Journal of Scientific Computing, Vol. 16, No. 1, March 2001 (© 2001)

A Compact Fourth-Order Finite Difference Scheme for Unsteady Viscous Incompressible Flows

Ming Li[1] and Tao Tang[2]

Received January 23, 2001; accepted (in revised form) April 11, 2001

In this paper, we extend a previous work on a compact scheme for the steady Navier–Stokes equations [Li, Tang, and Fornberg (1995), *Int. J. Numer. Methods Fluids*, **20**, 1137–1151] to the unsteady case. By exploiting the coupling relation between the streamfunction and vorticity equations, the Navier–Stokes equations are discretized in space within a 3×3 stencil such that a fourth order accuracy is achieved. The time derivatives are discretized in such a way as to maintain the compactness of the stencil. We explore several known time-stepping approaches including second-order BDF method, fourth-order BDF method and the Crank–Nicolson method. Numerical solutions are obtained for the driven cavity problem and are compared with solutions available in the literature. For large values of the Reynolds number, it is found that high-order time discretizations outperform the low-order ones.

KEY WORDS: Navier–Stokes equations; streamfunction; vorticity; compact scheme.

1991 Mathematics Subject Classification: 76D05, 65N06.

1. INTRODUCTION

In recent years, there has been some interest in designing high-order compact schemes for simulating viscous incompressible flows. For steady-state Navier–Stokes (NS) equations, these compact schemes [3, 5, 10, 18] are found to be computationally efficient and stable and yield highly accurate numerical solutions. In particular, E and Liu [7] were able to solve unsteady NS equations by using the fourth-order compact differencing in

[1] Department of Mathematics, Simon Fraser University, Burnaby, B.C. Canada V5A 1S6.

[2] Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong. E-mail: ttang@math.hkbu.edu.hk

space. With the use of fourth-order Runge–Kutta methods, they can solve the unsteady NS equations at very high Reynolds numbers. In their approach, the approximations needed to obtain the velocity to fourth-order accuracy extend outside the 3 × 3-point domain so the compactness holds only weakly for the vorticity equation. In a previous paper [18], we derive a compact fourth-order finite-difference scheme for the time-independent NS equations with the novelty of *genuine compactness*, i.e., the compact scheme is strictly within a nine-point stencil. It is shown that this scheme yields highly accurate numerical solutions while still allowing SOR-type iterations for large Reynolds numbers. The aim of the present work is to extend the genuine compact scheme proposed in [18] to deal with the unsteady NS equations.

The efficient solution procedure for the time-dependent NS equations requires elaborately designed finite difference schemes in both space and time, as well as the appropriate implementation of the numerical vorticity conditions. In this work we shall propose a numerical approach with these goals in mind. Using the idea of the method of lines, the NS equations are first discretized in space in a way similar to that of [18]. The time derivatives are then discretized in such a way as to maintain the compactness of the stencil. Several known time-stepping approaches will be investigated. The resulting system from the above procedure is nonlinear, but we can decompose it into two subsystems: a linear system with respect to streamfunction and a nonlinear system with respect to vorticity. We deal with the former by an LU decomposition at the beginning and simple back substitutions afterwards. The latter subsystem is solved by the GMRES procedure [20]. These two subprocesses are coupled by boundary conditions with several inner iterations. Numerical experiments show that the overall procedure is stable and efficient.

The paper is organized as follows. In next section we give the derivation and implementation of the numerical schemes. An application to the driven cavity problem is presented in Section 3, and some concluding remarks will be made in the final section.

2. NUMERICAL SCHEMES

2.1. Spatial Discretization

We shall first use the method of lines to carry out a semi-discretization, i.e., proceed with spatial discretization while retaining the temporal derivative. Time-marching is treated separately by efficient temporal approximations. The streamfunction-vorticity form of the unsteady NS equations is written by

$$\psi_{xx}+\psi_{yy}=-\zeta \tag{2..1}$$

$$\zeta_t+\psi_y\zeta_x-\psi_x\zeta_y=\frac{1}{\mathsf{R}}(\zeta_{xx}+\zeta_{yy}) \tag{2.2}$$

Here ψ is the streamfunction, ζ the vorticity, and R the non-dimensional Reynolds number. For brevity we only consider the grid network with the uniform meshsize h in both x- and y-directions. We number the grid points (x, y), $(x+h, y)$, $(x, y+h)$, $(x-h, y)$, $(x, y-h)$, $(x+h, y+h)$, $(x-h, y+h)$, $(x-h, y-h)$, $(x+h, y-h)$ as 0, 1, 2, 3, 4, 5, 6, 7, 8, respectively (see Fig. 1). In writing the finite difference approximations a single subscript j denotes the corresponding function value at the grid point numbered j.

The compact scheme for (2.1) can be given in a straightforward way (see, e.g., [18]):

$$\begin{aligned}&8(\psi_1+\psi_2+\psi_3+\psi_4)+2(\psi_5+\psi_6+\psi_7+\psi_8)-40\psi_0\\&\quad=-h^2(\zeta_1+\zeta_2+\zeta_3+\zeta_4+8\zeta_0)\end{aligned} \tag{2.3}$$

The existence of the nonlinear terms in Eq. (2.2) creates extra complexity in deriving a compact scheme. The basic idea of the derivation is to cancel the higher order terms in the Taylor expansions by using the original differential equations with additional differentiation to appropriate order. For (2.2) this process results in third derivatives such as ψ_{xxx} and ψ_{yyy} which can not be directly approximated within a nine-point stencil. However, these terms can be replaced by invoking the differentiation of (2.1) and (2.2) in a coupled way. The procedure is rather tedious and we will defer the details to an appendix. Here we only give the resulting scheme:

$$8(\zeta_1+\zeta_2+\zeta_3+\zeta_4)+2(\zeta_5+\zeta_6+\zeta_7+\zeta_8)-40\zeta_0+T_1=T_2 \tag{2.4}$$

