Computers & Fluids 39 (2010) 178–188

Contents lists available at ScienceDirect

Computers & Fluids

journal homepage: www.elsevier.com/locate/compfluid

# An explicit Chebyshev pseudospectral multigrid method for incompressible Navier–Stokes equations

W. Zhang, C.H. Zhang, G. Xi *

*Fluid Machinery and Engineering Department, School of Energy and Power Engineering, Xi'an Jiaotong University, No. 28 West Xianning Road, Xi'an 710049, Shaanxi, China*

ARTICLE INFO

*Article history:*
Received 19 May 2009
Received in revised form 2 August 2009
Accepted 6 August 2009
Available online 11 August 2009

ABSTRACT

The two-dimensional steady incompressible Navier–Stokes equations in the form of primitive variables have been solved by Chebyshev pseudospectral method. The pressure and velocities are coupled by artificial compressibility method and the NS equations are solved by pseudotime method with an explicit four-step Runge–Kutta integrator. In order to reduce the computational time cost, we propose the spectral multigrid algorithm in full approximation storage (FAS) scheme and implement it through V-cycle multigrid and full multigrid (FMG) strategies. Four iterative methods are designed including the single grid method; the full single grid method; the V-cycle multigrid method and the FMG method. The accuracy and efficiency of the numerical methods are validated by three test problems: the modified one-dimensional Burgers equation; the Taylor vortices and the two-dimensional lid driven cavity flow. The computational results fit well with the exact or benchmark solutions. The spectral accuracy can be maintained by the single grid method as well as the multigrid ones, while the time cost is greatly reduced by the latter. For the lid driven cavity flow problem, the FMG is proved to be the most efficient one among the four iterative methods. A speedup of nearly two orders of magnitude can be achieved by the three-level multigrid method and at least one order of magnitude by the two-level multigrid method.



## 1. Introduction

In the past several decades, many approaches have been proposed to solve the incompressible Navier–Stokes equations, while most of them were conducted with finite-difference (FD), finite-volume (FV) or finite-element (FE) method. These methods are widely used in practical computations in almost all the fields due to their simplicity, feasibility and reliability. They are easily to be employed in complex physical domain with the help of body-fitted multi-block structured or unstructured grid, and a lot of corresponding temporal-spatial discretization schemes and iterative techniques have been studied in detail. However, in certain situations such as weather forecasting and turbulence transition, the above conventional methods are not accurate enough to simulate the relevant physical phenomena since high accuracy solutions are required. This deficiency could not be easily resolved by grid refinement due to the memory storage limit, or by higher order discretization schemes due to the rapidly increasing time cost and complexity in dealing with boundary conditions, so various spectral methods are developed in order to get high accuracy results while minimizing the memory storage requirement. The pseudospectral method is a straightforward one because the corresponding program can be obtained directly by revising an original FD code. Moreover, it could be used to solve problems in complex geometries with the help of mapping [13] and domain decomposition method [24].

There are two main difficulties in solving the incompressible NS equations in primitive variables form by pseudospectral method. First, there is no independent governing equation for the pressure, only their gradients appear in the momentum equations, so specific procedures must be designed for the coupling of velocities and pressure. Second, the coefficient matrix of the discretized equations in FD/FV/FE is usually a sparse banded matrix while its counterpart in pseudospectral form is dense, which requires much more computations in one single iteration and the time cost increases exponentially with the number of collocation points. Botella [3,4] investigated the two-dimensional lid driven cavity flow problem with Chebyshev pseudospectral method and solved the discretized equations by projection method. In his computation the prediction step calculations of velocities are carried out by a direct Helmholtz solver and the projection step by Uzawa method; the pressure is obtained by solving a Poisson equation. This algorithm has been extended to the three dimensional cavity flow problem recently by Albensoeder and Kuhlmann [1]; similar method was used by Heinrich [23] for unsteady problem. However, this semi-implicit method is more complicated comparing with the explicit ones such as pseudotime method, which is widely used in the computations of compressible flows with FD/FV methods since

* Corresponding author. Tel.: +86 29 82665068; fax: +86 29 83234772.
*E-mail addresses:* zhangwei@stu.xjtu.edu.cn (W. Zhang), chzhang@mail.xjtu.edu.cn (C.H. Zhang), xiguang@mail.xjtu.edu.cn (G. Xi).

0045-7930/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compfluid.2009.08.001

Jameson [25] proposed it together with the multistep Runge–Kutta time stepping method. In the explicit solver, the incompressibility constraint can be fulfilled by the artificial compressibility method [10]. The main drawback of the explicit method, however, is that the time step is severely restricted by the stability limit as $\Delta t = O(1/N^4)$ in pseudospectral method with $N$ the number of collocation points, instead of $\Delta t = O(1/N^2)$ in FD/FV/FE methods. This restriction can be relieved by various acceleration techniques, the two commonly used are preconditioning [5,9] and spectral multigrid (SMG) [40,41].

The multigrid method (MG) has been widely used in accelerating the computation of NS equations since Brandt introduced it in solving the Poisson equation [6]; then it has been applied together with spectral element method [31,32]. However, based on the authors' experiences, there are not many researches carried out on pseudospectral multigrid method except the following ones. Zang et al. [40,41] applied the spectral multigrid method for elliptic equations, several simple model problems were tested in detail with Fourier and Chebyshev series as the base functions. Brandt et al. [7] improved the spectral multigrid for periodic elliptic problems and obtained an increase in efficiency and accuracy. Heinrichs et al. [21,22] and Phillips [34] performed spectral multigrid together with preconditioning, a finite difference preconditioner was proved effective to the test problems. The application of pseudospectral multigrid in CFD has been carried out by Chou [11] for the lid driven cavity problem and Krastev and Schäfer [28] for the thermally driven cavity problem; furthermore, it has been put into application in various research fields such as optimal control [17] and quantum mechanics [2]. The effectiveness of SMG has been validated by the previous researches, but most of them adopt the correction storage (CS) scheme. In the CS scheme, the variables are stored only on the finest grid level, while their corrections are stored and solved on the coarser grid levels; the updating of variables are realized by adding the corrections interpolated from coarse grid level. From a mathematical viewpoint, the CS scheme is only suitable for linear problems because the computation of corrections on coarser grid level relies on the linearity of the equation, so this scheme is not suitable for the NS equations if the convective effect is more important than the diffusive one. Moreover, there are multiple strongly-coupled variables and complex boundary conditions imposed when solving the NS equations in practical problems, the way of interpolating or extrapolating the variables and residuals between multi-level grids in a "high-order polynomial" way is still a problem to be further investigated.

In this paper, an explicit Runge–Kutta approach is used as the iterative solver for the incompressible NS equations. We propose the FAS scheme spectral multigrid method to accelerate the convergence. The FAS scheme multigrid allows for the nonlinearity of the equations by restricting both the variables and the residuals from fine to coarse grid level, while the coefficient matrix and the source term of the discretized equations are updated at the same time; it is the variables rather than the corrections that are computed iteratively on the coarser grid levels. For the interpolations of variables, residuals and corrections between multi-level grids, a procedure based on Fast Fourier Transform (FFT) is utilized. In order to demonstrate the efficiency of SMG, four iterative strategies are designed: the single grid method (SG); the full single grid method (FSG); the V-cycle spectral multigrid method (VMG) and the FMG method. Detailed comparisons in efficiency and accuracy of the four iterative strategies are carried out by three test cases: the modified one-dimensional Burgers equation, the modified two-dimensional Taylor vortices problem and lid driven cavity flow problem. The computational results are discussed and the convergence properties of the four iterative strategies are analyzed.

The paper is organized as follows. Section 2 presents the pseudospectral discretization method and the explicit solver. In Section 3, the spectral multigrid algorithm and its implementation to practical problems are discussed. In Section 4, the explicit solver and the SMG algorithm are performed on three test problems and the results are discussed. Some conclusions are in Section 5.

## 2. Governing equations and temporal-spatial discretization

### 2.1. Governing equations

We consider the laminar incompressible flow with consistent viscosity $v$ in the rectangular physical domain $\Omega = [-\xi,\xi] \times [-\eta,\eta]$ with the origin at the center of the domain. The fluid motion is governed by the nondimensional steady-state NS equations:

$$\begin{aligned} &\nabla \cdot \mathbf{u} = 0 \\ &(\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + Re^{-1}\nabla^2 \mathbf{u} \end{aligned} \tag{1}$$

By choosing the characteristic velocity $V_{ch}$ and length $d$, the velocity components $\mathbf{u} = (u, v)$ and pressure $p$ are nondimensionalized by $V_{ch}$ and $1/\rho V_{ch}^2$. $Re$ is the Reynolds number defined as $Re = V_{ch}d/v$. Since we are only interested in the steady-state solutions, the time derivative terms are removed from the momentum equations and the transient solutions are out of the scope of this paper.

