3}	0.3758×10^{-3}	0.3771×10^{-3}	85.1

RMS errors and CPU times for three different mesh sizes for the modified lid driven cavity problem with $Re = 20$. Dirichlet boundary conditions for pressure were imposed on all surfaces.

remain roughly the same as the Reynolds number is increased to 20, while the error in $\bar{v}^{xy}$ increases by as much as a factor of almost 2.

The problem was then solved using the pressure boundary conditions developed in Section 4. No-slip boundary conditions were imposed for velocities on all surfaces. The problem was solved for Reynolds number of 1, 10 and 20, on 5×5, 10×10 and 20×20 uniform meshes. See Tables 4–6. The RMS errors for $Re = 1$ are in general higher than the corresponding RMS errors found with Dirichlet boundary conditions for pressure. However, RMS errors in $\bar{u}^{xy}$ are lower by as much as a factor of 2. The RMS errors for velocities are in the range of 0.22×10^{-2}–0.95×10^{-4}. As the Reynolds number is increased to 10 and then 20, RMS error in velocities either remain approximately constant or increase, while RMS errors in pressure decrease significantly. It should be noted that several schemes based on central finite difference approach, tested and reported in [33], failed to converge to the correct solution for $Re > 10$, and those that did, converged to grossly inaccurate solutions [33]. Moreover, RMS errors in the numerical results obtained here are lower than the errors in eight of the nine schemes tested in [33]. Only the results obtained using the 4/4 non-staggered (HO) scheme are comparable with those obtained using the nodal scheme.

5.3. Taylor's decaying vortices

The third problem solved is the time-dependent Taylor's decaying vortices problem [34]. The problem was chosen because it has an exact analytical solution, allowing for accurate error analysis. Kim and Moin [35] utilized the exact solution to test the boundary conditions for the fractional step method. Henriksen

Table 4
RMS errors and CPU times for $Re = 1$ (pressure boundary conditions)

Mesh	$\bar{u}^{xy}$	$\bar{v}^{xy}$	$\bar{p}^{xt}$	$\bar{p}^{yt}$	CPU time (s)
5×5	0.2184×10^{-2}	0.2180×10^{-2}	0.7851×10^{-1}	0.9017×10^{-1}	0.8
10×10	0.5278×10^{-3}	0.4302×10^{-3}	0.1757×10^{-1}	0.1918×10^{-1}	7.56
20×20	0.1251×10^{-3}	0.9451×10^{-4}	0.4193×10^{-2}	0.4404×10^{-2}	133.3

RMS errors and CPU times for three different mesh sizes for the modified lid driven cavity problem with $Re = 1$. Pressure boundary conditions for pressure developed in Section 4 were imposed on all surfaces.

Table 5
RMS errors and CPU times for $Re = 10$ (pressure boundary conditions)

Mesh	$\bar{u}^{xy}$	$\bar{v}^{xy}$	$\bar{p}^{xt}$	$\bar{p}^{yt}$	CPU time (s)
5×5	0.2229×10^{-2}	0.2709×10^{-2}	0.1074×10^{-1}	0.1039×10^{-1}	6.2
10×10	0.5421×10^{-3}	0.5075×10^{-3}	0.2473×10^{-2}	0.2456×10^{-2}	30.3
20×20	0.1400×10^{-3}	0.1171×10^{-3}	0.6152×10^{-3}	0.6045×10^{-3}	375.1

RMS errors and CPU times for three different mesh sizes for the modified lid driven cavity problem with $Re = 10$. Pressure boundary conditions for pressure developed in Section 4 were imposed on all surfaces.

Table 6
RMS errors and CPU times for $Re = 20$ (pressure boundary conditions)

Mesh	$\bar{u}^{xy}$	$\bar{v}^{xy}$	$\bar{p}^{xt}$	$\bar{p}^{yt}$	CPU time (s)
5×5	0.3503×10^{-2}	0.4112×10^{-2}	0.9185×10^{-2}	0.7427×10^{-2}	20.5
10×10	0.7713×10^{-3}	0.7678×10^{-3}	0.2059×10^{-2}	0.1917×10^{-2}	64.0
20×20	0.1964×10^{-3}	0.1794×10^{-3}	0.5078×10^{-3}	0.4808×10^{-3}	582.8

RMS errors and CPU times for three different mesh sizes for the modified lid driven cavity problem with $Re = 20$. Pressure boundary conditions for pressure developed in Section 4 were imposed on all surfaces.

and Holmen [36] used it to test their algebraic splitting scheme for the incompressible Navier–Stokes equations. Quarteroni et al. [37] also tested their factorization methods using the exact solution.

