INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2003; **43**:537–553 (DOI: 10.1002/fld.521)

Fast preconditioned multigrid solution of the Euler and Navier–Stokes equations for steady, compressible flows

David A. Caughey[1,*,†] and Antony Jameson[2]

[1] *Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, New York 14853-7501, U.S.A.*
[2] *Department of Aeronautics and Astronautics, Stanford University, Stanford, California 94305-4035, U.S.A.*

SUMMARY

New versions of implicit algorithms are developed for the efficient solution of the Euler and Navier–Stokes equations of compressible flow. The methods are based on a preconditioned, lower-upper (LU) implementation of a non-linear, symmetric Gauss-Seidel (SGS) algorithm for use as a smoothing algorithm in a multigrid method. Previously, this method had been implemented for flows in quasi-one-dimensional ducts and for two-dimensional flows past airfoils on boundary-conforming 'O'-type grids for a variety of symmetric limited positive (SLIP) spatial approximations, including the scalar dissipation and convective upwind split pressure (CUSP) schemes. Here results are presented for both inviscid and viscous (laminar) flows past airfoils on boundary-conforming 'C'-type grids. The method is significantly faster than earlier explicit or implicit methods for inviscid problems, allowing solution of these problems to the level of truncation error in three to five multigrid cycles. Viscous solutions still require as many as twenty multigrid cycles. Copyright © 2003 John Wiley & Sons, Ltd.

KEY WORDS: Euler equations; implicit methods; lower–upper schemes; multigrid methods; Navier–Stokes equations; symmetric Gauss–Seidel iteration; transonic flow

1. INTRODUCTION

Multigrid methods [1] using both explicit [2, 3] and implicit [4–9] time-stepping algorithms are widely used for solving the Euler and Navier–Stokes equations for aerodynamic flows of practical interest. Although these methods have been quite intensively studied in the past several decades, the best methods developed to date still require many tens, if not hundreds, of multigrid cycles for adequate convergence. On the other hand, it is well known that elliptic problems are often solvable in ten, or fewer, cycles [10, 11]. This fact has motivated

*Correspondence to: D. A. Caughey, Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14853-7501, U.S.A.
†E-mail: caughey@mae.cornell.edu

Contract/grant sponsor: AFOSR; contract/grant number: AF F49620-98-1-022

Received 20 April 2002
Accepted 15 January 2003

recent efforts to achieve 'textbook' multigrid efficiency for non-elliptic problems [12–14]. Considering that the solution of a non-linear equation for one unknown by Newton's method will typically require three to five iterations, the present authors have held the view that one should not expect to solve the large number of coupled non-linear equations which result from discretization of the flow equations in many fewer than ten iterations. We felt that this was a target that might be achievable and have continued to search for methods capable of accomplishing it. The results presented in this paper suggest that it may, indeed, be possible to surpass this target, at least for inviscid, transonic flows. In fact, it appears that adequately converged solutions of the Euler equations for transonic flows past airfoils can be obtained in three to five multigrid 'W' cycles.

The principal ingredients of our new scheme are as follows. First, we have adopted the LU-SGS scheme in a fully non-linear form, in which the fluxes at each cell interface are recomputed for each cell using the most recently updated values of the flow solution variables while sweeping forward from left to right, and then backward from right to left. Second, the combined Jacobian matrices that would appear on the diagonal of a linearized implicit, flux-split scheme are used as a preconditioner of the non-linear scheme. Third, we observed a slower rate of convergence in the supersonic zones, so we introduced options for additional sweeps only over cells in which the local Mach number was supersonic. (In the case of the Burgers equation we have found that it can pay to use additional iterations of the interface fluxes in each cell before advancing to the next cell.) Finally, we combine this evolution algorithm with a fully non-linear multigrid algorithm of the type formulated by Jameson [1], which uses the full-approximation scheme (FAS) introduced by Brandt [10]. The multigrid algorithm provides options for 'V' and 'W' cycles, and is set up as a full multigrid scheme in which the calculation is initiated on a coarse grid before proceeding to successively finer grids in each of which the solution process is accelerated by cycles through coarser grids. At each stage of this process a second-order accurate discretization is employed on the (currently) finest grid, while a first order accurate discretization is used on all coarser grids.

A fast rate of convergence results from the effective combination of all these features, which are analyzed in more detail in Section 2. Results for inviscid transonic and viscous (laminar) subsonic flows past two-dimensional airfoils are presented in Section 3. In that section we also show comparisons with the earlier results obtained using 'O'-type grids.

2. ANALYSIS

2.1. Basic Algorithm—Euler equations

We consider the basic structure of the algorithm, describing it first for the Euler equations—or, more generally, any hyperbolic system of conservation laws. We consider the construction of implicit schemes to solve the conservation law

$$\frac{\partial w}{\partial t} + \frac{\partial}{\partial x} f(w) + \frac{\partial}{\partial y} g(w) = 0 \tag{1}$$

If we define Jacobian matrices

$$A = \frac{\partial f}{\partial w}, \quad B = \frac{\partial g}{\partial w} \tag{2}$$

the correction vector

$$\delta w = w^{n+1} - w^n \tag{3}$$

and replace $f(w^{n+1})$ and $g(w^{n+1})$ by the approximations

$$f^{n+1} \sim f^n + A\delta w, \quad g^{n+1} \sim g^n + B\delta w \tag{4}$$

we have the prototype implicit scheme

$$\{I + \mu\Delta t(D_x A + D_y B)\}\delta w + \Delta t R = 0 \tag{5}$$

where $\mu \geqslant \frac{1}{2}$ and

$$R = D_x f(w) + D_y g(w) + \text{artificial dissipation terms} \tag{6}$$

is the residual. If $\mu = 1$, this becomes a Newton iteration to solve the steady-state equation in the limit as $\Delta t \to \infty$.

A flux-split version of the fully implicit version ($\mu = 1$) of Equation (5) can be written

$$\{I + \Delta t(D_x^+ A^- + D_y^+ B^-) + \Delta t(D_x^- A^+ + D_y^- B^+)\}\delta w + \Delta t R = 0 \tag{7}$$

where D_x^+, D_y^+ are forward differences, D_x^-, D_y^- are backward differences, and

$$R = D_x^+ f^- + D_y^+ g^- + D_x^- f^+ + D_y^- g^+ \tag{8}$$

The fluxes are split so that the Jacobians

$$A^+ = \frac{\partial f^+}{\partial w} \quad \text{and} \quad B^+ = \frac{\partial g^+}{\partial w}$$

have positive eigenvalues, while

$$A^- = \frac{\partial f^-}{\partial w} \quad \text{and} \quad B^- = \frac{\partial g^-}{\partial w}$$

have negative eigenvalues.

To reduce the computation required for each time step one can approximate the left-hand side of Equation (7) by a product of more easily invertible factors. Historically, this is done either by introducing an alternating direction implicit (ADI) factorization [15–17] or a lower-upper (LU) factorization [9, 18–20]. The LU factorization results in

$$\{I + \Delta t(D_x^+ A^- + D_y^+ B^-)\}\{I + \Delta t(D_x^- A^+ + D_y^- B^+)\}\delta w + \Delta t R = 0 \tag{9}$$

Here, as earlier, A^+, B^+ are Jacobians having positive eigenvalues, and A^-, B^- are Jacobians having negative eigenvalues, with

$$A^+ + A^- = A, \quad B^+ + B^- = B \tag{10}$$

The LU scheme requires the solution only of block bidiagonal factors for each time step.

