INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids **30**: 441–459 (1999)

MULTIGRID SOLUTION OF THE INCOMPRESSIBLE NAVIER–STOKES EQUATIONS FOR THREE-DIMENSIONAL RECIRCULATING FLOW: COUPLED AND DECOUPLED SMOOTHERS COMPARED

M.F. PAISLEY*

Department of Mathematics and Statistics, School of Computing, Staffordshire University, Stafford ST18 0AD, UK

SUMMARY

A comparison of multigrid methods for solving the incompressible Navier–Stokes equations in three dimensions is presented. The continuous equations are discretised on staggered grids using a second-order monotonic scheme for the convective terms and implemented in defect correction form. The convergence characteristics of a decoupled method (SIMPLE) are compared with those of the cellwise coupled method (SCGS). The convergence rates obtained for computations of the three-dimensional lid-driven cavity problem are found to be very similar to those obtained for computations of the corresponding two-dimensional problem with comparable grid density. Although the convergence rate of SCGS is thus superior to that of SIMPLE, the decoupled method is found to be more efficient computationally and requires less computing time for a given level of convergence. The linewise implementation of the coupled method (CLGS) is also investigated and shown to be more efficient than SCGS, although the convergence rate and computing time required per cycle are both found to depend on the direction of sweep. The optimal implementation of CLGS is found to be only marginally more effective than SIMPLE, but a change to the structure of the data storage would increase the advantage. Copyright © 1999 John Wiley & Sons, Ltd.

KEY WORDS: incompressible Navier–Stokes; multigrid; three-dimensional flow

1. INTRODUCTION

In recent years the application of multigrid methods to the incompressible Navier–Stokes equations has been a subject of great research interest. The achievement of convergence rates that are independent of grid size has enabled researchers to take advantage of the advances in computer technology and generate solutions to complex flow problems on desktop workstations in computing times that are affordable. The most popular methods are those based on a discretisation of the primitive variables and an associated solver (smoother), with a multigrid algorithm used to accelerate the convergence of the solution process. The multigrid algorithm is essentially a nested procedure for transferring information between grids in a hierarchical structure and consists of appropriate coarse-to-fine and fine-to-coarse operators. Given the discrete equations and the transfer operators, the heart of the process is the solver, and two general types have been developed. Decoupled algorithms such as the semi-implicit method for

* Correspondence to: Department of Mathematics and Statistics, School of Computing, Staffordshire University, Stafford ST18 0AD, UK. E-mail: cmtmfp@soc.staffs.ac.uk

CCC 0271–2091/99/120441–19$17.50

Received February 1998
Revised July 1998

pressure-linked equations (SIMPLE) [1] historically predate coupled algorithms such as the symmetric coupled Gauss–Seidel (SCGS) method [2] and are quite opposite in their philosophy. The former solves each momentum equation over the entire domain in turn, and converts the continuity equation into a Poisson equation to achieve local coupling between the velocities and the pressure, while the latter solves a locally coupled set of all the equations together and achieves global coupling by sweeping over the domain. Each can be used as a smoother in a multigrid algorithm and both are in widespread use—see [3–5] for the decoupled method and [6–8] for the coupled method.

A limited number of comparisons have been made of the relative effectiveness of decoupled and coupled smoothers for the two-dimensional driven cavity. References [9,10] concluded that SCGS outperforms SIMPLE and while the same conclusion was drawn in [11], the results suggested that for a more complicated discretisation scheme than the hybrid scheme, the difference between the two is less pronounced. Both methods have been extended to deal with flows in three dimensions—see [12–14] for the coupled strategy and [15] for the decoupled, but it would appear that no corresponding comparison for the three-dimensional driven cavity problem has been made. No comparison is attempted in [12] or [15], while both [13] and [14] make the somewhat one-sided comparison of the coupled approach with multigrid against the decoupled approach without. It would appear also that no comparison of the smoothing rates obtained for these three-dimensional computations compared with those obtained for two dimensions has been carried out.

The usual discretisation techniques in all these computations are based either on staggered or collocated grids—both of which present their own complexity. The use of staggered grids complicates the geometry so that several types of control volumes must be taken into account, but the discretisation is relatively straightforward, while the use of collocated grids simplifies the geometry but complicates the discretisation. Decoupled methods have been applied to multigrid implementations in two and three dimensions on both staggered [3–5,9–11] and collocated grids [15–17]. The implementation of coupled methods, on the other hand, is only natural for staggered grids—there is no corresponding formulation for the collocated case. The trend away from staggered grids to collocated grids has thus meant a renewal of interest in decoupled methods.

Early attempts at discretising the convective terms relied heavily on the hybrid scheme, which switches from second-order central differencing to first-order upwinding when the grid Peclet number rises to 2 or more. Better discretisations are readily available that consist of the first-order upwind contribution and a second-order contribution specially constructed to enhance the accuracy but retain monotonicity. These are implemented in the form of a deferred correction and have a demonstrable effect in reducing the number of grid points required for a given level of resolution.

The original coupled method SCGS just solves for corrections to a pressure and the six adjacent staggered velocities (four in two dimensions), by inverting a locally coupled set of seven equations (five in two dimensions) based around a typical continuity control volume. This cellwise procedure is known to be ineffective when grid lines are stretched or the flow is largely unidirectional and a modification is required. It can be shown that coupling together all continuity control volumes in a row or line and solving the resulting block tridiagonal matrix to update all variables simultaneously considerably enhances the convergence characteristics under these circumstances. This approach is known as the collective line Gauss–Seidel (CLGS) [18]. It was shown in [11] that the convergence rate of such a linewise approach may deteriorate slightly for recirculating flow in a cavity compared with the cellwise approach, but on the other hand is more efficient in terms of the numbers of flux evaluations required and leads to a reduction in computing time per cycle. It was also shown that this approach had better convergence characteristics than the symmetric coupled alternating line (SCAL) method [19], especially for unidirectional flows. The three-dimensional equivalent of solving in lines in two dimensions is solving in planes. However, rather than develop a genuine planewise method where all variables in a plane are updated simultaneously, the three-dimensional line solver described in this paper merely sweeps in lines over the two other co-ordinate directions. This raises questions concerning the choice of the sweeping direction for optimal smoothing and empirical results in [11] for flows that are largely unidirectional suggested that sweeping in a direction perpendicular to the main flow was superior to sweeping parallel to the flow. Sweeping in all three co-ordinate directions is tested here to determine which if any is preferred.

Thus, the purposes of this paper are three-fold. The first is the comparison of the SIMPLE and SCGS methods as multigrid smoothers in terms of convergence rates and computing times required for a given level of convergence. The second is to compare the convergence rates obtained for the three-dimensional problem with those obtained for the two-dimensional case, while the third is to compare the performance of SIMPLE and SCGS approaches with the linewise approach CLGS. The paper continues with a description of the mathematical problem and the discretisation, after which the smoothers are described in more detail and the multigrid implemetation is given. Discussion of the results follows and the paper ends with some general conclusions.

2. THE CONTINUOUS PROBLEM

The equations of motion for steady laminar viscous flow in conservation form are

$$\frac{\partial}{\partial x_j}\left[\rho u_j \varphi - \Gamma \frac{\partial \varphi}{\partial x_j}\right] = S^{\varphi}, \tag{1}$$

where $\varphi = u_i$, $\Gamma = \mu$ and S^{φ} is a source term containing the pressure gradient, and

$$\frac{\partial}{\partial x_j}[\rho u_j] = 0. \tag{2}$$

Equation (1) represents conservation of momentum in each of the three co-ordinate directions, while (2) represents the continuity constraint. The geometry under consideration is that of a unit cubic cavity with boundary conditions consistent with an upper lid moving with unit velocity and all other faces static. Although this is a standard test case, it is not without its difficulties, since three-dimensional cavity flows reach transition at relatively low Reynolds numbers. Several steady state computations have been reported in the literature for $Re = 1000$, including [12–16], although analysis [20] suggests that unstable spanwise modes may occur at Reynolds numbers below this value. For simplicity, however, here it is assumed that a steady flow exists at Reynolds numbers up to and including 1000, and that it is symmetric about $y = 0.5$. Neumann conditions are imposed on this plane and the solution actually only computed in half the domain, as illustrated in Figure 1.

