NASA/CR-1999-209831
ICASE Report No. 99-51

Textbook Multigrid Efficiency for the Incompressible Navier-Stokes Equations: High Reynolds Number Wakes and Boundary Layers

James L. Thomas
NASA Langley Research Center, Hampton, Virginia

Boris Diskin
ICASE, Hampton, Virginia

Achi Brandt
The Weizmann Institute of Science, Rehovot, Israel

Institute for Computer Applications in Science and Engineering
NASA Langley Research Center
Hampton, VA
Operated by Universities Space Research Association

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-2199

Prepared for Langley Research Center
under Contract NAS1-97046

December 1999

TEXTBOOK MULTIGRID EFFICIENCY FOR THE INCOMPRESSIBLE NAVIER-STOKES EQUATIONS: HIGH REYNOLDS NUMBER WAKES AND BOUNDARY LAYERS

JAMES L. THOMAS*, BORIS DISKIN†, AND ACHI BRANDT‡

Abstract. Textbook multigrid efficiencies for high Reynolds number simulations based on the incompressible Navier-Stokes equations are attained for a model problem of flow past a finite flat plate. Elements of the Full Approximation Scheme multigrid algorithm, including distributed relaxation, defect correction, and boundary treatment, are presented for the three main physical aspects encountered: entering flow, wake flow, and boundary layer flow. Textbook efficiencies, i.e., reduction of algebraic errors below discretization errors in one full multigrid cycle, are attained for second order accurate simulations at a laminar Reynolds number of 10,000.

Key words. incompressible Navier-Stokes equations, textbook multigrid efficiency, distributive relaxation, defect-correction iteration

Subject classification. Applied and Numerical Mathematics

1. Introduction. In the mid-70's, Beam and Warming [1] presented an implicit scheme for the compressible Navier-Stokes equations which had a significant impact on the field known as Computational Fluid Dynamics (CFD). The method they presented, based upon a spatial factoring of the implicit equations in delta form, used alternating tridiagonal line relaxations to solve high Reynolds number viscous simulations. This method proved to be much more efficient than other approaches. The basic methodology is still widely used and has been extended to very general applications across the Mach number range, forming the foundation for many general purpose solvers worldwide, among them ARC3D [8] and CFL3D [7] at the NASA Ames and Langley Research Centers, respectively. This seminal contribution of Beam and Warming was a critical building block to the acceptance of CFD using Reynolds-Averaged Navier-Stokes (RANS) solvers by the aircraft industry. Today, computational methods for the cruise shapes of transport aircraft, designed to minimize viscous and shock wave losses at transonic speeds, are reasonably well in hand. Simulations of off-design performance, involving unsteady separated and vortical flows with stronger shock waves, require significantly greater computing resources; this requirement limits further inroads into the design process with CFD.

As a typical example of current RANS capability, the CFL3D code is based on the spatially-factored scheme of Beam and Warming and uses multigrid to accelerate convergence to steady state; using alternating-line implicit block 5x5 matrix solutions, approximately 200 updates are required to converge the lift and drag to one percent of their final values for wing-body geometries near transonic cruise conditions. Complex geometry and complex physics simulations generally require many more residual evaluations to converge, and sometimes convergence cannot be attained. Now, it is well-known for fully elliptic problems that solutions

*Computational Modeling and Simulation Branch (CMSB), Mail Stop 128, NASA Langley Research Center, Hampton, Virginia 23681 (email: j.l.thomas@larc.nasa.gov).

†Institute for Computer Applications in Science and Engineering, Mail Stop 132C, NASA Langley Research Center, Hampton, Virginia 23681 (email: bdiskin@icase.edu). This research was supported by the National Aeronautics and Space Administration under NASA Contract No. NAS1-97046 while the author was in residence at the Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, Hampton, Virginia 23681-2199.

‡The Weizmann Institute of Science, Rehovot 76100, Israel (email: achi@wisdom.weizmann.ac.il).

can be attained using a full multigrid (FMG) process in far fewer, on the order of 2-4, residual evaluations. Optimal convergence is defined by Brandt [2, 3, 4] as textbook multigrid efficiency (TME), meaning the solutions to the governing system of equations are attained in a computational work which is a small (less than 10) multiple of the operation count in the discretized system of equations. Thus, there is a potential gain of more than two orders of magnitude in operation count reduction if TME could be attained for the RANS equation sets. The principal difficulty stems from the fact that the RANS equation sets are a system of coupled nonlinear equations which are not, even for subsonic Mach numbers, fully elliptic, but contain hyperbolic factors. Brandt [4] has summarized the progress and remaining barriers to achieving TME for the equations of fluid dynamics.

The purpose of this paper is to present a multigrid method which attains textbook efficiencies for one of the most basic simulations encountered in fluid dynamics – the incompressible viscous flow past a finite flat plate at high Reynolds number. The flow, although relatively simple, contains several basic elements of the barriers to be overcome in extending textbook efficiencies to the compressible RANS equations, namely entering flows, far wake flows, and boundary layers. A central element of the multigrid method presented is the decomposition through distributed relaxation [3] of the the system of equations into separate, usually scalar, factors that can be treated optimally, i.e., through marching for the hyperbolic factors and through multigrid for the elliptic factors. Although we restrict ourselves to incompressible flow, the procedures carry over directly to the compressible flow case, at least for subcritical flow [3, 4, 10].

2. Governing Equations. The equations considered here are the steady, incompressible Navier-Stokes equations in nonconservative form, i.e., two momentum equations and the continuity equation,

$$\mathbf{r}(\mathbf{q}) \equiv \mathcal{L}\mathbf{q} = 0, \tag{2.1}$$

expressed in terms of primitive (velocities and pressure) variables $\mathbf{q} = (u, v, p)^T$, where

$$\mathcal{L} = \begin{bmatrix} Q_\nu & 0 & \partial_x \\ 0 & Q_\nu & \partial_y \\ \partial_x & \partial_y & 0 \end{bmatrix}. \tag{2.2}$$

The operator Q_ν represents convection and diffusion effects as

$$Q_\nu \equiv Q - \nu\Delta, \tag{2.3}$$

where $Q = u\partial_x + v\partial_y$, the Laplacian operator is $\Delta = \partial_{xx} + \partial_{yy}$, and the kinematic viscosity is $\nu = 1/Re$, where Re is Reynolds number. Extensions to conservation law form for the momentum equations and to inclusion of the energy equation are possible, but not considered here.

The determinant of the matrix of operators,

$$|\mathcal{L}| = -Q_\nu\,\Delta, \tag{2.4}$$

corresponds to an elliptic factor, represented by the Laplacian, and a convection-diffusion factor, generally recognized as the convection and diffusion of vorticity along a streamline. For high Reynolds number simulations, there are two important scales: the viscous scales in the thin viscous layers near bodies and

in their wakes and the inviscid scales, which predominate over most of the flow field. For the numerical calculations below, the thin-layer approximation, in which only the viscous terms associated with variations in the coordinate normal to the body are retained, is used.

3. Multigrid Method. The present approach uses a full multigrid (FMG) algorithm [2, 3], proceeding from the coarsest grid to finer grids. The solution is interpolated from the current grid to the next finer grid. The goal of the algorithm is fast reduction of the algebraic errors below the discretization errors on a given grid, before moving to the next finer grid. The algebraic errors of the discrete equations on a given grid are reduced through a Full Approximation Scheme (FAS) [3] multigrid scheme, in which corrections to the nonlinear equations are obtained from coarser grid solutions. The scheme is described below by means of a two-grid notation, in which the fine grid is denoted by superscript h and the coarse grid by superscript $2h$.

