ELSEVIER

Mathematics and Computers in Simulation 44 (1997) 369–385

MATHEMATICS AND COMPUTERS IN SIMULATION

Computation of turbulent flow in general domains[1]

P. Wesseling*, M. Zijlema, A. Segal, C.G.M. Kassels

Department of Technical, Mathematics and Informatics, Delft University of Technology, Mekelweg 4, 2628 CD Delft, Netherlands

Abstract

The computation of incompressible turbulent flow with two-equation closure models (k–ε and k–ω) is considered. The Cartesian staggered grid approach is generalized to general boundary-fitted coordinates. An accurate discretization on non-smooth grids is presented. For higher-order monotone discretization of the equations for the turbulence quantities, flux-limited versions of the κ-scheme are developed. In order to better assess the relative merits of explicit and implicit time discretization, a new approach to obtain von Neumann stability conditions is presented. A comparison is made between physical time scales for direct and large-eddy simulation, and stability restrictions on the time step for explicit schemes. Applications are presented for stationary turbulent flow computations with the k–ε model.

1. Introduction

The purpose of this contribution is to report on the experience that has been gathered in the development of a computing method for incompressible flow in complicated domains. We will mainly discuss the path that we have been following towards this goal, and mention alternative approaches only in passing; without any pretension, however, that our way is better. See Ref. [14] for a survey of the field. Our approach may be characterized as a coordinate-invariant generalization of the classical staggered grid discretization in Cartesian coordinates [7] and associated solution methods for incompressible flows.

2. Spatial discretization

We use the finite volume method, with a boundary-fitted coordinate mapping, resulting in a structured grid, and requiring block decomposition for complicated domains. There are basically two

* Corresponding author.

[1]Supported by the Netherlands Foundation for Mathematics (SMC) with financial aid from the Netherlands Organization for the Advancement of Scientific Research (NWO).

PII S0378-4754(97)00064-5

finite volume options for the incompressible Navier–Stokes equations: staggered and collocated (non-staggered). On Cartesian grids, the collocated approach needs special measures to avoid checkerboard oscillations, cf. [21]. However, generalizing the staggered approach to general grids is more complicated than for the collocated approach. This is caused by the fact that the cell face normal velocity components are used as unknowns, and that the staggered grid control volumes have a complicated shape. Furthermore, the results of a straightforward implementation may be inaccurate on non-smooth grids, due to the fact that practical boundary-fitted coordinate mappings are only piecewise differentiable. This difficulty may be overcome, however, by taking non-smoothness of the coordinate mapping into account, cf. [27,40]. As a consequence of these difficulties, in general coordinates the collocated version seems to be more popular than the staggered version at present; see Refs. [39,45,46]. We have chosen for the staggered approach, generalizing the classical Cartesian formulation [7] to general coordinates.

The viscous terms are generally discretized with central differences, but the discretization of the inertia terms is an issue. For the momentum equations, central second-order discretization has been employed successfully for large-eddy and direct simulation of turbulence, but some argue that higher-order central or upwind-biased schemes (such as QUICK [11]) are preferable. For convection of reacting chemical species, or of turbulence modelling quantities (e.g. k and ε) second-order accuracy may be and monotonicity is certainly required. We will illustrate how this may be achieved in general coordinates.

Let a boundary-fitted coordinate mapping in m dimensions be given:

$$\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{\xi}), \quad \boldsymbol{x} \in \Omega, \quad \boldsymbol{\xi} \in G$$

with Ω the physical domain and G the m-dimensional unit cube. In G we have a uniform grid with cells with vertices $\boldsymbol{\xi}_{i+1/2,j+1/2}$ (in two dimensions). The mapping is generated by a discrete numerical process specifying $\boldsymbol{x}(\boldsymbol{\xi}_{i+1/2,j+1/2})$ extended to all of G by multilinear interpolation. As a consequence, the cell faces are ruled surfaces for $m = 3$ and straight lines for $m = 2$, as illustrated in Fig. 1. In general coordinates, with summation over pairs of equal Greek super- and subscripts from 1 to m, the convection–diffusion equation for a scalar φ may be written as

$$\frac{\partial \varphi}{\partial t} + \frac{1}{\sqrt{g}} \frac{\partial V^\alpha \varphi}{\partial \xi^\alpha} - \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi^\alpha} (\sqrt{g} \boldsymbol{a}^{(\alpha)} . \nu \nabla \varphi) = s \tag{2.1}$$

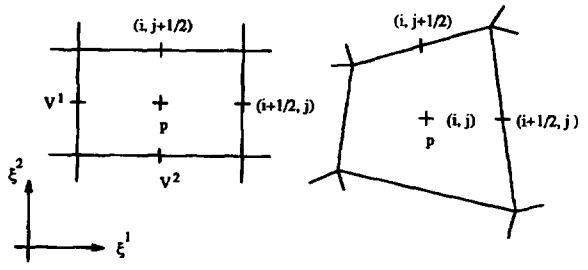

Fig. 1. Cell in G and its image in Ω.

Here $\sqrt{g}$ is the Jacobian of the mapping, s the source term, ν the diffusion coefficient, and

$$a^{(\alpha)} = \nabla\xi^\alpha, \quad V^\alpha = \sqrt{g}a^{(\alpha)} \cdot u$$

with u the velocity field, and ∇ the gradient operator. For $m = 2$, $a^{(\alpha)}$ can be determined from

$$a^{(1)} = (a^2_{(2)}, -a^1_{(2)})/\sqrt{g},\ a^{(2)} = (-a^2_{(1)}, a^1_{(1)})/\sqrt{g}$$

$$a_{(\alpha)} = \partial x/\partial\xi^\alpha, \sqrt{g} = a^1_{(1)}a^2_{(2)} - a^2_{(1)}a^1_{(2)} \tag{2.2}$$

The covariant basis vectors $a_{(\alpha)}$ can be determined exactly by straightforward finite differencing from the piecewise multilinear mapping $x = x(\xi)$.

