VISCOUS AIRFOIL COMPUTATIONS USING LOCAL PRECONDITIONING

D. Unrau*   D.W. Zingg†

University of Toronto Institute for Aerospace Studies
4925 Dufferin Street, Downsview, Ontario
Canada M3H 5T6

Abstract

A local preconditioning technique for compressible flows at low Mach number is examined in the context of turbulent flows over airfoils. The preconditioning technique is implemented in an approximately-factored algorithm for the steady thin-layer Navier-Stokes equations. Parametric studies are presented showing trade-offs between accuracy and convergence rates. Excellent convergence rates and solution accuracy are demonstrated for freestream Mach numbers as low as 0.01. Improvements in accuracy, especially in drag prediction, are seen for Mach numbers as high as 0.4. Thus the present approach is useful in the computation of low-speed airfoil flows, including those typically encountered during take-off and landing of aircraft.

Introduction

When a compressible flow solver is applied to a nearly incompressible flow, its performance can deteriorate in terms of both speed and accuracy. [1, 2] Local preconditioning techniques have been developed to address this difficulty in order to provide the capability of efficiently solving flows which have large regions of low Mach number flow but nevertheless include significant compressibility effects. The flow over an airfoil at high lift is an example of such a situation. Although the freestream Mach number is often less than 0.2, compressibility effects can be important locally and the onset of stall can be related to the formation of shock waves.

Difficulties with low Mach number flows are caused by the associated wide range of wave speeds, which can result in poor convergence of an iterative method. Local preconditioning is designed to minimize the range of wave speeds, while maintaining the steady-state solution. In addition, the numerical dissipation in the spatial discretization, arising either from some form of upwinding or artificial dissipation, generally scales badly as the Mach number is reduced. If the flux splitting or artificial dissipation is based on the preconditioned system, improved steady-state solutions can be obtained.

Early developments in local preconditioning were reviewed by Turkel, [3] including contributions from Choi and Merkle, [4] Turkel, [5] and van Leer at al. [6] More recently, Weiss and Smith [7] and Godfrey [8] have demonstrated good results for turbulent flows. These developments have been based primarily on two ideas. One is to minimize the range of wave speeds. The other is to design the preconditioner such that the governing equations approach the pseudo-compressible equations for incompressible flow [10] as the Mach number is reduced. The latter property can be obtained naturally from a pressure-based formulation. [9] A somewhat different approach to local preconditioning has been developed by Allmaras. [11]

The purpose of the present paper is to implement and study a local preconditioning technique, that of Weiss and Smith, [7] in the context of thin-layer Navier-Stokes computations of low-speed airfoil flows using an approximately-factored algorithm. After a brief description of the numerical algorithm and governing equations, the implementation of the local preconditioning technique is described in detail. This is followed by a section describing studies to determine appropriate values of various parameters. Finally, convergence histories and solutions are presented for several cases, including those previously studied in detail in Refs. 12 and 13. See Ref. 14 for related work.

*Graduate Student.
†Associate Professor, Member AIAA.

Governing Equations and Numerical Method

The local preconditioning technique of Weiss and Smith has been implemented in the thin-layer Navier-Stokes solver ARC2D.[15] This solver uses second-order centered differences in space together with a scalar nonlinear artificial dissipation model combining second- and fourth-difference terms. Convergence to steady state is achieved using the diagonal form of the Beam and Warming approximate-factorization algorithm with local time stepping.

In two-dimensional generalized coordinates, the thin-layer Navier-Stokes equations are given by[15]

$$\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial \xi} + \frac{\partial F}{\partial \eta} = \mathrm{Re}^{-1}\frac{\partial S}{\partial \eta} \tag{1}$$

where $Q = J^{-1}[\rho, \rho u, \rho v, e]^T$ is the vector of conservative dependent variables, E and F are the convective flux vectors, S is the viscous flux vector, ξ and η are the streamwise and normal generalized coordinates, respectively, J is the Jacobian of the coordinate transformation, and Re is the freestream Reynolds number based on the speed of sound. Applying the spatial operator, linearizing in time, and applying the implicit Euler time-marching method with an approximate factorization of the left-hand-side operator gives the following form

$$[I+\Delta t\delta_\xi A^n][I+\Delta t\delta_\eta B^n - \Delta t\mathrm{Re}^{-1}\delta_\eta K^n]\Delta Q^n = R^n \tag{2}$$

where

$$Q^n = Q(n\Delta t)$$

$$\Delta Q^n = Q^{n+1} - Q^n$$

$$R^n \;=\; -\Delta t[\delta_\xi E(Q^n) + \delta_\eta F(Q^n) - \mathrm{Re}^{-1}\delta_\eta S(Q^n)]$$

and the matrices A, B, and K are the flux Jacobians, defined by

$$A = \frac{\partial E}{\partial Q}, \quad B = \frac{\partial F}{\partial Q}, \quad K = \frac{\partial S}{\partial Q} \tag{3}$$

