AIAA JOURNAL
Vol. 38, No. 11, November 2000

Computational Treatment of Source Terms in Two-Equation Turbulence Models

B. Merci*
Universiteit Gent, B-9000 Ghent, Belgium
J. Steelant†
ESA, NL-2200 AG Noordwijk, The Netherlands
and
J. Vierendeels,‡ K. Riemslagh,§ and E. Dick¶
Universiteit Gent, B-9000 Ghent, Belgium

The source terms in turbulence models require careful treatment to obtain a stable discretization. The choice between implicit and explicit treatment has to be made. This can be done either on the basis of individual terms or on the basis of the exact Jacobian of the source terms. A comparison of both methods shows that the latter is generally applicable and superior to the first, approximate method with respect to convergence speed. This comes from the possibility of using the multigrid technique with the exact method, whereas this is not always possible with the approximate method. It is also shown that, in principle, for robustness a time-step restriction for the source terms has to be introduced to prevent the turbulence quantities from becoming negative or infinitely large. An approximation of the appropriate time step is calculated. Practical results, however, indicate that the time-step restriction is not always necessary. Different two-equation turbulence models are investigated confirming the generality of the approach.

Nomenclature

c_f	=	friction coefficient, $\sqrt{\tau_w}/0.5\rho u_\infty^2$
$\boldsymbol{F}$	=	flux vector in x direction
G	=	amplification factor
$\boldsymbol{G}$	=	flux vector in y direction
H	=	step height
k	=	turbulence kinetic energy
p	=	pressure
p'	=	effective kinematic pressure, $p/\rho + \frac{2}{3}k$
$\boldsymbol{Q}$	=	vector of primitive variables
q	=	square root of turbulence kinetic energy
S	=	source term
$\boldsymbol{S}$	=	source term vector
u, v	=	velocity components
u_τ	=	friction velocity, $\sqrt{(\tau_w/\rho)}$
u_∞	=	freestream velocity
x, y	=	coordinate directions
β	=	turbulence model constant
β_x, β_y	=	velocity in dissipation term
$\bar{\bar{\Gamma}}$	=	preconditioning matrix
ε	=	turbulence dissipation rate
κ	=	parameter in van Leer discretization
$\bar{\bar{\Lambda}}$	=	eigenvalue matrix
λ_1, λ_2	=	eigenvalue
ν	=	kinematic viscosity
ρ	=	density
σ	=	turbulence model constant
τ	=	pseudotime
τ_w	=	wall shear stress
ϕ	=	variable
$\boldsymbol{\phi}$	=	vector of turbulence variables
ω	=	specific dissipation rate

Subscripts

a	=	acoustic
c	=	convective
d	=	dissipation
E	=	explicit
I	=	implicit
i	=	index in x direction
$i+\frac{1}{2}$	=	right cell face
j	=	index in y direction
$j+\frac{1}{2}$	=	upper cell face
k	=	turbulence kinetic energy
L	=	left cell face
R	=	right cell face
t	=	turbulent
v	=	viscous
ε	=	turbulence dissipation rate
∞	=	freestream

Superscripts

(m)	=	pseudotime level in multistage scheme
$(m+1)^*$	=	pseudotime level in multistage scheme
n	=	pseudotime level
$-$	=	negative part

Received 7 June 1999; presented as Paper 99-3371 at the AIAA 14th Computational Fluid Dynamics Conference, Norfolk, VA, 28 June–1 July 1999; revision received 20 October 1999; accepted for publication 2 March 2000.

*Research Assistant, Department of Flow, Heat and Combustion Mechanics, Sint-Pietersnieuwstraat 41; Bart.Merci@rug.ac.be. Member AIAA.

†Senior Researcher, Section of Aerothermodynamics, European Space Research and Technology Centre, Keplerlaan 1, P.O. Box 299; jsteelan@estec.esa.nl. Member AIAA.

‡Senior Researcher, Department of Flow, Heat and Combustion Mechanics, Sint-Pietersnieuwstraat 41; Jan.Vierendeels@rug.ac.be.

§Senior Researcher, Department of Flow, Heat and Combustion Mechanics, Sint-Pietersnieuwstraat 41; Kris.Riemslagh@rug.ac.be.

¶Professor, Department of Flow, Heat and Combustion Mechanics, Sint-Pietersnieuwstraat 41; Erik.Dick@rug.ac.be. Member AIAA.

Introduction

LOW-REYNOLDS two-equation turbulence models are nowadays often used in both industry and research institutes. They describe the physics of turbulence more universally than the algebraic models and are less time consuming than the Reynolds stress models. When implementing the turbulence equations into computational fluid dynamics (CFD) codes, special attention must be given to the treatment of the source terms. During the iterative process, the turbulent quantities are not allowed to grow infinitely fast or to

become negative. Independently of the turbulence model, the generally used method is to treat the negative source terms implicitly and the positive terms explicitly. Examples of this are the methods suggested by Vandromme and Ha Minh[1] for k–ε turbulence models or the method suggested by Wilcox[2] for k–ω based turbulence models. In this paper, a thorough study is carried out to analyze in a general way differential equations with (nonlinear) source terms. Different methods of source term treatment are presented and numerically tested.

Navier–Stokes equations combined with low-Reynolds turbulence models are discretized on high $\mathcal{AR}$ meshes to resolve the boundary layer. The use of these meshes causes a very numerically anisotropic behavior of the diffusive and acoustic terms. A robust method to handle this stiffness consists of the use of a line implicit line solver combined with multigrid.[3] In this work, high grid $\mathcal{AR}$ appear in both directions. The stiffness is removed by the use of an alternating line solver with an implicit treatment of acoustic and diffusive terms. If there are only high $\mathcal{AR}$ in one direction, one-directional sweeps are used with lines in the direction of the shortest grid distances. Earlier, it was shown[4] that this approach, accelerated with the multigrid technique, performs well for incompressible and low Mach compressible laminar flow cases. This technique is here extended to turbulent flows.

Previous studies[5,6] showed that the application of the multigrid technique on turbulence equations is not evident. It was found that the coarse grid corrections for the turbulent quantities need to be damped to make the multigrid work. However, only a slight decrease in convergence rate was observed for the fluid equations when the multigrid technique was only applied to the Navier–Stokes (NS) equations, whereas the turbulence equations were solved on the finest grid. Similar difficulties have been reported by Lien and Leschziner[7] who also had to introduce damping of the coarse grid corrections. Gerlinger and Brüggemann[8] observed the same difficulties but proposed to freeze the nonlinear parts in the source term when they used the multigrid technique. This approach appears to be fruitful for a q–ω turbulence model with a large increase in convergence when both the fluid and turbulence equations are taken within the multigrid (MG) cycle.[8] In this study, results will be shown where both the NS equations and the turbulence equations are solved by the use of the MG technique combined with an implicit line solver, without any damping of coarse grid corrections or freezing of any source term parts, which results in a considerable convergence acceleration.

Governing Equations

The two-dimensional incompressible steady NS equations in conservative form are

$$\frac{\partial \boldsymbol{F}_c}{\partial x} + \frac{\partial \boldsymbol{F}_a}{\partial x} + \frac{\partial \boldsymbol{G}_c}{\partial y} + \frac{\partial \boldsymbol{G}_a}{\partial y} = \frac{\partial \boldsymbol{F}_v}{\partial x} + \frac{\partial \boldsymbol{G}_v}{\partial y} + \boldsymbol{S} \qquad (1)$$

where $\boldsymbol{F}_c$ and $\boldsymbol{G}_c$ are the convective fluxes, $\boldsymbol{F}_a$ and $\boldsymbol{G}_a$ the acoustic fluxes, $\boldsymbol{F}_v$ and $\boldsymbol{G}_v$ the viscous fluxes

$$\boldsymbol{F}_c = \begin{bmatrix} 0 \\ u^2 \\ uv \\ uk \\ u\varepsilon \end{bmatrix}, \qquad \boldsymbol{F}_a = \begin{bmatrix} u \\ p' \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\boldsymbol{F}_v = \begin{bmatrix} 0 \\ 2(\nu + \nu_t)\dfrac{\partial u}{\partial x} \\ (\nu + \nu_t)\left(\dfrac{\partial v}{\partial x} + \dfrac{\partial u}{\partial y}\right) \\ \left(\nu + \dfrac{\nu_t}{\sigma_k}\right)\dfrac{\partial k}{\partial x} \\ \left(\nu + \dfrac{\nu_t}{\sigma_\varepsilon}\right)\dfrac{\partial \varepsilon}{\partial x} \end{bmatrix}$$

$$\boldsymbol{G}_c = \begin{bmatrix} 0 \\ uv \\ v^2 \\ vk \\ v\varepsilon \end{bmatrix}, \qquad \boldsymbol{G}_a = \begin{bmatrix} v \\ 0 \\ p' \\ 0 \\ 0 \end{bmatrix}$$

$$\boldsymbol{G}_v = \begin{bmatrix} 0 \\ (\nu + \nu_t)\left(\dfrac{\partial u}{\partial y} + \dfrac{\partial v}{\partial x}\right) \\ 2(\nu + \nu_t)\dfrac{\partial v}{\partial y} \\ \left(\nu + \dfrac{\nu_t}{\sigma_k}\right)\dfrac{\partial k}{\partial y} \\ \left(\nu + \dfrac{\nu_t}{\sigma_\varepsilon}\right)\dfrac{\partial \varepsilon}{\partial y} \end{bmatrix}$$

and $\boldsymbol{S} = [0, 0, 0, S_k, S_\varepsilon]^T$ the source term. The source terms S_k and S_ε, the constants σ_k and σ_ε, and the turbulent kinematic viscosity ν_t are given by the turbulence model.