The steady-state residual operator to be solved on the fine grid is the discrete version of Eq. (2.1),

$$\mathbf{r}^h(\mathbf{q}^h) = 0. \tag{3.1}$$

The initial fine-grid approximation $\mathbf{q}^h$ is prolonged from the coarse-grid solution $\mathbf{q}^{2h}$, as

$$\mathbf{q}^h \leftarrow \mathcal{P}\,\mathbf{q}^{2h}. \tag{3.2}$$

where $\mathcal{P}$ denotes a prolongation operator. After relaxation(s) of the fine-grid operator to obtain an approximation $\tilde{\mathbf{q}}^h$, the coarse-grid equation at level $2h$ to be solved for a correction to the fine grid is

$$\mathbf{r}^{2h}(\mathbf{q}^{2h}) = \mathbf{r}^{2h}(\mathcal{R}\tilde{\mathbf{q}}^h) - \mathcal{R}\mathbf{r}^h(\tilde{\mathbf{q}}^h), \tag{3.3}$$

where $\mathcal{R}$ denotes a restriction operator for transfer of information to the coarser grid and the tilde superscript denotes a most recently available value. This coarse grid equation is then solved by some iterative method (or directly if the grid is coarse enough). The correction from the coarse grid (grid $2h$) is prolonged to the finer grid as

$$\tilde{\mathbf{q}}^h \leftarrow \tilde{\mathbf{q}}^h + \mathcal{P}(\mathbf{q}^{2h} - \mathcal{R}\tilde{\mathbf{q}}^h). \tag{3.4}$$

The restrictions $\mathcal{R}$ used here are volume-weighted for the continuity equations; for the momentum equations, the coarser cell values are found by volume-weighted restrictions in the direction parallel to the cell interface along with full-weighted restrictions in the orthogonal direction, i.e., for the $y-$momentum equation, volume-weighted horizontally and full-weighted vertically. The prolongations $\mathcal{P}$ are bicubic interpolations from coarser meshes for both the solution and the correction , although results with linear interpolations were nearly identical. The FAS cycle described above is used extensively in current Euler and Navier-Stokes solvers. The algorithm is critically dependent on the choice of relaxation operator; distributed relaxation is used here as described subsequently.

The coarse-grid equations are themselves solved with γ cycles of the algorithm applied recursively, where $\gamma = 1$ would correspond to a V-cycle and $\gamma = 2$ to a W-cycle; the number of relaxations on the downward and upward legs of the cycle are denoted as (ν_1, ν_2). We use here $(\nu_1, \nu_2) = (2, 1)$ and a variant of the

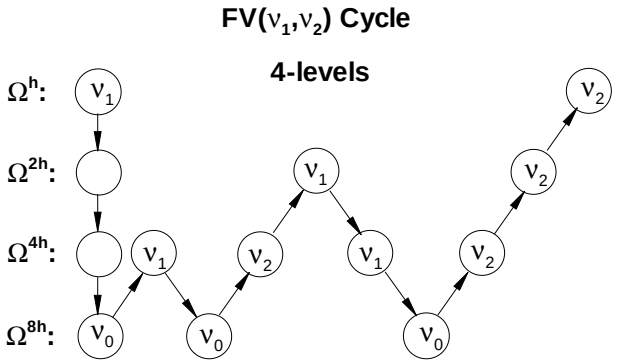


FIG. 3.1. *Schematic of the FV-cycle for 4-level multigrid where ν_0 denotes the number of relaxations on the coarsest mesh (Ω^{8h}).*

V-cycle, termed an FV-cycle, in which the initial approximation to the correction on the $2h$ grid is obtained through a FMG process. The cycle is sketched in Fig. 3.1; the amount of additional computational work compared to a standard V-cycle is small, in the ratio of $8/7$ in the limit of an infinite number of levels in two dimensions. For the simulations here, six levels were used wherever possible. The notation FMG-n denotes an FMG cycle with n FV(2,1) cycles at each level.

4. Distributed Relaxation. Away from boundaries, the correction δq to the current approximation q, introduced at the stage of distributed relaxation, [2, 3] is calculated from

$$\mathbf{L}\,\delta\mathbf{q} \;=\; -\mathbf{r}(\mathbf{q}), \tag{4.1}$$

where $\mathbf{L}$ is a principal linearization of $\mathcal{L}$, in which the coefficients u and v in Eq. (2.3) are evaluated from the current approximation and fixed throughout the relaxation. Note this is not a Newton linearization; only the principal terms at the viscous and inviscid scales are retained. The distributed relaxation method replaces $\delta\mathbf{q}$ by $\mathbf{M}\delta\mathbf{w}$ so that the resulting matrix $\mathbf{LM}$ becomes a diagonal or lower triangular matrix, as

$$\mathbf{L}\,\mathbf{M}\,\delta\mathbf{w} \;= -\mathbf{r}(\mathbf{q}). \tag{4.2}$$

The diagonal elements of $\mathbf{LM}$ are composed ideally of the separate factors of the determinant of the matrix $\mathbf{L}$ and represent the elliptic or hyperbolic features of the equation. For incompressible flow, the distribution matrix $\mathbf{M}$ can take on a particularly simple form, as determined by the cofactors of the third row of $\mathbf{L}$ divided by their common factor, as

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & -\partial_x \\ 0 & 1 & -\partial_y \\ 0 & 0 & Q_\nu \end{bmatrix}, \tag{4.3}$$

yielding

$$\mathbf{LM} = \begin{bmatrix} Q_\nu & 0 & 0 \\ 0 & Q_\nu & 0 \\ \partial_x & \partial_x & -\Delta \end{bmatrix}. \tag{4.4}$$

The determinant of the operator matrix **LM**,

$$|\mathbf{LM}| = -Q_\nu^2\,\Delta, \tag{4.5}$$

corresponds to an elliptic factor and two convection-diffusion factors; the additional term over Eq. (2.4), Q_ν, indicates that as a set of new variables, $\delta\mathbf{w}$ would generally need additional boundary conditions all around the boundary (or, just at inflow in the case $\nu = 0$). Brandt termed the variables $\delta\mathbf{w}$ as "ghost variables," since they need not explicitly appear in the calculations; here, they do appear in the calculations, although, as with the original intent, the boundary conditions are derived from the original primitive variables. The equations to solve for the ghost variables are given explicitly as

$$\begin{aligned} Q_\nu \delta w_1 &= -r_1, \\ Q_\nu \delta w_2 &= -r_2, \\ \Delta\,\delta w_3 &= +r_3 + \partial_x \delta w_1 + \partial_y \delta w_2. \end{aligned} \tag{4.6}$$

Near boundaries, the general approach, [3, 4] would be to relax the governing equations directly, since the equations do not necessarily decouple near boundaries as they do in the interior of the domain. One can make more general, but possibly slowly converging, relaxations, such as Kaczmarcz relaxation, in this region. This will not affect the overall complexity, because the number of boundary points is negligible in comparison to the number of interior points. Here, however, we use an approach which applies the interior distributive relaxation operator also at the boundaries, inferring boundary conditions for the ghost variables based on the boundary conditions of the governing equations. The cost is that the correction equations, Eq. (4.6), no longer assume a triangular form, requiring a block matrix solution at the boundaries rather than the scalar solutions attained away from the boundary. Assuming linearized flow, the appropriate ghost variable boundary conditions at the differential level are derived for inviscid inflow and outflow in Appendix I and tangency in Appendix II . These boundary conditions are implemented discretely at the corresponding boundaries. At the no-slip boundary, the corresponding discrete boundary conditions for the ghost variables are constructed in Appendix III. The procedure is effective for the simulations considered here; details are given in subsequent sections.

5. Defect Correction Relaxation. Since Eq. (4.2) is written in delta form, it is natural to consider defect correction for the update, namely a lower-order discretization of the left side of Eq. (4.6) in order to simplify the construction and reduce the bandwidth of the implicit operator. Here, we use a first-order upwind discretization for the convective part of the convection-diffusion operator, Q_ν, in Eq. (4.6). The distributed relaxation operator can thus be written as

$$[\mathbf{L}\,\mathbf{M}]_d\,\delta\mathbf{w} = -\mathbf{r}_t, \tag{5.1}$$

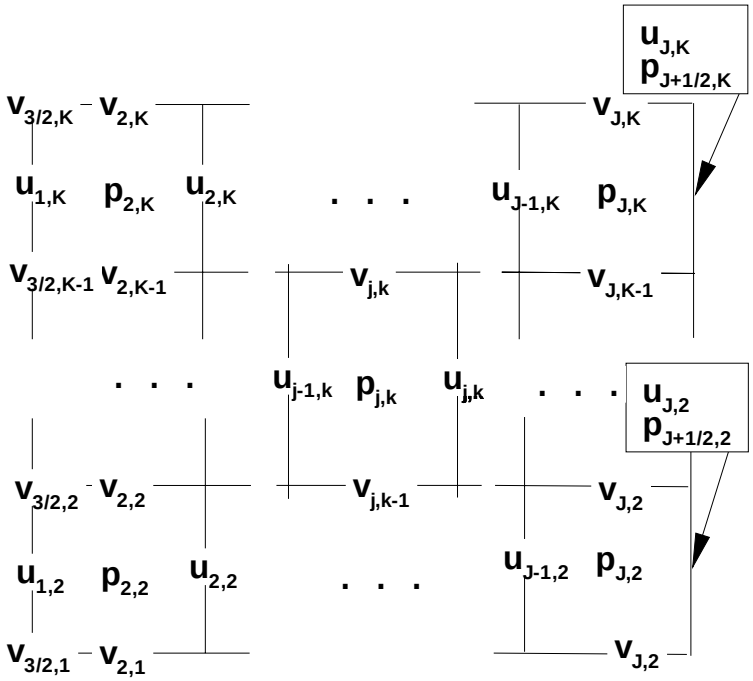


FIG. 6.1. *Variable description for a grid of* $J\text{x}K \equiv \text{N}_\text{x}\text{x}\text{N}_\text{y}$ *points.*

where the subscripts t and d denote some desired "target" and "driver" schemes on the right and left sides, respectively, of the equation.

