Electronic Transactions on Numerical Analysis.
Volume 6, pp. 234-245, December 1997.

ISSN 1068-9613.

ETNA
Kent State University
etna@mcs.kent.edu

A MULTIGRID SMOOTHER FOR HIGH REYNOLDS NUMBER FLOWS*

ERIK STERNER†

Abstract. The linearized Navier–Stokes equations are solved in two space dimensions using a multigrid method where a semi-implicit Runge–Kutta scheme is the smoother. Explicit time-integration in the streamwise direction is combined with implicit integration in the body-normal direction. Thereby the stiffness of the equations due to the disparate scales in the boundary layer is removed. Reynolds number independent convergence is demonstrated in analysis as well as in numerical experiments.

Key words. Navier–Stokes equations, semi-implicit, multigrid, convergence acceleration.

AMS subject classifications. 65L06, 65M12, 76N20.

1. Introduction. The solution to the steady compressible Navier–Stokes equations is often found by the method of lines. Space discretization leads to a large system of equations, which may be solved by integrating in time with an explicit Runge–Kutta scheme until a steady state is reached. This is a fairly straightforward and robust method, but for some viscous calculations it can be very time consuming. One way to speed up the calculations is to use a *semi-implicit Runge–Kutta scheme*, where explicit integration in the streamwise direction is combined with implicit integration in the body-normal direction, see [14]. Here we use the semi-implicit scheme as the smoother in a multigrid method. For a linear model problem this technique leads to Reynolds number independent convergence, whereas the number of iterations increases by $Re^{1/2}$ when the smoother is an explicit Runge–Kutta scheme.

2. The linearized Navier–Stokes equations. We restrict ourselves to two space dimensions and neglect the body force and the heat flux. The compressible Navier–Stokes equations can be written as

$$\frac{\partial W}{\partial t} + \frac{\partial F^c}{\partial x} + \frac{\partial G^c}{\partial y} = \frac{\partial F^v}{\partial x} + \frac{\partial G^v}{\partial y}, \tag{2.1}$$

where $W = (\rho, \rho u, \rho v, \rho E)^T$, ρ is the density, u and v the Cartesian velocity component and E the total energy. There are two convective flux vectors, F^c and G^c, and two viscous flux vectors, F^v and G^v. The system is closed by an equation of state that relates the pressure to the other variables, $p = (\gamma - 1)\rho(E - (u^2 + v^2)/2)$, where γ is the ratio of specific heats.

For a steady incompressible flow along a flat plate with a free-stream velocity U_∞ parallel to the x-axis, (2.1) may be simplified, yielding the *boundary layer equation*,

$$\begin{aligned} u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} &= \nu\frac{\partial^2 u}{\partial y^2} \\ \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} &= 0. \end{aligned} \tag{2.2}$$

Here, the boundary conditions are $u = v = 0$ at $y = 0$, and $u = U_\infty$ at $y = \infty$; see, e.g., [12]. The kinematic viscosity is $\nu = \mu/\rho$, the coefficient of viscosity μ, the density is $\rho = 1$ and the Reynolds number is defined as

$$Re = \frac{U_\infty L}{\nu}, \tag{2.3}$$

where L is the length scale. Exploiting the transformation $\eta = y(U_\infty/\nu x)^{1/2}$, the system of PDEs (2.2) is transformed into an ODE, the Blasius equation, which can easily be solved numerically. Back substituting gives us the solution

* Received May 15, 1997. Accepted for publication September 26, 1997. Communicated by D. Melson.
† Department of Scientific Computing, Uppsala University, Box 120, S-751 04 Uppsala, Sweden (erik@tdb.uu.se)

$$W_{Blasius} = (U(x,y)\ ,\ V(x,y)\ ,\ R(x,y))^T, \tag{2.4}$$

where $U(x,y)$ and $V(x,y)$ are the velocity profiles and $R(x,y) = 1$ is the density. We also obtain an expression for the *boundary layer thickness*,

$$\delta \approx 5.2 \left(\frac{\nu x}{U_\infty}\right)^{1/2}, \tag{2.5}$$

which is the distance from the plate where $u \approx 0.99 \cdot U_\infty$. Note that $\delta \sim Re^{-1/2}$, i.e., the boundary layers are very thin for high *Re*-flows.

By linearizing the compressible Navier–Stokes equations (2.1) around $W_{Blasius}$ we obtain a *linear* system of PDEs

$$\frac{\partial W}{\partial t} + A_1\frac{\partial W}{\partial x} + A_2\frac{\partial W}{\partial y} = B_1\frac{\partial^2 W}{\partial x^2} + B_2\frac{\partial^2 W}{\partial y^2} + B_3\frac{\partial^2 W}{\partial x\partial y} + F, \tag{2.6}$$

$$\begin{gathered} B_1 = \nu\begin{pmatrix} \frac{4}{3} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix},\quad B_2 = \nu\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{4}{3} & 0 \\ 0 & 0 & 0 \end{pmatrix},\quad B_3 = \nu\begin{pmatrix} 0 & \frac{1}{3} & 0 \\ \frac{1}{3} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \\ A_1 = \begin{pmatrix} U & 0 & \frac{\kappa}{R} \\ 0 & U & 0 \\ R & 0 & U \end{pmatrix},\quad A_2 = \begin{pmatrix} V & 0 & 0 \\ 0 & V & \frac{\kappa}{R} \\ 0 & R & V \end{pmatrix}, \end{gathered} \tag{2.7}$$

where $W = (u, v, \rho)^T$; cf. [7]. The pressure has been eliminated by the relation $c^2 = \gamma \cdot p/\rho$, where we set $\gamma = 1$. The function F in (2.6) is a sum of spatial derivatives of $W_{Blasius}$. In the sequel we let $F \equiv 0$ since it has no influence on the convergence. We also set $c = 1$ and thus $\kappa = c^2/\gamma = 1$. If Re is not too low and the Mach number, U_∞/c, is not too high, our linear model problem approximates very well the solution to the Navier–Stokes equations (2.1) for the subsonic flow over a flat plate.

