

Chapter 7

Numerical Considerations

Modern turbulence model equations pose special numerical difficulties that must be understood in order to obtain reliable numerical solutions, even for boundary-layer flows where the equations are parabolic. For one-equation, two-equation and second-order closure models, these difficulties include stiffness caused by the presence of an additional time scale, singular behavior near solid boundaries, and non-analytical behavior at sharp turbulent/nonturbulent interfaces. This chapter focuses on these difficulties and on the solution methods for turbulence-model equations that have evolved.

7.1 Multiple Time Scales and Stiffness

One key issue that must be addressed in developing a numerical algorithm for fluid-flow problems is that of the physically relevant time scales. Taking proper account of these time scales is a necessary condition for numerical accuracy. For example, when we deal with non-chemically-reacting laminar flow, there are two distinct time scales corresponding to different physical processes. If L and U denote characteristic length and velocity for the flowfield, a is sound speed and ν is kinematic viscosity, the time scales are:

- Wave propagation, $t_{wave} \sim L/|U \pm a|$
- Molecular diffusion, $t_{diff} \sim L^2/\nu$

When we use turbulence transport equations, we have yet another time scale corresponding to the rate of decay of turbulence properties. In terms of the specific dissipation rate, $\omega \sim \epsilon/k$, this time scale is:

- Dissipation, $t_{diss} \sim 1/\omega \sim k/\epsilon$

Any numerical algorithm designed for use with turbulence transport equations should reflect all three of these time scales.

In terms of the Reynolds number, $Re_L = UL/\nu$, and the Mach number, $M = U/a$, the ratio of t_{diff} to t_{wave} is given by

$$\frac{t_{diff}}{t_{wave}} \sim \frac{|M \pm 1|Re_L}{M} \quad (7.1)$$

Clearly, for high Reynolds number flows the diffusion time scale is much longer than the wave propagation time scale regardless of Mach number. Diffusion will generally be important over very short distances such as the thickness of a boundary layer, δ , i.e., when $L \sim \delta$. For specified freestream Mach and Reynolds numbers, the relative magnitudes of the diffusion and wave propagation time scales are more-or-less confined to a limited range. This is not the case for the dissipation time scale.

The specific dissipation rate can vary by many orders of magnitude across a turbulent boundary layer. Consequently, in the same flow, t_{diss} can range from values much smaller than the other time scales to much larger. This is a crude reflection of the physical nature of turbulence, which consists of a wide range of frequencies. Thus, regardless of the flow speed, we should expect the dissipation time to have a nontrivial impact on numerical algorithms.

Because of the multiplicity of time scales attending use of turbulence transport equations, especially two-equation models and second-order closure models, we must contend with an unpleasant feature known as **stiffness**. An equation, or system of equations, is said to be stiff when there are two or more very different scales of the independent variable on which the dependent variables are changing. For example, consider the equation

$$\frac{d^2y}{dt^2} = 100y \quad (7.2)$$

The general solution to this equation is

$$y(t) = Ae^{-10t} + Be^{10t} \quad (7.3)$$

If we impose the initial conditions

$$y(0) = 1 \quad \text{and} \quad \dot{y}(0) = -10 \quad (7.4)$$

the exact solution becomes

$$y_{exact}(t) = e^{-10t} \quad (7.5)$$

Unfortunately, any roundoff or truncation error in a numerical solution can excite the e^{10t} factor, viz., we can inadvertently wind up with

$$y_{\text{numerical}}(t) = e^{-10t} + \epsilon e^{10t}, \quad |\epsilon| \ll 1 \quad (7.6)$$

No matter how small ϵ is, the second term will eventually dominate the solution. The equivalent situation for a system of equations is to have eigenvalues of the characteristic equation of very different magnitudes.

It is easy to see that most turbulence transport equations hold potential for being stiff. The k - ϵ model is notoriously stiff when some of the commonly used viscous damping functions are introduced. Second-order closure models that use the ϵ equation are often so stiff as to almost preclude stable numerical solution. Some of the difficulty with the ϵ equation occurs because the dissipation time scale is a function of both k and ϵ . Transient solution errors in both parameters can yield large variations in k/ϵ , so that the dissipation time scale can assume an unrealistic range of values. By contrast, near-wall solutions to models based on the ω equation have well-defined algebraic solutions approaching a solid boundary, and are thus much easier to integrate.

7.2 Numerical Accuracy Near Boundaries

Proper treatment of boundary conditions is necessary for all numerical solutions, regardless of the equations being solved. Because of the special nature of turbulence transport equations, there are two types of boundary behavior that require careful treatment. Specifically, quantities such as dissipation rate, ϵ , and specific dissipation rate, ω , grow so rapidly approaching a solid boundary that they appear to be singular. In fact, ω is singular for a perfectly-smooth wall. Also, at interfaces between turbulent and nonturbulent regions, velocity and other properties have nearly discontinuous slopes approaching the interface. Because wall-bounded flows typically involve both types of boundaries, accurate numerical solutions must account for the special problems presented by this unusual solution behavior.

7.2.1 Solid Surfaces

We know that for a perfectly-smooth wall, the specific dissipation rate varies in the sublayer as y^{-2} approaching the surface (see Subsection 4.6.3). Even if we choose to use wall functions to obviate integration through the viscous sublayer, analysis of the log layer (see Subsection 4.6.1) shows that both ϵ

and ω are inversely proportional to distance from the surface. In either case, care must be taken to accurately compute derivatives of such functions.

To illustrate the difficulty imposed by singular behavior approaching a solid boundary, consider the function ϕ defined by

$$\phi = \frac{1}{y^n}; \quad n = 1 \text{ or } 2 \quad (7.7)$$

The exact first and second derivatives are

$$\frac{d\phi}{dy} = -\frac{n}{y^{n+1}} \quad \text{and} \quad \frac{d^2\phi}{dy^2} = \frac{n(n+1)}{y^{n+2}} \quad (7.8)$$

Using central differences on a uniform grid with $y_j = j\Delta y$, a straightforward calculation shows that

$$\left(\frac{d\phi}{dy}\right)_j \approx \frac{\phi_{j+1} - \phi_{j-1}}{2\Delta y} = \left[\frac{j^2}{j^2 - 1}\right]^n \left(\frac{d\phi}{dy}\right)_{exact} \quad (7.9)$$

and

$$\left(\frac{d^2\phi}{dy^2}\right)_j \approx \frac{\phi_{j+1} - 2\phi_j + \phi_{j-1}}{(\Delta y)^2} \approx \left[\frac{j^2}{j^2 - 1}\right]^n \left(\frac{d^2\phi}{dy^2}\right)_{exact} \quad (7.10)$$

where subscript j denotes the value at $y = y_j$. Table 7.1 lists the errors attending use of central differences as a function of $\Delta y/y_j$ for $n = 1$ and $n = 2$.

Table 7.1: Central Difference Errors for $\phi = y^{-n}$

j	$\Delta y/y_j$	(% Error) $_{n=1}$	(% Error) $_{n=2}$
2	.50	33	78
3	.33	13	27
5	.20	4	9
7	.14	2	4
10	.10	1	2

Clearly, significant numerical errors are introduced if the ratio $\Delta y/y_j$ is not small. If wall functions are used (corresponding to $n = 1$), regardless of how close the grid point nearest the surface lies, nontrivial numerical errors in derivatives result for $j < 5$. Consequently, simply using wall functions as effective boundary conditions applied at the first grid point above the

surface is unsatisfactory. Rather, the value for ω or ϵ should be specified for all points below $j = 4$ (at a minimum) to insure numerical accuracy. This is undoubtedly the primary reason why most researchers find their numerical solutions to be sensitive to near-wall grid-point spacing when they use wall functions. As an alternative, a relatively large cell can be used next to the surface, so that for example, $y_1 = 0$, $y_2 = \Delta y$, $y_3 = 1.2\Delta y$, etc. As a result of using the Rubel-Melnik (1984) transformation, Program DEFECT (Appendix C) automatically generates such a grid.

When the k - ω or multiscale model is integrated through the viscous sublayer for a perfectly-smooth surface (corresponding to $n = 2$), there is no practical way to avoid having $\Delta y/y_2 \sim 1$. The exact solution to the ω equation in the viscous sublayer is

$$\omega \sim \frac{N_\omega \nu_w}{y^2}, \quad y^+ < 2.5 \quad (7.11)$$

where

$$N_\omega = \begin{cases} 6/\beta, & \text{without viscous corrections} \\ 2/\beta^*, & \text{with viscous corrections} \end{cases} \quad (7.12)$$

If we simply use the value of ω according to Equation (7.11) at the first grid point above the surface, Table 7.1 shows that the molecular diffusion term will be in error by 78%. This, in turn, will increase values of ω at larger values of y . Recall that the surface value of ω has a strong effect on the additive constant, B , in the law of the wall (see Subsection 4.7.2). Thus, computing too large a value of ω near the surface will distort the velocity profile throughout the sublayer and into the log layer. That is, numerically inaccurate near-wall ω values can distort the entire boundary-layer solution.

The remedy that has proven very effective for eliminating this numerical error is to use Equation (7.11) for the first 7 to 10 grid points above the surface. Of course, these grid points must lie below $y^+ = 2.5$ since Equation (7.11) is not valid above this point. This procedure has been used in Programs PIPE and SUBLAY (Appendix C) and Program EDDYBL (Appendix D).

An alternative procedure for accurately computing near-surface behavior of ω is to use the rough-wall boundary condition. As shown in Subsection 4.7.2 for the k - ω model and Subsection 6.6.1 for the multiscale model,

$$\omega = \frac{u_\tau^2}{\nu_w} S_R \quad \text{at} \quad y = 0 \quad (7.13)$$

where

$$S_R = (50/k_R^+)^2, \quad k_R^+ < 25 \quad (7.14)$$

The quantity $k_R^+ = u_\tau k_R / \nu_w$ is the scaled surface roughness height.

In order to simulate a smooth surface, we simply require that k_R^+ be smaller than 5. Then, combining Equations (7.13) and (7.14), we arrive at the **slightly-rough-surface boundary condition** on ω , viz.,

$$\omega = \frac{2500\nu_w}{k_R^2} \quad \text{at} \quad y = 0 \quad (7.15)$$

It is important to select a small enough value of k_R to insure that $k_R^+ < 5$. If too large a value is selected, the skin friction values will be larger than smooth-wall values.

As a final comment, the near-wall solution to the ω equation for a rough wall is given by

$$\omega = \frac{\omega_w}{\left(1 + \sqrt{\frac{\omega_w}{N_\omega \nu_w}} y\right)^2}, \quad y^+ < 2.5 \quad (7.16)$$

where ω_w is the surface value of ω and N_ω is given in Equation (7.12). An important test for numerical accuracy of any finite-difference program implementing the ω equation is to verify that solutions match either Equation (7.11) or (7.16). If the program fails to accurately reproduce the near-wall ω variation, the program is unlikely to yield accurate results.

Rapid variation of the dependent variable is not the only potential source of numerical error near solid boundaries. Another serious consideration is roundoff error resulting from the relatively small difference between two numbers of comparable magnitude. This problem is frequently encountered with low-Reynolds-number k - ϵ models. For example, damping functions such as

$$f_2 = 1 - e^{-Re_T^2} \quad \text{and} \quad f_\mu = 1 - e^{-0.0115y^+} \quad (7.17)$$

appear in the Lam-Bremhorst (1981) and Chien (1982) models. Approaching the surface, desired asymptotic behavior depends upon accurate values of these damping functions. If single-precision accuracy is used, it is advisable to use Taylor series expansions for the damping functions close to the surface. For example, Chien's f_μ can be computed according to

$$f_\mu = \begin{cases} 1 - e^{-0.0115y^+}, & y^+ > 0.01 \\ 0.0115y^+, & y^+ \leq 0.01 \end{cases} \quad (7.18)$$

This procedure is used in Program EDDYBL (Appendix D) to insure numerically accurate solutions.

