JOURNAL OF COMPUTATIONAL PHYSICS 113, 26–34 (1994)

A Multigrid Solver for the Steady Incompressible Navier–Stokes Equations on Curvilinear Coordinate Systems*

LIN-BO ZHANG

State Key Laboratory of Scientific and Engineering Computing, Computing Center, Chinese Academy of Sciences, Beijing 100080, Peoples' Republic of China

Received January 4, 1993; revised October 8, 1993

A multigrid solver for the steady incompressible Navier–Stokes equations on a curvilinear grid is constructed. The Cartesian velocity components are used in the discretization of the momentum equations. A staggered, geometrically symmetric distribution of velocity components is adopted which eliminates spurious pressure oscillations and facilitates the transformation between Cartesian and co-or contravariant velocity components. The SCGS (symmetrical collective Gauss-Seidel) relaxation scheme proposed by Vanka on a Cartesian grid is extended to this case to serve as the smoothing procedure of the multigrid solver, in both "box" and "box-line" versions. Due to the symmetric distribution of velocity components of this scheme, the convergence rate and numerical accuracy are not affected by grid orientation, in contrast to a scheme proposed in the literature in which difficulties arise when the grid lines turn 90° from the Cartesian coordinates. Some preliminary numerical experiences with this scheme are presented.

1. INTRODUCTION

Finite difference and finite element methods for solving the incompressible Navier–Stokes equations are both under development. Although finite element methods have advantages in dealing with irregular geometries, they require complicated calculations and large storage space in the construction and solution of the discrete equations, compared to finite difference methods. An important step toward the treatment of complex geometries with finite difference methods is the introduction of curvilinear coordinate transformation, in which one transforms an irregular computational domain in the physical space into a regular (rectangular) one in the transformed space, and the transformed equations can then be solved using standard techniques. The curvilinear coordinate transformation can also be used to minimize the truncation error of the discrete solution by appropriately distributing the grid nodes; some studies on this aspect were reported in [9].

One of the most fundamental problems in solving the incompressible Navier–Stokes equations in primitive variables using finite difference or finite volume formulations is where to locate the unknowns. Two commonly used approaches are the staggered and non-staggered grids. On a non-staggered grid, all unknowns are located at the same positions as the grid nodes, while on a staggered grid one velocity component is located at the center of each grid surface and the pressure value is located at the center of the grid cell.

The advantage of the non-staggered discretization is that it is simpler to implement, and easier to extend to general curvilinear coordinate systems. But it generates spurious oscillations in the pressure field if central differencing is used in the discretization of the pressure gradient and the continuity equation (the well-known "checkerboard oscillations."). A remedy to this difficulty is the use of one-sided differencing; then the resulting scheme is only of first-order accuracy (perhaps second-order one-sided differencing may be used for discretizing the pressure gradient and/or the continuity equation). Another popular remedy is to add an artificial elliptic term of the pressure to the continuity equation. For example, Linden *et al.* developed a multigrid solver in which the term $-\omega h^2 \Delta p$ is added to the continuity equation, where ω is a positive parameter and h is the grid step size [10]. The main problem with this kind of scheme is that the discrete continuity equation cannot be satisfied to machine accuracy, and the results may be poor if rapid variations in the pressure field exist. The non-staggered discretization has another disadvantage in that artificial boundary conditions for the pressure are needed. In contrast, with the staggered discretization, spurious pressure oscillations cannot appear (at least on a Cartesian grid) and no boundary conditions for the pressure are needed.

The main difficulty in extending the staggered notion from the Cartesian coordinates to general curvilinear

* Supported by National Natural Science Foundation of China and China State Major Key Project for Basic Researches. Also supported by the Laboratory of Numerical Modelling for Atmospheric Sciences and Geophysical Fluid Dynamics, Institute of Atmospheric Physics, Chinese Academy of Sciences.

0021-9991/94 $6.00

coordinates resides in the choice of independent variables. As recommended in [7], from the viewpoint of numerical accuracy and conservation of physical laws, it is better to discretize the momentum equations using the Cartesian velocity components, since it permits the full conservation-law form and the accuracy of the numerical results is less sensitive to the skewness and non-orthogonality of the grid. Also the transformed equations have a much simpler form in the Cartesian velocity components than in the curvilinear ones. On the other hand, to retain the full advantage of the staggered discretization, we should define, on each surface of a grid cell, the velocity component normal to the surface; this means that the contravariant velocity components should be used as independent unknowns. These contradictory considerations make the choice of independent velocity unknowns somewhat delicate. Generally, when taking Cartesian velocity components as independent unknowns, a "naïve" extension of the notion of the staggered discretization is considered by defining one Cartesian velocity component at each grid surface. An example of this type of scheme is the solver developed by Shyy *et al.* [5, 6] which has been successfully applied to the computations of a wide range of flow problems. The performance of these schemes is generally orientation dependent, and they may produce spurious pressure oscillations in some special situations (i.e., the 90° turning of the grid lines with respect to the Cartesian directions). Although Shyy and Vu pointed out, after a detailed analysis, that the spurious pressure oscillations may only appear in very restricted situations which should not be encountered on general curvilinear grids [7], nevertheless the convergence rate may be affected by grid orientation, especially in a multigrid solver where a good h-ellipticity measure [1, 11] of the discrete system is crucial for the overall efficiency. Evidently, this difficulty is caused by the asymmetric location of velocity components and momentum equations.

