INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2006; **52**:867–881
Published online 13 March 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1207

A compact finite difference method on staggered grid for Navier–Stokes flows

K. K. Q. Zhang[‡], B. Shotorban[§], W. J. Minkowycz[¶] and F. Mashayek[*,†]

Department of Mechanical and Industrial Engineering, University of Illinois at Chicago, 842 West Taylor Street, Chicago, IL 60607, U.S.A.

SUMMARY

Compact finite difference methods feature high-order accuracy with smaller stencils and easier application of boundary conditions, and have been employed as an alternative to spectral methods in direct numerical simulation and large eddy simulation of turbulence. The underpinning idea of the method is to cancel lower-order errors by treating spatial Taylor expansions implicitly. Recently, some attention has been paid to conservative compact finite volume methods on staggered grid, but there is a concern about the order of accuracy after replacing cell surface integrals by average values calculated at centres of cell surfaces. Here we introduce a high-order compact finite difference method on staggered grid, without taking integration by parts. The method is implemented and assessed for an incompressible shear-driven cavity flow at $Re = 10^3$, a temporally periodic flow at $Re = 10^4$, and a spatially periodic flow at $Re = 10^4$. The results demonstrate the success of the method. Copyright © 2006 John Wiley & Sons, Ltd.

KEY WORDS: compact finite difference; staggered grid; collocated grid; incompressible Navier–Stokes flow

1. INTRODUCTION

Finite difference methods, invented far earlier than finite element methods, boundary element methods, spectral methods, and discontinuous spectral element methods [1], remain as a competitive class of direct domain discretization methods for a wide range of applications. With global mappings, finite difference methods can be used to tackle problems with complex geometry, such as free-surface flows [2, 3]. Generally, the approach of local geometric

*Correspondence to: F. Mashayek, Department of Mechanical and Industrial Engineering, University of Illinois at Chicago, 842 West Taylor Street, Chicago, IL 60607, U.S.A.
†E-mail: mashayek@uic.edu
‡E-mail: kenn_kq_zhang@yahoo.com
§E-mail: shotorba@uiuc.edu
¶E-mail: wjm@uic.edu

Contract/grant sponsor: National Science Foundation; contract/grant number: CTS-0237951

Received 22 September 2004
Revised 10 November 2005
Accepted 13 November 2005

mappings in finite element and spectral element methods perform more flexibly and efficiently in problems with complex geometry, especially in 3-D free-surface flows and fluid–structure interactions. However, for a problem with simple geometry a finite difference method performs appreciably faster than a finite element method, because the latter ignores to utilize the simplicity of the geometry. Since flows with simple geometry not only exist in nature and industry but also serve as the first step to test mathematical models or to understand underlying physics, research on design and analysis of finite difference algorithms remains active. There are at least two major topics, conservative schemes (such as Godunov-based schemes) and high-order schemes. In this paper, we focus on high-order compact finite difference methods.

The idea of compact finite difference is rather simple. For example, in the classical explicit-in-space finite difference methods, a 3-point interpolation for a second-order derivative produces an approximation with a second-order truncation error. But if the second-order derivative is interpolated *implicitly* in space, a higher-order approximation can be reached with involvement of the same three points. Here we illustrate through an example how this can be achieved. Figure 1 shows a distribution of three discrete unknowns and their second derivatives to be interpolated. Expanding all quantities at the same selected point, which in this example is selected at the position of φ_i and φ_i'', we have

$$\varphi_{i-1}'' = \varphi_i'' + \varphi_i'''(-h) + \varphi_i^{(4)}\frac{(-h)^2}{2} + \varphi_i^{(5)}\frac{(-h)^3}{6} + \mathcal{O}(h^4) \tag{1}$$

$$\varphi_{i+1}'' = \varphi_i'' + \varphi_i'''h + \varphi_i^{(4)}\frac{h^2}{2} + \varphi_i^{(5)}\frac{h^3}{6} + \mathcal{O}(h^4) \tag{2}$$

$$\varphi_{i-1} = \varphi_i + \varphi_i'(-h) + \varphi_i''\frac{(-h)^2}{2} + \varphi_i'''\frac{(-h)^3}{6} + \varphi_i^{(4)}\frac{(-h)^4}{24} + \varphi_i^{(5)}\frac{(-h)^5}{120} + \mathcal{O}(h^6) \tag{3}$$

$$\varphi_{i+1} = \varphi_i + \varphi_i'h + \varphi_i''\frac{h^2}{2} + \varphi_i'''\frac{h^3}{6} + \varphi_i^{(4)}\frac{h^4}{24} + \varphi_i^{(5)}\frac{h^5}{120} + \mathcal{O}(h^6) \tag{4}$$

Cancelling φ_i', φ_i''', $\varphi_i^{(4)}$, and $\varphi_i^{(5)}$ from the above four equations creates the ith interpolation equation

$$\frac{1}{10}\ \varphi_{i-1}'' + \varphi_i'' + \frac{1}{10}\varphi_{i+1}'' = \frac{6}{5}\frac{\varphi_{i-1} - 2\varphi_i + \varphi_{i+1}}{h^2} + \mathcal{O}(h^4) \tag{5}$$

With non-periodic boundary conditions, a closed tridiagonal system can be formed after similar treatments at other nodes on the same line and special treatments near boundaries. By solving this small set of equations one can *implicitly* interpolate quantities such as second derivatives with discrete unknowns.

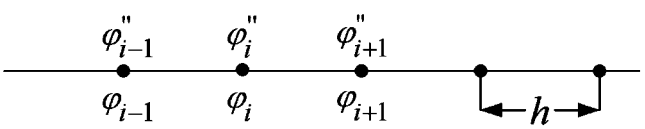

Figure 1. Collocated grid used in fourth-order interpolation for second-order derivatives. In all similar figures, quantities to be interpolated are placed above the line and quantities used to interpolate are placed below the line.