7.2.2 Turbulent/Nonturbulent Interfaces

More often than not, turbulence model equations that are in general usage appear to predict sharp interfaces between turbulent and nonturbulent regions, i.e., interfaces where discontinuities in derivatives of flow properties occur at the edge of the shear layer. As noted in earlier chapters, these interfaces bear no relation to the physical turbulent/nonturbulent interfaces that actually fluctuate in time and have smooth Reynolds-averaged properties. The mixing-length model, for example, exhibits a sharp interface for the far wake (see Subsection 3.3.1). That is, the predicted velocity profile is

$$U(x, y) = \begin{cases} U_\infty - 1.38\sqrt{\frac{D}{\rho x}} [1 - (y/\delta)^{3/2}]^2, & y < \delta \\ U_\infty, & y \geq \delta \end{cases} \quad (7.19)$$

where U_∞ is freestream velocity, D is drag per unit width, ρ is density, y is distance from the centerline and δ is the half-width of the wake. Clearly, all derivatives of U above $\partial^2 U/\partial y^2$ are discontinuous at $y = \delta$. Such a solution is called a **weak solution** to the differential equation.

By definition [see Courant and Hilbert (1966)], a weak solution to a partial differential equation

$$\mathcal{L}[u] = \frac{\partial}{\partial x}P(x, y, u) + \frac{\partial}{\partial y}Q(x, y, u) + S(x, y, u) = 0 \quad (7.20)$$

satisfies the following conditions.

1. $u(x, y)$ is piecewise continuous and has piecewise continuous first derivatives in two adjacent domains, R_1 and R_2 .
2. $\mathcal{L}[u] = 0$ in R_1 and R_2 .
3. For any test function $\phi(x, y)$ that is differentiable to all orders and that is identically zero outside of R_1 and R_2 , the following integral over the combined region $R = R_1 \cup R_2$ must be satisfied.

$$\iint_R \left[P \frac{\partial \phi}{\partial x} + Q \frac{\partial \phi}{\partial y} - S\phi \right] dx dy = 0 \quad (7.21)$$

A similar result holds for a system of equations. Clearly, Equation (7.21) can be rewritten as

$$\iint_R \left[\frac{\partial(\phi P)}{\partial x} + \frac{\partial(\phi Q)}{\partial y} \right] dx dy - \iint_R \phi \left[\frac{\partial P}{\partial x} + \frac{\partial Q}{\partial y} + S \right] dx dy = 0 \quad (7.22)$$

The second integral vanishes since P , Q and S satisfy the differential equation in both R_1 and R_2 . Then, using Gauss' theorem, if Γ is the curve of

discontinuity that divides R_1 and R_2 and $\mathbf{n} = (n_x, n_y)$ is the unit normal to Γ , there follows:

$$\int_{\Gamma} \phi ([P]n_x + [Q]n_y) ds = 0 \quad (7.23)$$

The symbols $[P]$ and $[Q]$ denote the jumps in P and Q across Γ . Since the function ϕ is arbitrary, we can thus conclude that the **jump condition** across the surface of discontinuity is given by

$$[P]n_x + [Q]n_y = 0 \quad (7.24)$$

For example, in the case of the far-wake solution given by the mixing length model, we have $P = U_{\infty}U$, $Q = -(\alpha\delta\partial U/\partial y)^2$ and $S = 0$. Inspection of Equation (7.19) shows that the jumps in P and Q are both zero, corresponding to the fact that the discontinuity appears in the second derivative rather than the first.

The occurrence of weak solutions causes problems on at least two counts. First, the jump condition is not unique. For example, if Q can be written as a function of P , we can always multiply Equation (7.20) by an arbitrary function $\psi(P)$, and rearrange as follows:

$$\frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} + S\psi = 0 \quad (7.25)$$

where

$$F = \int \psi(P) dP \quad \text{and} \quad G = \int \psi(P)Q'(P) dP \quad (7.26)$$

The jump condition then becomes

$$[F]n_x + [G]n_y = 0 \quad (7.27)$$

In other words, we can have any jump condition we want (and don't want!). This means we have no guarantee that our solution is unique.

The second difficulty posed by the presence of weak solutions is an adverse effect on accuracy and convergence of numerical solution methods. For example, a central-difference approximation for a first derivative is second-order accurate provided the function of interest is twice differentiable. However, if the function has discontinuous first or second derivative, the accuracy of the central-difference approximation becomes indeterminate. Maintaining second-order accuracy is then possible only if we know the location of the curve of discontinuity in advance. For a hyperbolic equation, this curve is a characteristic curve so that the method of characteristics, for example, can provide a high degree of accuracy in the vicinity

of such discontinuities. Since we don't know the location of the characteristics a priori in standard finite-difference computations, accuracy is suspect when the equations have weak solutions.

Turbulence energy equation models have problems similar to the mixing length model near turbulent/nonturbulent interfaces. Spalart and Allmaras (1992), for example, demonstrate existence of weak solutions to their one-equation model at such interfaces. Saffman (1970) was the first to illustrate weak solutions for a two-equation model. He discusses the nature of solutions to his $k-\omega^2$ model approaching a turbulent/nonturbulent interface. In fact, he builds in weak-solution behavior by choosing his closure coefficients to insure that approaching the interface from within the turbulent region, the streamwise velocity and turbulence length scale vary as

$$U_e - U \propto (\delta - y) \quad \text{and} \quad \ell = k^{1/2}/\omega \propto \text{constant} \quad \text{as} \quad y \rightarrow \delta \quad (7.28)$$

where the interface lies at $y = \delta$. Vollmers and Rotta (1977) discuss solution behavior near a turbulent/nonturbulent interface for their $k-k\ell$ model, while Rubel and Melnik (1984) perform a similar analysis for the $k-\epsilon$ model. Cazalbou, Spalart and Bradshaw (1994) confirm existence of weak solutions for most $k-\epsilon$, $k-k\ell$ and $k-\omega$ models. Finally, inspection of the $k-\omega$ and $k-\epsilon$ model free shear flow velocity profiles [Figures 4.5 - 4.8] illustrates the nonanalytic behavior at the edge of the shear layer.

Rubel and Melnik (1984) offer an interesting solution for thin shear layer flows that effectively maps the turbulent/nonturbulent interface to infinity and implicitly clusters grid points near the interface. Their transformation consists of introducing a new independent variable, ξ , defined in terms of the normal distance, y , by

$$d\xi = \frac{dy}{\nu_T} \quad \text{or} \quad \frac{d}{d\xi} = \nu_T \frac{d}{dy} \quad (7.29)$$

where ν_T is kinematic eddy viscosity. The Rubel-Melnik transformation, which is useful primarily for self-similar flows, improves numerical accuracy because the edge of the shear layer that occurs at a finite value of y moves to infinity in terms of the transformed independent variable ξ (provided $\nu_T = 0$ in the freestream). Since $\nu_T \rightarrow 0$, the transformation produces fine resolution near the interface. For example, if the freestream velocity, U_e , is constant, close to the shear-layer edge, convection balances turbulent diffusion in the streamwise momentum equation. Hence,

$$V \frac{dU}{dy} = \frac{d}{dy} \left(\nu_T \frac{dU}{dy} \right) \quad (7.30)$$

where V is the entrainment velocity, which must also be constant in order to satisfy continuity. Since shear layers grow in thickness, necessarily $V < 0$.

Multiplying both sides of Equation (7.30) by ν_T and using Equation (7.29), we arrive at

$$V \frac{dU}{d\xi} = \frac{d^2U}{d\xi^2} \quad (7.31)$$

for which the solution is

$$U = U_e - \mathcal{U}e^{V\xi} \quad (7.32)$$

where \mathcal{U} is a constant of integration.

Using the Rubel-Melnik transformation, it is a straightforward matter to determine the nature of solutions to turbulence model equations approaching a turbulent/nonturbulent interface. Applying the transformation to the k - ϵ model, for example, we find

$$V \frac{dk}{d\xi} = \left(\frac{dU}{d\xi} \right)^2 - C_\mu k^2 + \frac{1}{\sigma_k} \frac{d^2k}{d\xi^2} \quad (7.33)$$

$$V \frac{d\epsilon}{d\xi} = C_{\epsilon 1} \frac{\epsilon}{k} \left(\frac{dU}{d\xi} \right)^2 - C_{\epsilon 2} C_\mu k \epsilon + \frac{1}{\sigma_\epsilon} \frac{d^2\epsilon}{d\xi^2} \quad (7.34)$$

Provided the closure coefficients σ_k and σ_ϵ are both less than 2, the production and dissipation terms are negligible in both equations. The solution approaching the interface is

$$k \sim \mathcal{K}e^{\sigma_k V \xi}, \quad \epsilon \sim \mathcal{E}e^{\sigma_\epsilon V \xi} \quad (\sigma_k < 2, \quad \sigma_\epsilon < 2) \quad (7.35)$$

where \mathcal{K} and \mathcal{E} are integration constants. Thus, the eddy viscosity is

$$\nu_T \sim C_\mu \frac{\mathcal{K}^2}{\mathcal{E}} e^{(2\sigma_k - \sigma_\epsilon)V\xi} \quad (7.36)$$

Finally, substituting Equation (7.36) into Equation (7.29) and integrating yields

$$e^{V\xi} \propto (1 - y/\delta)^{(2\sigma_k - \sigma_\epsilon)^{-1}} \quad (7.37)$$

wherefore the solution to the k - ϵ model equations approaching a turbulent/nonturbulent interface from the turbulent side behaves according to

$$\left. \begin{aligned} U_e - U &\sim \mathcal{U}(1 - y/\delta)^{(2\sigma_k - \sigma_\epsilon)^{-1}} \\ k &\sim \mathcal{K}(1 - y/\delta)^{\sigma_k(2\sigma_k - \sigma_\epsilon)^{-1}} \\ \epsilon &\sim \mathcal{E}(1 - y/\delta)^{\sigma_\epsilon(2\sigma_k - \sigma_\epsilon)^{-1}} \end{aligned} \right\} \text{ as } y \rightarrow \delta \quad (7.38)$$

Using the standard values $\sigma_k = 1.0$ and $\sigma_\epsilon = 1.3$, the k - ϵ model predicts

$$\left. \begin{aligned} U_e - U &\sim \mathcal{U}(1 - y/\delta)^{10/7} \\ k &\sim \mathcal{K}(1 - y/\delta)^{10/7} \\ \epsilon &\sim \mathcal{E}(1 - y/\delta)^{13/7} \end{aligned} \right\} \text{ as } y \rightarrow \delta \quad (7.39)$$

The solution for the k - ω model is a bit more complicated when σ and σ^* are both equal to $1/2$. As a result, only the dissipation terms are negligible, and the production term in the transformed k equation yields a secular term, which complicates the solution. That is, the approximate transformed equations for k and ω are as follows.

$$\frac{d^2 k}{d\xi^2} - 2V \frac{dk}{d\xi} = 2V^2 \mathcal{U}^2 e^{2V\xi} \quad (7.40)$$

$$\frac{d^2 \omega}{d\xi^2} - 2V \frac{d\omega}{d\xi} = 2\alpha V^2 \mathcal{U}^2 \frac{\omega}{k} e^{2V\xi} \quad (7.41)$$