3. Space discretization. The domain $0 \le x \le 1$ and $0 \le y \le 1$ is subdivided into $M \times N$ cells. Along the x-axis there is a flat plate and the other boundaries are open. We only consider rectangular cells, but the time-stepping methods we study can be used for cells with arbitrary shape as long as the mesh is structured. In the y-direction the grid is stretched by

$$y(\zeta) = \frac{\exp\alpha\zeta - 1}{\exp\alpha - 1}, \tag{3.1}$$

where α is a constant, $\zeta = j/N$ and $j = 0, 1, \ldots, N$. This gives a constant stretching factor $\Delta y_{j+1}/\Delta y_j = 1 + \theta$, where $\theta = \exp(\alpha/N) - 1$.

A standard technique to discretize the Navier–Stokes equations in space is the cell centered finite volume method described in [6]. The discretization of the linearized Navier–Stokes equations can be performed in a similar manner. Applying (2.1) on integral form to each cell we obtain

$$\frac{dW_{i,j}}{dt} + R_{i,j} = 0, \quad R_{i,j} = \frac{1}{S_{i,j}}(Q_{i,j} - D_{i,j}), \tag{3.2}$$

where $W_{i,j}$ is an approximation of the flow quantity W at the center of cell (i,j), $R_{i,j}$ is the residual and $S_{i,j}$ is the area of the cell with edges of lengths Δx_i and Δy_j. In the calculation of the physical flux, $Q_{i,j}$, the values at the cell faces are computed by taking the average of the values in the centers of the two adjacent cells.

Artificial dissipation is introduced as fluxes

$$
\begin{aligned}
D_{i,j} &= d_{i+1/2,j} - d_{i-1/2,j} + d_{i,j+1/2} - d_{i,j-1/2}, \\
d_{i+1/2,j} &= -\varepsilon \cdot \alpha_{i+1/2,j}(W_{i+2,j} - 3W_{i+1,j} + 3W_{i,j} - W_{i-1,j}) \\
d_{i,j+1/2} &= -\varepsilon \cdot \alpha_{i,j+1/2}(W_{i,j+2} - 3W_{i,j+1} + 3W_{i,j} - W_{i,j-1}),
\end{aligned}
\tag{3.3}
$$

where ε is a small constant. For explicit Runge–Kutta schemes we set

$$\alpha_{i+1/2,j} = \lambda^I = \Delta y(|u| + c) \qquad \alpha_{i,j+1/2} = \lambda^J = \Delta x(|v| + c)$$

according to the *scalar anisotropic model* in [9]. However, for the semi-implicit schemes this may lead to an excessive amount of damping and therefore both α-factors are scaled by λ^I, which improves the accuracy of the numerical solution in boundary layers where $\lambda^J \gg \lambda^I$.

Finally boundary conditions are introduced by using two layers of ghost cells around the computational domain. At the solid wall the non slip condition is imposed and open boundaries are taken care of by using Riemann invariants; see, e.g., [5].

4. Time integration. Applying the space discretization above to the linearized Navier–Stokes equations we obtain a linear system of ODEs

$$\frac{dW}{dt} + AW = b, \tag{4.1}$$

where W is the vector of the flow variables, A is the discretization matrix and b is the right hand side vector with contributions from the boundary conditions. We seek the stationary solution

$$AW = b. \tag{4.2}$$

In computational fluid dynamics this has traditionally been done with an explicit *m*-stage Runge–Kutta scheme (ERK) as in [6]:

$$
\begin{aligned}
W^{(0)} &= W^n \\
W^{(1)} &= W^{(0)} + \alpha_1 \Delta t(b - AW^{(0)}) \\
W^{(2)} &= W^{(0)} + \alpha_2 \Delta t(b - AW^{(1)}) \\
&\vdots \\
W^{(m)} &= W^{(0)} + \alpha_m \Delta t(b - AW^{(m-1)}) \\
W^{n+1} &= W^{(m)},
\end{aligned}
\tag{4.3}
$$

where Δt is a diagonal matrix.

Another approach is to apply a semi-implicit time-integration technique. Consider the splitting $AW = (B + C)W$, where

$$
\begin{aligned}
A_2 \frac{\partial W}{\partial y} - B_2 \frac{\partial^2 W}{\partial y^2} &\overset{\text{discretization}}{\Longrightarrow} BW \\
A_1 \frac{\partial W}{\partial x} - B_1 \frac{\partial^2 W}{\partial x^2} - B_3 \frac{\partial^2 W}{\partial x \partial y} &\overset{\text{discretization}}{\Longrightarrow} CW
\end{aligned}
\tag{4.4}
$$

with the matrices A_1, A_2, B_1, B_2 and B_3 given by (2.7). We finally add the dissipative flux (3.3) to CW. If we copy (4.3), but place BW on the implicit side, we obtain a *semi-implicit Runge–Kutta scheme* (SIRK):

ETNA
Kent State University
etna@mcs.kent.edu

$$
\begin{array}{rcl}
W^{(0)} & = & W^n \\
(I+\alpha_1\Delta t B)W^{(1)} & = & W^{(0)}+\alpha_1\Delta t(b-CW^{(0)}) \\
(I+\alpha_2\Delta t B)W^{(2)} & = & W^{(0)}+\alpha_2\Delta t(b-CW^{(1)}) \\
 & \vdots & \\
(I+\alpha_m\Delta t B)W^{(m)} & = & W^{(0)}+\alpha_m\Delta t(b-CW^{(m-1)}) \\
W^{n+1} & = & W^{(m)}.
\end{array} \tag{4.5}
$$

In each stage M block tri-diagonal systems of equations, each with $3N$ unknowns, have to be solved. It is also possible to define blends of (4.3) and (4.5), e.g. the following two-stage scheme

$$
\begin{array}{rcl}
W^{(0)} & = & W^n \\
(I+\alpha_1\Delta t B)W^{(1)} & = & W^{(0)}+\alpha_1\Delta t(b-CW^{(0)}) \\
W^{(2)} & = & W^{(0)}+\alpha_2\Delta t(b-(B+C)W^{(1)}) \\
W^{n+1} & = & W^{(2)}.
\end{array} \tag{4.6}
$$

which we call an *explicit semi-implicit Runge–Kutta scheme* (ESIRK).