An exact solution of the two-dimensional, time-dependent Navier–Stokes equations, with $\rho = 1$, is given by the stream function [34]

$$\psi(x,y,t) = \frac{\omega}{(k_x^2 + k_y^2)} \exp[-\nu(k_x^2 + k_y^2)t] \cos(k_x x) \cos(k_y y), \tag{5.1}$$

which leads to the following $u(x,y,t)$ and $v(x,y,t)$ velocities:

$$u(x,y,t) = -\frac{\partial \psi}{\partial y} = \frac{-\omega k_y}{(k_x^2 + k_y^2)} \exp[-\nu(k_x^2 + k_y^2)t] \cos(k_x x) \sin(k_y y), \tag{5.2}$$

$$v(x,y,t) = \frac{\partial \psi}{\partial x} = \frac{\omega k_x}{(k_x^2 + k_y^2)} \exp[-\nu(k_x^2 + k_y^2)t] \sin(k_x x) \cos(k_y y), \tag{5.3}$$

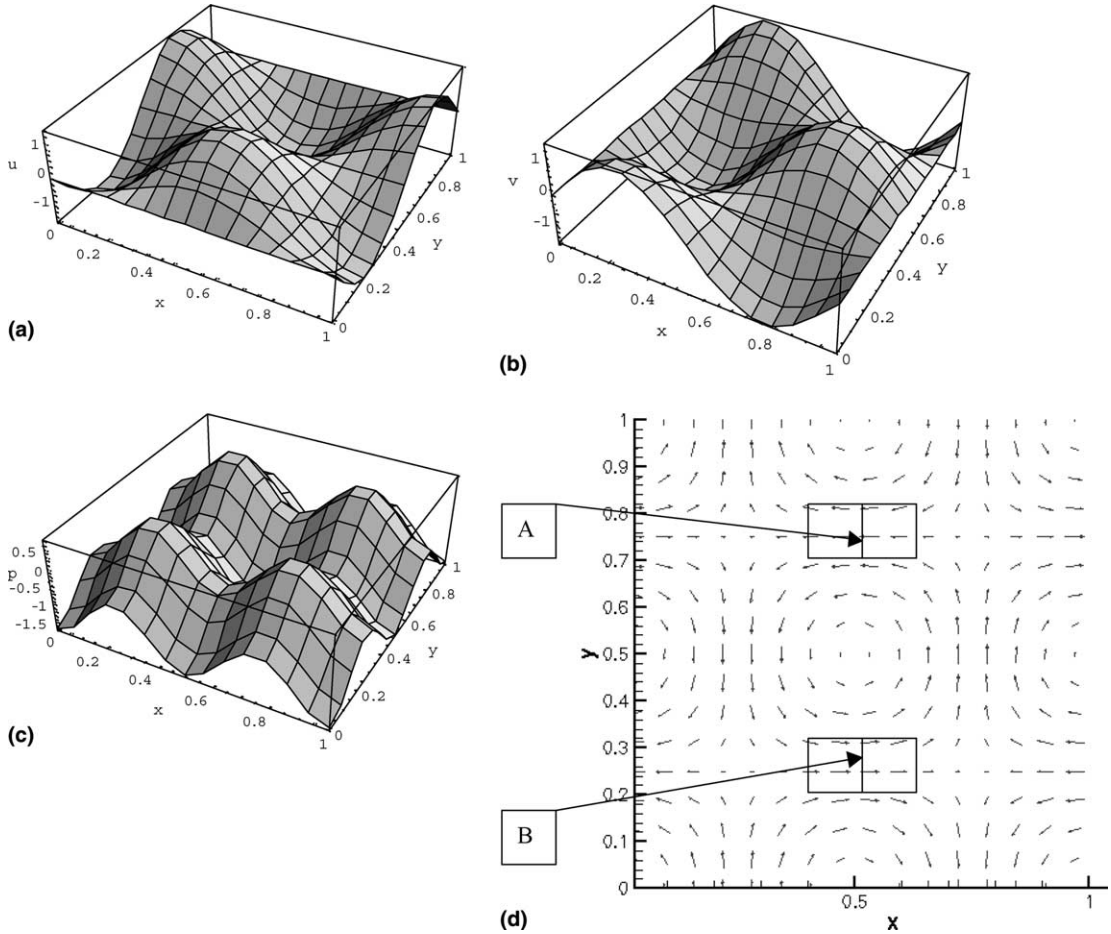


Fig. 10. (a–c) Velocity (u and v) and pressure fields for the Taylor's decaying vortices problem at $t = 0$. (d) Corresponding velocity vector plot. Coefficients of three neighboring discrete variables at two different locations (A and B) are shown in Table 7.