In the present paper we describe a multigrid solver on a curvilinear grid based on a staggered distribution of unknowns in which all Cartesian velocity components are defined at all gid surfaces. The symmetric distribution of unknowns is important to ensure the independence of the convergence rate and accuracy of the results on the grid orientation. Although in this scheme more discrete velocity components and discrete momentum equations are involved than on a traditional staggered grid, we will see that the extra amount of work introduced is relatively small. An extension of the SCGS relaxation scheme (symmetrical collective Gauss–Seidel relaxation) of Vanka [3] is used as the smoothing procedure of the multigrid solver, in which the projections of the momentum equations in the local curvilinear directions are used to simplify the linear systems of equations to be solved. A box-line version of the SCGS relaxation is also proposed. Some numerical results concerning this scheme are reported.

2. THE GOVERNING EQUATIONS AND THE DISCRETIZATION

We restrict our descriptions to two-dimensional case. The steady incompressible Navier–Stokes equations in primitive variables are written in the form

$$\begin{aligned} -\frac{1}{R}\Delta u + uu_x + vu_y + p_x &= f_1 \\ -\frac{1}{R}\Delta v + uv_x + vv_y + p_y &= f_2 \\ u_x + v_y &= 0 \\ + \text{boundary conditions,} & \end{aligned} \tag{1}$$

where x and y represent the Cartesian directions, u and v denote the Cartesian velocity components, f_1 and f_2 denote the external force, p denotes the pressure, and $R>0$ is the Reynolds number.

We introduce the following coordinate transformation which transforms the computational domain in the (x, y)-plan into a rectangular region in the (ξ, η)-plan:

$$x = x(\xi, \eta), \qquad y = y(\xi, \eta).$$

Then the governing equations are transformed into

$$\begin{aligned} -\frac{1}{R}\Delta_{\xi\eta} u + \frac{1}{J}(Uu_\xi + Vu_\eta) + \frac{1}{J}(y_\eta p_\xi - y_\xi p_\eta) &= f_1 \\ -\frac{1}{R}\Delta_{\xi\eta} v + \frac{1}{J}(Uv_\xi + Vv_\eta) + \frac{1}{J}(-x_\eta p_\xi + x_\xi p_\eta) &= f_2 \\ U_\xi + V_\eta &= 0, \end{aligned} \tag{2}$$

where $J = x_\xi y_\eta - x_\eta y_\xi$ is the Jacobian, $U = y_\eta u - x_\eta v$, and $V = -y_\xi u + x_\xi v$ are the contravariant velocity components and

$$\Delta_{\xi\eta} = \frac{1}{J^2}\left(\alpha\frac{\partial^2}{\partial\xi^2} - 2\beta\frac{\partial^2}{\partial\xi\,\partial\eta} + \gamma\frac{\partial^2}{\partial\eta^2}\right) + P\frac{\partial}{\partial\xi} + Q\frac{\partial}{\partial\eta}$$

is the Laplace operator in the (ξ, η) coordinates. The coefficients for $\Delta_{\xi\eta}$ are defined by

$$\begin{aligned} &\alpha = x_\eta^2 + y_\eta^2, \qquad \beta = x_\xi x_\eta + y_\xi y_\eta, \qquad \gamma = x_\xi^2 + y_\xi^2 \\ &P = -\frac{1}{J^3}[y_\eta(\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta}) \\ &\qquad - x_\eta(\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta})] \\ &Q = -\frac{1}{J^3}[-y_\xi(\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta}) \\ &\qquad + x_\xi(\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta})]. \end{aligned} \tag{3}$$

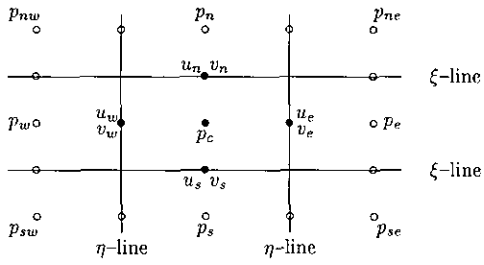

FIG. 1. The current box.

We assume that the above coordinate transformation does not degenerate, i.e., $\alpha > 0$, $\gamma > 0$, $J \neq 0$ (so J does not change sign).

To discretize Eqs. (2), an uniform $(N+1) \times (M+1)$ grid in the (ξ, η)-plan and a staggered distribution of unknowns are used. In contrast to the traditional staggered discretization on a Cartesian grid, both u and v are defined at the center of each grid surface and p is defined at the center of each grid cell (see Fig. 1).

The discrete equations are obtained by approximating all derivatives in Eqs. (2) except the convection terms by second-order central finite differences while second-order upwind differencing is used to discretize the convection terms (however, near boundaries, central differencing is also used for the discretization of the convection terms; refer to [12] for details on the treatment near boundaries). Note that boundary extrapolations of the pressure are needed in the momentum equations since both p_ξ and p_η appear at the same time (or, equivalently, second-order one-sided differencing may be used).

In the following sections, grid surfaces along the ξ-lines will be referred as "ξ-surfaces" and those along the η-lines will be referred as "η-surfaces."

3. THE EXTENDED SCGS RELAXATION SCHEME

The SCGS relaxation scheme was first proposed by S. P. Vanka [3] on a Cartesian grid. It lies in the category of "box relaxation" schemes and has been proved to be an efficient smoother for multigrid solvers, especially for recirculating flows with high Reynolds numbers. For the grid system and finite difference approximation of the Navier–Stokes equations described in the last section, it is easy to extend the SCGS scheme in a straightforward way, as explained below.