For hyperbolic equations, the initial convergence of defect correction may be slow for certain, not necessarily high, frequencies [9, 10, 6]. For a target second-order upwind-biased discretization corresponding to $\kappa = 0$, defined subsequently, the asymptotic convergence rate is approximately 0.5 per defect-correction iteration. Thus, it is well-matched with the convergence rate of 0.5 per relaxation expected for the elliptic parts of the operator with Gauss-Seidel relaxation.

Defect correction is implemented in the multigrid algorithm as follows: any discrete evaluations of the residuals of Eq. (3.1) (including residuals transferred to the coarse mesh) are done with the target discretization and any updates via distributed relaxation are done with the driver operator, which is first-order upwind for convection. This is similar to the "double-discretization" approach of Brandt [3] in practice, except that the target residual is evaluated on all of the meshes, including the finest mesh.

6. Numerical Discretization.

6.1. Spatial Discretization. The staggered-grid discretization used here, as shown in Fig. 6.1, is usual: p defined at the cell-centers of the grid, u defined at the cell interfaces tangent to the $y-$ or $k-$direction, and v defined at the cell interfaces tangent to the $x-$ or $j-$direction. Additional values of v and p are defined at inflow and outflow boundaries in order to accommodate boundary conditions, defined subsequently. The discrete scheme with such a staggered-grid arrangement of variables can be described as

$$\mathbf{L}^h\mathbf{q}^h \equiv \begin{bmatrix} Q_\nu^h & 0 & \partial_x^h \\ 0 & Q_\nu^h & \partial_y^h \\ \partial_x^h & \partial_y^h & 0 \end{bmatrix} \mathbf{q}^h = 0, \tag{6.1}$$

where ∂_x^h and ∂_y^h are generally distance-h central differences on the staggered grid. The operator Q_ν^h is composed of convection and diffusion elements, analagous to Eq. (2.3); the diffusion elements are treated with central differencing,

$$(\partial_{yy}u)^h_{j,k} = \frac{1}{(h_y)_{j,k}}[\frac{u_{j,k+1}-u_{j,k}}{(h_y)_{j,k+1/2}} - \frac{u_{j,k}-u_{j,k-1}}{(h_y)_{j,k-1/2}}], \tag{6.2}$$

where h_y denotes grid spacing in the y direction.

The discrete convection operator Q^h is upwind-biased, of either the standard upwind differencing (SUD) type or the narrow upwind differencing (NUD) type. The operator can be defined on a uniform grid in terms of translation operators $T_j^{\pm m}$ and $T_k^{\pm m}$, ($T_j^{\pm m}u_{j,k} = u_{j\pm m,k}$). The SUD scheme can be defined as

$$Q^h = \frac{|u|}{h_x}D(T_j^{sgn(u)}) + \frac{|v|}{h_y}D(T_k^{sgn(v)}) \tag{6.3}$$

where h_x is the grid spacing in the x direction, the sign function sgn is defined as

$$sgn(x) = \begin{cases} +1 & \text{if } x > 0, \\ -1 & \text{if } x < 0, \\ 0 & \text{otherwise,} \end{cases}$$

and D is defined as

$$D(z) \equiv c_{-2}z^{-2} + c_{-1}z^{-1} + c_0 + c_1 z^{+1}.$$

The NUD scheme can be defined as below for $\frac{|u|}{h_x} \geq \frac{|v|}{h_y}$,

$$Q^h = (\frac{|u|}{h_x} - \frac{|v|}{h_y})D(T_j^{sgn(u)}) + \frac{|v|}{h_y}D(T_k^{sgn(v)}T_j^{sgn(u)}) \tag{6.4}$$

and as below for $\frac{|u|}{h_x} < \frac{|v|}{h_y}$,

$$Q^h = \frac{|u|}{h_x}D(T_j^{sgn(u)}T_k^{sgn(v)}) + (\frac{|v|}{h_y} - \frac{|u|}{h_x})D(T_k^{sgn(v)}). \tag{6.5}$$

For uniform meshes or meshes in which the stretching ratio is $\beta = 1 + O(h)$, $\kappa-$schemes of at least second order accuracy (SUD-2 and NUD-2) are defined for $\kappa \in [-1, 1]$ as

$$\{c_{-2}, c_{-1}, c_0, c_1\} = \frac{1}{2+2\beta}\{1-\kappa,\ 3\kappa-5,\ 3(1-\kappa),\ 1+\kappa\}$$

and third-order accuracy (SUD-3) is attained for $\kappa = 1/3$ with uniform meshes or meshes in which the stretching ratio is $\beta = 1 + O(h^2)$. On stretched grids, the reference meshsize, $h_{i-1/2}$, appearing in (the denominator of) the discrete one-dimensional convection operator, $D(T_i)/h_{i-1/2}$, is a meshsize upstream of the i-th node where the discrete operator is defined. The coefficients for the first-order upwind schemes (SUD-1 and NUD-1) are

$$\{c_{-2}, c_{-1}, c_0, c_1\} = \{0,\ -1,\ 1,\ 0\}.$$

TABLE 7.1

Errors in u with the FMG-1 cycle for entering flow using a second order accurate discretization of the continuity equation; t=0.5.

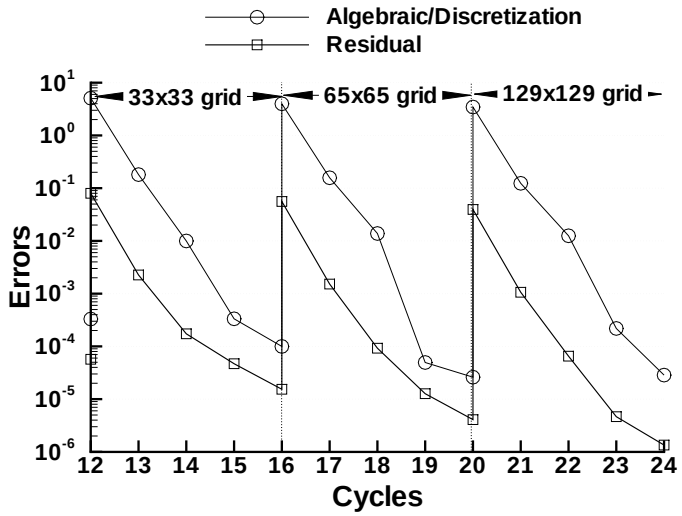
Scheme	h	$\|\|e_d\|\| : u$	$\|\|e_a\|\|/\|\|e_d\|\| : u$
SUD-1	1/16	$0.115556\text{x}10^{0}$	0.019
SUD-1	1/32	$0.664116\text{x}10^{-1}$	0.008
SUD-1	1/64	$0.357011\text{x}10^{-1}$	0.006
SUD-1	1/128	$0.185119\text{x}10^{-1}$	0.002
NUD-1	1/16	$0.476075\text{x}10^{-1}$	0.007
NUD-1	1/32	$0.246260\text{x}10^{-1}$	0.008
NUD-1	1/64	$0.125445\text{x}10^{-1}$	0.006
NUD-1	1/128	$0.633386\text{x}10^{-2}$	0.003
SUD-2	1/16	$0.689001\text{x}10^{-2}$	0.024
SUD-2	1/32	$0.154126\text{x}10^{-2}$	0.039
SUD-2	1/64	$0.368421\text{x}10^{-3}$	0.034
SUD-2	1/128	$0.905679\text{x}10^{-4}$	0.026
NUD-2	1/16	$0.251242\text{x}10^{-2}$	0.128
NUD-2	1/32	$0.637956\text{x}10^{-3}$	0.046
NUD-2	1/64	$0.159458\text{x}10^{-3}$	0.046
NUD-2	1/128	$0.397594\text{x}10^{-4}$	0.047

TABLE 7.2

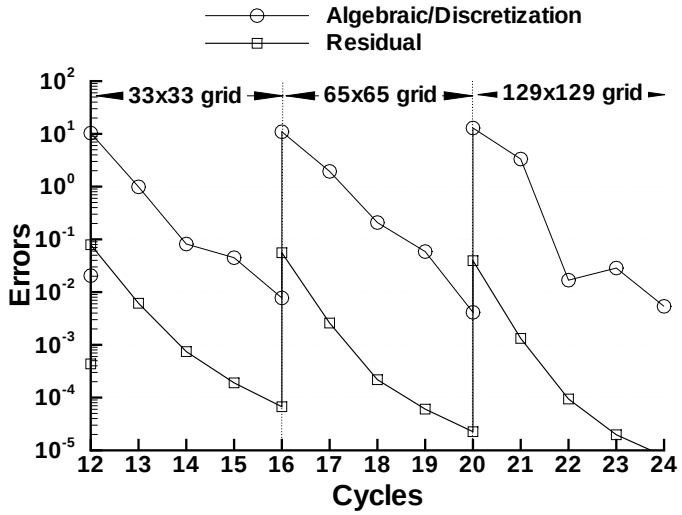
Errors in u and v for entering flow with a fourth order accurate discretization of the continuity equation; t=0.5.