A distinction is made between the convective and acoustic parts of the inviscid flux vector because a different spatial and temporal discretization is used for these parts. The source term treatment will be discussed later.

Discretization

A finite volume method with control volumes centred around the nodes is used. The discretization of the convective flux is based on velocity upwinding:

$$\boldsymbol{F}_{c_{i+\frac{1}{2}}} = u_{i+\frac{1}{2}}[0 \quad u \quad v \quad k \quad \varepsilon]^T_{L/R}$$

$$\boldsymbol{G}_{c_{j+\frac{1}{2}}} = v_{j+\frac{1}{2}}[0 \quad u \quad v \quad k \quad \varepsilon]^T_{L/R}$$

where

$$(\cdot)_{L/R} = \begin{cases} (\cdot)_L & \text{if } u_{\frac{1}{2}} > 0 \\ (\cdot)_R & \text{otherwise} \end{cases}$$

$$u_{i+\frac{1}{2}} = (u_i + u_{i+1})/2, \qquad v_{j+\frac{1}{2}} = (v_j + v_{j+1})/2$$

A short notation for the subscripts is used. The subscript that is not shifted with respect to i or j is omitted. The left and right values are computed with the van Leer–κ approach with $\kappa = \frac{1}{3}$ for third-order accuracy.

The acoustic and viscous fluxes are discretized in the central way. The discretization of the convective and acoustic terms corresponds to the original AUSM scheme[9] if the energy equation is omitted, a constant density is assumed, and Mach number moves toward zero. Because the pressure term is discretized in the central way, pressure stabilization is needed.

An artificial dissipation term for the pressure is added in the continuity equation in the following way:

$$\boldsymbol{F}_{d_{i+\frac{1}{2}}} = \begin{bmatrix} \dfrac{1}{2}\dfrac{p'_{i+1} - p'_i}{\beta_x} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \qquad \boldsymbol{G}_{d_{j+\frac{1}{2}}} = \begin{bmatrix} \dfrac{1}{2}\dfrac{p'_{j+1} - p'_j}{\beta_y} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where β_x and β_y have the dimension of velocity. We take

$$\beta_x = w_r + 2(\nu + \nu_t)/\Delta x, \qquad \beta_y = w_r + 2(\nu + \nu_t)/\Delta y$$

where w_r is, in our application, the maximum velocity within the flowfield. In the case of inviscid flow, this term corresponds to the dissipation term introduced by the flux-difference splitting method for incompressible flow.[10] According to Weiss and Smith,[11] β_x and

β_y are treated in such a way that they scale with the local diffusion velocities $(\nu + \nu_t)/\Delta x$ and $(\nu + \nu_t)/\Delta y$ when these terms become important.

Time Marching Method

A time marching method is used to reach the steady-state solution of the incompressible Reynolds-averaged NS equations. Applying the pseudocompressibility method to the conservative form of the inviscid part of Eq. (1) gives

$$\bar{\bar{\Gamma}}\frac{\partial \boldsymbol{Q}}{\partial \tau} + \frac{\partial \boldsymbol{F}_c}{\partial x} + \frac{\partial \boldsymbol{F}_a}{\partial x} + \frac{\partial \boldsymbol{G}_c}{\partial y} + \frac{\partial \boldsymbol{G}_a}{\partial y} = \boldsymbol{S}$$

$\boldsymbol{Q}$ is the vector of variables $[p', u, v, k, \varepsilon]^T$. The preconditioning matrix $\bar{\bar{\Gamma}}$ has nonzero entries $\Gamma_{11} = 1/\beta^2$, $\Gamma_{22} = \Gamma_{33} = \Gamma_{44} = \Gamma_{55} = 1$ where β has the dimension of velocity. The eigenvalues of the inviscid part of the preconditioned system are given by

$$\lambda\left[\bar{\bar{\Gamma}}^{-1}\frac{\partial(n_x \boldsymbol{F} + n_y \boldsymbol{G})}{\partial \boldsymbol{Q}}\right] = w, w, w, w \pm c \tag{2}$$

where $w = n_x u + n_y v$, $c = \sqrt{(w^2 + \beta^2)}$, and n_x and n_y denote an arbitrary direction with $n_x^2 + n_y^2 = 1$. If β is of the same order of magnitude as the convective speed, all eigenvalues will be properly scaled in at least one direction.

Stepping in Pseudotime

A multistage stepping with four stages is used

$$\boldsymbol{Q}^{(0)} = \boldsymbol{Q}^n, \qquad \boldsymbol{Q}^{(1)} = \boldsymbol{Q}^{(0)} + \alpha_1 cfl \Delta \boldsymbol{Q}^{(0)}$$

$$\boldsymbol{Q}^{(2)} = \boldsymbol{Q}^{(0)} + \alpha_2 cfl \Delta \boldsymbol{Q}^{(1)}, \qquad \boldsymbol{Q}^{(3)} = \boldsymbol{Q}^{(0)} + \alpha_3 cfl \Delta \boldsymbol{Q}^{(2)}$$

$$\boldsymbol{Q}^{(4)} = \boldsymbol{Q}^{(0)} + \alpha_4 cfl \Delta \boldsymbol{Q}^{(3)}, \qquad \boldsymbol{Q}^{n+1} = \boldsymbol{Q}^{(4)} \tag{3}$$

with $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$ equal to $\{\frac{1}{4}, \frac{1}{3}, \frac{1}{2}, 1\}$. The $\Delta \boldsymbol{Q}^{(m)}$ of each stage is given by $\Delta \boldsymbol{Q}^{(m)} = \boldsymbol{Q}^{(m+1)^*} - \boldsymbol{Q}^{(m)}$, where $\boldsymbol{Q}^{(m+1)^*}$ is computed with the semi-implicit line method (4). Lines in the y direction, for example, $\boldsymbol{Q}^{(m+1)^*}$, are calculated from

$$\left(\frac{\bar{\bar{\Gamma}}}{\Delta \tau} + 2\overline{\overline{A_d}} + \frac{2\overline{R_{xx}}}{\Delta x^2}\right)\left(\boldsymbol{Q}^{(m+1)^*} - \boldsymbol{Q}^{(m)}\right) + \frac{\partial \boldsymbol{F}_c^{(m)}}{\partial x} + \frac{\partial \boldsymbol{F}_a^{(m)}}{\partial x}$$

$$- \frac{\partial \boldsymbol{F}_d^{(m)}}{\partial x} + \frac{\partial \boldsymbol{G}_c^{(m)}}{\partial y} + \frac{\partial \boldsymbol{G}_a^{(m+1)^*}}{\partial y} - \frac{\partial \boldsymbol{G}_d^{(m+1)^*}}{\partial x}$$

$$- L^{(m),(m+1)^*}(\boldsymbol{Q}) = \boldsymbol{S}^{(m),(m+1)^*} \tag{4}$$

$\overline{\overline{A_d}}$ is a 5×5 matrix with zero entries, except $A_{d_{11}} = \frac{1}{2}/\beta_x \Delta x$. $\boldsymbol{S}^{(m),(m+1)^*}$ denotes that a part of the source term can be treated implicitly and another part explicitly. This will be discussed in the next section. $L^{(m),(m+1)^*}(\boldsymbol{Q})$ is the discretized form of the operator, for example, for lines in y direction,

$$L^{(m),(m+1)^*}(\boldsymbol{Q}) = \frac{\partial}{\partial x}\left(R_{xx}\frac{\partial \boldsymbol{Q}^{(m)}}{\partial x}\right) + \frac{\partial}{\partial x}\left(R_{xy}\frac{\partial \boldsymbol{Q}^{(m)}}{\partial y}\right)$$

$$+ \frac{\partial}{\partial y}\left(R_{yx}\frac{\partial \boldsymbol{Q}^{(m)}}{\partial x}\right) + \frac{\partial}{\partial y}\left(R_{yy}\frac{\partial \boldsymbol{Q}^{(m+1)^*}}{\partial y}\right)$$

where R_{xx}, R_{xy}, R_{yx}, and R_{yy} are 5×5 matrices with zero entries, except

$$(R_{xx})_{22} = 2(\nu + \nu_t), \qquad (R_{xx})_{33} = \nu + \nu_t$$

$$(R_{xx})_{44} = \nu + \nu_t/\sigma_k, \qquad (R_{xx})_{55} = \nu + \nu_t/\sigma_\varepsilon$$

$$(R_{xy})_{32} = \nu + \nu_t, \qquad (R_{yx})_{23} = \nu + \nu_t, \qquad (R_{yy})_{22} = \nu + \nu_t$$

$$(R_{yy})_{33} = 2(\nu + \nu_t), \qquad (R_{yy})_{44} = \nu + \nu_t/\sigma_k$$

$$(R_{yy})_{55} = \nu + \nu_t/\sigma_\varepsilon$$

$\overline{R_{xx}}$ is defined as

$$\overline{R_{xx}} = \tfrac{1}{2}(R_{xx,L} + R_{xx,R})$$

If high grid $\mathcal{AR}$ exist in both directions, an alternating line solver is used. This means that two multistage cycles are performed. In the first one, $\boldsymbol{Q}^{(m+1)^*}$ is calculated by a sweep with lines in one direction and in the second one by a sweep with lines in the other direction.

