INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 20, 59–74 (1995)

AN ILU SMOOTHER FOR THE INCOMPRESSIBLE NAVIER–STOKES EQUATIONS IN GENERAL CO-ORDINATES

S. ZENG AND P. WESSELING

Faculty of Technical Mathematics and Informatics, Delft University of Technology, Mekelweg 4, NL-2628 CD Delft, Netherlands

SUMMARY

The solution of the incompressible Navier–Stokes equations in general two- and three-dimensional domains using a multigrid method is considered. Because a great variety of boundary-fitted grids may occur, robustness is at a premium. Therefore a new ILU smoother called CILU (collective ILU) is described, based on r-transformations. In CILU the matrix that is factorized is block-structured, with blocks corresponding to the set of physical variables. A multigrid algorithm using CILU as smoother is investigated. The performance of the algorithm in two and three dimensions is assessed by numerical experiments. The results show that CILU is a good smoother for the incompressible Navier–Stokes equations discretized on general non-orthogonal curvilinear grids.

KEY WORDS Navier–Stokes equations Multigrid method Smoothing method ILU factorization General co-ordinates

1. INTRODUCTION

Theoretical and practical investigations for about two decades have shown that multigrid methods are very suitable for solving large systems of algebraic equations resulting from discretization of partial differential equations. Application of multigrid principles has led to efficient solution methods for the incompressible Navier–Stokes equations discretized on Cartesian grids in two dimensions (see e.g. References 1–3). In this paper we present a method for general curvilinear grids in two or three dimensions.

The main components in a multigrid algorithm are smoothing and coarse grid correction. The smoother should possess the smoothing property and the coarse grid approximation should have the approximation property.[4] Because in general curvilinear co-ordinates mixed derivatives and highly stretched non-uniform grids may occur, the method should be robust. Our purpose here is to develop a robust smoother. Coarse grid approximation, consisting of the choice of prolongation, restriction and the use of Galerkin coarse grid approximation, is discussed elsewhere.[5–7] Given these ingredients, our multigrid algorithm is standard.

The discretization of the incompressible Navier–Stokes equations on a staggered general curvilinear grid in two and three dimensions is a complicated affair; for details see References 8–12. Hence from the point of view of software development it is very attractive to separate the discretization and solution phases. Therefore we use the linear multigrid method inside an outer Newton iteration, together with Galerkin coarse grid approximation (GCA), so that changes in the discretization leave the solver code completely unaffected. In Reference 13 the smoothing

CCC 0271–2091/95/010059–16

Received 27 April 1993
Revised 15 November 1993

and approximation properties are studied for the incompressible Navier–Stokes equations discretized on a staggered grid in Cartesian co-ordinates. In general co-ordinates a theory is not available. Therefore the performance of our method is tested by numerical experiments.

For the convection–diffusion equation with dominating convection, and hence also for the Navier–Stokes equations with a high Reynolds number, coarse grid approximation presents problems, discussed in References 7, 14 and 15. Galerkin coarse grid approximation may give an inaccurate coarse grid correction. Our approach is to compensate this by use of a strong smoother. Smoothers of ILU type are found to be suitable. For an assessment of ILU smoothing see Reference 16.

We will use a coupled smoothing method, smoothing the velocity components and the pressure simultaneously. A well-known coupled smoothing method for the Navier–Stokes equations is the SCGS (symmetrical coupled Gauss–Seidel) method of Vanka,[3] which works well in Cartesian co-ordinates, including the case of highly stretched meshes, provided that the line version (see e.g. Reference 17) of Reference 3 is used. However, in general co-ordinates mixed derivatives occur and this method works less well. Therefore we use coupled or collective ILU (CILU). This method has not been used before. We found this to work better than uncoupled ILU.

Discrete systems approximating the incompressible Navier–Stokes equations are indefinite, so direct implementation of ILU decompositions is problematic. This problem is overcome by applying a so-called r-transformation as proposed in References 13 and 18–20.

2. PARTIAL DIFFERENTIAL EQUATIONS AND DISCRETIZATION

The tensor formulation of the incompressible Navier–Stokes equations in general co-ordinates is given by

$$U^{\alpha}_{,\alpha} = 0, \tag{1}$$

$$\frac{\partial}{\partial t}(U^{\alpha}) + (U^{\alpha}U^{\beta})_{,\beta} + (g^{\alpha\beta}p)_{,\beta} - \tau^{\alpha\beta}_{,\beta} = B^{\alpha}, \tag{2}$$

where $\tau^{\alpha\beta}$ is the deviatoric stress tensor given by

$$\tau^{\alpha\beta} = Re^{-1}(g^{\alpha\gamma}U^{\beta}_{,\gamma} + g^{\gamma\beta}U^{\alpha}_{,\gamma}), \tag{3}$$

with Re the Reynolds number, p is the pressure, t is the time, U^{α}, $\alpha = 1, 2, \ldots, nd$, are the contravariant components of velocity, with nd the number of space dimensions, and B^{α} is the contravariant component of the body force. U^{α} and B^{α} are derived from their physical counterparts $\mathbf{u}$ and $\mathbf{b}$ through the contravariant base vectors $\mathbf{a}^{(\alpha)}$ of the general co-ordinates by

$$U^{\alpha} = \mathbf{a}^{(\alpha)} \cdot \mathbf{u}, \qquad B^{\alpha} = \mathbf{a}^{(\alpha)} \cdot \mathbf{b}. \tag{4}$$

Furthermore, $g^{\alpha\beta}$ is the metric tensor given by $g^{\alpha\beta} = \mathbf{a}^{(\alpha)} \cdot \mathbf{a}^{(\beta)}$. For better accuracy the variable $V^{\alpha} = \sqrt{g}U^{\alpha}$ is used instead of U^{α}, where $\sqrt{g}$ is the Jacobian of the mapping; this is motivated in References 8, 11 and 12.

Equations (1) and (2) are discretized in general co-ordinates on a staggered grid by using the finite volume method. Central differencing is used for space discretization and the so-called θ-method is used for time discretization. As mentioned before, discretization itself is difficult and

so here we do not want to get into details, for which we refer to References 8–12. The discrete system can be written as

$$\frac{1}{\Delta t}\mathbf{V}^{n+1} + \theta\mathbf{Q}'(\mathbf{V}^{n+1}) + \theta\mathbf{G}\mathbf{p}^{n+1} = \mathbf{f}'_v, \qquad \mathbf{D}\mathbf{V}^{n+1} = \mathbf{f}_c, \tag{5}$$

with

$$\mathbf{f}'_v = \theta\mathbf{B}^{n+1} + (1-\theta)\mathbf{B}^n + \frac{1}{\Delta t}\mathbf{V}^n - (1-\theta)\mathbf{Q}'(\mathbf{V}^n) - (1-\theta)\mathbf{G}\mathbf{p}^n. \tag{6}$$

Here **V**, **B** and **p** denote algebraic vectors containing the velocity unknowns, the right-hand-side grid function and the pressure unknowns respectively. The superscript n indicates the time level. The operator $\mathbf{Q}'$ is non-linear owing to the convection terms. The parameter $\theta = 1$ in the numerical experiments here, which gives the backward Euler method. We assume that the arbitrary physical domain is mapped onto a square or cubic computational domain, as illustrated in Figure 1, by means of boundary-fitted co-ordinates. The underlying ordering of the unknowns is

$$V_1^1, V_2^1, \ldots, V_N^1, V_1^2, V_2^2, \ldots, V_N^2, p_1, p_2, \ldots, p_N, \tag{7}$$

with some ordering (e.g. natural) of the grid points and $N = n_1 \times n_2$, where n_i, $i = 1, 2$, is the number of cells (including virtual cells) in direction i (we specialize temporarily to two dimensions for simplicity). This will be called blockwise ordering. Excluding virtual grid points, the grid for equation m is denoted as $\mathcal{G}^m$, with $m = 1, 2, 3$ standing for the momentum equations in direction 1 and direction 2 and the pressure equation respectively.