For a given grid cell (also called a "box"), we have nine discrete unknowns (including eight Cartesian velocity components located on the four surfaces and one pressure located at the center) and nine discrete equations (eight momentum equations and one continuity equation), as shown by Fig. 1 (the unknowns in the current box are those defined at the locations marked by "•" in Fig. 1). The discrete momentum equations are first linearized and diagonalized by substituting the off-diagonal velocity components and the coefficients of the convection terms with their most recent old values; then they are coupled with the discrete continuity equation to form a 9×9 linear system of equations of the form

$$\begin{bmatrix} * & 0 & 0 & 0 & 0 & 0 & 0 & 0 & * \\ 0 & * & 0 & 0 & 0 & 0 & 0 & 0 & * \\ 0 & 0 & * & 0 & 0 & 0 & 0 & 0 & * \\ 0 & 0 & 0 & * & 0 & 0 & 0 & 0 & * \\ 0 & 0 & 0 & 0 & * & 0 & 0 & 0 & * \\ 0 & 0 & 0 & 0 & 0 & * & 0 & 0 & * \\ 0 & 0 & 0 & 0 & 0 & 0 & * & 0 & * \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & * & * \\ * & * & * & * & * & * & * & * & 0 \end{bmatrix} \begin{bmatrix} u_e \\ v_e \\ u_w \\ v_w \\ u_s \\ v_s \\ u_n \\ v_n \\ p_c \end{bmatrix} = \begin{bmatrix} * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \end{bmatrix}, \quad (4)$$

where "*" denotes nonzero entries. This is very similar to the SCGS relaxation on a Cartesian grid, except that the linear system of equations to be solved involves more unknowns and equations. Due to the special structure of the linear system the extra amount of work needed for its solution, relative to the solution of a 5×5 linear system of the same form as in the original SCGS relaxation, is small compared to the total amount of work.

But if we extend the above relaxation scheme to a corresponding box-line version following the steps of [13], then the linear system of equations to be solved for a ξ-line will have $7N-2$ unknowns with a coefficient matrix of band width 15, instead of a system of $4N-1$ unknowns and band width nine as on a Cartesian grid. In this case, the extra amount of work introduced in the solution of the linear systems will be important. In fact, the system (4) can be split into one 5×5 linear system of equations and four scalar equations by projecting the discrete momentum equations in curvilinear directions.

We consider the projection of the momentum equations in the contravariant directions since it permits the same form of relations between contravariant velocity components and the pressure as in the original SCGS relaxation. For example, on the east surface of the current grid cell, we can write the linearized and diagonalized momentum equations in the form

$$\begin{aligned} A_e u_e + \frac{1}{J_e}(y_\eta^e p_\xi^e - y_\xi^e p_\eta^e) &= g_1^e \\ A_e v_e + \frac{1}{J_e}(-x_\eta^e p_\xi^e + x_\xi^e p_\eta^e) &= g_2^e, \end{aligned} \quad (5)$$

where A_e, g_1^e, and g_2^e are constants (computed from the most recent values of the unknowns and other data),

$J_e = x_\xi^e y_\eta^e + y_\xi^e x_\eta^e$; (p_ξ^e, p_η^e) represents finite difference approximation of the pressure gradient on the east surface which is computed by

$$p_\xi^e = (p_e - p_c)/\Delta\xi$$
$$p_\eta^e = \tfrac{1}{4}(p_{ne} + p_n - (p_{se} + p_s))/\Delta\eta,$$

where $\Delta\xi$ and $\Delta\eta$ are the grid step sizes.

Subsequently, "c," "e," "w," "s," and "n" appearing in the super- or subscripts will always indicate the locations where corresponding values are evaluated ("c" means "central" or "current," "e" means "east," "w" means "west," etc.).

By projecting Eqs. (5) in the normal direction $(y_\eta^e, -x_\eta^e)$ we obtain the equation

$$A_e U_e + \frac{1}{J_e}(\alpha_e p_\xi^e - \beta_e p_\eta^e) = y_\eta^e g_1^e - x_\eta^e g_2^e,$$

where $U_e = y_\eta^e u_e - x_\eta^e v_e$ is the contravariant velocity component and $\alpha_e = (x_\eta^e)^2 + (y_\eta^e)^2$, $\beta_e = x_\xi^e x_\eta^e + y_\xi^e y_\eta^e$, as defined in (3). This equation can be further written as

$$A_e U_e - \frac{1}{J_e \Delta\xi}\alpha_e p_c = G_e,$$

where

$$G_e = y_\eta^e g_1^e - x_\eta^e g_2^e - \frac{1}{J_e \Delta\xi}\alpha_e p_e - \frac{1}{J_e}\beta_e p_\eta^e.$$

Since G_e does not contain unknowns defined in the current box, it can be considered as a known constant.

The normal projections of the momentum equations on the west, south, and north surfaces can be similarly constructed. By combining the four resulting equations with the continuity equation, we obtain the following 5×5 linear system of equations coupling the contravariant velocity components and the pressure:

$$\begin{aligned}
A_e U_e - \frac{\alpha_e}{J_e \Delta\xi} p_c &= G_e \\
A_w U_w + \frac{\alpha_w}{J_w \Delta\xi} p_c &= G_w \\
A_n V_n - \frac{\gamma_n}{J_n \Delta\eta} p_c &= G_n \\
A_s V_s + \frac{\gamma_s}{J_s \Delta\eta} p_c &= G_s \\
\frac{1}{\Delta\xi}(U_e - U_w) + \frac{1}{\Delta\eta}(V_n - V_s) &= 0.
\end{aligned} \tag{6}$$

It is easy to show that the system (6) always has a unique solution (note that A_e, A_w, A_s, and A_n are all positive coefficients).