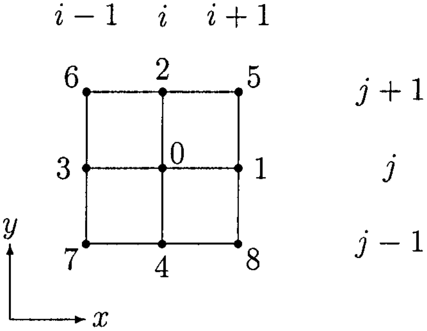

Fig. 1. Computational stencil.

where

$$T_1 = -\mathsf{R}h^2\left(\left(1-\frac{\mathsf{R}\psi_{24}}{4}\right)\frac{\partial\zeta_1}{\partial t}+\left(1+\frac{\mathsf{R}\psi_{13}}{4}\right)\frac{\partial\zeta_2}{\partial t}\right.$$

$$\left.+\left(1+\frac{\mathsf{R}\psi_{24}}{4}\right)\frac{\partial\zeta_3}{\partial t}+\left(1-\frac{\mathsf{R}\psi_{13}}{4}\right)\frac{\partial\zeta_4}{\partial t}+8\,\frac{\partial\zeta_0}{\partial t}\right) \tag{2.5}$$

$$T_2 = \mathsf{R}(\psi_{24}\zeta_{13}-\psi_{13}\zeta_{24}-\zeta_1\psi_{85}-\zeta_2\psi_{56}-\zeta_2\psi_{56}-\zeta_3\psi_{67}-\zeta_4\psi_{78}$$

$$-\zeta_5\psi_{12}-\zeta_6\psi_{23}-\zeta_7\psi_{34}-\zeta_8\psi_{41})$$

$$+\frac{\mathsf{R}^2}{4}\left(\psi_{13}\zeta_{13}\psi_{204}+\psi_{24}\zeta_{24}\psi_{103}+\frac{1}{2}\,\psi_{13}\psi_{24}(\zeta_{56}+\zeta_{78})\right.$$

$$\left.-\frac{1}{4}\,(\psi_{13}\zeta_{24}+\psi_{24}\zeta_{13})(\psi_{56}+\psi_{78})-\psi_{13}{}^2\zeta_{204}-\psi_{24}{}^2\zeta_{103}\right) \tag{2.6}$$

In the above expressions we have used the notation $f_{ij} := f_i - f_j$, $f_{ikj} := f_i - 2f_k + f_j$.

The schemes (2.3) and (2.4) formulate the fourth-order compact approximations in space only. It can be seen that T_1 in (2.4) is related to the temporal derivative of the vorticity function. Without T_1, the scheme (2.3)–(2.4) is in coincidence with the compact scheme for the steady problem as obtained in [18].

2.2. Temporal Discretizations

The semi-discretized scheme (2.4) is based on the method of lines. A variety of temporal discretizations can be applied to further formulate a full nonlinear system. Combined with (2.3), we have a system of differential-algebraic equations (DAE) which seemingly can be solved by typical numerical techniques for DAEs (see [2]). However, due to the complexity of (2.4) and the large size of the system, the DAE approach is not economical. We shall employ instead a sequential solution method for (2.3) and (2.4). In this subsection we discuss the temporal differencing. The detailed implementation of the whole process will be given later.

Equation (2.2) is of diffusion-convection type. Much effort has been made to design stable and efficient time differencing for such equations, often using explicit schemes for the convection term and implicit schemes for the diffusion term. This kind of approach has been termed as IMEX (implicit-explicit) schemes and is well documented in [1]. Unfortunately, the coefficients of the diffusion terms in (2.2) depend on the streamfunction values which need to be updated at each time level. Hence we cannot design an IMEX-type scheme with universal explicit treatment for ζ at all time levels. For this reason, we will use multistep implicit schemes based on backward time-differencing. This schemes, called backward differentiation formula (BDF), have been proved highly stable and widely used for solving stiff ODEs and DAEs (see, e.g., [2, 14]).

As a model ordinary differential equation, consider

$$\frac{\partial u}{\partial t} = f \tag{2.7}$$

Denoting the time level by a superscript and the stepsize by k, we summarize some relevant BDF formulas as follows. The detail discussion for these formulas can be found in, e.g., [21].

- Second-order BDF formula:

$$\frac{1}{2k}(3u^n - 4u^{n-1} + u^{n-2}) = f^n \tag{2.8}$$

- Fourth-order BDF formula:

$$\frac{1}{12k}(25u^n - 48u^{n-1} + 36u^{n-2} - 16u^{n-3} + 3u^{n-4}) = f^n \tag{2.9}$$

For multistep BDF methods we also need difference formulas for several initial steps. In this paper we shall use the following finite difference schemes:

- Crank–Nicolson (second-order accuracy) using two time levels:

$$\frac{1}{k}(u^n - u^{n-1}) = \frac{1}{2}(f^n + f^{n-1}) \tag{2.10}$$

- Fourth-order accuracy using three time levels:

$$\frac{1}{2k}(u^n - u^{n-2}) = \frac{1}{6}(f^n + 4f^{n-1} + f^{n-2}) \tag{2.11}$$

- Fourth-order accuracy using four time levels:

$$\frac{1}{24k}(17u^n - 9u^{n-1} - 9u^{n-2} + u^{n-3}) = \frac{1}{4}(f^n + 3f^{n-1}) \tag{2.12}$$