### 2.2. Spatial discretization

The governing equations are discretized on Chebyshev–Gauss–Lobatto collocation points defined as [33]

$$\mathbf{x}_{ij} = \left[\xi \cdot \cos\left(\frac{\pi i}{N_x}\right), \eta \cdot \cos\left(\frac{\pi j}{N_y}\right)\right]$$

with $i = 0,\ldots,N_x$, $j = 0,\ldots,N_y$. $N_x$ and $N_y$ are the maximum indices of collocation points in the $x$- and $y$-direction, respectively. All the variables $(p,u,v)$ are stored and updated on these points.

The so-called $P_N - P_{N-2}$ method is adopted to interpolate the variables [1,4]. The velocities are approximated by truncated double Chebyshev polynomials of order $N_xN_y$ on the whole $(N_x+1) \times (N_y+1)$ points, while the pressure is approximated by a Lagrange polynomial of the order $(N_x-2)(N_y-2)$ on the inner $(N_x-1) \times (N_y-1)$ points:

$$\begin{aligned} \mathbf{u}(\mathbf{x}) &= \sum_{i=0}^{N_x}\sum_{j=0}^{N_y} \hat{\mathbf{u}}_{ij} T_j\left(\frac{y}{\eta}\right) T_i\left(\frac{x}{\xi}\right), \\ p(\mathbf{x}) &= \sum_{i=1}^{N_x-1}\sum_{j=1}^{N_y-1} \hat{p}_{ij} L_j\left(\frac{y}{\eta}\right) L_i\left(\frac{x}{\xi}\right). \end{aligned} \tag{2}$$

The advantages of the $P_N - P_{N-2}$ method lie in that there is no necessity to prescribe the pressure boundary conditions and it prevents the pressure from pollution by nonphysical spurious modes. The first and second order derivatives of the velocities in Eq. (1) are computed by differentiating equation (2) through the Chebyshev differentiation matrices [33,38]. The differentiation of a variable at any collocation point involves a weighted average of the same variable over all the points in the same row/column. Take the differentiation of $u$ along the $x$-direction as an example:

$$u'(x_i) = \frac{1}{\xi}\sum_{j=0}^{N_x} D_{i,j}^N \cdot u(x_j), \quad i = 0,\ldots,N_x, \tag{3}$$

$$u''(x_i) = \frac{1}{\xi^2}\sum_{j=0}^{N_x} (D_{i,j}^N)^2 \cdot u(x_j), \quad i = 0,\ldots,N_x, \tag{4}$$

$$\begin{aligned} D_{i,j}^N &= \frac{\bar{c}_i}{\bar{c}_j}\frac{(-1)^{i+j}}{(x_i - x_j)}, \quad 0 \leqslant i,j \leqslant N_x, \ i \neq j, \\ D_{i,i}^N &= -\frac{x_i}{2(1-x_i^2)}, \quad 1 \leqslant i \leqslant N_x - 1, \end{aligned}$$

$$D^N_{0,0} = -D^N_{N_x,N_x} = \frac{2N_x^2+1}{6},$$

$$\bar{c}_i = \begin{matrix} 2 & i = 0 \text{ or } N_x \\ 1 & \text{otherwise} \end{matrix}.$$

Similarly, the differentiation formula for pressure on the inner $N_x - 1$ collocation points is:

$$p'(x_i) = \frac{1}{\xi}\sum_{j=1}^{N_x-1} \widehat{D}^N_{i,j} p(x_j), \quad i = 1,\ldots,N_x-1,$$

$$\widehat{D}^N_{i,j} = \frac{(-1)^{j+i}(1-x_j^2)}{(1-x_i^2)(x_i-x_j)}, \quad i,j = 1,\ldots,N_x-1, \ i \neq j, \tag{5}$$

$$\widehat{D}^N_{i,i} = \frac{3x_i}{2(1-x_i^2)}, \quad i = 1,\ldots,N_x-1.$$

The computations in Eqs. (3)–(5) are actually vector–vector multiplications and can be carried out efficiently by various mathematical software packages. In fact, we calculate the derivatives in the same row/column at one time by performing a matrix–vector multiplication which is more efficient in practice.

### 2.3. Explicit Runge–Kutta time stepping method

In this paper, the pseudo unsteady method is applied to solve the discretized equations. The explicit solver requires the addition of a pseudo time derivative term to each of the equations:

$$\begin{aligned} &\frac{1}{\beta^2}\frac{\partial p}{\partial \tau} + \nabla\cdot\mathbf{u} = 0 \\ &\frac{\partial \mathbf{u}}{\partial \tau} + (\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p + Re^{-1}\nabla^2\mathbf{u}. \end{aligned} \tag{6}$$

$\tau$ denotes the pseudo time. The first term in the continuity equation is the artificial compressibility term that is used to couple the velocities and the pressure so that they would update simultaneously during the iteration process. The value of the artificial compressibility coefficient $\beta^2$ does not affect the final result after the computation converged [10]. Based on experience [29], we choose $\beta^2 = 5.0$ for all the computation cases in this paper.

The explicit four-stage Runge–Kutta method is used to integrate equation (6) in pseudo time as follows:

$$\begin{aligned} \phi^{(1)} &= \phi^n + \frac{1}{4}\Delta\tau R(\phi^n) \\ \phi^{(2)} &= \phi^n + \frac{1}{3}\Delta\tau R(\phi^{(1)}) \\ \phi^{(3)} &= \phi^n + \frac{1}{2}\Delta\tau R(\phi^{(2)}) \\ \phi^{n+1} &= \phi^n + \Delta\tau R(\phi^{(3)}) \end{aligned} \tag{7}$$

with the variables $\phi = (p,u,v)$. $\Delta\tau$ denotes the time step. The function $R$ of Eq. (7) stands for the residuals with

$$R = -\beta^2(\nabla\cdot\mathbf{u})$$

for the continuity equation and

$$R = -(\mathbf{u}\cdot\nabla)\mathbf{u} - \nabla p + Re^{-1}\nabla^2\mathbf{u}$$

for the momentum equations, Eq. (7) accomplishes the iteration and advances the variables from $\phi^n$ to $\phi^{n+1}$.

The explicit and semi-implicit time stepping algorithms for convective–diffusive flows are generally subject to time step limitation. Here the stability region of the iterative method is analyzed and the time step is given explicitly [29,36]:

$$\begin{aligned} \Delta t &= \frac{\mathrm{CFL}}{\lambda_x + \lambda_y} \\ \lambda_x &= \frac{|u_{\max}| + \sqrt{u_{\max}^2 + \beta^2}}{\Delta x} + \frac{1}{Re\cdot\Delta x^2} \\ \lambda_y &= \frac{|v_{\max}| + \sqrt{v_{\max}^2 + \beta^2}}{\Delta y} + \frac{1}{Re\cdot\Delta y^2}. \end{aligned} \tag{8}$$

CFL denotes the Courant–Friedrichs–Lewy number and its value is dependent on the flow problem to ensure convergence; $\Delta x$ and $\Delta y$ denote the minimum collocation point spacing in $x$- and $y$-directions, respectively; $u_{\max}$ and $v_{\max}$ are the maximum velocities. The first term in $\lambda_x$ and $\lambda_y$ represents the limit contributed by the convective part of NS equations, while the second term by the diffusive part. Due to the existence of artificial compressibility term and pseudo time derivative term, the transient solutions during the iteration process are physically meaningless. These terms will vanish as the iteration converges ($\partial p/\partial\tau$, $\partial u/\partial\tau$, $\partial v/\partial\tau \to 0$) and the solutions corresponding to Eq. (1) are obtained.

## 3. Spectral multigrid method

The multigrid method has been proved to be very efficient in accelerating convergence by FD/FV/FE methods. Its effectiveness is based on the fact that in conventional iterative method, rapid convergence could be obtained only during the first few iterations. In typical iterative procedures, the error-smoothing operators are generally very efficient in damping errors whose wavelengths are comparable to the mesh size, but are weak in removing errors with longer wavelengths. In multigrid method, a few iterations are first preformed on the fine grid level with mesh spacing $h$ and the errors with corresponding wavelengths could be smoothed out rapidly. Then the remaining computations are transferred to the coarser grid level with a mesh spacing $2h$ to smooth out the errors with wavelengths of about $2h$. Finally the variable corrections on this level are transferred back to update the fine grid solutions and this two-level iteration procedure is complete. By successively utilizing the similar procedures on multi-level grids of spacing $h, 2h, 4h, \ldots, 4h, 2h, h$, errors with wide range of wavelengths are uniformly damped out and the fine grid solution could be obtained with less computational time.

### 3.1. V-cycle FAS multigrid algorithm

The CS scheme multigrid method is inadequate for nonlinear problems since it depends on the linearity of the coefficient matrix and source term of the discretized equations for computing the corrections on the coarse grid. In order to remedy this deficiency, the FAS scheme was developed which interpolates both the unknown variables and the corrections between multi-level grids. For clarity, only the standard two-level V-cycle multigrid procedure is outlined below.