The contravariant base vectors are discontinuous at cell faces. How to take this into account in order to avoid significant errors on non-smooth grids is discussed in Refs. [27,40]. Here we give a brief explanation for the viscous term. Integration over a finite volume Ω_{ij} with centre at (i,j) gives, noting that $\mathrm{d}x = \sqrt{g}\mathrm{d}\xi$,

$$\int_{G_{ij}} \frac{\partial}{\partial\xi^\alpha}(\sqrt{g}a^{(\alpha)} \cdot \nu\nabla\varphi)\mathrm{d}\xi^1\mathrm{d}\xi^2 = (\sqrt{g}a^{(1)} \cdot \nu\nabla\varphi)|^{i+1/2j}_{i-1/2j} + (\sqrt{g}a^{(2)} \cdot \nu\nabla\varphi)|^{ij+1/2}_{ij-1/2}$$

At the respective cell faces $\sqrt{g}a^{(1)}$ and $\sqrt{g}a^{(2)}$ are smooth and evaluated exactly by (2.2), and $\nu\nabla\varphi$ is smooth. Writing $\nabla\varphi$ in terms of ξ would introduce effects of the non-smoothness of $x = x(\xi)$. We, therefore, proceed as follows, allowing ν to be piecewise continuous per cell. We have

$$\varphi|^{i+1j}_{ij} = \int_{x_{ij}}^{x_{i+1j}} \frac{1}{\nu}\nu\nabla\varphi \cdot \mathrm{d}x \cong (c_{(1)} \cdot \nu\nabla\varphi)_{i+1/2j} \tag{2.3}$$

with

$$c_{(1)i+1/2j} \equiv \frac{1}{2}\{(a_{(1)}/\nu)_{ij} + (a_{(1)}/\nu)_{i+1j}\}$$

Similarly,

$$\frac{1}{4}(\varphi|^{ij+1}_{ij-1} + \varphi|^{i+1j+1}_{i+1j-1}) \cong (c_{(2)} \cdot \nu\nabla\varphi)_{i+1/2j} \tag{2.4}$$

with

$$c_{(2)} \equiv \frac{1}{8}\{(a_{(2)}/\nu)_{ij-1} + (a_{(2)}/\nu)_{ij+1} + (a_{(2)}/\nu)_{i+1j-1} + (a_{(2)}/\nu)_{i+1j+1} + 2(a_{(2)}/\nu)_{ij} + 2(a_{(2)}/\nu)_{i+1j}\}$$

Solving (2.3) and (2.4) for $\nu\nabla\varphi$ results in

$$(\nu\nabla\varphi)_{i+1/2j} = c^{(1)}\varphi|^{i+1j}_{ij} + \frac{1}{4}c^{(2)}(\varphi|^{ij+1}_{ij-1} + \varphi|^{i+1j+1}_{i+1j-1})$$

with

$$c^{(1)} \equiv (c^2_{(2)}, -c^1_{(2)})/C, c^{(2)} \equiv (-c^2_{(1)}, c^1_{(1)})/C$$

$$C = c^2_{(2)}c^1_{(1)} - c^2_{(1)}c^1_{(2)}$$

The other cell face diffusion fluxes are approximated in the same way. The resulting approximation is exact for $\nabla\varphi$ piecewise constant, irrespective of the smoothness of the grid and of ν. A similar procedure can be followed for the pressure and viscous terms in the Navier–Stokes equations.

Next, we turn to the discretization of the convection term in (2.1). Finite volume integration gives

$$\int_{G_{ij}} \frac{\partial V^\alpha \varphi}{\partial \xi^\alpha} \mathrm{d}\xi^1 \mathrm{d}\xi^2 = (V^1\varphi)|^{i+1/2,j}_{i-1/2,j} + (V^2\varphi)|^{i,j+1/2}_{i,j-1/2} \tag{2.5}$$

From (2.2) it follows that V^α is smooth at its points of definition on the staggered grid, so that the smoothness of the grid plays only a minor role here. In (2.5), $\varphi_{i+1/2,j}$, etc. have to be further approximated in terms of cell centre values. The equations for the turbulence quantities in two-equation turbulence models (e.g. k, ε or ω) are of the type (2.1). It is essential that the discretization does not allow spurious wiggles, and desirable that the spatial accuracy is of the second order. In order to have second-order accuracy, we take a three-point upwind biased scheme. Such schemes, requiring second-order accuracy, leave us with a free parameter κ, and can generally be written in the form of the κ-scheme [30]. Assuming $V^1_{i+1/2,j} \geq 0$, this scheme gives, deleting the subscript j for brevity,

$$\varphi_{i+1/2} \cong \frac{1}{2}(\varphi_i + \varphi_{i+1}) + \frac{1-\kappa}{4}(-\varphi_{i-1} + 2\varphi_i - \varphi_{i+1}) \tag{2.6}$$

For $\kappa = -1$, we have the fully upwind second-order scheme studied in Ref. [25]. For $\kappa = 0$, the zero-average phase error scheme proposed in Ref. [4] results; this scheme results from optimizing, among 5-point schemes, the propagation of a step function over one time step [35]. For $\kappa = 1/3$, used in Ref. [1], the local truncation error for the convection term becomes third order in space. For $\kappa = 1/2$, we have maximal accuracy for the cell face convection fluxes (QUICK scheme [11]). Finally, the standard second-order central scheme is obtained when $\kappa = 1$.

In order to avoid spurious wiggles the approximation (2.6) has to be made nonlinear. Nonlinear wiggle-free schemes can be formulated conveniently in a unifying framework laid out in Ref. [12]. The following normalized variable is introduced:

$$\tilde{\varphi} = (\varphi - \varphi_{i-1})/(\varphi_{i+1} - \varphi_{i-1})$$

For the κ-scheme, we have

$$\tilde{\varphi}_{i+1/2} = (1 - \kappa/2)\tilde{\varphi}_i + \frac{1+\kappa}{4}$$

A nonlinear scheme is obtained by taking

$$\tilde{\varphi}_{i+1/2} = f(\tilde{\varphi}_i) \tag{2.7}$$

Such schemes are analysed in Refs. [5,12,13]. We have second-order accuracy if $f(1/2) = 3/4$, and we may say that we have a nonlinear generalization of the κ-scheme if for $\tilde{\varphi}_i = 1/2$, $d\tilde{\varphi}_{i+1/2}/d\tilde{\varphi}_i$