The solution for k and ω is

$$k \sim \mathcal{U}^2 V \xi e^{2V\xi}, \quad \omega \sim \mathcal{W} \xi^{-\alpha} \quad (7.42)$$

where \mathcal{W} is an integration constant. Computing the eddy viscosity and substituting into Equation (7.29), we arrive at

$$y \sim \delta + \frac{\mathcal{U}^2 V}{\mathcal{W}} \int_{\xi}^{\infty} \xi^{1+\alpha} e^{2V\xi} d\xi \quad (7.43)$$

Integrating by parts, we can approximate the limiting form of the integral for $\xi \rightarrow \infty$ as follows.

$$\delta - y \sim \frac{\mathcal{U}^2}{2\mathcal{W}} \xi^{1+\alpha} e^{2V\xi} \quad (7.44)$$

Now, we must solve this equation for ξ as a function of $\delta - y$. To do this, let

$$\eta = \frac{2\mathcal{W}}{\mathcal{U}^2} (\delta - y) \quad (7.45)$$

Then, Equation (7.44) simplifies to

$$\eta \sim \xi^{1+\alpha} e^{2V\xi} \quad (7.46)$$

This equation can be solved for ξ as a function of η by assuming

$$2V\xi \sim \ell n \eta + \phi(\eta) \quad (7.47)$$

where $\phi(\eta)$ is a function to be determined. In the limit $\xi \rightarrow \infty$, which corresponds to $\eta \rightarrow 0$, the approximate solution for $\phi(\eta)$ is

$$\phi(\eta) \sim -(1 + \alpha) \ell n \left(\frac{\ell n \eta}{2V} \right) \quad (7.48)$$

With a bit more algebra, there follows

$$e^{V\xi} \propto \eta^{1/2} \left(\frac{2V}{\ell n \eta} \right)^{(1+\alpha)/2} \quad (7.49)$$

Thus, the behavior of solutions to the k - ω model approaching a turbulent/nonturbulent interface from within the turbulent region is

$$\left. \begin{aligned} U_e - U &\sim \mathcal{U}\sqrt{\lambda} \\ k &\sim -\mathcal{K}\lambda\ell n\lambda \\ \omega &\sim \mathcal{W}(-\ell n\lambda)^{-\alpha} \\ \lambda &\sim \frac{(1-y/\delta)}{[-\ell n(1-y/\delta)]^{1+\alpha}} \end{aligned} \right\} \text{ as } y \rightarrow \delta \quad (7.50)$$

Clearly, ω approaches zero very slowly from the turbulent side as compared to the variation of $\epsilon/k \sim (\delta - y)^{3/7}$ predicted by the k - ϵ model.

Usually it is more convenient to assign small nonzero values to k and other turbulence parameters in the freestream, especially when the parameter appears in the denominator of the eddy viscosity. Cazalbou, Spalart and Bradshaw (1994) show that when this is done in boundary-layer computations with the k - ϵ model, the weak solution prevails below the interface. Small gradients in k and ϵ appear above the interface that yield an asymptotic approach to the prescribed freestream values. There is “no significant influence on the predicted flow.”

By contrast, Menter (1992a) shows that for the far wake, in which the entrainment velocity increases in magnitude linearly with distance from the centerline, the k - ω model predicts that k and ω decay exponentially with distance squared. However, they decay at the same rate so that the eddy viscosity remains constant. As a consequence, consistent with results presented in Section 4.5, the freestream value of ω has a nontrivial effect on the solution. Menter indicates a smaller effect on boundary layers, primarily because of the large values of ω prevailing near the surface. The behavior of ω in Equation (7.50) is consistent with Menter’s observation that the k - ω model solutions appear to have discontinuous derivatives at the shear layer edge. However, the discontinuity in $d\omega/dy$ would probably be difficult to detect.

In principle, solutions with discontinuous derivatives will not occur if molecular viscosity is included in the diffusion terms of the equations of motion. As shown by Saffman (1970), there is a thin **viscous-interface layer** of thickness

$$\delta \sim \nu/|V| \quad (7.51)$$

in which the discontinuities are resolved. This is a singular perturbation problem in the limit $|V|\delta/\nu \rightarrow \infty$, and the weak solution discussed above is the outer solution. The inner solution holds in the viscous-interface layer. For example, in the interface layer, Saffman's equations simplify to

$$\left. \begin{aligned} V \frac{dU}{dy} &= \frac{d}{dy} \left[\left(\nu + \frac{k}{\omega} \right) \frac{dU}{dy} \right] \\ V \frac{dk}{dy} &= \frac{d}{dy} \left[\left(\nu + \sigma^* \frac{k}{\omega} \right) \frac{dk}{dy} \right] \\ V \frac{d\omega^2}{dy} &= \frac{d}{dy} \left[\left(\nu + \sigma \frac{k}{\omega} \right) \frac{d\omega^2}{dy} \right] \end{aligned} \right\} \quad (7.52)$$

These equations must be solved subject the following conditions that correspond to formal matching to the solutions on each side of the turbulent/nonturbulent interface:

$$U_e - U \rightarrow \mathcal{U}(\delta - y), \quad k \rightarrow \mathcal{K}(\delta - y)^2, \quad \omega \rightarrow \frac{\mathcal{K}}{|V|}(\delta - y) \quad \text{as} \quad \frac{|V|(\delta - y)}{\nu} \rightarrow \infty \quad (7.53)$$

and

$$U_e - U \rightarrow 0, \quad k \rightarrow 0, \quad \omega \rightarrow 0 \quad \text{as} \quad \frac{|V|(\delta - y)}{\nu} \rightarrow -\infty \quad (7.54)$$

As can be easily verified, for $\sigma = \sigma^* = 1/2$, the solution is given by

$$\left. \begin{aligned} U_e - U &= \frac{\mathcal{U}|V|^3}{\mathcal{K}^2\nu} \left(\frac{\omega^2}{1 + V^2\omega/\mathcal{K}\nu} \right) \\ k &= \frac{V^2\omega^2}{\mathcal{K}} \\ y - \delta &= \frac{|V|\omega}{\mathcal{K}} + \frac{2\nu}{|V|} \ell n \left(\frac{V^2\omega}{\mathcal{K}\nu} \right) \end{aligned} \right\} \quad (7.55)$$

In practice, finite difference grids are never sufficiently fine to resolve the viscous-interface layer. Generally, grid points are packed close to the surface to permit accurate resolution of the sublayer. Hence, even when molecular viscosity is included in a typical finite-difference computation, turbulent/nonturbulent interfaces are not sufficiently resolved. As a consequence, the interfaces are sharp, and the weak solutions generally prevail. However, truncation error, numerical diffusion and dissipation will generally yield diffused solutions close to the interfaces. The most significant numerical problem typically encountered is the appearance of nonphysical

negative values of k and/or other normally positive turbulence parameters such as ω , ϵ and ℓ .

For self-similar flows such as the far wake, mixing layer, jet and defect layer, the Rubel-Melnik transformation cures the problem by mapping the interface to ∞ . Programs WAKE, MIXER, JET and DEFECT described in Appendix C all use this transformation. In addition to eliminating difficulties associated with the turbulent/nonturbulent interface, the transformation linearizes the first and second derivative terms in the equations. This linearization tends to improve the rate of convergence of most numerical methods. The only shortcoming of the method is its sensitivity to the location of ∞ . Using too large or too small a value of ξ_{max} (the far-field value of ξ) sometimes impedes convergence of the numerical solution.

In general finite-difference computations, the correct jump condition will be obtained provided the diffusion terms in all equations are differenced in a conservative manner. For the same reasons, we use conservative differencing for the Navier-Stokes equation to guarantee that the exact shock relations are satisfied across a shock wave in a finite-difference computation. Program EDDYBL (Appendix D), for example, uses conservative differencing for diffusion terms and rarely ever encounters numerical difficulties attending the presence of sharp turbulent/nonturbulent interfaces.

For nonzero freestream values of k , etc., some researchers prefer zero-gradient boundary conditions at a boundary-layer edge. While such conditions are clean from a theoretical point of view, they are undesirable from a numerical point of view. That is, specified values at the edge are of the Neumann type while zero-gradient conditions are of the Dirichlet type. Almost universally, convergence of iterative schemes is much slower with Dirichlet conditions than with Neumann conditions.

In order to resolve this apparent dilemma, we can appeal directly to the equations of motion. Beyond the boundary-layer edge, we expect to have vanishing normal gradients so that the equations for k and ω simplify to the following:

$$U_e \frac{dk_e}{dx} = -\beta^* \omega_e k_e \quad (7.56)$$

$$U_e \frac{d\omega_e}{dx} = -\beta \omega_e^2 \quad (7.57)$$

where subscript e denotes the value at the boundary-layer edge. The solution to Equations (7.56) and (7.57) can be obtained by simple quadrature, independent of integrating the equations of motion through the boundary layer. Once k_e and ω_e are determined from Equations (7.56) and (7.57), it is then possible to specify Neumann-type boundary conditions that guarantee zero normal gradients. Clearly, the same procedure can be used for any turbulence model. Program EDDYBL (Appendix D) uses this procedure.

7.3 Parabolic Marching Methods

In general, numerical methods for solving parabolic systems of equations such as the boundary-layer equations are unconditionally stable. A second-order accurate scheme like the Blottner (1974) variable-grid method, for example, involves inversion of a tridiagonal matrix. If the matrix is diagonally dominant, the scheme will run stably with arbitrarily large streamwise stepsize, Δx . Turbulent boundary layer computations using algebraic models often run with $\Delta x/\delta$ between 1 and 10, where δ is boundary layer thickness. By contrast, early experience with two-equation models indicated that much smaller steps must be taken. Rastogi and Rodi (1978) found that their three-dimensional boundary-layer program based on the Jones-Launder (1972) k - ϵ model required initial steps of about $\delta/100$, and that ultimately Δx could not exceed $\delta/2$. Similar results hold for models based on the ω equation.

Wilcox (1981b) found that the problem stems from a loss of diagonal dominance caused by the production terms in the turbulence model equations. To illustrate the essence of the problem, consider the k - ω model's turbulence energy equation for an incompressible two-dimensional boundary layer, viz.,

$$U \frac{\partial k}{\partial x} + V \frac{\partial k}{\partial y} = \left[\frac{(\partial U / \partial y)^2}{\omega} - \beta^* \omega \right] k + \frac{\partial}{\partial y} \left[(\nu + \sigma^* \nu_T) \frac{\partial k}{\partial y} \right] \quad (7.58)$$

The following analysis is based on the Blottner variable-grid method, which is the scheme implemented in Program EDDYBL (Appendix D). This algorithm uses a three-point forward difference formula [Adams-Bashforth — see Roache (1976)] in the streamwise direction, central differencing for the normal convection term, and conservative differencing for the diffusion terms. Hence, discretization approximations for all except the source terms are as follows:

$$U \frac{\partial k}{\partial x} \doteq \frac{U}{\Delta x} (3k_{m+1,n} - 4k_{m,n} + k_{m-1,n}) \quad (7.59)$$

$$V \frac{\partial k}{\partial y} \doteq \frac{V}{2\Delta y} (k_{m+1,n+1} - k_{m+1,n-1}) \quad (7.60)$$

$$\frac{\partial}{\partial y} \left[(\nu + \sigma^* \nu_T) \frac{\partial k}{\partial y} \right] \doteq \frac{\nu^+ (k_{m+1,n+1} - k_{m+1,n}) - \nu^- (k_{m+1,n} - k_{m+1,n-1})}{(\Delta y)^2} \quad (7.61)$$

where $k_{m,n}$ denotes the value of k at $x = x_m$ and $y = y_n$, and Δy denotes the vertical distance between grid points. Unsubscripted quantities are assumed known during the typically iterative solution procedure. Also, the

quantity ν^- denotes the value of $(\nu + \sigma^* \nu_T)$ midway between y_{n-1} and y_n , while ν^+ denotes the value midway between y_n and y_{n+1} . For simplicity, we assume points are equally spaced in both the x and y directions, so that the grid consists of rectangular cells. Figure 7.1 shows the finite-difference molecule.