where ω is the initial maximum vorticity, and k_x and k_y are wave numbers. The field represents a decaying system of eddies in a rectangular array rotating alternately in opposite directions. The u and v velocities and pressure at time $t = 0$ are shown in Fig. 10(a)–(c) over $0 \leqslant x \leqslant 1,\ 0 \leqslant y \leqslant 1$. Fig. 10(d) shows the vector plot of the flow field. Parameter values for the flow shown in Fig. 10 are $k_x = k_y = 2\pi,\ \omega = 2\pi^2$ and $v = 1$.

The problem was solved over $0 \leqslant x \leqslant 1,\ 0 \leqslant y \leqslant 1$ with Dirichlet boundary conditions for all variables on all surfaces [34,35]. Numerical results for $\bar{u}^{xt}(y)$ and $\bar{p}^{xt}(y)$ for $0.4375 \leqslant x \leqslant 0.5$ at four different times are compared with the exact solutions in Fig. 11(a) and (b). These results were obtained on a 16×16 uniform mesh, and with $\Delta t = 0.005$. The RMS error at $t = 0.01$ for the 16×16 grid case is 1.1×10^{-3} for $\bar{u}^{xt}$ and 7.4×10^{-3} for $\bar{p}^{xt}$. An even coarser, 8×8 grid, calculation leads to an RMS error of only 6.8×10^{-3} for $\bar{u}^{xt}$ and 2.7×10^{-2} for $\bar{p}^{xt}$ at $t = 0.01$, again showing the near second-order accuracy of the method. In fact, in some cases, the results show a better than second-order accuracy.

The numerical scheme developed here, as was pointed out in Section 3, has "inherent upwinding." That is, based on flow directions at neighboring cells, the velocity coefficients in the discrete algebraic equations are automatically adjusted. This characteristic of the scheme is demonstrated by evaluating the velocity coefficients in the neighboring cells in determining $\bar{u}^{yt}$ values at two different locations in the Taylor's decaying vortices problem. Two neighboring cells at two different locations are shown schematically in Fig. 10(d) and identified by the letters A and B. The flow at A is to the left, and at B it is to the right. The u velocity $\bar{u}^{yt}_{i,j,k}$ at each of these locations is evaluated using Eq. (3.80) in terms of the neighboring velocities $\bar{u}^{yt}_{i-1,j,k}$ and $\bar{u}^{yt}_{i+1,j,k}$ (in addition to other discrete variables). Eq. (3.80) is rewritten as

$$\bar{u}^{yt}_{i,j} = t1\bar{u}^{yt}_{i-1,j} + t2\bar{u}^{yt}_{i,j} + t3\bar{u}^{yt}_{i+1,j} + t4(\bar{u}^{xy}_{i,j} + \bar{u}^{xy}_{i,j,k-1}) + t5(\bar{u}^{xy}_{i+1,j} + \bar{u}^{xy}_{i+1,j,k-1}) \tag{5.4}$$

and the coefficients at $t = 0.01$ (after 20 time steps) are shown in Table 7. These coefficients correspond to the simulation over a 10×10 grid for $v = 0.001$. The magnitude of the coefficients of $\bar{u}^{yt}_{i-1,j,k}$ and $\bar{u}^{yt}_{i+1,j,k}$, $t1$ and $t3$, for the two cases clearly show that the numerical scheme is automatically "weighting" the coefficients as a function of the flow direction. In addition, the magnitudes of the coefficients of the local space-averaged u velocities (averaged over x and $y, \bar{u}^{xy}$) are also adjusted as a result of the flow direction.