2.3. Boundary Condition

Applying appropriate time differencing in the previous subsection to (2.4) results in a full discretization for Eq. (2.2):

$$8(\zeta_1+\zeta_2+\zeta_3+\zeta_4)+2(\zeta_5+\zeta_6+\zeta_7+\zeta_8)-40\zeta_0+T_1'=T_2 \tag{2.13}$$

where T_1' is the variation form of T_1 with all time derivatives replaced by desired temporal discretizations. At first sight, (2.3) and (2.13) are a system of nonlinear equations with $2N$ unknowns, with N the total number of grid points used. However, by observing that (2.13) is linear with respect to vorticity, we can solve the whole system in a sequential or uncoupled way, i.e., first solve linear system (2.3) for ψ, then solve linear system (2.13) for ζ. This inner iteration at each time level will be performed until the difference between consecutive solutions for ψ or ζ falls within a pre-assigned tolerance. The description for this process can be found in [19].

In the streamfunction-vorticity formulation, numerical vorticity must be specified at the boundaries. Commonly this is accomplished by linking the desired boundary vorticity values with the adjacent streamfunction/vorticity values. Among the frequently used formulas are Wood's formula [23] and Thom's formula [22]. In addition, many other versions based on Taylor expansions have been developed, see, e.g., [13] and [15]. Recently, more rigorous investigations have been carried out towards the implementation of the boundary vorticity from a global and dynamical point of view, see [12, 24] and the references therein. An observation has also been made that some popular global constraints for the numerical boundary vorticity can be interpreted as variation forms of classical local formulas [8]. Mathematically, the boundary conditions are

$$\psi=0, \qquad \frac{\partial\psi}{\partial n}=v \tag{2.14}$$

where n is the direction of the outward normal vector and v is the tangential component of the velocity at the physical boundary. With regard to numerics, it is the boundary treatment that provides a coupling between two seemingly separate solution processes for ψ and ζ, and it is so crucial that inappropriate treatment will lead to failure of convergence in the inner iterations even if a large amount of under-relaxation is used. Here we shall use the numerical boundary conditions as described in [16, 17, 18]. The compact schemes (2.3) and (2.13) are to be implemented in the region $[2h, 1-2h]\times[2h, 1-2h]$ (assuming the physical region is $[0, 1]\times[0, 1]$), so the grid function values of ψ and ζ must be specified on the grid points next to the physical boundaries. Denoting the grid point on a physical boundary by the subscript 0 and the jth internal point from the boundary by the subscript j, we have

$$18\psi_1 - 9\psi_2 + 2\psi_3 = 11\psi_0 - 6h \left.\frac{\partial\psi}{\partial n}\right|_0 + O(h^4)$$

which becomes

$$\psi_1 = \frac{\psi_2}{2} - \frac{\psi_3}{9} - \frac{h}{3} v \tag{2.15}$$

by using (2.14). The vorticity value is determined by a second-order central difference for (2.1), i.e.,

$$\zeta_1 = -\frac{1}{h^2} (\psi_2 + \psi_a + \psi_b - 4\psi_1) \tag{2.16}$$

where subscript a and b denote the two points adjacent to point 1 along the direction parallel to the boundary.

2.4. Solving Nonlinear Systems

Our goal is to solve systems (2.3), (2.15) and (2.13), (2.16) in an effective way. The former system has a constant coefficient matrix so we can perform LU decomposition at the beginning, which is to be used to find solutions for ψ values at all time levels. The nonlinear system for ζ carries the main workload owing to the variable coefficients. We shall use an GMRES method with a Jacobian preconditioner to solve this linear system [20]. Although other linear system solvers can also be used, an advantage of the GMRES method is that it is particularly efficient fora large nonsymmetric system.

2.5. Algorithm

In summary, the solution process is implemented as follows:

- Step 1. Initialize the first few steps for multistep methods.
- Step 2. Evaluate boundary vorticity ζ_{bdry} from (2.16).
- Step 3. Solve for ζ using (2.13).
- Step 4. Solve for ψ using (2.3) and (2.15).

```
• Step 5. Store $\zeta_{bdry}$ as $\zeta_{bdry}^{old}$ and update $\zeta_{bdry}$ using (2.16).
  If $\|\zeta_{bdry} - \zeta_{bdry}^{old}\| \leqslant$ tolerance
      then goto Step 6
      then goto Step 3

• Step 6. Increase time step by $\Delta t$, and goto Step 3.
```

The tolerance in the above algorithm is dependent on the time step and should be sufficiently small. In practice, smaller mesh size requires smaller tolerance. In our computations, it was chosen in the order of $O(10^{-5})$ for the time step $k = O(10^{-2})$.

3. NUMERICAL RESULTS FOR THE DRIVEN CAVITY PROBLEM

The drive cavity flow is a standard test problem for numerical computations for the viscous incompressible flows. The fluid flow is governed by Eqs. (2.1) and (2.2) in a square cavity. The boundary conditions are given by (2.14) with $v = 1$ at the top boundary and $v = 0$ at the other three boundaries. For this problem there are ample computational results available in the literature. We shall compare our results with the benchmark solutions of Goodrich and Soh [11] and Chudanov *et al.* [4].