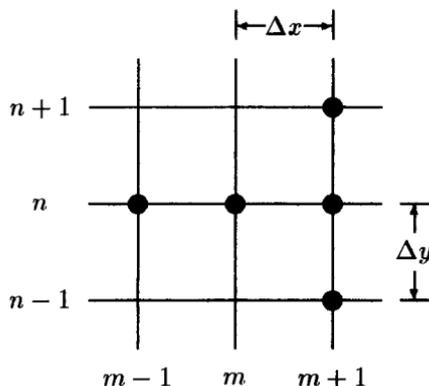


Figure 7.1: Finite-difference molecule for Blottner's variable-grid method.

Turning to the source terms, the simplest second-order accurate discretization approximation is

$$\left[\frac{(\partial U / \partial y)^2}{\omega} - \beta^* \omega \right] k \doteq \left[\frac{(\partial U / \partial y)^2}{\omega} - \beta^* \omega \right] k_{m+1,n} \quad (7.62)$$

Substituting Equations (7.59)-(7.62) into Equation (7.58) and regrouping terms leads to a tridiagonal matrix system as follows:

$$A_n k_{m+1,n-1} + B_n k_{m+1,n} + C_n k_{m+1,n+1} = D_n \quad (7.63)$$

where A_n , B_n , C_n and D_n are defined by

$$A_n = - \left[\frac{V}{2\Delta y} + \frac{\nu^-}{(\Delta y)^2} \right] \quad (7.64)$$

$$B_n = 3 \frac{U}{\Delta x} + \frac{\nu^- + \nu^+}{(\Delta y)^2} - \frac{(\partial U / \partial y)^2}{\omega} + \beta^* \omega \quad (7.65)$$

$$C_n = \left[\frac{V}{2\Delta y} - \frac{\nu^+}{(\Delta y)^2} \right] \quad (7.66)$$

$$D_n = \frac{U}{\Delta x} [4k_{m,n} - k_{m-1,n}] \quad (7.67)$$

Now, in order to have a diagonally dominant system, the condition

$$B_n \geq -(A_n + C_n) \quad (7.68)$$

must be satisfied. Substituting Equations (7.64)-(7.66) into Equation (7.68) yields the following condition.

$$3 \frac{U}{\Delta x} - \frac{(\partial U / \partial y)^2}{\omega} + \beta^* \omega \geq 0 \quad (7.69)$$

When dissipation exceeds production, so that $\beta^* \omega > (\partial U / \partial y)^2 / \omega$, Equation (7.69) is satisfied so long as we march in the direction of flow (i.e., so long as U and Δx are of the same sign). The system is then said to be **unconditionally stable**. However, when production exceeds dissipation, we have the following limit on stepsize.

$$\Delta x \leq (\Delta x)_{theory} \equiv \frac{3\omega U}{(\partial U / \partial y)^2 - \beta^* \omega^2} \quad (7.70)$$

Hence, the scheme is **conditionally stable**, the condition being that of Equation (7.70).

To demonstrate the validity of Equation (7.70), Wilcox (1981b) presents computed results for an incompressible flat-plate boundary layer using the Wilcox-Rubesin (1980) $k-\omega^2$ model. At a plate-length Reynolds number, Re_x , of $1.2 \cdot 10^6$, stable computation is found empirically to be possible provided $Re_{\Delta x} < 2.2 \cdot 10^4$, which corresponds to $\Delta x / \delta = 1.15$. Figure 7.2 shows $Re_{\Delta x}$ as predicted by Equation (7.70) throughout the boundary layer. As shown, the minimum value of $Re_{\Delta x}$ according to Equation (7.70) is $1.9 \cdot 10^4$ and occurs at $y / \delta \doteq 0.012$. This close agreement verifies that the source of instability is lack of diagonal dominance in the tridiagonal matrix system defined in Equations (7.63)-(7.67).

To remedy this situation, note first that because of nonlinearity, Equation (7.63) always requires an iterative solution. Letting superscript i denote iteration number, we replace B_n and D_n by the following revised discretization approximations:

$$B_n = 3 \frac{U}{\Delta x} + \frac{\nu^- + \nu^+}{(\Delta y)^2} - \frac{(\partial U / \partial y)^2}{\omega} + (1 + \psi_k) \beta^* \omega \quad (7.71)$$

$$D_n = \frac{U}{\Delta x} [4k_{m,n} - k_{m-1,n}] + \psi_k \beta^* \omega k_{m+1,n}^{i-1} \quad (7.72)$$

where ψ_k will be defined below. Then, Equation (7.63) is replaced by

$$A_n k_{m+1,n-1}^i + B_n k_{m+1,n}^i + C_n k_{m+1,n+1}^i = D_n \quad (7.73)$$

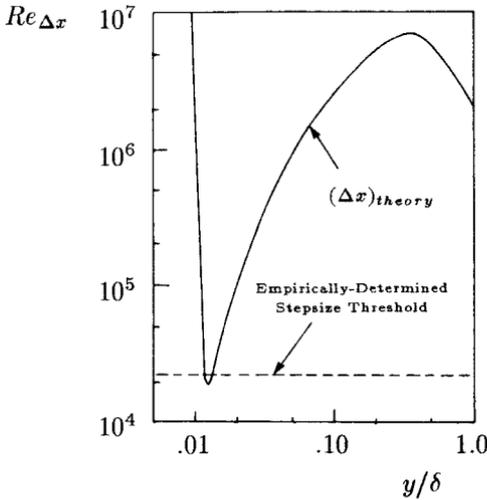


Figure 7.2: Theoretical and empirically determined stepsize threshold for a flat-plate boundary layer. [From Wilcox (1981b) — Copyright © AIAA 1981 — Used with permission.]

Inspection of Equations (7.71)-(7.73) shows that when convergence has been achieved (i.e., when $k_{m+1,n}^i$ and $k_{m+1,n}^{i-1}$ differ by a negligible amount), terms on the right- and left-hand sides of Equation (7.73) proportional to ψ_k cancel identically. Hence, $k_{m+1,n}^i$ satisfies the correct equation. The advantage of this procedure becomes obvious upon inspection of the stability condition, which now becomes

$$3 \frac{U}{\Delta x} - \frac{(\partial U / \partial y)^2}{\omega} + (1 + \psi_k) \beta^* \omega \geq 0 \quad (7.74)$$

Clearly, ψ_k can be chosen to insure that this inequality is always satisfied, regardless of the value of Δx . This corresponds to unconditional stability.

Numerical experimentation shows that the best results are obtained when $(1 + \psi_k) \beta^* \omega$ exceeds $(\partial U / \partial y)^2 / \omega$ by about 30%, a condition that is insured by defining ψ_k as follows.

$$\psi_k = \begin{cases} \frac{3}{10}, & (\partial U / \partial y)^2 \leq \beta^* \omega^2 \\ \frac{(\partial U / \partial y)^2}{\beta^* \omega^2} - \frac{7}{10}, & (\partial U / \partial y)^2 > \beta^* \omega^2 \end{cases} \quad (7.75)$$

A similar factor, ψ_ω , must be introduced for the ω equation, and experience

has shown that selecting

$$\psi_\omega = \psi_k \quad (7.76)$$

is satisfactory to achieve both unconditional stability and rapid convergence.

The prescription for ψ_k and ψ_ω given in Equations (7.75) and (7.76) permits stepsizes comparable to those used with algebraic models. While the numerical procedure is unconditionally stable for other values of ψ_k , using these values for ψ_k and ψ_ω optimizes $k-\omega^2$ and $k-\omega$ model computations with respect to the number of iterations required for the solution to converge. Interestingly, if ψ_k is too large, say $\psi_k = 2$, stable integration is inhibited. The value of ψ_ω cannot be too large either, although the upper bound appears to be dependent upon details of the specific model.

The same analysis applies to the $k-\epsilon$ model. For the k equation, writing Equation (7.75) in terms of the model's variables leads to the following entirely equivalent form.

$$\psi_k = \begin{cases} \frac{3}{10}, & \nu_T(\partial U/\partial y)^2 \leq \epsilon \\ \frac{\nu_T(\partial U/\partial y)^2}{\epsilon} - \frac{7}{10}, & \nu_T(\partial U/\partial y)^2 > \epsilon \end{cases} \quad (7.77)$$

By contrast, the value of the corresponding factor for the ϵ equation, ψ_ϵ , is very much dependent upon details of the model. Low-Reynolds-number viscous damping functions have a pronounced effect on the most appropriate value. Table 7.2 lists the values of ψ_ϵ used in Program EDDYBL (Appendix D) for six different low-Reynolds-number $k-\epsilon$ models. The values listed have been found empirically to yield optimum convergence rates for incompressible boundary layers.

Table 7.2: Values of ψ_ϵ for low-Reynolds-number $k-\epsilon$ models

Model	ψ_ϵ
Jones-Launder (1972)	0.50
Launder-Sharma (1974)	0.50
Lam-Bremhorst (1981)	0.50
Chien (1982)	-0.25
Yang-Shih (1993)	-0.25
Fan-Lakshminarayana-Barnett (1993)	-0.25

7.4 Elementary Time-Marching Methods

One of the most effective procedures for solving complex flowfields is the use of time-marching methods. If the desired solution is unsteady, time-marching solutions yield a true time history. Time-marching methods can also be used for steady-flow problems by letting the solution evolve in time until temporal variations become negligibly small. That is, we begin with an initial approximation and update the solution at each timestep until the solution differs between timesteps by less than a prescribed tolerance level. Prior to discussing the impact of turbulence-model source terms on explicit and implicit methods, this section presents a brief overview of these methods. For more complete details see a general text on Computational Fluid Dynamics such as Roache (1976), Peyret and Taylor (1983), Anderson et al. (1984) or Minkowycz et al. (1988).

The simplest time-marching schemes are **explicit methods**, such as the DuFort-Frankel (1953), Godunov (1959), Lax-Wendroff (1960) and MacCormack (1969) methods. Most explicit schemes were developed prior to 1970. In an explicit scheme, the solution at time t^{n+1} depends only on the past history, i.e., the solution at time t^n . For example, consider the one-dimensional wave equation:

$$\frac{\partial k}{\partial t} + U \frac{\partial k}{\partial x} = 0, \quad U > 0 \quad (7.78)$$

where k is a flow property, U is velocity, t is time and x is streamwise direction. Letting k_j^n denote $k(x_j, t^n)$, we approximate $\partial k/\partial t$ with a **forward-difference approximation** so that

$$\frac{\partial k}{\partial t} \doteq \frac{k_j^{n+1} - k_j^n}{\Delta t} + O(\Delta t) \quad (7.79)$$

where $\Delta t = t^{n+1} - t^n$. For simplicity, consider simple **upwind differencing** in which we approximate $\partial k/\partial x$ according to

$$\frac{\partial k}{\partial x} \doteq \frac{k_j^n - k_{j-1}^n}{\Delta x} + O(\Delta x) \quad (7.80)$$

Using these discretization approximations, we arrive at the following first-order accurate difference equation that approximates Equation (7.78).

$$k_j^{n+1} = k_j^n - \frac{U \Delta t}{\Delta x} (k_j^n - k_{j-1}^n) \quad (7.81)$$

This is not a particularly accurate method, but nevertheless illustrates the general nature of explicit schemes. Note that all terms on the right-hand

side of Equation (7.81) are known from time t^n . Hence, k_j^{n+1} is obtained from simple algebraic operations. Because only algebraic operations are needed (as opposed to inversion of a large matrix), explicit methods are easy to implement.