A symmetric Gauss-Seidel iteration of an implicit scheme can be constructed as follows; to simplify the notation at this point only the one-dimensional problem will be described. Consider the flux split scheme

$$\{I + r(\delta_x^+ A^- + \delta_x^- A^+)\}\delta v + \Delta t R = 0 \tag{11}$$

where $\delta_x^+ = \Delta x D^+$, $\delta_x^- = \Delta x D^-$ are forward and backward undivided differences, $r = \Delta t/\Delta x$ and R is the residual. Alternatively, we can write

$$\delta v_i + r(A_{i+1}^- \delta v_{i+1} - A_i^- \delta v_i + A_i^+ \delta v_i - A_{i-1}^+ \delta v_{i-1}) + \Delta t R_i = 0. \tag{12}$$

A symmetric Gauss-Seidel scheme consists of the forward sweep

$$\delta \tilde{v}_i + r(A_i^+ - A_i^-)\delta \tilde{v}_i - r A_{i-1}^+ \delta \tilde{v}_{i-1} + \Delta t R_i = 0 \tag{13}$$

followed by the backward sweep

$$\delta v_i + r(A_i^+ - A_i^-)\delta v_i + r A_{i+1}^- \delta v_{i+1} - r A_{i-1}^+ \delta \tilde{v}_{i-1} + \Delta t R_i = 0 \tag{14}$$

If we subtract (13) from (14) we obtain

$$\{I + r(A_i^+ - A_i^-)\}\delta v_i + r A_{i+1}^- \delta v_{i+1} = \{I + r(A_i^+ - A_i^-)\}\delta \tilde{v}_i \tag{15}$$

which can be written as

$$L D^{-1} U \delta v = -\Delta t R \tag{16}$$

where

$$\begin{aligned} L &\equiv I - rA^- + r\delta_x^- A^+ \\ U &\equiv I + rA^+ + r\delta_x^+ A^- \\ D &\equiv I + r(A^+ - A^-) \end{aligned} \tag{17}$$

If we take

$$A^+ = \frac{1}{2}(A + \rho I) \quad A^- = \frac{1}{2}(A - \rho I) \tag{18}$$

where $\rho = \max |\lambda(A)|$ (to ensure diagonal dominance) then D reduces to a scalar factor and this is a variation of the LU scheme.

The terms $\Delta t R_i - r A_{i-1}^+ \delta \tilde{v}_{i-1}$ of Equation (13) are a linearization of $\Delta t R_i$ evaluated with $\tilde{v}_{i-1} = v_{i-1} + \delta \tilde{v}_{i-1}$. Following this line of reasoning, the LU-SGS scheme can be recast as

$$\{I + r(A_i^+ - A_i^-)\}\delta \tilde{w}_i + \Delta t \tilde{R}_i = 0 \tag{19}$$

$$\{I + r(A_i^+ - A_i^-)\}\delta \tilde{\tilde{w}}_i + \Delta t \tilde{\tilde{R}}_i = 0 \tag{20}$$

where

$$\tilde{w}_i = w_i + \delta \tilde{w}_i, \quad \tilde{f}_i^{\pm} = f^{\pm}(\tilde{w}_i) \tag{21}$$

$$w_i^{n+1} = \tilde{\tilde{w}}_i = \tilde{w}_i + \delta \tilde{\tilde{w}}_i, \quad \tilde{\tilde{f}}_i^{\pm} = f^{\pm}(\tilde{\tilde{w}}_i) \tag{22}$$

and

$$\tilde{R}_i = \frac{1}{\Delta x}(f_{i+1}^- - f_i^- + f_i^+ - \tilde{f}_{i-1}^+) \tag{23}$$

$$\tilde{\tilde{R}}_i = \frac{1}{\Delta x}(\tilde{\tilde{f}}_{i+1}^- - \tilde{f}_i^- + \tilde{f}_i^+ - \tilde{f}_{i-1}^+) \tag{24}$$

With the definitions of Equation (18), Equations (19) and (20) can be written

$$\delta\tilde{w}_i = -\frac{\Delta t}{1+\mathscr{C}}\tilde{R}_i \tag{25}$$

$$\delta\tilde{\tilde{w}}_i = -\frac{\Delta t}{1+\mathscr{C}}\tilde{\tilde{R}}_i \tag{26}$$

where $\mathscr{C} = \rho\Delta t/\Delta x$ is the Courant number. Alternatively, with the Jacobian splitting defined as

$$A^+ = \frac{1}{2}(A + |A|), \quad A^- = \frac{1}{2}(A - |A|) \tag{27}$$

where $|A| = M|\Lambda|M^{-1}$, with $|\Lambda|$ the diagonal matrix whose entries are the absolute values of the eigenvalues of the Jacobian matrix A and M, M^{-1} are the modal matrix of A and its inverse, Equations (19) and (20) can be written

$$\{I + r|A|\}\delta\tilde{w}_i = -\Delta t\tilde{R}_i \tag{28}$$

$$\{I + r|A|\}\delta\tilde{\tilde{w}}_i = -\Delta t\tilde{\tilde{R}}_i \tag{29}$$

and, in the limit as the time step Δt (or, equivalently, $r = \Delta t/\Delta x$) goes to infinity, these equations represent the SGS Newton iteration

$$|A|\delta\tilde{w}_i = -\Delta x\tilde{R}_i \tag{30}$$

$$|A|\delta\tilde{\tilde{w}}_i = -\Delta x\tilde{\tilde{R}}_i \tag{31}$$

The introduction of the splitting defined by Equations (27) is motivated, in part, by the success of the similar preconditioner introduced by Allmaras [21] and used by Pierce and Giles [3] to accelerate the convergence of codes based on explicit Runge–Kutta time stepping. This preconditioner seems to have its roots in the diagonally-dominant ADI scheme [22, 23].

The scheme corresponding to Equations (30) and (31) is implemented for the finite-volume form [2] of the equations, which can be represented (for the Euler equations in two dimensions) as

$$\{|A| + |B|\}\delta\tilde{w}_{i,j} = -\sigma\tilde{R}_{i,j} \tag{32}$$

$$\{|A| + |B|\}\delta\tilde{\tilde{w}}_{i,j} = -\sigma\tilde{\tilde{R}}_{i,j} \tag{33}$$

where

$$\tilde{R}_{i,j} = F_{i+1,j}^- - F_{i,j}^- + F_{i,j}^+ - \tilde{F}_{i-1,j}^+ + G_{i,j+1}^- - G_{i,j}^- + G_{i,j}^+ - \tilde{G}_{i,j-1}^+ \tag{34}$$

and

$$\tilde{\bar{R}}_{i,j} = \tilde{\bar{F}}^{-}_{i+1,j} - \tilde{F}^{-}_{i,j} + \tilde{F}^{+}_{i,j} - \tilde{F}^{+}_{i-1,j} + \tilde{\bar{G}}^{-}_{i,j+1} - \tilde{G}^{-}_{i,j} + \tilde{G}^{+}_{i,j} - \tilde{G}^{+}_{i,j-1} \tag{35}$$

and σ is a relaxation factor that can be used to optimize convergence rates. In these equations F^+, F^-, G^+, and G^- represent the split approximations to the cell area h times the contravariant components of the flux vectors in the corresponding mesh co-ordinate directions. The residual fluxes are approximated here using the convective upwind split pressure (CUSP) version of the symmetric limited positive (SLIP) approximations developed by Jameson References [24, 25].

The implementation of this procedure is made computationally very efficient by locally transforming the residuals to those corresponding to the equations written in primitive variables (see, e.g. Reference [26]), then transforming the corrections back to the conserved variables. Numerical experiments indicate that it can be beneficial to perform additional corrections in supersonic zones, when they are present in the solution. The CPU time required for these multiple sweeps is reduced by 'freezing' the matrix coefficients $|A|$ and $|B|$ that appear in Equations (30) and (31). The additional memory required to store these coefficient matrices is minimized by storing only the symmetrized form of the Jacobians (which requires only seven additional quantities to be stored for each mesh cell).

2.2. *Symmetrization of Jacobians*

The Jacobians can be symmetrized by a transformation of dependent variables. For example, the quasi-linear form of the Euler equations

$$\frac{\partial w}{\partial t} + A\frac{\partial w}{\partial \xi} + B\frac{\partial w}{\partial \eta} = 0 \tag{36}$$

can be transformed to

$$\frac{\partial \bar{w}}{\partial t} + S^{-1}AS\frac{\partial \bar{w}}{\partial \xi} + S^{-1}BS\frac{\partial \bar{w}}{\partial \eta} = 0 \tag{37}$$

where

$$\bar{A} = S^{-1}AS = \begin{Bmatrix} Q & cS_x & cS_y & 0 \\ cS_x & Q & 0 & 0 \\ cS_y & 0 & Q & 0 \\ 0 & 0 & 0 & Q \end{Bmatrix} \tag{38}$$

where

$$Q = S_x u + S_y v \tag{39}$$

and S_x and S_y are the face areas projected parallel to the x and y axes, respectively. The Jacobian $\bar{B}$ has a similar form.

