Relaxation Revisited—A Fresh Look at Multigrid for Steady Flows

Thomas W. Roberts*, David Sidilkover† and R. C. Swanson*

1 Introduction

The year 1971 saw the publication of one of the landmark papers in computational aerodynamics, that of Murman and Cole [9]. As with many seminal works, its significance lies not so much in the specific problem that it addressed—small disturbance, plane transonic flow—but in the identification of a general approach to the solution of a technically important and theoretically difficult problem. The key features of Murman and Cole's work were the use of type-dependent differencing to correctly account for the proper domain of dependence of a mixed elliptic/hyperbolic equation, and the introduction of line relaxation to solve the steady flow equation. All subsequent work in transonic potential flows was based on these concepts. Jameson [6] extended Murman and Cole's ideas to the full potential equation with two important contributions. First, he introduced the rotated difference stencil, which generalized the Murman and Cole type-dependent difference operator to general coordinates. Second, he used the interpretation, introduced by Garabedian, of relaxation as an iteration in artificial time to construct stable relaxation schemes, generalizing the original line relaxation method of Reference [9]. The decade of the 1970s saw an explosion of activity in the solution of transonic potential flows, which has been summarized in the review article of Caughey [4].

At about the time of Caughey's survey, the main thrust of research in computational aerodynamics was moving away from the full potential equation and towards the solution of the steady Euler equations. By analogy with relaxation methods for potential equation, solution methods for the Euler equations can be thought of as iterations in pseudo-time. Unlike methods for the potential equation, by far the most common approach has been to solve for steady flows

* Research Scientist, Aerodynamic and Acoustic Methods Branch, NASA Langley Research Center, Hampton, VA 23681-0001

† Senior Staff Scientist, ICASE, Hampton, VA 23681-0001

as the long-time asymptotic solution of the unsteady equations, rather than directly solve the steady equations. This is much easier conceptually, as the unsteady Euler equations are a hyperbolic system, for which a considerable body of theory exists. Abandoning time accuracy allows considerable flexibility in the construction of the iterative scheme. To accelerate the convergence to the steady state, various types of preconditioning are used. In addition, starting with the unsteady equations leads to a straightforward extension of the iterative methods to the Reynolds-averaged Navier-Stokes equations, and considerable progress in this area has been made in recent years.

Nevertheless, there is still a great need to improve the convergence rates of existing methods. Both the line-relaxation methods for the potential equation and the time-iterative methods for the Euler and Navier-Stokes equations suffer from slow asymptotic convergence to the steady state. Interpreting the iteration as relaxation leads one naturally to consider convergence acceleration methods that have been successfully applied to classical relaxation schemes. The foremost among such methods is the multigrid algorithm. The theory of multigrid is highly developed for elliptic equations, for which $O(n)$ convergence rates are attainable, where n is the number of unknowns in the system. In other words, the work required to obtain a solution to the system of equations is proportional to the number of unknowns. Classical relaxation schemes for elliptic equations are extremely efficient at eliminating the short-wavelength components of the error, while the coarse grids in the multigrid process are efficient at removing the long-wavelength errors. Application of multigrid acceleration to the Euler or Reynolds-averaged Navier-Stokes equations leads one to consider temporal integration methods which also provide good damping of the short-wavelength error components.

There are two main classes of multigrid methods based on the unsteady equations. One class of methods uses upwind-differencing and implicit time integration as the smoother [1, 8, 17]. An alternative approach is one originally proposed by Jameson [7]. A finite-volume spatial discretization with explicit artificial viscosity is combined with a Runge-Kutta time integration as a smoother. This approach has been successfully extended to the Reynolds-averaged Navier-Stokes equations [16]. Unfortunately, these approaches have resulted in poor multigrid efficiency. When applied to high Reynolds number flows over complex geometries, convergence rates are often worse than 0.99 per multigrid cycle. Recently, significant improvements have been demonstrated by Pierce, et al. [10]. However, when one considers that for the Poisson equation on smooth domains convergence rates of nearly 0.1 per cycle are attainable in practice, it is clear that there is tremendous room for improvement of existing flow solvers.

In the remainder of this paper, a multigrid algorithm for the Euler equations which yields convergence rates comparable to those of the Poisson equation is presented. This algorithm abandons the time-marching approach to the steady

state, but relies on relaxation of the steady equations. In Section 2, the general principles underlying the algorithm are outlined. The mathematical formulation of the current approach is given in Section 3, while the solution procedure is described in Section 4. Results for incompressible, inviscid flow in two-dimensional channels and around airfoils are shown in Section 5. A brief discussion of the extension of the current method to the compressible flow equations is presented in Section 6, where a connection with potential flow solvers is also shown. A summary is found in Section 7.

2 An Approach to Multigrid

According to Brandt [2], one of the major obstacles to achieving ideal multigrid performance for advection dominated flows is that the coarse grid provides only a fraction of the needed correction for smooth error components. This particular obstacle can be removed by designing a solver that effectively distinguishes between the elliptic, parabolic, and hyperbolic (advection) factors of the system and treats each one appropriately. The efficiency of such a solver will be limited by the efficiency of the solvers for each of the factors of the system. For instance, advection can be treated by space marching, while elliptic factors can be treated by multigrid. In this example, all components of the error associated with the advection terms are eliminated in one sweep, and the convergence rate is limited by the speed of the elliptic solver. Brandt presents an approach called "distributive relaxation" by which one can construct smoothers that effectively distinguish between the different factors of the operator. Using this approach, Brandt and Yavneh have demonstrated textbook multigrid convergence rates for the incompressible Navier-Stokes equations [3]. Their results are for a simple geometry and a Cartesian grid, using a staggered-grid discretization of the equations.

In a closely related approach, Ta'asan [15] presents a fast multigrid solver for the compressible Euler equations. This method is based on a set of "canonical variables" which express the steady Euler equations in terms of an elliptic and a hyperbolic partition [14]. Ta'asan uses this partition to guide the discretization of the equations. A staggered grid is used, with different variables residing at cell, vertex, and edge centers. In Reference [15] it is shown that ideal multigrid efficiency can be achieved for the compressible Euler equations for two-dimensional subsonic flow using body-fitted grids. One possible limitation of the use of canonical variables is that the partition of the inviscid equations is not directly applicable to the viscous equations.