Scheme	h	$\|\|e_d\|\| : u$	$\|\|e_d\|\| : v$
SUD-3	1/32	$0.325327\text{x}10^{-3}$	$0.182866\text{x}10^{-3}$
SUD-3	1/64	$0.425745\text{x}10^{-4}$	$0.231228\text{x}10^{-4}$
SUD-3	1/128	$0.547187\text{x}10^{-5}$	$0.292787\text{x}10^{-5}$
NUD-2	1/32	$0.121481\text{x}10^{-3}$	$0.591146\text{x}10^{-4}$
NUD-2	1/64	$0.151229\text{x}10^{-4}$	$0.724727\text{x}10^{-5}$
NUD-2	1/128	$0.186856\text{x}10^{-5}$	$0.885510\text{x}10^{-6}$

6.2. Gauss-Seidel Line Relaxation. The equations for $\delta\mathbf{w}$ are relaxed with a line-y Gauss-Seidel algorithm marching from the inflow to the outflow boundary. The correction equations for $\delta\mathbf{w}$ are solved implicitly because of the highly stretched mesh used for the viscous calculations. Since the thin-layer approximation is made for the viscous terms, the convective operator is first-order upwind, and there is no streamwise reversed flow, the δw_1 and δw_2 correction (driver) equations of Eq. (4.6), corresponding to the linearized momentum equations at given pressure, are solved exactly. The line-y solutions require only inversions of tridiagonal (rather than block-tridiagonal) matrices, since the equations for $\delta\mathbf{w}$ form a lower triangular set except near the boundaries. The treatment at boundaries requires special consideration as discussed subsequently. Note that for the NUD schemes in inviscid flow with $\frac{|u|}{h_x} < \frac{|v|}{h_y}$, the tridiagonal equations for δw_1 and δw_2 reduce to diagonal equations.



(a) Second order accurate solution.



(b) Third order accurate solution.

FIG. 7.1. *Maximum residual and algebraic-to-discretization errors in u versus multigrid cycle for the three finest grids with the NUD-2 scheme.*

7. Entering Flow Simulation. The flow field upstream of an external aerodynamic simulation is basically inviscid. Brandt and Yavneh[5] considered multigrid solutions of such flows and showed the accuracy of the coarse grid correction to be critically dependent on the alignment of the flow relative to the mesh. Their numerical results indicated the necessity of W-cycles to converge the algebraic errors below discretization errors in the FMG-1 cycle. We revisit these simulations below with slightly different boundary conditions and show that the FMG-1 cycle with the use of FV-cycles is sufficient. The computations were done for a square domain with periodicity in the $y-$direction on a uniform mesh. Inflow boundary conditions were specified velocities as

$$u(0,y) = 1 + 0.5\cos(2\pi y),$$
$$v(0,y) = tu(0,y), \tag{7.1}$$

with constant pressure at the outflow boundary. The tangent of the angle of the flow relative to the grid is $t = 0.5$, corresponding to the maximum value studied by Brandt and Yavneh[5]. The exact solution corresponds to convection along a streamline at constant pressure,

$$u(x,y) = 1 + 0.5\cos(2\pi(y - tx)),$$
$$v(x,y) = tu(x,y). \tag{7.2}$$

The boundary conditions for the correction equations are implemented by applying the distributed relaxation equations $\delta\mathbf{q} = \mathbf{M}\delta\mathbf{w}$ at the boundary along with a Dirichlet condition for $(\delta w)_3$ at inflow. The resulting discrete boundary conditions at $x = 0$ are

$$\delta w_2 = 0,$$
$$\delta w_3 = 0,$$
$$\delta w_1 = \partial_x^h(\delta w_3). \tag{7.3}$$

This boundary condition is the discrete equivalent to the original problem statement for the constant coefficient problem. This boundary condition couples the δw_1 and δw_3 equations together at the line of cells adjacent to the inflow boundary, necessitating a block 2x2 block matrix solution procedure; away from this first line, the equations retain the triangular form of Eq. (4.6) and can be solved as scalar equations. The downstream boundary condition is implemented by solving for δw_3 at the last interior column of cells simultaneously with δw_3 at the outflow column, again necessitating a 2x2 block matrix tridiagonal solution. After sweeping through the domain, all of the momentum equation residuals are zero in the constant coefficient case; this local block matrix coupling at either boundary eliminates the need for the extra sweep of the residual equation advocated by Brandt and Yavneh[5]. The residuals remain non-zero in the general case because of subprincipal terms and are restricted to the coarse grids. Enforcing periodicity in the $y-$direction in the tridiagonal solver eliminates the need to consider any special boundary conditions in that direction. Special forms for the spatial discretization of the convective operator in Eq. (6.1) at inflow and outflow are given in Appendix IV.

The L_2-norms of the discretization errors in u after complete convergence and the ratios of the L_2-norm of the algebraic errors divided by the L_2-norm of the discretization errors after one cycle are shown in Table 7.1 for various grid sizes and orders of accuracy. The algebraic errors are reduced substantially below the discretization errors in one cycle. The error norms indicate a first order accuracy for SUD-1 and NUD-1, and second order for SUD-2 and NUD-2, as expected.

At this flow angle, $t = 0.5$, the NUD-2 scheme exhibits third-order accuracy for the linearized convection problem but does not for the full Euler equations because second order accurate discretizations are used for the continuity equation, for the pressure terms in the momentum equation, and for the reconstruction of the flow at an interface. To remedy this, these discretizations were improved to fourth order accuracy; the corresponding results shown in Table 7.2 for both the SUD-3 and NUD-2 schemes now exhibit third-order accuracy in u and v.

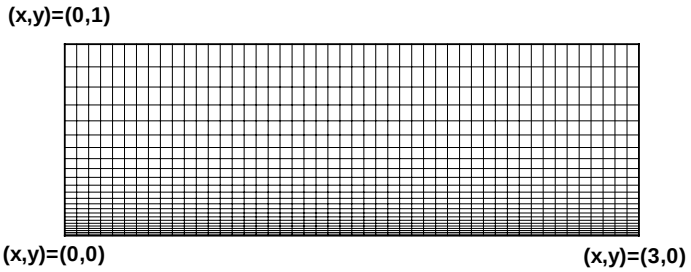


FIG. 8.1. *Grid used for the wake and finite flat plate simulation.*

TABLE 8.1
Computed values of centerline velocity at $x = 1.5$ for the wake simulation; SUD-2 scheme; $\kappa = 0$; $w_d = 0.5$; $Re = 10,000$.

N_xxN$_y$	u (FMG-10)	u (FMG-1)	$\|\|e_a\|\|/\|\|e_d\|\| : u$ (FMG-1)
49 x 25	0.730529	0.730585	0.00445
97 x 49	0.740382	0.740412	0.01135
193 x 97	0.742367	0.742385	0.02672

The reduction of the maximum residual and the algebraic-to-discretization errors over 4 cycles for the three finest grids in the calculation are shown in Fig. 7.1 for the NUD-2 scheme with second and fourth order accurate discretizations of the continuity equation. For second order accuracy, the residual and algebraic-to-discretization errors are reduced four orders of magnitude over the 4 cycles, close to the theoretical limit expected for elliptic equations of $(0.5)^3 = 0.125$ reduction per FV(2,1) cycle. The convergence for the third order accurate results deteriorate somewhat to three orders of magnitude over the four cycles but is still quite reasonable considering that defect correction with a first-order driver operator is being used. Further improvements could be made by additional sweeps or by a predictor-corrector sequence of the momentum equations only, since the deficiency resides with the first-order accuracy in the driver operator for convection.