The schemes (4.5) and (4.6) both belong to a class of methods that in [1] and [2] is called *additive Runge–Kutta schemes*. These schemes are not equivalent to the well-known alternating direction implicit (ADI) schemes, where an approximate factorization is performed and one system of equation is solved for each spatial direction. Instead we treat one direction, where the computational cells are very thin, implicitly and the rest explicitly. It is also clear that the schemes (4.5) and (4.6) are not equivalent to the line Gauss–Seidel method.

5. Convergence analysis.

5.1. Time step restrictions for a scalar model problem. Let us study a simple *scalar* PDE,

$$
\frac{\partial u}{\partial t}+c_1\frac{\partial u}{\partial x}+c_2\frac{\partial u}{\partial y}=a\frac{\partial^2 u}{\partial x^2}+b\frac{\partial^2 u}{\partial x\partial y}+c\frac{\partial^2 u}{\partial y^2}, \tag{5.1}
$$

where c_1, c_2, a, b and c are real constants. Discretizing (5.1) by centered differences, applying a Fourier transformation and separating *x*-, *xx*- and *xy*-derivatives from *y*- and *yy*-derivatives according to (4.4), we obtain the test equation

$$
\frac{d\hat{u}}{dt}=\lambda\hat{u}, \tag{5.2}
$$

with $\lambda=\lambda^{Ex}+\lambda^{Im}$ and

$$
\begin{array}{rcl}
\lambda^{Im} & = & -ic_2\dfrac{\sin\omega_y\Delta y}{\Delta y}-c\dfrac{4\sin^2\frac{\omega_y\Delta y}{2}}{\Delta y^2} \\
\lambda^{Ex} & = & -ic_1\dfrac{\sin\omega_x\Delta x}{\Delta x}-a\dfrac{4\sin^2\frac{\omega_x\Delta x}{2}}{\Delta x^2}-b\dfrac{\sin\omega_x\Delta x}{\Delta x}\cdot\dfrac{\sin\omega_y\Delta y}{\Delta y} \quad .
\end{array} \tag{5.3}
$$

Applying a Runge–Kutta scheme to (5.2), one time-step can be written

$$
\hat{u}^{n+1}=p(z_1,z_2)\hat{u}^n, \tag{5.4}
$$

where we have the two complex parameters $z_1 = \Delta t\lambda^{Ex}$ and $z_2 = \Delta t\lambda^{Im}$. For SIRK and ESIRK, $p(z_1,z_2)$ is a rational function, whereas for ERK it is a polynomial in $z=z_1+z_2$.

In [14] we gave stability restrictions for the Runge–Kutta schemes when applied to the test equation (5.2). For the explicit scheme this is straightforward; we numerically compute the domain in the complex plane where $|p(z_1, z_2)| \leq 1$. For the semi-implicit schemes, things are a little bit more complicated since we want our time step to be based solely on λ^{Ex} and not on λ^{Im}. In [14] we therefore derived a closed set, Ω, in the complex plane, so that the scheme is *A-stable* in z_2 if $z_1 \in \Omega$. In other words

$$z_1 \in \Omega \quad \text{and} \quad \Re(z_2) \leq 0 \quad \Rightarrow \quad |p(z_1, z_2)| \leq 1,$$

i.e., we have to choose Δt such that $z_1 = \Delta t \lambda^{Ex} \in \Omega$. Here we make the reasonable assumption that $c \geq 0$.

In Fig.5.1 we plot the stability domain Ω for the schemes (4.5) and (4.6), where we have $(\alpha_1, \alpha_2) = (1, 1)$. We also give the stability domain in the z-plane for the corresponding explicit scheme (4.3).

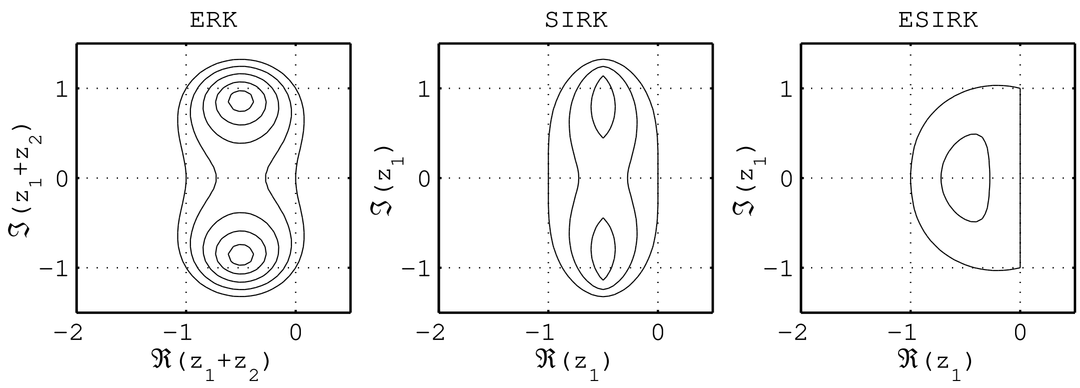

FIG. 5.1. *Stability regions for ERK, SIRK and ESIRK, when* $(\alpha_1, \alpha_2) = (1, 1)$

As in [11] time step restrictions are found by cutting out a rectangular area

$$\Omega = \{z : -\Omega_{\text{CFL}} \leq \Im(z) \leq \Omega_{\text{CFL}}, -\Omega_{\text{RK}} \leq \Re(z) \leq 0\}$$

of the stability domains in Fig.5.1 and choosing

$$\Delta t = \min\left(\Delta t_{\text{inviscid}}, \Delta t_{\text{viscous}}\right). \tag{5.5}$$