For the original momentum equations to hold, we need to involve their projections in other directions. The natural choice is the tangent directions of the grid surfaces, i.e., (x_η, y_η) for the east and west surfaces and (x_ξ, y_ξ) for the sourth and north surfaces. Since the resulting equations are decoupled from each other, we find easily the solutions

$$\begin{aligned}
\tilde{V}_e &= (x_\eta^e g_1^e + y_\eta^e g_2^e - p_\eta^e)/A_e \\
\tilde{V}_w &= (x_\eta^w g_1^w + y_\eta^w g_2^w - p_\eta^w)/A_w \\
\tilde{U}_s &= (x_\xi^s g_1^s + y_\xi^s g_2^s - p_\xi^s)/A_s \\
\tilde{U}_n &= (x_\xi^n g_1^n + y_\xi^n g_2^n - p_\xi^n)/A_n,
\end{aligned} \tag{7}$$

where $\tilde{V}_j = x_\eta^j u_j + y_\eta^j v_j$ $(j = e, w)$ and $\tilde{U}_j = x_\xi^j u_j + y_\xi^j v_j$ $(j = s, n)$ are the covariant velocity components.

Using the solutions of (6)–(7) the Cartesian velocity components are updated by:

$$\begin{aligned}
u_e &= (y_\eta^e U_e + x_\eta^e \tilde{V}_e)/\alpha_e \\
v_e &= (-x_\eta^e U_e + y_\eta^e \tilde{V}_e)/\alpha_e \\
u_w &= (y_\eta^w U_w + x_\eta^w \tilde{V}_w)/\alpha_w \\
v_w &= (-x_\eta^w U_w + y_\eta^w \tilde{V}_w)/\alpha_w \\
u_s &= (x_\xi^s \tilde{U}_s - y_\xi^s V_s)/\gamma_s \\
v_s &= (y_\xi^s \tilde{U}_s + x_\xi^s V_s)/\gamma_s \\
u_n &= (x_\xi^n \tilde{U}_n - y_\xi^n V_n)/\gamma_n \\
v_n &= (y_\xi^n \tilde{U}_n + x_\xi^n V_n)/\gamma_n.
\end{aligned} \tag{8}$$

A complete SCGS relaxation swap consists of scanning all grid cells in a pre-determined order (e.g., lexicographical) and of updating the unknowns defined on each cell using (6)–(8).

Remark. At convergence, the discrete solutions will satisfy the equations,

$$\begin{aligned}
&\text{on the } \eta\text{-surfaces,} && y_\eta L_1(u, v, p) - x_\eta L_2(u, v, p) = 0 \\
& && x_\eta L_1(u, v, p) + y_\eta L_2(u, v, p) = 0; \\
&\text{on the } \xi\text{-surfaces,} && x_\xi L_1(u, v, p) + y_\xi L_2(u, v, p) = 0 \\
& && -y_\xi L_1(u, v, p) + x_\xi L_2(u, v, p) = 0,
\end{aligned} \tag{9}$$

where $L_1(u, v, p) = Qu + (1/J)(y_\eta p_\xi - y_\xi p_\eta) - f_1$, $L_2(u, v, p) = Qv + (1/J)(-x_\eta p_\xi + x_\xi p_\eta) - f_2$, and $Q = (1/R)\Delta_{\xi\eta} + (1/J)(U(\partial/\partial\xi) + V(\partial/\partial\eta))$ (in a discrete sense). We obtain immediately from (9) that $L_1(u, v, p) = 0$ and $L_2(u, v, p) = 0$,

so the momentum equations are satisfied. It is interesting to note that if we only update the contravariant velocity components by setting $\tilde{V}_e = \tilde{V}_w = \tilde{U}_s = \tilde{U}_n = 0$ in (8) (in this case the values of V on the η-surfaces and those of U on the ξ-surfaces used in the coefficients of the convection terms (see Eqs. (2)) should be calculated by interpolations, since only the normal velocity component are available on each surface), then the final solution satisfies the equations,

$$\begin{aligned} y_\eta L_1(u, v, p) - x_\eta L_2(u, v, p) &= 0, \quad \text{on the } \eta\text{-surfaces} \\ -y_\xi L_1(u, v, p) + x_\xi L_2(u, v, p) &= 0, \quad \text{on the } \xi\text{-surfaces.} \end{aligned} \tag{10}$$

Equations (10) do not guarantee that the original momentum equations are satisfied. But if the grid is formed by two families of uniformly distributed parallel straight lines in the (x, y)-plan, i.e., if the metric derivatives x_ξ, x_η, y_ξ, and y_η are all constants, then it is easy to see that Eqs. (10) are equivalent to

$$\begin{aligned} L_1(U, V, p) &= 0, \quad \text{on the } \eta\text{-surfaces} \\ L_2(U, V, p) &= 0, \quad \text{on the } \xi\text{-surfaces.} \end{aligned}$$

The above equations just represent the traditional staggered discretization using the contravariant velocity components.

4. BOX-LINE VERSION OF THE SCGS RELAXATION

The relaxation scheme described in the last section is a "box" relaxation scheme. For efficiently handling anisotropic problems, a box-line version of it is needed. The box-line relaxation is constructed by, instead of a single grid cell, considering simultaneously all grid cells in a ξ-line or η-line and updating all unknowns located in these cells by solving a linear system of equations.

Consider the case of ξ-line relaxation. All unknowns at the locations marked by "$\bullet$" in Fig. 2 are to be updated simultaneously using corresponding discrete equations. In order to describe the relaxation scheme, consider, for example, the linearized momentum equations defined at the location marked by "$\bullet$" in Fig. 3. We write these eqations in the form

$$A_w u_w + A_c u_c + A_e u_e + \frac{1}{J}(y_\eta^c p_\xi - y_\xi^c p_\eta) = g_1$$

$$A_w v_w + A_c v_c + A_e v_e + \frac{1}{J}(-x_\eta^c p_\xi + x_\xi^c p_\eta) = g_2,$$

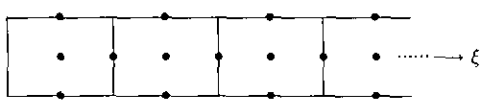

FIG. 2. Unknowns updated simultaneously on a ξ-line.