The idea of compact scheme was first introduced by Kreiss [4, 5] and implemented by Hirsh [5], then popularized by Lele [6]. Efforts in promotion of compact finite difference schemes on collocated grids are also witnessed in References [7–16] (in chronological order). Compact finite difference methods feature high-order accuracy with smaller stencils and easier application of boundary conditions, and have been employed as an alternative to spectral methods in direct numerical simulation and large eddy simulation of turbulence [6, 11]. More comments on pros and cons of compact schemes are postponed to Section 4.

The robustness of staggered grid for second-order finite difference methods prompts a similar treatment for compact finite difference methods. Nagarajan *et al.* [17] investigated compact schemes on staggered grid for compressible flows and applied the method to large eddy simulation of turbulence. Piller and Stalio [18] developed compact schemes for incompressible flows. Both papers reported preference of staggered grids over collocated grids. However, there is a concern about the order of accuracy of these two methods on staggered grids. Both methods take a locally conservative approach. First, through integration by parts several terms of the governing equations in a cell are transferred to cell surface integrals. This step introduces no approximation errors. Not only because this is a pure mathematical operation, but also because the differential form of the governing equations is actually a consequence of taking integration by parts on the more original integral form of the conservation equations. Next, cell surface integrals are replaced by average values sitting at centres of surfaces. This step, as we know, introduces second-order approximation errors throughout the computational domain. Finally, all quantities which are not defined as discrete unknowns, such as first derivatives, are interpolated by discrete unknowns with high-order accuracy. Nevertheless, due to the second step the high-order accuracy of whole method becomes questionable.

This paper is not intended to address how to reduce approximation errors of cell surface integrals in cell/element-based conservative methods. Instead, in the following three sections we introduce a point/node-based non-locally-conservative high-order compact finite difference method on the same staggered grid. It is worthwhile to mention that our treatment of boundary conditions will differ significantly from those in References [17, 18].

2. INTERPOLATIONS

In this section all interpolations are listed for later reference, together with treatment of boundaries. Equation (5), which will also be used, is not repeated here. The factors in choosing specific arrangement of points include order of truncation errors, more direct involvement of boundary conditions, and formation of tridiagonal system, which can be solved efficiently by a direct Gauss elimination (Thomas algorithm). We detail the first case of interpolation of zero-order derivatives on staggered grid, where both the ideas and notations are introduced. Although the compact finite difference method to be introduced in this paper is equally applicable to compressible flows, for simplicity we illustrate the idea via incompressible flows.

2.1. Interpolations on staggered grid

2.1.1. Zero-order derivatives. Figure 2 shows the staggered grid for interpolation of zero-order derivatives (function value itself). Quantities below the line, ψ_i's in this case, are discrete unknowns while quantities above the line, φ_i's in this case, are quantities to be interpolated.

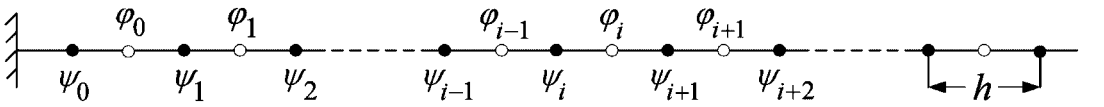

Figure 2. Staggered grid used in sixth-order interpolation for zero-order derivatives.

Instead of non-integers such as $i+\frac{1}{2}$ used in References [17, 18], we use integer indices since they are being used in actual computer programming. In this case of interpolation, boundary conditions are not required but the first interpolation equation, characterized by φ_0 and indexed by $i=0$, differs from interior interpolation equations. Following the same idea illustrated in Section 1, selecting a point and expanding all quantities at the same point we have

$$\varphi_0 = \psi_1 + \psi_1'\left(-\frac{h}{2}\right) + \psi_1''\frac{(-\frac{h}{2})^2}{2} + \psi_1'''\frac{(-\frac{h}{2})^3}{6} + \mathcal{O}(h^4) \tag{6}$$

$$\varphi_1 = \psi_1 + \psi_1'\frac{h}{2} + \psi_1''\frac{(\frac{h}{2})^2}{2} + \psi_1'''\frac{(\frac{h}{2})^3}{6} + \mathcal{O}(h^4) \tag{7}$$

$$\psi_0 = \psi_1 + \psi_1'(-h) + \psi_1''\frac{(-h)^2}{2} + \psi_1'''\frac{(-h)^3}{6} + \mathcal{O}(h^4) \tag{8}$$

$$\psi_1 = \psi_1 + \psi_1' h + \psi_1''\frac{h^2}{2} + \psi_1'''\frac{h^3}{6} + \mathcal{O}(h^4) \tag{9}$$

Cancelling three unwanted quantities ψ_1', ψ_1'', and ψ_1''' from the above four equations produces

$$\varphi_0 + \varphi_1 = \frac{3}{2}\psi_1 + \frac{1}{2}\frac{\psi_0 + \psi_2}{2} + \mathcal{O}(h^4) \tag{10}$$

Similarly, by expanding φ_{i-1}, φ_{i+1}, ψ_{i-1}, ψ_i, ψ_{i+1}, and ψ_{i+2} at location i, one obtains six equations and five undesirable quantities of φ_i', φ_i'', φ_i''', $\varphi_i^{(4)}$, and $\varphi_i^{(5)}$. An elimination of these five quantities produces

$$\frac{3}{10}\varphi_{i-1} + \varphi_i + \frac{3}{10}\varphi_{i+1} = \frac{3}{2}\frac{\psi_i + \psi_{i+1}}{2} + \frac{1}{10}\frac{\psi_{i-1} + \psi_{i+2}}{2} + \mathcal{O}(h^6) \tag{11}$$

The boundary treatment on the right end always can be done by simply replacing h by $-h$, provided care is taken in the bookkeeping of indices.