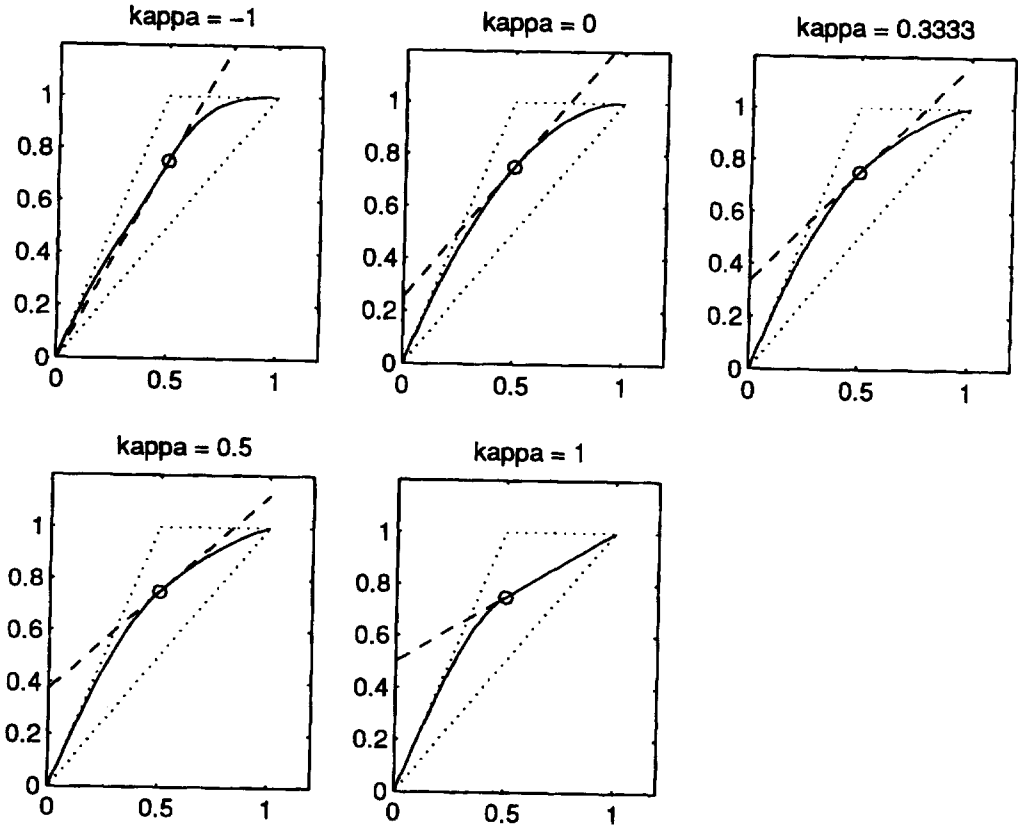

Fig. 2. Normalized variable flux functions for TVD κ-schemes.

agrees:

$$f'(1/2) = 1 - \kappa/2 \tag{2.8}$$

Using another formulation (the flux limiter), in Ref. [26] conditions are given for the scheme to be total variation diminishing (TVD). These conditions are easily reformulated in terms of normalized variables, and require that for $s \in [0, 1]$, $f(s)$ is inside the triangles shown in Fig. 2. For $s \notin [0, 1]$, we take the first-order upwind scheme: $f(s) = s$. Inside the triangle we take f differentiable, in order to facilitate iterative convergence. The following piecewise polynomial function satisfies, for $\kappa \in [-1, 1]$, the triangle condition, $f(1/2) = 3/4$ and (2.8):

$$f(s) = 2s + (\kappa - 1)s^2 - 2\kappa s^3, \quad 0 \le s \le 1/2$$

$$f(s) = 1 + \frac{1}{2}\kappa(s - 1) + (\kappa - 1)(s - 1)^2, \quad 0 \le \kappa \le 1, \ 1/2 < s \le 1$$

$$f(s) = 1 - (1 + \kappa)(s - 1)^2 - 2\kappa(s - 1)^3, \quad -1 \le \kappa < 0, \ 1/2 < s \le 1$$

This function is shown in Fig. 2.

For easy implementation we use defect correction on the first-order upwind scheme. Eq. (2.7) is rewritten as

$$\varphi_{i+1/2} = \varphi_i + \varphi_{i-1} - \varphi_i + (\varphi_{i+1} - \varphi_{i-1})f(\tilde{\varphi}_i)$$

and all terms but the first of the right-hand side are lagged behind one iteration or one time step, as the case may be.

3. Temporal discretization

We use time-stepping with the pressure correction method, which is simple, accurate and efficient for instationary flows on staggered grids. The second-order version given in Ref. [29] is used. In the following, by stability analysis we imply the von Neumann stability analysis, using Fourier transformation. In practice, the stability analysis for the incompressible Navier–Stokes and two-equation turbulence model equations then reduces to stability analysis for the convection–diffusion equation in Cartesian coordinates:

$$\frac{\partial \varphi}{\partial t} + \sum_{\alpha=1}^{m} \left(u_\alpha \frac{\partial}{\partial x_\alpha} - \nu \frac{\partial^2}{\partial x_\alpha^2} \right) \varphi = 0$$

The convection term is discretized with the κ-scheme and the diffusion term with central differences. Denoting the step-sizes by τ and h_α, stability is governed by the parameters c_α (CFL-numbers) and d_α, defined by

$$c_\alpha = |u_\alpha|\tau/h_\alpha, \quad d_\alpha = 2\nu\tau/h_\alpha^2$$

With $\kappa = 1$ and forward Euler temporal discretization (FTCS scheme) necessary and sufficient stability conditions are ([8,9,15]);

$$\sum_\alpha d_\alpha \leq 1 \quad \text{and} \quad \sum_\alpha c_\alpha^2/d_\alpha \leq 1$$

For leapfrog Euler in time and $\kappa = 1$, heavily used for large-eddy simulation in the past, the following criterion, put forward in Ref. [23] and proven to be sufficient in Ref. [20] is known:

$$\sum_\alpha (2d_\alpha + c_\alpha) \leq 1$$

whereas in Refs. [36,37] the following condition is shown to be necessary and sufficient:

$$\sum_\alpha (d_\alpha + \sqrt{d_\alpha^2 + c_\alpha^2}) \leq 1$$

For other temporal discretizations and/or other values of κ few analytic stability results seem to be known, especially for the number of space dimensions $m > 1$, and in practice heuristic criteria to determine τ are used. Here we briefly describe a method, put forward in Ref. [36], by which sufficient stability conditions are easily obtained in a wide variety of cases. Let $\hat{L}_h(\theta), \theta = (\theta_1, \theta_2, \ldots, \theta_m)$ be the

symbol (Fourier transform) of the spatial discretization operator L_h. Define $S_L \equiv \{-\tau\hat{L}_h(\theta) \in \mathbb{C} : \forall\theta\}$. Let $S \subset \mathbb{C}$ be the stability domain of the time integration method employed (in the usual sense in the context of solving ordinary differential equations). Then $S_L \subseteq S$ is sufficient for stability. Hence, it is useful to know something about S_L. In Refs. [36–38], the following theorems are proven for the κ-scheme. Define $\tilde{d} = \sum_\alpha \{d_\alpha + (1-\kappa)c_\alpha\}$.