The solution procedure is described in the previous section. Three time integration methods have been tested: the Crank–Nicolson scheme (CN), the second- and fourth-order BDF schemes (BDF2 and BDF4, respectively). For multistep BDF schemes, (2.10) is used to give solutions at the second time step. This is done by simple point SOR iterations and the convergence is fast in this very early stage. The use of a second-order scheme as initialization at the second time level for BDF4 will not deteriorate the global fourth-order accuracy for the NS equations (see, e.g., [21]). The initial values at the third and fourth time levels for BDF4 are obtained by using (2.10) and (2.11), respectively. All three schemes have been tested for different grid sizes and Reynolds numbers. In this paper we present results for the Reynolds number $\mathsf{R} = 400$, 1000, 3200. The choice of the time increment Δt is an interesting issue in time-dependent computations. Efficiency in time-marching can be achieved by tailored techniques, such as variable-size BDF formulas, self-adjusting time integrations etc. However, we will not go into detail here. With uniform time step, similar to the analysis given at the end of this section, it is expected that Δt is proportional to R^{-1}. Therefore, in all computations we use $\Delta t = 0.05$ for $\mathsf{R} = 400$, $\Delta t = 0.025$ for $\mathsf{R} = 1000$ and $\Delta t = 0.01$ for $\mathsf{R} = 3200$. The number of inner boundary iterations at each time level ranges from five to seven at the early stage and two to four for large time. For higher Reynolds number and/or finer

Table I. $\mathsf{R}=400$: The Top Row Represents the Maximum and Minimum Streamfunction Values and the Bottom Row the Corresponding Vorticity at the Same Positions

	N	$t=5$		$t=10$		$t=20$	
CN	41	−0.8978(−1)	0.1286(−4)	−0.1050	0.3920(−3)	−0.1120	0.5346(−3)
		3.4311	−0.2679(−1)	2.5516	−0.3002	2.3192	−0.3635
	61	−0.9006(−1)	0.1857(−4)	−0.1054	0.4406(−3)	−0.1124	0.5930(−3)
		3.4041	−0.3185(−1)	2.5383	−0.2875	2.2997	−0.4090
BDF2	41	−0.8981(−1)	0.1180(−4)	−0.1052	0.3842(−3)	−0.1124	0.5293(−3)
		3.4374	−0.2725(−1)	2.5427	−0.2989	2.3117	−0.3634
	61	−0.9007(−1)	0.1871(−4)	−0.1055	0.4471(−3)	−0.1126	0.6007(−3)
		3.4169	−0.3194(−1)	2.5484	−0.2871	2.3142	−0.4086
BDF4	41	−0.8980(−1)	0.1094(−4)	−0.1053	0.3765(−3)	−0.1125	0.5232(−3)
		3.5037	−0.3071(−1)	2.5560	−0.2931	2.3042	−0.3640
	61	−0.9004(−1)	0.1817(−4)	−0.1055	0.4457(−3)	−0.1126	0.5996(−3)
		3.5037	−0.3455(−1)	2.5627	−0.2808	2.3146	−0.4084

grids, smaller Δt must be used in order to guarantee the convergence of the boundary iteration.

In Tables I and II we present the flow features of the primary vortex at the Reynolds numbers 400 and 1000, where in each entry the top row represents the maximum and minimum streamfunction values and the bottom row represents the corresponding vorticity values at the same position. It can be seen that for $\mathsf{R}=400$ and 1000, all three schemes agree fairly well even when relatively coarse grids are used. These results also agree well with the data given in [4]. The time-evolutions of the unsteady flows for $\mathsf{R}=1000$ and 3200 are depicted in Figs. 2 and 3, which compare well with

Table II. Same as Table I, Except that $\mathsf{R}=1000$

		$t=5$		$t=10$		$t=20$	
CN	41	−0.7975(−1)	0.1707(−4)	−0.9840(−1)	0.7495(−3)	−0.1143	0.1257(−5)
		4.6102	−0.5159(−1)	3.2772	−0.3676	2.6542	−0.7917
	61	−0.7980(−1)	0.9279(−5)	−0.9928(−1)	0.6932(−3)	−0.1141	0.1500(−2)
		4.5811	−0.4776(−1)	3.0639	−0.3369	2.3336	−0.9287
BDF2	41	−0.7926(−1)	0.2699(−6)	−0.9878(−1)	0.4272(−3)	−0.1129	0.1122(−2)
		4.4376	−0.4218(−2)	2.9161	−0.2502	2.1393	−0.6969
	61	−0.7994(−1)	0.3070(−5)	−0.9940(−1)	0.5747(−3)	−0.1137	0.1405(−2)
		4.5433	−0.8378(−1)	2.9891	−0.3373	2.1973	−0.9204
BDF4	41	−0.7925(−1)	0.2529(−6)	−0.9860(−1)	0.5157(−3)	−0.1133	0.1183(−2)
		4.5105	−0.1879(−2)	2.9884	−0.3687	2.3142	−0.7187
	61	−0.7994(−1)	0.6616(−6)	−0.9939(−1)	0.5336(−3)	−0.1133	0.1345(−2)
		4.5603	−0.1243(−1)	2.9537	−0.3353	2.1255	−0.8921