2.1.2. First-order derivatives. Two cases of interpolation of first-order derivatives on staggered grid will be used and distributions of points are illustrated in Figures 3 and 4. The first case (Figure 3) needs no boundary conditions but the first interpolation equation differs from those for the interior

$$\varphi_0' - \varphi_1' = \frac{\psi_1 - \psi_0}{h} - \frac{\psi_2 - \psi_1}{h} + \mathcal{O}(h^3) \tag{12}$$

For the interior

$$\frac{9}{62}\varphi_{i-1}' + \varphi_i' + \frac{9}{62}\varphi_{i+1}' = \frac{63}{62}\frac{\psi_{i+1} - \psi_i}{h} + \frac{17}{62}\frac{\psi_{i+2} - \psi_{i-1}}{3h} + \mathcal{O}(h^6) \tag{13}$$

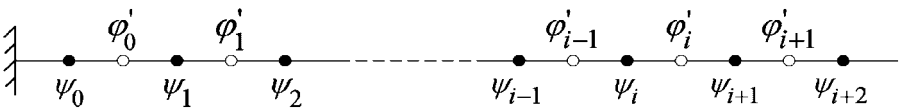

Figure 3. The first type of staggered grid used in sixth-order interpolation for first-order derivatives.

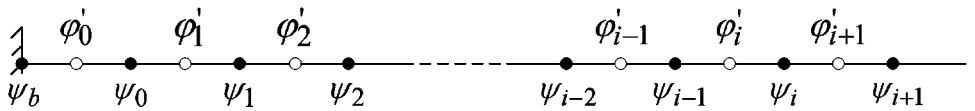

Figure 4. The second type of staggered grid used in sixth-order interpolation for first-order derivatives.

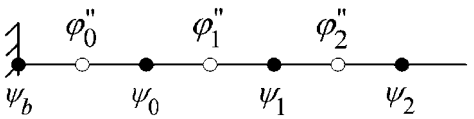

Figure 5. Staggered grid used in interpolation for second-order derivatives near boundaries.

The second case (Figure 4), which needs boundary conditions, differs from the first case mainly in indexing. The first interpolation equation becomes

$$\varphi'_0 - \varphi'_1 = \frac{\psi_0 - \psi_b}{h} - \frac{\psi_1 - \psi_0}{h} + \mathcal{O}(h^3) \tag{14}$$

where the subscript b indicates the boundary value. For the interior, except for $i = 1$, we have

$$\frac{9}{62}\varphi'_{i-1} + \varphi'_i + \frac{9}{62}\varphi'_{i+1} = \frac{63}{62}\frac{\psi_i - \psi_{i-1}}{h} + \frac{17}{62}\frac{\psi_{i+1} - \psi_{i-2}}{3h} + \mathcal{O}(h^6) \tag{15}$$

For the case of $i = 1$, simply replace the ψ_{i-2} in Equation (15) by ψ_b.

2.1.3. Second-order derivatives. In the point/node-based compact finite difference method presented in this paper, for the interior region second-order derivatives are interpolated on collocated grid only. However, in the vicinity of boundaries the interpolations of second-order derivatives both on collocated grid and on staggered grid will be useful. Interpolations (Figure 5) on staggered grid are

$$\varphi''_0 + \varphi''_1 = 2\frac{\psi_b - 2\psi_0 + \psi_1}{h^2} + \mathcal{O}(h^2) \tag{16}$$

$$\frac{5}{14}\varphi''_0 + \varphi''_1 + \frac{5}{14}\varphi''_2 = \frac{6}{7}\frac{\psi_b - 2\psi_0 + \psi_1}{h^2} + \frac{6}{7}\frac{\psi_0 - 2\psi_1 + \psi_2}{h^2} + \mathcal{O}(h^4) \tag{17}$$

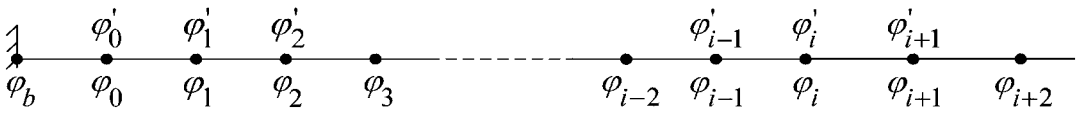

Figure 6. Collocated grid used in sixth-order interpolation for first-order derivatives.

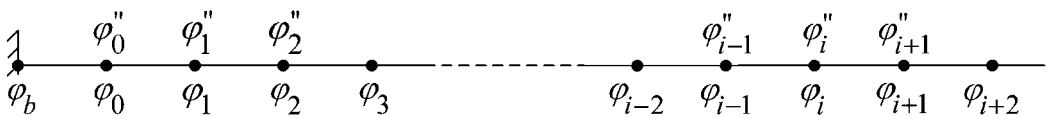

Figure 7. Collocated grid used in sixth-order interpolation for second-order derivatives.

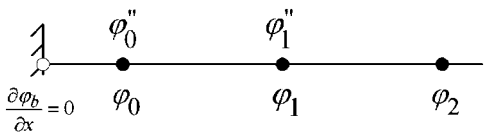

Figure 8. Collocated grid used in interpolation for second-order derivatives near boundaries.

2.2. *Interpolations on collocated grid*

2.2.1. First-order derivatives. Figure 6 shows the distribution of collocated grid points for interpolation of first-order derivatives. For $i = 0$, the interpolation equation is

$$\varphi'_0 + \varphi'_1 = \frac{3}{2}\frac{\varphi_1 - \varphi_0}{h} + \frac{1}{2}\frac{\varphi_2 - \varphi_b}{3h} + \mathcal{O}(h^4) \tag{18}$$

For interior region, the interpolation equation is

$$\frac{1}{3}\varphi'_{i-1} + \varphi'_i + \frac{1}{3}\varphi'_{i+1} = \frac{14}{9}\frac{\varphi_{i+1} - \varphi_{i-1}}{2h} + \frac{1}{9}\frac{\varphi_{i+2} - \varphi_{i-2}}{4h} + \mathcal{O}(h^6) \tag{19}$$