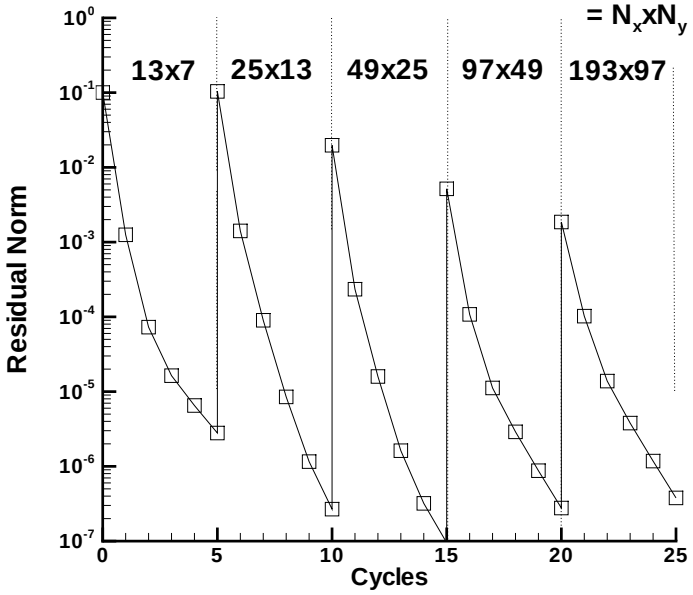
8. Wake Flow Simulation. The wake and the finite flat plate simulation to follow were computed for the computational domain shown in Fig. 8.1 at a Re=10,000 based on the height of the channel. The grid was stretched in the $y-$direction with a stretching factor on a specified mesh defined as

$$\beta_0 = (h_y)_{j,k+1}/(h_y)_{j,k}$$

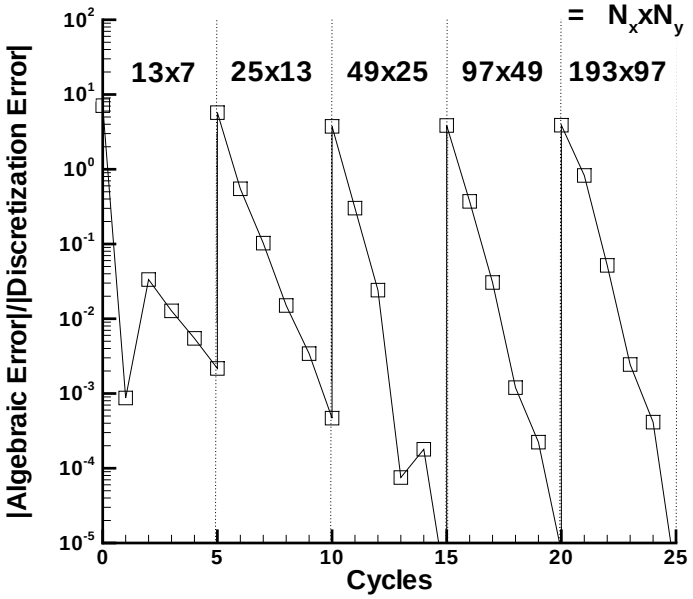
corresponding to $(N_y)_0$ grid points in the vertical direction. The stretching ratio on all other meshes is

$$\beta = \beta_0^{((N_y)_0-1)/(N_y-1)}.$$

Freestream pressure is specified at the outflow boundary; a wake deficit was prescribed at the inflow boundary, $x = 0$, according to



(a) L_2-norm of the residual.



(b) Algebraic-to-discretization errors in mass flow.

FIG. 8.2. *Wake simulation convergence using the FMG-5 cycle;* $w_d = 0.5$*; SUD-2 scheme;* $\kappa = 0$.

$$u(0,y) = 1 - w_d \exp(\frac{-\mathrm{Re}\,\mathrm{y}^2}{4}), \;\; \mathrm{v}(0,\mathrm{y}) = 0,$$

where $w_d = 0.5$. The mass flow is defined as the integral of velocity at constant x; the exact value is 0.9911377307. The boundary condition treatments at inflow and outflow are the same as those for the

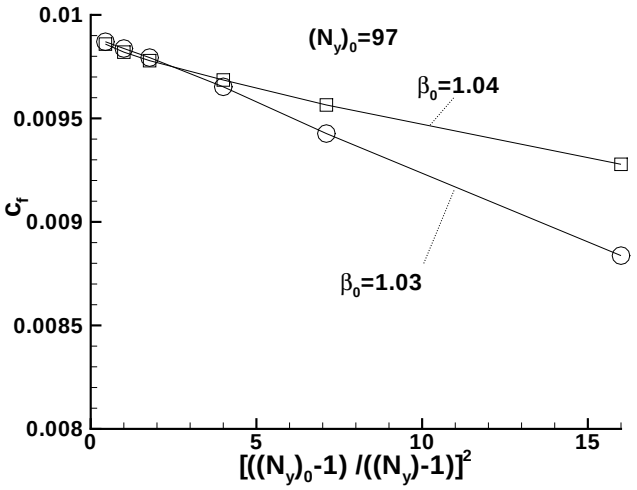


FIG. 9.1. *Convergence of the C_f values at $x = 1.5$ with nominal grid spacing for two stretching ratios; SUD-2 scheme; $\kappa = 0$; $Re = 10,000$.*

entering flow discussed previously. Symmetry conditions are applied at $y = 0$ for both $\mathbf{q}$ and $\delta\mathbf{w}$. A tangency condition, $v = 0$, was applied at $y = 1$; applying the distribution operator at this point with simple reflection for u across the boundary indicates that a Neumann condition can be applied to δw_3 at the boundary, as shown in Appendix II, along with reflection for δw_1 and a Dirichlet condition for δw_2, if needed.

The finest grid considered was N_xxN$_y$ = 193x97 with $\beta_0 = 1.03$ corresponding to $(N_y)_0 = 97$. In addition to residuals, the centerline velocity (obtained by second-order extrapolation) and the mass flow were monitored at $x = 1.5$, a location midway in the domain, as a measure of spatial convergence. The reductions of the maximum residual and the algebraic-discretization errors in mass flow for all the grids in an FMG-5 process are shown in Fig. 8.2 using the SUD-2 scheme. For each of the meshes, the residual is reduced 4-5 orders of magnitude and the algebraic errors are reduced far below discretization errors. The centerline velocities for the three finest grids, Table 8.1, demonstrate second order accuracy with algebraic errors reduced below discretization errors using the FMG-1 cycle. The reference centerline velocity was obtained by second order Richardson extrapolation.

Although not shown, parameter variations in w_d were made which indicated the results were not sensitive to w_d over the range investigated, 0 to 0.9. This is in contrast to an earlier application,[10] in which the ghost variable equations were solved with a correction scheme (CS) multigrid. Those results deteriorated for high values of w_d, emphasizing the advantage of applying the FAS multigrid scheme to the whole nonlinear system of equations. For linear equations, the performance of FAS multigrid is the same as CS multigrid .

9. Flat Plate Boundary Layer Simulation. For the flat plate simulation, no-slip conditions are prescribed from $x = 1$ to $x = 2$ along the lower boundary and symmetry conditions upstream and downstream of those points; a wake profile develops downstream of the trailing edge, $x = 2$. The inflow and outflow conditions are prescribed freestream velocities ($u_\infty = 1, v_\infty = 0$) and pressures, respectively. The discrete velocities adjacent to the plate for $y < 0$ are required to satisfy the no-slip condition at the plate, i.e. $u(x, -h_y/2) = -u(x, h_y/2); v(x, -h_y) = -v(x, h_y)$. The distributive relaxation equations applied at the boundary are shown in Appendix III.

TABLE 9.1

Computed values of total drag for the finite flat plate simulation; SUD-2 scheme; $\kappa = 0$; $Re = 10,000$; $\beta_0 = 1.03$.