Equation (5) gives rise to a sequence of systems of equations for a sequence of time levels. Since in the formulation of coarse grid operators we use GCA, which in principle is limited to linear problems, and (more importantly) because we wish to separate the discretization and solution phases, the non-linear system (5) has to be linearized. This implies that linearization takes place outside the multigrid algorithm. Linearization employs Newton's method; for example, for a typical non-linear term $(U^\alpha U^\beta)^{n+1}$ in $\mathbf{Q}'$,

$$(U^\alpha U^\beta)^{n+1} = (U^\alpha)^{n+1}(U^\beta)^n + (U^\alpha)^n(U^\beta)^{n+1} - (U^\alpha U^\beta)^n. \tag{8}$$

This splits $\mathbf{Q}'$ into a linear part and a non-linear part and gives $\mathbf{Q}'(\mathbf{V}^{n+1}) = \mathbf{Q}_1\mathbf{V}^{n+1} + \mathbf{Q}_2(\mathbf{V}^n)$,

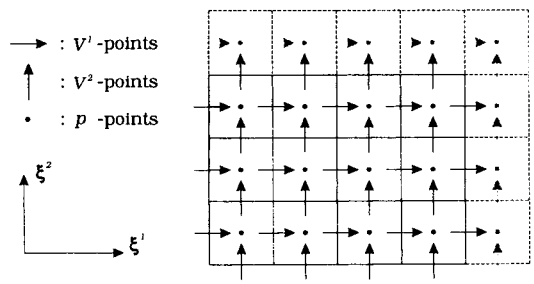


Figure 1. Staggered grid in computational domain. Virtual cells are used, which are indicated by dashed lines

with $\mathbf{Q}_1$ linear. Note that both $\mathbf{Q}_1$ and $\mathbf{Q}_2$ are evaluated by using $\mathbf{V}^n$. The resulting system is denoted by

$$\mathbf{Kx} = \mathbf{f}, \tag{9}$$

with

$$\mathbf{K} = \begin{pmatrix} \mathbf{Q} & \theta\mathbf{G} \\ \mathbf{D} & 0 \end{pmatrix}, \qquad \mathbf{x} = \begin{pmatrix} \mathbf{V}^{n+1} \\ \mathbf{p}^{n+1} \end{pmatrix}, \qquad \mathbf{f} = \begin{pmatrix} \mathbf{f}_v \\ \mathbf{f}_c \end{pmatrix}, \tag{10}$$

where

$$\mathbf{Q} = \frac{1}{\Delta t}\mathbf{I} + \theta\mathbf{Q}_1, \qquad \mathbf{f}_v = \mathbf{f}'_v - \theta\mathbf{Q}_2(\mathbf{V}^n). \tag{11}$$

For convenience let the parameter θ be absorbed into $\mathbf{G}$, i.e. $\theta\mathbf{G}$ will be simply written as $\mathbf{G}$ hereafter. If there exists a stationary solution, it satisfies

$$\mathbf{K}_s\mathbf{x} = \mathbf{f}_s, \tag{12}$$

with

$$\mathbf{K}_s = \begin{pmatrix} \mathbf{Q}' & \mathbf{G} \\ \mathbf{D} & 0 \end{pmatrix}, \qquad \mathbf{f}_s = \begin{pmatrix} \mathbf{B} \\ \mathbf{f}_c \end{pmatrix}. \tag{13}$$

3. THE R-TRANSFORMATION

3.1. Iteration with r-transformation

A classical iteration method for solving (9) is given by

$$\mathbf{x}^{i+1} = \mathbf{x}^i - \mathbf{M}^{-1}(\mathbf{K}\mathbf{x}^i - \mathbf{f}), \tag{14}$$

with $\mathbf{M}$ a splitting of $\mathbf{K}$:

$$\mathbf{K} = \mathbf{M} - \mathbf{N}. \tag{15}$$

This method converges if the splitting is regular.[21] The zero block in $\mathbf{K}$ makes a regular splitting impossible. A remedy is to introduce a matrix $\bar{\mathbf{K}}$ such that a regular splitting of the product $\mathbf{K}\bar{\mathbf{K}}$, i.e.

$$\mathbf{K}\bar{\mathbf{K}} = \mathbf{M} - \mathbf{N}, \tag{16}$$

is easy to find. This implies a splitting of $\mathbf{K}$ as

$$\mathbf{K} = \mathbf{M}\bar{\mathbf{K}}^{-1} - \mathbf{N}\bar{\mathbf{K}}^{-1}, \tag{17}$$

resulting in the iterative method

$$\mathbf{x}^{i+1} = \mathbf{x}^i - \bar{\mathbf{K}}\mathbf{M}^{-1}(\mathbf{K}\mathbf{x}^i - \mathbf{f}). \tag{18}$$

In Reference 22 $\bar{\mathbf{K}}$ is called a distributive operator and iteration (18) is called distributive iteration; in Reference 18 $\bar{\mathbf{K}}$ is called r-transformation and the iteration is called transforming iteration.

Here we adopt the latter terminology. Many iterative methods can be fitted into the framework of (16) and (18), such as the SIMPLE method of Patankar and Spalding[23] and its variants and the DGS method of Brandt and Dinar[24] (see e.g. References 4, 13, 18–20 and 25).