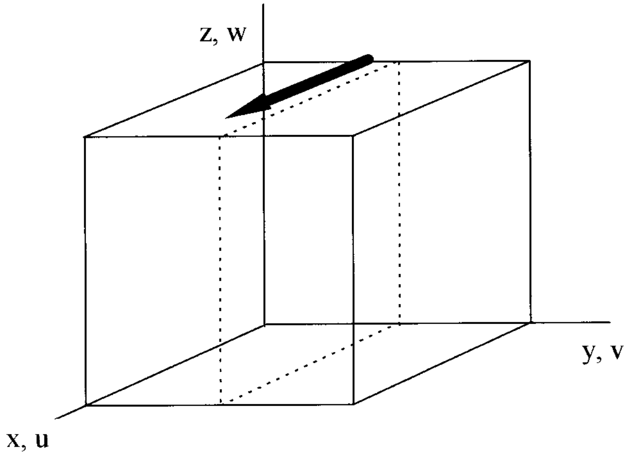

Figure 1. Geometry of unit cube showing the symmetry plane and direction of motion of upper boundary.

3. THE DISCRETE PROBLEM

The discretisation scheme used has been described in [5,11] and follows many of the standard practices of the finite volume method. The momentum equations (1) are integrated over a typical control volumes of volume ΔV and the divergence theorem used to give the usual discrete boundary integral

$$\sum\left[\left(u\varphi-\Gamma\frac{\partial\varphi}{\partial x}\right)\Delta A_{yz}+\left(v\varphi-\Gamma\frac{\partial\varphi}{\partial y}\right)\Delta A_{xz}+\left(w\varphi-\Gamma\frac{\partial\varphi}{\partial z}\right)\Delta A_{xy}\right]=S^{\varphi}\Delta V, \tag{3}$$

where φ denotes u, v or w and the summation is taken over the six faces of the control volume. Here ΔA_{yz} denotes the area of the cell face in the y–z plane etc. Staggered grids are used with the usual arrangement of variables and associated control volumes, Figure 2. Local interpolation is used for u, v and w and the diffusive terms are obtained by standard two-point differencing. The advective terms are approximated by a monotonic scheme derived by considering the flow across cell boundaries to be one-dimensional in directions normal to the cell faces. The value of the flow variable at the centre of the cell under consideration is denoted φ_{p}, with φ_{u} and φ_{d} denoting the corresponding values at the centres of the cells immediately upwind and downwind respectively. The estimate on the downwind cell face, φ_{f}, midway between φ_{p} and φ_{d} is given by the harmonic mean [21,22] of the values arising from averaging neighbouring values (central differencing) and second-order upwind interpolation, i.e.

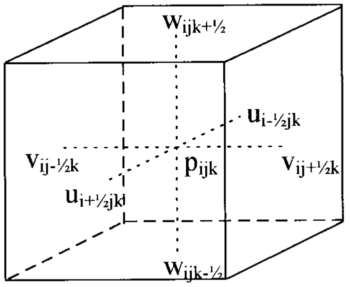

Figure 2. Staggered grid arrangement of flow variables and continuity control volume.

$$\varphi_{\rm f} = \frac{1}{2}(\varphi_{\rm p} + \varphi_{\rm d})\hat{\varphi}_{\rm p} + \frac{1}{2}(3\varphi_{\rm p} - \varphi_{\rm u})(1 - \hat{\varphi}_{\rm p}),$$

where $\hat{\varphi}_{\rm p}$ is the ratio of differences

$$\hat{\varphi}_{\rm p} = \frac{\varphi_{\rm p} - \varphi_{\rm u}}{\varphi_{\rm d} - \varphi_{\rm u}}.$$

The estimate is actually implemented as

$$\varphi_{\rm f} = \varphi_{\rm p} + (\varphi_{\rm d} - \varphi_{\rm p})\hat{\varphi}_{\rm p},$$

from which the monotonicity property easily follows, since $\varphi_{\rm p} \leq \varphi_{\rm f} \leq \varphi_{\rm d}$ is clearly satisfied if $0 \leq \hat{\varphi}_{\rm p} \leq 1$. The estimate thus consists of the value immediately upstream ('first-order upwinding') plus a second-order correction. Values of $\hat{\varphi}_{\rm p}$ outside the range $0 \leq \hat{\varphi}_{\rm p} \leq 1$ correspond to extreme points in the flow, and only the value immediately upstream is used.

The convective and diffusive contributions from the six cell faces are collected in this manner so that (3) can be expressed in the form

$$a_{\rm p}\varphi_{\rm p} = \sum a_{\rm m}\varphi_{\rm m} + S^{\varphi}_{\rm p}. \tag{4}$$

The summation is taken over the centres of the six neighbouring cells, and the multiplying coefficients contain the convective and diffusive flow rates. The source term includes the second-order corrections from the convective scheme and the pressure gradient term discretised using two-point differencing in the usual way.

A sequence of grids is defined where the maximum number of lines on grid $N_{\rm g}$ ($N_{\rm g} = 1$ corresponds to the finest grid) in the x-, y- and z-directions are $I{\rm MAX}_{N_{\rm g}}$, $J{\rm MAX}_{N_{\rm g}}$ and $K{\rm MAX}_{N_{\rm g}}$ respectively. To avoid wasteful use of storage space, single-dimensional arrays are used to store all variables in rows in the x-direction, so that consecutive elements contain horizontal neighbours in general. A pointer system is used to address the elements at position (I, J, K) on grid $N_{\rm g}$ of the form

$$M = I + (J-1) \times I{\rm MAX}_{N_{\rm g}} + (K-1) \times I{\rm MAX}_{N_{\rm g}} \times J{\rm MAX}_{N_{\rm g}} + {\rm NG}_{N_{\rm g}},$$

where ${\rm NG}_{N_{\rm g}}$ is a summation term over the grids defined by

$${\rm NG}_1 = 0, \quad {\rm NG}_{N_{\rm g}} = {\rm NG}_{N_{\rm g}-1} + I{\rm MAX}_{N_{\rm g}-1} \times J{\rm MAX}_{N_{\rm g}-1} \times K{\rm MAX}_{N_{\rm g}-1}, \quad N_{\rm g} > 1.$$

Although all the results described later were obtained on uniform Cartesian grids, the code retains the capacity for computation on non-uniform Cartesian grids.

4. SMOOTHING ALGORITHMS

With reference to a typical continuity control volume, Figure 2, it is supposed that the exact discrete values u, w and p satisfy the discrete steady momentum equations

$$a^u_{i-1/2jk}u_{i-1/2jk} = \sum a^u_{\rm m}u_{\rm m} + A^u_{i-1/2jk}(p_{i-1jk} - p_{ijk}) + S^u_{i-1/2jk}, \tag{5}$$

$$a^u_{i+1/2jk}u_{i+1/2jk} = \sum a^u_{\rm m}u_{\rm m} + A^u_{i+1/2jk}(p_{ijk} - p_{i+1jk}) + S^v_{i+1/2k}, \tag{6}$$

$$a^v_{ij-1/2k}v_{ij-1/2k}=\sum a^v_{\mathrm{m}}v_{\mathrm{m}}+A^v_{ij-1/2k}(p_{ij-1k}-p_{ijk})+S^v_{ij-1/2k}, \tag{7}$$

$$a^v_{ij+1/2k}v_{ij+1/2k}=\sum a^v_{\mathrm{m}}v_{\mathrm{m}}+A^v_{ij+1/2k}(p_{ijk}-p_{ij+1k})+S^v_{ij+1/2k}, \tag{8}$$

$$a^w_{ijk-1/2}w_{ijk-1/2}=\sum a^w_{\mathrm{m}}w_{\mathrm{m}}+A^w_{ijk-1/2}(p_{ijk-1}-p_{ijk})+S^w_{ijk-1/2}, \tag{9}$$

$$a^w_{ijk+1/2}w_{ijk+1/2}=\sum a^w_{\mathrm{m}}w_{\mathrm{m}}+A^w_{ijk+1/2}(p_{ijk}-p_{ijk+1})+S^w_{ijk+1/2}, \tag{10}$$

where $A^u_{i-1/2jk}=\Delta A^u_{i-1/2jk}/\Delta x^u_{i-1/2jk}$, etc, and the source terms have been multiplied by the volumes of respective control volumes. The discrete continuity equation is

$$(u_{i+1/2jk}-u_{i-1/2jk})\Delta A^x_{ijk}+(v_{ij+1/2k}-v_{ij-1/2k})\Delta A^y_{ijk}+(w_{ijk+1/2}-w_{ijk-1/2})\Delta \mathrm{A}^z_{ijk}=0. \tag{11}$$