Recently the authors [11] have presented an alternative scheme to Brandt's distributive relaxation and to Ta'asan's canonical variable decomposition. This scheme does not require staggered grids, but uses conventional vertex-based finite-volume or finite-difference discretizations of the primitive variables. This

simplifies the restriction and prolongation operations, because the same operator can be used for all variables. A projection operator is applied to the system of equations, resulting in a Poisson equation for the pressure. By applying the projection operator to the discrete equations rather than to the differential equations, the proper boundary condition on the pressure is satisfied directly. The Poisson equation for the pressure may be treated by Gauss-Seidel relaxation, while the advection terms of the momentum equation are treated by space-marching. Because the elliptic and advection parts of the system are decoupled, ideal multigrid efficiency can be achieved. Compared to distributive relaxation and the canonical variables approaches, this method is extremely simple.

3 Mathematical Formulation

The incompressible Euler equations in primitive variables are

$$\begin{aligned} uu_x + vu_y + p_x &= 0, \\ uv_x + vv_y + p_y &= 0, \\ u_x + v_y &= 0, \end{aligned}$$

where u and v are the components of the velocity in the x and y directions, respectively, and p is the pressure. The density is taken to be one. The advection operator is defined by

$$Q \equiv u\partial_x + v\partial_y, \tag{1}$$

where ∂_x, ∂_y are the partial differentiation operators. The Euler equations may be written as

$$\mathbf{Lq} = \begin{pmatrix} Q & 0 & \partial_x \\ 0 & Q & \partial_y \\ \partial_x & \partial_y & 0 \end{pmatrix} \begin{pmatrix} u \\ v \\ p \end{pmatrix} = 0. \tag{2}$$

Introducing the adjoint to Q, defined by

$$Q^*(f) \equiv -\partial_x(uf) - \partial_y(vf), \tag{3}$$

a projection operator $\mathbf{P}$ is defined:

$$\mathbf{P} = \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ \partial_x & \partial_y & Q^* \end{pmatrix}. \tag{4}$$

Applying the projection operator to the Euler equations yields

$$\tilde{\mathbf{L}}\mathbf{q} \equiv \mathbf{P}\mathbf{L}\mathbf{q} = \begin{pmatrix} Q & 0 & \partial_x \\ 0 & Q & \partial_y \\ 0 & 0 & \Delta \end{pmatrix} \begin{pmatrix} u \\ v \\ p \end{pmatrix} + 2 \begin{pmatrix} 0 \\ 0 \\ (\partial_x v)(\partial_y u) - (\partial_x u)(\partial_y v) \end{pmatrix}, \qquad (5)$$

where Δ is the Laplacian. The matrix operator on the right-hand side consists of the principal part of $\tilde{\mathbf{L}}$ (i.e., the highest-order terms of the operator), and the remaining terms are are the subprincipal terms. These terms arise because the coefficients u and v in the operators Q and Q^* are not constant. It is important to note that the subprincipal terms can be ignored for the purpose of constructing a relaxation scheme.

The system of equations (5) is a higher-order system than the original Euler equations (2). The continuity equation, which is a first-order partial differential equation, has been replaced by a second-order differential equation for the pressure. One might expect that Eq. (5) would require a boundary condition on the pressure in addition to the physical boundary condition of flow tangency at the wall, which is required by Eq. (2). However, at the boundary of the domain, the third equation of (5) takes the form

$$(-u\partial_y v + v\partial_y u + \partial_x p)\hat{n}_x + (u\partial_x v - v\partial_x u + \partial_y p)\hat{n}_y = 0, \qquad (6)$$

where $\hat{n}_x$, $\hat{n}_y$ are the components of the unit normal at the wall. This is simply the equation for the momentum normal to the wall. Because the pressure equation at the wall takes the form of Eq. (6), which in this case may be thought of as a compatibility condition of the governing equations, no auxiliary boundary condition on the pressure is needed.

The operator on the left-hand side of Eq. (5) is upper triangular. Because the pressure satisfies a Poisson equation a conventional relaxation method, such as Gauss-Seidel, can be used to solve it. Upwind differencing of the advection operator in the momentum equations and a downstream ordering of the grid vertices allows marching of the momentum equations. A collective Gauss-Seidel approach is used here, where the vertices are ordered in the flow direction. This is described more fully in the next section.

4 Solution Procedure

The first step in approximating $\tilde{\mathbf{L}}$ is to discretize the Euler equations (2). Unlike the methods of References [3, 15], which use staggered grids, the current approach is vertex-based, where all the unknowns are stored at the vertices of the grid. Discretizations for quadrilateral structured grids and triangular unstructured grids have been coded. A great deal of flexibility in the form of the discrete approximation to the momentum equations is possible with the current method.

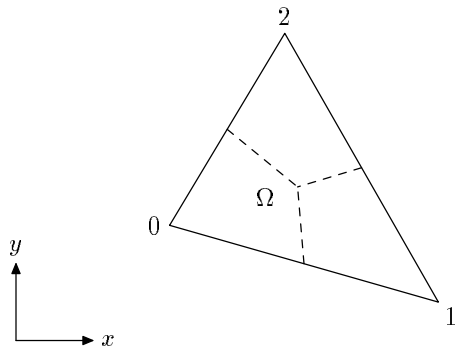

Figure 1: Triangular cell of an unstructured grid

By way of illustration, consider a cell-vertex discretization on an unstructured triangular grid. A typical grid cell Ω is shown in Fig. 1. Cell-averaged gradients of the unknowns are found by using a trapezoidal rule integration around the boundary of Ω. For example, the discrete approximation to the gradient of u on Ω is

$$\begin{aligned} \mathbf{i}\partial_x^h u + \mathbf{j}\partial_y^h u \equiv & \frac{\mathbf{i}}{2A_\Omega}\left(u_0(y_1-y_2)+u_1(y_2-y_0)+u_2(y_0-y_1)\right) \\ & -\frac{\mathbf{j}}{2A_\Omega}\left(u_0(x_1-x_2)+u_1(x_2-x_0)+u_2(x_0-x_1)\right), \end{aligned} \tag{7}$$