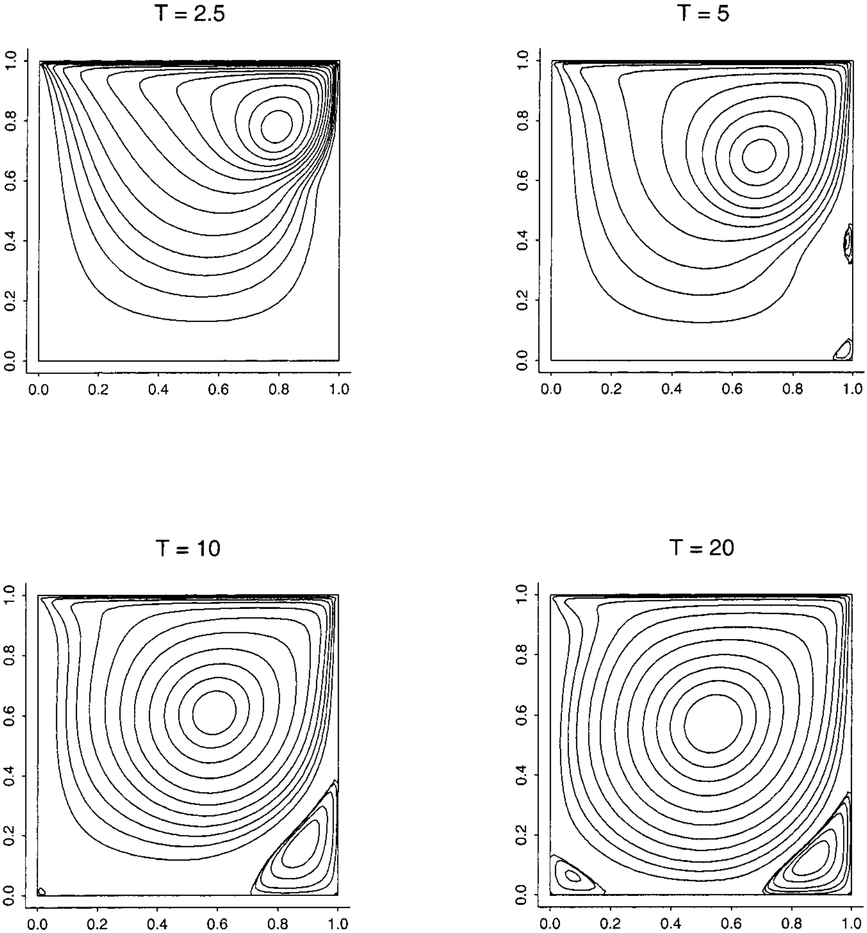

Fig. 2. Time-evolution of the unsteady cavity flow at $\mathsf{R} = 1000$.

those in [11, 7, 4]. All of the figures are obtained by using BDF4 with 81×81 grid points in space, and $\Delta t = 0.02$ and 0.01 for $\mathsf{R} = 1000$ and 3200, respectively.

For small values of the Reynolds number, the three time integration methods demonstrate a similar behavior. However, high-order time discretizations outperform the low-order ones for problems with larger values of the Reynolds number. For $\mathsf{R} = 3200$ with mesh size $h = 1/80$ and time step $\Delta t = 0.01$, BDF4 produces good numerical results, while BDF2 and CN *fail to converge* in the boundary iterations. The main reason for this is due to the different stability properties of these methods. Although the

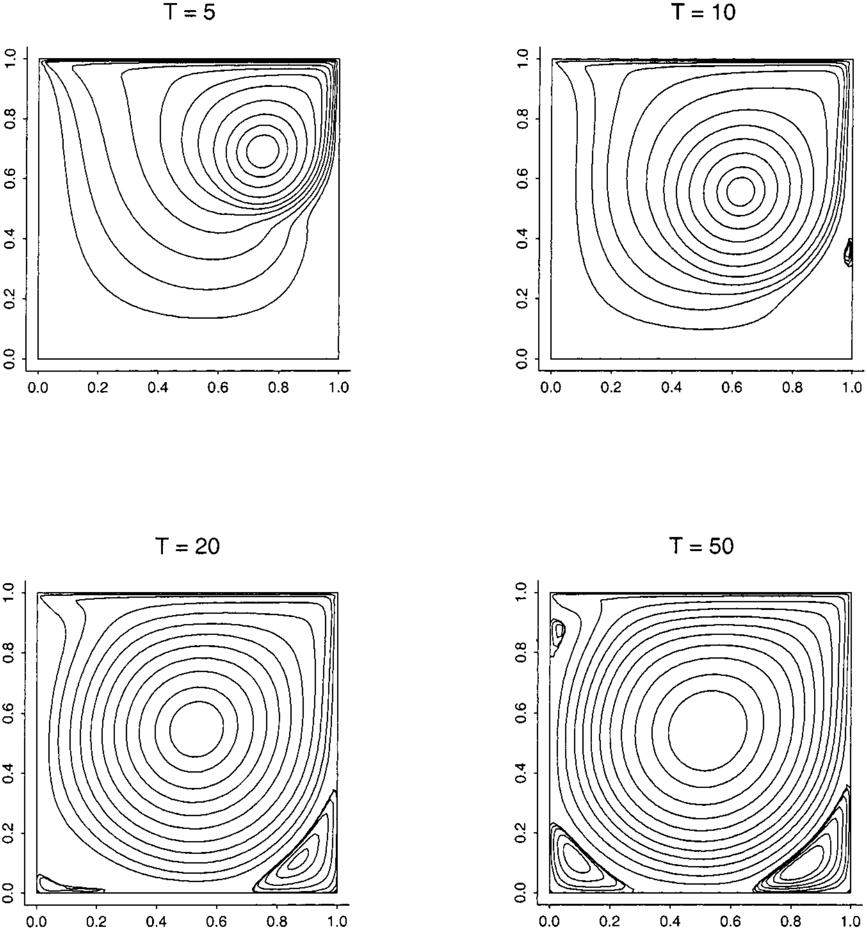

Fig. 3. Time-evolution of the unsteady cavity flow at $\mathsf{R} = 3200$.

formal CN method is implicit which guarantees the stability of the scheme, the way we are implementing it is based on simple iterations for the implicit system. Of course, this approach will make the overall procedure more efficient, but the iteration will significantly reduce the stability property of the method. It was found (and analyzed) by E and Liu [7] that some higher order methods such as 4th order Runge–Kutta method not only increase the accuracy of the time integration but also increase the stability of the overall scheme. The stability property (which controls the choice of the practical time steps) is more important than the temporal accuracy for the driven cavity problem, since the solution will yield a steady state for large time. With the same analysis method of [7] and [6], it can be shown that the higher order method BDF4 is more stable than the BDF2 and (the iterated version of) the Crank–Nicolson method.