4.1. Decoupled smoother (SIMPLE)

The SIMPLE pressure correction algorithm [1] has been discussed many times and only a brief resume is given here. Expression (4) comprises a diagonally dominant set of equations over the entire domain for each of the variables u, v and w. Using an estimated pressure field the coefficients in the discrete momentum equations (5)–(10) are calculated and the equations solved globally in turn via an ADI sweep to yield updates to the current velocity field. The rest of the iteration provides coupling between the local velocities and pressures and uses relations between corrections to velocities and adjacent pressure values (see Figure 2) derived from the discrete momentum equations:

$$u'_{i-1/2jk}=\frac{A^u_{i-1/2jk}}{a^u_{i-1/2jk}}(p'_{i-ljk}-p'_{ijk}),\qquad v'_{ij-1/2k}=\frac{A^v_{ij-1/2k}}{a^u_{ij-1/2k}}(p'_{ij-lk}-p'_{ijk}),$$
$$w'_{ijk-1/2}=\frac{A^w_{ijk-1/2}}{a^w_{ijk-1/2}}(p'_{ijk-l}-p'_{ijk}). \tag{12}$$

These are substituted into the discrete continuity equation written terms of velocity corrections

$$(u'_{i+1/2jk}-u'_{i-1/2jk})\Delta A^x_{ijk}+(v'_{ij+1/2k}-v'_{ij-1/2k})\Delta A^y_{ijk}+(w'_{ijk+1/2}-w'_{ijk-1/2})\Delta A^z_{ijk}=-R^c_{ijk}, \tag{13}$$

where R^c_{ijk} is the residual of the continuity equation for the current velocity field, to derive a Poisson-type equation for the pressure corrections:

$$A_{ijk}p'_{ijk}-\sum A_{\mathrm{m}}p'_{\mathrm{m}}=-R^c_{ijk}. \tag{14}$$

Equation (14) is solved using four ADI sweeps to yield the pressure corrections, which are used to update the velocities through relationships (12). Underrelaxation is required for a convergent iteration, with a typical value for the momentum equations being 0.7. The pressure correction equation itself is not underrelaxed, but only a fraction of the resulting pressure corrections is added to the current pressure field, the value of which is typically 0.3.

4.2. Coupled smoother—cellwise (SCGS)

Vanka's SCGS method [2,12] is based on solving equations derived from (5)–(11) for the corrections to the values of the six velocities and one pressure associated with a typical continuity control volume. The seven equations are written as the bordered matrix system:

$$\begin{bmatrix} a^u_{i-1/2jk} & 0 & 0 & 0 & 0 & 0 & A^u_{i-1/2jk} \\ 0 & a^u_{i+1/2jk} & 0 & 0 & 0 & 0 & -A^u_{i+1/2jk} \\ 0 & 0 & a^v_{ij-1/2k} & 0 & 0 & 0 & A^v_{ij-1/2k} \\ 0 & 0 & 0 & a^v_{ij+1/2k} & 0 & 0 & -A^v_{ij+1/2k} \\ 0 & 0 & 0 & 0 & a^w_{ijk-1/2} & 0 & A^w_{ijk-1/2} \\ 0 & 0 & 0 & 0 & 0 & a^w_{ijk+1/2} & -A^w_{ijk+1/2} \\ -\Delta A^x_{ijk} & \Delta A^x_{ijk} & -\Delta A^y_{ijk} & \Delta A^y_{ijk} & -\Delta A^z_{ijk} & \Delta A^z_{ijk} & 0 \end{bmatrix}$$

$$\times \begin{bmatrix} u'_{i-1/2jk} \\ u'_{i+1/2jk} \\ v'_{ij-1/2k} \\ v'_{ij+1/2k} \\ w'_{ijk-1/2} \\ w'_{ijk+1/2} \\ p'_{ijk} \end{bmatrix} = \begin{bmatrix} -R^u_{i-1/2jk} \\ -R^u_{i+1/2jk} \\ -R^v_{ij-1/2k} \\ -R^v_{ij+1/2k} \\ -R^w_{ijk-1/2} \\ -R^w_{ijk+1/2} \\ -R^c_{ijk} \end{bmatrix}.$$

The matrix is inverted analytically to yield the required updates which are immediately added to the values of the current solution. Underrelaxation is implemented by adding a fraction of the changes calculated to the respective field variables—typical values for the fraction being in the range 0.7–0.9 for the velocities and 0.8–1.0 for the pressure. The philosophy of the SCGS method is entirely opposite to that of the SIMPLE method in that for SCGS, immediate local linkage is provided between variables and the global coupling is achieved by sweeping through the domain in a prescribed manner, such as forward lexicographic ordering followed by backward. Continuity control volumes are visited in turn and all the coefficients of the transport equations are recalculated to take account of the updated values of flow variables. As discussed in [11], the bulk of the work is in assembling the equation coefficients needed to calculate the current value of the residuals, and this outweighs the overhead of inverting the matrix, particularly in three dimensions (see later).

4.3. Coupled smoother—linewise (CLGS)

The cellwise implementation described above is known to have poor convergence characteristic when grids are stretched or the flow is largely unidirectional, and an alternative is to update variables for an entire line of continuity control volumes. The CLGS [18] method does precisely this. In three dimensions there is clearly the choice of which direction to choose for the collective update. In two dimensions [11], sweeping vertically (the z-direction) was found to be better than horizontally (x-direction), while in this work, in three dimensions sweeping laterally (the y-direction) has been found better than either of the others. The formulation in [11] was given in terms of vertical sweeping, and the necessary extensions for the three-dimensional case are given here in terms of the same direction. The starting point is to write the variables in groups of six as indicated in Figure 3, i.e.

$$[\ldots, w_{ijk-1/2}, u_{i-1/2jk}, u_{i+1/2jk}, v_{ij-1/2k}, v_{ij+1/2k}, p_{ijk}, \ldots]^T.$$

Vertically adjacent pressures are now linked, giving a minor modification to the vertical momentum equation, while the form of the other two momentum equation remains unchanged. This gives rise to a matrix system with the following block structure:

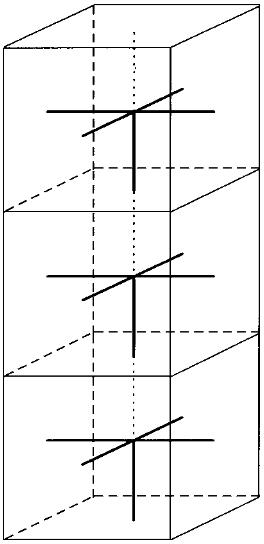

Figure 3. Grouping of flow variables associated with a vertical line of continuity control volumes for line sweeping in the z-direction.