where A_Ω is the area of the triangle. The superscript h is used to denote the difference approximation to the corresponding differential operator. Gradients of v and p are obtained likewise. These gradients are used to approximate Eq. (2) on the triangle. An upwind approximation to Q at the vertices of the grid is obtained by distributing the cell-averaged momentum equation residuals to the vertices of each triangle appropriately. The current scheme is not tied to any particular form of the upwind discretization. One choice, which was used to obtain the unstructured grid results presented in Section 5, is the advection scheme of Giles, et al. [5]. In this scheme, the residuals are distributed to the vertices of Ω using the weights

$$W_i = \frac{1}{3}\left(1-\frac{\Delta n_i}{\ell_n}\right), \qquad i=0,1,2,$$

where ℓ_n is the length of the projection of Ω onto the crossflow direction, and Δn_i is the component of the length in the crossflow direction of the edge opposite the i-th vertex. An alternative upwind discretization currently being developed by the authors is based on the multidimensional upwind formulation of Sidilkover [12].

Once the cell-averaged residuals of the continuity and momentum equations have been computed, the projection operator $\mathbf{P}$ is applied to these discrete

equations to obtain the residual for the pressure Poisson equation of Eq. (5). Letting R_{p_i} be the pressure equation residual at vertex i, the application of $\mathbf{P}$ can be written in integral form,

$$R_{p_i} = \sum_{\text{triangles}} \left(\oint_{\partial A_i} \left(\boxed{Q^h u + \partial_x^h p} - u \boxed{\left(\partial_x^h u + \partial_y^h v\right)} \right) dy \right.$$
$$\left. - \left(\boxed{Q^h v + \partial_y^h p} - v \boxed{\left(\partial_x^h u + \partial_y^h v\right)} \right) dx \right) \quad (8)$$

where A_i is the area of the control volume centered on the i-th vertex and the superscript h is used to denote the discrete approximations to the corresponding differential operators on the triangle as before. The summation is over all the triangles adjoining the i-th vertex. The dashed lines in Fig. 1 are the segments of the boundaries of A_0, A_1 and A_2 that lie in cell Ω. The boxed terms in (8) are the cell-averaged residuals of the x and y momentum equations and the continuity equation. The contributions of the cell-averaged residuals on Ω to R_{p_i} are found by evaluating Eq. (8) over the appropriate segment of the boundary of A_i lying in Ω, taking the boxed terms in Eq. (8) to be constant over the cell.

Applying the projection operator $\mathbf{P}$ at the discrete level in this way, rather than starting with the differential equations (5) and discretizing them, has two important advantages. First, the discrete approximation of Eq. (8) at boundary vertices reduces to Eq. (6), automatically providing the correct boundary condition for the pressure. Second, if the momentum and continuity equations are discretized on the triangles in conservation form, it is possible to obtain a fully conservative scheme. This is particularly important for compressible flows with shocks.

The multigrid algorithm uses a sequence of grids $G_K, G_{K-1}, \ldots, G_0$, where G_K is the finest grid and G_0 the coarsest. Call the discrete approximation to the operator $\tilde{\mathbf{L}}$ on the k-th grid $\tilde{\mathbf{L}}_k$, and let $\mathbf{q}_k$ be the solution on that grid. This system has the form $\tilde{\mathbf{L}}_k \mathbf{q}_k = \mathbf{f}_k$, where the entries of $\tilde{\mathbf{L}}_k$ are 3×3 block matrices which operate on the unknowns $(u, v, p)^T$ at each grid vertex. A general iteration scheme is constructed by writing the operator $\tilde{\mathbf{L}}_k$ as $\tilde{\mathbf{L}}_k = \mathbf{M}_k - \mathbf{N}_k$, where the splitting is chosen such that $\mathbf{M}_k$ is easily inverted. Lexicographic Gauss-Seidel is obtained by taking $\mathbf{M}_k$ to be the block lower-triangular matrix resulting from ignoring the terms above the diagonal blocks of $\tilde{\mathbf{L}}_k$. A further simplification is obtained if the diagonal blocks of $\mathbf{M}_k$ contain only those entries corresponding to the principal part of the operator. Because the operator in Eq. (5) is upper triangular the diagonal blocks of $\mathbf{M}_k$ will then be 3×3 upper triangular matrices.

Letting $\mathbf{q}_k^n$ be the n-th iterate of the solution on the k-th grid, the relaxation iteration is

$$\mathbf{M}_k \mathbf{q}_k^{n+1} = \mathbf{f}_k + \mathbf{N}_k \mathbf{q}_k^n.$$

The operator $\tilde{\mathbf{L}}_k$ is nonlinear, so $\mathbf{M}_k$ and $\mathbf{N}_k$ are functions of $\mathbf{q}_k^n$ and $\mathbf{q}_k^{n+1}$. Letting $\delta\mathbf{q}_k^n \equiv \mathbf{q}_k^{n+1} - \mathbf{q}_k^n$, the iteration may be rewritten as

$$\mathbf{M}_k \delta\mathbf{q}_k^n = \mathbf{f}_k - \tilde{\mathbf{L}}_k \mathbf{q}_k^n. \tag{9}$$

Because $\mathbf{M}_k$ is block lower-triangular, $\delta\mathbf{q}_k^n$ is found by forward substitution. At each vertex, a 3×3 upper triangular matrix must be inverted.

If the discrete approximation to the advection operator Q is fully-upwind and the grid points are ordered in the flow direction, then the 3×3 blocks of $\mathbf{N}_k$ will have zeroes in the first two rows. In this case, lexicographic Gauss-Seidel relaxation is equivalent to space-marching of the advection terms. The advected error is effectively eliminated in one relaxation sweep and the convergence rate of the system becomes that of the Poisson equation for the pressure. It is possible to get ideal multigrid convergence rates because each component of the error is treated appropriately.