For SIRK and ESIRK we have

$$\begin{aligned} \Delta t_{\text{inviscid}} &\leq \frac{\Omega_{\text{CFL}}}{\max|\Im(\lambda^{Ex})|}, \qquad \max|\Im(\lambda^{Ex})| = \frac{|c_1|}{\Delta x}, \\ \Delta t_{\text{viscous}} &\leq \frac{\Omega_{\text{RK}}}{\max|\Re(\lambda^{Ex})|}, \qquad \max|\Re(\lambda^{Ex})| = \frac{4|a|}{\Delta x^2} + \frac{|b|}{\Delta x \Delta y} \end{aligned} \tag{5.6}$$

and for ERK

$$\begin{aligned} \Delta t_{\text{inviscid}} &\leq \frac{\Omega_{\text{CFL}}}{\max|\Im(\lambda)|}, \qquad \max|\Im(\lambda)| = \frac{|c_1|}{\Delta x} + \frac{|c_2|}{\Delta y}, \\ \Delta t_{\text{viscous}} &\leq \frac{\Omega_{\text{RK}}}{\max|\Re(\lambda)|}, \qquad \max|\Re(\lambda)| = \frac{4|a|}{\Delta x^2} + \frac{|b|}{\Delta x \Delta y} + \frac{4|c|}{\Delta y^2}. \end{aligned} \tag{5.7}$$

Note that the stability domain for SIRK is not attached to the imaginary axis. However, since we are solving a viscous problem this does not cause any instability problems.

5.2. Wave propagation for a scalar model problem. Consider the Runge–Kutta smoothers ERK, SIRK and ESIRK, defined by (4.3), (4.5) and (4.6). We set $(\alpha_1, \alpha_2) = (1, 1)$, so that the stability regions in Fig.5.1 can be used. For problems with dominant first derivatives, the convergence process of an iterative method of Runge–Kutta type is a combination of propagation of smooth error modes out of the computational domain and damping of oscillatory modes [3]. Let us analyze the schemes with respect to wave propagation and damping as proposed in [3]. As a model problem we take the hyperbolic part of (5.1),

$$\frac{\partial u}{\partial t} + c_1 \frac{\partial u}{\partial x} + c_2 \frac{\partial u}{\partial y} = 0, \tag{5.8}$$

on a two-dimensional grid with step sizes Δx and Δy, where $(x, y) = (\Delta x \mu, \Delta y \nu)$. We assume that $\Delta x \gg \Delta y$, which is typical for a boundary layer region. The two non-negative constants c_1 and c_2 are of the same order. The Fourier representation of the grid function $u_{\mu\nu}$ is

$$\begin{aligned} &u_{\mu\nu} = u(\Delta x\mu, \Delta y\nu) = \int_D \exp\Big(i(\omega_x \Delta x\mu + \omega_y \Delta y\nu)\Big)\hat{u}(\omega)d\omega, \\ &\omega = (\omega_x, \omega_y)^T, \quad D = [-\pi/\Delta x, \pi/\Delta x] \times [-\pi/\Delta y, \pi/\Delta y]. \end{aligned} \tag{5.9}$$

Discretization by centered difference approximations gives

$$\begin{aligned} &(Au)_{\mu\nu} = \int_D \hat{A} \exp\Big(i(\omega_x \Delta x\mu + \omega_y \Delta y\nu)\Big)\hat{u}(\omega)d\omega, \\ &\hat{A} = -i\left(c_1 \frac{\sin \omega_x \Delta x}{\Delta x} + c_2 \frac{\sin \omega_y \Delta y}{\Delta y}\right), \end{aligned} \tag{5.10}$$

where $\hat{A}$ is the Fourier transform of the difference operator A. One iteration with the smoothers can be written $\hat{u}^{n+1} = p(z_1, z_2)\hat{u}^n$, where

$$\begin{aligned} p(z_1, z_2) &= 1 + (z_1 + z_2) + (z_1 + z_2)^2 && \text{for ERK}, \\ p(z_1, z_2) &= ((1 - z_2)(1 - z_2))^{-1}(1 + z_1 + {z_1}^2 - z_2) && \text{for SIRK}, \\ p(z_1, z_2) &= (1 - z_2)^{-1}(1 + z_1 + {z_1}^2 + z_1 z_2) && \text{for ESIRK}. \end{aligned} \tag{5.11}$$

Let us split the grid function in one smooth and one oscillatory part

$$\begin{aligned} u^{n+1}_{\mu\nu} &= (u^{n+1}_{\mu\nu})_S + (u^{n+1}_{\mu\nu})_O \\ &= \int_{D_0} \exp\Big(\ln p(z_1, z_2) + i(\omega_x \Delta x\mu + \omega_y \Delta y\nu)\Big)\hat{u}^n(\omega)d\omega + \\ &\quad \int_{D_1} p(z_1, z_2) \exp\Big(i(\omega_x \Delta x\mu + \omega_y \Delta y\nu)\Big)\hat{u}^n(\omega)d\omega. \end{aligned} \tag{5.12}$$

Correspondingly the wave number domain D is divided in two parts D_0 and D_1. The oscillatory part $(u^{n+1}_{\mu\nu})_O$ is damped out if $|p(z_1, z_2)| < 1$ for $\omega \in D_1$.