N_xxN$_y$	C_D (FMG-10)	C_D (FMG-1)	$\|\|e_a\|\|/\|\|e_d\|\| : C_D$ (FMG-1)
49 x 25	0.011552	0.011753	0.0784
97 x 49	0.013492	0.013412	0.1284
193 x 97	0.013961	0.014051	0.5760

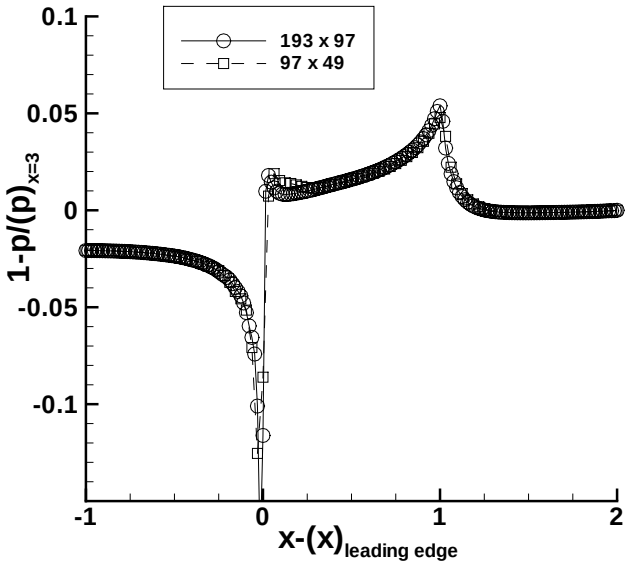


FIG. 9.2. *Pressures ($y = 0$) for the finite flate plate; $Re = 10,000$.*

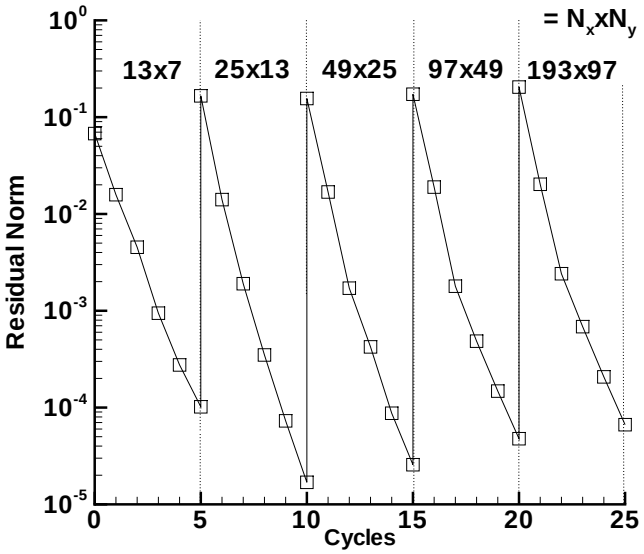
The spatial convergence of the local skin friction C_f midway down the plate versus the nominal grid spacing for two families of meshes for two stretching ratios is shown in Fig. 9.1, where

$$C_f = 2\nu(\partial_y^h u)/u_\infty^2.$$

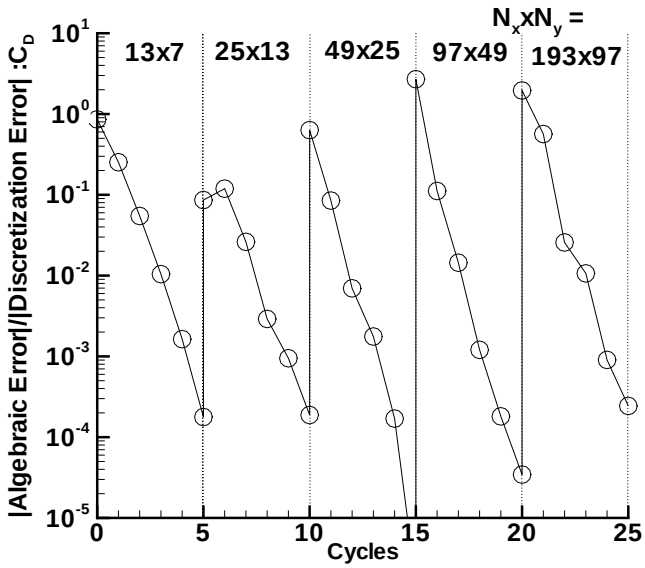
The two finest grids in each family are 289x145 and 193x97. Second order accuracy is evident; the results with higher stretching ratio are slightly more accurate on coarser grids. The results converge to a value approximately five percent higher than the Blasius value, $C_f = 0.664/\sqrt{Re_{\bar{x}}} = 0.00939$, where $\bar{x}$ denotes distance from the leading edge, because of the presence of a favorable pressure gradient (accelerating flow) over most of the plate, as shown in Fig. 9.2. Convergence of the L_2-norm of the residual and estimated algebraic-to-discretization errors in total drag C_D are shown in Fig. 9.3. The total drag is defined as

$$C_D = 2C_f^*(x^* - 1) + \int_{x^*}^{2} C_f dx,$$

where the C_f behavior ahead of $x^* = 1.25$ is assumed to be an inverse square root behavior in distance from the leading edge, as occurs with the Blasius solution. The infinite-grid result is extrapolated using the two



(a) L_2-norm of the residual.



(b) Algebraic-to-discretization errors in C_D.

FIG. 9.3. *Errors per cycle using the FMG-5 cycle;* $\beta_0 = 1.03$*; SUD-2 scheme;*$\kappa = 0$*;* $Re = 10,000$.

finest grids. Both the residual and algebraic-to-discretization errors are reduced nearly four orders of magnitude over five cycles for the four finest grids, close to the convergence expected for elliptic equations. The C_D values on the three finest meshes are given in Table 9.1, confirming that the algebraic-to-discretization errors are reduced below unity in a single cycle. The values extrapolate to a slightly larger value than the Blasius value, $C_D = 1.338/\sqrt{Re} = 0.013280$. Velocities normalized to the boundary layer edge velocity, u_e, versus the scaled normal coordinate, η, are shown in Fig. 9.4 for the two finest grids in one family; either computation is indistinguishable from the Falkner-Skan boundary layer analytic result that accounts for

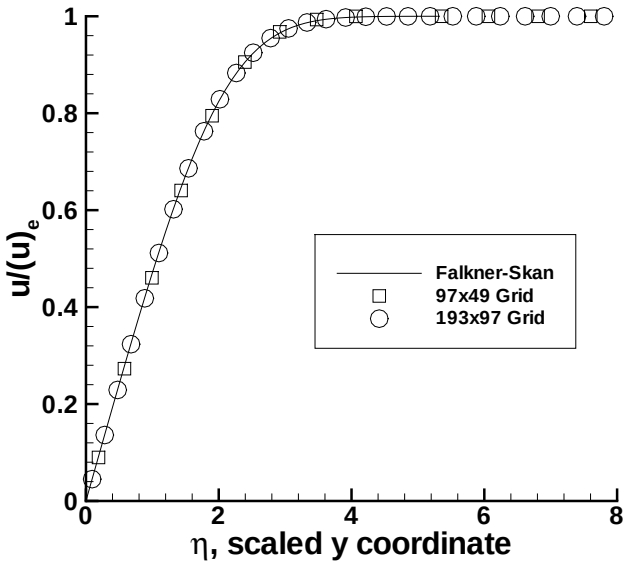


FIG. 9.4. *Scaled velocity profiles for the finite plate at* $x = 1.5$*;* $Re = 10,000$*;* $\eta = (y/\bar{x})\sqrt{Re_{\bar{x}}/2}$.

streamwise pressure gradient.

The largest discretization errors as well as the largest residuals occur near the leading edge singularity, as can be noted in Fig. 9.2. Although not tried, a local refinement near this boundary would be beneficial.

10. Concluding Remarks. A multigrid method for solving the incompressible Navier-Stokes equations has been applied to a classical model problem of fluid dynamics: flow past a finite flat plate at high Reynolds number. Elements of the Full Approximation Scheme multigrid algorithm, including distributed relaxation, defect correction, and boundary treatment, have been presented in some detail for the three main physical aspects encountered in the simulation: entering flow, wake flow, and boundary layer flow. Textbook efficiencies, i.e., reduction of algebraic errors below discretization errors in one multigrid cycle, and residual reduction rates approaching the value expected for elliptic equations of nearly one order of magnitude per cycle, are attained for second order accurate simulations at a laminar Reynolds number of 10,000.

REFERENCES

[1] R. M. BEAM AND R. F. WARMING, *An implicit factored scheme for the compressible Navier-Stokes equations*, AIAA Journal, 16 (1978), pp. 393–401.

[2] A. BRANDT, *Guide to multigrid development*, in Multigrid Methods, W. Hackbusch and U. Trottenberg, eds., Lecture Notes in Math. 960, Springer-Verlag, Berlin, 1982.

[3] ——, *Multigrid techniques: 1984 guide with applications to fluid dynamics*. Monograph (unpublished). GMD-Studie 85, GMD-FIT, Postfach 1240, D-5205, St. Augustin 1, Germany. Also available from Secretary, Department of Mathematics, University of Colorado at Denver, Colorado 80204-5300, 1985.

[4] ——, *Barriers to achieving textbook multigrid efficiency in CFD*, ICASE Interim Report 32, April 1998.

[5] A. BRANDT AND I. YAVNEH, *On multigrid solution of high-Reynolds incompressible entering flow*, J. Comput. Phys., 101 (1992), pp. 151–164.

[6] B. Diskin and J. L. Thomas, *Solving upwind-biased discretizations: Defect-correction iterations*, ICASE Report 99-14, March 1999.

[7] S. L. Krist, R. T. Biedron, and C. L. Rumsey, *CFL3D User's Manual (Version 5.0)*, NASA TM-1998-208444, June 1998.