Theorem 3.1: *If*

$$\tilde{d} \le a \quad \text{and} \quad \sum_\alpha c_\alpha^2/d_\alpha \le (2-\kappa)^{-2}b^2/a$$

then S_L is contained in the ellipse

$$(v/a+1)^2 + (w/b)^2 = 1, \; v+iw = z$$

Theorem 3.2: *If* $\tilde{d} \le a$ and *one or both of the following two conditions hold*:

$$\sum_\alpha (c_\alpha^4/d_\alpha)^{1/3} \le q_1(b^4/a)^{1/3} \quad \text{or} \quad \sum_\alpha c_\alpha \le q_2 b^2/a$$

where

$$q_1 = \frac{1}{4}(8-4\kappa)^{-5/3}(15-5\kappa-r)^{4/3}(5\kappa-3+r)^{1/3}(9-7\kappa+r)^{1/3},$$

$$r = (25\kappa^2 - 54\kappa + 33)^{1/2},$$

$$q_2 = (1-\kappa)^{3/2}(8/5-4\kappa/5)^{-5/2}, \quad -1 \le \kappa < 3/4,$$

$$q_2 = (1-\kappa)/2, \quad 3/4 \le \kappa \le 1$$

then S_L is contained in the oval given by

$$(v/a+1)^2 + (w/b)^4 = 1, \quad v+iw = z$$

Theorem 3.3: *If*

$$\tilde{d} \le \frac{a}{2} \quad \text{and} \quad \frac{2\tilde{d}}{b^2}\sum_\alpha c_\alpha^2/\tilde{d}_\alpha \le (2-\kappa)^{-2}(1+\sqrt{1-4\tilde{d}^2/a^2})$$

then S_L is contained in the ellipse given by

$$(v/a)^2 + (w/b)^2 = 1, \quad v+iw = z$$

Theorem 3.4: *If*

$$\sum_\alpha c_\alpha^2/d_\alpha \le q_3 b^2$$

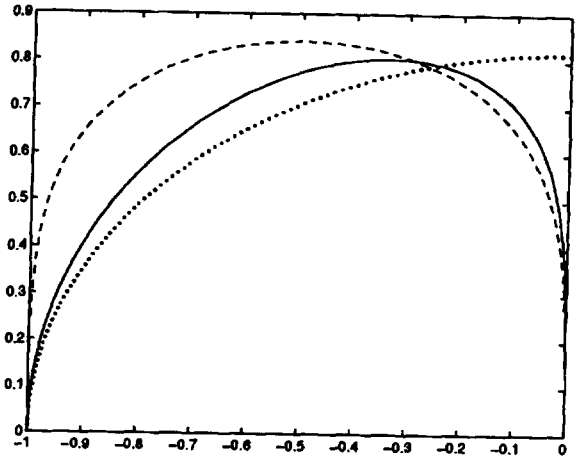

Fig. 3. Stability domain of second-order Adams–Bashforth scheme (———) with oval (- - -) and ellipse (· · ·).

where

$$q_3 = 1/2 \quad \textit{if } 1/2 \le \kappa \le 1,$$

$$q_3 = \frac{27}{8}(1-\kappa)/(2-\kappa)^3 \quad \textit{if } -1 \le \kappa \le 1/2$$

then S_L is contained in the parabola given by

$$v + (w/b)^2 = 0, \quad v + iw = z$$

Similar results are presented in Ref. [36] for $\kappa = 1$ and for fourth-order central discretization of the convection term. By fitting suitable unions or intersections of the conics and oval of these theorems inside S covering S to a satisfactory extent in order to stay close to necessity, sufficient stability conditions are easily obtained. We give a few examples. Fig. 3 shows S for the second-order Adams–Bashforth scheme with the oval of Theorem 3.2 (parameters: $a = 1/2$, $b = 2^{-1/4}$) and the ellipse of Theorem 3.3 (parameters: $a = 1$, $b = \sqrt{2/3}$). All stability domains to be encountered are symmetric with respect to the real axis, so that we show only the upper half. Requiring $S_L \subseteq$ *oval* $\cap$ *ellipse*, Theorems 3.2 and 3.3 immediately result in useful stability conditions. Fig. 4 shows S for the third-order Adams–Bashforth scheme with the ellipse of Theorem 3.3 (parameters: $a = 6/11$, $b = 72/11\sqrt{2/235} \cong 0.6038$). To show the versatility of the method we also apply it to a Runge–Kutta scheme, namely, the three-stage method described in Ref. [24] and called Wray's scheme. When applied to the single ordinary differential equation $dy/dt = \lambda y, \lambda \in C$ the method gives

$$y^{n+1} = P(z)y^n, \quad z = \lambda\tau$$

with the amplification polynomial $P(z)$ given by

$$P(z) = (1+\gamma_3 z)(1+\gamma_2 z)(1+\gamma_1 z) + \xi_1 z(1+\gamma_3 z) + \xi_2 z(1+\gamma_1 z)$$

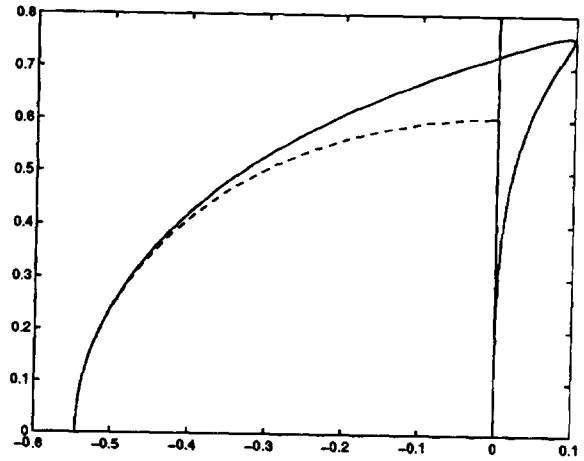

Fig. 4. Stability domain of third-order Adams–Bashforth scheme (———) with ellipse (— — —).

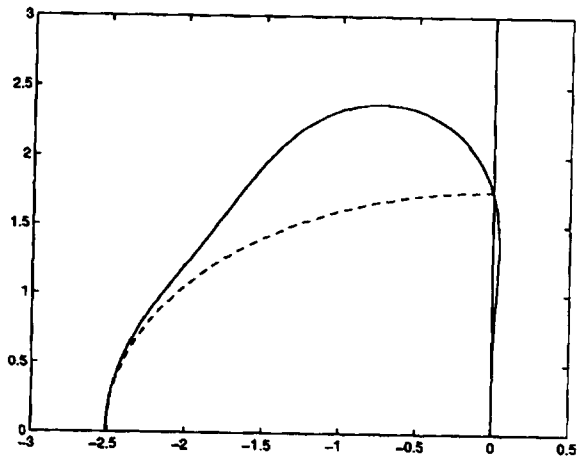

Fig. 5. Stability domain of Wray's Runge–Kutta scheme (———) with ellipse (- - -).

where $\gamma_1 = 8/15$, $\gamma_2 = 5/12$, $\gamma_3 = 3/4$, $\xi_1 = -17/60$, $\xi_2 = -5/12$. The stability domain is defined by

$$S = \{z \in \mathbb{C} : |P(z)| \leq 1\}$$

and is shown in Fig. 5, together with the ellipse of Theorem 3.3 (parameters: $a = 2.5127$, $b = 7/4$). Application of Theorem 3.3 gives sufficient stability conditions. Additional examples, including schemes with implicit time discretization of the diffusion term, are given in Refs. [36,37].