The primary shortcoming of explicit schemes is a limit on the timestep that can be used. For too large a timestep, solution errors will grow with increasing iterations and the computation becomes unstable. The most commonly used method for determining the stability properties of a time-marching finite-difference scheme is von Neumann stability analysis [see Roache (1976) or Anderson et al. (1984)]. In this method, we introduce a discrete Fourier series solution to the finite-difference equation under study, and determine the growth rate of each mode. If all Fourier modes decay as we march in time, the scheme is stable. However, if even a single mode grows, the scheme is unstable. We write each Fourier component as

$$k_j^n = G^n e^{i(j\kappa\Delta x)} \quad (7.82)$$

where G is called the **amplitude factor**, $i = \sqrt{-1}$ and κ is wavenumber. The stability of a scheme is determined as follows:

$$\left. \begin{array}{l} |G| < 1, \quad \text{Stable} \\ |G| = 1, \quad \text{Neutrally Stable} \\ |G| > 1, \quad \text{Unstable} \end{array} \right\} \quad (7.83)$$

In general, G is complex, and the notation G^n means G raised to the power n . The amplitude factor for Equation (7.81) is

$$G = 1 - \frac{U\Delta t}{\Delta x} (1 - e^{-i\theta}), \quad \text{where } \theta = \kappa\Delta x \quad (7.84)$$

Thus,

$$|G|^2 = 1 + 2(1 - \cos\theta) \frac{U\Delta t}{\Delta x} \left(\frac{U\Delta t}{\Delta x} - 1 \right) \quad (7.85)$$

In order to have a stable scheme, $|G|$ must be less than or equal to 1 for all possible values of θ . Clearly, for the upwind-difference scheme, errors will not grow provided the condition

$$\Delta t < \frac{\Delta x}{U} \quad \text{or} \quad N_{CFL} = \frac{U\Delta t}{\Delta x} < 1 \quad (7.86)$$

is satisfied. This is the famous Courant-Friedrichs-Lewy (1967), or CFL condition, which means a wave cannot propagate a distance exceeding Δx in a single timestep. N_{CFL} is known as the **CFL Number**.

Explicit methods are of interest in modern CFD applications mainly for time-dependent flows. There has been renewed interest in these methods because of their suitability for massively-parallel computers. In summary, their algebraic simplicity makes them especially easy to implement on any computer. Their primary drawback is their conditional stability, and thousands of timesteps are often needed to achieve steady-flow conditions.

Implicit methods date back to 1947 when the Crank-Nicolson (1947) method first appeared. Other methods such as the Euler [Lilley (1965)] and Alternating Direction Implicit (ADI) schemes [Peaceman and Rachford (1955)] are implicit. The solution at time t^{n+1} and location x_j in this type of scheme depends not only upon the solution at the earlier timestep, but upon the solution at other spatial locations at time t^{n+1} as well. For example, the Crank-Nicolson method uses

$$\frac{\partial k}{\partial x} \doteq \frac{1}{2} \left(\frac{k_{j+1}^n - k_{j-1}^n}{2\Delta x} + \frac{k_{j+1}^{n+1} - k_{j-1}^{n+1}}{2\Delta x} \right) + O[(\Delta x)^2] \quad (7.87)$$

Thus, Equation (7.78) is approximated by the following second-order accurate difference equation:

$$-\lambda k_{j-1}^{n+1} + k_j^{n+1} + \lambda k_{j+1}^{n+1} = k_j^n - \lambda (k_{j+1}^n - k_{j-1}^n) \quad (7.88)$$

where

$$\lambda = \frac{U\Delta t}{4\Delta x} \quad (7.89)$$

Hence, as with the Blottner method discussed in the preceding section, a tridiagonal matrix system of equations must be solved. Although inverting any matrix is more time consuming than solving a simple algebraic equation, the increased complexity is attended by a significant increase in the maximum permissible timestep. That is, stability analysis shows that the scheme defined in Equation (7.88) is unconditionally stable.

Implicit schemes have proven to be especially useful for steady-flow computations where the CFL limit can be exceeded by factors as large as 5. While these schemes will run at a larger CFL number, using larger values of Δt sometimes introduces significant truncation errors if convective effects have a significant effect on the physics of the flow. The number of timesteps required, relative to explicit methods, to achieve steady-flow conditions typically is reduced, although the factor is N_{CFL}^{-n} where $n < 1$.

Recall from Section 7.1 that there are three physically relevant time scales when turbulence model equations are used. If we use an explicit finite-difference scheme to approximate the Favre-averaged Navier-Stokes equation, stability analysis shows that the wave speed is $|\tilde{u}| + a$, where \tilde{u}

is mass averaged velocity and a is sound speed. If ν denotes kinematic viscosity, the wave-propagation and diffusion timestep limitations are as follows.

$$\Delta t \leq \frac{\Delta x}{|\bar{u}| + a} \quad \text{and} \quad \Delta t \leq \frac{(\Delta x)^2}{2\nu} \quad (7.90)$$

We might also anticipate that including source terms in the stability analysis would lead to an additional timestep constraint such as $\Delta t \leq t_{diss}$. This is indeed the case, and this timestep limitation is sometimes more restrictive than either condition in Equation (7.90).

To illustrate the problem, we add a source term, Sk , to Equation (7.78), wherefore

$$\frac{\partial k}{\partial t} + U \frac{\partial k}{\partial x} = Sk \quad (7.91)$$

The condition $S > 0$ corresponds to production exceeding dissipation, and vice versa for $S < 0$. To cast this equation in discretized form, we use Crank-Nicolson differencing and we approximate the source term as follows:

$$Sk \doteq S [\psi k_j^n + (1 - \psi)k_j^{n+1}] + O[(\psi - \frac{1}{2})\Delta t, (\Delta t)^2] \quad (7.92)$$

where ψ lies between 0 and 1. Hence, our finite-difference approximation to Equation (7.91) is

$$k_j^{n+1} = k_j^n - \lambda (k_{j+1}^{n+1} + k_{j+1}^n - k_{j-1}^{n+1} - k_{j-1}^n) + S\Delta t [\psi k_j^n + (1 - \psi)k_j^{n+1}] \quad (7.93)$$

The complex amplification factor for this scheme is

$$G = \frac{1 + \psi S\Delta t - 2i\lambda \sin \theta}{1 - (1 - \psi)S\Delta t + 2i\lambda \sin \theta} \quad (7.94)$$

Hence, in order for this scheme to be stable, we must require

$$|G|^2 = \frac{[1 + \psi S\Delta t]^2 + 4\lambda^2 \sin^2 \theta}{[1 - (1 - \psi)S\Delta t]^2 + 4\lambda^2 \sin^2 \theta} \leq 1 \quad (7.95)$$

After a little algebra, the stability condition simplifies to

$$S\Delta t [1 + (\psi - \frac{1}{2})S\Delta t] \leq 0 \quad (7.96)$$

When $S < 0$, we find

$$\left\{ \begin{array}{l} \Delta t \leq \frac{1}{(\psi - \frac{1}{2})|S|}; \quad \psi > \frac{1}{2}, \quad S < 0 \\ \text{Unconditionally Stable}; \quad \psi \leq \frac{1}{2}, \quad S < 0 \end{array} \right. \quad (7.97)$$

When $S > 0$, upon first inspection, von Neumann stability analysis indicates this scheme is unstable when $\psi \geq \frac{1}{2}$ and that Δt must have a lower bound (as opposed to an upper bound) to insure stable computation when $\psi < \frac{1}{2}$. However, these results are irrelevant. This is true because the exact solution to Equation (7.91) is proportional to e^{St} , and is thus unbounded as $t \rightarrow \infty$. When this occurs, even if the error is a small fraction of the exact solution, it will also be unbounded. The requirement $|G| \leq 1$ is thus too stringent for an unbounded function. According to von Neumann, the condition for stability when the exact solution is unbounded is:

$$|G| \leq 1 + O(\Delta t) \quad (7.98)$$

With a little rearrangement of terms, Equation (7.95) can be written as

$$|G|^2 = 1 + \left(\frac{2[1 + (\psi - \frac{1}{2})S\Delta t]}{[1 - (1 - \psi)S\Delta t]^2 + 4\lambda^2 \sin^2 \theta} \right) S\Delta t \quad (7.99)$$

Since the factor proportional to $\sin^2 \theta$ serves only to increase the denominator, we can omit it and say that

$$|G|^2 \leq 1 + \left(\frac{2[1 + (\psi - \frac{1}{2})S\Delta t]}{[1 - (1 - \psi)S\Delta t]^2} \right) S\Delta t \quad (7.100)$$

Clearly, the function in parentheses is bounded as $\Delta t \rightarrow 0$ as long as the denominator doesn't vanish, so that Equation (7.98) is satisfied provided:

$$\Delta t \leq \frac{1}{(1 - \psi)S}, \quad S > 0 \quad (7.101)$$

Although this analysis has been done for implicit Crank-Nicolson differencing of the convective term, the same result holds for explicit methods. While Equation (7.92) involves k_j^{n+1} , the terms in an explicit scheme can be rearranged to preserve its explicit nature. For example, if we use upwind differencing for the convective term in Equation (7.91), the discretized equation becomes

$$k_j^{n+1} = \frac{[1 + \psi S\Delta t - \frac{U\Delta t}{\Delta x}] k_j^n + \frac{U\Delta t}{\Delta x} k_{j-1}^n}{1 - (1 - \psi)S\Delta t} \quad (7.102)$$

We now have sufficient information to discuss the most suitable discretization approximations for source terms in both explicit and implicit methods. If second-order accuracy is required, as it would be for numerical simulation of an unsteady flow, ψ must be $1/2$. On the other hand, if only steady-state solutions are needed, we can take advantage of the fact that

using $\psi = 0$ when $S < 0$ and $\psi = 1$ when $S > 0$ yields an unconditionally stable (albeit first-order accurate in time) scheme. In summary, the following has proven satisfactory for turbulence model equations.

Second-Order Time Accuracy — Conditional Stability

$$Sk \doteq \frac{1}{2}S(k_j^n + k_j^{n+1}), \quad \Delta t \leq \frac{2}{|S|} \quad (7.103)$$

First-Order Time Accuracy — Unconditional Stability

$$Sk \doteq \begin{cases} Sk_j^{n+1} & \text{for } S < 0 \\ Sk_j^n & \text{for } S > 0 \end{cases} \quad (7.104)$$

All of the programs in Appendix C use Equation (7.104).