First, we denote the variable $Q_k$ on the fine grid $G_k$ satisfies Eq. (1):

$$A^k Q_k = b_k,$$

where $A^k$ and $b_k$ are, respectively, the coefficient matrix and source term derived from the discretization procedure. In fact, $Q_k$ is the solution we are trying to find out. Starting from an initial solution, after a few iterations on this grid, the approximation solution $\phi_k^{\mathrm{old}}$ is obtained:

$$A^k \phi_k^{\mathrm{old}} = b_k + r_k,$$

where $r_k$ is the residual. The approximation solution and the residual are then transferred to the next coarser grid $G_{k-1}$ by a restriction procedure to form the corresponding equation:

$$A^{k-1}\phi_{k-1}^{\text{old}} = b_{k-1} + R_k^{k-1}(r_k),$$

where $\phi_{k-1}^{\text{old}}$ is the initial solution restricted from $G_k$; $R_k^{k-1}$ is the restriction operator for the residual. The approximation solution $\phi_{k-1}^{\text{new}}$ can be obtained after a few time integrating steps on $G_{k-1}$. Finally the fine grid solution is updated via:

$$\phi_k^{\text{new}} = \phi_k^{\text{old}} + P_{k-1}^k(\phi_{k-1}^{\text{new}} - \phi_{k-1}^{\text{old}}),$$

which represents the prolongation of correction from coarse grid to fine grid with $P_{k-1}^k$ the prolongation operator. In this FAS scheme, the coefficient matrix and the source term on the coarse grid are updated using the newly-computed variables and the variables are calculated on every grid level, this feature preserves the nonlinear property of the discretized equation and helps to accelerate the convergence.

### 3.2. Full multigrid algorithm

The time cost of a computation is strongly related to the initial guess solution and the error smoothing rate. In the V-cycle multigrid method one usually starts the computation on the finest grid with zero as initial condition, but if a good initial guess is provided then fewer fine grid iterations will be required. This can be achieved by interpolating a coarse grid solution to the fine grid as the initial values:

$$(\phi_k)_{\text{initial}} = I_{k-1}^k(\phi_{k-1})_{\text{converged}}, \tag{9}$$

where $I_{k-1}^k$ is the coarse-to-fine interpolation operator for the variables. The above procedure in conjunction with V-cycle multigrid is termed the full multigrid strategy and a schematic illustration is given in Fig. 1. Starting with an initial guess on the coarsest grid level $G_1$, an accurate solution can be obtained economically by single grid iteration. This solution is then interpolated to the next finer grid $G_2$ through Eq. (9) and used as the initial guess of the two-level V-cycle multigrid iteration until it meets the convergence criterion on $G_2$. Then we interpolate the converged solution $\phi_2$ to the finest grid $G_3$ to initiate the three-level V-cycle multigrid iteration, this V-cycle multigrid iteration continues until the convergence is attained. The advantages of FMG are twofold: it provides a good initial guess solution to the fine grid computation at the expense of few computations on the coarse grid; the errors are smoothed out efficiently by the V-cycle multigrid iterations. These advantages guarantee the fast convergence rate of FMG and make it one of the most preferable acceleration techniques in CFD applications.

### 3.3. Restriction and prolongation

Various multigrid algorithms mainly differ in the restriction and prolongation operators. In two dimensional problem solved by FD/FV methods, the most commonly used restriction operators are the direct injection and the local averaging process, while the bilinear and biquadratic interpolations are often used as prolongation operators. However, these operators are not suitable for pseudospectral discretization scheme because global properties are required, while all of the above operators interpolate the variables or residuals just from the nearby grids. Here the interpolations based on high-order polynomials are introduced.

The restriction operator interpolates a function from the fine grid to the coarse grid. Due to the fact that the coarse-level Gauss–Lobatto collocation points are subsets of the collocation points on the next finer grid level, the restriction of variables is accomplished simply by direct injection, that is:

$$(\phi_{k-1}^{\text{old}})_{i,j} = (\phi_k)_{2i,2j}, \quad i = 0,\ldots,N_x^c,\ j = 0,\ldots,N_y^c,$$

where the superscript $c$ denotes the coarse grid level. For the interpolations of residuals and corrections, we adopt the row-column algorithm to decompose the two-dimensional interpolation into two one-dimensional interpolations. Take the restriction of residuals from $G_k$ to $G_{k-1}$ as an example, first the residuals are interpolated along the $x$-direction from $G_k$ to $G_{k-1}$ and an intermediate two-dimensional array is obtained. Then the intermediate array is interpolated along the $y$-direction to compute the residuals on $G_{k-1}$. In the $P_N - P_{N-2}$ method, the boundary values are already known for velocities and unnecessary for pressure, so the residuals and corrections on the boundary points are all set to zero. Given the residuals on the fine grid collocation points, we first compute the discrete Chebyshev coefficients by FFT [40,41]:

$$\widehat{R}_l = \frac{2}{N^f\overline{B}_l}\sum_{q=0}^{N^f}\frac{R_q^f}{\overline{B}_q}\cos\left(\frac{\pi lq}{N^f}\right) \quad l = 0,\ldots,N^f,$$

$$\overline{B}_l = \begin{cases} 2 & l = 0 \text{ or } N^f \\ 1 & 1 < l < N^f, \end{cases}$$

where the superscript $f$ denotes the fine grid level, $R_q^f$ are the residuals on the fine grid level. Afterwards we set the coefficients belonging to the high frequencies to zero and compute the residuals on the coarse grid collocation points:

$$R_m^c = \sum_{l=0}^{N^c}\widehat{R}_l\cos\left(\frac{\pi ml}{N^c}\right) \quad m = 0,\ldots,N^c,$$

The prolongation of corrections is more straightforward. Given the corrections $C_q^c$ on the coarse collocation points, the discrete Chebyshev coefficients are computed:

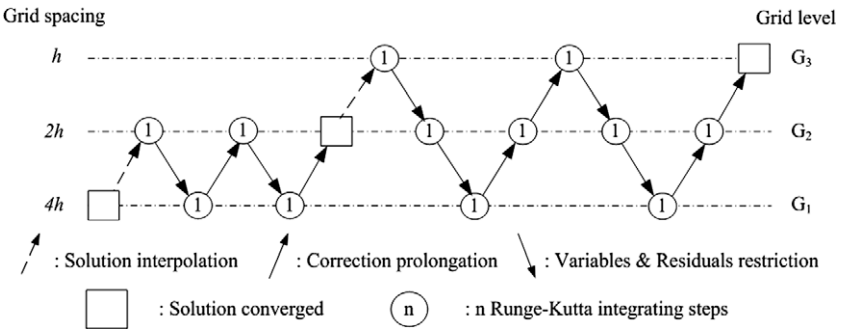


**Fig. 1.** A three-grid V-cycle FMG procedure.

$$\widehat{C}_l = \frac{2}{N^c \widetilde{B}_l} \sum_{q=0}^{N^c} \frac{C_q^c}{\widetilde{B}_q} \cos\left(\frac{\pi l q}{N^c}\right) \quad l = 0, \ldots, N^c, \tag{10}$$

$$\widetilde{B}_l = 2 \qquad l = 0 \text{ or } N^c$$
$$= 1 \qquad 1 < l < N^c$$

The fine grid approximation solution is then updated via:

$$\phi_j = \phi_j + \sum_{l=0}^{N^c} \widehat{C}_l \cos\left(\frac{\pi l j}{N^f}\right) \quad j = 0, \ldots, N^f. \tag{11}$$

For the FMG strategy, the variables have to be interpolated to the fine grid collocation points after the convergence on the coarse grid level. The way of interpolating the velocities is the same as the prolongation of corrections and can be achieved by means of Eqs. (10) and (11), but the interpolation of pressure cannot be realized in this way owing to the lack of boundary values. In this paper, we interpolate/extrapolate the pressure by Lagrange polynomial of order $N-2$; the coefficients of this polynomial are obtained from the $(N_x-1)\times(N_y-1)$ inner collocation points on the coarse grid level.

## 4. Numerical results and discussion

The solution methods described above were applied to three test problems. In all cases involving SMG, we took full coarsening for the generation of collocation points on the coarse grid levels, that is, $N^c = N^f/2$ in both directions. All the computations were carried out on a personal computer equipped with Intel 3.0-GHz E6850 processor, using the Intel Visual Fortran Compiler with default release settings under 32-bit Windows XP operating system.