When choosing between explicit and implicit schemes, not only stability but also accuracy restrictions on τ have to be taken into account. As an example we consider direct numerical simulation of turbulent flow in a pipe with diameter L. The length scale $\mathscr{L}$ and the velocity scale $\mathscr{U}$ of the turbulence macrostructure are approximately

$$\mathscr{L} = L/10, \qquad \mathscr{U} = \sqrt{\tau_w/\rho}$$

with τ_w the wall friction. From Blasius' resistance formula ([6], section 155)

$$\mathscr{U} = 0.2U\,Re^{-1/8}, \quad Re = UL/\nu$$

with U the mean velocity. Hence the microscale Reynolds number is

$$Re_m = \mathscr{U}\mathscr{L}/\nu = \frac{1}{50}Re^{7/8}$$

Away from the wall the length and time scales of the micro structure are according to Kolmogorov's law

$$\eta = \mathscr{L}\,Re_m^{-3/4}, \quad \mathscr{T} = \eta/U = \frac{L}{10U}Re_m^{-3/4} \tag{3.1}$$

Near the wall the time scale is

$$\mathscr{T}_w = \nu/\mathscr{U}^2 = 0.9\frac{L}{U}Re_m^{-6/7}$$

In practice $Re_m < 10^9$, so that $\mathscr{T}_w > \mathscr{T}$. For accuracy, the mesh sizes must satisfy

$$\tau \le \mathscr{T}, \qquad h_\alpha \le \eta, \quad \alpha = 1,2,3$$

For the present qualitative considerations we disregard the need for polar coordinates in pipe flow, and assume a Cartesian grid. In direct simulation the viscous sublayer has to be resolved. It is reasonable to put the nearest grid point to the wall at $2 = y^+ = y\mathscr{U}/\nu$, resulting in the following step size perpendicular to the wall:

$$h_2 = 2\eta Re_m^{-1/4} \tag{3.2}$$

whereas h_1 and h_3 can be as in the interior.

The physical time scale $\mathscr{T}$ can be compared to the stability restrictions on τ for explicit schemes as follows. First, consider the interior. The maximum values of $|u_\alpha|$ can be estimated as (taking the x^1-axis in the mean flow direction)

$$|u_1| = U, \qquad |u_2| = |u_3| = \mathscr{U} = \gamma U, \quad \gamma = 0.2Re^{-1/8} \tag{3.3}$$

Furthermore, $h_1 = h_2 = h_3 = h$, so that the (maximum) CFL numbers become, with $c = \sum_\alpha c_\alpha$

$$c_1 = U\tau/h, \quad c_2 = c_3 = \gamma c_1, \quad c = (1+2\gamma)c_1 \tag{3.4}$$

The numbers d_α become

$$d_1 = d_2 = d_3 = \frac{1}{3}d, \quad d = \frac{6}{Re}\frac{\tau U}{L}\left(\frac{L}{h}\right)^2 \tag{3.5}$$

Taking as an example the second-order Adams–Bashforth method with the $\kappa = 1$ scheme, the stability conditions following from Theorems 3.2 and 3.3 and Fig. 3 can be written as

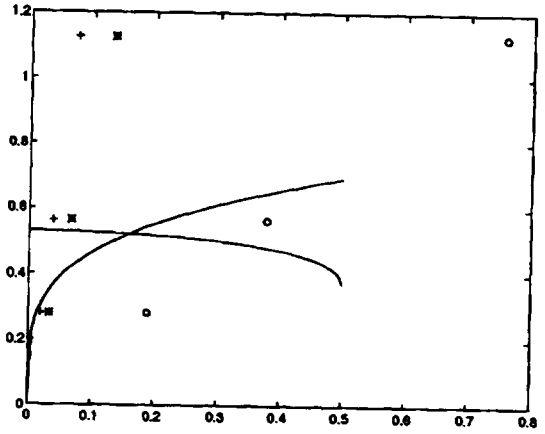

Fig. 6. c–d diagram for interior. -: equality in Eq. (3.7); * : direct simulation, (d,c) according to Eq. (3.8); +: large eddy simulation, (d,c) according to Eq. (3.11) with $\delta = 1$; o: $\delta = 10$.

$$d \le 1/2 \quad \text{and} \quad \sum_\alpha (c_\alpha^4/d_\alpha)^{1/3} \le 1 \quad \text{and} \quad 3d \sum_\alpha c_\alpha^2/d_\alpha \le 1 + \sqrt{1-4d^2} \tag{3.6}$$

If $c_2 = c_3 = \gamma c_1$ and $d_1 = d_2 = d_3$ this is equivalent with

$$d \le 1/2 \text{ and } c \le \left(\frac{d}{3}\right)^{1/4} \frac{1+2\gamma}{(1+2\gamma^{4/3})^{3/4}} \quad \text{and} \quad c \le \frac{1}{3}(1+2\gamma)\{(1+\sqrt{1-4d^2})/(1+2\gamma^2)\}^{1/2} \tag{3.7}$$

We will call the graph of $c(d)$ a c–d diagram, of which Fig. 6 is an example. For (3.7) to be satisfied, the point (d,c) must lie below the two curves. Choosing $\tau = s\mathscr{T}$ and $h = \eta$ with $\mathscr{T}, \eta$ given by (3.1) and s a parameter to be chosen we find

$$c = (1+2\gamma)s, \quad d = 6s\gamma Re_m^{-1/4} \tag{3.8}$$

For $Re_m = 64$ (i.e. $Re \cong 10^4$) and $s = 1, 1/2, 1/4$, (d,c) has been plotted in Fig. 6. We see that we have stability for $s = 1/4$, so that the stability requirement does not seem overly restrictive, and this explicit scheme seems efficient. Repeating the same analysis for the wall region, we choose h_2 according to (3.2) and $h_1 = h_3 = \eta$. Write $h_2 = \eta/\alpha$, $\alpha = \frac{1}{2}Re_m^{1/4}$. Furthermore, we assume $|u_1| = \beta\mathscr{U}(\beta \cong 2)$, $|u_2| = |u_3| = \mathscr{U}$. This gives

$$c_1 = \beta s\gamma, \; c_2 = \alpha s\gamma, \; c_3 = s\gamma, \; d_1 = d_3 = 2s\gamma Re_m^{-1/4}, \; d_2 = 2s\alpha^2\gamma Re_m^{-1/4}$$

Substitution in (3.6) gives

$$\begin{aligned} d \le 1/2 \quad &\text{and} \quad c \le d^{1/4}(1+\alpha+\beta)(2+\alpha^2)^{-1/4}(1+\alpha^{2/3}+\beta^{4/3})^{-3/4} \\ &\text{and} \quad c \le \frac{1}{\sqrt{3}}(1+\alpha+\beta)(2+\alpha^2)^{-1/2}(2+\beta^2)^{-1/2}(1+\sqrt{1-4d^2})^{1/2} \end{aligned} \tag{3.9}$$