If the multigrid technique is used, such a cycle is performed on each grid. Typically, W cycles have been used with four grids, with one multistage cycle on each grid, except on the coarsest grid, where four multistage cycles were performed. Note that for the calculation of the turbulence production terms on coarse grids, the data on those coarse grids have been used, rather than some restriction of fine-grid values.

The implemented solution procedure differs slightly from the method just described because the subsystem of the turbulence equations has been decoupled from the subsystem of the NS equations. The steady solution is obtained in the following way. First, a complete cycle is performed for the NS subsystem, providing an update for p', u, and v. The turbulence quantities are kept constant during this cycle. After this cycle, the flow quantities are injected onto the coarse grids, after which a cycle for the turbulent subsystem with constant flow quantities is performed. Then, the turbulent quantities are injected onto the coarse grids. In other words, one complete cycle for the coupled system of NS and turbulence equations has been replaced by one complete cycle for the NS equations, followed by a complete cycle for the turbulence equations. This results in solving a 3×3 and a 2×2 system instead of a 5×5 system.

The pseudotime step $\Delta \tau$ in Eq. (4) has contributions from the convective-diffusive terms and from the source term treatment. In the next subsection, the convective-diffusive part is studied. The source term contribution is studied in the next section.

Determination of the Pseudotime Step

Consider a uniform Cartesian mesh with constant Δx and Δy. The time step $\Delta \tau$ for a cell on this mesh is computed for an explicit method as

$$\Delta \tau_{\mathrm{NS}} = 1/[(u + c_x)/\Delta x + (v + c_y)/\Delta y]$$

with

$$c_x = \sqrt{(u^2 + \beta^2)}, \qquad c_y = \sqrt{(v^2 + \beta^2)}$$

Assume that the flow is inviscid and aligned to the x direction, that is, $v = 0$. If β is chosen in the order of u, all three eigenvalues [Eq. (2)] have the same order of magnitude in the x direction and all waves are convected into this direction with a Courant–Friedrichs–Lewy (CFL) number on the order of unity.

Note that if the allowable time step becomes smaller, the convergence will slow down. This happens for large grid $\mathcal{AR}$. We define the grid $\mathcal{AR}$ g_{ar} for the Cartesian grid as $g_{\mathrm{ar}} = \Delta x/\Delta y$. If g_{ar} is very large, the allowable time step $\Delta \tau$ is equal to $\Delta y/c_y$ and the maximum allowable CFL number in the x direction is

$$\mathrm{CFL}_x = \frac{(u + c_x)\Delta \tau}{\Delta x} = \frac{u + c_x}{c_y}\frac{1}{g_{\mathrm{ar}}} \approx \frac{1}{g_{\mathrm{ar}}}$$

This will lead to a slower convergence rate. If the acoustic fluxes in the y direction are discretized implicitly, the time-step definition is changed into

$$\Delta \tau_{\mathrm{NS}} = 1/[(u + c_x)/\Delta x + \omega_1 v/\Delta y] \tag{5}$$

where ω_1 is a scaling factor. If the flow is aligned to the x direction, CFL_x will be equal to unity. If the flow is aligned to the y direction, a good value for ω_1 is around two. This can be seen from a Fourier analysis.[4] When the viscous terms become important, the maximum allowable time step is determinated by the von Neumann number $\Delta \tau = \Delta y^2/[2(\nu + \nu_t)]$ and CFL_x becomes small.

If the viscous terms are treated with a line implicit method in y direction too, this von Neumann restriction on the time step disappears, and the CFL_x number is again on the order of unity. Therefore, a line implicit method will be used for the acoustic and viscous terms in the direction of the smallest grid distances. The expression for β is appropriately chosen by the use of the same philosophy. For example, in the case of small cell dimensions in the

y direction, the β factor in the preconditioning matrix is given by $\beta = \sqrt{(u^2 + v^2)} + 2(\nu + \nu_t)/\Delta x$. The method has been completely described for laminar flow.[4]

Treatment of Source Terms

First a scalar equation is studied. Afterward, the extension toward a system of equations is made. The analysis is performed as if a single-stage stepping were used. However, if the discretization is stable for a single stage, it may be assumed to be stable for a multistage stepping, too.

Scalar Equation

The discussion is first limited to linear source terms and then extended toward nonlinear source terms.

Linear Source Term

To investigate the influence of discretization schemes for source terms in differential equations, a simplified equation is considered first, which contains a linear source term

$$\frac{\mathrm{d}\phi}{\mathrm{d}\tau} = A\phi \tag{6}$$

where A is constant. It is well known that, when A is real, explicit treatment of the source term should be used if $A > 0$, whereas $A < 0$ requires an implicit treatment. If A is complex, A can be written as $A = a + jb$, with a and b real constants and $j = \sqrt{-1}$. There are four ways of discretizing now: both the real and the imaginary part can be treated explicitly or implicitly.

It is desirable that numerically the same behavior is obtained as for the analytical solution. The analytical solution is $\phi = \phi_0 e^{A\tau}$, with ϕ_0 determined by the initial condition ($t = 0$). If $a > 0$, the initial value will be amplified, which results in an exponential increase in time. If $a < 0$, the variable decays from its initial value toward zero, which corresponds to a stable process. If $b \neq 0$, the solution is oscillatory in time. The numerical behavior is described by means of a Neumann stability analysis, which yields the amplification factor

$$G = \phi^{n+1}/\phi^n \tag{7}$$

the value of which depends on the choice of discretization. Analytically, the following relationships exist:

$$a > 0 \Rightarrow |G| > 1, \qquad a < 0 \Rightarrow |G| < 1 \tag{8}$$

Numerically, the same behavior is desirable. As an example, the case $a > 0$ is considered. If both the real and imaginary part are treated explicitly, $|G| = \sqrt{[(1 + a\Delta\tau)^2 + b^2\Delta\tau^2]} > 1$ for all $\Delta\tau$. If the imaginary part is treated implicitly, as suggested by Merkle et al.,[12] $|G| = [(1 + a\Delta\tau)/\sqrt{(1 + b^2\Delta\tau^2)}]$, which can become smaller than 1, depending on the values of a, b, and $\Delta\tau$. This may improve the stability, which is particularly useful in (nonlinear) cases where the variable is not allowed to become infinitely large. The disadvantage is that the numerical behavior may differ from the analytical behavior in the sense that $|G| < 1$ is possible even when $a > 0$. Therefore, both the real and imaginary part are treated explicitly in this work when $a > 0$. Analogously, when $a < 0$, both the real and imaginary part are treated implicitly. This means that the conclusions for real source terms can be generalized: an explicit treatment is used when the real part is positive, whereas an implicit treatment is used when the real part is negative. Because of this strong analogy, the further analysis will be restricted to real source terms. The case of complex source terms has been mentioned here because, for a coupled system of equations, the eigenvalues of the Jacobian will take over the role of the constant A and, particularly for a k–ε system, these eigenvalues can be complex.

Nonlinear Source Term

The clear distinction of discretization in the preceding paragraphs is a consequence of the real parts of the source term $S(\phi)$ and $\partial S/\partial\phi$ always having the same sign. This does not hold any longer for a nonlinear source term, which makes the choice of discretization less evident. To investigate this, Eq. (6) is generalized to

$$\frac{\mathrm{d}\phi}{\mathrm{d}\tau} = S(\phi) \tag{9}$$

with S a nonlinear function in ϕ. Based on the source term $S(\phi)$, a nonzero steady solution can be obtained if a value $\phi_e \neq 0$ exists for which $S(\phi_e) = 0$. This happens in many physical processes. The steady solution is assumed to be stable in the sense that an infinitesimal perturbation from the steady solution does not alter the steady solution. This implies that $\partial S/\partial\phi < 0$ for $\phi = \phi_e$. In Fig. 1 a possible evolution of the source term $S(\phi)$ is given in a function of ϕ. The source term is positive for $0 < \phi < \phi_e$ and negative for $\phi > \phi_e$.

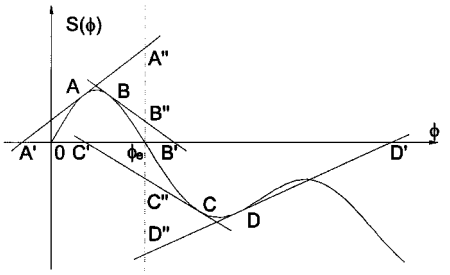

Fig. 1 Source term $S(\phi)$ as function of ϕ.