The stability problem is also well understood by using the terminology of the so-called cell Reynolds number, see e.g., [6]. Consider the simplest convection-diffusion equation

$$u_t = au_x = \nu u_{xx} \tag{3.1}$$

The analog of Reynolds number is $\mathsf{R} = aL/\nu$. If we solve this problem using forward Euler in time, and central difference in space

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + a\,\frac{u_{j=1}^n - u_{j-1}^n}{2\Delta x} = \nu\,\frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta x^2} \tag{3.2}$$

then the amplification factor in the von Neumann analysis is

$$g(\xi) = 1 + \Delta t\left\{-a\,\frac{\sin\xi}{\Delta x} + \nu\,\frac{2(\cos\xi - 1)}{\Delta x^2}\right\} \tag{3.3}$$

For stability, we need

$$|g(\xi)| \leqslant 1 + \bar{C}\,\Delta t \tag{3.4}$$

for all ξ. This translates to $\bar{C} = a^2/\nu = a\mathsf{R}/L$, and the stability constraint is $\Delta t/\nu\,\Delta x^2 < \frac{1}{2}$. If we had treated the diffusion term implicitly,

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + a\,\frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} = \nu\,\frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{\Delta x^2} \tag{3.5}$$

Then stability requirement (3.4) would be satisfied with $\bar{C} = 2a^2/\nu$ but no other conditions on Δt other than Δt being sufficiently small. In any case, these schemes are useless at high Reynolds number even though they are stable at fixed Reynolds number since the error grows as $e^{\bar{C}t} = e^{c\mathsf{R}t}$, where c is a positive constant. This means that to design schemes that work for high Reynolds number, the standard stability concept is not enough as a designing principle. What we need is uniform stability with respect to Reynolds number, i.e., $\bar{C}$ in (3.4) should be independent of R. It was pointed out in [7, 6] that this can be done by switching to a time stepping scheme whose stability region covers a neighborhood of the imaginary axis near the origin. Examples of such schemes include the 3rd and 4th order Runge–Kutta, 3rd and 4th order Adams–Bashforth formulas. The above analysis seems also valid for the present high-order space discretization approach, since here the main problem is also the stability of the temporal discretization.

4. CONCLUDING REMARKS

In this paper we have extended a previous work on a compact scheme for the steady Navier–Stokes equations to the unsteady case. The genuine compact scheme developed in [18] provides a useful framework for spatial discretization. Using the method of lines with implicit time differencing, we have obtained a fully discretized system which has good stability properties for time-marching. The numerical solution of the resulting system is obtained by a sequential solution process. This idea can be traced back as early as in 70s, see e.g., [19], but few successful applications have been reported in the literature since. As pointed out by E and Liu [7], the main difficulty is how to guarantee the convergence of the inner boundary iterations at each time level. The numerical experiments presented in this paper justify the applicability and robustness of the genuine compact schemes. In practice, the efficiency of the proposed schemes should be improved by using variable step (higher-order) BDF formulas. A specific procedure is required to estimate the amount of enlargement of the time step. This issue remains to be further investigated.

APPENDIX: FOURTH-ORDER COMPACT SCHEME FOR THE VORTICITY EQUATION

In this appendix we present the derivation of the fourth-order compact scheme for the unsteady vorticity equation. The stencil structure and grid numbering are described in Section 2. For convenience we restate the Ns equations:

$$\psi_{xx} + \psi_{yy} = -\zeta \tag{4.1}$$

$$\frac{\partial \zeta}{\partial t} + g = \frac{1}{\mathsf{R}} (\zeta_{xx} + \zeta_{yy}) \tag{4.2}$$

where we have denoted

$$g(x,\ y) = \psi_y \zeta_x - \psi_x \zeta_y \tag{4.3}$$

By Taylor expansion, (4.2) can be rewritten as

$$\frac{\partial \zeta}{\partial t} + g = \frac{1}{\mathsf{R}} \left(\frac{\zeta_1 + \zeta_2 + \zeta_3 + \zeta_4 - 4\zeta_0}{h^2} - \frac{h^2}{12} (\zeta_{xxxx} + \zeta_{yyyy}) \right) + O(h^4) \tag{4.4}$$

On the other hand, differentiating (4.2) yields

$$\frac{\partial(\zeta_{xx} + \zeta_{yy})}{\partial t} + g_{xx} + g_{yy} = \frac{1}{\mathsf{R}} (\zeta_{xxxx} + \zeta_{yyyy} + 2\zeta_{xxyy}) \tag{4.5}$$

It follows from (4.4) and (4.5), (4.4) + $(h^2/12) \times$ (4.5), that

$$\begin{aligned} &\frac{\partial \zeta}{\partial t} + \frac{h^2}{12} \frac{\partial}{\partial t} (\zeta_{xx} + \zeta_{yy}) + g + \frac{h^2}{12} (g_{xx} + g_{yy}) \\ &\quad = \frac{1}{\mathsf{R}} \left(\frac{\zeta_1 + \zeta_2 + \zeta_3 + \zeta_4 - 4\zeta_0}{h^2} + \frac{h^2}{6} \zeta_{xxyy} \right) + O(h^4) \\ &\quad = \frac{1}{\mathsf{R}} \frac{1}{6/h^2} (4(\zeta_1 + \zeta_2 + \zeta_3 + \zeta_4) + \zeta_5 + \zeta_6 + \zeta_7 + \zeta_8 - 20\zeta_0) + O(h^4) \end{aligned} \tag{4.6}$$

where we have used the fact that

$$\zeta_{xxyy} = h^{-4}(\zeta_5 + \zeta_6 + \zeta_7 + \zeta_8 - 2(\zeta_1 + \zeta_2 + \zeta_3 + \zeta_4 + 4\zeta_0)) + O(h^2)$$