7.5 Block-Implicit Methods

The most efficient numerical methods currently available for complex flow fields are **block-implicit** methods. They differ from elementary implicit methods in one very important respect. Specifically, when an elementary implicit scheme is applied to a coupled set of equations, each equation is solved in sequence. In the context of a system of equations, this is usually referred to as a **sequentially-implicit** method. By contrast, a block-implicit scheme solves all of the equations simultaneously at each grid point. The block-implicit formulation, generally requiring inversion of block-tridiagonal matrices, entails more computational effort than a sequentially-implicit method. The additional computation at each grid point and timestep is usually compensated for by a dramatically improved convergence rate. Block-implicit solvers can achieve CFL numbers in excess of 100, and often converge in 100 to 200 timesteps for flows including boundary-layer separation. For example, using a block-implicit method, a supersonic two-dimensional shock-separated turbulent flow can be simulated on an 80486-based microcomputer in about 3 hours of CPU time [Wilcox (1991)]. On the same computer, a similar computation would take about 25 hours using a sequentially-implicit method [Wilcox (1990)] and 75 hours using an explicit method [Wilcox (1974)].

As in the preceding section, we begin with a brief overview of block-implicit methods. For simplicity, we focus on a one-dimensional system. The primary concern in this section is, of course, upon how turbulence model source terms impact such methods.

Consider the one-dimensional conservation equations for flow of a viscous, perfect gas, written in vector form, viz.,

$$\frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial}{\partial x} (\mathbf{F} - \mathbf{F}_v) = \mathbf{0} \quad (7.105)$$

where

$$\mathbf{Q} = \begin{Bmatrix} \bar{\rho} \\ \bar{\rho}\tilde{u} \\ \bar{\rho}E \end{Bmatrix}, \quad \mathbf{F} = \begin{Bmatrix} \bar{\rho}\tilde{u} \\ \bar{\rho}\tilde{u}^2 + P \\ (\bar{\rho}E + P)\tilde{u} \end{Bmatrix}, \quad \mathbf{F}_v = \begin{Bmatrix} 0 \\ \hat{\tau}_{xx} \\ \tilde{u}\hat{\tau}_{xx} - \hat{q}_x \end{Bmatrix} \quad (7.106)$$

where $\hat{\tau}_{xx}$ and \hat{q}_x denote total stress and heat flux, respectively. Also, the total energy for one-dimensional flow is $E = \tilde{e} + \frac{1}{2}\tilde{u}^2$ and the pressure is given by $P = (\gamma - 1)\bar{\rho}\tilde{e}$.

The first step often taken in establishing a block-implicit scheme for this system of equations is to introduce a first-order backward-difference (implicit backward-Euler) scheme, which can be written symbolically as follows.

$$\frac{\mathbf{Q}^{n+1} - \mathbf{Q}^n}{\Delta t} + \left[\frac{\partial}{\partial x} (\mathbf{F} - \mathbf{F}_v) \right]^{n+1} = \mathbf{0} \quad (7.107)$$

Now, we expand the flux vectors \mathbf{F} and \mathbf{F}_v in a Taylor series about time level n , wherefore

$$\mathbf{F}^{n+1} \doteq \mathbf{F}^n + \frac{\partial \mathbf{F}}{\partial t} \Delta t + O[(\Delta t)^2] \quad (7.108)$$

and similarly for \mathbf{F}_v . Then, using the chain rule of calculus, we have

$$\frac{\partial \mathbf{F}}{\partial t} = \frac{\partial \mathbf{F}}{\partial \mathbf{Q}} \frac{\partial \mathbf{Q}}{\partial t} \quad (7.109)$$

where $\partial \mathbf{F} / \partial \mathbf{Q}$ is the **inviscid-flux Jacobian matrix**. The incremental change in the dependent-variable vector, $\Delta \mathbf{Q}$, is defined by

$$\Delta \mathbf{Q} = \mathbf{Q}^{n+1} - \mathbf{Q}^n \quad (7.110)$$

Since we approximate the unsteady term according to $\partial \mathbf{Q} / \partial t \doteq \Delta \mathbf{Q} / \Delta t$, we can rewrite Equation (7.108) as

$$\mathbf{F}^{n+1} \doteq \mathbf{F}^n + \frac{\partial \mathbf{F}}{\partial \mathbf{Q}} \Delta \mathbf{Q} + O[(\Delta t)^2] \quad (7.111)$$

Because of the prominent role played by $\Delta \mathbf{Q}$, this approach is usually referred to as the **delta formulation**.

Finally, we must introduce a discretization approximation for the spatial derivatives of the vectors \mathbf{F} and \mathbf{F}_v . In general, this means forming a matrix that multiplies $(\mathbf{F} - \mathbf{F}_v)$, and yields a desired degree of accuracy. Details of this matrix are unimportant for our discussion, and it is sufficient to introduce symbolic notation with the understanding that an approximation to spatial differentiation is implied. Thus, we introduce a finite-difference matrix operator, δ_x , so that

$$\left[\frac{\partial}{\partial x} (\mathbf{F} - \mathbf{F}_v) \right]^{n+1} \doteq \delta_x (\mathbf{F}^n - \mathbf{F}_v^n) + \delta_x \left(\frac{\partial \mathbf{F}}{\partial \mathbf{Q}} - \frac{\partial \mathbf{F}_v}{\partial \mathbf{Q}} \right) \Delta \mathbf{Q} \quad (7.112)$$

where $\partial \mathbf{F}_v / \partial \mathbf{Q}$ is the **viscous-flux Jacobian matrix**. Collecting all of this, we arrive at the symbolic form of a typical block-implicit method:

$$\left[\frac{I}{\Delta t} + \delta_x \left(\frac{\partial \mathbf{F}}{\partial \mathbf{Q}} - \frac{\partial \mathbf{F}_v}{\partial \mathbf{Q}} \right) \right] \Delta \mathbf{Q} = -\delta_x (\mathbf{F}^n - \mathbf{F}_v^n) \quad (7.113)$$

where I is the unit (identity) matrix. The matrix multiplying $\Delta \mathbf{Q}$ in Equation (7.113) is of block-tridiagonal form. In the present example, the blocks are 3 by 3, corresponding to the three equations being solved simultaneously at each mesh point.

Now, suppose we choose to use a two-equation turbulence model to determine the Reynolds stress. The following three points that must be considered in modifying a block-implicit solution scheme.

1. Decide whether to solve all equations simultaneously or to solve the model equations and mean-flow equations sequentially.
2. If the preferred option is to solve all equations simultaneously, determine the changes to the flux-Jacobian matrices.
3. Make provision for handling source terms.

In principle, solving all equations simultaneously will yield the most rapidly convergent scheme in the number of iterations, but not necessarily in CPU time. However, the coupling between the turbulence-model equations and the mean-flow equations appears to be relatively weak. The primary coupling is through the diffusion terms, and the eddy viscosity is usually treated as a constant in forming the viscous-flux Jacobian matrix. Limited experience to date seems to indicate there is little advantage to solving all equations simultaneously as opposed to solving the model equations and mean-flow equations sequentially.

If all equations are solved simultaneously, the basic system of equations for the k - ω model would be as follows:

$$\frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial}{\partial x} (\mathbf{F} - \mathbf{F}_v) = \mathbf{S} \quad (7.114)$$

where the dependent-variable and inviscid-flux vectors are

$$\mathbf{Q} = \begin{Bmatrix} \bar{\rho} \\ \bar{\rho}\tilde{u} \\ \bar{\rho}E \\ \bar{\rho}k \\ \bar{\rho}\omega \end{Bmatrix}, \quad \mathbf{F} = \begin{Bmatrix} \bar{\rho}\tilde{u} \\ \bar{\rho}\tilde{u}^2 + P \\ (\bar{\rho}E + P)\tilde{u} \\ \bar{\rho}\tilde{u}k \\ \bar{\rho}\tilde{u}\omega \end{Bmatrix} \quad (7.115)$$

The viscous-flux and source-term vectors are given by

$$\mathbf{F}_v = \begin{Bmatrix} 0 \\ \tilde{u} \left(\frac{4}{3}\mu \frac{\partial \tilde{u}}{\partial x} + \tau_{xx} \right) \\ \tilde{u} \left(\frac{4}{3}\mu \frac{\partial \tilde{u}}{\partial x} + \tau_{xx} \right) - \hat{q}_x \\ (\mu + \sigma^* \mu_T) \frac{\partial k}{\partial x} \\ (\mu + \sigma \mu_T) \frac{\partial \omega}{\partial x} \end{Bmatrix}, \quad \mathbf{S} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ \tau_{xx} \frac{\partial \tilde{u}}{\partial x} - \beta^* \bar{\rho} \omega k \\ \alpha \left(\frac{\omega}{k} \right) \tau_{xx} \frac{\partial \tilde{u}}{\partial x} - \beta \bar{\rho} \omega^2 \end{Bmatrix} \quad (7.116)$$

There are two places where the turbulence kinetic energy appears that has an impact on the flux-Jacobian matrices. Specifically, the total energy, E , should be written as

$$E = \tilde{e} + \frac{1}{2}\tilde{u}^2 + k \quad (7.117)$$

and the Reynolds stress tensor is

$$\tau_{xx} = \frac{4}{3}\mu_T \frac{\partial \tilde{u}}{\partial x} - \frac{2}{3}\bar{\rho}k \quad (7.118)$$

Hence, since the vector \mathbf{Q} contains $\bar{\rho}k$ as one of its elements, the inviscid- and viscous-flux Jacobian matrices must be evaluated from scratch. Some of the original 9 elements appropriate for laminar flow or an algebraic model will be affected by the appearance of k in E and $\hat{\tau}_{xx}$. For this system, the inviscid-flux Jacobian matrix assumes the following form:

$$\frac{\partial \mathbf{F}}{\partial \mathbf{Q}} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ \left(\frac{\gamma-3}{2} \right) \tilde{u}^2 & (3-\gamma)\tilde{u} & (\gamma-1) & -(\gamma-1) & 0 \\ - \left[H - \frac{\gamma-1}{2} \tilde{u}^2 \right] \tilde{u} & \left[H - (\gamma-1)\tilde{u}^2 \right] & \gamma\tilde{u} & -(\gamma-1)\tilde{u} & 0 \\ -\tilde{u}k & k & 0 & \tilde{u} & 0 \\ -\tilde{u}\omega & \omega & 0 & 0 & \tilde{u} \end{bmatrix} \quad (7.119)$$

where H is the total enthalpy defined by

$$H = \tilde{h} + \frac{1}{2}\tilde{u}^2 + k \quad (7.120)$$

As shown in Equation (7.119), the first two components on row 3 involve H , and are thus affected by k . In modifying an existing computer program

based on this block-implicit scheme, all that would be required to modify the inviscid-flux Jacobian matrix components would be to have H appear as indicated, and to include k in the computation of H .