Analytically, this implies that ϕ_e is reached in a monotonic way. It also implies that, for any initial condition ϕ_0,

$$\text{if } S(\phi_0) > 0, \quad \text{then } \phi_e/\phi_0 > 1$$
$$\text{if } S(\phi_0) < 0, \quad \text{then } \phi_e/\phi_0 < 1 \tag{10}$$

Because of these two reasons, it seems a reasonable requirement that the discretization always results in an update $\delta\phi$ in accordance with the source term sign:

$$S(\phi) \geq 0 \Rightarrow \delta\phi \geq 0, \qquad S(\phi) \leq 0 \Rightarrow \delta\phi \leq 0 \tag{11}$$

There are two other requirements more specifically related to turbulence modeling. The first demand is that ϕ cannot become infinitely large, and the second one is that ϕ cannot become negative. To fulfill these three requirements, an appropriate discretization has to be looked for.

An implicit discretization of Eq. (9) results in

$$\frac{\phi^{n+1} - \phi^n}{\Delta\tau} = S(\phi^n) + \left.\frac{\partial S}{\partial\phi}\right|_{\phi^n} (\phi^{n+1} - \phi^n)$$

$$\phi^{n+1} = \phi^n + S(\phi^n)\Delta\tau \Big/ \left(1 - \left.\frac{\partial S}{\partial\phi}\right|_{\phi^n} \Delta\tau\right) \tag{12}$$

The explicit counterpart is given by

$$\phi^{n+1} = \phi^n + S(\phi^n)\Delta\tau \tag{13}$$

The amplification factors G_E and G_I are, respectively,

$$G_E = 1 + [S(\phi^n)/\phi^n]\Delta\tau_E \tag{14}$$

$$G_I = 1 + \left[\frac{S(\phi^n)}{\phi^n}\Delta\tau_I\right] \Big/ \left(1 - \left.\frac{\partial S}{\partial\phi}\right|_{\phi^n} \Delta\tau_I\right) \tag{15}$$

If an approximation $\tilde{\phi}$ of the steady solution ϕ_e can be made, the corresponding amplification factor $\tilde{G} = \tilde{\phi}/\phi^n$ is known. For such a prescribed amplification rate $\tilde{G}$, the corresponding time step is determined as

$$\frac{1}{\Delta\tau_E} = \frac{S(\phi^n)}{(\tilde{G} - 1)\phi^n} \tag{16}$$

$$\frac{1}{\Delta\tau_I} = \frac{S(\phi^n)}{(\tilde{G} - 1)\phi^n} + \left.\frac{\partial S}{\partial\phi}\right|_{\phi^n} \tag{17}$$

The implicit time step can also be written as

$$\frac{1}{\Delta\tau_I} = \frac{1}{\Delta\tau_E} + \left.\frac{\partial S}{\partial \phi}\right|_{\phi^n} \tag{18}$$

Hence, an unlimited implicit time step $\Delta\tau_I \to \infty$ corresponds to an explicit time step:

$$\frac{1}{\Delta\tau_E} = -\left.\frac{\partial S}{\partial \phi}\right|_{\phi^n}$$

Regardless of the time step, the numerical amplification should always be $1 < |G| < \infty$ if $S > 0$ and $0 < |G| < 1$ when $S < 0$ [Eq. (11)]. From the expressions (14) and (15), one can observe that, as opposed to the case of a linear source term, this cannot be guaranteed unconditionally, because it depends not only on the sign of the source term but also on the sign of the Jacobian $\partial S/\partial\phi$. To analyze this, the four different situations with respect to the signs of $S(\phi)$ and $\partial S/\partial\phi$ that can occur are studied. They are shown schematically in Fig. 1 by the points A, B, C, and D.

Point A, $S > 0$ *and* $\partial S/\partial\phi > 0$. The explicit treatment (14) always guarantees $|G| > 1$ independent of the time step. However, the solution $\phi^{n+1} \to \infty$ for $\Delta\tau \to \infty$, which is not desirable, because a finite, stable steady solution is assumed. The best time step should result in a solution close to the exact one. If an approximation $\tilde{\phi}$ can be made for the steady solution ϕ_e, the explicit time step can be calculated from Eq. (16).

The implicit treatment (15) demands $1/\Delta\tau_I \geq (\partial S/\partial\phi^n)|_{\phi^n}$ to assure $|G| > 1$. For the upper limit of the time step, the solution will also be $\phi^{n+1} \to \infty$.

An explicit treatment is the best choice for point A. If an approximation can be set forward, the necessary time step is calculated from Eq. (16).

Point B, $S > 0$ *and* $\partial S/\partial\phi < 0$. The situation is not much different from point A, except that within a linear approximation, a natural additional upper time limit can be proposed for the explicit discretization. With Taylor's expansion, the source term S is approximated as

$$S(\phi^{n+1}) \cong S(\phi^n) + \left.\frac{\partial S}{\partial \phi}\right|_{\phi^n} (\phi^{n+1} - \phi^n)$$

Because we want $S(\phi^{n+1}) = 0$, this results in

$$\phi^{n+1} = \phi^n - \left[S(\phi^n) \Big/ \frac{\partial S}{\partial \phi} \right]_{\phi^n}$$

which corresponds for an explicit treatment (14) to a time step $1/\Delta\tau_E = -(\partial S/\partial\phi)|_{\phi^n}$. The corresponding ϕ^{n+1} coincides with point B′ in Fig. 1. If, by a nonlinear theory, a better estimation $\tilde{\phi}$ can be found, the needed time step is given by Eq. (16). In most cases, however, some hypotheses have to be made to obtain this nonlinear approximation. To ensure a stable iteration process, the maximum allowed time step is taken as the smallest time step from the linear and the nonlinear analysis:

$$\frac{1}{\Delta\tau_E} = \mathrm{Max}\left[-\left.\frac{\partial S}{\partial \phi}\right|_{\phi^n}, \frac{S(\phi^n)}{(G-1)\phi^n} \right] \tag{19}$$

An implicit treatment (15) always fulfills the condition $|G| > 1$ and, moreover, guarantees $|G| < \infty$ for all $\Delta\tau_I$. Therefore, an implicit treatment is preferable for point B.

An implicit discretization with infinite time step, however, corresponds to a solution $\phi^{n+1} = \phi_{B'}$, which can be far away from ϕ_e. The time step to be taken for a nonlinear approximation $\tilde{\phi}$ is given by Eq. (17). For the same reasons as in the explicit discretization, the most conservative time step is taken:

$$\frac{1}{\Delta\tau_I} = \mathrm{Max}\left[0, \frac{S(\phi^n)}{(G-1)\phi^n} + \left.\frac{\partial S}{\partial \phi}\right|_{\phi^n} \right] \tag{20}$$

Using the identity (18), one can verify that the relations (19) and (20) actually express the same time-step restriction.

Point C, $S < 0$ *and* $\partial S/\partial\phi < 0$. Apart from $S(\phi) < 0$ and, hence, a desired amplification $0 < G < 1$, the conclusions and time-step restrictions are the same as in the preceding case. Implicit treatment is preferable. Note that the linear approximation (point C′) can be negative, so that the non-linear approximation could be even more important here.

Point D, $S < 0$ *and* $\partial S/\partial\phi > 0$. The situation of point D is very similar to the one of point A. The difference is that, as $S(\phi) < 0$ the amplification should be $0 < G < 1$. Explicit treatment is preferable.

Because a time-step restriction seems inevitable, at first glance, an implicit treatment does not seem useful. In practice, however, it is difficult to find $\tilde{G}$. An implicit treatment when $\partial S/\partial\phi < 0$ is then more robust. So as not to counteract the implicit treatment, Eqs. (19) and (20) are slightly altered. In Eq. (19), the sum is taken instead of the maximum. In view of the relationship (18), this corresponds to dropping the term $(\partial S/\partial\phi)|_{\phi^n}$ in Eq. (20). Based on this discussion, the time step to be taken in any case is given by

$$1/\Delta\tau = S(\phi^n)/(\tilde{G} - 1)\phi^n \tag{21}$$

combined with an explicit treatment of the source term $S(\phi)$ if $\partial S/\partial\phi > 0$, and an implicit one if $\partial S/\partial\phi < 0$.

The practical use of Eq. (21) requires that a reasonable approximation $\tilde{G}$ can be determined. This is the topic of the next section. Also, note that the equation considered here does not contain any convective or diffusive terms. In practice, their presence drastically diminishes the importance of the introduced time step. The conclusions about the choice between explicit and implicit treatment, however, remain unchanged, as will be discussed in the next paragraph and in the numerical results.

Coupled System

The source terms in two-equation turbulence models are, in general, strongly nonlinear and coupled. Two major methods exist to describe the source terms.