3.2. *Construction of r-transformation*

A theory of constructing smoothers with r-transformation is given in Reference 13. Some applications to the Stokes and the Navier–Stokes equations can be found in References 18 and 19. We summarize some results. For use as smoother in a multigrid method, an iterative method must have the smoothing property defined in Reference 4; see Reference 25 for an elementary introduction. In Reference 13 it is shown generally that if $\mathbf{K}\bar{\mathbf{K}}$ is of the block triangular form

$$\mathbf{K}\bar{\mathbf{K}} = \begin{pmatrix} \mathbf{A} & 0 \\ \mathbf{B} & \mathbf{C} \end{pmatrix} \tag{19}$$

and can be split regularly into $\mathbf{M} - \mathbf{N}$, then the smoothing property holds for system (19) if the iteration matrix

$$\mathbf{S} = \mathbf{M}^{-1}\mathbf{N}, \tag{20}$$

with

$$\mathbf{S} = \begin{pmatrix} \mathbf{S}_{11} & 0 \\ \mathbf{S}_{21} & \mathbf{S}_{22} \end{pmatrix}, \tag{21}$$

has the smoothing property for its diagonal blocks. Hence the study of the smoothing property for systems is essentially reduced to the study of the smoothing property for single equations. Therefore it is attractive to choose $\bar{\mathbf{K}}$ such that $\mathbf{K}\bar{\mathbf{K}}$ has the block triangular form (19). A possible choice for $\bar{\mathbf{K}}$ is

$$\bar{\mathbf{K}} = \begin{pmatrix} \mathbf{I} & \mathbf{K}_{12} \\ 0 & \mathbf{K}_{22} \end{pmatrix}. \tag{22}$$

Then we have

$$\mathbf{K}\bar{\mathbf{K}} = \begin{pmatrix} \mathbf{Q} & \mathbf{Q}\mathbf{K}_{12} + \mathbf{G}\mathbf{K}_{22} \\ \mathbf{D} & \mathbf{D}\mathbf{K}_{12} \end{pmatrix}. \tag{23}$$

Choosing $\mathbf{K}_{12}$ and $\mathbf{K}_{22}$ such that $\mathbf{Q}\mathbf{K}_{12} + \mathbf{G}\mathbf{K}_{22} = 0$ results in the form given in (19).

There are many possibilities for choosing $\bar{\mathbf{K}}$. Wittum's theory gives us a guide. In (23) we do not have problems in constructing a smoother for $\mathbf{Q}$ provided that the discretization is appropriate ($\mathbf{Q}$ should be an M-matrix). What we should do then is choose $\bar{\mathbf{K}}$ such that the smoothing property exists also for the block $\mathbf{D}\mathbf{K}_{12}$.

Choosing

$$\bar{\mathbf{K}} = \begin{pmatrix} \mathbf{I} & -\mathbf{Q}^{-1}\mathbf{G}\mathbf{E}^{-1}\mathbf{F} \\ 0 & \mathbf{E}^{-1}\mathbf{F} \end{pmatrix}, \tag{24}$$

with $\mathbf{E} = \mathbf{D}\mathbf{Q}^{-1}\mathbf{G}$, results in

$$\mathbf{K}\bar{\mathbf{K}} = \begin{pmatrix} \mathbf{Q} & 0 \\ \mathbf{D} & -\mathbf{F} \end{pmatrix}, \tag{25}$$

where $\mathbf{F}$ is still to be chosen. We discretize (1) and (2) with central differences. As a consequence, $\mathbf{Q}$ is not an M-matrix for Re sufficiently large (approximately $Re > 2/h$ in our examples, with h the local mesh size). If we choose $-\mathbf{F}$ to be an M-matrix, then it will be easy (for Re small enough) to obtain a smoother for the product system $\mathbf{K}\bar{\mathbf{K}}$ as discussed before. The first choice is $\mathbf{F} = \mathbf{DG}$, corresponding to the distributed Gauss–Seidel method of Reference 24.

The second choice is $\mathbf{F} = \mathbf{E}$, giving

$$\bar{\mathbf{K}} = \begin{pmatrix} \mathbf{I} & -\mathbf{Q}^{-1}\mathbf{G} \\ 0 & \mathbf{I} \end{pmatrix}, \tag{26}$$

leading to iterative methods of the so-called SIMPLE type.[23] This gives

$$\mathbf{K}\bar{\mathbf{K}} = \begin{pmatrix} \mathbf{Q} & 0 \\ \mathbf{D} & -\mathbf{D}\mathbf{Q}^{-1}\mathbf{G} \end{pmatrix}. \tag{27}$$

For practical purposes $\mathbf{E}^{-1}$ in (24) and $\mathbf{Q}^{-1}$ in (26) and (27) have to be approximated.

4. CILU DECOMPOSITION

4.1. *Approximation of* $\bar{\mathbf{K}}$ *and* $\bar{\mathbf{K}}$

Temporarily using the blockwise ordering (7), we choose the r-transformation (cf. (26))

$$\bar{\mathbf{K}} = \begin{pmatrix} \mathbf{I} & -\mathbf{Q}^{-1}\mathbf{G} \\ 0 & \zeta\mathbf{I} \end{pmatrix}, \tag{28}$$

where the parameter ζ will be used to enhance multigrid convergence. This gives

$$\mathbf{K}\bar{\mathbf{K}} = \begin{pmatrix} \mathbf{Q} & (\zeta - 1)\mathbf{G} \\ \mathbf{D} & -\mathbf{D}\mathbf{Q}^{-1}\mathbf{G} \end{pmatrix}. \tag{29}$$

Because $\mathbf{Q}^{-1}$ is not readily available, we approximate $\mathbf{Q}$ by $\tilde{\mathbf{Q}} = diag(\mathbf{Q})$. This gives an approximation $\tilde{\mathbf{K}}$ of $\bar{\mathbf{K}}$ and an approximation $\widetilde{\mathbf{K}\bar{\mathbf{K}}}$ of $\mathbf{K}\bar{\mathbf{K}}$:

$$\tilde{\mathbf{K}} = \begin{pmatrix} \mathbf{I} & -\tilde{\mathbf{Q}}^{-1}\mathbf{G} \\ 0 & \zeta\mathbf{I} \end{pmatrix}, \qquad \widetilde{\mathbf{K}\bar{\mathbf{K}}} = \begin{pmatrix} \mathbf{Q} & (\zeta - 1)\mathbf{G} \\ \mathbf{D} & -\mathbf{D}\tilde{\mathbf{Q}}^{-1}\mathbf{G} \end{pmatrix}. \tag{30}$$

The matrix $\widetilde{\mathbf{K}\bar{\mathbf{K}}}$ looks like the original matrix $\mathbf{K}$ with the zero block replaced by a non-zero block $-\mathbf{D}\tilde{\mathbf{Q}}^{-1}\mathbf{G}$, which is easy to formulate.

4.2. *Incomplete decomposition*

The residual amplification matrix of (18) is $\mathbf{I} - \mathbf{K}\bar{\mathbf{K}}\mathbf{M}^{-1}$. Hence $\mathbf{M}$ should be close to $\mathbf{K}\bar{\mathbf{K}}$ but easily invertible. Since in practice $\mathbf{K}\bar{\mathbf{K}}$ is approximated by $\widetilde{\mathbf{K}\bar{\mathbf{K}}}$, we require that $\mathbf{M}$ be close to $\widetilde{\mathbf{K}\bar{\mathbf{K}}}$. With the approximation $\tilde{\mathbf{K}}$ of $\bar{\mathbf{K}}$ the iteration (18) becomes

$$\mathbf{x}^{i+1} = \mathbf{x}^i - \tilde{\mathbf{K}}\mathbf{M}^{-1}(\mathbf{K}\mathbf{x}^i - \mathbf{f}). \tag{31}$$

This iteration is of course different from (18), since both $\bar{\mathbf{K}}$ and $\mathbf{K}\bar{\mathbf{K}}$ are approximated. Reference 13 gives conditions under which the smoothing property for the perturbed method

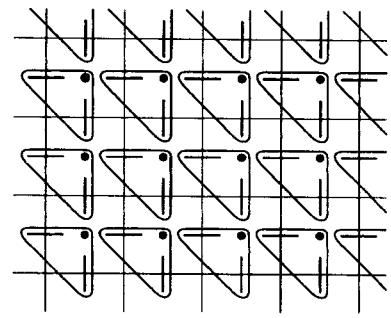


Figure 2. Collective ordering of grid points

(31) follows from the smoothing property for (18). Obviously (31) converges to the solution if it converges. Convergence may be enhanced by underrelaxation:

$$\mathbf{x}^{i+1} = \mathbf{x}^i - \omega\tilde{\mathbf{K}}\mathbf{M}^{-1}(\mathbf{K}\mathbf{x}^i - \mathbf{f}). \tag{32}$$

The choice of ω will be discussed later. Method (32) is the smoothing iteration method that we use.