The symbol δ denotes the complete spatial operator, including the artificial dissipation. Finally, the diagonal form of the algorithm is given by[15]

$$T_\xi[I + \Delta t\delta_\xi \Lambda_\xi]N[I + \Delta t\delta_\eta \Lambda_\eta]T_\eta^{-1}\Delta Q^n = R^n \tag{4}$$

where

$$N = T_\xi^{-1}T_\eta \tag{5}$$

and the matrices Λ_ξ and Λ_η are diagonal matrices whose elements are the eigenvalues of the flux Jacobians, including a contribution from the viscous flux Jacobian.[15] The matrix T_ξ has the eigenvectors of A as columns, and T_η has the eigenvectors of B as columns.

Implementation of Local Preconditioning

The flux Jacobian matrices A and B have eigenvalues $U_\xi, U_\xi, U_\xi \pm c\sqrt{{\xi_x}^2 + {\xi_y}^2}$ and $U_\eta, U_\eta, U_\eta \pm c\sqrt{{\eta_x}^2 + {\eta_y}^2}$, respectively, where U_ξ and U_η are contravariant velocity components in the ξ and η directions, and c is the speed of sound. As the Mach number approaches zero, these eigenvalues have widely differing magnitudes. The local preconditioning matrix Γ is introduced as follows

$$\Gamma\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial \xi} + \frac{\partial F}{\partial \eta} = \mathrm{Re}^{-1}\frac{\partial S}{\partial \eta} \tag{6}$$

This time-derivative preconditioning destroys the time accuracy of the system, but as long as both Γ and Γ^{-1} exist, the steady state solution is, at least potentially, unaffected. Writing the inviscid terms in quasi-linear form and premultiplying by Γ^{-1} gives

$$\frac{\partial Q}{\partial t} + \Gamma^{-1}A\frac{\partial Q}{\partial \xi} + \Gamma^{-1}B\frac{\partial Q}{\partial \eta} = \mathrm{Re}^{-1}\Gamma^{-1}\frac{\partial S}{\partial \eta} \tag{7}$$

Thus the preconditioned flux Jacobians are $\Gamma^{-1}A$ and $\Gamma^{-1}B$. With a suitable choice of Γ, the spread of the eigenvalues of the preconditioned flux Jacobians can be substantially reduced.

Symmetry Variables

It is convenient to introduce the symmetry variables given by

$$\partial\hat{Q} = J^{-1}\begin{bmatrix} \frac{1}{\rho c}\partial p \\ \partial u \\ \partial v \\ \partial p - c^2\partial\rho \end{bmatrix}. \tag{8}$$

Temporarily ignoring the viscous terms, we can write the locally-preconditioned Euler equations in terms of the symmetry variables as follows

$$\hat{\Gamma}\frac{\partial \hat{Q}}{\partial t} + \hat{A}\frac{\partial \hat{Q}}{\partial \xi} + \hat{B}\frac{\partial \hat{Q}}{\partial \eta} = 0 \tag{9}$$

where $\hat{\Gamma}$ is the local preconditioner formulated in terms of the symmetry variables, and the inviscid flux Jacobians for the symmetry variables are

$$\hat{A},\ \hat{B} = \begin{bmatrix} U_\kappa & \kappa_x c & \kappa_y c & 0 \\ \kappa_x c & U_\kappa & 0 & 0 \\ \kappa_y c & 0 & U_\kappa & 0 \\ 0 & 0 & 0 & U_\kappa \end{bmatrix} \tag{10}$$

where $\kappa = \xi$ gives $\hat{A}$, and $\kappa = \eta$ gives $\hat{B}$. The symmetry variables are related to the conservative variables through the transfer matrices given by

$$M = \frac{\partial Q}{\partial \hat{Q}}, \; M^{-1} = \frac{\partial \hat{Q}}{\partial Q} \tag{11}$$

which are given in the Appendix.

The local preconditioning matrices and the flux Jacobians in the two variable sets are related by

$$\Gamma = M\hat{\Gamma}M^{-1} \tag{12}$$
$$A = M\hat{A}M^{-1} \tag{13}$$
$$B = M\hat{B}M^{-1} \tag{14}$$

Thus a local preconditioner formulated in terms of the symmetry variables can be applied to the thin-layer Navier-Stokes equations written in terms of the conservative variables as follows

$$M\hat{\Gamma}M^{-1}\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial \xi} + \frac{\partial F}{\partial \eta} = \mathrm{Re}^{-1}\frac{\partial S}{\partial \eta} \tag{15}$$