We take $\beta = 2$, $Re_m = 64$. In order to satisfy (3.9) the point (d,c) must be below the two curves in Fig. 7. Choosing $\tau = s\mathscr{T}$, we find

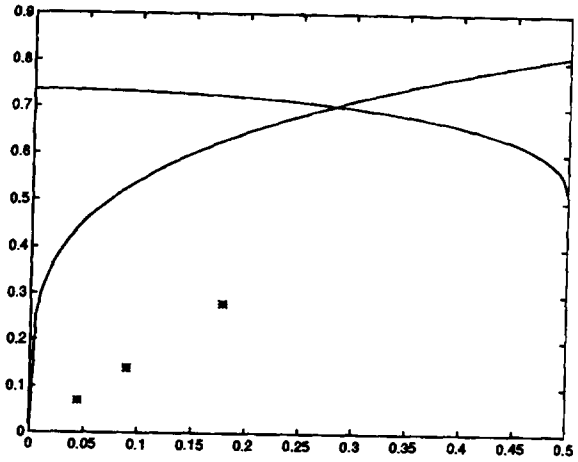

Fig. 7. c–d diagram for wall region -: equality in Eq. (3.9); * : direct simulation, (d,c) according to Eq. (3.10).

$$c = (1 + \alpha + \beta)\gamma s, \quad d = 2s\gamma(2 + \alpha^2)Re_m^{-1/4}, \tag{3.10}$$

For $Re_m = 64$ and $s = 1, 1/2, 1/4$, (c, d) has been plotted in Fig. 7. We see that in this case stability is less restrictive on τ than accuracy.

We repeat the analysis for large-eddy simulation in the interior of the pipe. The physical time scale to be resolved is $\mathcal{T} = h/U$. The eddy viscosity ν_e varies; we take $\nu_e = \delta\nu$ where typically $1 \leq \delta \simeq 10$. This gives, with $\tau = s\mathcal{T}$ and using (3.3):

$$c_1 = s, \quad c_2 = c_3 = \gamma c_1 \quad d_1 = d_2 = d_3 = \delta s \frac{L}{hRe} \tag{3.11}$$

The boundaries of the admissible region in the c–d diagram are again given by (3.7). When the turbulence intensity γ corresponds to (3.3), Fig. 6 gives (d,c) for $s = 1, 1/2, 1/4$ and $\delta = 1, 10$. We have stability for $s = 1/4$, so that again the stability requirement does not seem too restrictive.

4. Solution methods

We use implicit time stepping with the second-order pressure-correction method proposed in Ref. [29] to compute stationary flows, although using stationary methods instead of time-stepping may be more efficient. The nonlinear inertia term is Newton-linearized. The k–ε equations are discretized with the $\kappa = 1/2$ scheme (QUICK) with limiting implemented by defect correction on the first-order upwind scheme. The k–ε equations are solved after the momentum equations. The velocity components are updated simultaneously. The stencil of the discretization of a velocity component has 13 elements in two and 51 elements in three dimensions; for the stencil corresponding to the pressure correction equation these numbers are 9 and 19, respectively. Various solution methods have been studied, cf. [16–19] [31–34,39,41–44]. We use GCR [3] preconditioned with multigrid for the pressure and GMRESR [28] (a combination of GCR and GMRES [22]) for the velocity. In three dimensions this requires much storage. Per finite volume cell, we store five reals for geometric quantities, seven for $\nu, \rho, k, \varepsilon$ and a right-hand side, eight for two solution fields, $3 \times 51 = 153$ for the velocity matrix, six

for the preconditioner and 45 for building a 15-dimensional Krylov subspace, giving a total of 224 reals per cell. Solving for the pressure, k and ε requires no additional storage, since the same memory can be reused. In the explicit case, the pressure matrix requires 19 reals per cell, the preconditioner two and a workspace of 10 is needed, giving a total of 51 reals per cell. This would seem to be not much above a lower limit for the storage requirement for solving Navier–Stokes in general coordinates. One could economize on the above storage requirement for implicit methods by decoupled solution of the velocity components or using other methods that do not use the full matrix, such as ADI, but this would decrease robustness and might increase computing time. Obviously, explicit methods require significantly less memory than implicit methods, and also consume less computer time per time step.

5. Applications

We computed a turbulent flow in a two-dimensional channel with a restriction and symmetry with respect to the centreline. The computational domain is shown in Fig. 8. The problem is fully specified in Ref. [2], which also contains experimental results. Fig. 9 gives some results computed on a 150 × 100 grid, which gives sufficient resolution, as verified by mesh refinement. A turbulent kinetic energy and velocity profile are shown at a distance of 2 radii from the inlet, together with experimental values. The standard k–ε model [10] is used, with two discretizations: first-order upwind ('UDS') and QUICK with limiting ('ISNAS' [47]). Total computing time was 50 and 64 min, respectively, on a HP 9000/735 workstation (445 and 435 time steps, respectively). There is hardly any difference between the two discretizations. The reason is that here production and dissipation of turbulent energy are in good balance, and the role of the convection term is minor. The discrepancies with experiment are not due to the numerics but to the turbulence modelling.

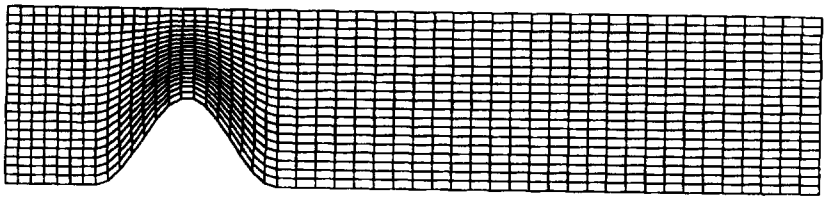

Fig. 8. Computational domain with grid.

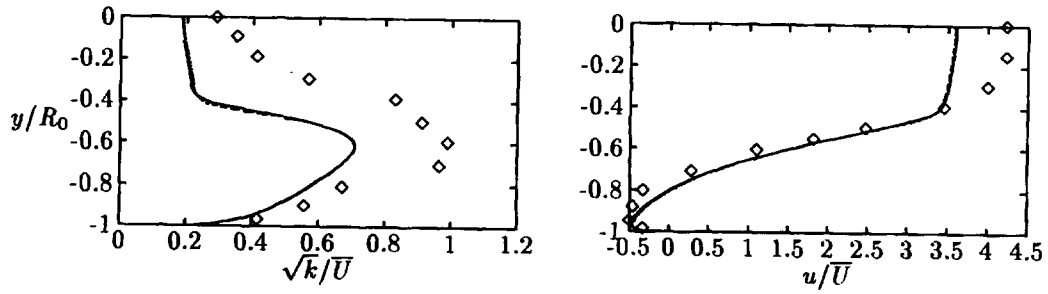

Fig. 9. Turbulent kinetic energy (k) profile (left) and velocity profile (right); UDS, - - -: ISNAS, – ◇ –: experiment.