2.2.2. Second-order derivatives. Figure 7 shows the collocated grid for interpolation of second-order derivatives, and the corresponding interpolation equations are

$$\varphi''_0 - \varphi''_1 = \frac{\varphi_b - 2\varphi_0 + \varphi_1}{h^2} - \frac{\varphi_0 - 2\varphi_1 + \varphi_2}{h^2} + \mathcal{O}(h^3) \tag{20}$$

$$\frac{2}{11}\varphi''_{i-1} + \varphi''_i + \frac{2}{11}\varphi''_{i+1} = \frac{12}{11}\frac{\varphi_{i-1} - 2\varphi_i + \varphi_{i+1}}{h^2} + \frac{3}{11}\frac{\varphi_{i-2} - 2\varphi_i + \varphi_{i+2}}{(2h)^2} + \mathcal{O}(h^6) \tag{21}$$

2.2.3. Second-order derivatives for the pressure. A pressure Poisson equation, to be explained in the next section, will be solved in a pseudo time marching manner for the incompressible Navier–Stokes flows. Figure 8 shows the collocated grid for interpolation of second-order derivative for the pressure near boundaries only. By expanding φ''_0, φ''_1, φ_0, φ_1, and φ_2 at the

boundary and utilizing $\partial\varphi/\partial x = 0$, one obtains

$$\varphi_0'' - \frac{11}{23}\varphi_1'' = \frac{36}{23h^2}(\varphi_1 - \varphi_0) - \frac{12}{23h^2}(\varphi_2 - \varphi_1) + \mathcal{O}(h^3) \tag{22}$$

3. FORMULATION AND IMPLEMENTATION

3.1. Formulation

The non-dimensional governing equations for incompressible Navier–Stokes flows in conservative form read

$$\partial_t u_i + \partial_j(u_j u_i) = -\partial_i p + \frac{1}{Re}\partial_j\partial_j u_i + f_i \tag{23}$$

$$\partial_j u_j = 0 \tag{24}$$

where Re is the Reynolds number, ∂_t denotes partial derivative with respect to time, and ∂_i and ∂_j denote partial derivatives with respect to space. Instead of the recently developed exact factorization technique [19], a classical splitting method is chosen to tackle Equations (23) and (24). This is because the nature of point/node-based compact finite difference makes matrices less convenient. For the same reason, explicit time schemes are preferred. Otherwise, as adopted in Reference [18], one has to solve a much larger linear system compared with classical explicit-in-space schemes. Most flows are rather convection-dominated. In such flows often it is not the diffusion term but the nonlinear convection term that imposes time step restriction. In this sense, popular semi-implicit time schemes do not offer a significant advantage over explicit schemes. Also, semi-implicit schemes are indeed more robust but at the same time more deceptive. A large time step may not incur a blowup of the solution in boundary layers, but the time accuracy of the solution may be lost. With a second-order Adams–Bashforth scheme, the semi-discrete form of Equations (23) and (24) becomes

$$u_i^* = u_i^n + \Delta t\left\{-\frac{3}{2}\partial_j(u_j u_i)^n + \frac{1}{2}\partial_j(u_j u_i)^{n-1} + \frac{3}{2Re}\partial_j\partial_j u_i^n - \frac{1}{2Re}\partial_j\partial_j u_i^{n-1} + \frac{3}{2}f_i^n - \frac{1}{2}f_i^{n-1}\right\} \tag{25}$$

$$p^{m+1} = p^m + \Delta\tau\left(\partial_j\partial_j p^m - \frac{1}{\Delta t}\partial_j u_j^*\right) \tag{26}$$

$$u_i^{n+1} = u_i^* - \Delta t\partial_i p \tag{27}$$

where the pressure Poisson equation (26) is solved by a pseudo-time marching procedure. The superscript m in Equation (26) denotes the pseudo time level and the pseudo time step size $\Delta\tau$ can be chosen to guarantee the convergence of pseudo time marching and to retain a reasonable efficiency.

3.2. Implementation

Although the major advantage of staggered grid resides in its ease for imposing local conservation, one can use this grid in point/node-based non-locally-conservative methods.

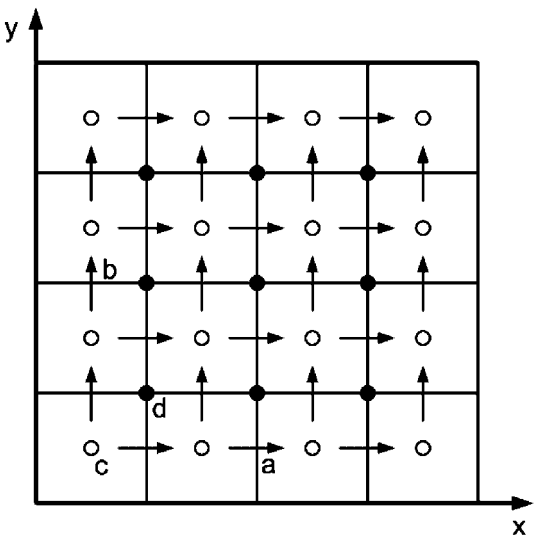

Figure 9. Staggered grid for compact finite difference in shear-driven cavity flow and periodic cavity flow: u, u^*, $\partial(uu)/\partial x$, $\partial(vu)/\partial y$, $\partial^2 u/\partial x^2$, $\partial^2 u/\partial y^2$, and $\partial p/\partial x$ are defined at a-type points; v, v^*, $\partial(uv)/\partial x$, $\partial(vv)/\partial y$, $\partial^2 v/\partial x^2$, $\partial^2 v/\partial y^2$, and $\partial p/\partial y$ are defined at b-type points; p, $\partial^2 p/\partial x^2$, $\partial^2 p/\partial y^2$, and $\partial u^*/\partial x$, $\partial v^*/\partial y$ are defined at c-type points; d-type points are for temporary quantities, u^s and v^s.