A straightforward Full Approximation Scheme (FAS) multigrid iteration is applied to the system of equations. Let $\tilde{\mathbf{L}}_{k-1}$ be the coarse grid operator, I_{k-1}^k be the fine-to-coarse grid restriction operator, and I_k^{k-1} be the coarse-to-fine grid prolongation operator. If $\hat{\mathbf{q}}_k$ is the current solution on grid k, the residual on this grid is $\mathbf{r}_k \equiv \mathbf{f}_k - \tilde{\mathbf{L}}_k \hat{\mathbf{q}}_k$. This leads to the coarse-grid equation

$$\tilde{\mathbf{L}}_{k-1} \hat{\mathbf{q}}_{k-1} = \mathbf{f}_{k-1} = I_{k-1}^k \mathbf{r}_k + \tilde{\mathbf{L}}_{k-1} \left(I_{k-1}^k \hat{\mathbf{q}}_k \right). \tag{10}$$

After solving the coarse-grid equation for $\mathbf{q}_{k-1}$, the fine-grid solution is corrected by

$$\hat{\mathbf{q}}_k^{\text{new}} \leftarrow \hat{\mathbf{q}}_k + I_k^{k-1} \left(\hat{\mathbf{q}}_{k-1} - I_{k-1}^k \hat{\mathbf{q}}_k \right). \tag{11}$$

Equation (10) is solved by applying the same relaxation procedure that is used to solve the fine-grid equation. Multigrid is applied recursively to the coarse-grid equation. On the coarsest grid, many relaxation sweeps are performed to insure that the equation is solved completely. A conventional V–cycle or W–cycle is used.

5 Results

Both unstructured grid and structured grid flow solvers based on the theory in Sections 3 and 4 have been written. These codes are described in Reference [11], where extensive solutions are presented. Results illustrating the efficiency of the scheme are presented here.

Solutions for incompressible, inviscid flow in a channel have been obtained with both solvers. The channel geometry and boundary conditions are shown in Fig. 2. The shape of the lower wall between $0 \le x \le 1$ is $y(x) = \tau \sin^2 \pi x$. For the computations shown here, the thickness ratio τ is 0.05. The flow angle

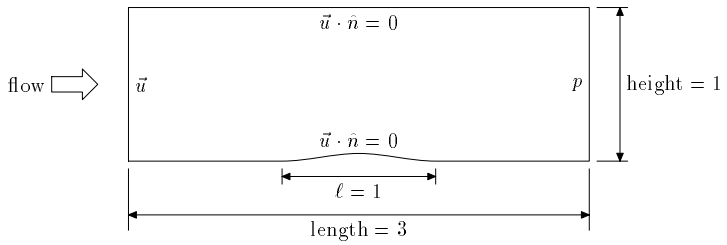

Figure 2: Channel geometry.

and total pressure are specified at the inlet and the pressure is specified at the outlet. The flow tangency condition $\vec{u} \cdot \hat{n} = 0$ is enforced at the upper and lower walls of the channel. Solutions were obtained on quasi-uniform quadrilateral grids. A simple shearing transformation was used in the center part of the channel to obtain boundary conforming grids. For the unstructured grid solver, the grids were triangulated by dividing each quadrilateral cell along a diagonal. A series of nested coarse grids was obtained by coarsening the fine grids by a factor of two in each coordinate direction. In all cases shown below, the coarsest grid was 7×3 vertices. Lexicographic Gauss-Seidel relaxation was used, with the grid vertices ordered from the lower-left to the upper-right of the channel. This resulted in downstream relaxation of the momentum equations. A $V(2,1)$ multigrid cycle was used; that is, two relaxation sweeps were performed on each grid before restricting to the coarse grid, and one relaxation sweep was performed after the coarse-grid correction was added to the fine-grid solution.

The computed pressure on a grid of 97×33 vertices is shown in Fig. 3 for the unstructured grid flow solver and in Fig. 4 for the structured grid solver. Comparisons of convergence rates for different grid densities are shown in Figs. 5 and 6 for the unstructured and structured grid flow solvers, respectively. The L_1 norm of the pressure equation residual is shown; the momentum equation residuals show the same behavior. The finest grid used for each flow solver contained 385×129 vertices, with a total of 7 grid levels.

The convergence rate of the unstructured grid solver on the finest grid is approximately 0.190 residual reduction per multigrid cycle. The structured grid results are slightly better at 0.167 per cycle. These rates are comparable to the ideal rate of 0.125 per cycle for the Poisson equation. The better performance of the structured grid solver is most likely because of better restriction and prolongation operators; the unstructured flow solver performs bilinear interpolation using only the locations of a fine-grid vertex and the three vertices of the coarse-grid cell containing that vertex. What is most important is that the figures show nearly ideal multigrid convergence rates, independent of the grid spacing. This shows that convergence is achieved in order n operations.

For complex geometries it may not be practical to generate a series of nested unstructured grids, and the performance of the multigrid solver may be expected

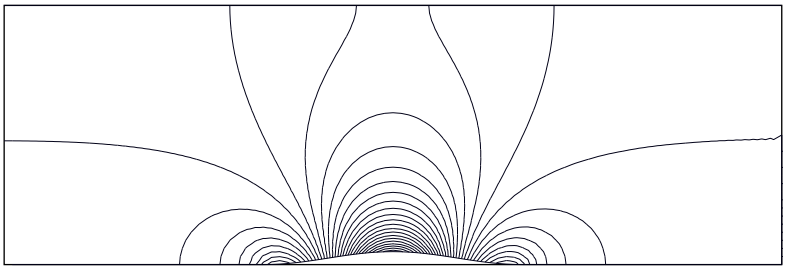

Figure 3: Pressure, contour increment $\Delta p = 0.01$, for an unstructured grid of 97×33 vertices.