$$\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & -A^{w}_{ijk-1/2} & a^{w}_{ijk-1/2} & 0 & 0 & 0 & 0 & A^{w}_{ijk-1/2} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & a^{u}_{i-1/2jk} & 0 & 0 & 0 & A^{u}_{i-1/2jk} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a^{u}_{i+1/2jk} & 0 & 0 & -A^{u}_{i+1/2k} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a^{v}_{ij-1/2k} & 0 & A^{v}_{ij-1/2k} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a^{v}_{ij+1/2k} & -A^{v}_{j+1/2k} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -\Delta A^{z}_{ijk} & -\Delta A^{x}_{ijk} & \Delta A^{x}_{ijk} & -\Delta A^{y}_{ijk} & \Delta A^{y}_{ijk} & 0 & \Delta^{z}_{ijk} & 0 & 0 & 0 & 0 & 0
\end{bmatrix}$$

$$\times \begin{bmatrix} \text{::::} \\ \text{::::} \\ w'_{ijk-1/2} \\ u'_{i-1/2jk} \\ u'_{i+1/2jk} \\ v'_{ij-1/2k} \\ v'_{ij+1/2k} \\ p'_{ijk} \\ \text{::::} \\ \text{::::} \end{bmatrix} = \begin{bmatrix} \text{::::} \\ \text{::::} \\ -R^{w}_{ijk-1/2} \\ -R^{u}_{i-1/2jk} \\ -R^{u}_{i+1/2jk} \\ -R^{v}_{ij-1/2k} \\ -R^{v}_{ij+1/2k} \\ -R^{c}_{ijk} \\ \text{::::} \\ \text{::::} \end{bmatrix}.$$

The structure of this matrix is block-tridiagonal and is solved using a specially written routine. In terms of flux evaluations, more cell faces can be considered shared than before, and the work count reduced compared with the cellwise version. Underrelaxation is again achieved by adding a fraction of the computed changes as described for the cellwise version.

5. MULTIGRID IMPLEMENTATION

The implementation of the multigrid algorithm has been described for the two-dimensional case in [5,11]. Here the same basic method is used with appropriate extensions for the three-dimensional problem. The manner of grid coarsening is that continuity control volumes on a coarse grid are formed from the sum of eight such control volumes on the immediately preceding fine grid. Restriction of the velocities to a coarse grid is carried out using the means of the four nearest fine grid neighbours, while equation residuals are restricted using summation with appropriate scaling to preserve volume integrals The cycling algorithm was kept the same for all the smoothing algorithms tested, with W-cycles used to visit the grids in the hierarchy, with one pre-smoothing and one post-smoothing iteration on each. The number of iterations used to solve the equations on the coarsest grid ranged between 5 and 15 depending on the Reynolds number. Prolongation of the solution corrections was implemented using trilinear interpolation.

6. RESULTS AND DISCUSSION

Converged solutions have been obtained using each of the smoothing methods SIMPLE, SCGS and CLGS described earlier on three uniform grids containing $16 \times 8 \times 16$, $32 \times 16 \times 32$ and $64 \times 32 \times 64$ cells and for three Reynolds numbers $Re = 100$, 400 and 1000. The features of the flow are shown in the velocity vectors of Figure 4, where the main vortex is accompanied by wall vortices of increasing strength as the Reynolds number rises. Rather than compare line plots of velocity profiles to assess the accuracy of the underlying discretisation, the author uses the simpler measure given by the extreme velocity values attained in the centre (symmetry) plane. The three values used are the maximum negative value of the horizontal velocity on the vertical centre line and the maximum and minimum values of the vertical velocity on the horizontal centre line. These quantities obtained on the three grids are plotted in Figure 5 against the square of the grid spacing for the three Reynolds numbers used. For the two lower Reynolds numbers, the discretistion exhibits clear second-order behaviour, although less clearly for $Re = 1000$ and further data is required. Plotted for comparison, however, are the values given for each quantity by Deng *et al.* [23] for a second-order scheme on the finest grid, which indicate very close agreement.

The multigrid performances of the SIMPLE and SCGS methods are given in Tables I and II, while that of the CLGS method is given in Tables III, IV and V, which correspond to linewise coupling in the x-, y- and z-directions respectively. The tables give the number of multigrid cycles and the computing time required on a Silicon Graphics Indy Workstation in double precision arithmetic for the absolute value of the L_2 norm of the residual to fall below 10^{-4}, where the definition of the norm of the residual is the usual one:

$$R = \left[\frac{\sum (R^u)^2 + \sum (R^v)^2 + \sum (R^w)^2 + \sum (R^c)^2}{4 \times I\mathrm{MAX}_{N_g} \times J\mathrm{MAX}_{N_g} \times K\mathrm{MAX}_{N_g}}\right]^{1/2}.$$

The individual residuals are normalised by the volumes of the control volumes so that the their dimensions correspond to those of the original partial differential equations. It is seen immediately that a good degree of grid independence is achieved for all the implementations in terms of the number of multigrid cycles required to reach the level of convergence specified. This should lead to an increase in computing time with one level of grid refinement of approximately eight. In fact the increases are somewhat greater than this in general, due to the fact that the computing time per cycle per grid point rises with grid refinement, Table VI. This is not surprising given that as the dimension of the problem increases, the data becomes more scattered within computer memory and access times are longer.

Comparing the results for SIMPLE (Table I) with those of SCGS (Table II) it is clear that although SCGS requires fewer cycles, SIMPLE outperforms SCGS in terms of computing time at all Reynolds numbers. This is not the case for the two-dimensional problem, where in all comparisons made [9–11], SCGS outperforms SIMPLE, and this observation thus merits some discussion. Firstly, it is the case that the convergence rates achieved for the two-dimensional problem are being achieved in three dimensions. This is confirmed in Figure 6, where the convergence histories for the computations with both methods at each Reynolds number on

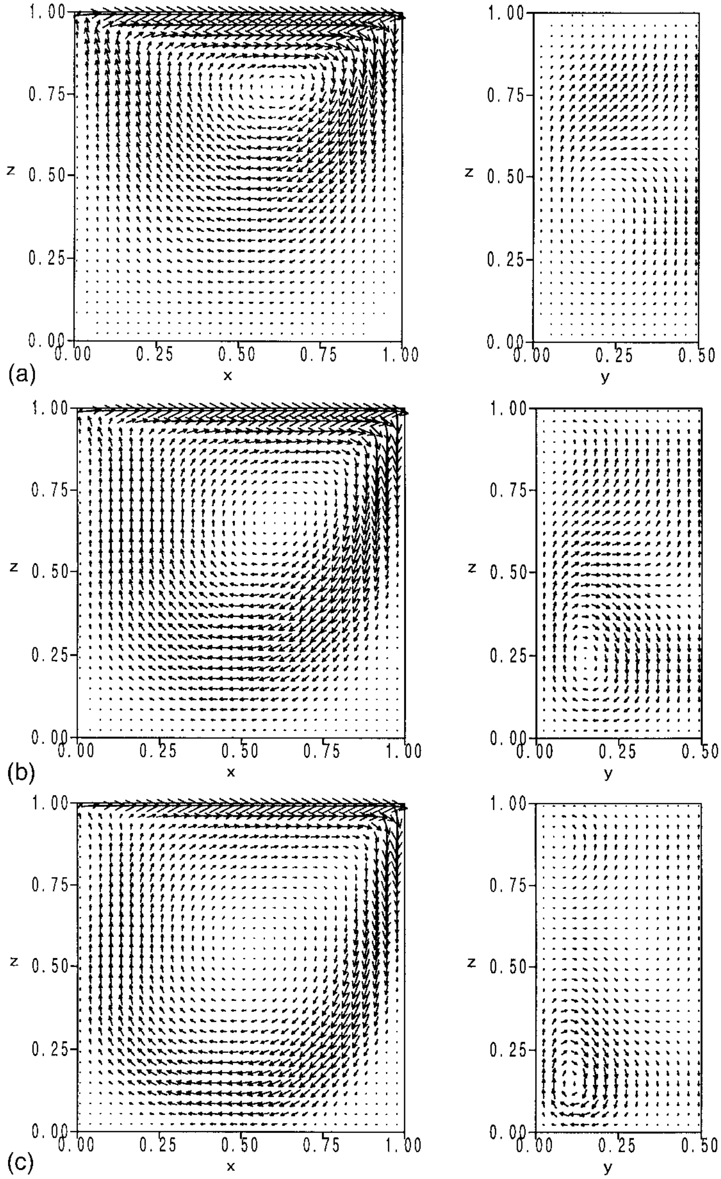

Figure 4. Velocity vectors in the centre plane ($y = 0.5$) and in a spanwise plane ($x = 0.547$) for converged solutions on the finest grid. Note that for clarity only alternate vectors are shown and that the vector lengths in the spanwise plane are magnified by 2.5 compared with those in the centre plane. (a) $Re = 100$, (b) $Re = 400$, (c) $Re = 1000$.