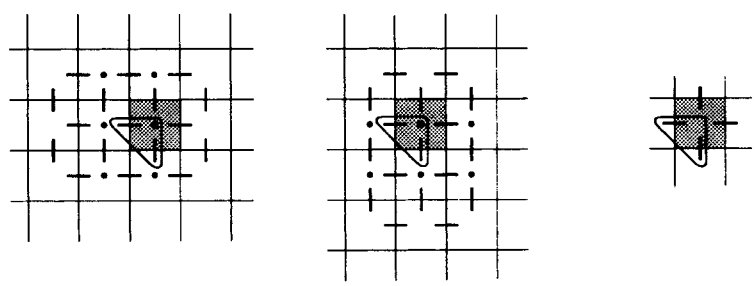
With incomplete decomposition one chooses

$$\mathbf{M} = (\mathbf{L} + \mathbf{D})\mathbf{D}^{-1}(\mathbf{D} + \mathbf{U}), \tag{33}$$

with **L** and **U** strictly lower and upper triangular matrices respectively and **D** a diagonal matrix. A possible choice for **L**, **D** and **U** is as follows. Let G be a non-zero pattern and let **L**, **D**, **U** $\neq 0$ only on G. Then we require

$$\mathbf{M}_{ij} = (\widetilde{\mathbf{K}\tilde{\mathbf{K}}})_{ij}, \quad (i, j) \in G, \tag{34}$$

from which **L**, **D** and **U** follow. It is known that the rate of convergence of the resulting iterative method depends on the ordering of the unknowns. We will number the cells in natural order. The V^1-unknown in the left cell face, V^2 in the lower face and p in the centre are grouped together in a 3-vector $\mathbf{u}_i = (V^1, V^2, p)_i$, with i the number of the cell (see Figure 2). Extension to three dimensions is straightforward. Because of this collective treatment of the three unknowns, we call the resulting method collective ILU (CILU) decomposition. This collective ordering induces a 3×3 block matrix representation of $\widetilde{\mathbf{K}\tilde{\mathbf{K}}}$. With our discretization the momentum equations and the continuity equation have the structures shown in Figure 3. The



V^1-momentum equation V^2-momentum equation Continuity equation

Figure 3. Structures of the discrete equations at cell i

grid points that have index i are grouped together. A typical row in $\mathbf{K}$, say number i, has non-zero elements at positions $(i, i \pm I \pm 1, i \pm I, i \mp I \pm 1, i \pm 1, i + I - 2, i - 2, i - 2I, i - 2I + 1)$, where I is the number of cells in the ξ^1-direction. The stencil of $\mathbf{K}$ has the structure

$$[\mathbf{K}] = \begin{bmatrix} * & * & * & * \\ * & * & \underline{*} & * \\ & * & * & * \\ & & * & * \end{bmatrix}, \tag{35}$$

and so does the matrix $\widetilde{\mathbf{K}\bar{\mathbf{K}}}$, since $\mathbf{D}\tilde{\mathbf{Q}}^{-1}\mathbf{G}$ has a nine-point stencil, which does not change the structure of $\mathbf{K}$. Here each * represents a 3×3 matrix. The * with an underscore corresponds to cell number i. We choose the non-zero pattern $G = (i, i \pm I \pm 1, i \pm I, i \mp I \pm 1, i \pm 1)$ in the approximate factorization.

5. TEST PROBLEMS AND RESULTS

5.1. *Multigrid algorithm*

To investigate the multigrid algorithm with CILU as smoother, we apply the method to three test problems. In the first two test problems, starting from the finest grid from the zero solution, two time steps, each accompanied by two (multigrid) iterations, are performed first to give a more or less representative flow field. Then we test the multigrid solver by carrying out 20 multigrid iterations, in which we measure the average reduction factor and the limiting reduction factor, which are to be defined shortly. Furthermore, the steady solution (obtained for t large enough) is computed, using one multigrid iteration per time step. The multigrid algorithm uses the W-cycle, with one pre- and one post-smoothing. The coarsest grid is fixed at 2×2. Coarse grid operators are formulated by means of Galerkin coarse grid approximation; detailed discussions can be found in References 5–7. Bilinear interpolation and piecewise constant interpolation are used respectively for the prolongation of V^α and p. Restriction takes the adjoint of the so-called hybrid prolongation, in which for V^α, linear interpolation is used in direction α but piecewise constant interpolation in other directions; for p, piecewise constant interpolation is used.

Let $\mathbf{r} = \mathbf{f} - \mathbf{K}\mathbf{x}$ be the residual of equation (9). The norm of the residual is measured by

$$r = \|\mathbf{r}\| = \left[\frac{1}{M}\sum_{m=1}^{M}\left(\sum_{j\in\mathscr{G}_m}\frac{(\mathbf{f}-\mathbf{K}\mathbf{x})_j^2}{N_g^m}\right)\right]^{1/2}, \tag{36}$$

with M the number of unknowns and N_g^m the number of grid points in $\mathscr{G}^m$; $M = 3$ in two dimensions and $M = 4$ in three dimensions. After linearization a number of multigrid iterations are carried out, after which a time step is completed. Let r_0 be the initial norm of the residual on the finest grid and r_n be the norm of the residual on the finest grid after n multigrid iterations. The average reduction factor $\bar{\rho}_n$ is defined by

$$\bar{\rho}_n = \left(\frac{r_n}{r_0}\right)^{1/n}. \tag{37}$$

The reduction factor at the ith iteration is defined by

$$\rho_i = \frac{r_i}{r_{i-1}}. \tag{38}$$

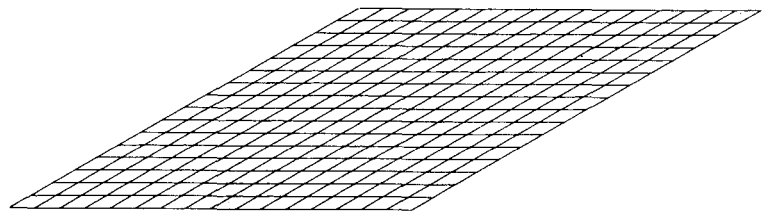


Figure 4. The skewed driven cavity problem and the grid used

If ρ_i has a limit for $i \to \infty$, then it is the asymptotic reduction factor. Let $\mathbf{r}_s = \mathbf{f}_s - \mathbf{K}_s\mathbf{x}$ be the residual of equation (12) and $r_s = \|\mathbf{r}_s\|$. A steady state is reached if

$$\frac{r_s^t}{r_s^0} \ll 1 \tag{39}$$

is satisfied, with r_s^0 being r_s at the initial time level and r_s^t being r_s at time level t.