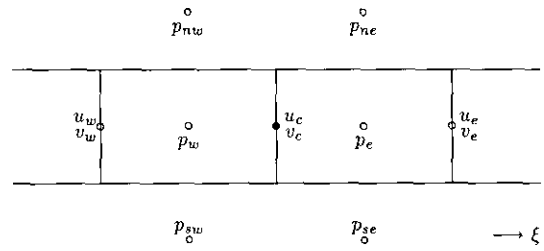

FIGURE 3

where A_e, A_c, A_w, g_1, and g_2 represent known coefficients (the notations used here are different from those in the last section). As in the box version, we project the above equations in the direction $(y_\eta^c, -x_\eta^c)$ and the resulting equation is

$$A_w(y_\eta^c u_w - x_\eta^c v_w) + A_c(y_\eta^c u_c - x_\eta^c v_c) + A_e(y_\eta^c u_e - x_\eta^c v_e) + \frac{1}{J}(\alpha p_\xi - \beta p_\eta) = y_\eta^c g_1 - x_\eta^c g_2.$$

For obtaining a linear system of equations on the contravariant velocity components, we write the above equation in the form

$$\begin{aligned} A_w U_w + A_c U_c + A_e U_e + \frac{\alpha}{J} p_\xi &= y_\eta^c g_1 - x_\eta^c g_2 + \frac{\beta}{J} p_\eta \\ &\quad + A_w[(y_\eta^w - y_\eta^c) u_w - (x_\eta^w - x_\eta^c) v_w)] \\ &\quad + A_e[(y_\eta^e - y_\eta^c) u_e - (x_\eta^e - x_\eta^c) v_e)], \end{aligned}$$

where $U_j = -y_\eta^j u_j - x_\eta^j v_j$ $(j = w, c, e)$. Replacing p_ξ and p_η by their finite difference approximations and substituting the most recent old values of the velocity components into the right-hand side, we obtain a linear equation coupling the contravariant velocity components and the pressures of the form:

$$A_w U_w + A_c U_c + A_e U_e + \frac{\alpha}{J \Delta\xi}(p_w - p_e) = G.$$

The other momentum equations can be treated similarly. The resulting linear equations are closed by the continuity

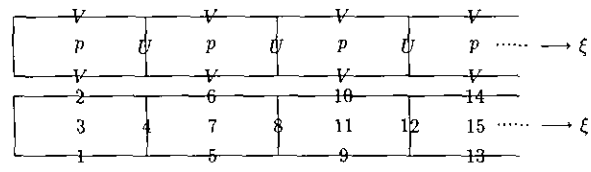

FIG. 4. Numbering of unknowns in a ξ-line.

equations defined on the current ξ-line. Thus we obtain a linear system of $4N-1$ equations and $4N-1$ unknowns coupling the contravariant velocity components and the pressures. Using the numbering of unknowns shown in Fig. 4, the band width of the coefficient matrix equals nine, so the linear system can be solved by a direct method.

The linear equations on covariant velocity components are obtained in the same way. Since these equations are decoupled from the pressures of the current ξ-line, we only have to solve three tri-diagonal linear systems, with N or $N-1$ unknowns, respectively.

5. THE MULTIGRID PROCEDURE AND NUMERICAL RESULTS

The FAS (full approximation storage, cf. [1, 2]) scheme is used in the multigrid solver. The coarser grids are obtained from the finest one by dropping successively odd grid lines in both ξ and η directions (the grid lines are numbered from 0 to N in the ξ direction and from 0 to M in the η direction on the finest grid). The restriction and prolongation operators constructed in [12] for a Cartesian grid are used here for transferring functions and residuals between grids (however, these operators are of only first-order accuracy on non-smooth grids). The other details of the multigrid solver are just the same as in [12].

The first numerical example is the driven cavity problem. The finest grid is a 65×65 one formed by zigzag lines. The orthogonality and skewness of the grid is controlled by a parameter $\delta \geqslant 0$. With $\delta = 0$ we obtain a uniform Cartesian grid. Computations are done with $R = 100$ for $\delta = 0$, 0.2, 0.4, and 0.6, respectively. The corresponding grids are shown in Fig. 5. For $\delta = 0.6$ we have an excessively skew, non-smooth, and non-orthogonal grid.

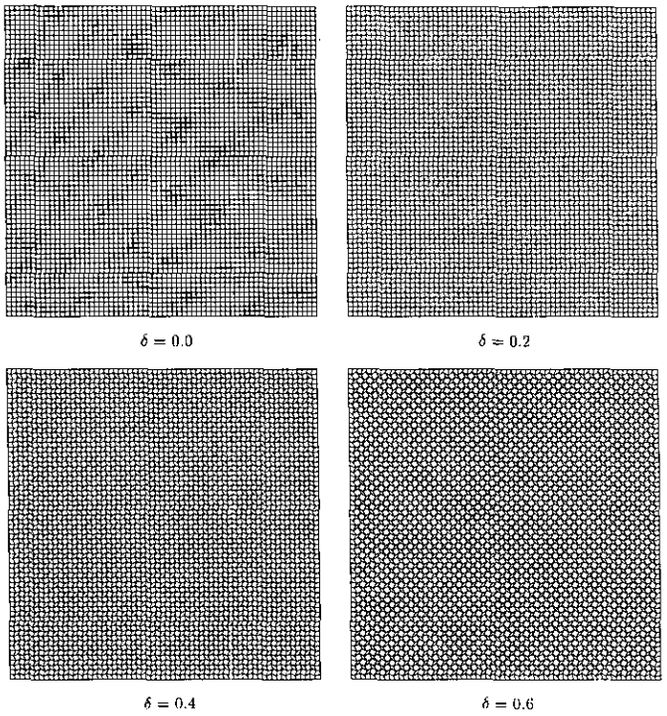

FIG. 5. Test problem 1—the finest grid.