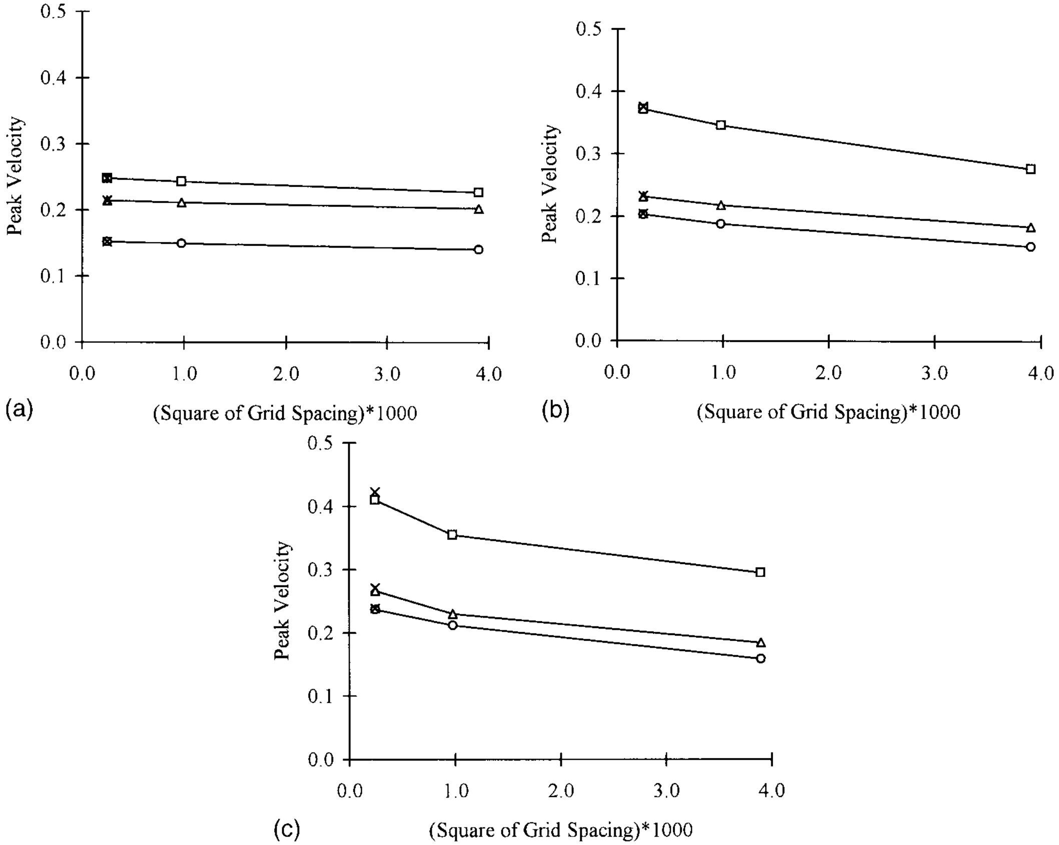

Figure 5. Variation with grid refinement of peak velocity values on the centre plane. (a) $Re = 100$, (b) $Re = 400$, (c) $Re = 1000$. □, $|W_{\min}|$; △, $|U_{\min}|$; ○, $W_{\max}$; ×, Deng *et al.* [23].

the three-dimensional 64 × 32 × 64 grid are compared with corresponding computations on a two-dimensional 64 × 64 grid. The reason for the degradation in the performance of SCGS compared with SIMPLE in extending from two to three dimensions is rather directly related to the nature of the methods themselves, and the additional work per grid cell required in three dimensions compared with two as the following discussion makes clear.

The number of momentum flux evaluations required by SCGS to update the variables associated with one continuity control volume in two dimensions is 16 (or 14 if the flux through shared faces is evaluated only once), so for a forward sweep followed by a backward

Table I. Number of multigrid cycles and computing times for convergence to 10^{-4} for SIMPLE

Grid size	16 × 8 × 16	32 × 16 × 32	64 × 32 × 64
Re			
100	8 (10.4 s)	8 (86 s)	8 (797 s)
400	10 (12.9 s)	10 (107 s)	9 (896 s)
1000	13 (16.4 s)	14 (150 s)	12 (1195 s)

Table II. Number of multigrid cycles and computing times for convergence to 10^{-4} for SCGS

Grid size	$16 \times 8 \times 16$	$32 \times 16 \times 32$	$64 \times 32 \times 64$
Re			
100	5 (15.4 s)	5 (132 s)	4 (962 s)
400	6 (18.4 s)	6 (158 s)	5 (1205 s)
1000	8 (24.4 s)	8 (211 s)	7 (1685 s)

sweep this is 32 (or 28). In three dimensions, this becomes 36 (or 33) for a single sweep, and 72 (or 66) for the two sweeps. Thus, the increase in work per continuity control volume in terms of flux evaluations from two dimensions to three is approximately $72/32 = 2.25$ ($66/28 = 2.36$). The work required to invert the matrix rises by a factor of 1.5 (from 42 floating point operations to 62) but this is only a small proportion of the total computational effort. For the SIMPLE method, on the other hand, the momentum equation coefficients are evaluated once per sweep assuming that all variables are at the same stage of the computation, and so all cell faces can be considered shared. In two dimensions, this means two flux evaluations per continuity control volume for each momentum equation, giving a total of four. In three dimensions, three flux evaluations per continuity control volume are required for each of the three momentum equations giving a total of nine. In terms of the flux evaluations, therefore, the increase in work for SIMPLE is approximately $9/4 = 2.25$, the same as for SCGS. However, much of the work in the SIMPLE method is the solution of the pressure perturbation equation, which is based on the continuity equation. Since this is a single equation, the work required to achieve this in three dimensions is only 1.5 times greater than that required in two dimensions. This leads to the observation that in extending from two to three dimensions, the work per continuity control volume for the SIMPLE method is likely to increase by a smaller factor than for the SCGS method.

Compared with SCGS, the CLGS method is somewhat more efficient, however. Since variables belonging to a whole line of continuity cells are updated together, fewer momentum flux evaluations are required. In two dimensions, the method only requires three evaluations per continuity control volume for the component of momentum aligned with the direction of sweep and five for the other component, giving a total of eight per sweep. Since two sweeps are performed this is 16, compared with a minimum of 28 for SCGS. The reduction seen in [11] was not as great as 16/28, partly because CLGS requires the use of array variables, whereas

Table III. Number of multigrid cycles and computing times for convergence to 10^{-4} for CLGS-X

Grid size	$16 \times 8 \times 16$	$32 \times 16 \times 32$	$64 \times 32 \times 64$
Re			
100	5 (8.8 s)	5 (70 s)	6 (709 s)
400	7 (12.8 s)	9 (127 s)	9 (1065 s)
1000	9 (17.4 s)	12 (171 s)	14 (1661 s)

Table IV. Number of multigrid cycles and computing times for convergence to 10^{-4} for CLGS-Y

Grid size	$16\times 8\times 16$	$32\times 16\times 32$	$64\times 32\times 64$
Re			
100	5 (10.9 s)	5 (92 s)	4 (666 s)
400	6 (13.6 s)	6 (111 s)	5 (840 s)
1000	8 (19.0 s)	8 (149 s)	7 (1186 s)

SCGS does not. In three dimensions, the method requires five evaluations per continuity control volume for the component aligned with the sweeping direction, and nine for each of the other two components. This totals 23 flux evaluations, which doubles to 46 for the two sweeps, compared with a minimum of 66 for SCGS. Thus, on this crude basis, one might expect a reduction in computing time per cycle of around $46/66 \approx 70\%$ for CLGS over SCGS.