We now study the smooth part. For small wave numbers, i.e. $\omega \in D_0$, we expand the logarithm of $p(z_1, z_2)$ in a Taylor series, which for all three schemes gives

$$
\begin{aligned}
\ln p(z_1,z_2) &= z_1+z_2+0.5(z_1+z_2)^2+\mathcal{O}(h^3)\\
&= -i\Delta t\left(c_1\frac{\sin\omega_x\Delta x}{\Delta x}+c_2\frac{\sin\omega_y\Delta y}{\Delta y}\right)-\\
&\qquad 0.5\Delta t^2\left(c_1\frac{\sin\omega_x\Delta x}{\Delta x}+c_2\frac{\sin\omega_y\Delta y}{\Delta y}\right)^2+\mathcal{O}(h^3)\\
&= -i\Delta t(c_1\omega_x+c_2\omega_y)-0.5\Delta t^2(c_1\omega_x+c_2\omega_y)^2+\mathcal{O}(h^3),
\end{aligned}
\tag{5.13}
$$

where $w\in D_0$ and $h=\max(\Delta x,\Delta y)$. Exploiting (5.13) the smooth part can be written

$$
\begin{aligned}
&\exp\Big(\ln p(z_1,z_2)+i(\omega_x\Delta x\mu+\omega_y\Delta y\nu)\Big)\\
&= \exp\left(i\left(\left(-c_1\frac{\Delta t}{\Delta x}+\mu\right)\omega_x\Delta x+\left(-c_2\frac{\Delta t}{\Delta y}+\nu\right)\omega_y\Delta y\right)-\right.\\
&\qquad \left.0.5\Delta t^2\left(c_1\omega_x+c_2\omega_y\right)^2+\mathcal{O}(h^3)\right).
\end{aligned}
\tag{5.14}
$$

The time step Δt is restricted by the stability conditions (5.5)–(5.7), where we choose $\Omega_{\rm CFL}=0.7$ for SIRK and ESIRK and $\Omega_{\rm CFL}=1.0$ for ERK. Inserting the maximum time steps into (5.14) and assuming small enough space steps yields

$$
\begin{aligned}
&\exp\Big(\ln p(z_1,z_2)+i(\omega_x\Delta x\mu+\omega_y\Delta y\nu)\Big)\\
&\approx \exp\left(i\left(\left(-\frac{c_1\Delta y}{c_2\Delta x}\cdot\Omega_{\rm CFL}+\mu\right)\omega_x\Delta x+\left(-\Omega_{\rm CFL}+\nu\right)\omega_y\Delta y\right)-\right.\\
&\qquad \left.0.5\cdot\Omega_{\rm CFL}^2\cdot\Delta y^2\left(\frac{c_1}{c_2}\omega_x+\omega_y\right)^2\right),
\end{aligned}
\tag{5.15}
$$

for ERK and correspondingly

$$
\begin{aligned}
&\exp\Big(\ln p(z_1,z_2)+i(\omega_x\Delta x\mu+\omega_y\Delta y\nu)\Big)\\
&\approx \exp\left(i\left(\left(-\Omega_{\rm CFL}+\mu\right)\omega_x\Delta x+\left(-\frac{c_2\Delta x}{c_1\Delta y}\cdot\Omega_{\rm CFL}+\nu\right)\omega_y\Delta y\right)-\right.\\
&\qquad \left.0.5\cdot\Omega_{\rm CFL}^2\cdot\Delta x^2\left(\omega_x+\frac{c_2}{c_1}\omega_y\right)^2\right),
\end{aligned}
\tag{5.16}
$$

for SIRK and ESIRK. Inserting (5.15) and (5.16) in (5.12) we obtain

$$
\left(u^{n+1}_{\mu\nu}\right)_S\approx\left(u^n_{\mu-\Omega_{\rm CFL}\cdot(c_1\Delta y)/(c_2\Delta x),\nu-\Omega_{\rm CFL}}\right)_S,
\tag{5.17}
$$

for ERK and

$$
\left(u^{n+1}_{\mu\nu}\right)_S\approx\left(u^n_{\mu-\Omega_{\rm CFL},\nu-\Omega_{\rm CFL}\cdot(c_2\Delta x)/(c_1\Delta y)}\right)_S,
\tag{5.18}
$$

ETNA
Kent State University
etna@mcs.kent.edu

for SIRK and ESIRK, i.e., in one iteration smooth waves are transported a distance

$$(5.19)\qquad \begin{array}{rclll} (x_{\mathrm{dist}}, y_{\mathrm{dist}}) &=& \Omega_{\mathrm{CFL}} \cdot \Delta y \cdot (c_1/c_2, 1) & \text{by} & \text{ERK}, \\ (x_{\mathrm{dist}}, y_{\mathrm{dist}}) &=& \Omega_{\mathrm{CFL}} \cdot \Delta x \cdot (1, c_2/c_1) & \text{by} & \text{SIRK and ESIRK}. \end{array}$$

Since $\Delta x \gg \Delta y$ smooth waves are transported faster out of the computational domain by the semi-implicit schemes.

Taylor expansion of the functions in (5.11) for small z_1 and z_2 yields

$$(5.20)\qquad \begin{array}{rclll} |p(z_1, z_2)| &=& 1 - \mathcal{O}(\Delta y^2) & \text{for} & \text{ERK}, \\ |p(z_1, z_2)| &=& 1 - \mathcal{O}(\Delta x^2) & \text{for} & \text{SIRK and ESIRK}, \end{array}$$

i.e., smooth waves are better damped by the semi-implicit schemes.