In the first method, positive parts of the source terms are not linearized and an approximate Jacobian is constructed for the negative parts of the source terms, in such a way that negative real eigenvalues are obtained. The positive parts are treated explicitly, and the negative parts are treated implicitly. This method has proven to work well for different kinds of flow problems such as flat plates, backward facing steps, and turbine cascades. One way of constructing an approximated Jacobian is described in several works.[1,2,5] For the k–ε model developed by Yang and Shih[13]

$$\frac{\partial k}{\partial \tau} + \frac{\partial}{\partial x_k}(k v_k) = P_k - \varepsilon + \frac{\partial}{\partial x_k}\left[\left(\nu + \frac{\nu_t}{\sigma_k} \right) \frac{\partial k}{\partial x_k} \right]$$

$$\frac{\partial \varepsilon}{\partial \tau} + \frac{\partial}{\partial x_k}(\varepsilon v_k) = (c_{\varepsilon 1} P_k - c_{\varepsilon 2} f_2 \varepsilon)\frac{1}{T_t} + \frac{\partial}{\partial x_k}\left[\left(\nu + \frac{\nu_t}{\sigma_\varepsilon} \right) \frac{\partial \varepsilon}{\partial x_k} \right] + E$$

the source terms are

$$S_k = P_k - \varepsilon, \qquad S_\varepsilon = (c_{\varepsilon 1} P_k - c_{\varepsilon 2} f_2 \varepsilon)(1/T_t) + E$$

P_k denotes the turbulence energy production term

$$P_k = \nu_t \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) \frac{\partial v_i}{\partial x_j} \tag{22}$$

with the Einstein summation convention and

$$\nu_t = c_\mu f_\mu k T_t$$

In this expression, $c_\mu = 0.09$, f_μ is a damping function, and

$$T_t = k/\varepsilon + \sqrt{\nu/\varepsilon}$$

Other model constants are $c_{\varepsilon 1} = 1.44$ and $c_{\varepsilon 2} = 1.92$. Also, f_2 is a damping function and $E = \nu\nu_t[(\partial^2 v_i/\partial x_k \partial x_j)(\partial^2 v_i/\partial x_k \partial x_j)]$. The approximate Jacobian is given by[5]

$$\frac{\partial \boldsymbol{S}^-(k,\varepsilon)}{\partial(k,\varepsilon)} = \begin{bmatrix} \frac{\partial S_k^-}{\partial k} & \frac{\partial S_k^-}{\partial \varepsilon} \\ \frac{\partial S_\varepsilon^-}{\partial k} & \frac{\partial S_\varepsilon^-}{\partial \varepsilon} \end{bmatrix}$$

$$= \begin{bmatrix} -\frac{\left(\sqrt{\nu} + \sqrt{\nu + 4kT_t}\right)^2}{2kT_t^2} & 0 \\ c_{\varepsilon 2} f_2 \frac{1}{T_t^2} & -c_{\varepsilon 2} f_2 \frac{1}{T_t^2}\left(2T_t - \frac{1}{2}\sqrt{\frac{\nu}{\varepsilon}}\right) \end{bmatrix} \tag{23}$$

The superscript − denotes that only the negative parts of the source terms are taken into account for the construction of the approximate Jacobian. For the low-Reynolds version of the k–ω model developed by Wilcox[2]

$$\frac{\partial k}{\partial \tau} + \frac{\partial}{\partial x_k}(k v_k) = P_k - \beta^* k\omega + \frac{\partial}{\partial x_k}\left[(\nu + \sigma_k \nu_t)\frac{\partial k}{\partial x_k}\right]$$

$$\frac{\partial \omega}{\partial \tau} + \frac{\partial}{\partial x_k}(\omega v_k) = \alpha\frac{\omega}{k}P_k - \beta\omega^2 + \frac{\partial}{\partial x_k}\left[(\nu + \sigma_\omega \nu_t)\frac{\partial \omega}{\partial x_k}\right]$$

the source terms are

$$S_k = P_k - \beta^* k\omega, \qquad S_\omega = \alpha(\omega/k)P_k - \beta\omega^2$$

The model constants are

$$\alpha = \frac{5}{9}\frac{\alpha_0 + Re_T/R_\omega}{1 + Re_T/R_\omega}(\alpha^*)^{-1}, \qquad \alpha^* = \frac{\alpha_0^* + Re_T/R_k}{1 + Re_T/R_k}$$

$$\beta^* = 0.09\frac{(5/18) + (Re_T/R_\beta)^4}{1 + (Re_T/R_\beta)^4}, \qquad Re_T = \frac{k}{\omega\nu}$$

and $\beta = 0.075$, $\alpha_0^* = \beta/3$, $\alpha_0 = 0.1$, $R_\beta = 8$, $R_k = 6$, and $R_\omega = 2.7$. The approximate Jacobian is[2]

$$\frac{\partial \boldsymbol{S}^-(k,\omega)}{\partial(k;\omega)} = \begin{bmatrix} -2\beta^*\omega & 0 \\ 0 & -2\beta\omega \end{bmatrix} \tag{24}$$

A second method starts with the construction of the exact Jacobian of the source terms $\partial \boldsymbol{S}/\partial\phi$ and was suggested by Merkle et al.[12] For the k–ε system, this yields

$$\frac{\partial \boldsymbol{S}(k,\varepsilon)}{\partial(k;\varepsilon)} = \begin{bmatrix} \frac{\partial S_k}{\partial k} & \frac{\partial S_k}{\partial \varepsilon} \\ \frac{\partial S_\varepsilon}{\partial k} & \frac{\partial S_\varepsilon}{\partial \varepsilon} \end{bmatrix} \tag{25}$$

with

$$\frac{\partial S_k}{\partial k} = \frac{\partial P_k}{\partial k}, \frac{\partial S_k}{\partial \varepsilon} = \frac{\partial P_k}{\partial \varepsilon} - 1, \frac{\partial S_\varepsilon}{\partial k}$$

$$= \left[c_{\varepsilon 1}\left(\frac{\partial P_k}{\partial k}\right) - c_{\varepsilon 2}\varepsilon\left(\frac{\partial f_2}{\partial k}\right)\right]\left(\frac{1}{T_t}\right)$$

$$+ (c_{\varepsilon 1}P_k - c_{\varepsilon 2}f_2\varepsilon)\left[\frac{\partial(1/T_t)}{\partial k}\right] + \left(\frac{\partial E}{\partial k}\right)$$

$$\frac{\partial S_\varepsilon}{\partial \varepsilon} = \left[c_{\varepsilon 1}\left(\frac{\partial P_k}{\partial \varepsilon}\right) - c_{\varepsilon 2}\varepsilon\left(\frac{\partial f_2}{\partial \varepsilon}\right) - c_{\varepsilon 2}f_2\right]\left(\frac{1}{T_t}\right)$$

$$+ (c_{\varepsilon 1}P_k - c_{\varepsilon 2}f_2\varepsilon)\left[\frac{\partial(1/T_t)}{\partial \varepsilon}\right] + \left(\frac{\partial E}{\partial \varepsilon}\right)$$

For the k–ω system, the exact Jacobian is

$$\frac{\partial \boldsymbol{S}(k,\omega)}{\partial(k;\omega)} =$$

$$\begin{bmatrix} \frac{\partial P_k}{\partial k} - \beta^*\omega - \frac{\partial \beta^*}{\partial k}\omega k & \frac{\partial P_k}{\partial \omega} - \beta^* k - \frac{\partial \beta^*}{\partial \omega}\omega k \\ \frac{\partial}{\partial k}\left(\alpha\frac{\omega}{k}\right)P_k + \alpha\frac{\omega}{k}\frac{\partial P_k}{\partial k} & \frac{\partial}{\partial \omega}\left(\alpha\frac{\omega}{k}\right)P_k + \alpha\frac{\omega}{k}\frac{\partial P_k}{\partial \omega} - 2\beta\omega \end{bmatrix} \tag{26}$$

This exact Jacobian can easily be diagonalized into its eigenvalue matrix $\bar{\bar{\Lambda}}$. This matrix can be split into a positive part and a negative part: $\bar{\bar{\Lambda}} = \bar{\bar{\Lambda}}^+ + \bar{\bar{\Lambda}}^-$, with

$$\bar{\bar{\Lambda}}^\pm = \begin{bmatrix} \lambda_1^\pm & 0 \\ 0 & \lambda_2^\pm \end{bmatrix}$$

Here, λ_i^- $(i = 1, 2)$ is equal to λ_i if the real part of λ_i is negative, and is equal to zero otherwise. Also, $\lambda_i^+ = \lambda_i - \lambda_i^-$. Transforming back yields a splitting of the Jacobian $\partial \boldsymbol{S}/\partial\phi = \partial \boldsymbol{S}^+/\partial\phi + \partial \boldsymbol{S}^-/\partial\phi$. The different terms are then treated according to the earlier analysis: implicitly for the negative part, explicitly for the positive part. If the eigenvalues are complex, the sign of the real part is considered in accordance with the analysis for the linear source term scalar equation.