The computing times per gridpoint per multigrid cycle for each of the methods were referred to above and are given in Table VI. SCGS is clearly the least efficient in these terms, especially when compared with SIMPLE for the reasons discussed earlier. The corresponding times for CLGS are also less than for SCGS as anticipated, with the reductions for CLGS-X, CLGS-Y and CLGS-Z being approximately 49, 69 and 92% respectively on the finest grid. It is perhaps surprising at first sight that the times for these three implementations of the same algorithm should be so different—after all, the same arithmetic operations are being performed. The explanation lies in the manner of data storage described earlier, where data is stored as single-dimensional vectors, with consecutive elements containing variables that are horizontally adjacent neighbours in the discretisation. Sweeping in a direction that is aligned with the data structure in this way is bound to be favourable, meaning that CLGS-X is the most efficient implementation. Elements in the vector corresponding to laterally adjacent and vertically adjacent neighbours in the discretisation appear separated by an offset (given by IMAX and $I\text{MAX} \times J\text{MAX}$ respectively), so CLGS-Y and CLGS-Z will be less efficient, with the larger offset giving the least efficient method.

Computational efficiency is only one consideration, however, and rate of convergence is equally important. For completeness, the convergence histories of CLGS-X and CLGS-Z for the three-dimensional problem are compared with those of the corresponding two-dimensional problem in Figure 7. Whereas the SIMPLE and SCGS methods reproduced the two-dimensional convergence rates in three dimensions, it is clear that CLGS-X does not perform as well in three dimensions as in two. CLGS-Z on the other hand appears more robust in three dimensions and gives similar rates as in the two-dimensional case.

Table V. Number of multigrid cycles and computing times for convergence to 10^{-4} for CLGS-Z

Grid size	$16\times 8\times 16$	$32\times 16\times 32$	$64\times 32\times 64$
Re			
100	5 (11.9 s)	5 (118 s)	5 (1110 s)
400	8 (19.7 s)	8 (190 s)	6 (1430 s)
1000	12 (29.4 s)	13 (288 s)	12 (2897 s)

Table VI. Variation in computing time (in ms) per grid point per cycle for each smoother with grid refinement

	SIMPLE	SCGS	CLGS-X	CLGS-Y	CLGS-Z
16 × 8 × 16	0.63	1.50	0.86	1.06	1.16
32 × 16 × 32	0.66	1.61	0.86	1.12	1.44
64 × 32 × 64	0.76	1.84	0.90	1.27	1.69

The asymptotic convergence rates of the five implementations are compared in Figure 8 for computations on the finest grid at the three Reynolds numbers. The SIMPLE method is the slowest, while SCGS is the fastest. CLGS-Y is the best of the implementations of the line

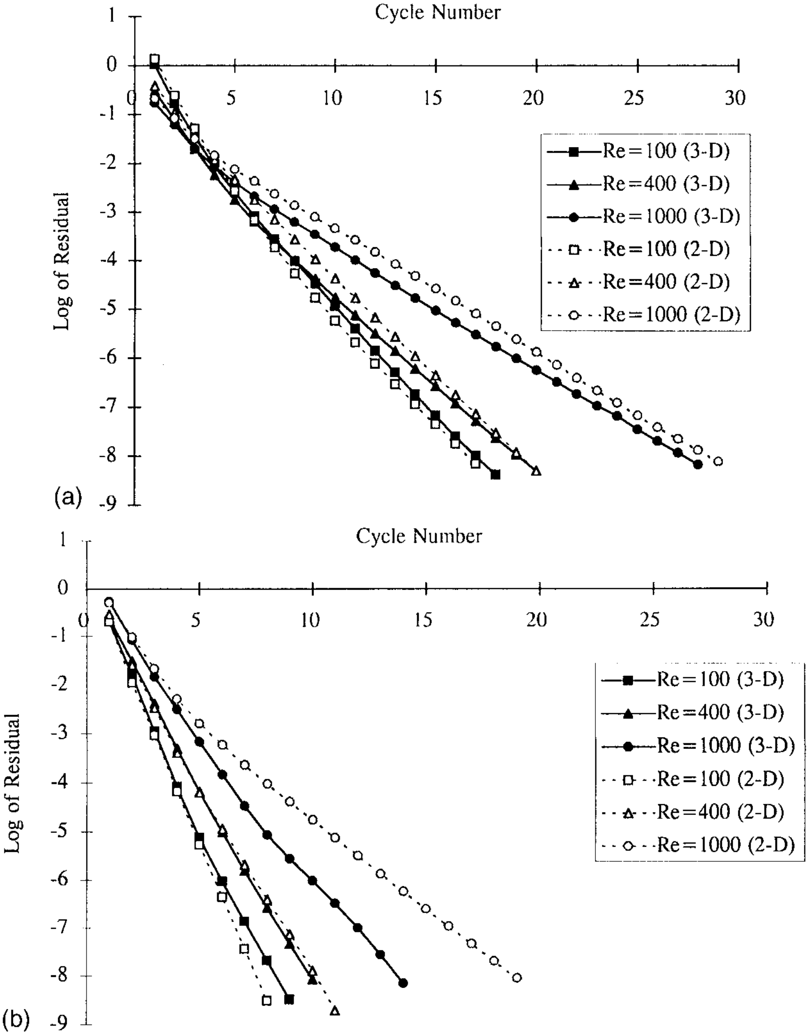

Figure 6. Convergence rates of smoothers obtained for the three-dimensional cubic cavity compared with those obtained for the two-dimensional square cavity with the same grid spacing; (a) SIMPLE, (b) SCGS.