By contrast, if we choose to solve the mean-flow and turbulence model equations sequentially, we retain the original conservation equations [Equation (7.105)]. All of the flux-Jacobian matrices and, in fact, the entire algorithm remain the same. To determine k and ω , we then consider the following vector equation:

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial}{\partial x} (\mathbf{f} - \mathbf{f}_v) = \mathbf{s} \quad (7.121)$$

where

$$\mathbf{q} = \begin{Bmatrix} \bar{\rho}k \\ \bar{\rho}\omega \end{Bmatrix}, \quad \mathbf{f} = \begin{Bmatrix} \bar{\rho}\tilde{u}k \\ \bar{\rho}\tilde{u}\omega \end{Bmatrix}, \quad \mathbf{f}_v = \begin{Bmatrix} (\mu + \sigma^* \mu_T) \frac{\partial k}{\partial x} \\ (\mu + \sigma \mu_T) \frac{\partial \omega}{\partial x} \end{Bmatrix} \quad (7.122)$$

$$\mathbf{s} = \begin{Bmatrix} \tau_{xx} \frac{\partial \tilde{u}}{\partial x} - \beta^* \bar{\rho} \omega k \\ \alpha \left(\frac{\omega}{k} \right) \tau_{xx} \frac{\partial \tilde{u}}{\partial x} - \beta \bar{\rho} \omega^2 \end{Bmatrix} \quad (7.123)$$

Consistent with the block-implicit approach, we linearize the flux and source vectors according to

$$(\mathbf{f} - \mathbf{f}_v)^{n+1} \doteq (\mathbf{f} - \mathbf{f}_v)^n + \left(\frac{\partial \mathbf{f}}{\partial \mathbf{q}} - \frac{\partial \mathbf{f}_v}{\partial \mathbf{q}} \right) \Delta \mathbf{q} \quad (7.124)$$

$$\mathbf{s} \doteq \mathbf{s}^n + \frac{\partial \mathbf{s}}{\partial \mathbf{q}} \Delta \mathbf{q} \quad (7.125)$$

where $\partial \mathbf{s} / \partial \mathbf{q}$ is the **source-Jacobian matrix**. The flux-Jacobian matrices are generally much simpler than their counterparts in the mean-flow equations. For example, the inviscid-flux Jacobian matrix is

$$\frac{\partial \mathbf{f}}{\partial \mathbf{q}} = \begin{bmatrix} \tilde{u} & 0 \\ 0 & \tilde{u} \end{bmatrix} \quad (7.126)$$

This brings us to the all important question of how to handle the source-term vector \mathbf{s} . Several prescriptions are possible, and the primary considerations are to: maintain numerical stability; achieve rapid convergence rate; and guarantee that k and ω are positive definite. Wilcox (1991) has found the following linearization of the source terms to be quite satisfactory for the k - ω model, within the framework of MacCormack's (1985) block-implicit method. Specifically, the source-term vector is rearranged as follows.

$$\mathbf{s} = \begin{Bmatrix} \tau_{xx} \frac{\partial \tilde{u}}{\partial x} - \beta^* \left(\frac{\omega}{k} \right) \frac{(\bar{\rho}k)^2}{\bar{\rho}} \\ \alpha \left(\frac{\omega}{k} \right) \tau_{xx} \frac{\partial \tilde{u}}{\partial x} - \beta \frac{(\bar{\rho}\omega)^2}{\bar{\rho}} \end{Bmatrix} \quad (7.127)$$

Then, treating $\tau_{xx}\partial\tilde{u}/\partial x$ and ω/k as constant in computing the source-Jacobian matrix, we arrive at

$$\mathbf{s}^n = \left\{ \begin{array}{c} \tau_{xx} \frac{\partial\tilde{u}}{\partial x} - \beta^* \bar{\rho} \omega k \\ \alpha \left(\frac{\omega}{k} \right) \tau_{xx} \frac{\partial\tilde{u}}{\partial x} - \beta \bar{\rho} \omega^2 \end{array} \right\}, \quad \frac{\partial \mathbf{s}}{\partial \mathbf{q}} = \begin{bmatrix} -2\beta^* \omega & 0 \\ 0 & -2\beta \omega \end{bmatrix} \quad (7.128)$$

In this treatment of the source-term vector the production terms are evaluated explicitly (i.e., computed at time level n), and the dissipation terms are treated implicitly (computed at time level $n + 1$). The block-tridiagonal scheme for the turbulence model equations becomes

$$\left[\frac{I}{\Delta t} + \delta_x \left(\frac{\partial \mathbf{f}}{\partial \mathbf{q}} - \frac{\partial \mathbf{f}_v}{\partial \mathbf{q}} \right) - \frac{\partial \mathbf{s}}{\partial \mathbf{q}} \right] \Delta \mathbf{q} = -\delta_x (\mathbf{f}^n - \mathbf{f}_v^n) + \mathbf{s}^n \quad (7.129)$$

Since $\partial \mathbf{s} / \partial \mathbf{q}$ is a diagonal matrix and its diagonal elements are always negative, its contribution is guaranteed to enhance diagonal dominance of the matrix multiplying $\Delta \mathbf{q}$. Additionally, Spalart and Allmaras (1992) show that this form guarantees that k and ω (or ϵ for a k - ϵ model) will always be positive.

However, Spalart and Allmaras also point out that in regions where production and dissipation are both large and dominate the overall balance of terms in the equation, this form can result in slow convergence. This appears to be a more serious problem for the k - ϵ model than it is for the k - ω model. Wilcox (1991), for example, has shown that the scheme described above yields very rapid convergence in flows with attached equilibrium boundary layers and in flows with large regions of separation. The procedure recommended by Spalart and Allmaras is similar to the procedure recommended for elementary implicit methods in Equation (7.104). That is, they recommend linearizing the source term according to

$$\mathbf{s} \doteq \mathbf{s}^n + \text{neg} \left(\frac{\partial \mathbf{s}}{\partial \mathbf{q}} \right) \Delta \mathbf{q} \quad (7.130)$$

where the function $\text{neg}(x)$ is defined as

$$\text{neg}(x) = \begin{cases} x, & x < 0 \\ 0, & x \geq 0 \end{cases} \quad (7.131)$$

The production terms are then included in computing the source-Jacobian matrix. The neg operator is understood to apply to each element of the resulting (diagonal) matrix. Thus, as long as dissipation exceeds production, both production and dissipation are treated implicitly, and explicitly when production exceeds dissipation. Huang and Coakley (1992) have successfully applied a linearization similar to that recommended by Spalart and Allmaras.

7.6 Solution Convergence and Grid Sensitivity

Regardless of the application, there is a need for control of numerical accuracy in CFD [Roache (1990)]. This need is just as critical in CFD work as it is in experiments where the experimenter is expected to provide estimates for the accuracy of his or her measurements. All CFD texts of any value stress this need.

One key issue determining numerical accuracy is **iteration convergence**. Most numerical methods used in CFD applications require many iterations to converge. The iteration convergence error is defined as the difference between the current iterate and the exact solution to the difference equations. Often, the difference between successive iterates is used as a measure of the error in the converged solution, although this in itself is inadequate. A small relaxation factor can always give a false indication of convergence [Anderson et al. (1984)]. Whatever the algorithm is, you should always be careful to check that a converged solution has been obtained. This can be done by trying a stricter than usual convergence criterion, and demonstrating that there is a negligible effect on the solution. Most reputable engineering journals require demonstration of iteration convergence as a condition for publication. This is not specific to turbulence-model applications — all of the usual criteria for standard CFD applications apply.

Specific to turbulence-model computations, the approach to iteration convergence often is more erratic, and typically much slower, than for laminar-flow computations. A variety of factors including stiffness and non-linearity of the equations, as well as the severely stretched finite-difference grids needed to resolve thin viscous layers, yield less rapid and less monotone convergence. Ferziger (1989) explains the slow convergence often observed in terms of the eigenvalues of the matrix system corresponding to the discretized equations. He notes that any iteration scheme for a linear system can be written as

$$\phi^{n+1} = A\phi^n + S \quad (7.132)$$

where ϕ^n is the solution after the n^{th} iteration, A is a matrix, and S is a source term. He then shows that the actual solution error is given by

$$\phi_{\text{exact}} - \phi^n \approx \frac{\phi^{n+1} - \phi^n}{1 - \lambda_{\text{max}}} \quad (7.133)$$

where ϕ_{exact} denotes the **exact solution to the discretized equations** and λ_{max} is the largest eigenvalue of the matrix A . Of course, all eigenvalues of A must be less than 1 for the solution to converge. This result

shows that the solution error is larger than the difference between iterates. Furthermore, the closer λ_{max} is to 1, the larger the ratio of solution error to the difference between iterates. In other words, the slower the rate of convergence of the method, the smaller the difference between iterates must be to guarantee iteration convergence.

A second key issue is **grid convergence** or **grid insensitivity**. Because of the finite size of finite-difference cells, discretization errors exist that represent the difference between the solution to the difference equations and the **exact (continuum) solution to the differential equations**. It is important to know the magnitude of these discretization errors and to insure that a fine enough grid has been used to reduce the error to an acceptable level.

As with iteration convergence, all CFD work should demonstrate grid convergence, regardless of what equations are being solved. In most engineering journals, it is no longer sufficient to publish results performed on a single fixed grid. While grid sensitivity studies should be done for all CFD work, they are even more crucial for turbulence-model computations because of the need to separate numerical error from turbulence-model error. This issue came into sharp focus at the 1980-81 AFOSR-HTTM-Stanford Conference on Complex Turbulent Flows [see Kline, Cantwell, and Lilley (1981)]. Clearly, no objective evaluation of the merits of different turbulence models can be made unless the discretization error of the numerical algorithm is known.

The most common way to demonstrate grid convergence is to repeat a computation on a grid with twice as many grid points, and compare the two solutions. If computer resources are unavailable to facilitate a grid doubling, a grid halving is also appropriate, although the error bounds will not be as sharp. Using results for two different grids, techniques such as **Richardson extrapolation** [see Roache (1976)] can be used to determine discretization error. This method is very simple to implement, and should be used whenever possible. For a second-order accurate method with central differences, Richardson extrapolation assumes the error, $E_h \equiv \phi_{exact} - \phi_h$, where ϕ_h denotes the solution when the grid-point spacing is h , can be expanded as a Taylor series in h , wherefore

$$E_h = e_2 h^2 + e_4 h^4 + e_6 h^6 + \dots \quad (7.134)$$

Note that for three-point upwind differences the leading term is still $e_2 h^2$, but the next term is $e_3 h^2$, and Richardson extrapolation is only $O(h^3)$ rather than $O(h^4)$. By hypothesis, the e_i are, at worst, functions of the coordinates, but are nevertheless independent of h . Now, if we halve the

number of grid points so that h is doubled, the error is given by

$$E_{2h} = 4e_2h^2 + 16e_4h^4 + 64e_6h^6 + \dots \quad (7.135)$$

For small values of h , we can drop all but the leading terms, whence the discretization error is given by

$$E_h \approx \frac{1}{3}(\phi_h - \phi_{2h}) \quad (7.136)$$

As a final comment, Richardson extrapolation has limitations. First, if it is applied to primitive variables such as velocity and internal energy, its implications regarding momentum and energy conservation may be inaccurate. Second, the method implicitly assumes the solution has continuous derivatives to all orders. Hence, its results are not meaningful near shock waves or turbulent/nonturbulent interfaces of the type discussed in Subsection 7.2.2.

There is another grid-related factor affecting solution accuracy. In order to resolve thin viscous layers, for example, highly stretched grids are normally used. Conventional central-difference approximations are only first-order accurate on such a grid, and care must be taken to account for this. Also, the location of the grid point nearest the surface has a nontrivial effect on the accuracy of skin friction and surface heat flux. Wilcox (1989), for example, has found that grid-insensitive computations using wall functions that account for pressure gradient [e.g., Equation (5.111)] can be obtained with block-implicit methods provided:

$$10 < y_2^+ < 100, \quad (\text{wall functions}) \quad (7.137)$$

where y_2^+ is the sublayer-scaled value of the first grid point above the surface. This range appears to hold for boundary-layer computations as well [Chambers and Wilcox (1977)], again provided pressure gradient is accounted for. When turbulence-model equations are integrated through the viscous sublayer, many researchers have shown that it is imperative to require:

$$y_2^+ < 1, \quad (\text{integration through the sublayer}) \quad (7.138)$$

When these limits are not adhered to, consistent with the discussion in Subsection 7.2.1, solution errors throughout the boundary layer generally are large.