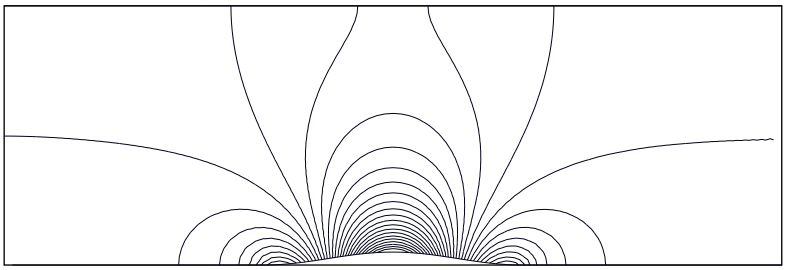

Figure 4: Pressure, contour increment $\Delta p = 0.01$, for a structured grid of 97×33 vertices.

to deteriorate. To show the robustness of the current method, the triangular grid solver was run for a series of non-nested coarse grids. These were generated by randomly perturbing the locations of the vertices on each of the nested grids independently. The perturbed 49×17 grid is shown in Fig. 7. The computed pressure on a perturbed 97×33 fine grid with 5 grid levels is shown in Fig. 8 and the convergence rate is shown in Fig. 9. The pressure contours are very smooth, showing no sign of the lack of grid smoothness. The asymptotic convergence rate has deteriorated to a still-respectable 0.24 per cycle.

Solutions for nonlifting flow over a symmetric Kármán-Trefftz airfoil have been obtained with the structured grid solver. A fine O-grid of 385×193 vertices was generated from a conformal mapping, and the coarse grids are nested by recursively eliminating every other vertex in each coordinate direction. The grid spacing was chosen to obtain unit aspect ratio grid cells. The outer boundary is approximately 13 chord lengths from the airfoil. Far-field boundary conditions are given by the analytic solution. At inflow points along the outer boundary the total pressure and flow inclination angle are specified. For outflow points the pressure is specified. On the airfoil surface the tangency condition is enforced.

To obtain ideal multigrid convergence rates, it is necessary to sort the vertices in a downstream order so that the advection terms in the momentum equations are marched. This is easily done here by relaxing along the radial grid lines from the outer boundary to the airfoil surface over the forward half of

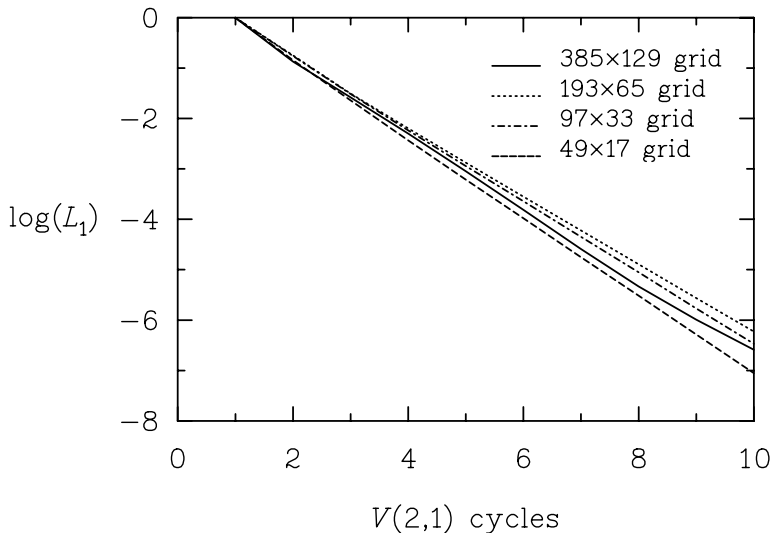

Figure 5: Comparison of convergence rates on unstructured grids.

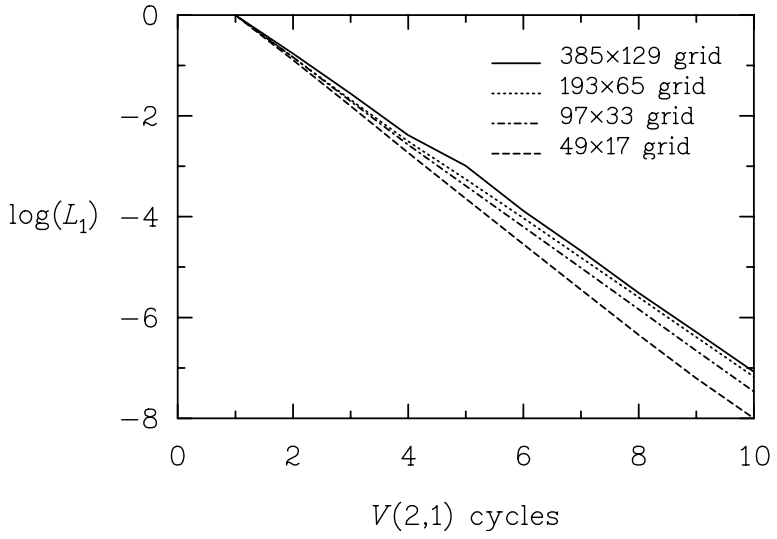

Figure 6: Comparison of convergence rates on structured grids.

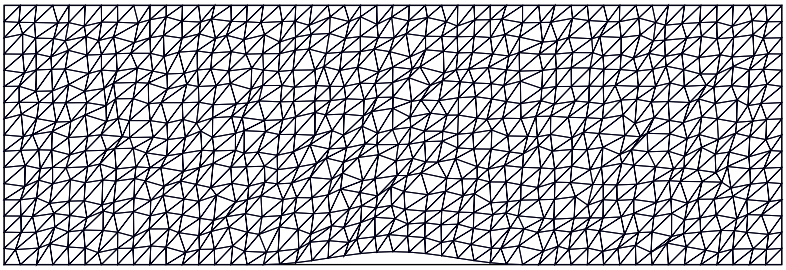

Figure 7: Grid generated by perturbing the vertices of the 49×17 grid.

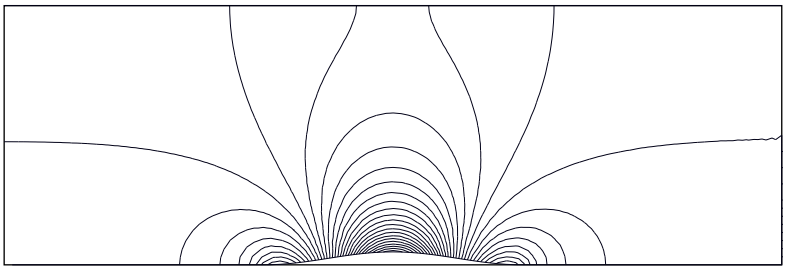

Figure 8: Pressure, contour increment $\Delta p = 0.01$, randomly perturbed unstructured grid of 97×33 vertices.

the domain, and from the airfoil surface to the outer boundary over the latter half of the domain. For each case run, the coarsest grid consisted of 13×7 vertices.