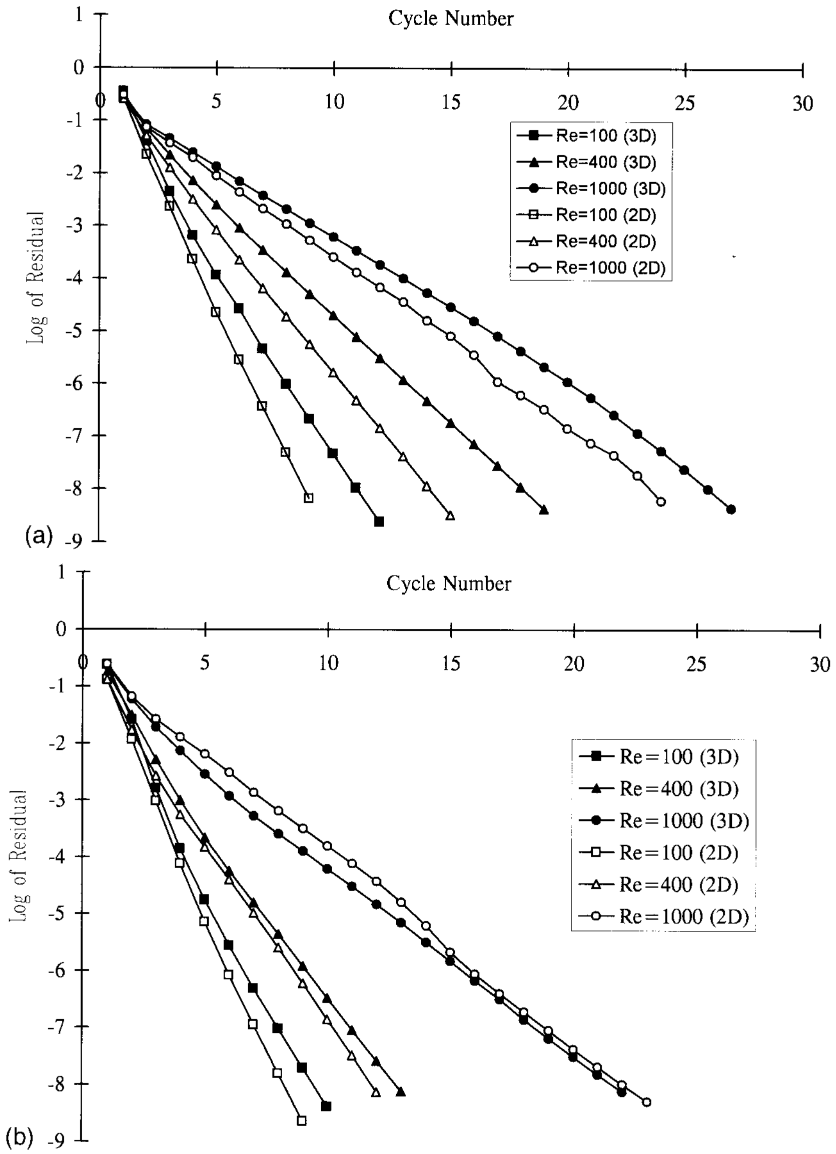

Figure 7. Convergence rates of smoothers obtained for the three-dimensional cubic cavity compared with those obtained for the two-dimensional square cavity with the same grid spacing; (a) CLGS-X, (b) CLGS-Z. Note that CLGS-Y has no counterpart in two dimensions.

solver and reproduces the convergence rate of SCGS for low Reynolds numbers. This is of interest for the direction of sweep of CLGS-Y is perpendicular to the direction of the bulk of the flow and concurs with the observation in [11], where for unidirectional flows in two dimensions, the best strategy was to sweep in a direction perpendicular to that of the flow. The convergence rates of both CLGS-X and CLGS-Z deteriorate in comparison, particularly for the higher Reynolds number. Thus, given its relatively high computational efficiency compared with the other coupled algorithms and its good convergence characteristics over the range of Reynolds numbers tested, CLGS-Y is clearly the best implementation of any of the coupled algorithms described. Compared with SIMPLE, however, although the convergence rate is much better, the greater computational efficiency of the decoupled method means that in fact neither is consistently better than the other for all Reynolds numbers and grids (compare Tables I and IV). As noted in [11] however, a change to the structure of the data storage would

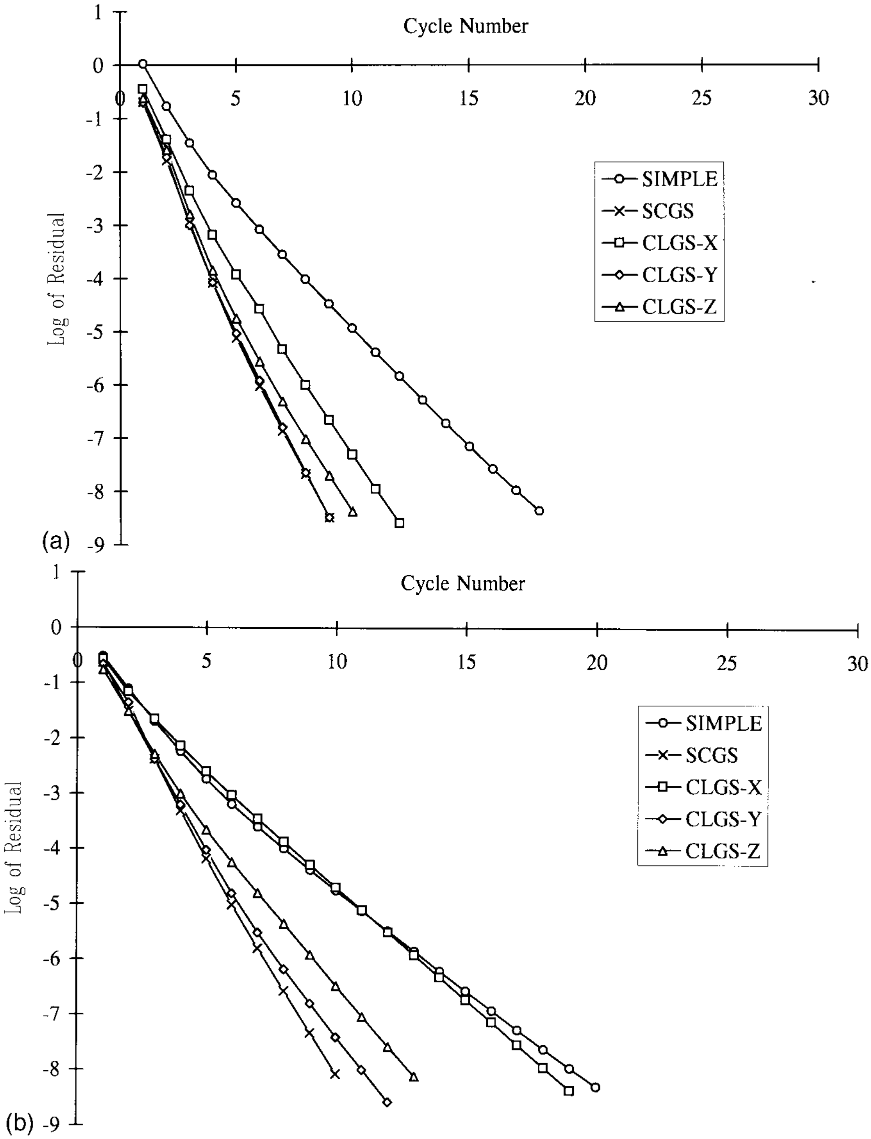

Figure 8. Comparison of the convergence rates obtained with each smoother for the cubic cavity; (a) $Re = 100$, (b) $Re = 400$, (c) $Re = 1000$.

(c)

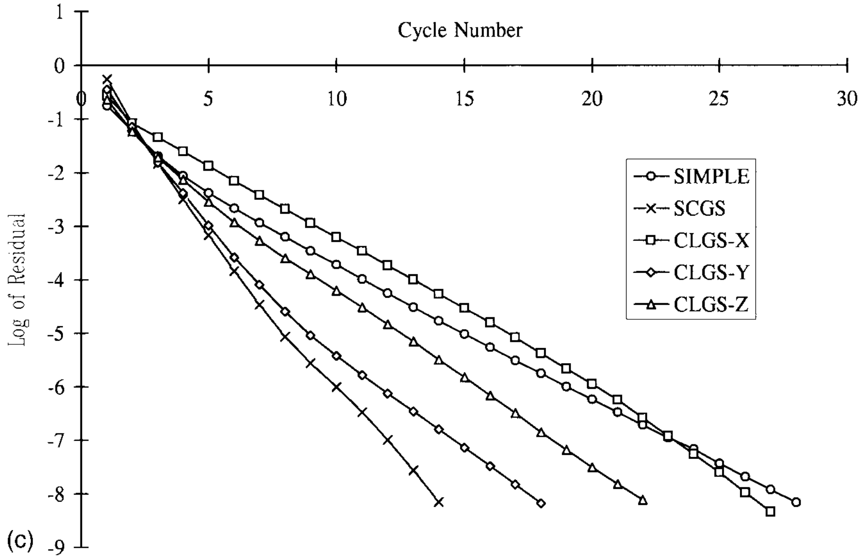

Figure 8 (*Continued*)

provide CLGS-Y with a clear advantage. If the single-dimensional arrays stored variables in rows in the y-direction so that consecutive elements contained spanwise neighbours, the direction of sweep would align with the order in which data is stored. The figures in Table VI imply that computing times for CLGS-Y would then reduce by up to 30%.