5.3. The linearized Navier–Stokes equations. We here derive time-step restrictions for the linearized Navier–Stokes equations in the same way as we did for the scalar model problem in Section 5.1. Discretizing (2.6) by centered differences and applying the Fourier transformation we obtain an ODE,

$$(5.21)\qquad \frac{d\hat{W}_\omega}{dt} + A_\omega \hat{W}_\omega = b,$$

where $A_\omega = B_\omega + C_\omega$ and

$$(5.22)\qquad \begin{array}{rcl} B_\omega &=& iA_2 \dfrac{\sin \omega_y \Delta y}{\Delta y} + B_2 \dfrac{4 \sin^2 \frac{\omega_y \Delta y}{2}}{\Delta y^2} \\ C_\omega &=& iA_1 \dfrac{\sin \omega_x \Delta x}{\Delta x} + B_1 \dfrac{4 \sin^2 \frac{\omega_x \Delta x}{2}}{\Delta x^2} + B_3 \dfrac{\sin \omega_x \Delta x}{\Delta x} \cdot \dfrac{\sin \omega_y \Delta y}{\Delta y} \quad ; \end{array}$$

cf. (4.4). We proceed by splitting the matrices into one inviscid and one viscous part, e.g., $A_\omega = A_\omega^{\mathrm{inviscid}} + A_\omega^{\mathrm{viscous}}$. Thereby the time step for ERK can be chosen according to (5.5), where

$$(5.23)\qquad \begin{array}{rcl} \Delta t_{\mathrm{inviscid}} &\leq& \dfrac{\Omega_{\mathrm{CFL}}}{\max \lambda(A_\omega^{\mathrm{inviscid}})} \\ \Delta t_{\mathrm{viscous}} &\leq& \dfrac{\Omega_{\mathrm{RK}}}{\max \lambda(A_\omega^{\mathrm{viscous}})} \end{array}$$

are based on the following estimates of the spectral radii

$$(5.24)\qquad \begin{array}{rcl} \max \lambda(A_\omega^{\mathrm{inviscid}}) &=& \dfrac{|U|}{\Delta x} + \dfrac{|V|}{\Delta y} + c \left(\dfrac{1}{\Delta x^2} + \dfrac{1}{\Delta y^2} \right)^{1/2} \\ \max \lambda(A_\omega^{\mathrm{viscous}}) &=& \dfrac{4}{3} \nu \left(\dfrac{4}{\Delta x^2} + \dfrac{4}{\Delta y^2} \right). \end{array}$$

Correspondingly we have for semi-implicit schemes

$$(5.25)\qquad \begin{array}{rcl} \Delta t_{\mathrm{inviscid}} &\leq& \dfrac{\Omega_{\mathrm{CFL}}}{\max \lambda(C_\omega^{\mathrm{inviscid}})} \\ \Delta t_{\mathrm{viscous}} &\leq& \dfrac{\Omega_{\mathrm{RK}}}{\max \lambda(C_\omega^{\mathrm{viscous}})} \quad , \end{array}$$

and

$$\text{(5.26)}\qquad \begin{aligned} \max\lambda(C_\omega^{\text{inviscid}}) &= \frac{|U|}{\Delta x} + \frac{c}{\Delta x} \\ \max\lambda(C_\omega^{\text{viscous}}) &= \frac{4}{3}\nu\left(\frac{4}{\Delta x^2} + \frac{1}{4\Delta x\Delta y}\right). \end{aligned}$$

In [13] the stability restrictions (5.23) and (5.25) were studied for the flow over a flat plate. Assuming a reasonable refinement in the boundary layer region, the inviscid condition proved to be the more restrictive. We also assume that the grid is refined so that the truncation error is of the same order for all Re. Since the boundary layer thickness is $\mathcal{O}(Re^{-1/2})$ according to (2.5), we have a typical cell size $\Delta x = \mathcal{O}(1)$ and $\Delta y = \mathcal{O}(Re^{-1/2})$ in the boundary layer. Relation (5.19) thus says that smooth waves are transported a distance $\mathcal{O}(Re^{-1/2})$ in one ERK iteration and correspondingly $\mathcal{O}(1)$ in one SIRK or ESIRK iteration. The number of iterations before smooth error modes are outside the computational domain is therefore

$$\text{(5.27)}\qquad \begin{aligned} \#\text{iter} &= \mathcal{O}(Re^{1/2}) && \text{for ERK,} \\ \#\text{iter} &= \mathcal{O}(1) && \text{for SIRK and ESIRK,} \end{aligned}$$

indicating Re-independent convergence for the semi-implicit schemes.

6. Multigrid. Since only a linear problem is considered, we use the following linear formulation of the multigrid method:

$$\text{(6.1)}\qquad \begin{array}{l} \textbf{procedure}\ MG(l,u,f); \\ \textbf{if}\ (l=0)\ \textbf{then}\ u = \mathcal{S}^{(\nu_1+\nu_2)}(u,f); \\ \textbf{else begin} \\ \quad u = \mathcal{S}^{(\nu_1)}(u,f); \\ \quad d = r*(L_l u - f); \\ \quad v = 0; \\ \quad \textbf{for}\ i = 1(1)\gamma\ \textbf{do}\ MG(l-1,v,d); \\ \quad u = u - p*v; \\ \quad u = \mathcal{S}^{(\nu_2)}(u,f); \\ \textbf{end}; \end{array}$$

given in [4] for solving $L_l u_l = f_l$, cf. the linear system (4.2). As smoother, $\mathcal{S}$, we use the Runge–Kutta schemes (4.3), (4.5) and (4.6). The type of cycle is determined by the parameter γ and the number of pre- and post smoothing steps are given by ν_1 and ν_2. On the coarsest level we have no exact solver, but instead $\nu_1+\nu_2$ smoothing iterations. In all experiments below we have $\nu_1 = \nu_2 = 1$.

To transfer grid functions between two levels we use a prolongation operator, p, and a restriction operator, r, which are both based on second order interpolation. On the finest grid level there are $M\times N$ cells. If there are $M/2\times N/2$ cells on the second finest level etc. we have *full coarsening*. Another possibility is to use *semi-coarsening*, as suggested in [10], where the grid is refined in one direction at a time.

7. Numerical experiments. Here we solve the linearized Navier–Stokes equations (2.6) on the unit square, where there is a flat plate at $y=0$ and the other boundaries are open. The iterations are terminated when the norm

$$\text{(7.1)}\qquad \|R^n\|^2 = \sum_{i,j}\Delta x_i\Delta y_j\left(\frac{W_{i,j}^{n+1} - W_{i,j}^n}{\Delta t_{i,j}}\right)^2$$

ETNA
Kent State University
etna@mcs.kent.edu

of the initial residual has been decreased a factor 10^5. As an initial guess we set $u = U_\infty = 0.1, v = 0$ and $\rho = \rho_\infty = 1$. The dissipation parameter is here $\varepsilon = 0.02$.