As for a scalar equation, an approximation for the time step for each equation of the coupled system can be made if the amplification factors corresponding with the end solution can be estimated. To allow an approximate analytical solution, we suggest simplifying the source terms. This yields a relationship between the amplification factors G_k and G_ε (or G_ω). For the k–ε model, the model (22) is rewritten as

$$\frac{\partial k}{\partial \tau} = P_k - \varepsilon, \qquad \frac{\partial \varepsilon}{\partial \tau} = c_{\varepsilon 1}\frac{\varepsilon}{k}(P_k - \varepsilon)$$

where the convective and diffusive terms are neglected and the source term for the ε equation is simplified. Division of the two equations gives a relationship between the amplification factors:

$$\frac{\mathrm{d}k}{\mathrm{d}\varepsilon} = \frac{k}{c_{\varepsilon 1}\varepsilon} \Rightarrow \ell n(k^{c_{\varepsilon 1}})\big|_n^{n+1} = \ell n(\varepsilon)\big|_n^{n+1} \Rightarrow \left(\frac{k^{n+1}}{k^n}\right)^{c_{\varepsilon 1}} = \frac{\varepsilon^{n+1}}{\varepsilon^n}$$

leading to

$$G_\varepsilon = G_k{}^{c_{\varepsilon 1}} \tag{27}$$

P_k is proportional to ν_t, Eq. (22). The simplification $\nu_t = c_\mu(k^2/\varepsilon)$ implies $P_k \sim (k^2/\varepsilon)$, so that

$$P_k^{n+1}\big/P_k^n = G_k^2\big/G_\varepsilon$$

The purpose is to obtain a zero source term on time level $n + 1$:

$$P_k^{n+1} - \varepsilon^{n+1} = 0 \Rightarrow \left(G_k^2\big/G_\varepsilon\right)P_k^n - G_\varepsilon\varepsilon^n = 0$$

The amplification rates are then, by the use of Eq. (27),

$$G_k = \left(P_k^n\big/\varepsilon^n\right)^{1/2(c_{\varepsilon 1}-1)}, \qquad G_\varepsilon = \left(P_k^n\big/\varepsilon^n\right)^{c_{\varepsilon 1}/2(c_{\varepsilon 1}-1)}$$

A similar estimation for the amplification rates can be obtained for the k–ω turbulence models. The following approximations have been made:

$$\frac{\partial k}{\partial \tau} = P_k - \beta^* k\omega, \qquad \frac{\partial \omega}{\partial \tau} = \alpha\frac{\omega}{k}(P_k - \beta\omega k)$$

Because $\beta \approx \beta^*$, as in the k–ε system, the relationship

$$\frac{\mathrm{d}k}{\mathrm{d}\omega} = \frac{k}{\alpha\omega}$$

is obtained, which leads to $G_\omega = G_k^\alpha$. Then it is desirable that the source term at pseudotime level $n + 1$ become zero again. With P_k again proportional to $\nu_T = \alpha^* k/\omega$, so that $P_k^{n+1}/P_k^n = G_k/G_\omega$, this results in

$$G_k = \left(\frac{P_k^n}{\beta^* k^n \omega^n}\right)^{1/2\alpha}, \qquad G_\omega = \left(\frac{P_k^n}{\beta^* k^n \omega^n}\right)^{\frac{1}{2}}$$

Now that the amplification matrix

$$\bar{\bar{G}} = \begin{bmatrix} G_k & 0 \\ 0 & G_\varepsilon \end{bmatrix}$$

is known, the corresponding time-step matrix

$$\overline{\overline{\Delta\tau_t}} = \begin{bmatrix} \Delta\tau_k & 0 \\ 0 & \Delta\tau_\varepsilon \end{bmatrix} \tag{28}$$

for the k and ε (or ω) equation can be calculated from

$$(\overline{\overline{\Delta\tau_t}})^{-1}(\bar{\bar{G}} - \bar{\bar{I}})\phi^n = S(\phi^n) \tag{29}$$

similar to Eq. (21), with $\bar{\bar{I}}$ the unity matrix.

Because the turbulent quantities can still become negative during the iteration process, particularly in the beginning, corrections have to be applied when this occurs. For k, a threshold value is prescribed when k becomes negative. For ε or ω, an averaging of the values in the neighboring nodes is performed. If the latter does not supply a positive value, no update for ε or ω is performed.

Coupled System with Convection and Diffusion

In the preceding analysis, convective and diffusive terms were ignored. However, these terms are normally present and impose time-step restrictions for stability reasons. To retain stability, in principle the minimum of the time-step restriction imposed by the source terms and the one imposed by the convection and/or diffusion should be taken. As a practical implementation, instead of taking the minimum, the inverses of all of the time steps are added, which results in a safer time step when all of the time steps are of the same order. When one of the restrictions is much more severe than the others, the addition of the inverses of the time steps is approximately the same as taking the minimum. Therefore, the final time-step matrix for the turbulence equations is

$$\overline{\overline{\Delta\tau}} = \begin{bmatrix} \dfrac{1}{\Delta\tau_{\mathrm{NS}}} + \dfrac{1}{\Delta\tau_k} & 0 \\ 0 & \dfrac{1}{\Delta\tau_{\mathrm{NS}}} + \dfrac{1}{\Delta\tau_\varepsilon} \end{bmatrix}^{-1} \tag{30}$$

with $\Delta\tau_{\mathrm{NS}}$ (for a line in the y direction) given by Eq. (5).

Numerical Results

Both source term discretization methods are numerically investigated for fully developed channel flow,[14] flat-plate flow,[15] and backward-facing step (BFS) flow.[16] The low-Reynolds k–ε model by Yang and Shih[13] and the low-Reynolds k–ω model by Wilcox[2] are studied. The flows are incompressible. From now on, the method based on the approximated Jacobian [Eq. (23) or (24)] will be called the approximate method, and the method based on the exact Jacobian [Eq. (25) or (26)] will be called the rigorous method.

Unless otherwise stated, the global CFL number in Eq. (3) is 1.4. In principle, an alternating line solver has been used, except for the channel flow, where only lines in the y direction have been used. When the MG technique is used, it is always a W cycle with four grids. The evolution of the maximum residual is shown in terms of work units, where one work unit has been defined as the time necessary for a single stage (in the multistage scheme) calculation on the finest grid with lines in one direction.

For the channel flow, the convective terms and the diffusive terms in the streamwise direction are set to zero, so that those terms do not prescribe a time-step restriction. This way, the source term system's behavior can be investigated. Figure 2 gives the convergence evolution for a fully developed channel flow with $Re_h = 8 \times 10^4$ for both turbulence models. The grid has 1×129 nodes with the first grid point at $y^+ = u_\tau y/\nu = 1$, with y being the distance from the wall. The approximate method (curves 1 and 4) results in a good convergence rate. For the rigorous method without the time-step restriction (29), however, the CFL factor in the multistaging Eq. (3) has to be lowered to 0.5 for the rigorous method for stability reasons. This results in a poorer convergence rate, both for k–ε and k–ω (curves 2 and 5). When the time-step restriction is introduced, the lowering of the CFL number is not necessary, which results in a better convergence rate again (curves 3 and 6). This confirms the need for the time-step restriction to prevent the turbulent quantities becoming infinite or negative. The method based on Eq. (23) or (24) does not suffer from this difficulty because for that method there is always a negative part, treated implicitly, even when both eigenvalues of the exact Jacobian have positive real parts. This is equivalent to the introduction of a time-step restriction. Figure 3 shows the dimensionless velocity profile u/u_τ as a function of y^+. The profiles match the direct numerical simulation (DNS) profile very well. The deviation from the DNS data of the velocity profile obtained with the k–ω model is due to that model.[2] Calculations on a refined grid (1×257 nodes) provide the same results, which indicates grid independence of the results.

Figure 4 shows the convergence history for a flat-plate flow on a stretched grid (145×89 gridpoints). The approximate method and the rigorous method perform equally well (curves 1 and 2 and 4 and 5). The reason is that the convective and diffusive time steps (which contain grid dimensions) are more restrictive than the source term time step in Eq. (30). For the same reason, the introduction of the

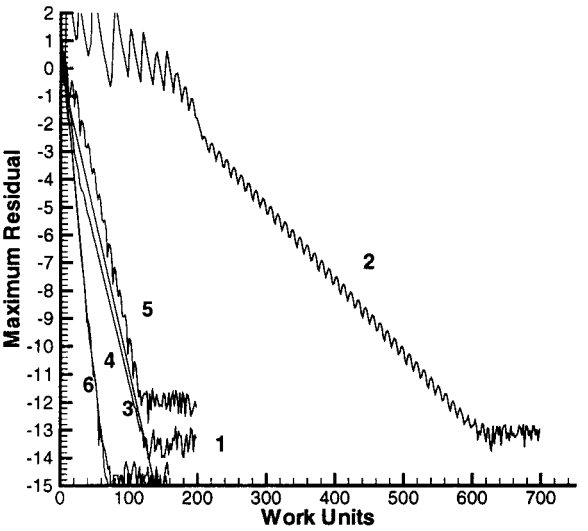

Fig. 2 Convergence history for channel flow: k–ε, approximated Jacobian, line 1; k–ε, exact Jacobian, line 2; k–ε, exact Jacobian, with $\overline{\overline{\Delta\tau}}$ from Eq. (29), line 3; k–ω, approximate Jacobian, line 4; k–ω, exact Jacobian, line 5; and k–ω, exact Jacobian, with $\overline{\overline{\Delta\tau}}$ from Eq. (29), line 6.