The grids used for the second and third test problems were generated by the L$_i$SS package.[26]

5.2. *The skewed driven cavity problem*

The skewed driven cavity problem is chosen first (see Figure 4). Skewness introduces mixed derivatives. A benchmark solution for this problem is available in Reference 27, where a collocated grid is used. This problem is also solved in Reference 28 by a non-linear multigrid method for the steady case on a staggered grid. Here we do not want to solve the differential equations very accurately, since our purpose is to investigate the performance of the multigrid algorithm with the CILU smoother. In accordance with Reference 28, the Reynolds numbers will be 100 and 1000 respectively. The numbering of cells is natural. Figures 5 and 6 give the

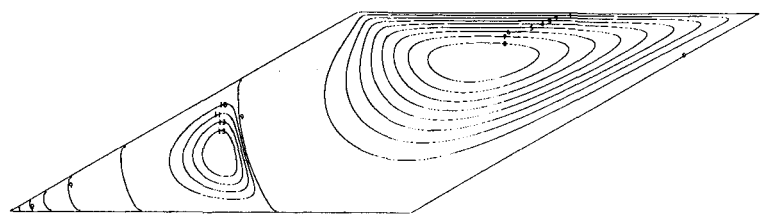


Figure 5. Streamlines for the skewed driven cavity problem, $Re = 100$, $\Delta t = 1$, 20 time steps, $r_s^t/r_s^0 < 3{\cdot}570 \times 10^{-12}$, on a 128 × 128 grid

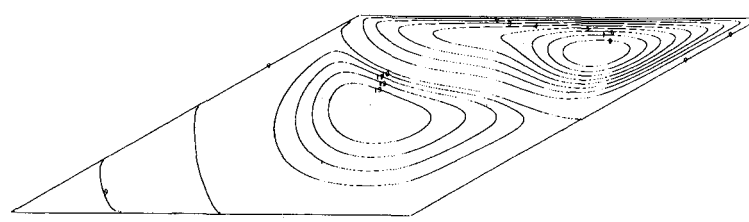


Figure 6. Streamlines for the skewed driven cavity problem, $Re = 1000$, $\Delta t = 1$, 20 time steps, $r_s^t < 6{\cdot}424 \times 10^{-5}$, on a 128 × 128 grid

Table I. Reduction factors for the skewed driven cavity problem, $\zeta = 2$

Grid	16 × 16	32 × 32	64 × 64	128 × 128
		Re = 100		
i	16	16	16	16
ρ_i	0·3466	0·3504	0·3213	0·4825
ρ_{i+1}	0·3481	0·3447	0·3506	0·4829
ρ_{i+2}	0·3493	0·3409	0·4266	0·4832
ρ_{i+3}	0·3505	0·3396	0·4573	0·4833
ρ_{i+4}	0·3515	0·3407	0·4381	0·4832
r_0	$0{\cdot}9608 \times 10^{-3}$	$0{\cdot}1270 \times 10^{-2}$	$0{\cdot}8138 \times 10^{-3}$	$0{\cdot}5088 \times 10^{-3}$
r_{i+4}	$0{\cdot}8049 \times 10^{-13}$	$0{\cdot}1306 \times 10^{-12}$	$0{\cdot}2364 \times 10^{-11}$	$0{\cdot}5613 \times 10^{-11}$
$\bar{\rho}$	0·3134	0·3167	0·3742	0·4001
		Re = *1000*		
i	16	16	16	16
ρ_i	0·4735	0·4733	0·4105	0·4396
ρ_{i+1}	0·4839	0·4724	0·4156	0·4541
ρ_{i+2}	0·5059	0·4721	0·4087	0·4643
ρ_{i+3}	0·5191	0·4713	0·4146	0·4706
ρ_{i+4}	0·5206	0·4717	0·4118	0·4741
r_0	$0{\cdot}8115 \times 10^{-3}$	$0{\cdot}1435 \times 10^{-2}$	$0{\cdot}1245 \times 10^{-2}$	$0{\cdot}5575 \times 10^{-3}$
r_{i+4}	$0{\cdot}6937 \times 10^{-9}$	$0{\cdot}2613 \times 10^{-9}$	$0{\cdot}2429 \times 10^{-10}$	$0{\cdot}2036 \times 10^{-11}$
$\bar{\rho}$	0·4973	0·4603	0·4116	0·3786

streamlines obtained after 20 time steps, which agree well with the solutions presented in References 27 and 28. Here $\zeta = 2$, the time step $\Delta t = 1$ and one multigrid iteration is performed for each time step. The underrelaxation parameter ω is fixed at 0·7 for all two-dimensional problems. Larger time steps are not used, because it is found that with a larger time step r_s^l decreases at a slower speed than with a smaller time step and therefore the termination time for achieving the same r_s^l is larger with a larger Δt. Furthermore, convergence problems arise after several time steps with larger time steps for high-Reynolds-number cases; Δt has to be taken sufficiently small in order to maintain diagonal dominance and improve smoothing. This can be seen in the next test problem. Table I presents reduction factors for five successive multigrid iterations on various grids before rounding error takes effect, taking the solution obtained after two time steps with $\Delta t = 0{\cdot}5$ as the initial solution for the linear multigrid iterations. The convergence rate is satisfactory and independent of mesh size.

The dependence of multigrid convergence on ζ is given in Table II. Apparently, ζ should not be too large. The rate of convergence is not very sensitive to ζ and good values of ζ do not depend much on *Re*.

5.3. The L-shaped driven cavity problem

This problem is proposed in Reference 28 and is illustrated in Figure 7(a). The compuational domain is depicted in Figure 7(b). The top wall moves from right to left. The well-known phenomenon appears that convergence of ILU smoothers is influenced by the numbering of grid points. The reason is explained in Reference 25 (see section 7.8 and the references cited therein) for anisotropic convection–diffusion equations. Therefore, instead of the natural ordering, a backward (lexicographical) ordering is employed, in which the numbering of cells takes

Table II. Dependence of multigrid convergence on ζ in the skewed driven cavity problem, finest grid 128 × 128