For simplicity, the 2-D shear-driven cavity flow at $Re = 10^3$ is selected to illustrate the compact scheme. Figure 9 shows the staggered grid and locations where various quantities are defined. Suppose p, u^n, v^n, u^{n-1}, and v^{n-1} are known, a time marching process is comprised of the following steps:

1. Update u^* and v^* according to Equation (25).
2. Interpolate $\partial u^*/\partial x$ and $\partial v^*/\partial y$ using Equations (14) and (15). According to Reference [19], on solid boundaries $u^* = 0$ and $v^* = 0$.
3. Solve p using the pseudo time marching, Equation (26). $\partial^2 p/\partial x^2$ and $\partial^2 p/\partial y^2$ are interpolated according to Equations (5), (21), and (22). According to Reference [19], on solid boundaries $\partial p/\partial n = 0$.
4. Interpolate $\partial p/\partial x$ and $\partial p/\partial y$ according to Equations (12) and (13).
5. Update u^{n+1} and v^{n+1} according to Equation (27).
6. Bookkeep all time dependent data. In particular, update u^n and v^n.
7. Interpolate temporary quantities u^s and v^s (illustrated in Figure 9) from u^n and v^n, according to Equations (10) and (11).
8. Interpolate $\partial(vu)^n/\partial y$ and $\partial(uv)^n/\partial x$ from $v^s u^s$, according to Equations (14) and (15).
9. Interpolate $\partial(uu)^n/\partial x$ and $\partial(vv)^n/\partial y$ according to Equations (18) and (19).
10. Interpolate $\partial^2 u^n/\partial x^2$ and $\partial^2 v^n/\partial y^2$ according to Equations (20) and (21).
11. Near boundaries, use Equations (16) and (17) to interpolate $\partial^2 u^n/\partial y^2$ and $\partial^2 v^n/\partial x^2$; in the interior region use Equation (21). For the cavity flow, the only non-vanishing boundary condition comes into play through the term $\partial^2 u^n/\partial y^2$.
12. Go back to step 1.

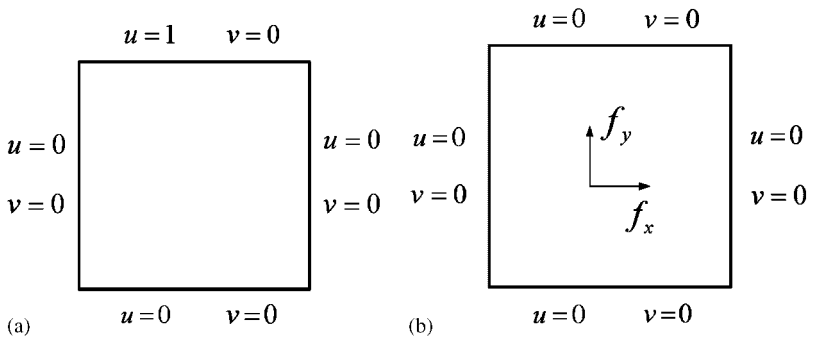

Figure 10. (a) Configuration and boundary conditions for shear-driven cavity; and (b) configuration and boundary conditions for temporally periodic flow.

4. NUMERICAL RESULTS

In this section we use three numerical examples to demonstrate the success of the method. The first example is the classical benchmark problem of shear-driven cavity flow, where one wall is moving. In the second example all walls are stationary, and the flow is driven by a temporally periodic body force; the exact solution can be found for this example. In the third example, walls are replaced by spatially periodic boundary conditions and the flow is driven by a developing body force. These three examples are also useful to evaluate the effect of the walls on the overall order of accuracy based on truncation errors. This is important since the order of accuracy of discretization used near the walls is lower than that used for the interior nodes.

4.1. Shear-driven cavity

First, the classical temporally developing shear-driven cavity flow (Figure 10(a)) at $Re = 10^3$ is calculated with a time step $\Delta t = 10^{-4}$ to ensure the time accuracy. Figure 11(a) shows velocity profiles on three different grids, where the coarse grid of 32×32 gives rather rough results. However, as the mesh size decreases to half, the results quickly converge so that a further decrease of the mesh size makes little difference. This is the feature of high-order methods. In Figure 11(b), a comparison between velocity profile based on the current method and that based on multigrid second-order finite difference method with vorticity-streamfunction formulation [20] is presented, and an excellent agreement is observed. Figure 11(c) shows the pressure contour for the cavity flow at $Re = 10^3$. From the figure it can be seen that the top right corner has high pressure and high pressure gradient. In contrast, the low pressure area resides in the centre part of the flow.

The method presented in this paper has a formal sixth-order accuracy in the region away from boundaries, which is based on the truncation error. However, truncation errors become negligible only under the condition that the grid size is small enough, which is hard to realize in practice. The use of lower-order boundary condition approximations, such as those through Equations (12) and (14), also pollute the accuracy.