Let us first study a flow where $Re = 100$ on a uniform grid with $M = N$ and perform a grid refinement study for the multigrid method (6.1) with full coarsening and $\gamma = 2$, which gives a W-cycle. The smoother is the three-stage ERK (4.3) with $(\alpha_1, \alpha_2, \alpha_3) = (0.6, 0.6, 1)$, $\Omega_{\mathrm{CFL}} = 1.5$ and $\Omega_{\mathrm{RK}} = 0.8$. This is a more efficient smoother than the two-stage ERK in Fig.5.1. In Table 7.1 we see that more or less grid independent convergence is achieved by increasing the number of grid levels as the grid is refined, which is in agreement with theory in [8]. But introducing more grid levels is not always possible, especially in more complex flow simulations where one can seldom define more than three or four grid levels.

TABLE 7.1
Grid refinement study for full coarsening MG.

N	16	32	64	128	256
# grid levels	2	3	4	5	6
# cycles	73	51	48	49	67

We continue with an experiment where Re is varied by changing ν in (2.3). Here the grid is designed to give sufficient accuracy for all Re rather than impressive convergence rates for the multigrid method. The number of cells in the body-normal direction, N, is increased with Re according to Table 7.2 so that there are approximately 25 cells in the boundary layer at $x = 0.6$. The other parameters, $M = 64$ and $\theta = 0.05$ are fixed, i.e., the stretching parameter in (3.1) is given by $\alpha = N \cdot \ln(1 + \theta)$.

TABLE 7.2
Number of cells for various Reynolds numbers.

N	56	80	104	128
Re	10^3	10^4	10^5	10^6

The multigrid method (6.1) is used for 1, 2 and 3 grid levels. Full coarsening is applied and we have $\gamma = 1$ yielding a V-cycle. In Fig.7.1 we see that the number of cycles grows approximately as $Re^{1/2}$ when ERK is the smoother, i.e., the same kind of behavior as in the single grid case, cf. (5.27). But with the ESIRK scheme (4.6) as smoother, the result is more or less Re-independent convergence.

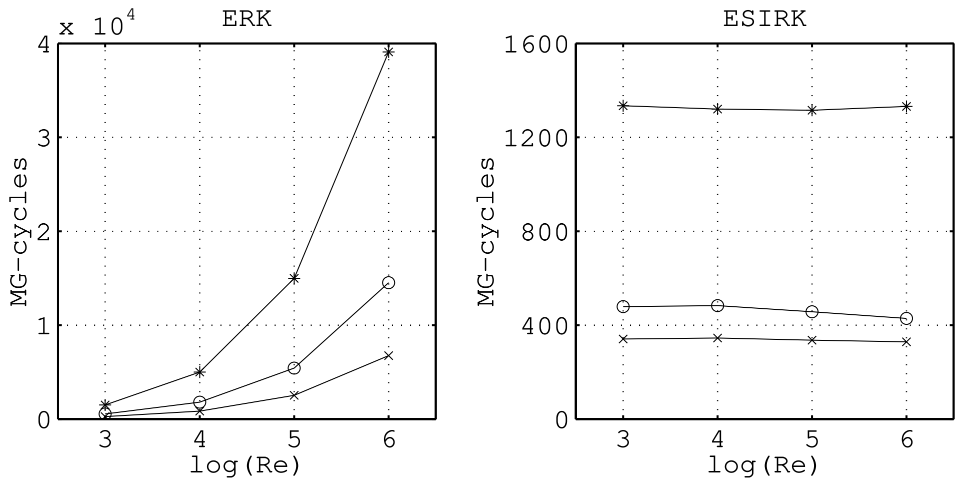

FIG. 7.1. *Number of full coarsening MG cycles for 1 ($*$), 2 (o), and 3 ($\times$) grid levels*

We now change to semi-coarsening and repeat the experiment above. Coarsening is applied only in the body normal direction. Since we thereby improve the aspect ratio of cells in the boundary layer on coarser levels, the stiffness is hopefully alleviated and the convergence rate is increased for high *Re*. However, in Fig.7.2 we see that the number of iterations is almost the same as for full coarsening when ERK is the smoother, and with ESIRK as smoother the number of iterations is actually increased from the full coarsening experiment. Moreover, the Re-dependency seems to be very similar no matter what coarsening strategy is used, i.e., the key factor in order to obtain *Re*-independent convergence proves to be the choice of smoother.

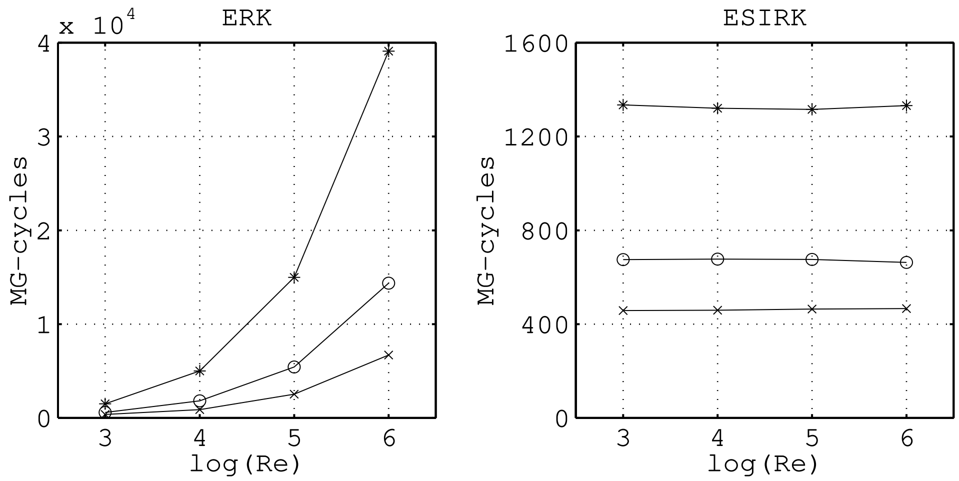

FIG. 7.2. *Number of semi-coarsening MG cycles for 1 ($*$), 2 (o), and 3 ($\times$) grid levels*

