PERGAMON

Computers & Fluids 28 (1999) 427–442

computers & fluids

An algorithm for ideal multigrid convergence for the steady Euler equations

Thomas W. Roberts[a,*], R.C. Swanson[a], David Sidilkover[b]

[a]*Mail Stop 128, NASA Langley Research Center, Hampton, VA 23681-2199, USA*
[b]*Institute for Computer Applications in Science and Engineering, Mail Stop 132C, NASA Langley Research Center, Hampton, VA 23681-2199, USA*

Received 20 November 1997; accepted 18 May 1998

Abstract

A fast multigrid solver for the steady incompressible Euler equations is presented. Unlike time-marching schemes this approach uses relaxation of the steady equations. Application of this method results in a discretization that correctly distinguishes between the advection and elliptic parts of the operator, allowing efficient smoothers to be constructed. Solvers for both unstructured triangular grids and structured quadrilateral grids have been written. Flows in two-dimensional channels and over airfoils have been computed. Using Gauss–Seidel relaxation with the grid vertices ordered in the flow direction, ideal multigrid convergence rates of nearly one order-of-magnitude residual reduction per multigrid cycle are achieved, independent of the grid spacing. This approach also may be applied to the compressible Euler equations and the incompressible Navier–Stokes equations.

1. Introduction

Over the past three decades, computational fluid dynamics has become a powerful and widely used tool for aerodynamic analysis and design. This began primarily with the introduction of methods for solving transonic potential flow, and has progressed to methods for solving the Euler and Reynolds-averaged Navier–Stokes equations [1]. Today, viscous solutions for real aircraft geometries are routinely possible. At the same time, the utility of the numerical methods is restricted by their slow asymptotic convergence rates. There has always

* Corresponding author. Fax: +1-757-864-8816; e-mail: t.w.roberts@larc.nasa.gov.

0045-7930/99/$ - see front matter
PII: S0045-7930(98)00040-1

been, and continues to be, a critical need to improve the convergence rates of computational fluid dynamics codes in order to increase throughput and their overall utility.

Multigrid [2] has long been recognized as the most promising class of methods for obtaining fast convergence rates for iterative solutions of partial differential equations. For purely elliptic problems, multigrid methods can be shown to be $O(n)$: that is, the work and memory required to obtain a solution is directly proportional to the number of unknowns n in the problem. This has led to many efforts over the years to apply multigrid methods to the non-elliptic problems of fluid dynamics in the hope that similar convergence rates can be achieved. A pioneering effort was made by South and Brandt [3] for the transonic small perturbation equation. This work demonstrated both the promise of multigrid for non-elliptic and non-linear problems, as well as some of the difficulties that would have to be overcome to achieve ideal, or "textbook" convergence rates. Since that time, multigrid has been widely applied to methods for the Euler and Navier–Stokes equations.

The application of multigrid to the Euler and Reynolds-averaged Navier–Stokes equations generally is based on a discretization of the unsteady equations using some temporal integrator as the smoother, combined with a full-approximation scheme (FAS) multigrid iteration. These schemes fall into two classes. One approach is to use upwind-differencing and implicit time integration as the smoother [4–6]. The more common approach is one originally proposed by Jameson [7]. Starting with the unsteady equations, a finite-volume spatial discretization with explicit artificial viscosity is combined with a Runge–Kutta time integration as a smoother. This approach has also been successfully applied to the Reynolds-averaged Navier–Stokes equations [8]. However, 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. Recently, significant improvements have been demonstrated by Pierce et al. [9]. Nevertheless, there is clearly a need to develop substantially more efficient multigrid solvers.

According to Brandt [2], one of the major obstacles to achieving better multigrid performance for advection dominated flows is that the coarse grid provides only a fraction of the needed correction for smooth error components. This 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. For instance, advection can be treated by space marching, while elliptic factors can be treated by multigrid. The efficiency of such an algorithm will be essentially identical to that of the solver for the elliptic factor only, and thereby attain textbook multigrid efficiency. 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 for the incompressible Navier–Stokes equations [10]. 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 [11] 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 hyperbolic partition [12]. This form of the Euler equations is essentially Crocco's relation. Ta'asan uses this form of the equations to guide the discretization. A staggered grid is used, with different variables residing at cell, vertex and edge centers. In Ref. [11] 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.