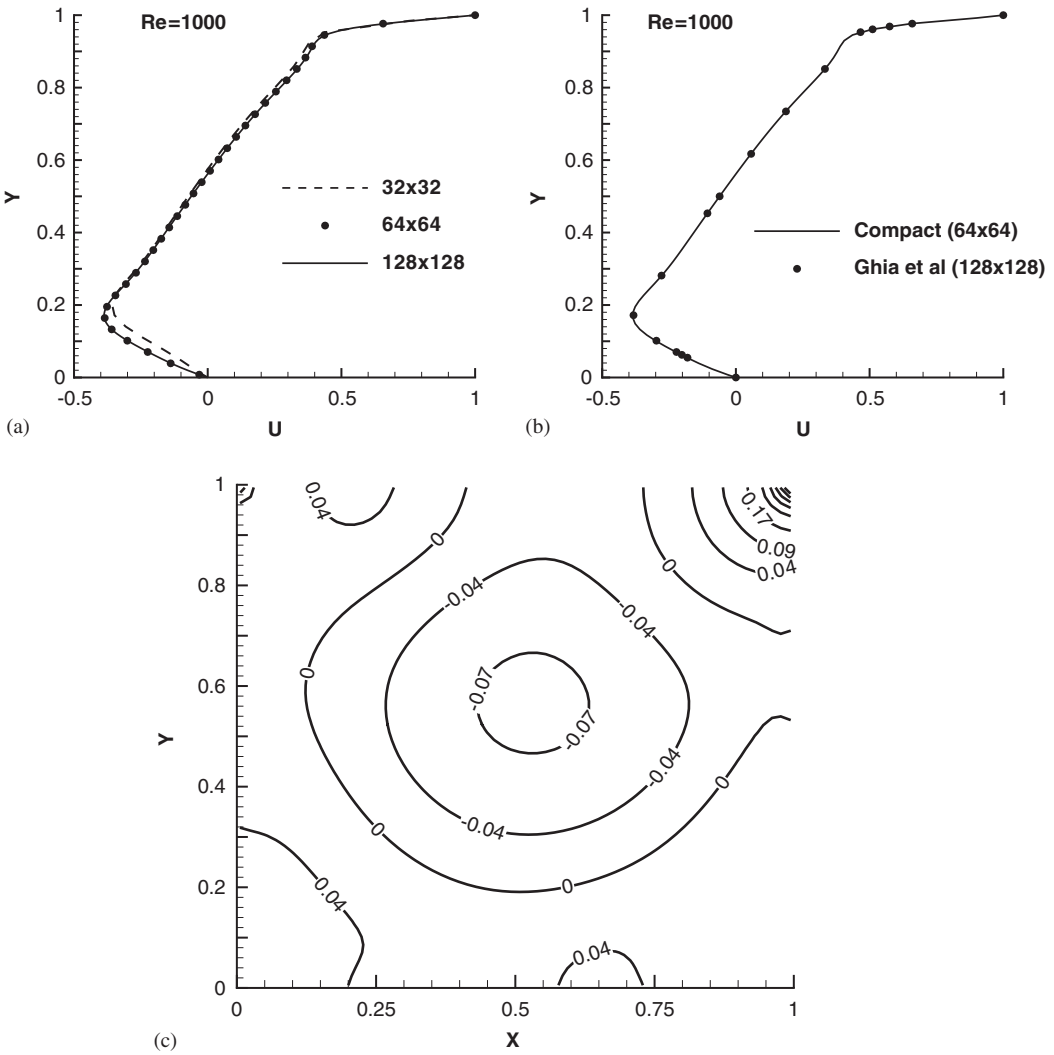

Figure 11. Results for the shear-driven cavity flow with the compact finite difference method: (a) horizontal velocity profiles (on vertical midplane) for grid independence study; (b) comparison of horizontal velocity profiles (on vertical midplane) between the compact finite difference method and Ghia *et al.* [20]; and (c) pressure contour at $Re = 10^3$.

Due to the staggered grid, the calculated horizontal velocity components on the vertical mid-plane never exactly stay at the same y positions for three grid sizes of 32×32, 64×64, and 128×128. To make a numerical comparison more convenient, raw data is post processed as follows. Using the interpolations (10) and (11) on staggered grid, raw data on staggered positions (a-type positions in Figure 9) are transferred to data on collocated positions (d-type positions in Figure 9) along the vertical mid-plane. Resulting data for the 32×32 grid is regarded as the base, and excessive data for the 64×64 and 128×128 grids are discarded.

Now we have three sets of post processed data on the same y positions, and each set contains 31 data points for different y at $x = 0.5$. Let the post processed data on the 32×32 grid be denoted by u_{32}, similarly for data on other two grids. The ratio of root mean square, $\sqrt{\frac{1}{31}\sum_{31}(u_{32} - u_{64})^2}/\sqrt{\frac{1}{31}\sum_{31}(u_{64} - u_{128})^2}$, is 14.7, suggesting an approximately fourth-order convergence rate.

4.2. *Temporally periodic flow*

Next, we consider a transient Navier–Stokes flow inside a unity square (Figure 10(b)) with all boundaries fixed, but under the influence of a supplied body force. The time-periodic exact solution is prescribed similar to [21]

$$u(x, y, t) = -\sin t \sin^2 \pi x \sin \pi y \cos \pi y$$
$$v(x, y, t) = \sin t \sin \pi x \cos \pi x \sin^2 \pi y$$
$$p(x, y, t) = \sin t \sin \pi x \cos \pi y$$

where both velocities and pressure vanish at $t = 0$ and velocities vanish on four boundaries $x = 0$, $x = 1$, $y = 0$, and $y = 1$. The appropriate body force functions can be derived by substituting exact solutions into the momentum equation (23)

$$f_x = -\cos t \sin^2 \pi x \sin \pi y \cos \pi y + \pi \sin^2 t \sin^3 \pi x \cos \pi x \sin^2 \pi y$$
$$+ \pi \sin t \cos \pi x \cos \pi y - \frac{\pi^2}{Re} \sin t (6 \sin^2 \pi x - 2 \cos^2 \pi x) \sin \pi y \cos \pi y$$
$$f_y = \cos t \sin \pi x \cos \pi x \sin^2 \pi y + \pi \sin^2 t \sin^2 \pi x \sin^3 \pi y \cos \pi y$$
$$- \pi \sin t \sin \pi x \sin \pi y + \frac{\pi^2}{Re} \sin t \sin \pi x \cos \pi x (6 \sin^2 \pi y - 2 \cos^2 \pi y)$$

The same grid (Figure 9) is used for this flow. Components of body forces stay at the same positions as velocity components. A comparison between the numerical solution and the exact solution at $Re = 10^4$ is presented in Figure 12, which shows an excellent agreement.

The order of accuracy of the numerical method is also studied for the temporally periodic flow. Table I shows root mean square (r.m.s) of the differences between numerical and exact solutions on two sets of meshes. The ratio of two r.m.s. values is 31.2, which displays a fifth-order accuracy.