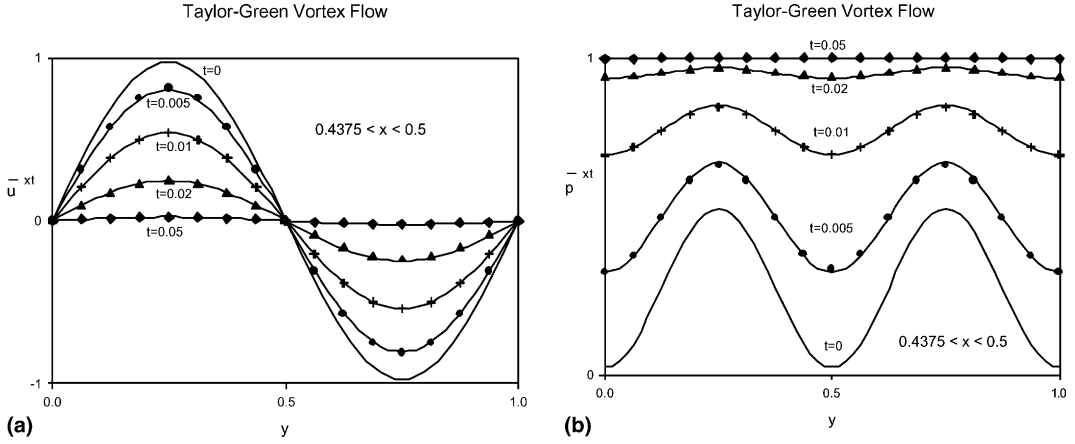


Fig. 11. Numerical and exact solutions of the Taylor's decaying vortices problem at different times. (a) u velocity; (b) pressure.

Table 7
Coefficients of discrete variables showing inherent upwinding

Coeff	A	B
$t1$	0.23	918.56
$t2$	958.78	958.78
$t3$	918.56	0.23
$t4$	−10.23	−928.56
$t5$	−928.56	−10.23

Coefficients of the neighboring discrete variables at two different locations (A and B, shown in Fig. 10(d)) in the Taylor's decaying vortices problem. See Eq. (5.4).

6. Summary and conclusions

A modified nodal integral method has been developed for the time-dependent, Navier–Stokes equation. Pressure is determined by solving a Poisson equation. The scheme is developed by first reducing the Navier–Stokes equations to a set of ordinary differential equations by transverse integrating the partial differential equations in each cell. The formulation leads to a quadratic local profile for the cell interior pressure in each spatial direction, and a *constant + linear + exponential* variation for each of the transverse-integrated velocities in each direction. Despite the fact that there are more unknowns per cell than in typical finite difference or finite volume schemes, the nodal scheme is very efficient. A desired level of accuracy is achieved by the nodal scheme with much coarser mesh size, hence making up for the larger number of discrete unknowns per cell.

The numerical scheme developed here is semi-implicit. Introduction of the convection term, with the velocity evaluated at the *previous* time step, leads to a formulation in which the exponential terms are evaluated only at the previous time step. Hence, these exponential functions are to be evaluated only once for each time step instead of repeated evaluation at each iteration. The scheme has inherent upwinding, automatically adjusting weights of its coefficients depending upon the flow direction. Applications to two steady-state and one time-dependent problems show very good agreement with the exact solutions or with the fine mesh solution, even for relatively coarse meshes. Numerical results also show the near second-order accuracy of the scheme, consistent with previously reported results for nodal schemes based on approximations similar to those introduced here. Applications to the steady-state problems reported here show that the scheme converges to the correct solution from a very wide range of initial guesses.

It is clear that the analytical pre-processing of the set of PDEs leads to an efficient numerical scheme. However, this efficiency is not without its limitations. Most severe limitation of the nodal schemes based on transverse-integrating procedure is that they are limited to domains with boundaries that are parallel to the x or y axes. However, hybrid schemes applicable to complex domains recently developed for the convection–diffusion equation – that couple nodal schemes (applied to brick-like cells in the interior of the physical domain as well as adjacent to horizontal or vertical boundaries) with finite analytic or finite element method (for regions adjacent to boundaries that are not parallel to the Cartesian axes) – have been very promising [38]. Their extension to the Navier–Stokes equations will relax this limitation of the nodal approach.

Acknowledgements

Research funded in part by the U.S. Department of Energy through the University of California under subcontract number B341494. First author (F.W.) would also like to acknowledge support under the Computational Science and Engineering Fellowship program at UIUC.