In this paper, an alternative to distributive relaxation and to Ta'asan's canonical variable decomposition is presented. It is a generalization of the approach of Sidilkover and Ascher [13], and has been applied to the incompressible Euler equations by the authors [14]. The theoretical background to this method is presented in the paper by Sidilkover in this volume [15]. A conventional vertex-based finite-volume or finite-difference discretization of the primitive variables is used, avoiding the need for staggered grids. 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 with distributive relaxation and the canonical variables approaches, this method is extremely simple.

2. Mathematical formulation

The incompressible Euler equations in primitive variables are

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

when 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

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

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 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 Eq. (5) is a higher-order system than the original Euler Eq. (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 Eq. (5) takes the form

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

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 Section 4.

3. Discretization of the equations

The first step in approximating $\tilde{\mathbf{L}}$ is to discretize the Euler Eq. (2). Unlike the methods of Refs. [10, 11], which use staggered grids, the current approach is vertex-based, where all the unknowns are stored at the vertices of the grid. A great deal of flexibility in the form of the discrete approximation to the momentum equations is possible with the current method. However there are some subtle, but important, differences in the way the Poisson equation for the pressure is discretized on triangular or quadrilateral grids. To understand these differences, both schemes are described in turn below.

The unstructured triangular grid scheme is based on a cell-vertex discretization of the equations. Consider a typical grid cell Ω as 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

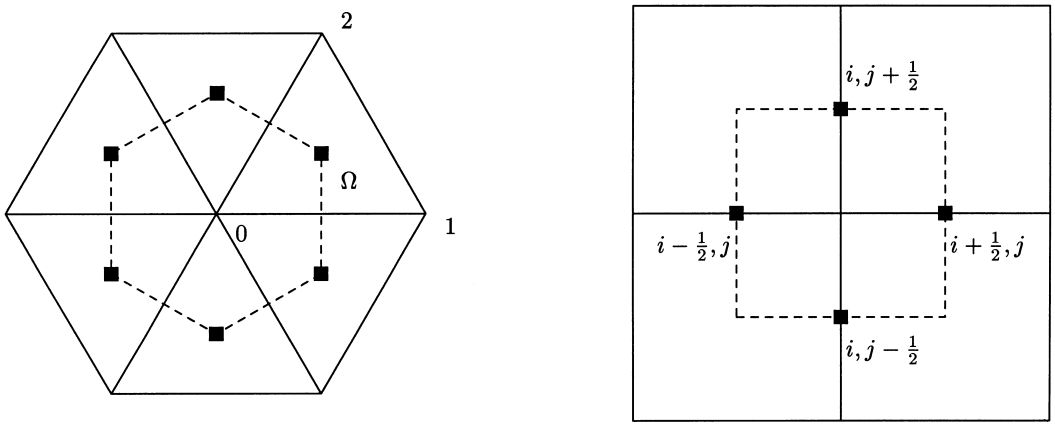

Fig. 1. Primary and dual cells on triangular and quadrilateral grids.

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

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. [16]. In this scheme, the residuals are distributed to the vertices of Ω using the weights

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

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 ith vertex. An alternative upwind discretization currently being developed by the authors is based on the multidimensional upwind formulation of Sidilkover [17].

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_0} be the pressure equation residual at vertex 0, the application of $\mathbf{P}$ can be written in integral form,

$$R_{p_0} = \oint_{\partial A_0} \left(\boxed{Q^h u + \partial_x^h p} - u \boxed{\left(\partial_x^h u + \partial_y^h v \right)} \right) \mathrm{d}y$$
$$- \left(\boxed{Q^h v + \partial_y^h p} - v \boxed{\left(\partial_x^h u + \partial_y^h v \right)} \right) \mathrm{d}x \tag{8}$$

where A_0 is the area of the control volume centered on vertex 0. This control volume is the area bounded by the dual grid cell, shown as the dashed lines in Fig. 1. The superscript h is used to denote the discrete approximations to the corresponding differential operators on the primary cell triangles, as before. The boxed terms in Eq. (8) are the residuals of the x and y momentum equations and the continuity equation averaged over the primary grid cells, as given by Eq. (7). The contributions of these residuals in R_{p_0} are found by taking the boxed terms in Eq. (8) to be constant over the primary grid cells when evaluating Eq. (8) over the segment of the boundary of A_0 lying in each cell.