The comparisons of computational complexity by various iterative strategies have been carried out with FV method [30], but so far there are no similar comparisons with spectral method. In this paper, the efficiency of SMG is validated by comparing the results and time costs of four carefully designed iterative strategies as mentioned in Section 1. The SG method performs the computation only on one single grid with zero field values as the initial condition. This version is represented as, for example, 48-SG, the number indicates the collocation points $N$ of the grid system. In two dimensional problems we apply $N_x = N_y = N$ for all cases in this paper. The FSG method performs single grid iterations on a series of grid levels; the initial condition on the coarsest grid is zero, while on all the other finer grid levels the initial condition is extrapolated from the converged solution on the next coarser grid. It is represented as 48-FSG-3; the number behind FSG indicates that there are three grid levels used in the computation. The VMG method starts on the finest grid level with zero values and employs FAS V-cycle multigrid to speed up convergence. It is represented as 48-VMG-123, the three-digit number behind VMG means that there are three grid levels involved in the computation, the number of time integrating steps for presmoothing (Runge–Kutta time integrating steps on the fine grid before the restriction to the coarse grid) on the finest grid is 1, while on the coarser and coarsest grid this number is 2 and 3, respectively. The last one is the FMG method as discussed above and represented as 48-FMG-123, for example. By comparing the time costs of the above four versions of iterative strategies, we can assess how much the initial condition improvement contributes to the acceleration and how much the contribution of SMG is. Based on our experiences, postsmoothing (Runge–Kutta time integrating steps on the finer grid after the prolongation of corrections from coarser grid) is unnecessary since in most cases it brings no improvements but sometimes brings instabilities to the computation.

The convergence is obtained if the root-mean-square of the continuity equation residuals on all the inner collocation points is under a predefined small value, that is:

$$E_{\mathrm{RMS}} = \sqrt{\frac{\sum_{i=1}^{N_x-1}\sum_{j=1}^{N_y-1}[(u_x+v_y)_{i,j}]^2}{(N_x-1)(N_y-1)}} < \varepsilon.$$

$\varepsilon$ is the small number used as the convergence criterion and will be discussed below. In SG and VMG the solution is considered to be converged if the $E_{\mathrm{RMS}}$ meets the convergence criterion ($E_{\mathrm{RMS}}$ is computed only on the finest grid level for VMG), while in FSG and FMG we choose different $\varepsilon$ as the 0 criterion for the variables to interpolate from coarse grid level to fine grid level.

It has to be emphasized here that in this paper, we will present the speedup properties of SMG in the manner of wall clock time rather than equivalent work unit because it is difficult to determine the number of floating point operations in practical problems by the latter method. The conventional viewpoint that the coarse grid calculation is one-fourth of the work on the finer grid is inappropriate in pseudospectral method, especially in VMG and FMG where the restriction and prolongation are realized by FFT. The estimation based on the wall clock time includes all auxiliary efforts in the computation but has the disadvantage of depending on hardware and the experiences of the programmer. However, the influence of these disadvantages are of minor importance since FSG, VMG and FMG versions are all modified from the original SG version, so the major structures and arithmetic operations are identical in the above four versions of programs. In this paper we use the relative time cost ratios to demonstrate the speedup of the SMG methods.

### 4.1. Validation using modified 1D Burgers equation

First, we consider a modified steady-state 1D Burgers equation with exact solution to validate the reliability and accuracy of the explicit solver. The equation can be described as

$$u\frac{du}{dx} = \frac{1}{Re}\frac{d^2u}{dx^2} + \left(e^{2x} - \frac{1}{Re}e^x\right), \quad x \in [-0.5, 0.5],$$
$$u(-0.5) = e^{-0.5}, \quad u(0.5) = e^{0.5}.$$

This modified Burgers equation has the exact solution $u = e^x$; the Reynolds number is fixed at 100; the source term on the right hand side of the equation is created to balance the convection–diffusion terms. The computations are implemented on a series of grids with $N = 4$–96 by the SG method. In order to test the influence of convergence criterion on the accuracy of the solution, different values of $\varepsilon$ from $10^{-4}$ to $10^{-10}$ are used. We maintain the CFL a fixed number (CFL = 2.5) around the upper bound of the stability region of the four-step Runge–Kutta method for the calculations with large $N$, but with small $N$ ($N = 4$–10) the iteration could not be converged unless smaller CFL (CFL = 0.3–0.5) is used. However, since we are aiming to test the accuracy of the explicit method in this problem, the time cost variations induced by different CFL numbers are ignored.

To take advantage of the global nature of pseudospectral method, the computational result is first interpolated to an $N = 201$ equispaced grid (including the endpoints). The maximum error is defined as the absolute value of the maximum difference between the interpolated result and the exact solution. To illustrate the accuracy of the explicit method and the impact of the convergence criterion, we plot the maximum error against the number of collocation points $N$ under various convergence criterions. Fig. 2 shows that the error is strongly related to the convergence criterion for this problem; $\varepsilon = 10^{-10}$ seems to be enough and 10-digit accuracy can be obtained with large $N$ in this problem. Since the exact solution is smooth over the entire computational domain, the spectral convergence is obvious with $N < 10$ and the error does not change much as $N$ increases, especially in the cases with strict convergence

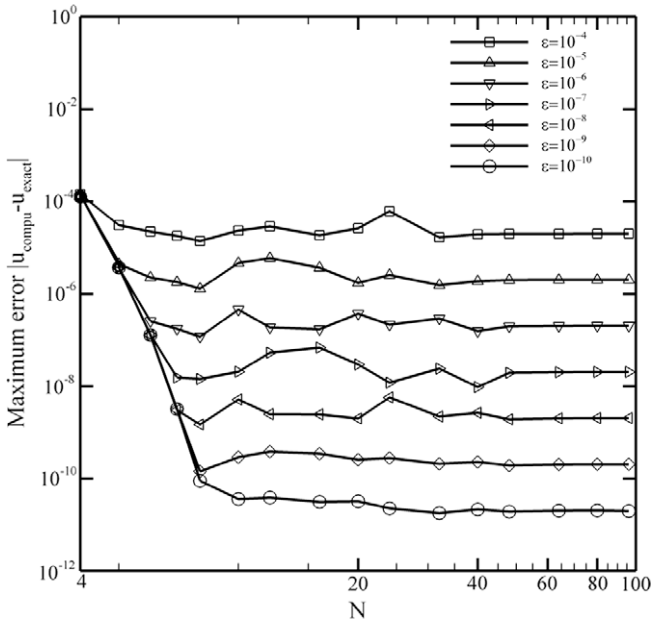


**Fig. 2.** Comparison of maximum error vs. collocation points subject to different convergence criterion for the modified 1D Burgers equation.

criterions. It is emphasized here that for this and the next Taylor vortices problem the convergence criterion for all calculations is fixed at $10^{-10}$, while for the lid driven cavity flow problem we took $\varepsilon = 10^{-4}$ since it can be seen that converged results are obtained at this convergence criterion.

### 4.2. Modified 2D Taylor vortices

In this part we present a verification of 2D incompressible NS equations by the multigrid solver. The test problem is known as the *Taylor vortices* [16]. We artificially reconstruct the original problem to accommodate the steady-state NS equations given as

$$\begin{aligned}
&\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0\\
&u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=-\frac{\partial p}{\partial x}+\frac{1}{Re}\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)+\phi_x\\
&u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}=-\frac{\partial p}{\partial y}+\frac{1}{Re}\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)+\phi_y\\
&\phi_x=-\frac{2}{Re}\cos(x)\cdot\sin(y),\quad \phi_y=\frac{2}{Re}\sin(x)\cdot\cos(y).
\end{aligned} \tag{12}$$

These equations are defined in $[-0.5,0.5]^2$ with the corresponding exact solution

$$\begin{aligned}
&p=-(\cos(2x)+\cos(2y))/4\\
&u=-\cos(x)\cdot\sin(y)\\
&v=\sin(x)\cdot\cos(y).
\end{aligned} \tag{13}$$

The initial conditions are zero for all three variables on the inner collocation points and the source terms in Eq. (12) are designed to balance the equations under boundary conditions specified by Eq. (13). The Reynolds number is 100 and the CFL is 2.5 for all cases to facilitate the comparisons. In order to evaluate the accuracy of the SMG solver, the computational result is interpolated to a $201 \times 201$ equispaced grid on which the error is calculated.

To illustrate the accuracy and speedup property of the SMG method, three versions of iterative methods are presented: the SG method; the two-level and three-level VMG method. Fig. 3 demonstrates the maximum error of velocity $u$ against $N$ for the three methods. It can be seen that they have approximately the same orders of accuracy as $N$ increases. In Fig. 4 the comparisons of wall clock time are presented. In most cases the two-level VMG could decrease the time cost by about 50%, while the three-level VMG could achieve a time reduction of almost 90%. Moreover, the time cost increases much slower for VMG than that of SG as $N$ increases; this is obvious by estimating the slopes of the curves in Fig. 4. If we increases the number of collocation points continuously, the time cost by SG would increase exponentially and finally be intolerable, in which situation the SMG method becomes a necessity for solving the problem.

It is mentioned here that in the beginning period for developing this code, we applied the local time stepping technique [14,18,27,36] to calculate the time step. That is, the velocities used in Eq. (8) are local values (velocities at each collocation point) rather than the maximum values, so the time step differs from point to point and the convergence on some points are much faster than others. This acceleration technique is widely used in FD/FV codes and commercial CFD software such as NUMECA, but we found that it brought only a small time cost reduction (less than 3%) in this test case based on pseudospectral method. We believe that the local time stepping in pseudospectral method is a problem dependent acceleration technique and is not so efficient in this case, but it is still worth trying when solving other problems.