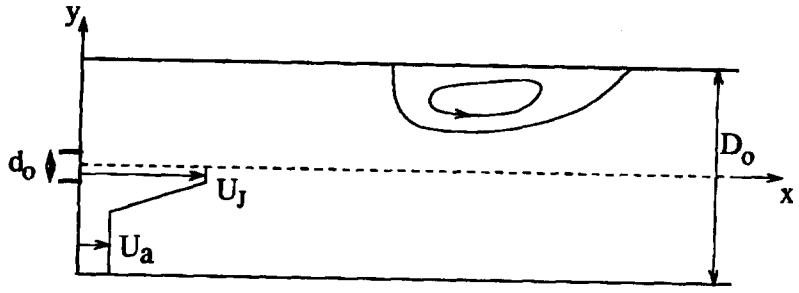

Fig. 10. Flow configuration.

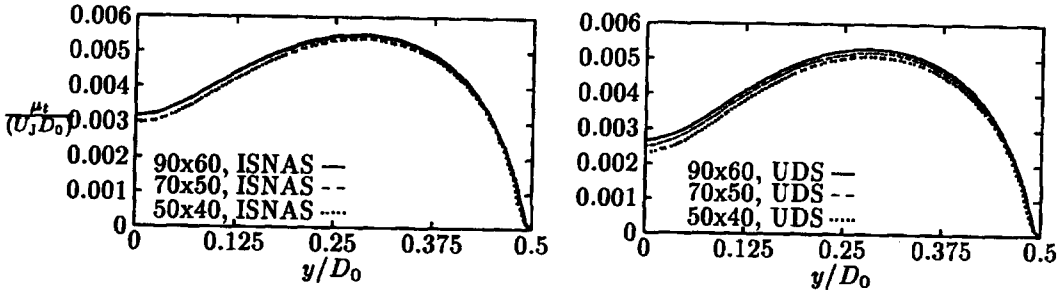

Fig. 11. Turbulent viscosity profiles at $x/D_0 = 1.875$.

In the following example the role of the convection term is more important. We consider a co-flow jet in a planar channel (Fig. 10). The problem is specified in Ref. [47]. We take $U_J/U_a = 31.49$ and $Re = U_J d_0/\nu = 1.5 \times 10^5$. The standard k–ε model is used again, both with UDS and ISNAS discretization of convection. Fig. 11 shows the effect of grid resolution and discretization on the turbulent viscosity profile. There is a noticeable difference between UDS and ISNAS. The latter is assumed to be more accurate, and allows a coarser grid. The ISNAS solution on the 70×50 grid required 135 min (1500 time steps, this flow takes a long time to become steady) on a HP 9000/735 workstation. We find that the computing times reported can be decreased by a factor of about 2.5 by using the solution on a coarser grid (with mesh-size doubled) as initial condition. At first sight one might think that replacing time-stepping for stationary problems by a stationary solution method would save time, but in the case of the k–ε turbulence model, due to its strong nonlinearity, methods other than time-stepping are hard to make converge and require many iterations.

6. Concluding remarks

Although more complicated than for collocated schemes, generalization of the staggered grid discretization of the incompressible Navier–Stokes equations from Cartesian to general coordinates is feasible. Inclusion of associated strongly nonlinear convection–diffusion equations for scalars, such as the k–ε equations for turbulence modelling, presents no particular problems compared to the Cartesian case. The flux-limited κ-scheme approach carries over from Cartesian to general coordinates. In two dimensions, engineering applications are feasible on workstations. Storage and computing time

requirements need to be reduced in three dimensions. For certain flow problems, such as direct and large-eddy simulation, explicit schemes may be efficient. For implicit schemes, storage requirements may be reduced by uncoupled solution of velocity components, or even more by using matrix-free solvers. However, this will probably increase computing time and lower robustness. For stationary flows, computing time can probably be reduced significantly by replacing time stepping by a stationary solver.

References

[1] W.K. Anderson, J.L. Thomas, B. van Leer, A comparison of finite volume flux vector splittings for the Euler equations, AIAA Paper (1985) 85–0122.

[2] M.D. Deshpande, D.P. Giddens, Turbulence measurements in a constricted tube, J. Fluid Mech. 97 (1980) 65–89.

[3] S.C. Eisenstat, H.C. Elman, M.H. Schultz, Variable iterative methods for nonsymmetric systems of linear equations, SIAM J. Numer. Anal. 20 (1983) 345–357.

[4] J.E. Fromm, A method for reducing dispersion in convective difference schemes, J. Comp. Phys. 3 (1968) 176–189.

[5] P.H. Gaskell, K.C. Lau, Curvature-compensated convective transport: SMART, a new boundedness-preserving transport algorithm, Int. J. Numer. Methods in Fluids 8 (1988) 617–641.

[6] S. Goldstein (Ed.), Modern developments in fluid dynamics, Vol. 2, New York, 1965, Dover.

[7] F.H. Harlow, J.E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface, The Physics of Fluids 8 (1965) 2182–2189.

[8] A.C. Hindmarsh, P.M. Gresho, D.F. Griffiths, The stability of explicit Euler time-integration for certain finite difference approximations of the multi-dimensional advection-diffusion equation, Int. J. Num. Meth. Fluids 4 (1984) 853–897.

[9] C.W. Hirt, Heuristic stability theory for finite difference equations, J. Comp. Phys. 2 (1968) 339–355.

[10] B.E. Launder, D.B. Spalding, The numerical computation of turbulent flows, Comp. Methods Appl. Mech. Eng. 3 (1974) 269–289.

[11] B.P. Leonard, A stable and accurate convective modelling procedure based on quadratic upstream interpolation, Comput. Meth. Appl. Mech. Eng. 19 (1979) 59–98.

[12] B.P. Leonard, Locally modified quick scheme for highly convective 2-d and 3-d flows, In Numerical methods in laminar and turbulent flows, Vol. 5, Part 1, 35–47, Swansea 1987, Pineridge Press.

[13] B.P. Leonard, Simple high-accuracy resolution program for convective modelling of discontinuities, Int. J. Numer. Meth. Fluids 8 (1988) 1291–1318.

[14] M.A. Leschziner, Modeling turbulent recirculating flows by finite volume methods – current status and future directions, Int. J. Heat and Fluid Flow 10 (1989) 186–201.

[15] K.W. Morton, Stability and convergence in fluid flow problems, Proc. Roy. Soc. London A 323 (1971) 237–253.

[16] C.W. Oosterlee, P. Wesseling, A multigrid method for an invariant formulation of the incompressible Navier–Stokes equations in general co-ordinates, Communications in Applied Numerical methods 8 (1992) 721–734.