Each factor of the split scheme can then be written

$$\{I + \Delta t(|\bar{A}| + |\bar{B}|)\}\delta\bar{w}_{i,j} = \Delta t S^{-1}\mathrm{Res}_{i,j} \tag{40}$$

where

$$|\bar{A}| = Q|\Lambda_A|Q^{-1}, \quad |\bar{B}| = Q|\Lambda_B|Q^{-1} \tag{41}$$

and

$$\delta w_{i,j} = S\delta\bar{w}_{i,j} \tag{42}$$

The 'absolute' Jacobians have the (symmetric) form

$$|\bar{A}| = \begin{pmatrix} R_1 & \bar{S}_x R_2 & \bar{S}_y R_2 & 0 \\ \bar{S}_x R_2 & \bar{S}_y^2 Q_1 + \bar{S}_x^2 R_1 & \bar{S}_x \bar{S}_y [R_1 - Q_1] & 0 \\ \bar{S}_y R_2 & \bar{S}_x \bar{S}_y [R_1 - Q_1] & \bar{S}_x^2 Q_1 + \bar{S}_y^2 R_1 & 0 \\ 0 & 0 & 0 & Q_1 \end{pmatrix} \tag{43}$$

where

$$R_1 = \frac{Q_2 + Q_3}{2} \quad \text{and} \quad R_2 = \frac{Q_2 - Q_3}{2} \tag{44}$$

$$\bar{S}_x = \frac{S_x}{\sqrt{S_x^2 + S_y^2}} \quad \text{and} \quad \bar{S}_y = \frac{S_y}{\sqrt{S_x^2 + S_y^2}} \tag{45}$$

and Q_1, Q_2, Q_3 are the absolute values of the eigenvalues of $\bar{A}$.

Transformations described by Abarbanel and Gottlieb [27] can simultaneously symmetrize the Euler and Viscous Jacobians. We here consider the implicit treatment only of the non-mixed second derivatives appearing in the Navier–Stokes equations, which can be written

$$\frac{\partial w}{\partial t} + A\frac{\partial w}{\partial \xi} + B\frac{\partial w}{\partial \eta} - A_v\frac{\partial^2 w}{\partial \xi^2} - B_v\frac{\partial^2 w}{\partial \eta^2} = 0 \tag{46}$$

The transformation sequence then becomes

$$\frac{\partial \bar{w}}{\partial t} + S_v^{-1}AS_v\frac{\partial \bar{w}}{\partial \xi} + S_v^{-1}BS_v\frac{\partial \bar{w}}{\partial \eta} - S_v^{-1}A_vS_v\frac{\partial^2 \bar{w}}{\partial \xi^2} - S_v^{-1}B_vS_v\frac{\partial^2 \bar{w}}{\partial \eta^2} = 0 \tag{47}$$

where now

$$\hat{A} = S_v^{-1}AS_v$$

$$= \begin{Bmatrix} Q & \frac{cS_x}{\sqrt{\gamma}} & \frac{cS_y}{\sqrt{\gamma}} & 0 \\ \frac{cS_x}{\sqrt{\gamma}} & Q & 0 & cS_x\sqrt{\frac{\gamma-1}{\gamma}} \\ \frac{cS_y}{\sqrt{\gamma}} & 0 & Q & cS_y\sqrt{\frac{\gamma-1}{\gamma}} \\ 0 & cS_x\sqrt{\frac{\gamma-1}{\gamma}} & cS_y\sqrt{\frac{\gamma-1}{\gamma}} & Q \end{Bmatrix} \tag{48}$$

Also, for the viscous Jacobians

$$\hat{A}_v = S_v^{-1} A_v S_v = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & (\frac{4}{3}S_y^2 + S_x^2)\frac{\mu}{\rho h} & -\frac{2}{3}S_x S_y \frac{\mu}{\rho h} & 0 \\ 0 & -\frac{2}{3}S_x S_y \frac{\mu}{\rho h} & (S_x^2 + S_y^2)\frac{\mu}{\rho h} & 0 \\ 0 & 0 & 0 & (S_x^2 + S_y^2)\frac{\mu}{\rho h} \end{pmatrix} \tag{49}$$

and $\hat{B}$ and $\hat{B}_v$ have similar forms.

Then, each factor of the split scheme for the Navier–Stokes equations can then be written

$$\{I + \Delta t(|\hat{A}| + 2\hat{A}_v + |\hat{B}| + 2\hat{B}_v)\}\delta\bar{w}_{i,j} = \Delta t S_v^{-1} \mathrm{Res}_{i,j} \tag{50}$$

where

$$|\hat{A}| = Q_v |\Lambda_A| Q_v^{-1}, \quad |\hat{B}| = Q_v |\Lambda_B| Q_v^{-1} \tag{51}$$

The 'absolute' Jacobians $|\hat{A}|$ and $|\hat{B}|$ now take the form

$$|\hat{A}| = \begin{pmatrix} \frac{(\gamma-1)Q_1+R_1}{\gamma} & \frac{\bar{S}_x R_2}{\sqrt{\gamma}} & \frac{\bar{S}_y R_2}{\sqrt{\gamma}} & \frac{\gamma-1}{\gamma}(R_1 - Q_1) \\ \frac{\bar{S}_x R_2}{\sqrt{\gamma}} & \bar{S}_y^2 Q_1 + \bar{S}_x^2 R_1 & \bar{S}_x \bar{S}_y (R_1 - Q_1) & \sqrt{\frac{\gamma-1}{\gamma}}\,\bar{S}_x R_2 \\ \frac{\bar{S}_y R_2}{\sqrt{\gamma}} & \bar{S}_x \bar{S}_y (R_1 - Q_1) & \bar{S}_x^2 Q_1 + \bar{S}_y^2 R_1 & \sqrt{\frac{\gamma-1}{\gamma}}\,\bar{S}_y R_2 \\ \frac{\gamma-1}{\gamma}(R_1 - Q_1) & \sqrt{\frac{\gamma-1}{\gamma}}\,\bar{S}_x R_2 & \sqrt{\frac{\gamma-1}{\gamma}}\,\bar{S}_y R_2 & \frac{Q_1+(\gamma-1)R_1}{\gamma} \end{pmatrix} \tag{52}$$

where

$$R_1 = \frac{Q_2 + Q_3}{2} \quad \text{and} \quad R_2 = \frac{Q_2 - Q_3}{2} \tag{53}$$

and Q_1, Q_2, Q_3 are now the absolute values of the eigenvalues of $\hat{A}$. In spite of the lack of sparsity, there is a significant reduction in memory required to save the Jacobians due to the symmetrization.

2.3. *Boundary conditions and multigrid*

Boundary conditions for two-dimensional problems are enforced in the standard way. For the Euler equations, the no-flux condition is enforced directly at solid boundaries, and the pressure is determined from the normal momentum equation. Values of density and total energy are required in dummy cells and are set by extrapolating the entropy and total enthalpy with zero gradients normal to the boundary. For the Navier–Stokes equations, both components of velocity are set to zero, as are the normal gradients of pressure and temperature.

The flows considered here are subsonic at infinity, so at far field boundaries the outgoing Riemann invariant is extrapolated from the interior and the incoming Riemann invariant is set by free stream quantities. At inflow boundaries, the total enthalpy and entropy also are fixed at their free stream values. At outflow boundaries, the entropy and total enthalpy are

extrapolated from the interior for Euler solutions. For Navier–Stokes solutions, the velocities and entropy are extrapolated from the interior, and the pressure is set to its free stream value.

The algorithm is implemented within the framework of the multigrid method, following the (now standard) procedure of Jameson [1].