### 4.3. Lid driven cavity flow problem

The lid driven cavity flow is a popular and classic problem in CFD since the pioneer work done by Ghia et al. [15]. It has been numerically studied in detail either by FD [8,39] or FV [12] method. In this paper we will solve this problem by pseudospectral method.

The well-known difficulty of the cavity problem is the corner singularities. At the two upper corners where the moving wall meets the stationary wall, the vorticity becomes unbounded and the horizontal velocity is multivalued. Since the accuracy of the pseudospectral method is generally associated to the smoothness of the solution, the errors caused by the singularities will spread over the entire computational domain in the form of Gibbs

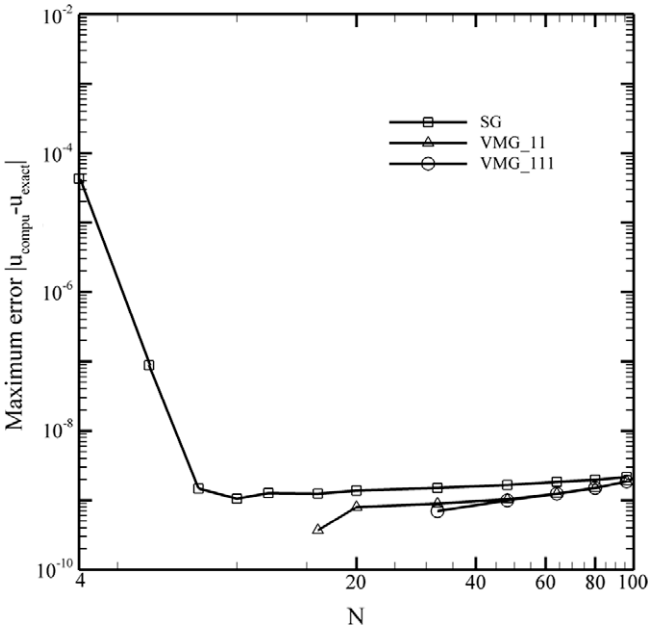


**Fig. 3.** Comparison of maximum error of velocity $u$ vs. collocation points for the modified 2D Taylor vortices problem.

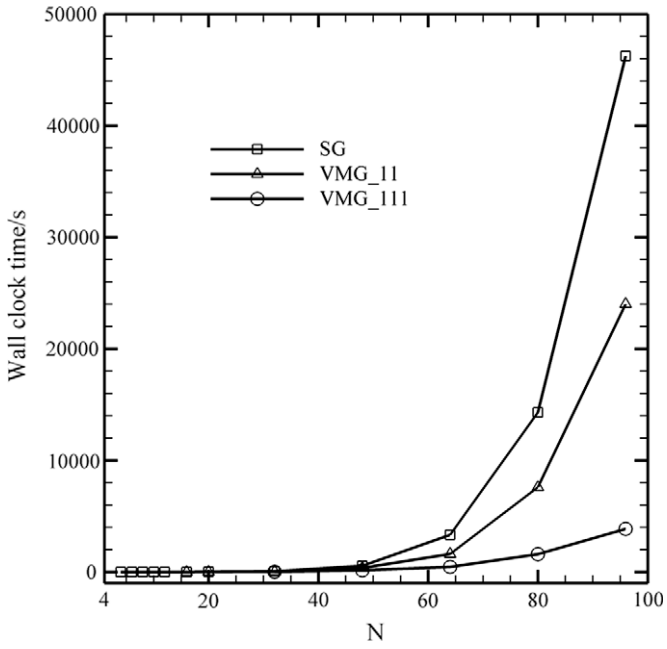


**Fig. 4.** Comparison of wall clock time with respect to collocation points for the modified 2D Taylor vortices problem.

oscillations [33]. In previous studies three kinds of remedies have been employed to deal with this difficulty. The first remedy is the so-called regularized cavity problem in which the speed of the moving wall is defined as $V_W = 16(x-0.5)^2(x+0.5)^2$ [35]. In the second manner the velocity near the upper corners is adjusted over the first few collocation points [26]. For example, $V_W = 0$ at the corner point, $V_W = 0.3$ (or 0.5) at the next point and $V_W = 1.0$ the third one. The above two methods smooth the sharp velocity discontinuities at the corners and proved to be efficient, but they are approximations to the standard cavity problem. In this paper, we adopt the third way, the subtraction method to deal with the singularities [1,4,37]. The variables in Eq. (1) are first divided into two categories:

$$\mathbf{u} = \mathbf{u}_c + \mathbf{u}_s, \quad p = p_c + p_s,$$

with $\mathbf{u}_s = \mathbf{u}_s^A + \mathbf{u}_s^B$ and $p_s = p_s^A + p_s^B$. The subscripts $c$ and $s$ denote the computational part and the singular part of the solution, and the superscripts A and B denote the two upper corners.

The singular solution near either of the two corners can be determined by introducing the steady-state streamfunction $\psi$ in a local polar coordinate system with the corner as the origin. The streamfunction equation is:

$$-\nabla^4\psi + \mathrm{Re}\cdot\mathbf{u}\cdot\nabla(\nabla^2\psi) = 0.$$

According to Hancock et al. [20], the solution of the streamfunction equation can be expressed in the form of a regular perturbation expansion:

$$\psi = \nu\sum_{n=1}^{\infty}\left(\frac{ru}{\nu}\right)^n f_n(\theta). \tag{14}$$

In this paper, only the first term $n = 1$ is taken into consideration. By expanding Eq. (14) under certain boundary conditions, the singular part solution can be explicitly expressed. More details of this procedure and the expressions of the singular solution $(\mathbf{u}_s, p_s)$ can be found in [4,19,20].

After subtracting the singular part from the primitive variables, the computational part $(\mathbf{u}_c, p_c)$ is smooth enough to be discretized by Chebyshev pseudospectral method and the pollution brought by the singularities is minimized. The modified NS equations to be solved are:

$$\nabla\cdot\mathbf{u}_c = 0,$$
$$\mathbf{u}_c\cdot\nabla\mathbf{u}_c + \mathbf{u}_s\cdot\nabla\mathbf{u}_c + \mathbf{u}_c\cdot\nabla\mathbf{u}_s + \mathbf{u}_s\cdot\nabla\mathbf{u}_s = -\nabla p_c + Re^{-1}\nabla^2\mathbf{u}_c.$$

The corresponding boundary conditions are $\mathbf{u}_c = (V_W - u_s, -v_s)$ on the side $y = 1/2$ and $\mathbf{u}_c = -\mathbf{u}_s$ on all the other three sides. The pressure boundary condition is unnecessary.

#### 4.3.1. Flow at Re = 100

First, the accuracy of the pseudospectral method with different iterative strategies is checked. Considering the global feature of spectral method, all the results presented in the following tables are obtained by interpolating the computational results $(\mathbf{u}_c, p_c)$ on an equispaced grid of mesh size $10^{-4}$ and add them to the singular part $(\mathbf{u}_s, p_s)$ to get the final results. Table 1 shows the extreme velocities along the centerlines and their corresponding positions. The minimum horizontal velocity on the vertical centerline is denoted $u_{\min}$ and its location $y_{\min}$. The maximum and minimum vertical velocities on the horizontal centerline are denoted $v_{\max}$ and $v_{\min}$, with their locations $x_{\max}$ and $x_{\min}$, respectively. It can be observed that the accuracy of these results mainly depends on the number of collocation points; furthermore, our results are in good agreement with those found in literature, especially with Botella and Peyret [4] who performed similar discretization scheme with the singular solution taken as the first two terms in Eq. (14), while in this paper only the first term is considered. The locations of the extreme velocities agree well with those in [4] in that a 4-digit accuracy agreement is achieved.