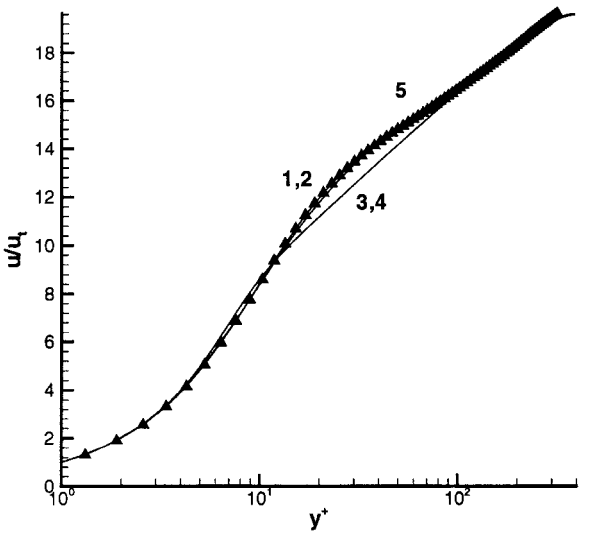

Fig. 3 Dimensionless velocity profile for fully developed channel flow: k–ε, line 1; k–ε on refined grid, line 2; k–ω, line 3; k–ω on refined grid, line 4; and DNS data (symbols), line 5.

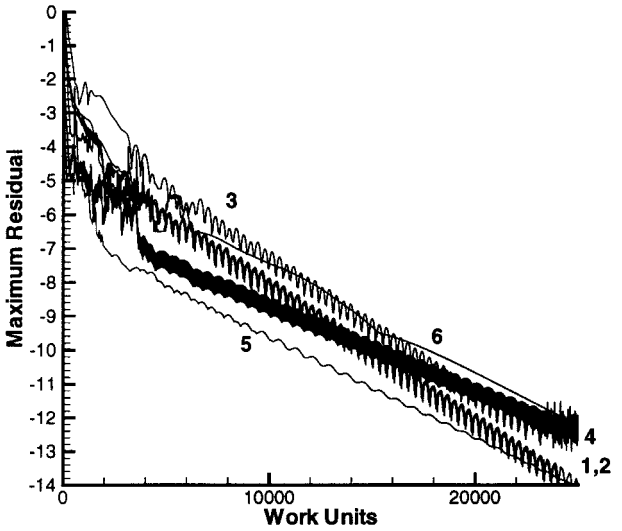

Fig. 4 Single grid (SG) convergence history for flat-plate flow: k–ε, approximated Jacobian, line 1; k–ε, exact Jacobian, line 2; k–ε, exact Jacobian, with $\overline{\overline{\Delta\tau}}$ from Eq. (30), line 3; k–ω, approximate Jacobian, line 4; k–ω, exact Jacobian, line 5; and k–ω, exact Jacobian, with $\overline{\overline{\Delta\tau}}$ from Eq. (30), line 6.

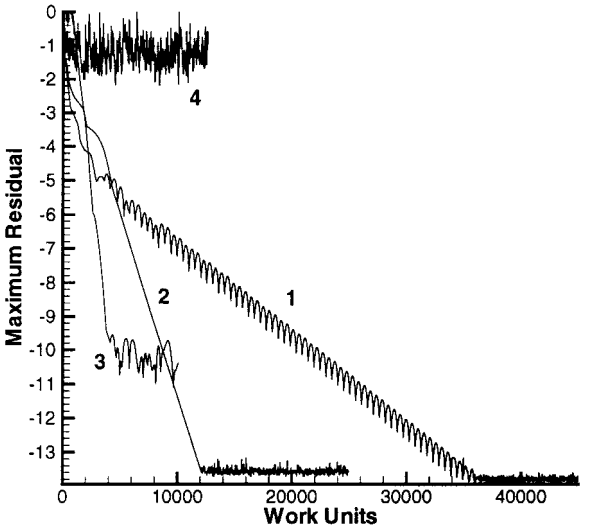

Fig. 5 Convergence acceleration with the multigrid technique for flat-plate flow: SG, line 1; NS solved with MG, turbulence equations SG, line 2; all equations with MG, rigorous method, line 3; and all equations with MG, approximate method, line 4.

calculated time step (29) is not necessary here (curves 3 and 6). It can be seen that introducing the extra time-step restriction leads to a slightly worse convergence rate.

Figure 5 shows convergence results for the k–ε model for a flat plate-flow[15] (193 × 97 gridpoints) using MG. The convergence rate when all of the equations are solved only on the finest grid is given by curve 1. Curve 2 corresponds to the NS equations solved with MG, whereas the turbulence equations are only computed on the finest grid. A substantial improvement is obtained with the rigorous method when both the NS equations and the turbulence equations are solved on the coarse grids (curve 3). However, no convergence at all can be obtained when the approximate method is used in combination with MG for all of the equations (curve 4). This indicates the superiority of the rigorous method because that method does not suffer from this convergence problem. Note that, again, the introduction of the time-step restriction (29) is not necessary to keep the rigorous method stable. For the k–ω model, the difference between the approximate and the rigorous method is much smaller. The reason is that, because of the form of the turbulence equations, the approximate and exact Jacobian do not differ as much as for the

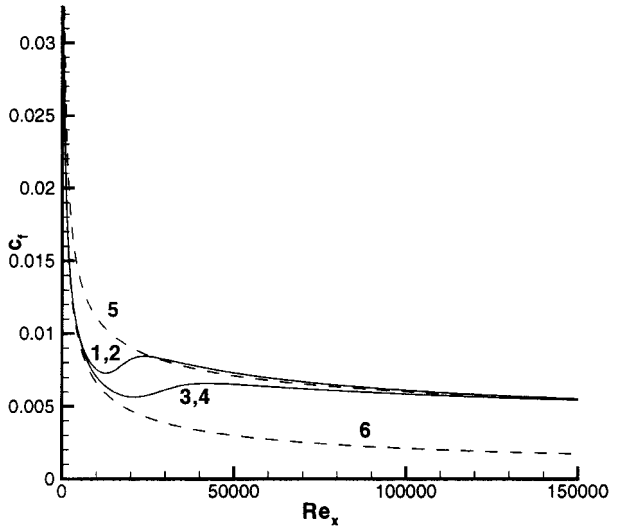

Fig. 6 Evolution of friction coefficient for flat plate flow: k–ε model, line 1; k–ε model, refined grid, line 2; k–ω model, line 3; k–ω model, refined grid, line 4; turbulent c_f, line 5; and laminar c_f, line 6.

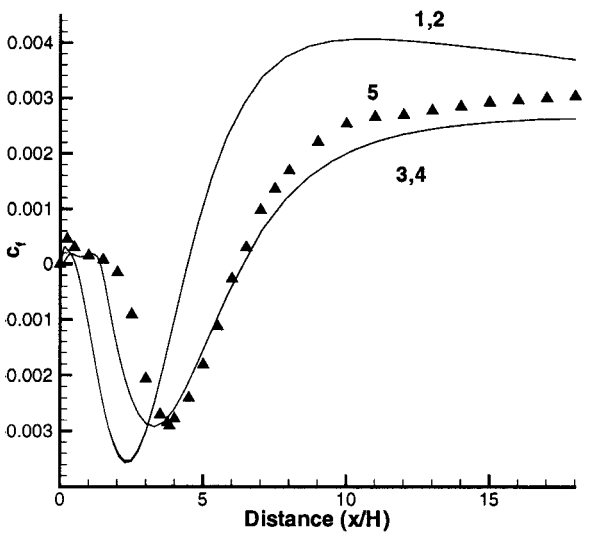

Fig. 7 Friction coefficient for BFS flow: k–ε model, line 1; k–ε model, refined grid, line 2; k–ω model, line 3; k–ω model, refined grid, line 4; and DNS (symbols), line 5.

k–ε model. Both methods converge with MG for the k–ω model, again leading to a substantial improvement on single grid calculations (results not shown). Figure 6 shows the evolution of the friction coefficient as a function of $Re_x = u_\infty x/\nu$. The theoretical laminar and turbulent evolution of the friction coefficient[17] are also shown for comparison. Calculations have also been performed on a refined grid (385 × 193 gridpoints) to check grid independence. The results coincide, which proves that the results are indeed grid independent.