Appendix A

Exact solution of the modified lid driven cavity problem ($0 \leqslant x \leqslant 1$ and $0 \leqslant y \leqslant 1$) is given by [33]

$$u(x,y) = 8(x^2 - 2x^3 + x^4)(-2y + 4y^3),$$

$$v(x,y) = -8(2x - 6x^2 + 4x^3)(-y^2 + y^4)$$

and

$$p(x,y,Re) = \frac{8}{Re}\left(24\left(\frac{x^3}{3} - \frac{x^4}{2} + \frac{x^5}{5}\right)y + (2x - 6x^2 + 4x^3)(-2y + 4y^3)\right) + 64\left(\frac{x^4}{2} - 2x^5 + 3x^6 - 2x^7 + \frac{x^8}{2}\right)\left(-(-2y + 4y^3)^2 + (-2 + 12y^2)(-y^2 + y^4)\right),$$

where the non-uniformly distributed body forces are given by

$$b_x(x,y,Re) = -\frac{8}{Re}\left(24\left(\frac{x^3}{3} - \frac{x^4}{2} + \frac{x^5}{5}\right) + 2(2x - 6x^2 + 4x^3)(-2 + 12y^2) + (-12 + 24x)(-y^2 + y^4)\right) + 64\Big(-(-2x^2 + 8x^3 - 14x^4 + 12x^5 - 4x^6)(-2y + 4y^3)(-y^2 + y^4) + \left(\frac{x^4}{2} - 2x^5 + 3x^6 - 2x^7 + \frac{x^8}{2}\right)\left(-(-2 + 12y^2)(-2y + 4y^3) + 24y(-y^2 + y^4)\right)\Big),$$

$$b_y = 0.$$

The lid velocity is given by

$$u_{\text{lid}}(x) = u(x, y = 1) = 16(x^2 - 2x^3 + x^4).$$

References

[1] T.J. Burns, J.J. Dorning, A new computational method for the solution of multidimensional neutron problems, in: Proceedings of the Joint NEACRP/CSNI Specialists' Meeting on New Developments in Three-Dimensional Neutron Kinetics and Review of Kinetics Benchmark Calculations, Laboratorium fur Reaktorregelung and Anlagensicherung, Garching, Munich, Germany, 1975, pp. 109–130.

[2] T.J. Burns, The partial current balance method: a local Green's function technique for the numerical solution of multidimensional diffusion problems, Ph.D. Thesis, University of Illinois, 1975.

[3] R.D. Lawrence, A nodal Green's function method for multidimensional diffusion calculations, Ph.D. Thesis, University of Illinois, 1979.

[4] R.D. Lawrence, J.J. Dorning, A nodal Green's function method for multidimensional neutron diffusion calculations, Nucl. Sci. Engrg. 76 (1980) 218–231.

[5] R.D. Lawrence, J.J. Dorning, A discrete nodal integral transport theory method for multidimensional reactor physics and shielding, in: Proceedings of the ANS Conference on Advances in Reactor Physics and Shielding, Sun Valley, Idaho, 1980.

[6] W.C. Horak, Local Green's function techniques for the solution of heat conduction and incompressible fluid flow problems, Ph.D. Thesis, University of Illinois, Urbana, 1980.

[7] W.C. Horak, J.J. Dorning, A nodal coarse-mesh method for the efficient numerical solution of laminar flow problems, J. Comp. Phys. 59 (1985) 405–440.

[8] H.D. Fischer, H. Finnemann, The nodal integration method – a diverse solver for neutron diffusion problems, Atomkernenergie Kerntechnik 39 (1981) 229–236.