[8] T. H. Pulliam and J. L. Steger, *On implicit finite difference simulations of three dimensional flow*, AIAA Paper 78-10, 1978.

[9] J. L. Thomas, D. L. Bonhaus, W. K. Anderson, C. L. Rumsey, and R. T. Biedron, *An $O(n\,m^2)$ plane solver for the compressible Navier-Stokes equations*, AIAA Paper 99-0785, 37th Aerospace Sciences Meeting & Exhibit, Reno, NV, January 1999.

[10] J. L. Thomas, B. Diskin, and A. Brandt, *Distributed relaxation multigrid and defect correction applied to the compressible Navier-Stokes equations*, AIAA Paper 99-3334, 14th Computational Fluid Dynamics Conference, Norfolk, VA, July 1999.

Appendix A. Linearized Euler Equations.

The linearized Euler equations,

$$\mathbf{L}_\infty(\ \mathbf{q}\ ) = 0, \tag{A.1}$$

with periodicity in the $y-$direction over a finite domain, $0 \le x \le L$, are considered, where $\mathbf{q}$ represents a perturbation from freestream values. The convection operator is assumed to be *constant* as

$$Q_0 \equiv \partial_x + t\partial_y, \tag{A.2}$$

where $t \equiv v_\infty/u_\infty$ represents the incidence of the freestream flow with the $x-$axis. The boundary conditions are taken as prescribed velocity components at inflow and pressure at outflow,

$$\begin{aligned} \begin{pmatrix} u \\ v \end{pmatrix}_{x=0} &= \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} e^{i\omega y}, \\ \begin{pmatrix} p \end{pmatrix}_{x=L} &= \begin{pmatrix} p_L \end{pmatrix} e^{i\omega y}. \end{aligned} \tag{A.3}$$

Brandt and Yavneh[5] considered entering flow ($L \to \infty$) with inclusion of the first differential approximations of the discrete equations to confirm algebraic convergence below discretization error in one FMG cycle, neglecting boundary effects. Here, we consider only the differential solution using distributed relaxation, $\mathbf{q} = \mathbf{M}_\infty \mathbf{w}$, and include boundary effects. Considering $\mathbf{w}$ of the form

$$\mathbf{w} = \begin{pmatrix} a \\ b \\ c \end{pmatrix} e^{-\alpha x} e^{i\omega y}, \tag{A.4}$$

then $\mathbf{L}_\infty \mathbf{M}_\infty \mathbf{w} = 0$ implies

$$\begin{bmatrix} -\alpha + i\omega t & 0 & 0 \\ 0 & -\alpha + i\omega t & 0 \\ -\alpha & i\omega & -\alpha^2 + \omega^2 \end{bmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = 0. \tag{A.5}$$

A non-trivial solution (zero determinant) exists for values of α as below,

$$\alpha \subset \{i\omega t, i\omega t, \omega, -\omega\}. \tag{A.6}$$

Thus, the general solution, $\mathbf{w} \equiv \bar{\mathbf{w}} e^{i\omega y}$, can be written as

$$\begin{aligned}\bar{\mathbf{w}} = B_1 \begin{pmatrix} 0 \\ 1 \\ \frac{-i}{\omega(1+t^2)} \end{pmatrix} e^{-i\omega t x} + B_2 \begin{pmatrix} 1 \\ 0 \\ \frac{it}{\omega(1+t^2)} \end{pmatrix} e^{-i\omega t x} \\ + B_3 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} e^{-\omega x} + B_4 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} e^{\omega x},\end{aligned} \tag{A.7}$$

which requires four boundary conditions to close the system, instead of the three required with the primitive equations. Applying a Dirichlet condition for w_3 at inflow supplemented with the original boundary conditions, as below,

$$\begin{aligned}\begin{pmatrix} w_1 - \partial_x(w_3) \\ w_2 \\ w_3 \end{pmatrix}_{x=0} &= \begin{pmatrix} u_0 \\ v_0 \\ 0 \end{pmatrix} e^{i\omega y}, \\ \begin{pmatrix} Q_0 w_3 \end{pmatrix}_{x=L} &= \begin{pmatrix} p_L \end{pmatrix} e^{i\omega y},\end{aligned} \tag{A.8}$$

the coefficients $B_1 - B_4$ can be determined and are given below:

$$\begin{aligned}B_1 &= v_0, \\ B_2 &= u_0 + \frac{1}{D_1}\left[(tu_0 - v_0) i D_2 + 2 p_L e^{-\omega L}\right], \\ B_3 &= \frac{1}{\omega D_1 (t+i)}\left[(tu_0 - v_0) - i p_L e^{-\omega L}\right], \\ B_4 &= \frac{1}{\omega D_1 (t-i)}\left[-(tu_0 - v_0) e^{-2\omega L} - i p_L e^{-\omega L}\right],\end{aligned} \tag{A.9}$$

where $D_1 \equiv 1 + e^{-2\omega L}$ and $D_2 \equiv 1 - e^{-2\omega L}$. Note that w_2 is a function of v_0 only; w_1 is primarily a function of u_0 but is coupled to v_0 and p_L through the boundary conditions, Eq. (A.8); the coupling is rather weak, however, as it disappears completely for $v_0 = tu_0$, as is usually the case, and $L \to \infty$.

The primitive variables, $\mathbf{q} \equiv \bar{\mathbf{q}} e^{i\omega y}$, can be determined from $\mathbf{q} = \mathbf{M}_\infty \mathbf{w}$, as below,

$$\begin{aligned}\bar{\mathbf{q}} = \frac{A_1}{1+t^2} \begin{pmatrix} 1 \\ t \\ 0 \end{pmatrix} e^{-i\omega t x} + \frac{A_2}{D_1} \begin{pmatrix} \frac{-i}{t+i} \\ \frac{-1}{t+i} \\ 1 \end{pmatrix} e^{-\omega x} \\ + \frac{A_3}{D_1} \begin{pmatrix} \frac{i}{t-i} \\ \frac{-1}{t-i} \\ 1 \end{pmatrix} e^{\omega x},\end{aligned} \tag{A.10}$$

$$
\begin{aligned}
A_1 &= u_0 + tv_0 + \frac{D_2}{D_1}[i(tu_0 - v_0) + 2p_L e^{-\omega L}], \\
A_2 &= +i(tu_0 - v_0) + p_L e^{-\omega L}, \\
A_3 &= -i(tu_0 - v_0)e^{-2\omega L} + p_L e^{-\omega L},
\end{aligned} \tag{A.11}
$$

It can be verified that the solution above satisfies (A.1) and the boundary conditions (A.3). The boundary conditions for $\mathbf{w}$ discussed in the main body of the text are discrete forms of the differential boundary conditions given by (A.8) above. For the linearized, constant coefficient case considered here, both the discrete and differential forms share the property that a solution to the distributed relaxation equations with boundary conditions (A.8) satisfy identically the differential equations (A.1) with boundary conditions (A.3).

Appendix B. Tangency.

The linearized Euler equations, Eq. (A.1), are again considered but with $t = 0$ ($\mathbf{Q}_0 \equiv \partial_x$) and with periodicity in the $x-$direction over a finite domain, $0 \leq y \leq H$. Linearized tangency boundary conditions are prescribed as v at the top and bottom of the channel,

$$
\begin{pmatrix} v(x,0) \\ v(x,H) \end{pmatrix} = \begin{pmatrix} v_0 \\ v_H \end{pmatrix} e^{i\omega x}. \tag{B.1}
$$

Considering $\mathbf{w}$ of the form

$$
\mathbf{w} = \begin{pmatrix} a \\ b \\ c \end{pmatrix} e^{-\alpha y} e^{i\omega x}, \tag{B.2}
$$

then $\mathbf{L}_\infty \mathbf{M}_\infty \mathbf{w} = 0$ implies

$$
\begin{bmatrix} i\omega & 0 & 0 \\ 0 & i\omega & 0 \\ i\omega & -\alpha & -\alpha^2 + \omega^2 \end{bmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = 0. \tag{B.3}
$$

A non-trivial solution (zero determinant) exists for values of $\alpha \subset \{\omega, -\omega\}$.
Thus, the general solution, $\mathbf{w} \equiv \bar{\mathbf{w}} e^{i\omega x}$, can be written as

$$
\bar{\mathbf{w}} = B_1 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} e^{\omega y} + B_2 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} e^{-\omega y}, \tag{B.4}
$$

which only requires two boundary conditions, consistent with the primitive equations; no boundary conditions can be given for w_1 and w_2. Applying a Neumann condition for w_3, as below,

$$
\begin{pmatrix} (-\partial_y(w_3))(x,0) \\ (-\partial_y(w_3))(x,H) \end{pmatrix} = \begin{pmatrix} v_0 \\ v_H \end{pmatrix} e^{i\omega x}, \tag{B.5}
$$