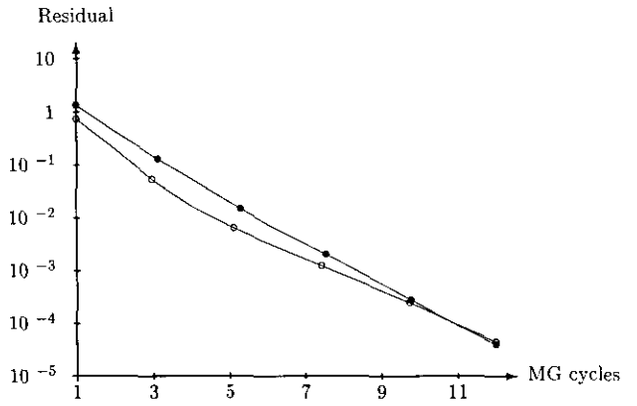

FIG. 6. Test problem 1—convergence history for $\delta = 0.0$: "$\circ$," box relaxation; "$\bullet$," box-line relaxation.

Both box and box-line relaxations are tested for this problem. The $V(1, 1)$ cycles are used in the multigrid iterations. The box-line relaxations are realized in a symmetric way; i.e., the ξ-line relaxation is used in the pre-relaxations and the η-line relaxation is used in the post-relaxations. Through numerical experience we find that when evaluating the coefficients of the convection terms, only the normal velocity components should be used. This means that on a η-surface, for example, V in Eqs. (2) should be calculated by interpolating those values defined on the four nearest ξ-surfaces. Otherwise we obtain very slow convergence.

The convergence histories of the multigrid solver are shown in Fig. 6–9. These figures show that the convergence rate of the multigrid solver is affected by the grid skewness

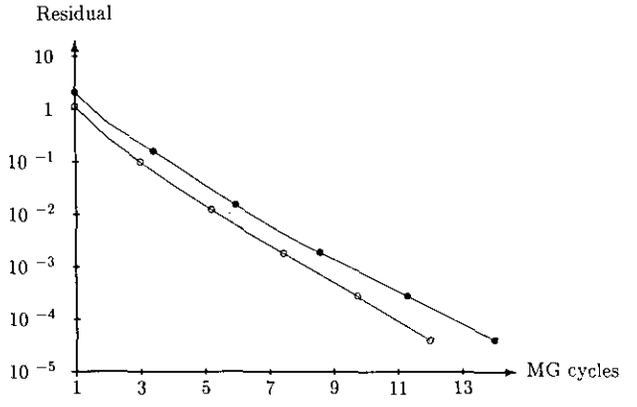

FIG. 7. Test problem 1—convergence history for $\delta = 0.2$: "$\circ$," box relaxation; "$\bullet$," box-line relaxation.

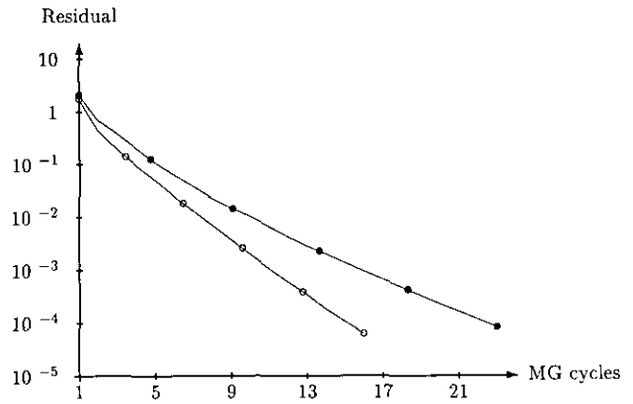

FIG. 8. Test problem 1—convergence history for $\delta = 0.4$: "$\circ$," box relaxation; "$\bullet$," box-line relaxation.

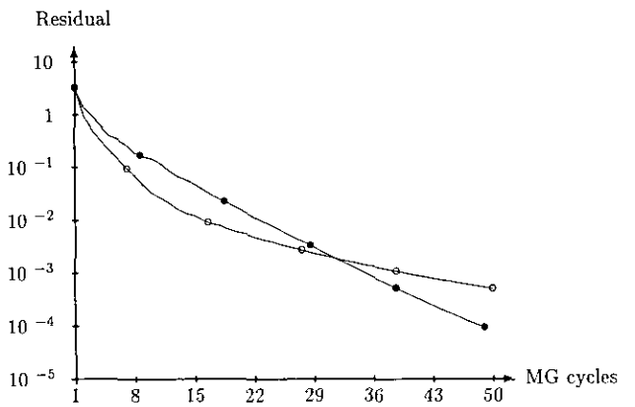

FIG. 9. Test problem 1—convergence history for $\delta = 0.6$: "$\circ$," box relaxation; "$\bullet$," box-line relaxation.

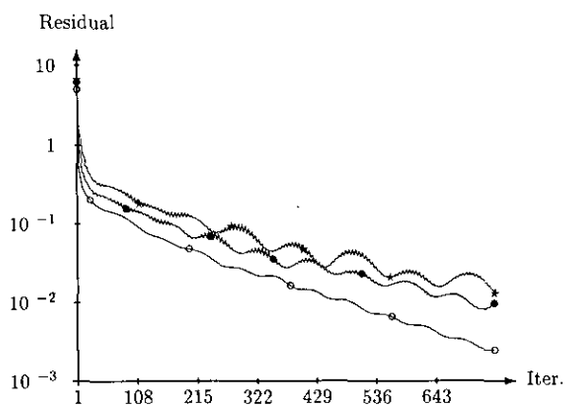

FIG. 10. Test problem 1—convergence history of the box-line relaxation scheme as a single-grid solver: $\circ$, $\delta = 0$; $\bullet$, $\delta = 0.4$; $*$, $\delta = 0.6$.