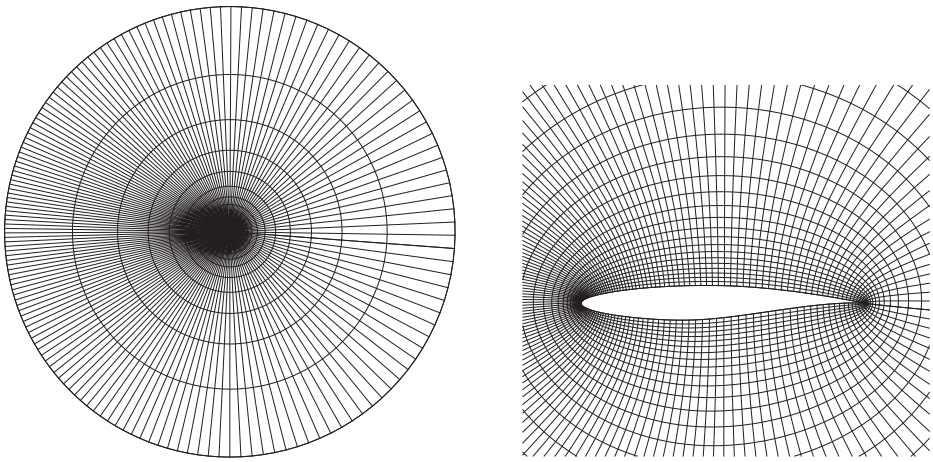
3. RESULTS

The results of several sample computations are presented here to illustrate both, the asymptotic rate of convergence of the method, and the speed with which global measures of the solutions, such as force and moment coefficients and surface pressure distributions, converge. First, inviscid solutions for transonic flows past airfoils computed on 'C'-type grids are compared with solutions reported earlier on 'O'-type grids [28]. Then the results of subsonic laminar (viscous) airfoil computations are presented.

The airfoil results presented here are computed on 'O'-type grids extending from the airfoil to a far-field boundary located approximately 26 chord lengths from the body surface and on 'C'-type grids extending from the airfoil surface to far-field boundaries located approximately 15 chord lengths laterally and 7.5 chord lengths in the streamwise directions from the body surface. Results are presented for computations using the CUSP spatial dissipation models. (See Reference [28] for results using the scalar SLIP dissipation model for the 'O'-grid computations.) It is found from numerical experiments that best convergence rates are achieved when extra correction sweeps are performed locally in supersonic zones. In the present computations, three extra (bi-directional) sweeps are performed on the fine grid, and one extra (bi-directional) sweep is performed on each of the coarser grids. Corrections on the fine grid are under-relaxed slightly (typically $\sigma = 0.95$), and are over-relaxed on all coarser grids (typically $\sigma = 1.40$),

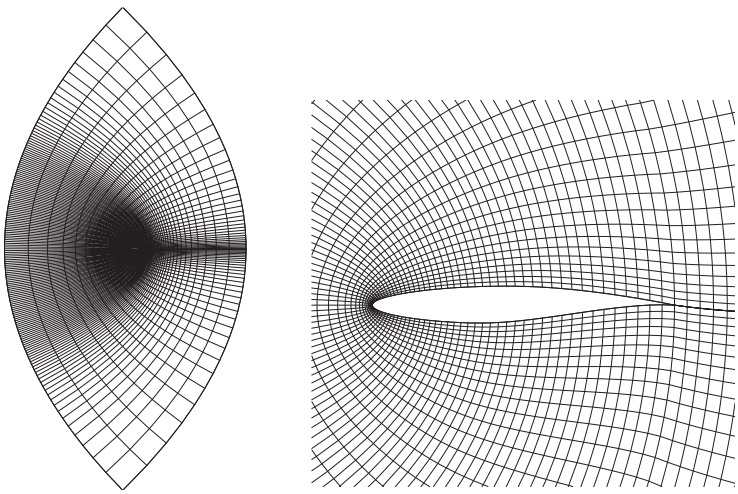
All results for the Euler equations are computed using a sequence of four grids; having 20×4, 40×8, 80×16, and 160×32 mesh cells in the wrap-around and body-normal directions, respectively. An uniform free stream is used as the initial condition for the solution on the coarsest grid, and interpolated coarser grid solutions are used as initial conditions for the subsequent, finer, grids. Two views of the final fine mesh used for computing the flow past the RAE 2822 airfoil are shown in Figures 1 and 2 for the 'O'- and 'C'-grids, respectively.

Plates 1, 2 and 3 show the convergence histories for the 'O'-grid and 'C'-grid solutions of three representative transonic flow problems; the flow past the RAE 2822 airfoil at Mach 0.75 and 3.00° incidence, and the flow past the NACA 0012 airfoil at Mach 0.80 and 1.25° incidence and at Mach 0.85 and 1.00° incidence. In each case, a sequence of up to 100 multigrid cycles on each of the four successive grids is shown; the iterations are halted if the residual or the fractional deviation in total enthalpy reaches 10^{-12}. Results are displayed for computations using the CUSP spatial discretization scheme, which is designed to drive the total enthalpy of iteratively-converged solutions to a constant, and the fractional deviation in total enthalpy is plotted in the figures along with the residual. (Surface pressure distributions for these cases will be displayed later.) These figures demonstrate that the asymptotic rate of convergence of the method is nearly independent of problem size (i.e. number of grid cells), and that residuals approaching 64-bit round-off levels can usually be reached in fewer than about 75 multigrid cycles.



(a) Entire grid (b) Region near body

Figure 1. 'O'-type mesh, containing 160×32 cells in wrap-around and body normal directions, respectively, for computing the transonic flow past the RAE 2822 airfoil. (a) Complete mesh; (b) mesh in vicinity of airfoil surface.



(a) Entire grid (b) Region near body

Figure 2. 'C'-type mesh, containing 160×32 cells in wrap-around and body normal directions, respectively, for computing the transonic flow past the RAE 2822 airfoil. (a) Complete mesh; (b) mesh in vicinity of airfoil surface.

Table I. Average rates of convergence per multigrid cycle for 100 cycles, or until residual or fractional error in total enthalpy is reduced to less than 10^{-12}. Inviscid airfoil computations on 'O'- and 'C'-type grids using CUSP scheme.

Number of Grid Cells		20×4	40×8	80×16	160×32
RAE 2822; $\mathbf{M} = 0.75,\ \alpha = 3.0^\circ$	'O'-grid	0.833	0.656	0.705	0.747
	'C'-grid	0.694	0.657	0.716	0.779
NACA 0012; $\mathbf{M} = 0.80,\ \alpha = 1.25^\circ$	'O'-grid	0.773	0.731	0.722	0.759
	'C'-grid	0.691	0.724	0.750	0.794
NACA 0012; $\mathbf{M} = 0.85,\ \alpha = 1.0^\circ$	'O'-grid	0.737	0.684	0.767	0.779
	'C'-grid	0.669	0.633	0.736	0.791

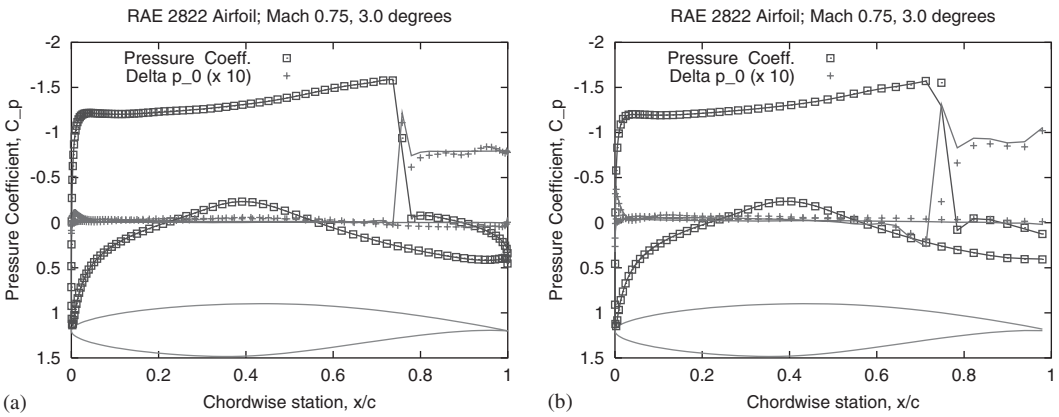


Figure 3. Comparison of full multigrid solutions after only three cycles on each grid (symbols) with fully converged solutions on identical grid (lines); RAE 2822 airfoil at Mach 0.75 and 3.0° incidence on 160×32 cell grids. (a) 'O'-grid, and (b) 'C'-grid.