ζ, i	ρ_i	ρ_{i+1}	ρ_{i+2}	ρ_{i+3}	ρ_{i+4}	r_{i+4}	$\bar{\rho}$
			$Re = 100$, $r_0 = 0{\cdot}5088 \times 10^{-3}$				
1, 16	0·7707	0·7719	0·7726	0·7730	0·7731	$0{\cdot}6219 \times 10^{-7}$	0·6373
1·5, 16	0·4116	0·4127	0·4138	0·4148	0·4157	$0{\cdot}1752 \times 10^{-11}$	0·3774
2, 16	0·4825	0·4829	0·4832	0·4833	0·4832	$0{\cdot}5613 \times 10^{-11}$	0·4001
4, 16	0·5712	0·5729	0·5728	0·5745	0·5908	$0{\cdot}2079 \times 10^{-9}$	0·4793
5, 16	0·6841	0·7250	0·7467	0·7330	0·6975	$0{\cdot}6709 \times 10^{-8}$	0·5702
6, *				Divergence			
			$Re = 1000$, $r_0 = 0{\cdot}5575 \times 10^{-3}$				
1, *				Divergence			
1·5, 16	0·4097	0·4108	0·4118	0·4127	0·4136	$0{\cdot}8955 \times 10^{-12}$	0·3633
2, 16	0·4396	0·4541	0·4643	0·4706	0·4741	$0{\cdot}2036 \times 10^{-11}$	0·3786
4, 16	0·4619	0·4628	0·4648	0·4715	0·4812	$0{\cdot}6364 \times 10^{-10}$	0·4497
5, 16	0·6322	0·6711	0·6851	0·6760	0·6490	$0{\cdot}8905 \times 10^{-9}$	0·5131
6, *				Divergence			

place first in the reverse direction of the ξ^1-direction and then in the reverse direction of the ξ^2-direction, starting from corner D. The Reynolds numbers are 100 and 1000 respectively. Figures 8 and 9 give the streamlines for the two cases, which are in good agreement with those given in Reference 28. One multigrid iteration is employed for each time step. Note that the time steps for $Re = 100$ and 1000 are different. Compared with the time step for the skewed driven cavity problem, the time step for $Re = 1000$ has to be smaller, otherwise the multigrid algorithm fails after a few time steps because of lack of diagonal dominance, as discussed before. Of course, the time step can be larger for lower Reynolds numbers. However, we take $\Delta t = 0{\cdot}5$ for both cases in measuring reduction factors. The reduction factors and the dependence of multigrid convergence on ζ are presented in Tables III and IV.

In this test problem the parameter ζ must be greater than unity and can be rather large. It is clear that the optimal value of ζ is problem-dependent and an appropriate choice of ζ improves the multigrid performance. However, in both cases $\zeta = 2$ would give satisfactory convergence. Table III shows that the rate of convergence is independent of the mesh size.

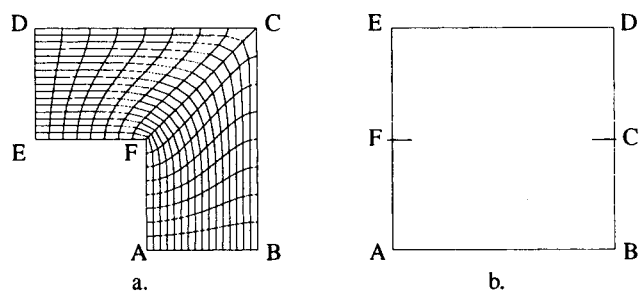


Figure 7. (a) The L-shaped driven cavity problem with a grid generated by a biharmonic grid generator. (b) The computational domain

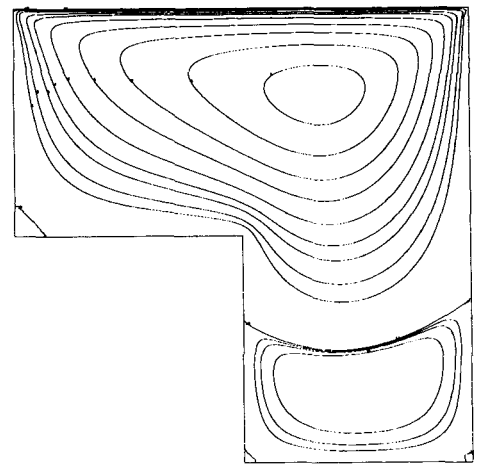


Figure 8. Streamlines for the L-shaped driven cavity problem, $Re = 100$, $\Delta t = 1$, 20 time steps, $r_s^t/r_s^0 < 1{\cdot}905 \times 10^{-9}$, on a 128 × 128 grid

5.4. *A 3D channel flow*

A further test on CILU is performed in three dimensions for a flow in a channel with a backward-facing step, as illustrated in Figure 10. The grid is uniform in direction 3 and biharmonic in plane 1–2. Here we consider only $Re = 100$. The final time is 20 and the time interval $\Delta t = 0{\cdot}25$ is chosen based on numerical experiments. For the sake of saving computation time, the F-cycle with one pre- and one post-smoothing is employed. The underrelaxation factor $\omega = 0{\cdot}8$ and the parameter $\zeta = 2$. At each time step one multigrid iteration is performed. CILU is generalized to three dimensions by using Gauss–Seidel in the third direction. In a smoothing sweep, planes parallel to plane 1–2 are visited successively along direction 3, with CILU applied

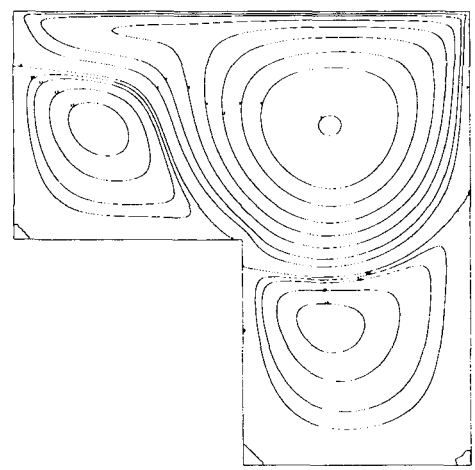


Figure 9. Streamlines for the L-shaped driven cavity problem, $Re = 1000$, $\Delta t = 0{\cdot}2$, 100 time steps, $r_s^t/r_s^0 < 1{\cdot}172 \times 10^{-4}$, on a 128 × 128 grid

Table III. Reduction factors for the L-shaped driven cavity problem, $\zeta = 2$

Grid	16 × 16	32 × 32	64 × 64	128 × 128
		Re = 100		
i	16	16	16	16
ρ_i	0·3256	0·3338	0·3392	0·5285
ρ_{i+1}	0·3274	0·3360	0·3418	0·2465
ρ_{i+2}	0·3290	0·3377	0·3438	0·2215
ρ_{i+3}	0·3303	0·3392	0·3453	0·4931
ρ_{i+4}	0·3313	0·3404	0·3465	0·3166
r_0	$0{\cdot}3096 \times 10^{-3}$	$0{\cdot}3104 \times 10^{-3}$	$0{\cdot}3076 \times 10^{-3}$	$0{\cdot}4837 \times 10^{-3}$
r_{i+4}	$0{\cdot}9552 \times 10^{-14}$	$0{\cdot}1471 \times 10^{-13}$	$0{\cdot}1531 \times 10^{-13}$	$0{\cdot}1873 \times 10^{-13}$
$\bar{\rho}$	0·2982	0·3046	0·3054	0·3016
		Re = 1000		
i	16	16	16	16
ρ_i	0·3126	0·2737	0·3200	0·3384
ρ_{i+1}	0·3143	0·3098	0·3255	0·3395
ρ_{i+2}	0·3156	0·3135	0·3312	0·3405
ρ_{i+3}	0·3164	0·2879	0·3360	0·3414
ρ_{i+4}	0·3169	0·2834	0·3391	0·3423
r_0	$0{\cdot}1208 \times 10^{-3}$	$0{\cdot}1859 \times 10^{-3}$	$0{\cdot}1438 \times 10^{-3}$	$0{\cdot}9281 \times 10^{-4}$
r_{i+4}	$0{\cdot}8155 \times 10^{-15}$	$0{\cdot}5311 \times 10^{-14}$	$0{\cdot}1032 \times 10^{-13}$	$0{\cdot}1025 \times 10^{-13}$
$\bar{\rho}$	0·2764	0·2970	0·3110	0·3178