In comparison to classical high-order difference schemes, compact schemes reach higher order of accuracy with smaller stencils, consequently boundary conditions are easier to impose. Compared with a second-order finite difference method, the compact finite difference method involves more operations during each time step, if the same grid is used. Our numerical experiments show that with a finer grid, a second-order conservative finite difference method (identical to finite volume method due to the simple geometry) displays comparable efficiency as the formally six-order compact finite difference method. However, to reach the same accuracy, compact schemes consume less memory than second-order explicit-in-space

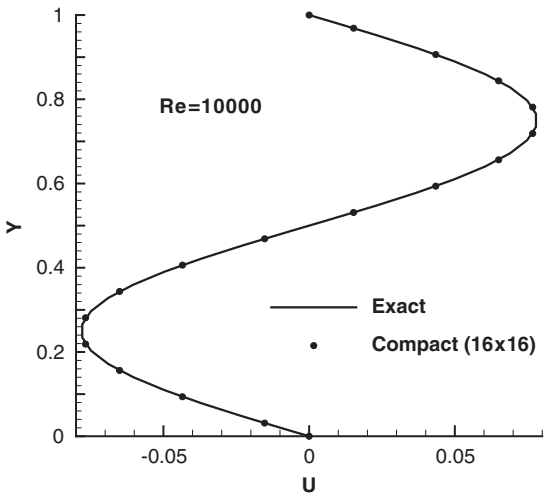

Figure 12. Comparison between the numerical solution and the exact solution for the temporally periodic flow.

Table I. Order of accuracy study for temporally periodic flow at $Re = 10^4$ and at non-dimensional time $t = 0.8$, where k represents number of grid points.

Grid	8×8	16×16
r.m.s. $\equiv \sqrt{\frac{1}{k}\sum_k (u_{\text{numerical}} - u_{\text{exact}})^2}$	0.0079152	0.0002536

difference schemes. More importantly, high-order compact schemes resolve short wavelengths better [6], and this is the reason that compact finite difference has been applied to direct numerical simulation of turbulence [11]. However, as mentioned in Reference [18], a filtering technique is required for higher *Re* flows.

4.3. Spatially periodic flow

Finally, we consider a spatially periodic flow, a benchmark problem used by Chorin [22]. Figure 13 shows the mesh for the flow. Periodic boundary conditions are applied on all four boundaries, which are simply bookkeeping of indices. Initially at rest, the flow is driven by a body force $\mathbf{f} = 0.1(1 - e^{-t})[\sin(2\pi y)\hat{i} + \sin(2\pi x)\hat{j}]$. Due to the spatial periodicity, it is more convenient to define additional unknown velocity components on the top boundary and on the right boundary. In contrast to the previous two numerical examples in this paper, *implicity in space* (as discussed in Section 1) does not result in a tridiagonal system. The resulting sparse system is solved by a Gauss–Seidel linear solver with compressed storage. Figure 14 shows the results at non-dimensional time $t = 4.0$. The vector field in Figure 14(a) demonstrates the same pattern as in Reference [22]. Figure 14(b) shows that the results based on coarse and fine grids are virtually identical. In the previous numerical examples, no fluids

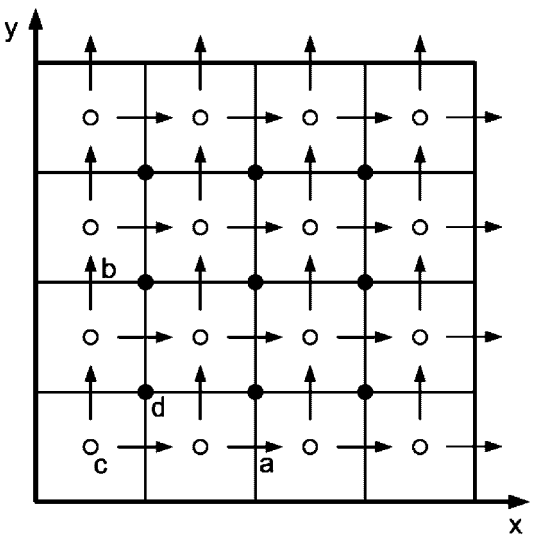

Figure 13. Mesh for the spatially periodic flow. In contrast to Figure 9, due to the spatial periodicity it is more convenient to define additional unknown velocity components on the right boundary and on the top boundary. The a, b, c, d-type points are used in the same way as in Figure 9.

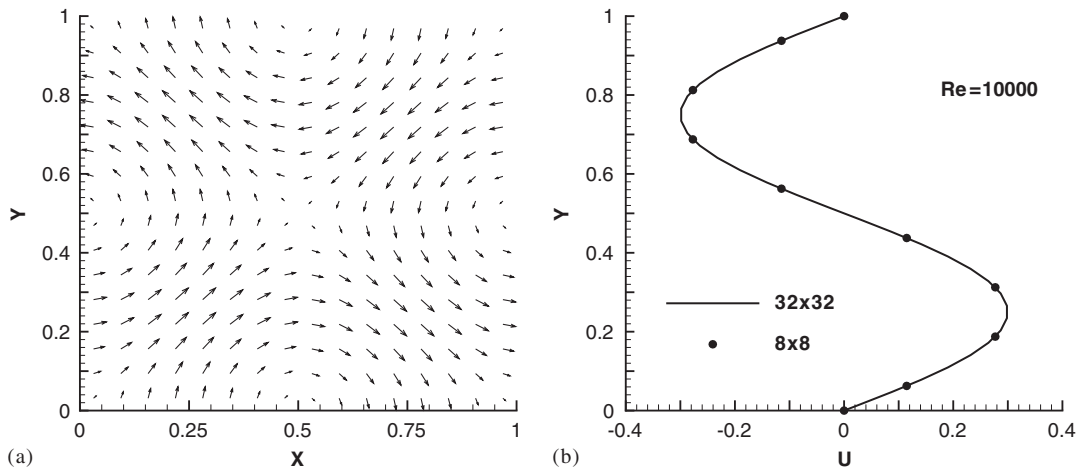

Figure 14. Results for the spatially periodic flow: (a) velocity vector field at $Re = 10^4$; and (b) horizontal velocity profile (on vertical midplane) on two different grids.

penetrate through boundaries and wall effects reduce interpolation accuracy. This spatially periodic flow is an ideal example to demonstrate the order of accuracy, which is checked at the point $(0.5, 0.25)$. In Table II, let u_1 represent the result based on the 8×8 grid and likewise u_2 and u_3 represent those based on two other grids, then $|u_1 - u_2|/|u_2 - u_3| = 67.4$, which indicates a sixth-order accuracy. In the shear-driven cavity flow, the moving boundary

Table II. Order of accuracy study for spatially periodic flow at $Re = 10^4$ and at $t = 4.0$.