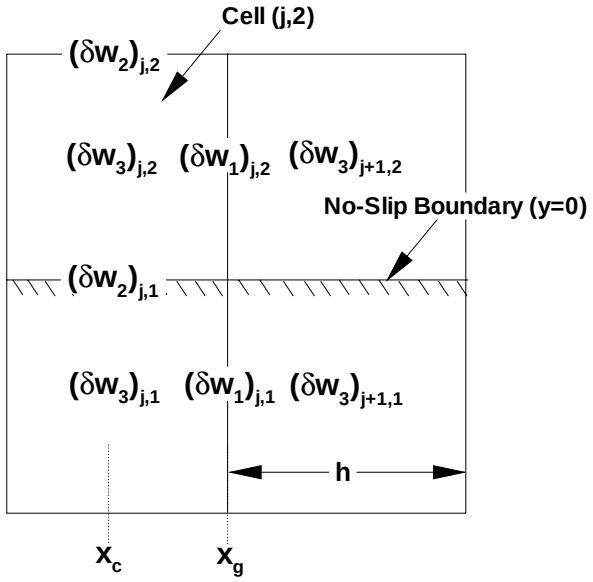


FIG. C.1. *Schematic of variables near no-slip boundary.*

the coefficients B_1 and B_2 can be determined and are given below:

$$B_1 = \frac{-e^{-\omega H}}{\omega D}\left[v_H - v_0 e^{-\omega H}\right],$$
$$B_2 = \frac{1}{\omega D}\left[v_0 - v_H e^{-\omega H}\right], \tag{B.6}$$

where $D \equiv 1 + e^{-2\omega H}$.

The primitive variables, $\mathbf{q} \equiv \bar{\mathbf{q}} e^{i\omega x}$, can be determined from $\mathbf{q} = \mathbf{M}_\infty \mathbf{w}$, as below,

$$\bar{\mathbf{q}} = B_1 \begin{pmatrix} -i\omega \\ -\omega \\ i\omega \end{pmatrix} e^{\omega y} + B_1 \begin{pmatrix} -i\omega \\ \omega \\ i\omega \end{pmatrix} e^{-\omega y}. \tag{B.7}$$

It can be verified that the solution above satisfies (A.1) and the boundary conditions (B.1), recovering the classical aerodynamic model problem for the flow past a wavy wall. It is clear that w_3 takes the role of the perturbation potential; the Neumann boundary conditions for w_3 are implemented discretely at the tangency surfaces in the main body of the text.

Appendix C. No-Slip Boundary.

The no-slip boundary conditions along the plate in terms of the ghost variables are

$$[\delta w_2](x_c, 0) = [\partial_y^h(\delta w_3)](x_c, 0), \tag{C.1}$$
$$[\delta w_1](x_g, 0) = [\partial_x^h(\delta w_3)](x_g, 0), \tag{C.2}$$

where x_c denotes the x position of the cell-center for cell (j,k) and $x_g = x_c + \frac{h}{2}$, as in Fig. C.1. Since a third boundary condition for the ghost variables is required at the plate, we choose to split Eq. (C.1) into two separate equations as

$$[\delta w_2](x_c, 0) = 0, \tag{C.3}$$

$$[\partial_y^h(\delta w_3)](x_c, 0) = 0. \tag{C.4}$$

At the location $(x_g, 0)$, δw_1 and δw_3 can be approximated in terms of nearby values as

$$[\delta w_1](x_g, 0) = \frac{1}{2}(\delta w_1)_{j,2} + \frac{1}{2}(\delta w_1)_{j,1}, \tag{C.5}$$

$$\begin{aligned}[\partial_x^h(\delta w_3)](x_g, 0) =& \frac{1}{2h}[(\delta w_3)_{j+1,2} - (\delta w_3)_{j,2} \\ &+ (\delta w_3)_{j+1,1} - (\delta w_3)_{j,1}],\end{aligned} \tag{C.6}$$

In relaxing the j^{th} column, we assume that $(\delta w_3)_{j+1,k} = 0$. From Eq. (C.4), we also have $(\delta w_3)_{j,1} = (\delta w_3)_{j,2}$. Then Eq. (C.2) can be written as

$$(\delta w_1)_{j,1} = -(\delta w_1)_{j,2} - \frac{2}{h}(\delta w_3)_{j,2}, \tag{C.7}$$

which is an implicit boundary condition equation to be implemented in relaxing Eq. (4.6) at the wall.

Now assume the convection-diffusion operator is constant, defined with a computational stencil as below,

$$Q_\nu \equiv \begin{bmatrix} & c_N & \\ c_W & c_0 & c_E \\ & c_S & \end{bmatrix}. \tag{C.8}$$

In a lexicographic pointwise relaxation, the matrix to solve for the $(\delta \mathbf{w})_{j,2}$ values is as below,

$$\begin{bmatrix} c_0 & 0 & 0 \\ 0 & c_0 - c_S & -2c_S h^{-1} \\ h^{-1} & h^{-1} & 3h^{-2} \end{bmatrix} \begin{pmatrix} \delta w_2 \\ \delta w_1 \\ \delta w_3 \end{pmatrix}_{j,2} = -\begin{pmatrix} r_2 \\ r_1 \\ r_3 \end{pmatrix}_{j,2}. \tag{C.9}$$

This system couples the implicit equations for δw_1 and δw_3 at the cell adjacent to the no-slip boundary, necessitating a local 2x2 block matrix solution. After solving for $(\delta \mathbf{w})_{j,2}$ (and thereby $(\delta w_3)_{j,1}$ and $(\delta w_1)_{j,1}$) and changing the primitive variables through $\mathbf{M}\delta\mathbf{w}$, it can be shown that the updated residuals of cell (j, k) are zero. For variable coefficients in the convection-diffusion operator, the residuals differ from zero, as they do in the interior of the mesh.

Considering relaxation of the entire column of cells, the implicit equations for the cells away from the boundary remain in lower triangular form. Thus, the equations can be solved using an LU decomposition with only a small overhead. In this instance, the entire column of residuals are zeroed out for a constant coefficient convection-diffusion operator.

Appendix D. Boundary Stencil Modifications.

The four-point upwind-biased stencil considered here requires special treatment near boundaries. For prescribed velocity boundary conditions at inflow, a modification is required at $j = 2$ for the $x-$momentum equation and at $j = 2$ and $j = 3$ for the $y-$momentum equation. For prescribed pressure at outflow, a

modification is required at $j = N_x$ for both momentum equations. The simplest approach to maintain second order accuracy, used herein for the wake and boundary layer simulations, is to use a first-order two-point stencil for these points near inflow and a fully-upwind stencil ($\kappa = -1$) near outflow. For the entering flow simulation, more accurate stencils were used at inflow as shown below; $u_{1,k}$ and $v_{3/2,k}$ represent given boundary values at $x = 0$, as in Fig. 6.1.

Considering the $x-$momentum equation, for the SUD-2 scheme, the required term u_x is computed using nearby points and the gradient at $x = 0$, i.e.,

$$\begin{aligned}(\partial_x^h u)|_{h,y} =& \frac{1}{4h}[-5u_{1,k} + 4u_{2,k} + u_{3,k}] \\ & - \frac{1}{2}(\partial_x^h u)|_{0,y},\end{aligned} \tag{D.1}$$

where (h, y) denotes the vertical interface midpoint of the $(2, k)$ cell and $(\partial_x^h u)|_{0,y} = -(\partial_y^h v)|_{0,y}$ is given at inflow from continuity, as

$$\begin{aligned}(\partial_y^h v)|_{0,y} = \frac{1}{24h}[&27v_{3/2,k} - 27v_{3/2,k-1} \\ & - v_{3/2,k+1} + v_{3/2,k-2}].\end{aligned} \tag{D.2}$$

For the NUD-2 scheme, central differencing ($\kappa = +1$) is used.

Considering the $y-$momentum equation with either scheme, central differencing is used at the $j = 2$ column of cells and a third-order 4-point formula at the $j = 3$ column of cells, i.e.,

$$(\partial_x^h v)|_{h/2,y} = \frac{1}{3h}[-4v_{3/2,k} + 3v_{2,k} + v_{3,k}], \tag{D.3}$$

$$\begin{aligned}(\partial_x^h v)|_{3h/2,y} = \frac{1}{30h}[&16v_{3/2,k} - 45v_{2,k} \\ & + 20v_{3,k} + 9v_{4,k}],\end{aligned} \tag{D.4}$$

where $(h/2, y)$ and $(3h/2, y)$ denote the horizontal interface midpoints of the $(2, k)$ and $(3, k)$ cells, respectively.