Comparisons between computed and analytic surface pressure coefficients for nonlifting flow around the Kármán-Trefftz airfoil are shown in Fig. 10. A $W(2,1)$ multigrid cycle was used for these computations. The computed solution agrees very well with the analytic solution, except for the recompression at the trailing edge. Note that there is no clustering of the grid in this region, which exacerbates the problem.

A comparison of the convergence rates of the pressure equation residual for three grid densities is shown in Fig. 11. A slight deterioration of the convergence rate with increasing grid refinement is observed: on the 385×193 grid, the rate is 0.153 per cycle. Nevertheless, as with the channel flow results, the convergence rates are very nearly grid independent, and are very close to the ideal rate of 0.125 per cycle.

A summary of the convergence rate on the finest grids is presented in Table 1. Two sets of results are shown: the convergence rate per multigrid cycle, and the convergence rate per work unit. For the purposes of the discussion a work unit (WU) is taken to be one Gauss-Seidel relaxation sweep on the finest grid. This is essentially the cost of one residual evaluation on the finest grid. The actual convergence rates are compared to the ideal convergence rates, which are computed as follows. Let μ be the smoothing rate of the relaxation method. For

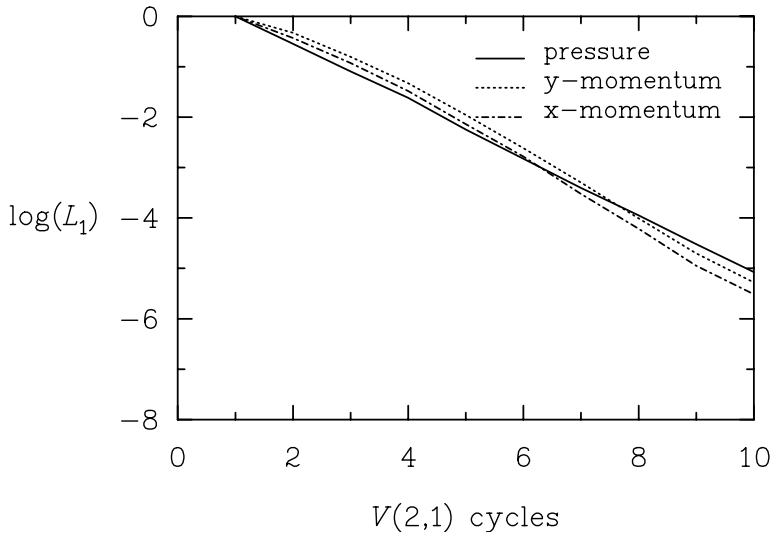

Figure 9: Convergence rate, randomly perturbed unstructured grid of 97×33 vertices, 5 grid levels, $V(2,1)$ cycle.

a $V(m,n)$ or $W(m,n)$ cycle, the ideal convergence rate is μ^{m+n}. Lexicographic Gauss-Seidel for the Poisson equation has a smoothing rate $\mu = 0.5$. This gives an ideal convergence rate of $0.5^3 = 0.125$ for a $V(2,1)$ or a $W(2,1)$ cycle. To compute the convergence rate per work unit we use the following formula. For each cycle, there are a total of $m+n$ fine-grid relaxation sweeps. Examination of Eq. 10 shows that the fine-to-coarse grid restriction requires one residual evaluation on the fine grid and an additional residual evaluation on the coarse grid. The coarse grid residual evaluation is $1/4$ the cost of a fine grid residual evaluation. Because most of the cost of a relaxation sweep is in the evaluation of the residual, we have that each cycle requires a total of $(m+n+1+1/4)$ work units on the finest grid. The cost of interpolating the residuals and solutions between grid levels is neglected.

For a $V(m,n)$-cycle, we have that

$$\begin{aligned}\frac{\text{WU}}{V\text{–cycle}} &\approx (m+n+1+\frac{1}{4})(1+\frac{1}{4}+\frac{1}{16}+\cdots) \\ &= \frac{4}{3}(m+n+\frac{5}{4}).\end{aligned}$$

Because a W-cycle involves two coarse-grid solutions per cycle, we have

$$\begin{aligned}\frac{\text{WU}}{W\text{–cycle}} &\approx (m+n+1+\frac{1}{4})(1+\frac{1}{2}+\frac{1}{4}+\cdots) \\ &= 2(m+n+\frac{5}{4}).\end{aligned}$$

These numbers yield ideal convergences rates of $\mu^{3(m+n)/(4(m+n+5/4))}$ per work unit for a V–cycle and $\mu^{(m+n)/(2(m+n+5/4))}$ per work unit for a W–cycle.

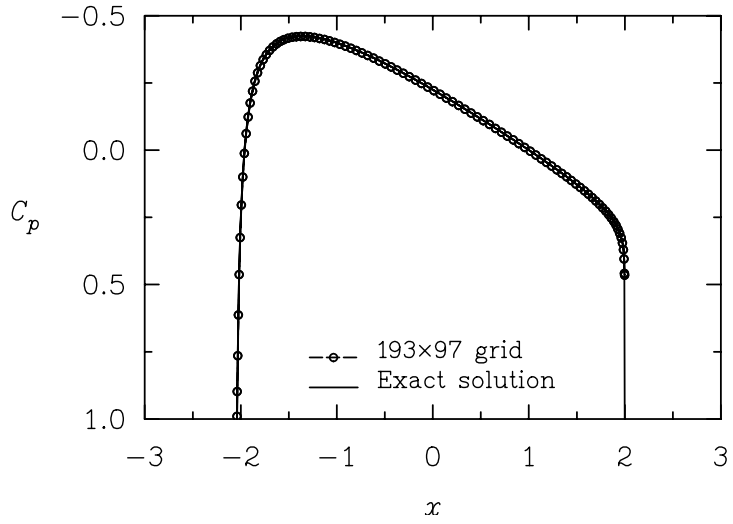

Figure 10: Surface pressure coefficient, nonlifting Kármán-Trefftz airfoil, 193 × 97 grid.