7. CONCLUSIONS

Results have been presented in this paper for the decoupled smoother SIMPLE, the cellwise coupled smoother SCGS and the linewise coupled smoother CLGS when used in a multigrid algorithm to solve the discrete Navier–Stokes equations in one half of a symmetric cubic cavity. In conclusion, several remarks can be made. Firstly, with the exception of CLGS-X, the convergence characteristics of each of the methods obtained for the two-dimensional driven cavity problem are generally reproduced when applied to the three-dimensional problem. Since the work on a coarse grid is only an eighth of that on a fine grid (in two dimensions it is one quarter), this should lead to greater speed-up ratios for the three-dimensional problem when compared with the corresponding two-dimensional problem with the same grid spacing.

Secondly, the performance of SIMPLE has been compared with that of SCGS. Although the convergence rate of SCGS is easily superior to that of SIMPLE, the relative inefficiency of SCGS when implemented in three dimensions means that it usually requires at least 50% more computing time to reach a specified convergence level than does SIMPLE.

Thirdly, the relative performance of the line solver CLGS when implemented in the three co-ordinate directions has been investigated. As anticipated from a crude estimate of the number of flux evaluations required, CLGS was found to be more efficient than SCGS. The manner of data storage has an impact on the efficiency of the algorithm, with the most efficient implementation being CLGS-X when the sweeping direction is aligned with the order in which variables are stored, and the least being CLGS-Z, when the offset between variables is largest. In terms of the convergence characteristics, CLGS-Y reproduces most closely the good convergence rates of SCGS, while those of CLGS-X and CLGS-Z deteriorate, particularly for high Reynolds numbers. Given its greater efficiency over SCGS and its good convergence rates, CLGS-Y is found to be the optimal coupled method. The optimal performance of sweeping in a direction perpendicular the bulk of the flow concurs with previous observations for two-dimensional flows [11]. Compared with SIMPLE, however, the gains associated with CLGS-Y are only marginal, and the computing times for a given level of convergence appear to be very similar on all three grids and at all three Reynolds numbers tried. However, it is anticipated that a change in the manner of the data storage would reduce the computing times of CLGS-Y by up to 30%, thus giving the method a clear advantage over SIMPLE for these flows.

It was found that for two-dimensional problems [11], the good convergence rates of the coupled methods evident for idealised test problems, such as the driven cavity, were not reproduced for flows over obstacles where grids are non-uniform, the flow is largely unidirectional and more complicated boundary conditions are imposed. For these flows, the decoupled method was found to be superior to the coupled methods, but whether the same is true in three dimensions remains to be seen

REFERENCES

1. S.V. Patankar, 'A calculation procedure for two-dimensional elliptic situations', *Numer. Heat Transf.*, **4**, 409–425 (1981).
2. S.P. Vanka, 'Block implicit multigrid solution of the Navier–Stokes equations in primitive variables', *J. Comput. Phys.*, **65**, 138–158 (1986).
3. S. Sivaloganathan and G.J. Shaw, 'A multigrid method for recirculating flows', *Int. J. Numer. Methods Fluids*, **8**, 417–440 (1988).
4. F.S. Lien and M.A. Leschziner, 'Multigrid convergence acceleration for complex flow including turbulence', *Multigrid Methods III*, Birkhauser, Basel, 1991, pp. 277–288.
5. M.F. Paisley, 'Multigrid computation of stratified flow over two-dimensional obstacles', *J. Comput. Phys.*, **136**, 411–424 (1997).
6. R. Webster, 'An algebraic multigrid solver for Navier–Stokes problems', *Int. J. Numer. Methods Fluids*, **18**, 761–780 (1989).
7. S. Zeng and P. Wesseling, 'Multigrid solution of the incompressible Navier–Stokes in generalised co-ordinates', *SIAM J. Numer. Anal.*, **31**, 1764–1784 (1994).
8. A.T. Degani and G.C. Fox, 'Parallel multigrid computation of the unsteady incompressible Navier–Stokes equations', *J. Comp. Phys.*, **128**, 223–236 (1996).
9. S. Sivaloganathan, G.J. Shaw, T.M. Shah and D.F. Mayers, 'A comparison of multigrid methods for the incompressible Navier–Stokes equations', in K.W. Morton and M.J. Baines (eds.), *Numerical Methods for Fluid Dynamics*, Oxford University Press, Oxford, 1988, pp. 401–417.
10. C.H. Arakawa, A.O. Demuren, W. Rodi and B. Schonung, 'Application of multigrid methods for the coupled and decoupled solution of the incompressible Navier–Stokes equations', *Notes on Numerical Fluid Dynamics*, vol. 20, Vieweg, Wiesbaden, 1988, pp. 1–8.
11. M.F. Paisley and N.M. Bhatti, 'Comparison of multigrid methods for neutral and stably stratified flows over two-dimensional obstacles', *J. Comput. Phys.*, **142**, 581–610 (1998).
12. S.P. Vanka, 'A calculation procedure for three-dimensional steady recirculating flows using multigrid methods', *Comput. Methods Appl. Mech. Eng.*, **55**, 321–338 (1986).
13. P.H. Gaskell and A.K.C. Lau, 'Comparison of two solution strategies for use with higher-order discretisation schemes in fluid flow simulation', *Int. J. Numer Methods Fluids*, **8**, 1203–1215 (1988).
14. K.C. Karki, P.S. Sathyamurthy and S.V. Patankar, 'Performance of a multigrid method with an improved discretisation scheme for three-dimensional fluid flow calculations', *Numer. Heat Transf. B*, **29**, 275–288 (1996).
15. F.S. Lien and M.A. Leschziner, 'Multigrid acceleration for turbulent flow with a non-orthogonal collocated scheme', *Comput. Methods Appl. Mech Eng.*, **118**, 351–371 (1994).
16. K.M. Smith, W.K. Cope and S.P. Vanka, 'A multigrid procedure for three-dimensional flows on non-orthogonal collocated grids', *Int. J. Numer. Methods Fluids*, **17**, 887–904 (1993).
17. R. Jyotsna and S.P. Vanka, 'Pressure-based multigrid procedure for the Navier–Stokes equations on unstructured grids', in N.D. Melson, T.A. Monteuffel, S.F. McCormick and C.C. Douglas (eds.), *Seventh Copper Mountain Conference on Multigrid Methods*, NASA Conference Publication 3339, 1996, pp. 409–424.

18. S. Zeng and P. Wesseling, 'Numerical study of a multigrid method with four smoothing methods for the incompressible Navier–Stokes equations in general co-ordinates', in N.D. Melson, T.A. Monteuffel and S.F. McCormick (eds.), *Sixth Copper Mountain Conference on Multigrid Methods*, NASA Conference Publication 3224, 1993, pp. 691–708.
19. M.C. Thompson and J.H. Ferziger, 'An adaptive multigrid technique for the incompressible Navier–Stokes equations', *J. Comput. Phys.*, **82**, 94–121 (1989).
20. N. Ramanan and G.M. Homsy, 'Linear stability of lid-driven cavity flow', *Phys. Fluids*, **6**, 2690–2701 (1994).
21. B. Van Leer, 'Towards the ultimate conservative difference scheme II. Monotonicity and conservation combined in a second-order scheme', *J. Comput. Phys.*, **14**, 361–370 (1974).
22. B.P. Leonard and S. Mokhtari, 'Beyond first-order upwinding: the ultra-sharp alternative for non-oscillatory steady state simulation of convection', *Int. J. Numer. Methods Eng.*, **30**, 729–766 (1990).
23. G.B. Deng, J. Piquet, P.Queutey and M. Visonneau, 'A new fully coupled solution of the Navier–Stokes equations', *Int. J. Numer. Methods Fluids*, **19**, 605–639 (1994).