Problems

7.1 For a Mach 3 turbulent flat-plate boundary layer, it is a fact that $Mc_f Re_L \approx Re_{\delta^*}$.

- In the viscous sublayer, the appropriate scaling for the specific dissipation rate is $\omega \sim u_\tau^2/\nu$. Noting that $u_\tau \approx U\sqrt{c_f}$, express the ratio of t_{diss} to t_{wave} as a function of Re_{δ^*} in the sublayer.
- In the defect layer, the appropriate scaling for the specific dissipation rate is $\omega \sim u_\tau/\Delta$ where $\Delta = U\delta^*/u_\tau$. Express the ratio of t_{diss} to t_{wave} as a function of Re_{δ^*} in the defect layer.
- Comment on the implications of your estimates in Parts (a) and (b).

7.2 Determine whether or not the following systems of equations are stiff with regard to the specified initial conditions.

(a)

$$\frac{d}{dt} \begin{Bmatrix} x \\ y \end{Bmatrix} = \begin{bmatrix} -3 & 4 \\ 4 & 3 \end{bmatrix} \begin{Bmatrix} x \\ y \end{Bmatrix}, \quad \begin{Bmatrix} \dot{x}(0) \\ \dot{y}(0) \end{Bmatrix} = -5 \begin{Bmatrix} x(0) \\ y(0) \end{Bmatrix}$$

(b)

$$\frac{d}{dt} \begin{Bmatrix} x \\ y \end{Bmatrix} = \begin{bmatrix} -3 & 1 \\ 4 & -3 \end{bmatrix} \begin{Bmatrix} x \\ y \end{Bmatrix}, \quad \begin{Bmatrix} \dot{x}(0) \\ \dot{y}(0) \end{Bmatrix} = - \begin{Bmatrix} x(0) \\ y(0) \end{Bmatrix}$$

7.3 Consider the high-Reynolds-number k - ω model's near-wall variation of specific dissipation rate, ω , for a rough wall, i.e.,

$$\omega = \frac{\omega_w}{[1 + Ay]^2}, \quad A = \sqrt{\frac{\beta\omega_w}{6\nu_w}}$$

- Assuming equally spaced grid points, show that the central-difference approximation to $d^2\omega/dy^2$ at the first grid point above the surface (i.e., at $y = \Delta y$) is given by

$$\left(\frac{d^2\omega}{dy^2}\right)_2 \approx \Phi(\Delta y) \left(\frac{d^2\omega}{dy^2}\right)_{exact}$$

where

$$\Phi(\Delta y) = \frac{[1 + A\Delta y]^2 [1 + 2A\Delta y + \frac{2}{3}(A\Delta y)^2]}{[1 + 2A\Delta y]^2}$$

(b) Assuming a slightly-rough wall so that

$$\omega_w = \frac{2500\nu_w}{k_R^2}$$

and that $\beta = 3/40$, show that

$$A\Delta y = \frac{5\sqrt{5}}{2} \frac{\Delta y^+}{k_R^+}$$

(c) Determine the percentage error introduced by the central-difference approximation in computing $d^2\omega/dy^2$ when we assume a hydraulically smooth wall so that $k_R^+ = 5$, and set $\Delta y^+ = 1/\sqrt{5} = 0.447$.

7.4 This problem shows that while trapezoidal-rule integration is second-order accurate for a piecewise continuous function with a discontinuous first derivative, the truncation error depends upon placement of the nodes. Using the trapezoidal rule, the integral of a function $f(x)$ is

$$\int_a^b f(x) dx \approx \sum_{k=1}^N f(x_k)\Delta x + \frac{1}{2}[f(a) - f(b)]\Delta x$$

where

$$x_k = k\Delta x \quad \text{and} \quad \Delta x = \frac{b-a}{N}$$

Consider the following piecewise continuous function $f(x)$:

$$f(x) = \begin{cases} x^2, & 0 \leq x \leq 1 \\ 1, & 1 < x \leq 2 \end{cases}$$

Note that a node lies at $x = 1$ only for even values of N .

(a) Verify that the exact integral of $f(x)$ for x ranging from 0 to 2 is

$$I \equiv \int_0^2 f(x) dx = \frac{4}{3}$$

(b) Assuming N is odd show that the trapezoidal rule yields

$$I \approx \frac{4}{3} \left[1 - \frac{1}{16}(\Delta x)^2 \right]$$

(c) Assuming N is even show that the trapezoidal rule yields

$$I \approx \frac{4}{3} \left[1 + \frac{1}{8}(\Delta x)^2 \right]$$

HINT: Use the fact that $\sum_{k=1}^M k^2 = \frac{1}{6}M(M+1)(2M+1)$.

7.5 Consider the mixing-length model with $\ell_{mix} = \alpha\delta$, where α is a constant and δ is shear layer thickness.

- (a) Assuming that $dU/dy > 0$, verify that according to the Rubel-Melnik transformation,

$$\nu_T = \ell_{mix} \sqrt{\frac{dU}{d\xi}}$$

- (b) For flow near a turbulent/nonturbulent interface with constant entrainment velocity, $V < 0$, determine the velocity difference, $U_e - U$, as $y \rightarrow \delta$. Express your answer as a function of $|V|$, α and y/δ .

7.6 The object of this problem is to verify that Equation (7.55) is the solution to Equations (7.52) - (7.54).

- (a) Integrate Equations (7.52) once and impose the freestream boundary condition [Equation (7.54)].
- (b) Observing that $\sigma = \sigma^* = 1/2$ for Saffman's model, combine the k and ω^2 equations to show that

$$\frac{dk}{d\omega^2} = \frac{k}{\omega^2}$$

Solve this equation subject to the boundary conditions.

- (c) Introduce the dimensionless variables

$$\bar{y} = \frac{|V|(\delta - y)}{\nu} \quad \text{and} \quad \bar{\omega} = \frac{V^2 \omega}{\mathcal{K}\nu}$$

and substitute the solution for k into the equation for ω . Set any arbitrary constant of integration equal to zero, and verify the solution for $y \rightarrow \delta$.

- (d) Letting $\bar{U} = U/|V|$, rewrite the momentum equation. Using the dimensionless equation for $\bar{\omega}$ derived in Part (c), verify that

$$\left(\frac{1 + \bar{\omega}}{2 + \bar{\omega}} \right) \frac{d\bar{U}}{d\bar{\omega}} = \frac{\bar{U} - \bar{U}_e}{\bar{\omega}}$$

and verify the solution for $U_e - U$.

7.7 This problem illustrates how nonlinear terms affect numerical stability for parabolic marching methods. Consider the following limiting form of the k - ω model.

$$U \frac{\partial \omega}{\partial x} = \alpha \left(\frac{\partial U}{\partial y} \right)^2 - \beta \omega^2$$

We wish to analyze the stability of the solution to this equation under the following discretization approximations.

$$U \frac{\partial \omega}{\partial x} \doteq \frac{U}{\Delta x} [3\omega_{m+1}^i - 4\omega_m + \omega_{m-1}]$$

$$\alpha \left(\frac{\partial U}{\partial y} \right)^2 \doteq \frac{\alpha (\partial U / \partial y)^2}{\omega_{m+1}^{i-1}} \omega_{m+1}^i$$

$$\beta \omega^2 \doteq (1 + \psi_\omega) \beta \omega_{m+1}^{i-1} \omega_{m+1}^i - \psi_\omega \beta (\omega_{m+1}^{i-1})^2$$

- (a) Assuming that ω_{m+1}^i is the sum of the exact solution to the discretized equation, ω_{m+1} , and an error term, $\delta \omega^i$, viz.,

$$\omega_{m+1}^i = \omega_{m+1} + \delta \omega^i$$

linearize the discretized equation for ω and verify that

$$\frac{\delta \omega^i}{\delta \omega^{i-1}} = \frac{(\psi_\omega - 1) \beta \omega_{m+1}^2 - \alpha (\partial U / \partial y)^2}{3U \omega_{m+1} / \Delta x - \alpha (\partial U / \partial y)^2 + (\psi_\omega + 1) \beta \omega_{m+1}^2}$$

- (b) Using the fact that ω_{m+1} satisfies the exact discretized equation, simplify the denominator and show that

$$\frac{\delta \omega^i}{\delta \omega^{i-1}} = \frac{(\psi_\omega - 1) - \alpha (\partial U / \partial y)^2 / (\beta \omega_{m+1}^2)}{\psi_\omega + U(4\omega_m - \omega_{m-1}) / (\omega_{m+1}^2 \Delta x)}$$

- (c) Assuming the term proportional to U is negligible, determine the condition that ψ_ω must satisfy in order to insure that $|\delta \omega^i / \delta \omega^{i-1}| < 1$.

7.8 Using von Neumann stability analysis, determine G and any condition required for stability of the following finite-difference schemes. Assume $U > 0$, $\nu > 0$ and $S < 0$.

(a) Euler's method:

$$k_j^{n+1} = k_j^n - \frac{U\Delta t}{2\Delta x} (k_{j+1}^{n+1} - k_{j-1}^{n+1})$$

(b) Richardson's method:

$$k_j^{n+1} = k_j^{n-1} + \frac{2\nu\Delta t}{(\Delta x)^2} (k_{j+1}^n - 2k_j^n + k_{j-1}^n)$$

(c) Crank and Nicolson's method:

$$k_j^{n+1} = k_j^n - \frac{U\Delta t}{4\Delta x} (k_{j+1}^{n+1} + k_{j+1}^n - k_{j-1}^{n+1} - k_{j-1}^n) + \frac{1}{2}S\Delta t (k_j^{n+1} + k_j^n)$$

7.9 Consider the following one-dimensional wave equation with source and diffusion terms.

$$\frac{\partial k}{\partial t} + U \frac{\partial k}{\partial x} = Sk + \nu \frac{\partial^2 k}{\partial x^2}$$

where $U > 0$, $\nu > 0$ and S can be either positive or negative.

(a) Cast this equation in finite-difference form using Crank-Nicolson differencing and the following approximation for the source term.

$$Sk \doteq S [\psi k_j^n + (1 - \psi)k_j^{n+1}], \quad 0 \leq \psi \leq 1$$

(b) Using von Neumann stability analysis, determine G and any condition required for stability of this finite-difference scheme. How do your results compare to the analysis of Equation (7.91) in Section 7.4?

7.10 Verify that the dependent-variable and inviscid-flux vectors in Equation (7.115) can be written as

$$\mathbf{Q} = \begin{Bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \\ Q_5 \end{Bmatrix}, \quad \mathbf{F} = \begin{Bmatrix} Q_2 \\ (\frac{3-\gamma}{2})Q_2^2/Q_1 + (\gamma-1)Q_3 - (\gamma-1)Q_4 \\ \gamma Q_2 Q_3 / Q_1 - (\frac{\gamma-1}{2})Q_2^3 / Q_1^2 - (\gamma-1)Q_2 Q_4 / Q_1 \\ Q_2 Q_4 / Q_1 \\ Q_2 Q_5 / Q_1 \end{Bmatrix}$$

and show that the flux-Jacobian matrix is given by Equation (7.119).

7.11 The following table represents partial results for a one-dimensional finite-difference computation using a second-order accurate, time-marching method. The computation was done on grids with 50, 100 and 200 points. Use Richardson extrapolation to estimate the discretization error at each point for the two finest grids. Based on your results, make a table of the results below and add a column with your best estimate of the continuum solution (grid-point spacing $\rightarrow 0$) to the differential equation.

j	ϕ_{50}	j	ϕ_{100}	j	ϕ_{200}
1	.5592	1	.5628	1	.5607
2	.5700	3	.5740	5	.5726
3	.5737	5	.5748	9	.5745
4	.5615	7	.5557	13	.5573