Then, the distribution of horizontal velocity $u$ along the vertical centerline of the cavity is given in Table 2. The sampling points are selected in accordance with Ghia et al. [15]. Good agreement can be found between present results and those from [15] with at least two digits accuracy agreement. However, due to the singularities at

**Table 1**
Extreme of velocity through the centerline of the cavity, $Re = 100$.

| Source | Method and grid | $u_{\min}$ | $y_{\min}$ | $v_{\max}$ | $x_{\max}$ | $v_{\min}$ | $x_{\min}$ |
|---|---|---|---|---|---|---|---|
| Bruneau et al. [8] | FD, 129 × 129 | −0.2106 | −0.0469 | 0.1786 | −0.2656 | −0.2521 | 0.3125 |
| Ghia et al. [15] | FD, 129 × 129 | −0.21090 | −0.0469 | 0.17527 | −0.2656 | −0.24533 | 0.3047 |
| Deng et al. [12] | FV, Richardson extrapolation | −0.21405 | – | 0.17896 | – | −0.25399 | – |
| Botella and Peyret [4] | Pseudospectral, $N = 96$ | −0.2140424 | −0.0419 | 0.1795728 | −0.2630 | −0.2538030 | 0.3104 |
| Present | 48-VMG-111 | −0.2140495 | −0.0419 | 0.1795796 | −0.2630 | −0.2538177 | 0.3104 |
| Present | 48-FMG-111 | −0.2140482 | −0.0419 | 0.1795763 | −0.2630 | −0.2538134 | 0.3104 |
| Present | 48-FMG-123 | −0.2140481 | −0.0419 | 0.1795758 | −0.2630 | −0.2538127 | 0.3104 |
| Present | 64-VMG-111 | −0.2140442 | −0.0419 | 0.1795766 | −0.2630 | −0.2538057 | 0.3104 |
| Present | 64-FMG-111 | −0.2140444 | −0.0419 | 0.1795777 | −0.2630 | −0.2538115 | 0.3104 |
| Present | 96-VMG-111 | −0.2140428 | −0.0419 | 0.1795833 | −0.2630 | −0.2538057 | 0.3104 |
| Present | 96-FMG-111 | −0.2140426 | −0.0419 | 0.1795717 | −0.2630 | −0.2538003 | 0.3104 |
| Present | 96-FMG-1111 | −0.2140427 | −0.0419 | 0.1795718 | −0.2630 | −0.2538003 | 0.3104 |

**Table 2**
Horizontal velocity $u$ along vertical centerline of the cavity, $Re = 100$.

| $y$ position | −0.5000 | −0.4453 | −0.4375 | −0.4297 | −0.3984 | −0.3281 | −0.2187 | −0.0469 | 0.0000 |
|---|---|---|---|---|---|---|---|---|---|
| Ghia et al. [15] | 0.00000 | −0.03717 | −0.04192 | −0.04775 | −0.06434 | −0.10150 | −0.15662 | −0.21090 | −0.20581 |
| 48-FMG-111 | 0.0000000 | −0.0372280 | −0.0419755 | −0.0466203 | −0.0644321 | −0.1017433 | −0.1576766 | −0.2139834 | −0.2091543 |
| 64-FMG-111 | 0.0000000 | −0.0372278 | −0.0419751 | −0.0466199 | −0.0644315 | −0.1017424 | −0.1576745 | −0.2139795 | −0.2091515 |
| 96-VMG-111 | 0.0000000 | −0.0372269 | −0.0419743 | −0.0466192 | −0.0644312 | −0.1017433 | −0.1576762 | −0.2139779 | −0.2091480 |
| 96-FMG-111 | 0.0000000 | −0.0372276 | −0.0419749 | −0.0466198 | −0.0644313 | −0.1017420 | −0.1576735 | −0.2139776 | −0.2091497 |
| | 0.1172 | 0.2344 | 0.3516 | 0.4531 | 0.4609 | 0.4688 | 0.4766 | 0.5000 | |
| Ghia et al. [15] | −0.13641 | 0.00332 | 0.23151 | 0.68717 | 0.73722 | 0.78871 | 0.84123 | 1.00000 | |
| 48-FMG-111 | −0.1387991 | 0.0041891 | 0.2365530 | 0.6910334 | 0.7404783 | 0.7919530 | 0.8437707 | 1.0000000 | |
| 64-FMG-111 | −0.1387983 | 0.0041873 | 0.2365516 | 0.6910275 | 0.7404705 | 0.7919415 | 0.8437433 | 1.0000000 | |
| 96-VMG-111 | −0.1387912 | 0.0041969 | 0.2365582 | 0.6910286 | 0.7404711 | 0.7919408 | 0.8437356 | 1.0000000 | |
| 96-FMG-111 | −0.1387979 | 0.0041870 | 0.2365516 | 0.6910272 | 0.7404694 | 0.7919388 | 0.8437334 | 1.0000000 | |

**Table 3**
Wall clock time by various versions of iterative strategies, $Re = 100$.

| Method | Wall clock time/s | Method | Wall clock time/s | Method | Wall clock time/s |
|---|---|---|---|---|---|
| 48-SG | 2447.86 | 64-SG | 12,364.77 | 96-SG | 141,679.63 |
| 48-FSG-3 | 230.02 | 64-FSG-3 | 1094.97 | 96-FSG-3 | 7730.98 |
| 48-VMG-111 | 92.70 | 64-VMG-11 | 2335.09 | 96-VMG-111 | 2131.11 |
| 48-FMG-111 | 86.25 | 64-VMG-111 | 288.17 | 96-FMG-11 | 7599.78 |
| 48-FMG-123 | 94.03 | 64-FMG-11 | 766.06 | 96-FMG-111 | 1742.41 |
| | | 64-FMG-111 | 251.25 | 96-FMG-123 | 1845.63 |
| | | 64-FMG-123 | 276.02 | 96-FMG-1111 | 1651.09 |

the two endpoints of the lid, the accuracy of the results varies along with the sampling point position.

Table 3 shows the time cost results. First, in general, the SG and FMG are the most time-consuming and time efficient methods, respectively; while the FSG and VMG provide different degrees of time reductions relative to the SG method. In some cases, the three-level FSG is more time efficient than the two-level VMG method (64-FSG-3 vs. 64-VMG-11), or nearly the same efficient with the two-level FMG version (96-FSG-3 vs. 96-FMG-11). These findings reflect the relative importance of initial condition. Second, the three-level SMG achieves an outstanding time cost reduction compared with the two-level SMG and FSG methods. The three-level VMG offers a speedup about 4–5 compared with the three-level FSG. Furthermore, the time cost increase of VMG is much slower than SG and FSG as $N$ increases. Third, we also found that more presmoothing steps on coarser grids are of helpless in reducing the time cost such as $T_{48\text{-FMG-}111} < T_{48\text{-FMG-}123}$, which is different from the previous experiences with FD/FV method. In our opinion, this may be due to the global feature of spectral method, more fine grid iterations are required to smooth out the errors brought by the correction prolongation if more presmoothing steps on coarser grid are used. Moreover, three-level multigrid seems to be enough to deal with this problem, the four-level FMG version 96-FMG-1111 reduces only a little time cost (about 5%) compared with its three-level counterpart 96-FMG-111.

#### 4.3.2. Flow at Re = 400 and 1000

Detailed results on the cavity flow at $Re = 400$ are listed through Tables 4–6. The advantage of high accuracy with larger $N$ is more obvious when $Re$ is increasing. The locations of extreme

**Table 4**
Extreme of velocity through the centerline of the cavity, $Re = 400$.

| Source | Method and grid | $u_{min}$ | $y_{min}$ | $v_{max}$ | $x_{max}$ | $v_{min}$ | $x_{min}$ |
|---|---|---|---|---|---|---|---|
| Ghia et al. [15] | FD, 129 × 129 | −0.32726 | −0.2187 | 0.30203 | −0.2734 | −0.44993 | 0.3594 |
| Deng et al. [12] | FV, Richardson extrapolation | −0.32873 | – | 0.30379 | – | −0.45431 | – |
| Present | 48-FMG-11 | −0.3287575 | −0.2201 | 0.3038908 | −0.2748 | −0.4541622 | 0.3622 |
| Present | 64-FMG-11 | −0.3287118 | −0.2200 | 0.3038539 | −0.2747 | −0.4540923 | 0.3622 |
| Present | 96-FMG-111 | −0.3287301 | −0.2200 | 0.3038316 | −0.2747 | −0.4540731 | 0.3622 |

**Table 5**
Horizontal velocity $u$ along vertical centerline of the cavity, $Re = 400$.

| $y$ position | −0.5000 | −0.4453 | −0.4375 | −0.4297 | −0.3984 | −0.3281 | −0.2187 | −0.0469 | 0.0000 |
|---|---|---|---|---|---|---|---|---|---|
| Ghia et al. [15] | 0.00000 | −0.08186 | −0.09266 | −0.10338 | −0.14612 | −0.24299 | −0.32726 | −0.17119 | −0.11477 |
| 48-FMG-11 | 0.0000000 | −0.0819433 | −0.0927277 | −0.1034364 | −0.1463531 | −0.2439601 | −0.3287412 | −0.1713655 | −0.1149543 |
| 64-FMG-11 | 0.0000000 | −0.0818447 | −0.0926244 | −0.1033286 | −0.1462293 | −0.2438233 | −0.3286973 | −0.1714010 | −0.1149783 |
| 96-FMG-111 | 0.0000000 | −0.0818232 | −0.0926005 | −0.1033020 | −0.1461924 | −0.2437812 | −0.3287166 | −0.1714811 | −0.1150525 |
| | 0.1172 | 0.2344 | 0.3516 | 0.4531 | 0.4609 | 0.4688 | 0.4766 | 0.5000 | |
| Ghia et al. [15] | 0.02135 | 0.16256 | 0.29093 | 0.55892 | 0.61756 | 0.68439 | 0.75837 | 1.00000 | |
| 48-FMG-11 | 0.0210719 | 0.1625552 | 0.2920260 | 0.5615752 | 0.6199052 | 0.6871836 | 0.7606830 | 1.0000000 | |
| 64-FMG-11 | 0.0210808 | 0.1625801 | 0.2920358 | 0.5615848 | 0.6199286 | 0.6872147 | 0.7606770 | 1.0000000 | |
| 96-FMG-111 | 0.0210340 | 0.1625663 | 0.2920315 | 0.5615645 | 0.6199137 | 0.6872048 | 0.7606698 | 1.0000000 | |