For a structured quadrilateral grid, the discretization of the momentum equations is straightforward. At each vertex upwind differencing is used for the advection operator Q and central differencing for the pressure. As with triangular grids, the exact choice of the upwind discretization is not critical. In the results shown below, conventional second-order one-sided difference formulas were used for the advection terms. The discretization of the Poisson equation for the pressure is done by first discretizing the boxed terms of Eq. (8) on the edges of the dual grid cell, shown as the dashed lines on the left of Fig. 1. The integral (8) is then evaluated over the boundary of the dual grid cell. Unlike a triangular grid, where trapezoidal rule integration over the primary grid cells is used to get cell-averaged gradients, the gradients on a quadrilateral grid must be evaluated on the dual grid edge centers. This distinction is indicated in Fig. 1 by the solid squares, which show the evaluation points of the gradients for the two cases. If a conventional cell-vertex type of discretization is used to get the gradients at the primary grid cell centers, as for the triangular grid, the resulting stencil for the pressure equation is not h-elliptic, and admits a checkerboard error mode.

To evaluate the gradients of pressure p at the dual grid face $(i - 1/2, j)$ in Fig. 1, the partial derivatives are written as

$$p_x = p_\xi \xi_x + p_\eta \eta_x \tag{9a}$$

$$p_y = p_\xi \xi_y + p_\eta \eta_y \tag{9b}$$

where ξ and η are the generalized coordinates corresponding to the i and j directions on the grid, respectively. The derivatives $\partial_\xi p$ and $\partial_\eta p$ are approximated on the face center by

$$\partial_\xi^h p\big|_{i-1/2,j} = p_{i,j} - p_{i-1,j}, \tag{10a}$$

$$\partial_\eta^h p\big|_{i-1/2,j} = \frac{1}{4}(p_{i,j+1} + p_{i-1,j+1} - p_{i,j-1} - p_{i-1,j-1}) \tag{10b}$$

with similar expressions for the faces $(i + 1/2, j)$, $(i, j + 1/2)$, and $(i, j - 1/2)$. The gradients of u and v are found the same way. The grid metric terms ξ_x, ξ_y, η_x and η_y are also evaluated on

the dual grid face centers. These expressions are used in the boxed terms in Eq. (8). The integral (8) is then evaluated to get the pressure equation residual at the vertex (i, j), taking the boxed terms to be constant over each face of the dual grid cell. For a uniform cartesian grid, the principal part of the resulting stencil is a conventional five-point approximation to the Laplacian operating on the pressure.

Applying the projection operator $\mathbf{P}$ at the discrete level in this way, rather than starting with the differential Eq. (5) and discretizing them directly, 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 in conservation form, it is possible to obtain a fully conservative scheme. This will be particularly important for extensions of the present scheme to compressible flows with shocks.

4. Solution procedure

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 kth 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 $(u, v, p)^{\mathrm{T}}$ 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 $\mathbf{q}^n{}_k$ be the nth iterate of the solution on the kth 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 $\delta\mathbf{q}^n{}_k \equiv \mathbf{q}_k^{n+1} - \mathbf{q}_k^n$, the iteration may be rewritten as

$$\mathbf{M}_k\delta\mathbf{q}_k^n = \mathbf{f}_k - \tilde{\mathbf{L}}_k\mathbf{q}_k^n \tag{11}$$

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}_{k-1} be the fine-to-coarse grid restriction operator, and I^{k-1}_{k} 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

$$\tilde{\mathbf{L}}_{k-1}\hat{\mathbf{q}}_{k-1} = \mathbf{f}_{k-1} = I^{k}_{k-1}\mathbf{r}_k + \tilde{\mathbf{L}}_{k-1}\left(I^{k}_{k-1}\hat{\mathbf{q}}_k\right) \tag{12}$$

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

$$\hat{\mathbf{q}}^{\text{new}}_{k} \leftarrow \hat{\mathbf{q}}_k + I^{k-1}_{k}\left(\hat{\mathbf{q}}_{k-1} - I^{k}_{k-1}\hat{\mathbf{q}}_k\right) \tag{13}$$