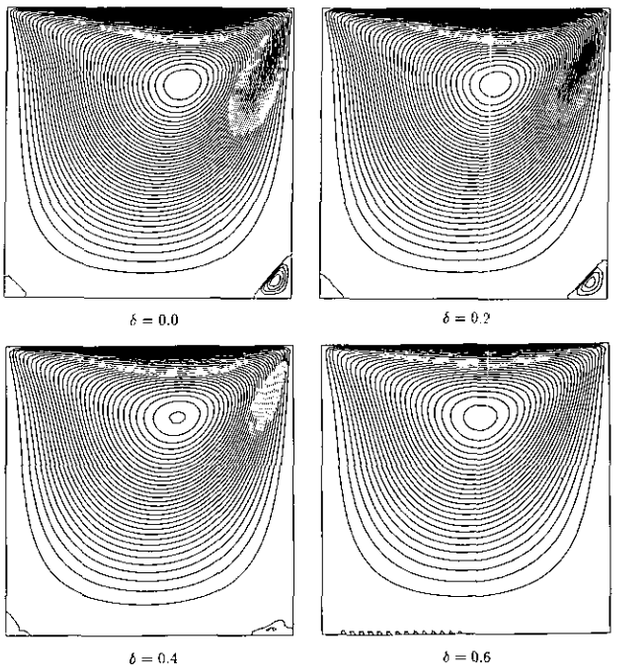

FIG. 11. Test problem 1—streamlines.

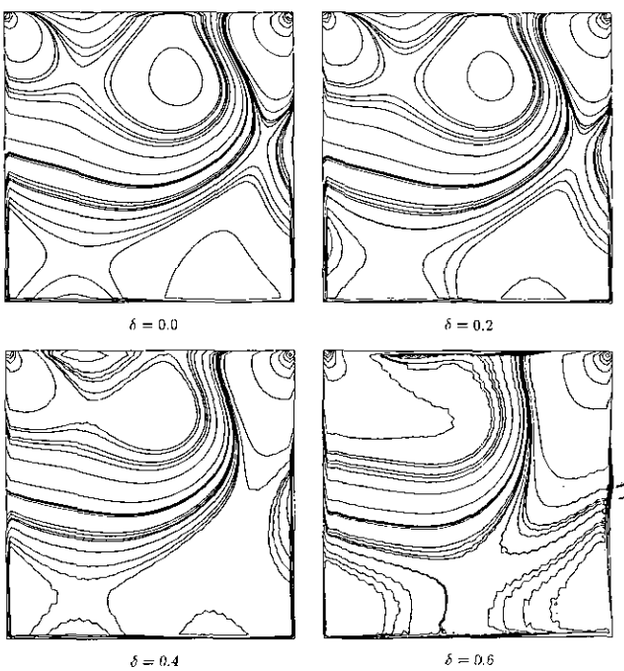

FIG. 12. Test problem 1—isobars.

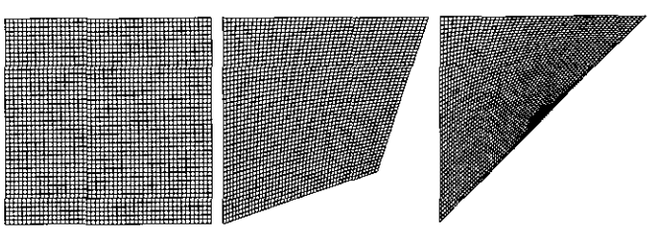

FIG. 13. Test problem 2—the grids.

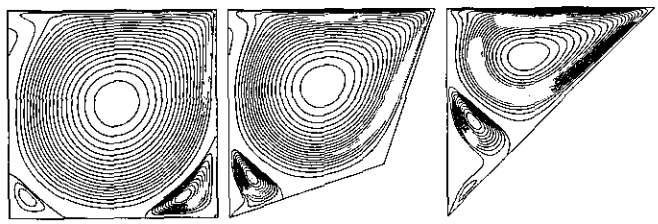

FIG. 14. Test problem 2—streamlines.

TABLE I

CPU Time per Iteration in Seconds

Metric derivatives precalculated		Metric derivatives calculated at run-time	
One relaxation	One $V(1, 1)$ cycle	One relaxation	One $V(1, 1)$ cycle
4.66	15.91	7.37	25.79
5.94	20.04	8.43	29.36

and non-orthogonality. But we think that an important reason of the slow convergence for large values of δ may be the poor interpolation operators used in the multigrid solver since they are second-order operators only on smooth grids. This is to be confirmed by further studies.

In above calculations, the under-relaxation factor is set to 0.75, 0.75, 0.5, and 0.4 for $\delta = 0$, 0.2, 0.4, and 0.6, respectively. They are chosen after several simple trials. We have not sought the optimal relaxation factors.

Another remark about Figs. 6–9 is that the asymptotic convergence rate of the box-line relaxation scheme is faster than the box relaxation for $\delta = 0$ and 0.6, but slower for $\delta = 0.2$ and 0.4. We cannot draw any conclusion here about this fact because the relaxation factors used are not the optimal ones. On a vector computer like the Convex C120, the box-line relaxation requires less CPU time than the box relaxation because the former is more suitable for vector processing.