We let our f90-code run on a Digital AlphaServer 8200 with a EV5/300 MHz processor. In Fig.7.3 we plot the CPU-time in seconds for the MG algorithm with three grid levels, both for full coarsening and semi-coarsening. Since we are solving a linear problem, the CPU-time may be decreased by LU-factorizing the systems in (4.6) once and in each iteration just back substitute. However, this has been avoided since it would give ESIRK an unfair advantage over ERK. After all, our interest is primarily in *non linear* problem. Of course ESIRK requires more operations than ERK, for instance in the experiment where $Re = 10^6$ one MG cycle with ESIRK is roughly 40 % more expensive. Nevertheless, since the number of iterations is drastically decreased we gain as much as one order of magnitude in CPU-time by switching from ERK to ESIRK. Finally note that semi-coarsening gives rise to more work on coarser grid levels than full coarsening, and therefore it proves to be slower than full coarsening in all our experiments.

8. Concluding remarks. In [13] we solved the Navier–Stokes equations (2.1) and the linearized equations (2.6) on a single grid. In both cases the convergence rate was substantially improved by switching from an explicit to a semi-implicit Runge–Kutta scheme. Here we have solved the linearized equations and obtained similar results for the multigrid method. Thus, there are strong reasons to believe that the semi-implicit scheme also works well as a smoother in a non linear multigrid method. In [15] it has been shown that this is indeed the case.

REFERENCES

[1] G. J. COOPER AND A. SAYFY, *Additive methods for the numerical solution of ordinary differential equations*, Math. Comp., 35 (1980), pp. 1159–1172.
[2] ———, *Additive Runge–Kutta methods for stiff ordinary differential equations*, Math. Comp., 40 (1983), pp. 207–218.
[3] B. GUSTAFSSON AND P. LÖTSTEDT, *The GMRES method improved by securing fast wave propagation*, Appl. Num. Math., 12 (1993), pp. 135–152.
[4] W. HACKBUSCH, *Multi-Grid Methods and Applications*, Springer-Verlag, Berlin, 1985.
[5] C. HIRSCH, *Numerical Computation of Internal and External Flows, Volume 2*, John Wiley & Sons Ltd., New York, 1990.
[6] A. JAMESON, *Computational transonics*, Comm. Pure Appl. Math., XLI (1988), pp. 507–549.
[7] H.-O. KREISS AND J. LORENZ, *Initial-Boundary Value Problems and the Navier–Stokes Equations*, Academic Press, San Diego, 1989.

ETNA
Kent State University
etna@mcs.kent.edu

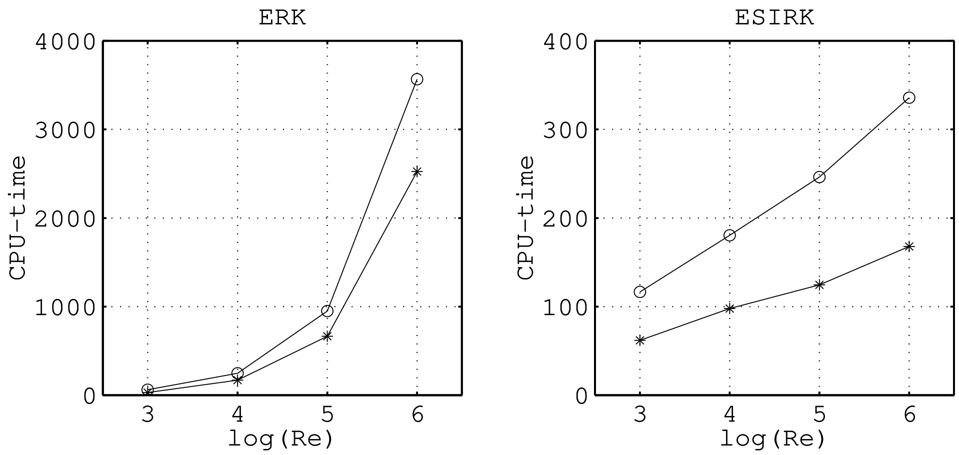

FIG. 7.3. *CPU-times for 3 level MG, full coarsening (∗) and semi-coarsening (o)*

[8] P. LÖTSTEDT, *Grid independent convergence of the multigrid method for first-order equations*, SIAM J. Numer. Anal., 29 (1992), pp. 1370–1394.
[9] L. MARTINELLI, *Calculations of viscous flow with a multigrid method*, PhD thesis, Dept. of Mechanical and Aerospace Engineering, Princeton University, Princeton, New Jersey, 1987.
[10] W. A. MULDER, *A new multigrid approach to convection problems*, J. Comput. Phys., 83 (1989), pp. 303–323.
[11] B. MÜLLER, *Linear stability condition for explicit Runge–Kutta methods to solve the compressible Navier–Stokes equations*, Math. Methods Appl. Sci., 12 (1992), pp. 139–151.
[12] H. SCHLICHTING, *Boundary-Layer Theory*, McGraw–Hill, New York, 7th ed., 1979.
[13] E. STERNER, *Semi-implicit Runge–Kutta schemes for the Navier–Stokes equations*, Tech. Rep. No. 166, Dept. of Scientific Computing, Uppsala University, Uppsala, Sweden, 1995.
[14] ———, *Semi-implicit Runge–Kutta schemes for the Navier–Stokes equations*, BIT, 37 (1997), pp. 164–178.
[15] P. WEINERFELT, *Convergence acceleration of the steady 2D Navier–Stokes equations by a semi-implicit Runge–Kutta method*, tech. rep., SAAB-SCANIA, Linköping, Sweden, to appear.