**Table 6**
Wall clock time by various versions of iterative strategies, $Re = 400$.

| Method | Wall clock time/s | Method | Wall clock time/s | Method | Wall clock time/s |
|---|---|---|---|---|---|
| 48-SG | 2623.75 | 64-SG | 11,280.45 | 96-SG | 111,151.25 |
| 48-FSG-2 | 458.83 | 64-FSG-3 | 1059.33 | 96-FSG-3 | 3485.53 |
| 48-VMG-11 | 864.61 | 64-VMG-11 | 3157.11 | 96-VMG-111 | 3398.89 |
| 48-FMG-11 | 245.19 | 64-FMG-11 | 774.61 | 96-FMG-111 | 2008.64 |

**Table 7**
Extreme of velocity through the centerline of the cavity, $Re = 1000$.

| Source | Method and grid | $u_{min}$ | $y_{min}$ | $v_{max}$ | $x_{max}$ | $v_{min}$ | $x_{min}$ |
|---|---|---|---|---|---|---|---|
| Ghia et al. [15] | FD, $129 \times 129$ | −0.38289 | −0.3281 | 0.37095 | −0.3437 | −0.51550 | 0.4063 |
| Deng et al. [12] | FV, Richardson extrapolation | −0.38867 | – | 0.37072 | – | −0.52724 | – |
| Botella and Peyret [4] | Pseudospectral, $N = 96$ | −0.3885698 | −0.3283 | 0.3769447 | −0.3422 | −0.5270771 | 0.4092 |
| Present | 64-FMG-11 | −0.3889995 | −0.3283 | 0.3774515 | −0.3422 | −0.5275155 | 0.4092 |
| Present | 96-FMG-11 | −0.3886040 | −0.3283 | 0.3769905 | −0.3422 | −0.5270888 | 0.4092 |

velocities can be estimated accurately with $N = 64$ and 96, while there is a minor discrepancy on the results of $N = 48$. Similar conclusions can be drawn from the velocity profile results in Table 5.

In Table 6 the time costs by various methods are listed. The time cost ratios of two-level SMG at $Re = 400$ are $T_{64\text{-VMG-}11}/T_{64\text{-SG}} = 28.0\%$ and $T_{64\text{-FMG-}11}/T_{64\text{-SG}} = 6.9\%$, respectively, while their counterparts at $Re = 100$ are $T_{64\text{-VMG-}11}/T_{64\text{-SG}} = 18.9\%$ and $T_{64\text{-FMG-}11}/T_{64\text{-SG}} = 6.2\%$. The speedup ratios of VMG and FMG do not change much as $Re$ increases. As we expected, the most time efficient computation on the $N = 96$ grid is realized by FMG method with the time cost ratio $T_{96\text{-FMG-}111}/T_{96\text{-SG}} = 1.8\%$, which is a little higher than 1.2% at $Re = 100$. In all computations at $Re = 400$, the FMG is always the most efficient method, especially when comparing with the VMG version. In our opinion, the notable advantage of FMG at $Re = 400$ is partly the result of the increasing influence of initial condition. This influence is confirmed by comparing $T_{48\text{-FSG-}2}$ and $T_{48\text{-VMG-}11}$, the FSG version costs about half time that of the VMG version.

Tables 7–9 show the results at $Re = 1000$. The extreme velocities and their locations agree well with the benchmark solutions found in literature, especially those in [4]. Similar agreement exists for the velocity profiles. As $Re$ increases from 400 to 1000, the initial condition has a greater influence on the convergence rate, the two-level FSG version converges faster than the two-level VMG version. Again, the FMG is the fastest method of the four since the two-level FMG is almost the same efficient with the three-level FSG version.

Some interesting conclusions could be drawn when we perform comparisons of the wall clock time results obtained at different Reynolds numbers. Traditionally it is believed that the problem is more difficult to converge when $Re$ increases because of the increasing nonlinear effects, but this viewpoint is not always true in the present research. The SG method costs more time at $Re = 1000$ with $N = 48$ $\left(T_{48-SG}^{Re=1000} > T_{48-SG}^{Re=400} > T_{48-SG}^{Re=100}\right)$, but the opposite trend is true with $N = 96$ $\left(T_{96-SG}^{Re=1000} < T_{96-SG}^{Re=400} < T_{96-SG}^{Re=100}\right)$. Similar conclusions were made for three-dimensional cavity flow problem by FV method [30]. In our opinion, this phenomenon is possibly due to the influences of initial conditions and the number of collocation points. First, the initial conditions are zero for all the variables and in all the computations, but actually they would have different importance in different cases. Second, the spatial resolution is improved as $N$ increases and more flow details are revealed. It seems that more iterations are needed for $Re = 100$ to meet the same convergence criterion. It is the comprehensive effect of the initial condition, the nonlinearity, the number of collocation point and the iteration strategy that makes the complex variation trend of time cost.

Finally, some analysis is carried out for the convergence histories. Fig. 5 shows the convergence histories of the four iterative

**Table 9**
Wall clock time by various versions of iterative strategies, $Re = 1000$.

| Method | Wall clock time/s | Method | Wall clock time/s | Method | Wall clock time/s |
|---|---|---|---|---|---|
| 48-SG | 3248.08 | 64-SG | 12,173.67 | 96-SG | 10,0255.61 |
| 48-FSG-2 | 1073.78 | 64-FSG-2 | 1696.47 | 96-FSG-3 | 5624.63 |
| | | 64-VMG-11 | 4285.03 | 96-VMG-11 | 35,238.39 |
| | | 64-FMG-11 | 1181.66 | 96-FMG-11 | 6073.50 |

**Table 8**
Horizontal velocity $u$ along vertical centerline of the cavity, $Re = 1000$.

| $y$ position | −0.5000 | −0.4453 | −0.4375 | −0.4297 | −0.3984 | −0.3281 | −0.2187 | −0.0469 | 0.0000 |
|---|---|---|---|---|---|---|---|---|---|
| Ghia et al. [15] | 0.00000 | −0.18109 | −0.20196 | −0.22220 | −0.29730 | −0.38289 | −0.27805 | −0.10648 | −0.06080 |
| Botella and Peyret [4] | 0.0000000 | −0.1812881 | −0.2023300 | −0.2228955 | −0.3004561 | −0.3885691 | −0.2803696 | −0.1081999 | −0.0620561 |
| 64-FMG-11 | 0.0000000 | −0.1814380 | −0.2025066 | −0.2230993 | −0.3007705 | −0.3889989 | −0.2809581 | −0.10855939 | −0.0622351 |
| 96-FMG-11 | 0.0000000 | −0.1813444 | −0.2023891 | −0.2229565 | −0.3005181 | −0.3886031 | −0.2803835 | −0.10822142 | −0.0620833 |
| | 0.1172 | 0.2344 | 0.3516 | 0.4531 | 0.4609 | 0.4688 | 0.4766 | 0.5000 | |
| Ghia et al. [15] | 0.05702 | 0.18719 | 0.33304 | 0.46604 | 0.51117 | 0.57492 | 0.65928 | 1.00000 | |
| Botella and Peyret [4] | 0.0570178 | 0.1886747 | 0.3372212 | 0.4723329 | 0.5169277 | 0.5808359 | 0.6644227 | 1.0000000 | |
| 64-FMG-11 | 0.0573151 | 0.1893086 | 0.3379464 | 0.4728319 | 0.5173501 | 0.5812020 | 0.6647214 | 1.0000000 | |
| 96-FMG-11 | 0.0569866 | 0.1886758 | 0.3372538 | 0.4723622 | 0.5169523 | 0.5808571 | 0.6644381 | 1.0000000 | |

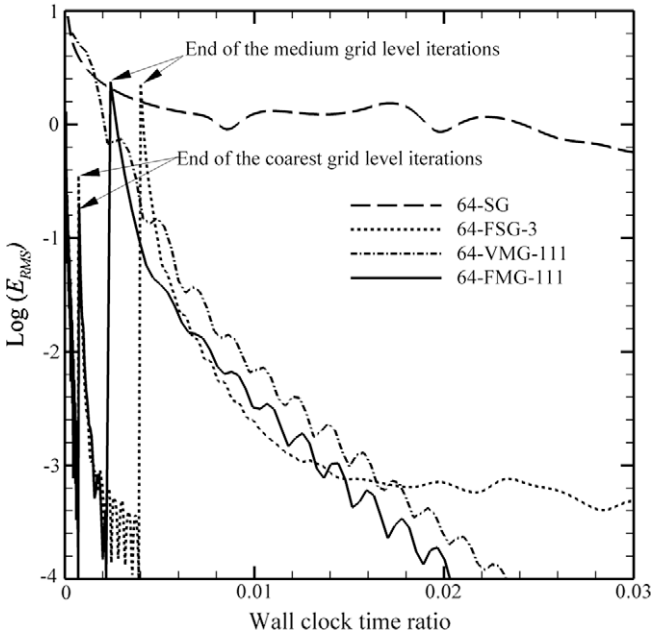


**Fig. 5.** Comparison of convergence history vs. wall clock time ratio ($T/T_{SG,\ converged}$) at $Re = 100$, $N = 64$.

methods at $Re = 100$. It is obvious that the FSG profits from the good initial conditions and converge much faster than the SG method at first, but the convergence rate is approximately the same after the initial period. The VMG and FMG have about the same convergence rate on the finest grid level, too, but the FMG version starts with a lower residual level and converges faster than VMG in the initial period, which makes the slight difference of time costs after they converge.