The average rates of convergence for these computations are summarized in Table I. It should be noted that the same relaxation factors were used for all grids in these computations; slightly better rates of convergence can be achieved if the relaxation factors are optimized for each grid.

In addition to having an impressive asymptotic rate of convergence, the method also converges the global measures of error quickly. Errors in lift and drag coefficients typically are reduced to the level of truncation error (on the order of 1–2% of their iteratively converged values) in three to five multigrid cycles. For optimal efficiency it may be beneficial to use more cycles on the coarser grids, since additional cycles on the coarser grids cost relatively little in CPU time. To illustrate this rapid convergence, surface pressure distributions computed using just three multigrid cycles are compared with fully converged distributions in Figures 3–5 for the RAE 2822 and NACA 0012 airfoil test cases presented above. In each case just three multigrid cycles were performed on a full multigrid sequence of grids. Six successively finer grids, starting with a 5×1 cell grid, were used for the 'O'-grid computations, and five successively finer grids, starting with a 10×2 cell grid, were used

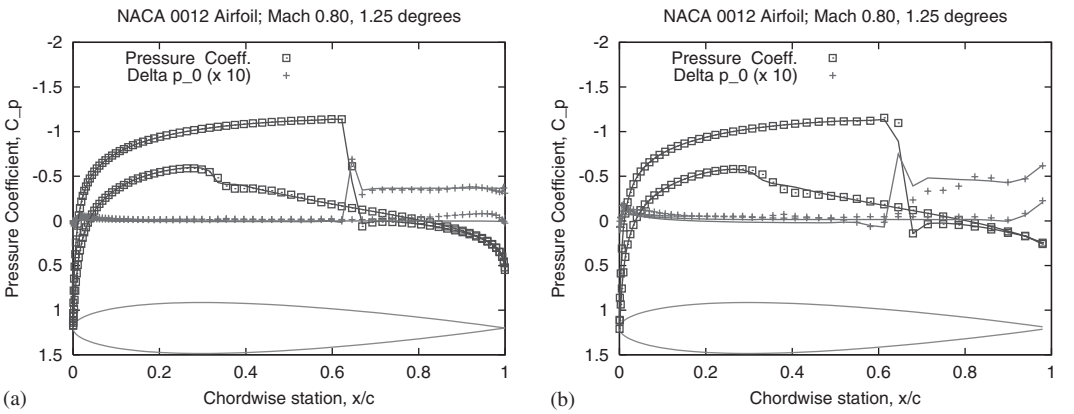


Figure 4. Comparison of full multigrid solutions after only three cycles on each grid (symbols) with fully converged solutions on identical grid (lines); NACA 0012 airfoil at Mach 0.80 and 1.25° incidence on 160×32 cell grids. (a) 'O'-grid, and (b) 'C'-grid.

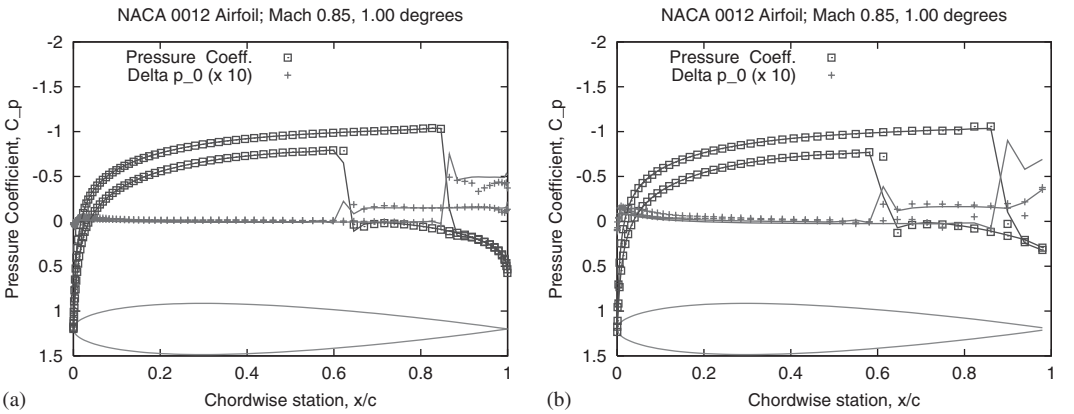


Figure 5. Comparison of full multigrid solutions after only three cycles on each grid (symbols) with fully converged solutions on identical grid (lines); NACA 0012 airfoil at Mach 0.85 and 1.00° incidence on 160×32 cell grids. (a) 'O'-grid, and (b) 'C'-grid.

for the 'C'-grid computations. The figures demonstrate that the surface pressure distributions are well converged in just three cycles for these cases, although the distribution of entropy (plotted as fractional total pressure loss) are not as well converged.

In order to converge the more sensitive measures of error, such as the entropy, it appears that several more cycles are required. Surface pressure and entropy (total pressure loss) distributions computed using just five multigrid cycles are compared with fully converged distributions in Figures 6–8 for the same RAE 2822 and NACA 0012 airfoil test cases presented above. In each case just five multigrid cycles were performed on the same sequence of successively finer grids as for the preceding computations. The figures demonstrate that the surface entropy distributions are well converged in just five cycles for these cases.

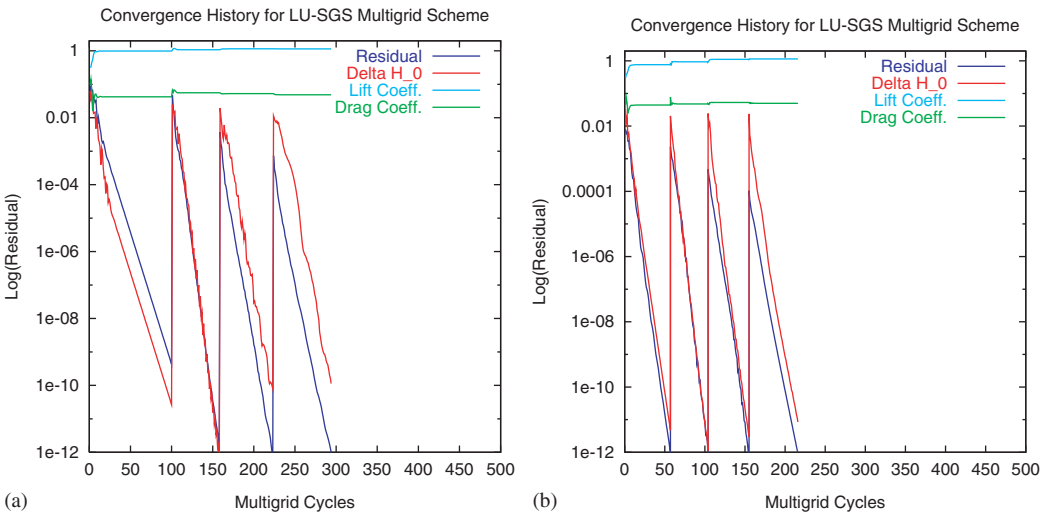


Plate 1. Convergence rates for transonic flow past the RAE 2822 airfoil at Mach 0.75 and 3.0° incidence. Computations proceed on a sequence of four grids, with the finest grids containing 160×32 cells. (a) 'O'-grid, and (b) 'C'-grid.