[9] Y.Y. Azmy, J.J. Dorning, A nodal integral approach to the numerical solution of partial differential equations, in: Advances in Reactor Computations, vol. II, American Nuclear Society, LaGrange Park, IL, 1983, pp. 893–909.
[10] Y.Y. Azmy, Nodal method for problems in fluid mechanics and neutron transport, Ph.D. thesis, University of Illinois, 1985.
[11] R.D. Lawrence, Progress in nodal methods for the solutions of the neutron diffusion and transport equations, Prog. Nucl. Energy 17 (3) (1986) 271–301.
[12] G.L. Wilson, R.A. Rydin, Y.Y. Azmy, Time-dependent nodal integral method for the investigation of bifurcation and nonlinear phenomena in fluid flow and natural convection, Nucl. Sci. Engrg. 100 (1988) 414–425.
[13] O.A. Elnawawy, A.J. Valocchi, A.M. Ougouag, The cell analytical–numerical method for solution of the advection–dispersion equation: two-dimensional problems, Water Resour. Res. 26 (11) (1990) 2705–2716.
[14] Rizwan-uddin, An improved coarse-mesh nodal integral method for partial differential equations, Numer. Methods Partial Differential Equations 13 (1997) 113–145.
[15] B. Wescott, Rizwan-uddin, An efficient formulation of the modified nodal integral method and application to the two-dimensional Burgers' equation, Nucl. Sci. Engrg. 139 (2001) 293–305.
[16] P.D. Esser, R.J. Witt, An upwind nodal integral method for incompressible fluid flow, Nucl. Sci. Engrg. 114 (1993) 20–35.
[17] E.P.E. Michael, J.J. Dorning, Rizwan-uddin, Studies on nodal integral methods for the convection–diffusion heat equation, Nucl. Sci. Engrg. 137 (2001) 380–399.
[18] J.P. Hennart, On the numerical analysis of analytical nodal methods, Numer. Methods Partial Differential Equations 4 (1988) 233–254.
[19] J.P. Hennart, A general family of nodal schemes, SIAM J. Sci. Stat. Comp. 7 (1) (1986) 264–287.
[20] Rizwan-uddin, Comparison of the nodal integral method and non-standard finite-difference schemes for the Fisher equation, SIAM J. Sci. Comput. 22 (6) (2001) 1926–1942.
[21] C. Gunther, Conservative versions of the locally exact consistent upwind scheme of second order (LECUSSO-SCHEME), Int. J. Numer. Methods Engrg. 34 (1992) 793.
[22] J.D. Anderson Jr., Computational Fluid Dynamics: The Basics with Applications, McGraw-Hill, New York, 1995.
[23] Fei Wang, Rizwan-uddin, A nodal scheme for the time-dependent incompressible Navier–Stokes equations, Trans. Am. Nucl. Soc. 83 (2000) 422–424.
[24] E.P.E. Michael, J.J. Dorning, A primitive-variable nodal method for steady-state fluid flow, Trans. Am. Nucl. Soc. 83 (2000) 420–422.
[25] E.P.E. Michael, J.J. Dorning, A primitive-variable nodal method for the time-dependent Navier–Stokes equations, in: Proceedings of the ANS Topical Meeting on Math. and Comp. 2001, Salt Lake City, 2001.
[26] J.C. Tannehill, D.A. Anderson, R.H. Pletcher, Computational Fluid Mechanics and Heat Transfer, second ed., Taylor & Francis, 1997.
[27] K.N. Ghia, W.L. Hankey Jr., J.K. Hodge, Study of incompressible Navier–Stokes equations in primitive variables using implicit numerical technique, AIAA Paper 77-648, 1977.
[28] J.H. Ferziger, M. Peric, Computational Methods for Fluid Dynamics, Springer, Berlin, 1996.
[29] P.M. Gresho, R.L. Sani, On pressure boundary conditions for the incompressible Navier–Stokes equation, Int. J. Numer. Methods Fluids 7 (1987) 1111–1145.
[30] F. Pan, A. Acrivos, Steady flows in rectangular cavities, J. Fluid Mech. 28 (1967) 643–655.
[31] U. Ghia, K.N. Ghia, C.T. Shin, High-Re solutions for incompressible flow using the Navier–Stokes equatons and a multigrid method, J. Comput. Phys. 48 (1982) 387–411.
[32] P.J. O'Rourke, M.S. Sahota, A variable explicit/implicit numerical method for calculating advection on unstructured meshes, J. Comput. Phys. 143 (1998) 312–345.
[33] T.M. Shin, C.H. Tan, B.C. Hwang, Effects of grid staggering on numerical schemes, Int. J. Numer. Methods Fluids 9 (1989) 193–212.
[34] G.I. Taylor, On the decay of vortices in a viscous fluid, Philos. Mag. 46 (1923) 671–675.
[35] J. Kim, P. Moin, Application of a fractional-step method to incompressible Navier–Stokes equations, J. Comput. Phys. 59 (1985) 308.
[36] M.O. Henriksen, J. Holmen, Algebraic splitting for incompressible Navier–Stokes equations, J. Comput. Phys. 175 (2002) 438–453.
[37] A. Quarteroni, F. Saleri, A. Veneziani, Factorization methods for the numerical approximation of Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 188 (2000) 505–526.
[38] A.J. Toreja, Rizwan-uddin, Hybrid numerical methods for convection–diffusion problems in arbitrary geometries, Comput. Fluids 32 (6) (2003) 835–872.