Eq. (12) 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 2–4 have been written. Many solutions obtained with these codes were presented in Ref. [14]. Only results chosen to illustrate 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 $0 \leqslant x \leqslant 1$ is $y(x) = \tau \sin^2 \pi x$. For the computations shown here, the thickness ratio τ is 0.05. The flow angle and total pressure are specified at the inlet and the pressure is specified at the outlet. The flow tangency condition $\mathbf{u}\cdot\hat{n} = 0$ 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

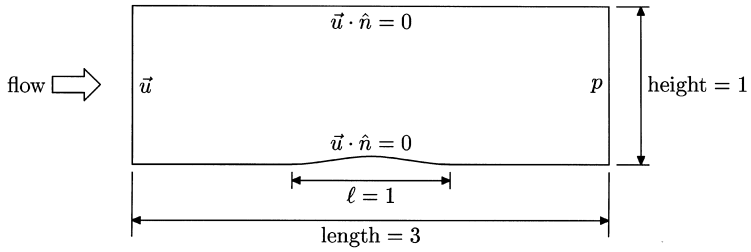

Fig. 2. Channel geometry.

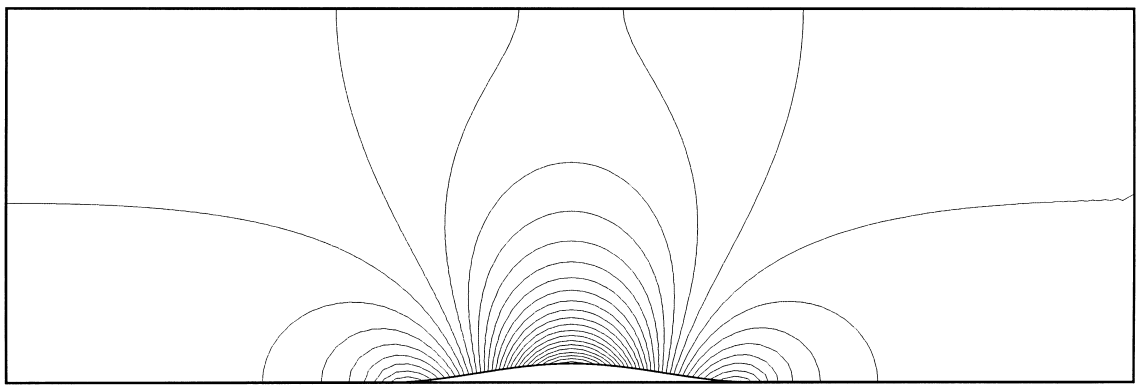

Fig. 3. Pressure, contour increment $\Delta p = 0.01$, for an unstructured grid of 97 × 33 vertices.

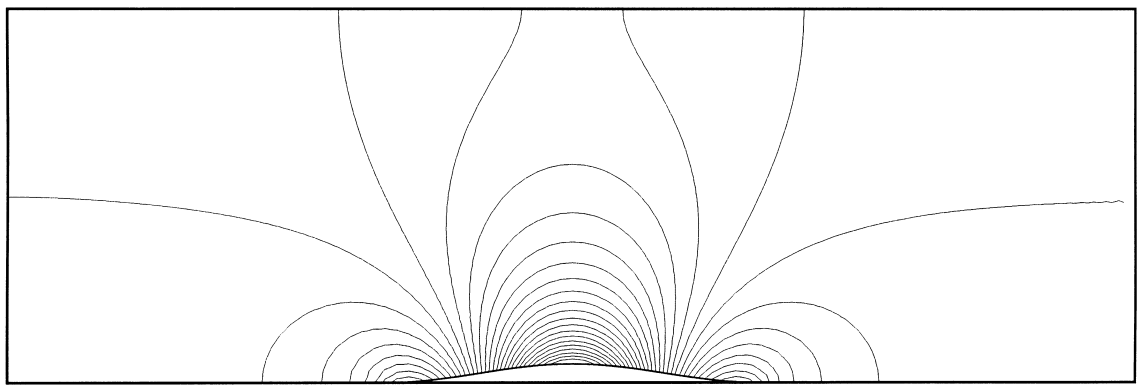

Fig. 4. Pressure, contour increment $\Delta p = 0.01$, for an structured grid of 97 × 33 vertices.