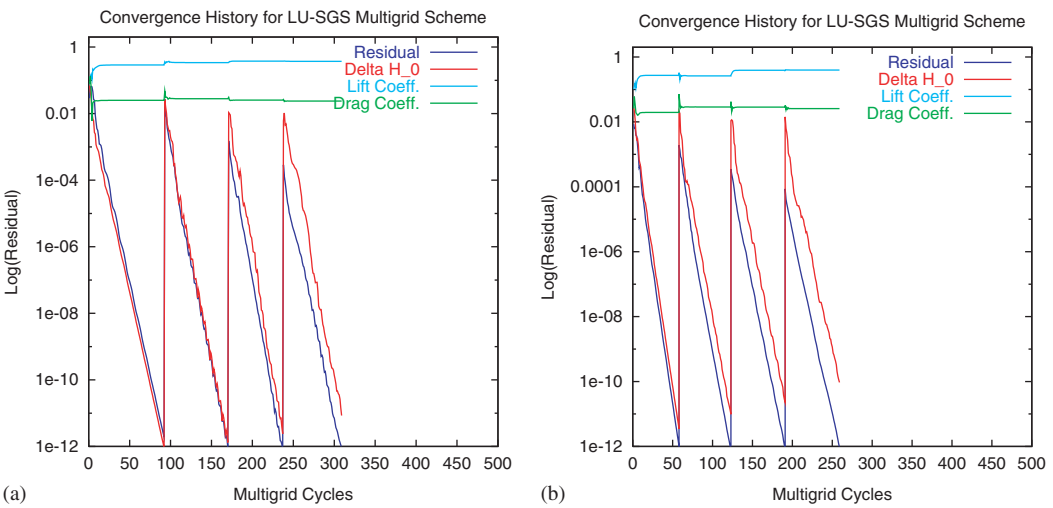


Plate 2. Convergence rates for transonic flow past the NACA 0012 airfoil at Mach 0.80 and 1.25° incidence. Computations proceed on a sequence of four grids, with the finest grids containing 160×32 cells. (a) 'O'-grid, and (b) 'C'-grid.

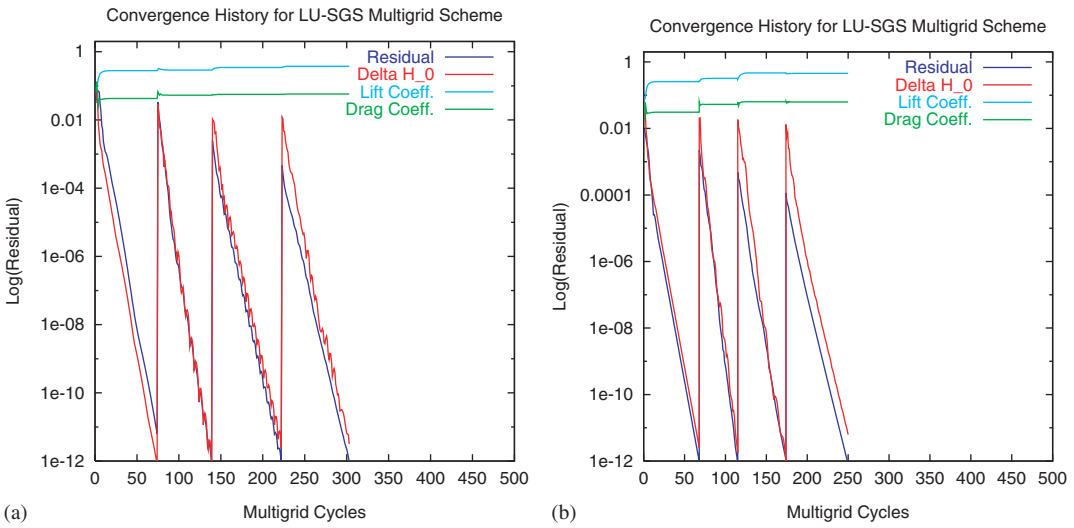


Plate 3. Convergence rates for transonic flow past the NACA 0012 airfoil at Mach 0.85 and 1.00° incidence. Computations proceed on a sequence of four grids, with the finest grids containing 160×32 cells. (a) 'O'-grid, and (b) 'C'-grid.

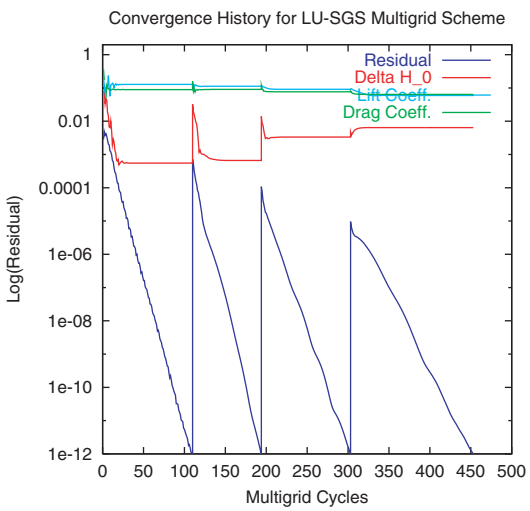


Plate 4. Convergence rates for flow past the NACA 0012 airfoil at Mach 0.50 and 1.00° incidence, and $\mathbf{Re}_c = 5000$. Computations proceed on a sequence of four grids, with the finest grids containing 160×48 cells. (a) 'O'-grid, and (b) 'C'-grid.

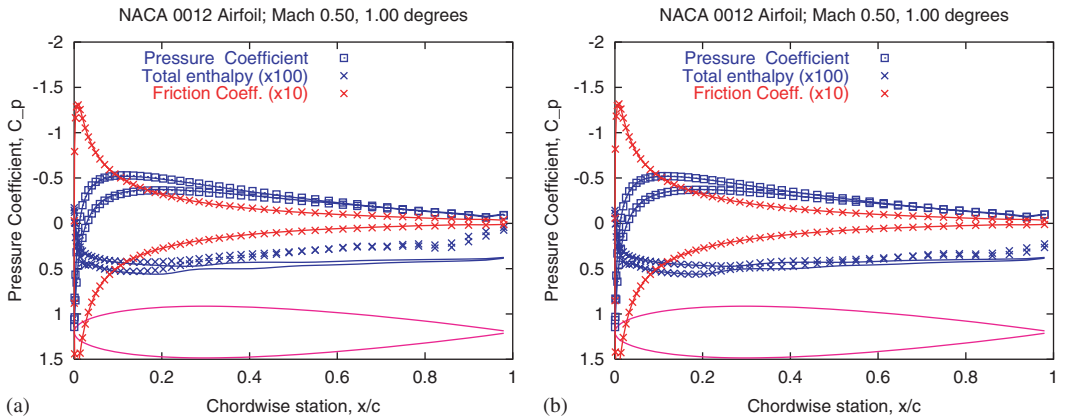


Plate 5. Comparison of surface pressure and skin friction coefficients and percent change in total enthalpy for flow past NACA 0012 airfoil at Mach 0.50 and 1.00° incidence and $\mathbf{Re}_c = 5000$ on 160×48 cell grid. Surface distributions after (a) 10 and (b) 20 full multigrid cycles (symbols) are compared with those of fully converged solutions (lines). (a) 10 Multigrid Cycles, and (b) 20 Multigrid Cycles.

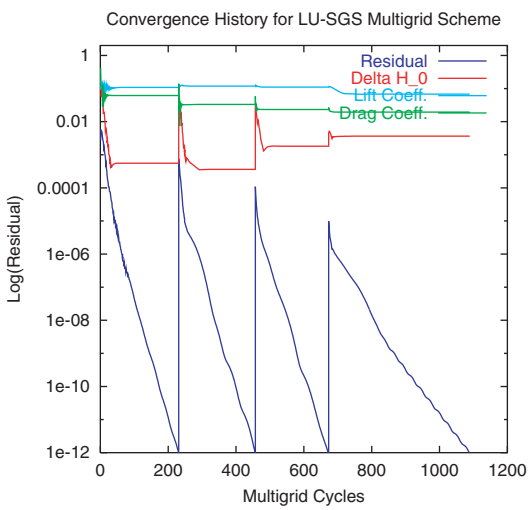


Plate 6. Convergence rates for flow past the NACA 0012 airfoil at Mach 0.50 and 1.00° incidence, and $\mathbf{Re}_c = 80\,000$. Computations proceed on a sequence of four grids, with the finest grids containing 160×48 cells.