Table IV. Dependence of multigrid convergence on ζ in the L-shaped driven cavity problem, finest grid 128 × 128

ζ, i	ρ_i	ρ_{i+1}	ρ_{i+2}	ρ_{i+3}	ρ_{i+4}	r_{i+4}	$\bar{\rho}$
			$Re = 100$, $r_0 = 0{\cdot}4837 \times 10^{-3}$				
1, *				Divergence			
1·5, 16	0·6650	0·6855	0·6890	0·7198	0·7366	$0{\cdot}1207 \times 10^{-8}$	0·5246
2, 16	0·5285	0·2465	0·2215	0·4931	0·3166	$0{\cdot}1873 \times 10^{-13}$	0·3016
4, 10	0·1919	0·1951	0·1934	0·1934	0·1978	$0{\cdot}2582 \times 10^{-13}$	0·1846
6, 10	0·1504	0·1550	0·1566	0·1561	0·1554	$0{\cdot}1940 \times 10^{-14}$	0·1534
10, 10	0·1308	0·1358	0·1406	0·1454	0·1685	$0{\cdot}3214 \times 10^{-15}$	0·1350
20, 10	0·1476	0·1500	0·1523	0·1553	0·1933	$0{\cdot}2704 \times 10^{-15}$	0·1333
			$Re = 1000$, $r_0 = 0{\cdot}9281 \times 10^{-4}$				
1, *				Divergence			
1·5, 16	0·6405	0·7172	0·7516	0·7658	0·7729	$0{\cdot}9667 \times 10^{-11}$	0·4476
2, 16	0·3384	0·3395	0·3405	0·3414	0·3423	$0{\cdot}1025 \times 10^{-13}$	0·3178
4, 14	0·2188	0·2196	0·2202	0·2215	0·2232	$0{\cdot}5794 \times 10^{-16}$	0·2099
6, 13	0·1840	0·1842	0·1845	0·1846	0·2058	$0{\cdot}1683 \times 10^{-16}$	0·1780
10, 12	0·1730	0·1728	0·1726	0·1730	0·1856	$0{\cdot}2123 \times 10^{-16}$	0·1622
20, 12	0·1787	0·1786	0·1785	0·1788	0·1910	$0{\cdot}2378 \times 10^{-16}$	0·1633

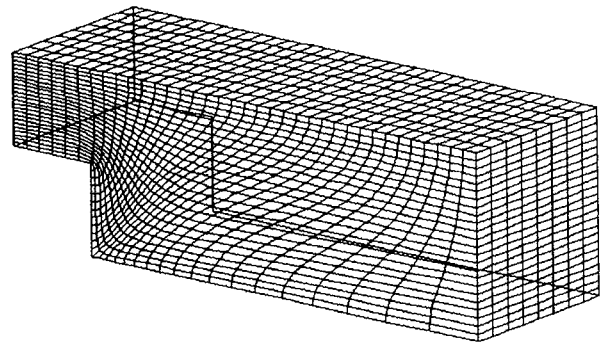


Figure 10. The channel with a backward-facing step and a 24 × 38 × 8 grid

to each plane. On the coarsest grid 10 sweeps of CILU smoothing are carried out to solve the equation system. As time marches, the reduction factor gradually approaches a constant value. In Table V we present the reduction factor on two grids at the final time, as well as the ratio r_s^{20}/r_s^0. The rate of convergence is very satisfactory. Figure 11 shows two particle traces obtained on the 24 × 36 × 8 grid. We clearly see that the flow is very different from that obtained in two dimensions,[29] where the flow in the backward-facing step forms a closed recirculation region.

Table V. Reduction factors $\bar{\rho}$ and ratios r_s^{20}/r_s^0

Grid	$\bar{\rho}$	r_s^{20}/r_s^0
12 × 18 × 4	0·0489	$0{\cdot}7457 \times 10^{-4}$
24 × 36 × 8	0·1118	$0{\cdot}9494 \times 10^{-5}$

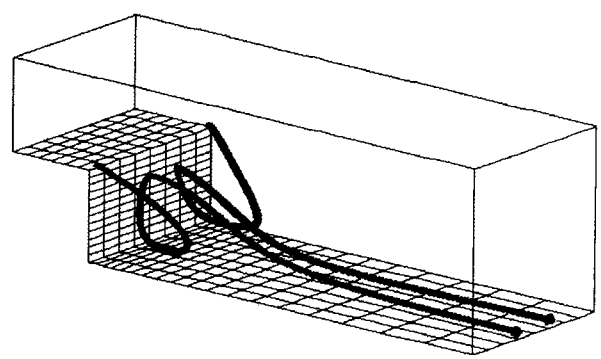


Figure 11. Two particle traces in the channel flow on the 24 × 38 × 8 grid

6. CONCLUSIONS

Based on collective incomplete LU factorization with r-transformation, a new smoother called CILU is presented for the incompressible Navier–Stokes equations in general co-ordinates. Apart from the underrelaxation factor ω, another parameter ζ is introduced to enhance smoothing. A multigrid algorithm using CILU as smoother is investigated numerically, using the skewed driven cavity and the L-shaped driven cavity as test problems. The performance of the multigrid algorithm is studied by measuring reduction factors on various grids.

The numerical experiments show that the reduction factors are almost independent of mesh sizes and slightly dependent on the Reynolds number.

The effect of parameter ζ is investigated on 128×128 grids. The results show that a proper choice of ζ improves the multigrid performance, sometimes very much as in the L-shaped driven cavity problem. The optimal value of ζ is problem-dependent, but a fixed choice $\zeta = 2$ seems to be a good compromise. (Some numerical tests on convergence dependence on the underrelaxation parameter ω and comparisons of performance between CILU and other smoothers can be found in Reference 10. The results for various problems show that it is possible to fix ω at a certain value, say 0·8.)

The well-known anisotropy of ILU smoothers is encountered here in the L-shaped driven cavity problem, where meshes are stretched more in one direction than in another. This problem is cured by simply changing the ordering of cells.

Owing to central differencing of the partial differential equations, the time step should be sufficiently small for high Reynolds numbers. Otherwise the algorithm may fail after several time steps.

Numerical experiments are carried out in three dimensions for a flow in a channel with a backward-facing step. Satisfactory results are achieved.

To sum up, the CILU smoother is a good smoother.