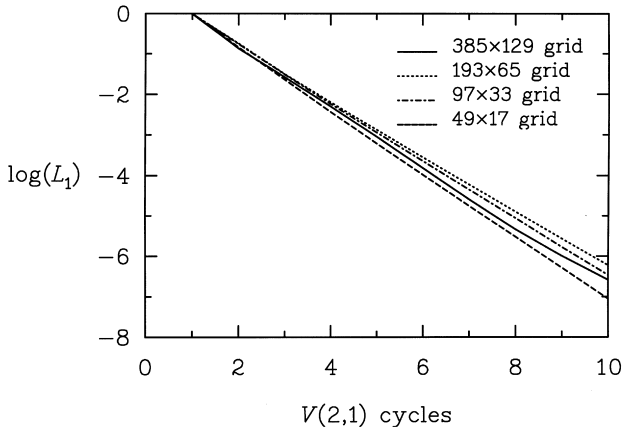

Fig. 5. Comparison of convergence rates on unstructured grids.

momentum equation residuals show the same behavior. The finest grid used for each flow solver contained 385×129 vertices, with a total of seven 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.

Solutions for flow over a symmetric Kármán–Trefftz airfoil have been obtained with the structured grid solver. A conformal mapping of a circular cylinder was used to generate the airfoil. The center of the cylinder was placed at the point (–0.1, 0) in the circle plane, yielding

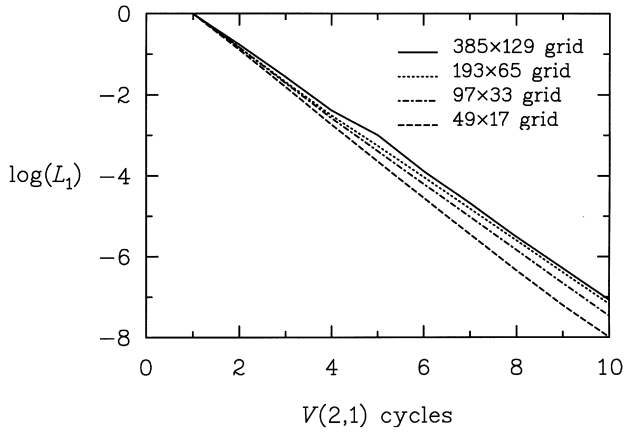

Fig. 6. Comparison of convergence rates on structured grids.

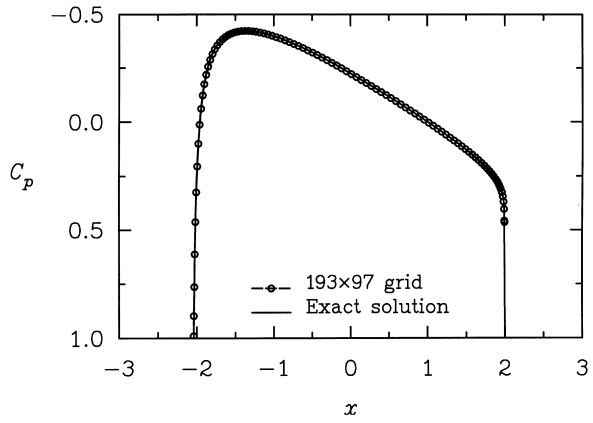

Fig. 7. Surface pressure coefficient, nonlifting Kármán–Trefftz airfoil, 193 × 97 grid.

an airfoil of approximately 15% thickness. The trailing edge angle is 10°. A fine O-grid of 385 × 193 vertices was generated from the conformal mapping, and the nested coarse grids were generated 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 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.

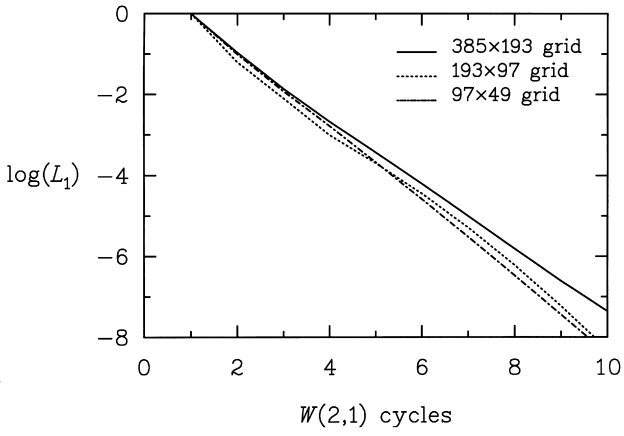

Fig. 8. Comparison of convergence rates for nonlifting Kármán–Trefftz airfoil.