## 5. Conclusions

In this paper, the pseudospectral multigrid method for incompressible flow problems was investigated. By adding an artificial time-derivative term, the explicit Runge–Kutta method was used to time-march the NS equations to obtain the steady solutions. Four multigrid strategies were designed to investigate their speed-up properties and computation accuracy. Three typical cases were numerically calculated by the proposed methods. The numerical results lead to the following conclusions:

(1) The explicit Runge–Kutta time-stepping method is proved reliable to obtain high accuracy results with pseudospectral method in the three test cases. Moreover, the time cost can be greatly reduced if it is applied in conjunction with a spectral multigrid method.
(2) The SG is the most time-consuming method of the four iterative strategies and the FMG is the most time efficient one, while the FSG and VMG methods offer different degrees of time reductions depending on the problem. In general, the multigrid method could decrease the time cost by at least 50%, in particular, a time cost reduction of at most two orders of magnitude is achieved by the three-level FMG method for the lid driven cavity problem.
(3) For the lid driven cavity flow problem, high accuracy results are obtained and they are in good agreement with the benchmark solutions. For low *Re* cases, results with 5- to 6-digit accuracy can be easily achieved; while for high *Re* case, at least 3-digit accuracy results can be obtained.

## Acknowledgements

The authors wish to acknowledge the financial support provided by National Natural Science foundation of China (Nos. 50725621 and 10572113).

## References

[1] Albensoeder S, Kuhlmann HC. Accurate three-dimensional lid-driven cavity flow. J Comput Phys 2005;206:536–58.
[2] Borzi A, Hohenester U. Multigrid optimization schemes for solving Bose–Einstein condensate control problems. SIAM J Sci Comput 2007;30:441–62.
[3] Botella O. On the solution of the Navier–Stokes equations using Chebyshev projection schemes with third-order accuracy in time. Comput Fluids 1997;26:107–16.
[4] Botella O, Peyret R. Benchmark spectral results on the lid-driven cavity flow. Comput Fluids 1998;27:421–33.
[5] Boyd JP. Chebyshev and Fourier spectral method (2nd revised edition). New York: Dover; 2001.
[6] Brandt A. Multi-level adaptive solutions to boundary-value problems. Math Comput 1977;31:333–90.
[7] Brandt A, Fulton SR, Taylor GD. Improved spectral multigrid methods for periodic elliptic problems. J Comput Phys 1985;58:96–112.
[8] Bruneau CH, Jouron C. An efficient scheme for solving steady incompressible Navier–Stokes equations. J Comput Phys 1990;89:389–413.
[9] Canuto C, Hussaini MY, Quarteroni A, Zang TA. Spectral methods in fluid dynamics. New York: Springer-Verlag; 1987.
[10] Chorin AJ. A numerical method for solving incompressible viscous flow problems. J Comput Phys 1967;2:12–26.
[11] Chou MH. A multigrid pseudospectral method for steady flow computation. Int J Numer Methods Fluids 2003;43:25–42.
[12] Deng GB, Piquet J, Queutey P, Visonneau M. Incompressible flow calculations with a consistent physical interpolation finite volume approach. Comput Fluids 1994;23:1029–47.
[13] Dimitropoulos CD, Edwards BJ, Chae KS, Beris AN. Efficient pseudospectral flow simulations in moderately complex geometries. J Comput Phys 1998;144:517–49.
[14] Gauthier S. A semi-implicit collocation method: application to thermal convection in 2D compressible fluids. Int J Numer Methods Fluids 1991;12:985–1002.
[15] Ghia U, Ghia KN, Shin CT. High-Re solutions for incompressible flow using the Navier–Stokes equations and a multigrid method. J Comput Phys 1982;48:387–411.
[16] Gjesdal T, Wasberg CE, Reif BAP. Spectral element benchmark simulations of natural convection in two-dimensional cavities. Int J Numer Methods Fluids 2006;50:1297–319.
[17] Gong Q, Fahroo F, Ross IM. Spectral algorithm for pseudospectral methods in optimal control. AIAA J Guid Control Dyn 2008;31:460–71.
[18] Guillard H, Male JM, Peyret R. Adaptive spectral methods with application to mixing layer computations. J Comput Phys 1992;102:114–27.
[19] Gupta MM, Manohar RP, Noble B. Nature of viscous flows near sharp corners. Comput Fluids 1981;9:379–88.
[20] Hancock C, Lewis E, Moffatt HK. Effects of inertia in forced corner flows. J Fluid Mech 1981;112:315–27.
[21] Heinrichs W. A 3D spectral multigrid method. Appl Math Comput 1991;41:117–28.
[22] Heinrichs W. Spectral multigrid techniques for the Navier–Stokes equations. Comput Method Appl M 1993;106:297–314.
[23] Heinrichs W. Splitting techniques for the unsteady Stokes equations. SIAM J Numer Anal 1998;35:1646–62.
[24] Heinrichs W, Kattelans T. A direct solver for the least-squares spectral collocation system on rectangular elements for the incompressible Navier–Stokes equations. J Comput Phys 2008;227:4776–96.
[25] Jameson A. Numerical solution of the Euler equations by finite volume methods using Runge–Kutta time-stepping schemes. AIAA Paper. p. 81–1259.
[26] Ku HC, Hirsh RS, Taylor TD. A pseudospectral method for solution of the three-dimensional incompressible Navier–Stokes equations. J Comput Phys 1987;70:439–62.
[27] Ku HC, Taylor TD, Hirsh RS. Pseudospectral methods for solution of the incompressible Navier–Stokes equations. Comput Fluids 1987;15:195–214.
[28] Krastev K, Schäfer M. A multigrid pseudo-spectral method for incompressible Navier–Stokes flows. C R Mecanique 2005;333:59–64.
[29] Kwak D, Chakravarthy SR. A three-dimensional incompressible Navier–Stokes flow solver using primitive variables. AIAA J 1986;24:390–6.
[30] Lilek Z, Muzaferija S, Peric M. Efficiency and accuracy aspects of a full-multigird simple algorithm for three-dimensional flows. Numer Heat Transfer B - Fund 1996;31:23–42.
[31] Lottes JW, Fischer PF. Hybrid multigrid/Schwarz algorithms for the spectral element method. J Sci Comput. 2005;24:613–46.
[32] Olson L. Algebraic multigrid preconditioning of high-order spectral elements for elliptic problems on a simplicial mesh. SIAM J Sci Comput 2007;29:2189–209.

[33] Peyret R. Spectral methods for incompressible viscous flow. New York: Springer-Verlag; 2002.
[34] Phillips TN. Relaxation schemes for spectral multigrid methods. J Comput Appl Math 1987;18:149–62.
[35] Pinelli A, Vacca A. Chebyshev collocation method and multidomain decomposition for the incompressible Navier–Stokes equations. Int J Numer Methods Fluids 1994;18:781–99.
[36] Sakell L. Use of local time stepping with full pseudospectral solutions to the Navier–Stokes equations of motion for flows with shock waves. Comput Math Appl 1990;19:85–94.
[37] Schultz WW, Lee NY, Boyd JP. Chebyshev pseudospectral method of viscous flows with corner singularities. J Sci Comput 1989;4:1–24.
[38] Trefethen LN. Spectral methods in Matlab. Philadelphia: SIAM; 2000.
[39] Vanka SP. Block-implicit multigrid solution of Navier–Stokes equations in primitive variables. J Comput Phys 1986;65:138–58.
[40] Zang TA, Wong YS, Hussaini MY. Spectral multigrid methods for elliptic equations. J Comput Phys 1982;48:485–501.
[41] Zang TA, Wong YS, Hussaini MY. Spectral multigrid methods for elliptic equations II. J Comput Phys 1984;54:489–507.