Next we proceed to approximate g and $g_{xx} + g_{yy}$ on the left hand side of (4.6) within a nine-point stencil. Using a Taylor expansion for $\psi_1 - \psi_3 := \psi_{13}$ and $\psi_2 - \psi_4 := \psi_{24}$ up to $O(h^4)$ gives

$$\begin{aligned} &\frac{\psi_{24}\psi_{13} - \psi_{13}\psi_{24}}{4h^4} \\ &\quad = g + \frac{h^2}{6} (\zeta_x \psi_{yyy} + \zeta_y \psi_{xxx} - \psi_x \zeta_{yyy} - \psi_y \zeta_{xxx}) + O(h^4) \end{aligned} \tag{4.7}$$

The third derivative terms in (4.7) are troublesome since the corresponding $O(h^2)$ approximations require grid function values outside the 3×3 stencil.

However, this can be fixed by replacing them with some mixed third derivatives. From (4.1) and (4.2) we have

$$\psi_{xxx} = -\psi_{xyy} - \zeta_x$$

$$\psi_{yyy} = -\psi_{xxy} - \zeta_y$$

$$\zeta_{xxx} = \mathsf{R}\left(\frac{\partial \zeta_x}{\partial t} + g_x - \zeta_{xyy}\right)$$

$$\zeta_{yyy} = \mathsf{R}\left(\frac{\partial \zeta_y}{\partial t} + g_y - \zeta_{xxy}\right)$$

Substituting the above terms into the right hand side of (4.7) and making suitable rearrangements, yield

$$g = \frac{\psi_{24}\zeta_{13} - \psi_{13}\zeta_{24}}{4h^2} + \frac{\mathsf{R}h^2}{6}\left(\psi_x \frac{\partial \zeta_y}{\partial t} - \psi_y \frac{\partial \zeta_y}{\partial t} + \psi_x g_y - \psi_y g_x\right)$$

$$-\frac{h^2}{6}(\psi_x \zeta_{xxy} + \zeta_y \psi_{xyy} - \psi_y \zeta_{xyy} - \zeta_x \psi_{xxy}) + O(h^4)$$

For the term $g_{xx} + g_{yy}$, we differentiate $g = \psi_y \zeta_x - \psi_x \zeta_y$ to obtain

$$g_{xx} + g_{yy} = \zeta_x(\psi_{xx} + \psi_{yy})_y - \zeta_y(\psi_{xx} + \psi_{yy})_x + 2\zeta_{xy}(\psi_{xx} - \psi_{yy})$$

$$+ 2\psi_{xy}(\zeta_{xx} - \zeta_{yy}) + \psi_y(\zeta_{xx} + \zeta_{yy})_x - \psi_x(\zeta_{xx} + \zeta_{yy})_y$$

By using the NS Eqs. (4.1)–(4.2) the above equation can be simplified as

$$g_{xx} + g_{yy} = 2\zeta_{xy}(\psi_{xx} - \psi_{yy}) + 2\psi_{xy}(\zeta_{xx} - \zeta_{yy})$$

$$+ \frac{\mathsf{R}h^2}{6}\left(\psi_x \frac{\partial \zeta_y}{\partial t} - \psi_y \frac{\partial \zeta_y}{\partial t} + \psi_x g_y - \psi_y g_x\right)$$

Combining the above results for g and $g_{xx} + g_{yy}$ yields

$$g + \frac{h^2}{12}(g_{xx} + g_{yy}) = \frac{\psi_{24}\zeta_{13} - \psi_{13}\zeta_{24}}{4h^2} + \frac{h^2}{6}(\zeta_{xy}(\psi_{xx} - \psi_{yy}) + \psi_{xy}(\zeta_{xx} - \zeta_{yy}))$$

$$-\frac{h^2}{6}(\psi_x \zeta_{xxy} + \zeta_y \psi_{xyy} - \psi_y \zeta_{xyy} - \zeta_x \psi_{xxy})$$

$$-\frac{\mathsf{R}h^2}{12}\left(\psi_y \frac{\partial \zeta_x}{\partial t} - \psi_x \frac{\partial \zeta_y}{\partial t}\right) - \frac{\mathsf{R}h^2}{12}(\psi_y g_x - \psi_x g_y) + O(h^4)$$

By expanding g_x and g_y, we obtain

$$\begin{aligned}
g+\frac{h^2}{12}(g_{xx}+g_{yy})
&=\frac{\psi_{24}\zeta_{13}-\psi_{13}\zeta_{24}}{4h^2}+\frac{h^2}{6}(\zeta_{xy}(\psi_{xx}-\psi_{yy})+\psi_{xy}(\zeta_{xx}-\zeta_{yy}))\\
&\quad-\frac{h^2}{6}(\psi_x\zeta_{xxy}+\zeta_y\psi_{xyy}-\psi_y\zeta_{xyy}-\zeta_x\psi_{xxy})\\
&\quad-\frac{\mathsf{R}h^2}{12}\left(\psi_y\frac{\partial\zeta_x}{\partial t}-\psi_x\frac{\partial\zeta_y}{\partial t}\right)+O(h^4)\\
&\quad-\frac{\mathsf{R}h^2}{12}((\psi_x\zeta_y+\zeta_x\psi_y)\,\psi_{xy}-2\psi_x\psi_y\zeta_{xy}-\psi_x\zeta_x\psi_{yy}\\
&\quad-\psi_y\zeta_y\psi_{xx}+\psi_x^2\zeta_{yy}+\psi_y^2\zeta_{xx})
\end{aligned}\tag{4.8}$$