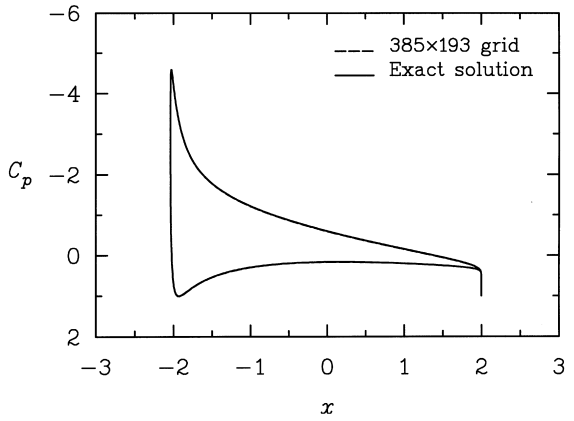

Fig. 9. Surface pressure coefficient, lifting Kármán–Trefftz airfoil, 385 × 193 grid, $\alpha = 10^\circ$.

Comparisons between computed and analytic surface pressure coefficients for non-lifting flow around the Kármán–Trefftz airfoil are shown in Fig. 7. A $W(2, 1)$ multigrid cycle was used for these comparisons. 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. 8. 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 lifting solution for the Kármán–Trefftz airfoil on the 385 × 193 grid is shown in Fig. 9. The airfoil is at an angle-of-attack of 10°. The exact and numerical solutions essentially lie on top of one another. The convergence rate for the two momentum and the pressure equations are shown in Fig. 10. The convergence rate for pressure is 0.186 per cycle, which is slightly

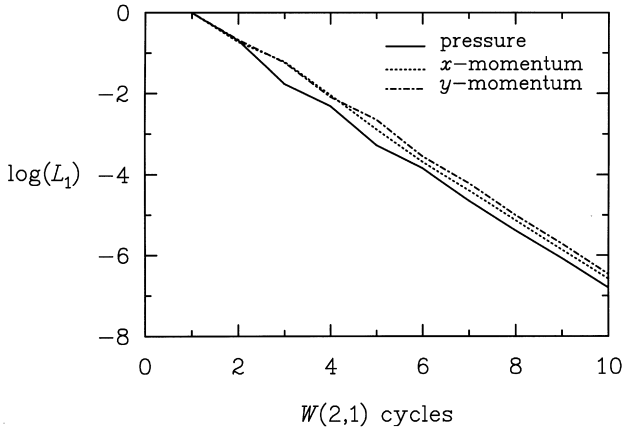

Fig. 10. Convergence rates for lifting Kármán–Trefftz airfoil, 385 × 193 grid, $\alpha = 10^\circ$.

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

		Convergence rate			
		Per cycle		Per work unit	
Case	Cycle	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, nonlifting	$W(2, 1)$	0.125	0.153	0.783	0.802
Airfoil, 385 × 193, structured, lifting	$W(2, 1)$	0.125	0.186	0.783	0.821

worse than for the nonlifting case. It is seen that the asymptotic convergence rates for the two momentum equations are comparable to that of the pressure equation.

A summary of the convergence rates on the finest grid 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 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. (12) 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 one-quarter 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

$$\frac{\mathrm{WU}}{V-\mathrm{cycle}} \approx \left(m+n+1+\frac{1}{4}\right)\left(1+\frac{1}{4}+\frac{1}{16}+\cdots\right) = \frac{4}{3}\left(m+n+\frac{5}{4}\right)$$

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

$$\frac{\text{WU}}{W-\text{cycle}} \approx \left(m+n+1+\frac{1}{4}\right)\left(1+\frac{1}{2}+\frac{1}{4}+\cdots\right)$$
$$= 2\left(m+n+\frac{5}{4}\right)$$

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.