Finally, a turbulent flow[16] over a BFS has been computed on a stretched grid (113 × 137 gridpoints). Figure 7 shows the evolution of the friction coefficient c_f as a function of the dimensionless distance x/H from the step. Again, calculations have been performed on a refined grid (225 × 273 points) to check grid independence. The results practically coincide again, indicating the grid independence of the results. DNS results are also shown for comparison. The differences between calculation results and DNS results are not due to numerical inaccuracy, but are due to well-known deficiencies of two-equation turbulence models. A detail of the calculated flowfield near the step is shown for the k–ε model in Fig. 8. Figure 9 shows convergence results for the k–ε model. Very similar conclusions can be drawn to those for the flat plate with respect to convergence rates. However, now the approximate method also converges with MG and performs equally as well as the rigorous method (curves 3 and 4). Similar results are obtained for the k–ω model (Fig. 10).

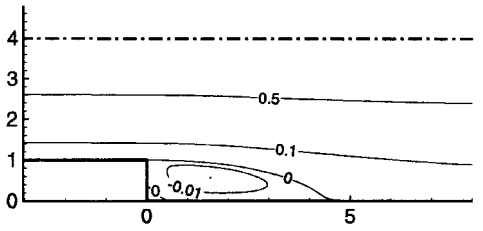

Fig. 8 Streamlines for BFS flow (k–ε model).

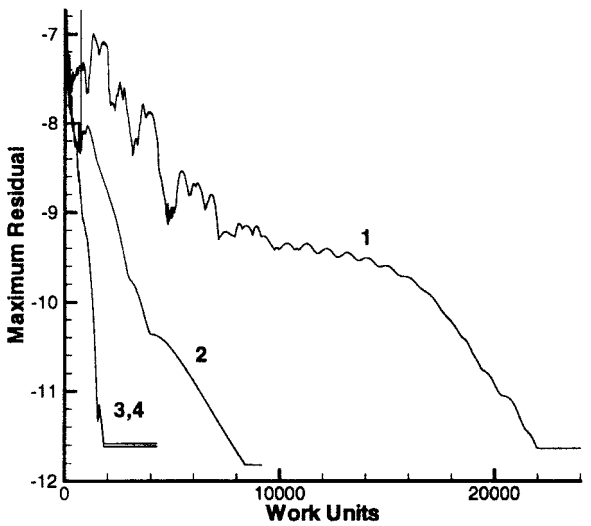

Fig. 9 Convergence acceleration with the multigrid technique for BFS flow for the k–ε model: SG, line 1; NS solved with MG, turbulence equations SG, line 2; all equations with MG, rigorous method, line 3; and all equations with MG, approximate method, line 4.

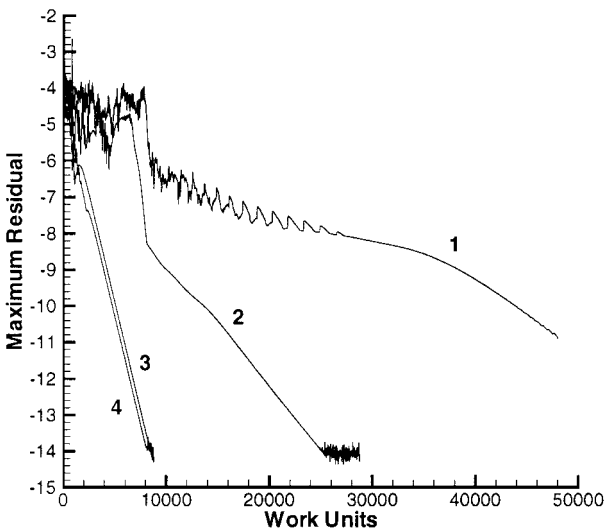

Fig. 10 Convergence acceleration with the multigrid technique for BFS flow for the k–ω model: SG, line 1; NS solved with MG, turbulence equations SG, line 2; all equations with MG, rigorous method, line 3; and all equations with MG, approximate method, line 4.

As mentioned by Gerlinger and Brüggemann,[8] the flat-plate flow seems to be more demanding for the multigrid technique. The reason is probably to be found in the grid alignment of the flow and the more extreme grid AR for the calculation of the flat-plate flow compared to the BFS flow. This indicates that the rigorous method is more capable of dealing with high AR than the approximate method.

Conclusions

A general method for discretizing the source terms in turbulence models has been presented. It is based on the eigenvalues of the exact source term Jacobian and allows the use of the MG technique on both the NS equations and the turbulence equations, without any damping of coarse grid corrections or freezing of parts of the source terms on coarse grids. This substantially increases the convergence speed and allows the extension of a robust, grid-independent solving technique developed for laminar flows, toward turbulent flows. The presented method is capable of dealing with high AR, as encountered in grids necessary to resolve turbulent boundary layers. In principle, the mathematical analysis indicates the need for a time-step restriction for the source terms to keep the suggested method stable, but in practice the time-step limitations coming from convective and diffusive terms are more restrictive, so that this necessity has not been observed.

Acknowledgments

The research reported here was performed under Contract G.0283.96 by the Fund for Scientific Research (FWO), Flanders, Belgium. Part of the work is related to research activities, undertaken in the ESA's facilities in the framework of a fellowship granted through the Training and Mobility of Researchers Programme financed by the European Community. The first author works as a Research Assistant of the FWO.

References

[1] Vandromme, D., and Ha Minh, H., "About the Coupling of Turbulence Closure Models with Averaged Navier–Stokes Equations," *Journal of Computational Physics*, Vol. 65, No. 2, 1986, pp. 386–409.

[2] Wilcox, D. C., *Turbulence Modeling for CFD*, Griffin, Glendale, CA, 1993, pp. 147–152, 156, and 302.

[3] Mavriplis, D. J., "Multigrid Strategies for Viscous Flow Solvers on Anisotropic Unstructured Meshes," *Proceedings of the 13th AIAA Computational Fluid Dynamics Conference*, AIAA, Reston, VA, 1997, pp. 659–675.

[4] Vierendeels, J., Riemslagh, K., and Dick, E., "A Multigrid Semi-Implicit Line-Method for Viscous Incompressible and Low-Mach-Number Flows on High Aspect Ratio Grids," *Journal of Computational Physics*, Vol. 154, No. 2, 1999, pp. 310–341.

[5] Steelant, J., and Dick, E., "A Multigrid Method for the Compressible Navier–Stokes Equations Coupled to the k–ε Turbulence Equations," *International Journal of Numerical Methods in Heat and Fluid Flow*, Vol. 4, No. 2, 1994, pp. 99–113.

[6] Steelant, J., Dick, E., and Pattijn, S., "Analysis of Robust Multigrid Methods for Steady Viscous Low Mach Number Flows," *Journal of Computational Physics*, Vol. 136, No. 2, 1997, pp. 603–628.

[7] Lien, F. S., and Leschziner, M. A., "Multigrid Acceleration for Recirculating and Turbulent Flows Computed with a Non-Orthogonal, Collocated Finite-Volume Scheme," *Computer Methods in Applied Mechanical Engineering*, Vol. 118, 1994, pp. 351–371.

[8] Gerlinger, P., and Brüggemann, D., "Multigrid Convergence Acceleration for Turbulent Supersonic Flows," *International Journal for Numerical Methods in Fluids*, Vol. 24, No. 10, 1997, pp. 1019–1035.

[9] Liou, M. S., and Steffen, C. J., "A New Flux Splitting Scheme," *Journal of Computational Physics*, Vol. 107, No. 1, 1993, pp. 23–39.

[10] Dick, E., and Linden, J., "A Multigrid Method for Steady Incompressible Navier–Stokes Equations Based on Flux Difference Splitting," *International Journal of Numerical Methods for Fluids*, Vol. 14, No. 11, 1992, pp. 1311–1323.

[11] Weiss, J., and Smith, W., "Preconditioning Applied to Variable and Constant Density Flows," *AIAA Journal*, Vol. 33, No. 11, 1995, pp. 2050–2057.

[12] Merkle, C. L., Deshpande, M., and Venkateswaran, S., "Efficient Implementation of Turbulence Modeling in Computational Schemes," *Proceedings of the Second U.S. National Congress on Computational Mechanics*, Pergamon, Oxford, 1993, p. 19.

[13] Yang, Z. Y., and Shih, T. H., "New Time-Scale-Based k–ε Model for Near-Wall Turbulence," *AIAA Journal*, Vol. 31, No. 7, 1993, pp. 1191–1198.

[14] Kim, J., Moin, P., and Moser, R., "Turbulence Statistics in Fully Developed Channel Flow at Low Reynolds Number," *Journal of Fluid Mechanics*, Vol. 177, 1987, pp. 133–166.

[15] Yang, Z. Y., and Voke, P. R., "Large-Eddy Simulation Studies of Bypass Transition," *Engineering Turbulence Modelling and Experiments 2*, edited by W. Rodi and F. Martelli, Elsevier, Amsterdam, 1993, pp. 603–611.

[16] Le, H., Moin, P., and Kim, J., "Direct Numerical Simulation of Turbulent Flow over a Backward-Facing Step," *Journal of Fluid Mechanics*, Vol. 333, 1997, pp. 349–374.

[17] Schlichting, H., *Boundary Layer Theory*, 4th ed., Mechanical Engineering Series, McGraw–Hill, New York, 1979, pp. 139 and 636–640.

P. Givi
Associate Editor