Similarly, the diagonal form of the algorithm, eq. 4, can be written in terms of the eigenvectors of the preconditioned flux Jacobians written in terms of the symmetry variables. Referring to equations 12 to 14, it is easy to show that:

$$M\hat{\Gamma}^{-1}\hat{A}M^{-1} = \Gamma^{-1}A \tag{16}$$

and a similar relation holds for $\hat{\Gamma}^{-1}\hat{B}$ and $\Gamma^{-1}B$. Since $\hat{\Gamma}^{-1}\hat{A}$ and $\Gamma^{-1}A$ have the same eigenvalues, their eigenvector matrices must be related by

$$M\hat{T}_{\xi,\Gamma} = T_{\xi,\Gamma} \tag{17}$$

where the subscript Γ indicates the preconditioned system. A similar relation holds for the eigenvectors of $\hat{\Gamma}^{-1}\hat{B}$ and $\Gamma^{-1}B$:

$$M\hat{T}_{\eta,\Gamma} = T_{\eta,\Gamma} \tag{18}$$

With these identities, the diagonal form of the algorithm becomes:

$$\begin{aligned} &M\hat{\Gamma}\hat{T}_{\xi,\Gamma}\left[I + \Delta t\delta_\xi \Lambda_{\xi,\Gamma}\right]\hat{T}^{-1}_{\xi,\Gamma}\hat{T}_{\eta,\Gamma} \\ &\cdot \left[I + \Delta t\partial_\eta(\Lambda_{\eta,\Gamma} - \mathrm{Re}^{-1}N_\eta)\right]\hat{T}^{-1}_{\eta,\Gamma}M^{-1}\Delta Q = R^n \end{aligned} \tag{19}$$

where $\Lambda_{\kappa,\Gamma}$ is a diagonal matrix containing the eigenvalues of $\hat{\Gamma}^{-1}\hat{A}$ and $\hat{\Gamma}^{-1}\hat{B}$ with $\kappa = \xi$ and η, respectively, and R^n is as given below eq. 2. The term N_η is a diagonal contribution from the viscous terms which we will define later.

Weiss-Smith Preconditioner

When written in terms of the symmetry variable, the preconditioner of Weiss and Smith [7] has a particularly simple form:

$$\hat{\Gamma} = \begin{bmatrix} \frac{1}{\epsilon} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \; \hat{\Gamma}^{-1} = \begin{bmatrix} \epsilon & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{20}$$

where ϵ will be discussed later. With this preconditioner, the following eigenvalue matrices are obtained:

$$\Lambda_{\kappa,\Gamma} = \begin{bmatrix} U_\kappa & 0 & 0 & 0 \\ 0 & U_\kappa & 0 & 0 \\ 0 & 0 & \lambda_{\kappa,+} & 0 \\ 0 & 0 & 0 & \lambda_{\kappa,-} \end{bmatrix} \tag{21}$$

The eigenvalues $\lambda_{\kappa,\pm}$ are the roots of:

$$\lambda_{\kappa,\pm}{}^2 - (\epsilon+1)U_\kappa\lambda_{\kappa,\pm} + \epsilon\left(U_\kappa^2 - \left(\kappa_x^2 + \kappa_y^2\right)c^2\right) = 0 \tag{22}$$

Hence they are

$$\lambda_{\kappa,\pm} = \frac{1}{2}(1+\epsilon)U_\kappa \pm \frac{1}{2}\sqrt{(\epsilon-1)^2U_\kappa^2 + 4\epsilon\left(\kappa_x^2+\kappa_y^2\right)c^2} \tag{23}$$

One normalization of the eigenvector matrices, $\hat{T}_{\kappa,\Gamma}$ is:

$$\hat{T}_{\kappa,\Gamma} = \begin{bmatrix} 0 & 0 & \frac{\lambda_{\kappa,+} - U_\kappa}{\bar{\kappa}c} & \frac{U_\kappa - \lambda_{\kappa,-}}{\bar{\kappa}c} \\ 0 & \tilde{\kappa}_y & \tilde{\kappa}_x & -\tilde{\kappa}_x \\ 0 & -\tilde{\kappa}_x & \tilde{\kappa}_y & -\tilde{\kappa}_y \\ 1 & 0 & 0 & 0 \end{bmatrix}, \tag{24}$$

where $\tilde{\kappa} = \kappa/\bar{\kappa}$ and $\bar{\kappa} = \sqrt{2(\kappa_x^2 + \kappa_y^2)}$. As $\epsilon \to 0$, the condition numbers of the preconditioned flux Jacobians remain finite. If $\epsilon = 1$, the preconditioner becomes the identity matrix, and the algorithm reverts to the unpreconditioned form.