The compact scheme (2.4) can be obtained by combining (4.6) and (4.8), together with the following finite difference formulas:

$$u_{xy}=\frac{1}{4h^2}(u_5-u_6+u_7-u_8)+O(h^2)$$

$$u_{xxy}=\frac{1}{2h^3}(u_5+u_6-u_7-u_8-2(u_2-u_4))+O(h^2)$$

$$u_{xyy}=\frac{1}{2h^3}(u_5+u_6-u_6-u_7-2(u_1-u_3))+O(h^2)$$

ACKNOWLEDGMENTS

We would like to thank the referees for careful reading for the manuscript and for helpful suggestions. This research was partially supported by NSERC Canada under grant number OGP0105545 and by Hong Kong Baptist University under grant numbers FRG/98-99/II-14 and 15.

REFERENCES

1. Ascher, U., Ruuth, S. J., and Wetton, B. T. R. (1995). Implicit-explicit methods for time-dependent partial differential equations. *SIAM J. Numer. Anal.* **32**, 797–823.
2. Ascher, U., and Petzold, L. R. (1998). *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*, SIAM, Philadelphia.
3. Chen, G. Q., Gao, Z., and Yang, Z. F. (1993). A perturbation h^4 exponential finite difference scheme for the convective diffusion equation. *J. Comput. Phys.* **104**, 129–139.

4. Chudanov, V. V., Popkov, A. G., Churbanov, A. G., Vabishchevich, P. N, and Makarov, M. M. (1995). Operator-splitting schemes for the stream function-vorticity formulation. *Comput. Fluids* **24**, 771–786.
5. Dennis, S. C. R, and Hudson, J. D. (1989). Compact h^4 finite-difference approximations to operators of Navier–Stokes type. *J. Comput. Phys.* **85**, 390–416.
6. E, W. (2001). Numerical methods for viscous incompressible flows: some recent advances. In *Proceedings of Workshops in Scientific Computing Hong Kong '99*, Z.-C. Shi *et al.* (eds.), Science Press, to appear.
7. E, W., and Liu, J.-G. (1996). Essentially compact schemes for unsteady viscous incompressible flows. *J. Comput. Phys.* **126**, 122–138.
8. E, W., and Liu, J.-G. (1996). Vorticity boundary condition and related issues for finite difference schemes. *J. Comput. Phys.* **124**, 368–382.
9. Ghia, U., Ghia, K. N., and Shin, C. T. (1982). High-Re solutions for incompressible flow using the Navier–Stokes equations and a multigrid method. *J. Comput. Phys.* **48**, 387–411.
10. Gupta, M. M. (1991). High accuracy solutions of incompressible Navier–Stokes equations. *J. Comput. Phys.* **93**, 343–359.
11. Goodrich, J. W., and Soh, W. Y. (1989). Time-dependent viscous incompressible Navier–Stokes equations: The finite difference galerkin formulation and streamfunction algorithms. *J. Comput. Phys.* **84**, 207–241.
12. Gresho, P. M. (1992). Some interesting issues in incompressible fluid dynamics, both in the continuum and in numerical simulation. *Adv. Appl. Mech.* **28**, 45–140.
13. Gupta, M. M., and Manohar, R. P. (1979). Boundary approximations and accuracy in viscous flow computations. *J. Comput. Phys.* **31**, 265–288.
14. Hairer, E., Norsett, S. P., and Wanner, G. (1993). *Solving Ordinary Differential Equations*, Springer, Berlin/New York, 2nd ed.
15. Hou, T. Y., and Wetton, B. T. R. (1992). Convergence of a finite difference scheme for the Navier–Stokes equations using vorticity boundary conditions. *SIAM J. Numer. Anal.* **29**, 615–639.
16. Huang, H., and Seymour, B. R. (1996). The no-slip boundary condition in finite difference approximations. *Int. J. Numer. Methods Fluids* **22**, 713–729.
17. Huang, H., and Seymour, B. R. (2000). Finite difference solutions for incompressible flow problems with corner singularities. *J. Sci. Comput.* **15**, 265–292.
18. Li, M., Tang, T., and Fornberg, B. (1995). A compact fourth-order finite difference scheme for the steady incompressible Navier–Stokes equations. *Int. J. Numer. Methods Fluids* **20**, 1137–1151.
19. Roache, P. J. (1972). *Computational Fluid Dynamics*, Hermosa Publishers, Albuquerque, New Mexico.
20. Saad, Y., and Schultz, M. H. (1986). GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.* **7**, 856–869.
21. Strikwerda, J. C. (1997). High-order-accurate schemes for incompressible viscous flow. *Int. J. Numer. Methods Fluids* **24**, 715–734.
22. Thom, A. (1933). The flow past circular cylinder at low speeds. *Proc. Roy. Soc. London* (*Ser. A*) **141**, 651–666.
23. Woods, L. C. (1954). A note on the numerical solution of fourth order differential equations. *Aeronaut. Quart.* **5**, 176–184.
24. Wu, J. Z., and Wu, J. M. (1996). Vorticity dynamics on boundaries. *Adv. Appl. Mech.* **32**, 119–275.