Grid	8×8	16×16	32×32
u at $y = 0.25$	0.277020	0.298116	0.298429

is more influential on the core flow and the method shows a fourth-order accuracy. In the temporally periodic flow, the stationary wall is less effective in influencing the core flow and the method possesses a fifth-order accuracy. Finally in the spatially periodic flow, the lack of boundaries eliminates the need of lower-order interpolations such as Equation (18); as a result, the formal sixth-order accuracy is maintained.

5. CONCLUSIONS

In this paper we have presented the point/node-based compact finite difference method on staggered grid. The underpinning idea of a compact finite difference method is to cancel lower-order errors by treating spatial Taylor expansions implicitly. In the point/node-based compact finite difference method, all quantities appeared in a specific governing equation are defined on the same selected nodes and interpolated by unknowns on staggered or collocated grid. The governing equations are then imposed on these selected nodes. All relevant interpolations as well as boundary treatments are presented. The system is solved using a fully explicit second-order time scheme, where the pressure Poisson equation is solved by a pseudo-time marching procedure. The implementation of the method is illustrated through application to the incompressible Navier–Stokes flow in a shear-driven cavity, a temporally periodic flow, and a spatially periodic flow. The results demonstrate the success of the method, and error study supports the high order of accuracy of the method. It is also demonstrated that wall effects reduce the formal order of accuracy.

ACKNOWLEDGEMENTS

The support for this work was in part provided by Grant CTS-0237951 from the National Science Foundation.

REFERENCES

1. Kopriva DA, Kolias JH. A conservative staggered-grid Chebyshev multidomain method for compressible flows. *Journal of Computational Physics* 1996; **125**:244–261.
2. Yeung RW, Ananthakrishnan P. Oscillation of a floating body in a viscous fluid. *Journal of Engineering Mathematics* 1992; **26**:211–230.
3. Yeung RW, Ananthakrishnan P. Viscosity and surface-tension effects on wave generation by a translating body. *Journal of Engineering Mathematics* 1997; **32**:257–280.
4. Orszag SA, Israeli M. Numerical simulation of viscous incompressible flows. *Annual Review of Fluid Mechanics* 1974; **6**:281–318.
5. Hirsh RS. Higher order accurate difference solutions of fluid mechanics problems by a compact differencing technique. *Journal of Computational Physics* 1975; **19**:90–109.
6. Lele SK. Compact finite difference schemes with spectral-like resolution. *Journal of Computational Physics* 1992; **103**:16–42.

7. Adam Y. Highly accurate compact implicit methods and boundary conditions. *Journal of Computational Physics* 1977; **24**:10–22.
8. Rubin SG, Khosla PK. Polynomial interpolation methods for viscous flow calculations. *Journal of Computational Physics* 1977; **24**:217–244.
9. Gupta MM, Manohar RP, Stephenson JW. A single cell high order scheme for the convection-diffusion equation with variable coefficients. *International Journal for Numerical Methods in Fluids* 1984; **4**:641–651.
10. Li M, Tang T, Fornberg B. A compact fourth-order finite difference scheme for the steady incompressible Navier–Stokes equations. *International Journal for Numerical Methods in Fluids* 1995; **20**:1137–1151.
11. Gamet L, Ducros F, Nicoud F, Poinsot T. Compact finite difference schemes on non-uniform meshes. Application to direct numerical simulations of compressible flows. *International Journal for Numerical Methods in Fluids* 1999; **29**:159–191.
12. Zingg DW. Comparison of high-accuracy finite-difference methods for linear wave propagation. *SIAM Journal on Scientific Computing* 2000; **22**(2):476–502.
13. Visbal MR, Gaitonde DV. On the use of higher-order finite-difference schemes on curvilinear and deforming meshes. *Journal of Computational Physics* 2002; **181**:155–185.
14. Zhang J. Numerical simulation of 2D square driven cavity using fourth-order compact finite difference schemes. *Computers and Mathematics with Applications* 2003; **45**:43–52.
15. Tian Z, Ge Y. A fourth-order compact finite difference scheme for the steady stream function-vorticity formulation of the Navier–Stokes/Boussinesq equations. *International Journal for Numerical Methods in Fluids* 2003; **41**:495–518.
16. Sengupta TK, Ganeriwal G, De S. Analysis of central and upwind compact schemes. *Journal of Computational Physics* 2003; **192**:677–694.
17. Nagarajan S, Lele SK, Ferziger JH. A robust high-order compact method for large eddy simulation. *Journal of Computational Physics* 2003; **191**:392–419.
18. Piller M, Stalio E. Finite-volume compact schemes on staggered grids. *Journal of Computational Physics* 2004; **197**:299–340.
19. Zhang KKQ, Minkowycz WJ, Mashayek F. Exact factorization technique for numerical simulations of incompressible Navier–Stokes flows. *International Journal of Heat and Mass Transfer* 2005, in press.
20. Ghia U, Ghia KN, Shin CT. High-Re solutions for incompressible flow using the Navier–Stokes equations and a multigrid method. *Journal of Computational Physics* 1982; **48**:387–411.
21. Johnstonm H, Liu JG. Finite difference schemes for incompressible flow based on local pressure boundary conditions. *Journal of Computational Physics* 2002; **180**:120–154.
22. Chorin AJ. Numerical solution of the Navier–Stokes equations. *Mathematics of Computation* 1968; **22**:745–762.