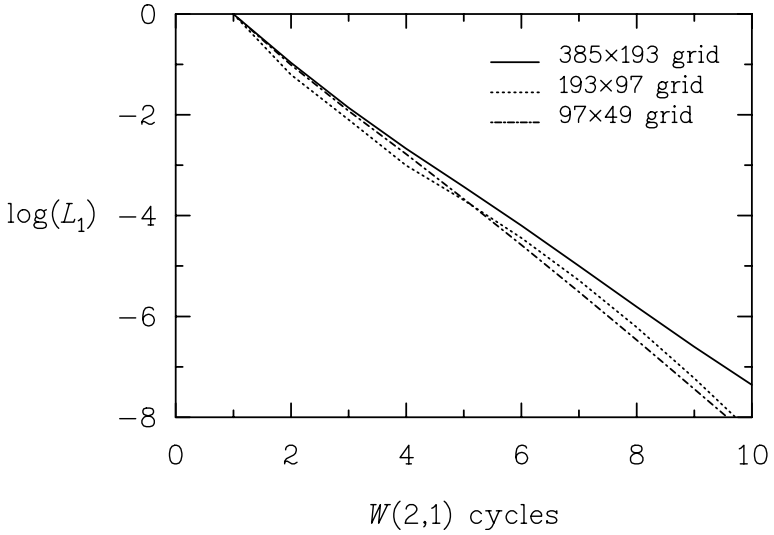

Figure 11: Comparison of convergence rates for nonlifting Kármán-Trefftz airfoil.

Case	Cycle	Convergence Rate			
		per cycle		per work unit	
		ideal	actual	ideal	actual
channel, 385×129, unstructured	$V(2,1)$	0.125	0.190	0.693	0.746
channel, 385×129, structured	$V(2,1)$	0.125	0.167	0.693	0.729
airfoil, 385×193, structured	$W(2,1)$	0.125	0.153	0.783	0.802

Table 1: Summary of convergence rates for multigrid solver on finest grids for channel and airfoil flows, with a comparison to the ideal rates.

The $V(2,1)$ cycle is seen to require $5\frac{2}{3}$ WU per cycle. The $W(2,1)$ is 50% more expensive, requiring $8\frac{1}{2}$ WU per cycle. By way of comparison, one $V(2,1)$ cycle is only slightly more work than a single time step of a 5-stage Runge-Kutta scheme on the finest grid. The ideal convergence rates for lexicographic Gauss-Seidel is of 0.693 per WU for a $V(2,1)$ cycle and 0.783 per WU for a $W(2,1)$ cycle. The actual rates shown in Table 1 are seen to be very close to ideal.

The convergence rates in Table 1 can be used to estimate the work required to obtain a solution to the level of the discretization error on the fine grid. Let p be the order of approximation of the discrete operator and let h_k be the grid spacing parameter on the k-th grid. An initial guess to the solution on the fine grid G_K is obtained by interpolating a solution computed on grid G_{K-1}. Assume that the solution on G_{K-1} has been obtained to the level of the discretization error $\tau_{K-1} = O(h_{K-1}^p)$ on that grid. The multigrid cycle is used to reduce the error from τ_{K-1} to τ_K. Letting μ_W be the convergence rate per work unit, the amount of work W_K required to get the solution on G_K from the initial solution on G_{K-1} is

$$W_K = \frac{1}{\log \mu_W} \log \left(\frac{\tau_K}{\tau_{K-1}} \right) = \frac{p}{\log \mu_W} \log \left(\frac{h_K}{h_{K-1}} \right).$$

A Full Multigrid (FMG) cycle starts with a solution on the coarsest grid, G_0, and recursively generates improved solutions on the finer grids using the strategy above.

For the nested grids considered here, the grid spacing parameters are related by $h_{k-1} = 2h_k$, and the amount of work on each grid is related by $W_{k-1} = W_k/4$. The discretization is second-order accurate, i.e., $p = 2$. This gives us the estimate for the total work to obtain a solution accurate to τ_K to be

$$\begin{aligned} W_{\text{total}} &= \frac{p}{\log \mu_W} \log \left(\frac{1}{2} \right) \left(1 + \frac{1}{4} + \frac{1}{16} + \cdots \right) \\ &= -\frac{8}{3} \frac{\log 2}{\log \mu_W} \end{aligned} \tag{12}$$

Using the values in the last column of Table 1 for μ_W, we see that channel flow solutions can be obtained to the level of discretization error in approximately 6.4 WU using a FMG cycle. Airfoil solutions can be obtained in about 8.3 WU. These estimates are generally low, and in fact are less than work of a single FMG cycle (7.6 and 11.3 WU for the channel and airfoil cases, respectively). The work computed using Eq. (12) also does not account for the introduction of short-wavelength errors in the interpolation of the coarse grid solutions to the fine grids. Nevertheless, Eq. (12) is a useful guide to the expected performance of the multigrid scheme.

6 Extension to Compressible Flow

The scheme presented here has a straightforward extension to the compressible Euler equation. In primitive variables the equations are

$$\mathbf{Lq} = \begin{pmatrix} Q & 0 & 0 & 0 \\ 0 & \rho Q & 0 & \partial_x \\ 0 & 0 & \rho Q & \partial_y \\ 0 & \rho\partial_x & \rho\partial_y & \frac{1}{c^2}Q \end{pmatrix} \begin{pmatrix} s \\ u \\ v \\ p \end{pmatrix} = 0, \tag{13}$$

where ρ is the density, c is the speed of sound, and s is the entropy. The projection operator $\mathbf{P}$ for this system is

$$\mathbf{P} = \begin{pmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & \partial_x & \partial_y & Q^* \end{pmatrix}. \tag{14}$$

The operators Q and Q^* are defined as in Eqs. (1) and (3). Applying this to Eq. (13) and ignoring the subprincipal terms as before yields

$$\mathbf{PLq} = \begin{pmatrix} Q & 0 & 0 & 0 \\ 0 & \rho Q & 0 & \partial_x \\ 0 & 0 & \rho Q & \partial_y \\ 0 & 0 & 0 & \Delta - M^2\partial_s^2 \end{pmatrix} \begin{pmatrix} s \\ u \\ v \\ p \end{pmatrix} + \text{s.p.t.}, \tag{15}$$

where M is the Mach number, ∂_s is the partial derivative in the streamwise direction, and "s.p.t." are the subprincipal terms.