ACKNOWLEDGEMENTS

The authors would like to thank A. Segal and C. G. M. Kassels for their efforts in making the ISNaS code for discretization on staggered grids in general co-ordinates available for this computation. The authors are also grateful to C. W. Oosterlee and E. Brakkee for providing test problems and grids as well as some useful discussions.

REFERENCES

1. P. H. Gaskell and N. G. Wright, 'A multigrid algorithm for the investigation of thermal recirculating fluid flow problems', in R. W. Lewis, K. Morgan and W. G. Habashi (eds), *Numerical Methods in Thermal Problems*, Vol. 5, Pineridge, Swansea, 1987, pp. 1194–1215.
2. C. P. Thompson, G. K. Leaf and S. P. Vanka, 'Application of a multigrid method to a buoyancy-induced flow problem', in S. F. McCormick (ed), *Lecture Notes in Pure and Applied Mathematics*, Vol. 110, *Multigrid Methods*, Marcel Dekker, New York, 1988, pp. 605–629.
3. S. P. Vanka, 'Block-implicit solution of Navier–Stokes equations in primitive variables', *J. Comput. Phys.*, **65**, 138–158 (1986).
4. W. Hackbusch, *Multi-grid Methods and Applications*, Springer, Berlin, 1985.
5. S. Zeng and P. Wesseling, 'Galerkin coarse grid approximation for the incompressible Navier–Stokes equations in general coordinates', *Rep. 92-35*, Faculty of Technical Mathematics and Informatics, Delft University of Technology, 1992.
6. S. Zeng and P. Wesseling, 'An efficient algorithm for the computation of Galerkin coarse grid approximation for the incompressible Navier–Stokes equations', *Rep. 92-40*, Faculty of Technical Mathematics and Informatics, Delft University of Technology, 1992.
7. S. Zeng and P. Wesseling, 'Multigrid solution of the incompressible Navier–Stokes equations in general coordinates', *SIAM J. Numer. Anal.*, in press.
8. A. E. Mynett, P. Wesseling, A. Segal and C. G. M. Kassels, 'The ISNaS incompressible Navier–Stokes solver: invariant discretization', *Appl. Sci. Res.*, **48**, 175–191 (1991).

9. C. W. Oosterlee, 'Robust multigrid methods for the steady and unsteady incompressible Navier–Stokes equations in general coordinates', *Ph.D. Thesis*, Delft University of Technology, 1993.
10. C. W. Oosterlee and P. Wesseling, 'A multigrid method for an invariant formulation of the incompressible Navier–Stokes equations in general coordinates', *Commun. Appl. Numer. Methods*, **8**, 721–734 (1992).
11. A. Segal, P. Wesseling, J. van Kan, C. W. Oosterlee and C. G. M. Kassels, 'Invariant discretization of the incompressible Navier–Stokes equations in boundary fitted co-ordinates', *Int. j. numer. methods fluids*, **15**, 411–426 (1992).
12. P. Wesseling, A. Segal, J. van Kan, C. W. Oosterlee and C. G. M. Kassels, 'Finite volume discretization of the incompressible Navier–Stokes equations in general coordinates on staggered grids', *Comput. Fluid Dyn. J.*, **1**, 27–33 (1992).
13. G. Wittum, 'On the convergence of multi-grid methods with transforming smoothers—theory with applications to the Navier–Stokes equations', *Numer. Math.*, **57**, 15–38 (1990).
14. A. Brandt and I. Yavneh, 'On multigrid solution of high-Reynolds incompressible entering flows', *J. Comput. Phys.*, **101**, 151–164 (1992).
15. P. M. de Zeeuw and E. J. van Asselt, 'The convergence rate of multi-level algorithms applied to the convection–diffusion equation', *SIAM J. Sci. Stat. Comput.*, **6**, 492–508 (1985).
16. P. Wesseling, 'A survey of Fourier smoothing analysis results', in W. Hackbusch and U. Trottenberg (eds), *International Series of Numerical Mathematics*, Vol. 98, *Multigrid Methods III*, Birkhäuser, Basel, 1991, pp. 105–127.
17. M. C. Thompson and J. H. Ferziger, 'An adaptive multigrid technique for the incompressible Navier–Stokes equations', *J. Comput. Phys.*, **82**, 94–121 (1989).
18. G. Wittum, 'Multi-grid methods for Stokes and Navier–Stokes equations—transforming smoothers: algorithm and numerical results', *Numer. Math.*, **54**, 543–563 (1989).
19. G. Wittum, 'The use of fast solvers in computational fluid dynamics', in P. Wesseling (ed.), *Notes on Numerical Fluid Mechanics*, Vol. 29, Vieweg, Braunschweig, 1990, pp. 574–581.
20. G. Wittum, 'R-transforming smoothers for the incompressible Navier–Stokes equations', in W. Hackbusch (ed.), *Notes on Numerical Fluid Mechanics*, Vol. 30, Vieweg, Braunschweig, 1990, pp. 153–162.
21. R. S. Varga, *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1962.
22. A. Brandt, *GMD-Studien*, Vol. 85, *Multigrid Techniques: 1984 Guide with Applications to Fluid Dynamics*, Gesellschaft für Mathematik und Datenverarbeitung, Sankt Augustin, 1984.
23. S. V. Patankar and D. B. Spalding, 'A calculation procedure for heat and mass transfer in three-dimensional parabolic flows', *Int. J. Heat Mass Transfer*, **15**, 1787–1806 (1972).
24. A. Brandt and N. Dinar, 'Multigrid solutions to flow problems', in S. Parter (ed.), *Numerical Methods for Partial Differential Equations*, Academic, New York, 1979, pp. 53–147.
25. P. Wesseling, *An Introduction to Multigrid Methods*, Wiley, Chichester, 1992.
26. G. Lonsdale and K. Stüben, 'The L_1SS package', *Arbeitspapiere der GMD 524*, 1991.
27. I. Demirdžić, Ŭ. Lilek and M. Perić, 'Fluid flow and heat transfer test problems for non-orthogonal grids: bench-mark solutions', *Int. j. numer. methods fluids*, **15**, 329–354 (1992).
28. C. W. Oosterlee, P. Wesseling, A. Segal and E. Brakkee, 'Benchmark solutions for the incompressible Navier–Stokes equations in general coordinates on staggered grids', *Int. j. numer. methods fluids*, **17**, 301–321 (1993).
29. K. J. Morgan, J. Périaux and F. Thomasset (eds), *Analysis of Laminar Flow over a Backward Facing Step*, Vieweg, Braunschweig, 1984.
30. S. Zeng and P. Wesseling, 'Numerical study of a multigrid method with four smoothing methods for the incompressible Navier–Stokes equations in general coordinates', In: N. Duane Melson, T. A. Mantenffel and S. F. McCormick (eds.), Sixth Copper Mountain Conference on Multigrid Methods, NASA Conference Publication 3224, 691–708, 1993.
31. P. Wesseling, 'Two remarks on multigrid methods', in W. Hackbusch (ed.), *Notes on Numerical Fluid Mechanics*, Vol. 23, *Robust Multigrid Methods*, Vieweg, Braunschweig, 1988, pp. 209–216.