Figure 10 shows the convergence history of the box-line relaxation as a single-grid solver. In this computation, the relaxations are also performed in a symmetric manner by using ξ-line relaxation when the iteration number is odd and η-line relaxation when the iteration number is even. The asymptotic convergence rate is not much slower for $\delta = 0.6$ than for $\delta = 0$. Figures 6–10 can also serve as a demonstration of the performance of the multigrid solver versus the corresponding single-grid solver. (The CPU time of one $V(1, 1)$-FAS cycle is roughly 3–4 times that of one relaxation swap on the finest grid.)

The streamlines and isobars are shown in Figs. 11–12. We observe that the results obtained with $\delta = 0.2$ are in good accordance with those obtained on the uniform grid ($\delta = 0$). Even for $\delta = 0.4$ and 0.6 the results are acceptable (some oscillations of the isobars can be observed for $\delta = 0.4$ and 0.6; they correspond to small pressure values).

The second numerical example consists of non-square cavities. The boundary condition is the same as in the first example. Three problems are considered. The domains and corresponding grids are shown in Fig. 13, and Fig. 14 shows the streamlines obtained for $R = 2000$ on 97×97 grids. The number of multigrid iterations needed to reduce the total residual below 10^{-4} is respectively 121, 59, and 98 for the three cases, using $V(1, 1)$ cycle and box relaxation (the underrelaxation factor is set to 0.55).

The last numerical example is the flow between rotating non-concentric cylinders, as shown in Fig. 15. The domain is bounded by two circles $x^2 + y^2 = 5^2$ and $(x-1)^2 + y^2 = 2^2$. The angular velocity of the inner cylinder is 1 and the angular velocity of the outer cylinder is -1. The streamlines shown in Fig. 15 are obtained on a 321×65 grid for $R = 50$. The underrelaxation factor is set to 0.5 and 50 multigrid iterations are needed to reduce the residual below 10^{-4}.

Finally, to compare the extra amount of work introduced in the present scheme, we give in Table I the CPU time of one iteration on a 65×65 grid using box relaxation, evaluated on a personal computer (33 MHz 80386/80387). In the table, two numbers can be found in each entry; the first one corresponds to the CPU time needed when only one velocity component is defined at each grid surface (the traditional MAC grid), while the second one corresponds to the CPU time needed by the present scheme. It is clear from this table that the extra amount of work introduced is about 26% if the metric derivatives (x_ξ, x_η, etc.) are precalculated and stored, and 14% if the metric derivatives are re-calculated during each iteration.

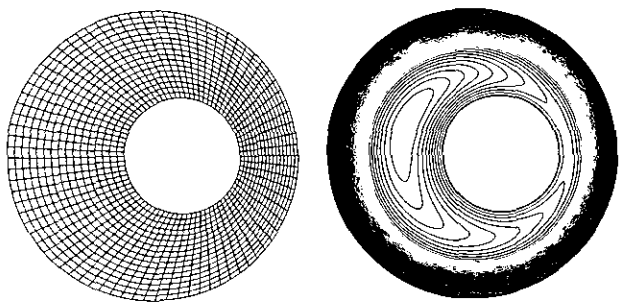

FIG. 15. Test problem 3—the grid and streamlines.

6. CONCLUSIONS

A multigrid solver for the steady incompressible Navier–Stokes equations is proposed. This solver emphasizes on the symmetric distribution of velocity unknowns and thus eliminates the effect of grid orientation on the convergence rate and numerical accuracy (this point is evident from the scheme itself). The numerical results

presented here have no practical interest and little significance about the ability of the solver for handling general flow problems on general domains; they only serve as an indication on the convergence rate and accuracy of the solver on skew, non-orthogonal grids. Further numerical experiences with this solver are needed.

We conclude this paper by pointing out that the staggered distribution of unknowns used in the present solver is also applicable to the pressure correction methods. This can be done in the following way: in the pressure correction step, only the contravariant velocity components are updated, while in the solution of the momentum equations for a given pressure field, both u and v are calculated at all grid surfaces. The extra amount of work introduced is not significant because the linearized momentum equations for u and v have the same coefficient matrices.

REFERENCES

1. A. Brandt, *Monograph GMD-Studien 85, (GMD, St. Augustin, 1984).*
2. A. Brandt and N. Dinar, "Multigrid Solutions to Elliptic Flow Problems," in *Numerical Methods for PDEs*, edited by S. V. Parter (Academic Press, New York/London, 1984).
3. S. P. Vanka, *J. Comput. Phys.* **65**, 138 (1986).
4. S. P. Vanka, B. C.-J. Chen, and W. T. Sha, *Numer. Heat Transfer* **3**, 1 (1980).
5. W. Shyy, S. S. Tong, and S. M. Correa, *Numer. Heat Transfer* **8**, 99 (1985).
6. M. Braaten and W. Shyy, *Numer. Heat Transfer* **9**, 559 (1986).
7. W. Shyy and T. C. Vu, *J. Comput. Phys.* **92**, 82 (1991).
8. H. Q. Yang, K. T. Yang, and J. R. Lloyd, *Int. J. Numer. Methods Eng.* **25**, 231 (1988).
9. D. Lee and Y. M. Tusei, *J. Comput. Phys.* **98**, 90 (1992).
10. J. Linden, B. Steckel, and K. Stüben, "Parallel Multigrid Solution of the Navier–Stokes Equations on General 2D Domains," in "Parallel Computing 1988" (North Holland, Amsterdam, 1989).
11. J. Linden, G. Lonsdale, B. Steckel, and K. Stüben, "Multigrid for the Steady-State Incompressible Navier–Stokes Equations: A survey," Lec. Notes in Phys., Vol. 323 (Springer-Verlag, New York/Berlin, 1989).
12. L.-B. Zhang, *Math. Modelling Numer. Anal.* **24** (1), 133 (1990).
13. L.-B. Zhang, *Proceedings 4th Symp. on CFD, Davis, CA, 1991* (unpublished).