The most significant difference between the compressible and the incompressible equations is that a Prandtl-Glauert-like operator acts on the pressure. Note that this system approaches the system for the incompressible equations in the limit of vanishing Mach number. For subsonic flow the compressible equations can be solved by the same relaxation scheme as the incompressible equations. Unlike time marching methods, the convergence rate will not deteriorate as the Mach number approaches zero.

The appearance of the Prandtl-Glauert operator in the pressure equation is significant. In effect, the problem of solving the pressure equation is no different than that of solving the full potential equation. This is a two-edged sword. On the one hand, the difficulties of relaxing the pressure equation in the transonic case are precisely those of relaxing the transonic full potential equation. This is a fundamental difficulty which is faced by any method that works directly on the steady flow equation. On the other hand, one can expect that the wealth of experience in solving potential flows can be directly applied to the current scheme for the compressible Euler equations. The treatment of the advection terms is not essentially different from the incompressible case.

7 Conclusions

Murman and Cole introduced type-dependent differencing and relaxation methods into computational aerodynamics; practical and efficient methods for solving nonlinear flow equations were the result. In subsequent years, the emphasis has shifted toward iterative methods based on the unsteady equations. In this paper, it has been shown that great improvements in the efficiency of flow solvers can be achieved by changing the point of view from the unsteady to the steady equations. As Murman and Cole introduced type-dependent differencing, so the current method relies on a discretization which distinguishes between the elliptic and hyperbolic parts of the system. As Murman and Cole used relaxation to solve the steady equations, so the current method applies relaxation with multigrid to the steady equations. This approach yields textbook multigrid efficiency for the steady Euler equations. It is a particularly simple approach; conventional finite-difference or finite-volume discretizations of the governing equations may be used, allowing flexibility in the choice of the underlying numerical method. Unlike time-marching approaches, but like potential flow methods, the convergence rate of the method does not degrade for low-speed flows, and the correct incompressible limit is recovered. Finally, this method can be applied to incompressible, viscous flow following the ideas of Sidilkover and Ascher [13].

There remains a great deal of work to be done before the full Reynolds-averaged Navier-Stokes equations can be solved as efficiently as the simple problems shown here. The present work only addresses one particular, but nevertheless important, aspect of the problem, namely the appropriate discretization of the governing equations. If textbook multigrid efficiency is to be achieved for compressible, viscous flow, it will likely require an approach along the general outlines presented here.

Acknowledgments

The work presented in this paper would not have started without the the advocacy and encouragement of Jerry South. The authors thank him for his interest, enthusiasm, and encouragement during the course of this research.

References

[1] Anderson, W. K., Thomas, J. L., and Whitfield, D. L., "Three-Dimensional Multigrid Algorithms for the Flux-Split Euler Equations," NASA Technical Paper 2829, 1988.

[2] Brandt, A., "Multigrid Techniques: 1984 Guide with Applications to Fluid Dynamics," GMD-Studie 85, GMD-FIT, 1985.

[3] Brandt, A., and Yavneh, I., "Accelerated Multigrid Convergence and High-Reynolds Recirculating Flows," *SIAM J. Sci. Statist. Comput.*, vol. 14, no. 3, pp. 607–626, 1993.

[4] Caughey, D. A., "The Computation of Transonic Potential Flows," *Ann. Rev. Fluid Mech.*, vol. 14, pp. 261–283, 1982.

[5] Giles, M., Anderson, W. K., and Roberts, T. W., "Upwind Control Volumes: A New Upwind Approach," *AIAA Paper* 90-0104, 1990.

[6] Jameson, A., "Iterative Solution of Transonic Flows over Airfoils and Wings, Including Flows at Mach 1", *Comm. Pure Appl. Math.*, vol. 27, pp. 283–309, 1974.

[7] Jameson, A., "Solution of the Euler Equations for Two Dimensional Transonic Flow by a Multigrid Method," *Appl. Math. Comput.*, vol. 13, nos. 3 and 4, pp. 327–355, 1983.

[8] Mulder, W., "Multigrid Relaxation for the Euler Equations," *J. Comput. Phys.*, vol. 60, no. 2, pp. 235–252, 1985.

[9] Murman, E. M., Cole, J. D., "Calculation of Plane, Steady Transonic Flows," *AIAA J.*, vol. 9, no. 1, pp. 114-121, 1971.

[10] Pierce, N. A., Giles, M., Jameson, A., Martinelli, L., "Accelerating Three Dimensional Navier-Stokes Calculations," *AIAA Paper* 97-1953, 1997.

[11] Roberts, T. W., Sidilkover, D., Swanson, R. C., "Textbook Multigrid Efficiency for the Steady Euler Equations," *AIAA Paper* 97-1949, 1997.

[12] Sidilkover, D., "A Genuinely Multidimensional Upwind Scheme and Efficient Multigrid Solver for the Compressible Euler Equations," ICASE Report 94–84, 1994.

[13] Sidilkover, D., and Ascher, U. M., "A Multigrid Solver for the Steady State Navier-Stokes Equations using the Pressure-Poisson Formulation," *Comp. Appl. Math.* vol. 14, no. 1, pp. 21–35, 1995.

[14] Ta'asan, S., "Canonical Forms of Multidimensional Steady Inviscid Flow," ICASE Report 93–34, 1993.

[15] Ta'asan, S., "Canonical-Variables Multigrid Method for Steady-State Euler Equations," ICASE Report 94–14, 1994.

[16] Vatsa, V., Wedan, B. W., "Development of a Multigrid Code for 3-D Navier-Stokes Equations and its Application to a Grid-Refinement Study," *Computers & Fluids*, vol. 18, no. 4, pp. 391–403, 1990.

[17] Warren, G. P., and Roberts, T. W., "Multigrid Properties of Upwind-Biased Data Reconstructions," *Sixth Copper Mountain Conference on Multigrid Methods,* NASA Conference Publication 3224, Part 2, 1993.