[17] C.W. Oosterlee, P. Wesseling, A robust multigrid method for a discretization of the incompressible Navier–Stokes equations in general coordinates. In Ch. Hirsch, J. Périaux, W. Kordulla (Eds.), Computational Fluid Dynamics '92. Proc., First European Computational Fluid Dynamics Conf., Sept. 1992, Brussels, 101–108, Amsterdam, 1992. Elsevier.

[18] C.W. Oosterlee, P. Wesseling, Multigrid schemes for time-dependent incompressible Navier–Stokes equations, Impact Comp. Science Engng. 5 (1993) 153–175.

[19] C.W. Oosterlee, P. Wesseling, A robust multigrid method for a discretization of the incompressible Navier–Stokes equations in general coordinates, Impact Comp. Science Engng. 5 (1993) 128–151.

[20] M.J.B.M. Pourquié, Large-eddy simulation of a turbulent jet. PhD. thesis, Delft University of Technology, 1994.

[21] C.M. Rhie, W.L. Chow, Numerical study of the turbulent flow past an airfoil with trailing edge separation, AIAA Journal 21 (1983) 1525–1532.

[22] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving non-symmetric linear systems, SIAM J. Sci. Stat. Comp. 7 (1986) 856–869.

[23] U. Schumann, Linear stability of finite difference equations for three-dimensional flow problems, J. Comp. Phys. 18 (1975) 465–470.
[24] P.R. Spalart, R.D. Moser, M.M. Rogers, Spectral methods for the Navier–Stokes equations with one infinite and two periodic directions, J. Comp. Phys. 96 (1991) 297–324.
[25] J.L. Steger, R.F. Warming, Flux-vector splitting of the inviscid gas-dynamic equations with applications to finite-difference methods, J. Comp. Phys. 32 (1981) 263–293.
[26] P.K. Sweby, High resolution schemes using flux-limiters for hyperbolic conservation laws, SIAM J. Num. Anal. 21 (1984) 995–1011.
[27] P. van Beek, R.R.P. van Nooyen, P. Wesseling, Accurate discretization on non-uniform curvilinear staggered grids, J. Comp. Phys. 117 (1995) 364–367.
[28] H.A. van der Vorst, C. Vuik, GMRESR: A family of nested GMRES methods, Num. Lin. Alg. Appl. 1 (1994) 369–386.
[29] J.J.I.M. Van Kan, A second-order accurate pressure correction method for viscous incompressible flow, SIAM J. Sci. Stat. Comp. 7 (1986) 870–891.
[30] B. Van Leer, Upwind-difference methods for aerodynamic problems governed by the Euler equations, Lectures in Appl. Math. 22 (1985) 327–336.
[31] C. Vuik, Further experiences with GMRESR, Supercomputer 55 (1993) 13–27.
[32] C. Vuik, The solution of the discretized incompressible Navier–Stokes equations with iterative methods, In R. Himeno, R. Maly (Eds.), Proceedings of the 26th International Symposium on Automotive Technology and Automation; Dedicated conference on supercomputer applications in the automotive industries, 69–76, Croydon, 1993, Automotive Automation Limited.
[33] C. Vuik, Solution of the discretized incompressible Navier–Stokes equations with the GMRES method, Int. J. for Num. Meth. Fluids 16 (1993) 507–523.
[34] C. Vuik, GMRES-like methods with variable preconditioners, In J.D. Brown, M.T. Chu, D.C. Ellison, R.J. Plemmons (Eds.), Proceedings of the Cornelius Lanczos Centenary Conference, Raleigh, North Carolina, December 12–17 (1993) 282–284, Philadelphia, 1994, SIAM.
[35] P. Wesseling, On the construction of accurate difference schemes for hyperbolic partial differential equations, J. Eng. Math. 7 (1973) 1–31.
[36] P. Wesseling, A method to obtain von Neumann stability conditions for the convection–diffusion equation. In M.J. Baines, K.W. Morton (Eds.), Numerical Methods in Fluid Dynamics V, 211–224, Clarendon Press, Oxford, 1995.
[37] P. Wesseling, Von Neumann stability conditions for the convection–diffusion equation. IMA J. of Numer. Anal. 16 (1996) 583–598.
[38] P. Wesseling, Von Neumann stability conditions for the convection–diffusion equation, Report 95-48, Delft University of Technology, Faculty of Technical Mathematics and Informatics, Delft, The Netherlands, 1995.
[39] P. Wesseling, C.G.M. Kassels, C.W. Oostezlee, A. Segal, C. Vuik, S. Zeng, M. Zijlema, Computing incompressible flows in general domains, In F.-K. Hebeker, R. Rannacher, G. Wittum (Eds.), Numerical methods for the Navier–Stokes equations, 298–314, Braunschweig, 1994, Vieweg.
[40] P. Wesseling, P. van Beek, R.R.P. van Nooyen, Aspects of non-smoothness in flow computations, In A. Peters, G. Wittum, B. Herrling, U. Meissner, C.A. Brebbia, W.G. Gray, G.F. Pinder (Eds.), Computational Methods in Water Resources X, 1263–1271, Dordrecht, 1994, Kluwer.
[41] S. Zeng, C. Vuik, P. Wesseling, Numerical solution of the incompressible Navier–Stokes equations by Krylov subspace and multigrid methods, Advances in Computational Mathematics 4 (1995) 27–50.
[42] S. Zeng, 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. Manteuffel, S.F. McCormick (Eds.), Sixth Copper Mountain Conference on Multigrid Methods, NASA Conference Publication 3224, 691–708, Hampton VA., 1993, NASA.
[43] S. Zeng, P. Wesseling, Multigrid solution of the incompressible Navier–Stokes equations in general coordinates, SIAM J. Num. Anal. 31 (1994) 1764–1784.
[44] S. Zeng, P. Wesseling, An ILU smoother for the incompressible Navier–Stokes equations in general coordinates, Int. J. Numer. Meth. in Fluids 20 (1995) 59–74.
[45] M. Zijlema, A. Segal, P. Wesseling, Finite volume computation of incompressible turbulent flows in general coordinates on staggered grids, Int. J. Numer. Meth. Fluids 20 (1995) 621–640.

[46] M. Zijlema, A. Segal, P. Wesseling, Invariant discretization of the k–ϵ model in general co-ordinates for prediction of turbulent flow in complicated geometries, Comput. Fluids, 24 (1995) 209–225.
[47] M. Zijlema, P. Wesseling, On accurate discretization of turbulence transport equations in general coordinates, In C. Taylor, P. Durbetaki (Eds.), Numerical Methods in Laminar and Turbulent Flow, Vol. 9, Part 1, 34–45, Swansea, UK, 1995, Pineridge Press, Proc. Ninth Int. Conf. on Numer. Meth. Laminar and Turbulent Flow.