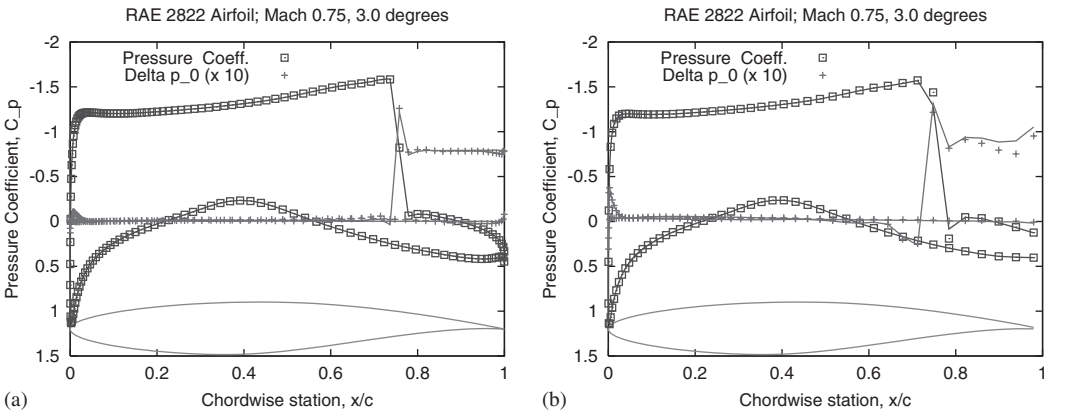


Figure 6. Comparison of full multigrid solutions after only five cycles on each grid (symbols) with fully converged solutions on identical grid (lines); RAE 2822 airfoil at Mach 0.75 and 3.0° incidence on 160×32 cell grids. (a) 'O'-grid, and (b) 'C'-grid.

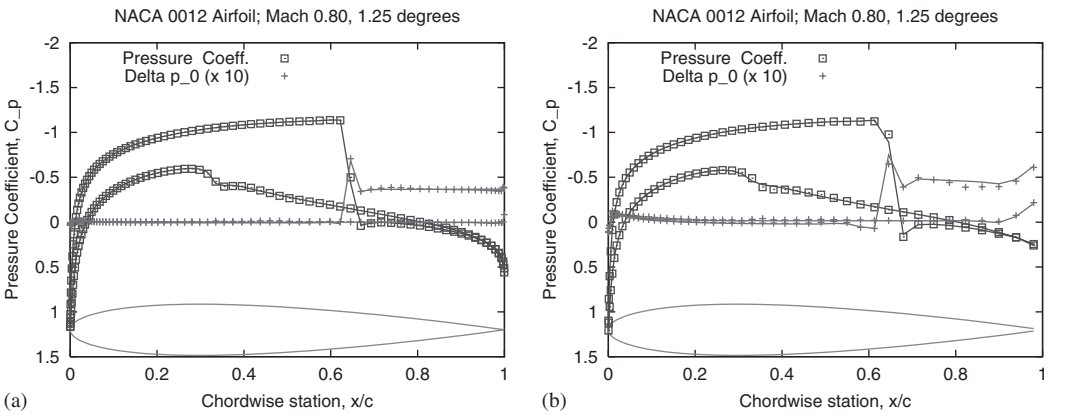


Figure 7. Comparison of full multigrid solutions after only five cycles on each grid (symbols) with fully converged solutions on identical grid (lines); NACA 0012 airfoil at Mach 0.80 and 1.25° incidence on 160×32 cell grids. (a) 'O'-grid, and (b) 'C'-grid.

The lift and drag coefficients for these computations are compared with their (iteratively) converged values in Tables II and III. The data in the table verify that the lift and drag coefficients are within about 1% of their iteratively-converged values after only three multigrid cycles, and to within a fraction of 1% after only 5 cycles. It is worth emphasizing that the convergence studied here is the *iterative* convergence to the solution of the difference equations on a given grid. The solutions on the finest grids presented here also have *truncation* errors that are easily the same order as the iterative convergence errors for the three-cycle computations. In fact, the lift coefficient for the higher Mach number NACA 0012 airfoil case differs by approximately 15% for the two computations presented here (due, most likely, to a significant overprediction of the lift on the 'C'-grid, which is relatively coarse in the vicinity of the trailing edge of the airfoil).

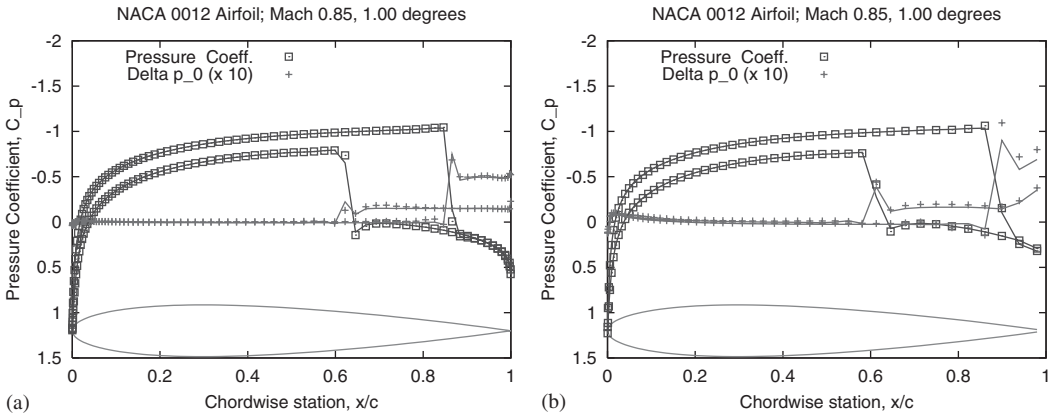


Figure 8. Comparison of full multigrid solutions after only five cycles on each grid (symbols) with fully converged solutions on identical grid (lines); NACA 0012 airfoil at Mach 0.85 and 1.00° incidence on 160×32 cell grids. (a) 'O'-grid, and (b) 'C'-grid.

Table II. Force coefficients for the fast, preconditioned multigrid solutions on 'O'-type grids; see Figures 3(a), 4(a), 5(a), 6(a), 7(a) and 8(a).

Case	Figure	MG Cycles	C_L	C_D
RAE 2822; $\mathbf{M} = 0.75$; $\alpha = 3.00$	—	100	1.1417	0.04851
	6(a)	5	1.1429	0.04851
	3(a)	3	1.1451	0.04886
NACA 0012; $\mathbf{M} = 0.80$; $\alpha = 1.25$	—	100	0.3725	0.02377
	7(a)	5	0.3746	0.02391
	4(a)	3	0.3770	0.02387
NACA 0012; $\mathbf{M} = 0.85$; $\alpha = 1.00$	—	100	0.3753	0.05748
	8(a)	5	0.3721	0.05722
	5(a)	3	0.3644	0.05670

Table III. Force coefficients for the fast, preconditioned multigrid solutions for the Euler equations on 'C'-type grids; see Figures 3(b), 4(b), 5(b), 6(b), 7(b), and 8(b).

Case	Figure	MG Cycles	C_L	C_D
RAE 2822; $\mathbf{M} = 0.75$; $\alpha = 3.00$	—	100	1.1268	0.0481
	6(b)	5	1.1301	0.0486
	3(b)	3	1.1391	0.0499
NACA 0012; $\mathbf{M} = 0.80$; $\alpha = 1.25$	—	100	0.3863	0.0255
	7(b)	5	0.3872	0.0256
	4(b)	3	0.3923	0.0258
NACA 0012; $\mathbf{M} = 0.85$; $\alpha = 1.00$	—	100	0.4323	0.0613
	8(b)	5	0.4324	0.0619
	5(b)	3	0.4179	0.0611

Finally, a result will be presented for the solution of the Navier–Stokes equations for the (laminar) flow past the NACA 0012 airfoil at $\mathbf{M} = 0.50$ and $\alpha = 1^\circ$ incidence, and a (chordwise) Reynolds number of $\mathbf{Re}_c = 5000$. The convergence history for this computation on a sequence of four grids, the finest containing 160×48 cells in the wrap-around and body-normal directions, respectively, is shown in Plate 4.

In spite of the poorer asymptotic convergence rates, reasonably accurate solutions of the Navier–Stokes equations can still be obtained in a relatively small number of cycles. The surface pressure and skin friction coefficient distributions are shown for this solution in Plate 5 for computations that were terminated after 10 and 20 multigrid cycles, respectively, on a sequence of 5 grids. It is seen that the distributions of pressure and skin friction coefficient agree well with the fully-converged values after about 10 cycles, while more than 20 cycles are required for good agreement in the distribution of total enthalpy.