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 five-stage Runge–Kutta scheme on the finest grid. The ideal convergence rates for lexicographic Gauss–Seidel is 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 kth 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 $\tau_{K-1} = O(h^p_{K-1})$ 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 $h_{k-1} = 2\,h_k$, 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

$$W_{\text{total}} = \frac{p}{\log \mu_W}\log\left(\frac{1}{2}\right)\left(1+\frac{1}{4}+\frac{1}{16}+\cdots\right)$$
$$= -\frac{8}{3}\frac{\log 2}{\log \mu_W} \tag{14}$$

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. (14) 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. (14) is a useful guide to the expected performance of the multigrid scheme.

6. Conclusions

A multigrid algorithm which yields textbook multigrid efficiency for the steady Euler equations has been developed. It has the virtue of simplicity; conventional finite-difference or finite-volume discretizations of the governing equations may be used, allowing flexibility in the choice of the underlying numerical method. The correct boundary condition for the pressure equation is obtained directly. Ideal multigrid convergence rates have been demonstrated for both two-dimensional channel flows and airfoil flows. This method has a direct extension to subsonic compressible flows as shown by Sidilkover [15]. Finally, this method can be applied to incompressible, viscous flow following the ideas of Sidilkover and Ascher [13].

Acknowledgements

The authors would like to thank Kyle Anderson and Daryl Bonhaus for providing the unstructured triangular grid solver `FUN2D` which has provided the basis for the triangular grid multigrid solver described in this paper. Also, we thank Jerry South for fostering the environment in which this research was initiated and for his active encouragement during the course of this work.

References

[1] Jameson A. Computational transonics. Commun Pure Appl Math 1988;16:507–49.
[2] Brandt A. Multigrid techniques: 1984 guide with applications to fluid dynamics. GMD-Studie 85, GMD-FIT, 1985.
[3] South JC, Brandt A. Application of a multi-level grid method to transonic flow calculations. In: Adamson T.C., Platzer M., editors. Transonic Flow Problems in Turbomachinery. Washington, DC: Hemisphere, 1977. p. 180–97.
[4] Mulder W. Multigrid relaxation for the Euler equations. J Comput Phys 1985;60(2):235–52.
[5] Anderson WK, Thomas JL, Whitfield DL. Three-dimensional multi-grid algorithms for the flux-split Euler equations. NASA Technical Paper 2829, 1988.
[6] Warren GP, Roberts TW. Multigrid properties of upwind-biased data reconstructions. In: Sixth Copper Mountain Conference on Multigrid Methods. NASA Conference Publication 3224, Part 2, 1993.
[7] Jameson A. Solution of the Euler equations for two dimensional transonic flow by a multigrid method. Appl Math Comput 1983;13(34):327–55.
[8] Vatsa V, Wedan BW. Development of a multigrid code for a 3-D Navier–Stokes equations and its application to a grid-refinement Study. Comput Fluids 1990;18(4):391–403.
[9] Pierce NA, Giles M, Jameson A, Martinelli L. Accelerating three dimensional Navier–Stokes calculations. AIAA Paper 97–1953, 1997.
[10] Brandt A, Yavneh I. Accelerated multigrid convergence and high-Reynolds recirculating flows. SIAM J Sci Statist Comput 1993;14(3):607–26.
[11] Ta'asan S. Canonical-variables multigrid method for steady-state Euler equations. ICASE Report 94-14, 1994.
[12] Ta'asan S. Canonical forms of multidimensional steady inviscid flow. ICASE Report 93-34, 1993.
[13] Sidilkover D, Ascher UM. A multigrid solver for the steady state Navier–Stokes equations using the pressure-Poisson formulation. Comput Appl Math 1995;14(1):21–35.
[14] Roberts TW, Sidilkover D, Swanson RC. Textbook multigrid efficiency for the steady Euler equations. AIAA Paper 97-1949, 1997.

[15] Sidilkover D. Some approaches toward constructing optimally efficient multigrid solver for the inviscid flow equations. Comput Fluids 1998;28:551–71.
[16] Giles M, Anderson WK, Roberts TW. Upwind control volumes: a new upwind approach. AIAA Paper 90-0104, 1990.
[17] Sidilkover D. A genuinely multidimensional upwind scheme and efficient multigrid solver for the compressible Euler equations. ICASE Report 94-84, 1994.