It is clear that the convergence rates are not as good as for the Euler equations, and are deteriorating somewhat on successively finer grids. Reasons for this degradation in performance, relative to the Euler equation results, continue to be investigated. Here, one final result is presented. The viscous calculation is repeated at a Reynolds number of $\mathbf{Re} = 80\,000$, with the normal spacing of the first row of grid cells adjacent to the wall reduced to one quarter that of the earlier grid (since the boundary layer should be approximately $\frac{1}{4}$ as thick at the higher Reynolds number). Thus, approximately the same number of grid cells should lie within the boundary layer for both flows, but the maximum grid aspect ratio, which occurs on the wake cut at the downstream boundary, is four times that of the earlier computation. The convergence rates for this computation, presented in Plate 6, indicate that the degradation in convergence rate is due, at least in part, to the larger aspect ratio grid cells required for the viscous computations. We continue to investigate alternatives to semi-coarsening [3], which is computationally expensive and problematic in complex geometries, to overcome this sensitivity to grid aspect ratio on highly stretched grids.

The results displayed in this section indicate that the asymptotic convergence rates of the present method for the Euler equations are significantly faster than previous results using the well-tuned Runge–Kutta multigrid method of Jameson [1] and the diagonalized implicit method of Caughey [4, 5]. In addition, the CPU time per multigrid cycle is as much as 50% less than that of the previous methods (including those based on explicit Runge–Kutta integration), indicating that converged solutions can be obtained with the current method with almost an order of magnitude less computer time than those methods.

4. CONCLUDING REMARKS

Results of computations verifying the efficiency of a new preconditioned, LU-SGS implicit multigrid method for solving the Euler equations have been presented. Results presented in this paper have focused on implementations using ‘C’-type grids, computed using the CUSP version of the SLIP spatial discretizations and on extensions to viscous flows. The present results indicate that two-dimensional Euler solutions, converged to approximately the level of truncation error, can be obtained on fine grids in three to five multigrid cycles for several commonly used transonic benchmark cases.

The results presented here approach the goal of ‘textbook’ multigrid efficiency, at least for inviscid flows. We continue to work on extensions to three-dimensional flows, and to

improve the performance of the algorithm for the solution of viscous flows modelled by the Reynolds-averaged Navier–Stokes (RANS) equations. While we are very encouraged by the results to date, we recognize that solution of the RANS equations presents a much more severe challenge.

The progress of our work has been greatly facilitated by the extraordinary advances in computer technology over the past decade. The algorithm development and numerical tests have been carried out entirely on the laptop computers that we each possess, running the Linux OS. In fact, the Sony VAIO 505 owned by the second author, which has a 750 MHz Pentium III processor, is quite sufficient for three-dimensional flow calculations. The Sony VAIO GR-100 used by the first author, which has a 1 GHz Pentium III processor, completes Euler computations for 5 full multigrid cycles on the standard 160×32 cell grid in less than 2.5 s of CPU time, and interactive graphical display of the evolution of the solution has been helpful in program development. We have benefited equally from the ease with which we can exchange information and code updates over the Internet. All this provides an environment for algorithm development entirely different from that which prevailed during most of our careers and which has allowed us to collaborate easily in spite of our being at opposite ends of the continent.

ACKNOWLEDGEMENTS

The second author has benefited greatly from the continuing support of the AFOSR under Grant No. AF F49620-98-1-022. While the principal goal of this grant is improved methods for aerodynamic shape optimization, these depend critically on the availability of fast solution algorithms, and this has been one of the prime motivations of the work described herein.

REFERENCES

1. Jameson A. Solution of the Euler equations for two-dimensional, transonic flow by a multigrid method. *Applied Mathematics and Computation* 1983; **13**:327–356.
2. Jameson A, Schmidt W, Turkel E. Numerical solutions of the Euler equations by finite-volume methods using Runge-Kutta time-stepping schemes. *AIAA Paper 81-1259*, 1981.
3. Pierce NA, Giles MB. Preconditioned multigrid methods for compressible flow codes on stretched meshes. *Journal of Computational Physics* 1997; **136**:425–445.
4. Caughey DA. Diagonal implicit multigrid algorithm for the Euler equations. *AIAA Journal* 1988; **26**:841–851.
5. Caughey DA. Implicit multigrid Euler solutions with symmetric total variation diminishing dissipation. *Proceedings of AIAA 11th CFD Conference*, 1993; 676–684.
6. Jameson A, Yoon S. Multigrid solutions of the Euler equations using implicit schemes. *AIAA Journal* 1986; **24**:1737–1743.
7. Rieger H, Jameson A. Solution of steady three-dimensional compressible Euler and Navier-Stokes equations by an implicit LU scheme. *AIAA Paper 88-0619*, 1988.
8. Yoon S, Jameson A. Lower-upper symmetric-Gauss-Seidel method for the Euler and Navier-Stokes equations. *AIAA Journal* 1988; **26**:1025–1026.
9. Yokota JW, Caughey DA. *LU* Implicit multigrid algorithm for the three-dimensional Euler equations. *AIAA Journal* 1988; **26**:1061–1069.
10. Brandt A. Multi-level adaptive solution to boundary value problems. *Mathematics of Computation* 1977; **31**(138):333–390.
11. Brandt A. Multigrid Techniques: 1984 Guide with Applications to Fluid Dynamics. *GMD-Studie*, vol. 85. GMD-FIT, 1985.
12. Roberts TW, Sidilkover D, Swanson RC. Textbook multigrid efficiency for the steady Euler equations. *AIAA Paper 97-1949*, 1997.
13. Roberts TW, Sidilkover D, Thomas JL. Multigrid relaxation of a factorizable, conservative discretization of the compressible flow equations. *AIAA Paper 2000-2252*, 2000.
14. Ta'asan S. Canonical-variables multigrid method for steady-state Euler equations. *ICASE Report 94-14*, 1994.

15. Beam RM, Warming RF. An implicit finite-difference algorithm for hyperbolic systems in conservation law form. *Journal of Computational Physics* 1976; **22**:87–110.
16. Beam RM, Warming RF. An implicit factored scheme for the compressible Navier-Stokes equations. *AIAA Journal* 1978; **16**:393–402.
17. Briley WR, McDonald H. *Solution of the Three-Dimensional Compressible Navier Stokes Equations by an Implicit Technique*. Lecture Notes in Physics, vol. 35. Springer: New York, 1974; 105–110.
18. Jameson A, Turkel E. Implicit schemes and LU decompositions. *Mathematics of Computation* 1981; **37**: 385–397.
19. Buratynski EW, Caughey DA. An implicit LU scheme for the Euler equations applied to arbitrary cascades. *AIAA Journal* 1986; **24**:39–46.
20. MacCormack RW. A numerical method for solving the equations of compressible. Viscous flow. *AIAA Paper 81-0110*, 1981.
21. Allmaras S. Analysis of a local matrix preconditioner for the 2-D Navier-Stokes equations. *AIAA Paper 93-3330*, 1993.
22. Bardina J, Lombard CK. Three-dimensional hypersonic flow simulations with the CSCM implicit upwind Navier-Stokes method. *AIAA Paper 87-1114*, 1987.
23. MacCormack RW. A new implicit algorithm for fluid flow. *Proceedings of AIAA 13th CFD Conference*, Snowmass, Colorado, 1997; 112–119.
24. Jameson A. Analysis and design of numerical schemes for gas dynamics 1: artificial diffusion, upwind biasing, limiters and their effect on accuracy and multigrid convergence. *International Journal of Computational Fluid Dynamics* 1995; **4**:171–218.
25. Jameson A. Analysis and design of numerical schemes for gas dynamics 2: artificial diffusion and discrete shock structure. *International Journal of Computational Fluid Dynamics* 1995; **5**:1–38.
26. Warming RF, Beam RM, Hyett BJ. Diagonalization and simultaneous symmetrization of the gas dynamic equations. *Mathematics of Computation* 1975; **29**:1037–1045.
27. Abarbanel S, Gottleib D. Optimal time splitting for two- and three-dimensional Navier–Stokes equations with mixed derivatives. *Journal of Computational Physics* 1981; **41**:1–33.
28. Jameson A, Caughey DA. How many steps are required to solve the Euler equations of steady, compressible flow: in search of a fast solution algorithm. *AIAA Paper 2001-2673*, AIAA 15th Computational Fluid Dynamics Conference, June 11–14, Anaheim, California, 2001.