Finally, the approximate linearization of the viscous terms, N_η, is determined from

$$\begin{aligned} N_\eta &= \mathrm{diag}\left(\hat{T}^{-1}_{\eta,\Gamma}\hat{\Gamma}^{-1}M^{-1}k_{\mathrm{ARC2D}}M\hat{T}_{\eta,\Gamma}\right) \\ &= k_{\mathrm{ARC2D}}\mathrm{diag}\left(\hat{T}^{-1}_{\eta,\Gamma}\hat{\Gamma}^{-1}\hat{T}_{\eta,\Gamma}\right) \end{aligned} \tag{25}$$

where k_{ARC2D} is the scalar term used in ARC2D. This gives

$$N_\eta = k_{\mathrm{ARC2D}}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \beta + \frac{\epsilon+1}{2} & 0 \\ 0 & 0 & 0 & -\beta + \frac{\epsilon+1}{2} \end{bmatrix} \tag{26}$$

where

$$\beta = \frac{(\epsilon-1)^2U_\eta}{2\sqrt{(1-\epsilon)^2U_\eta^2 + 4\epsilon(\eta_x^2+\eta_y^2)c^2}} \tag{27}$$

Once again, this matrix reverts to the unpreconditioned form when $\epsilon = 1$.

Artificial Dissipation

Whether a scalar or matrix model is used, the artificial dissipation should be based on the preconditioned system, due to its superior conditioning. Here we use a scalar model based on the spectral radii of the preconditioned flux Jacobians, given by

$$\sigma_\kappa = \frac{1}{2}(1+\epsilon)\left|U_\kappa\right| + \frac{1}{2}\sqrt{(\epsilon-1)^2 U_\kappa^2 + 4\epsilon\left(\kappa_x^2+\kappa_y^2\right)c^2} \tag{28}$$

where σ_κ denotes the spectral radii of $\Gamma^{-1}A$ and $\Gamma^{-1}B$ with $\kappa = \xi$ and η, respectively. Note that the difficulties associated with the scalar model, discussed for example in Ref. 13, are greatly reduced due to the improved conditioning of the preconditioned flux Jacobian matrices.

The second- and fourth-difference artificial dissipation terms on the right-hand-side of eq. 19 are in the form

$$D^{(2)}_{\kappa,\Gamma} = \nabla_\kappa \Gamma k^{(2)} \Delta_\kappa \tilde{Q} \tag{29}$$

$$D^{(4)}_{\kappa,\Gamma} = -\nabla_\kappa \Gamma k^{(4)} \Delta_\kappa \nabla_\kappa \Delta_\kappa \tilde{Q} \tag{30}$$

where $k^{(2)}$ and $k^{(4)}$ contain the preconditioned spectral radii and the standard switches. On the left-hand-side, the preconditioned spectral radii are used in place of the original spectral radii.

Far-Field Boundary Conditions

The far-field boundary treatment must be modified to incorporate the characteristic variables (or Riemann invariants) of the preconditioned system. We use the approach of Barth [16] based on characteristic projectors. At the boundary nodes, Q is determined as follows:

$$\begin{aligned} Q &= \frac{1}{2}\left(Q_\infty + Q_{ext}\right) \\ &- \frac{1}{2} M \hat{T}_{\kappa,\Gamma} \operatorname{sign}\left(\Lambda_{\kappa,\Gamma}\right) \hat{\mathrm{T}}^{-1}_{\kappa,\Gamma} \mathrm{M}^{-1}\left(\mathrm{Q}_\infty - \mathrm{Q}_{ext}\right) \end{aligned} \tag{31}$$

where κ is chosen in the direction normal to the boundary, ∞ indicates values obtained from free stream conditions, and *ext* indicates values extrapolated from the interior nodes of the mesh.

The Choice of ϵ

Note that the Weiss-Smith preconditioner only alters the values of the third and fourth eigenvalues of the inviscid flux Jacobians. This imposes a major limitation on the performance of the preconditioner, as there is a requirement that the sign of either the third or the fourth eigenvalue must change at a Mach number of unity, giving all eigenvalues the same sign for supersonic flows. This is a reflection of the physical requirement that information cannot flow upstream in supersonic flows, and is necessary to produce physically consistent results based on the standard application of boundary values. This requirement means that the stiffness at the sonic point cannot be removed with this preconditioner. Hence this preconditioner is primarily intended for use as a low Mach number preconditioner, in contrast to that of Van Leer et al., [6] for example. Consequently, ϵ is chosen such that $\epsilon = 1$ for Mach numbers greater than one. Thus the preconditioner reduces to the identity matrix for supersonic flows, and the original system of equations is recovered.

With $\epsilon = \mathrm{Ma}^2$, where Ma is the local Mach number, the Weiss-Smith preconditioner provides effective low Mach number preconditioning. However, ϵ must remain finite in stagnation regions. To this end, a lower limit is placed on ϵ based on the freestream Mach number:

$$\epsilon_1 = \phi \mathrm{Ma}_\infty^2 \tag{32}$$

where ϕ is a free parameter. In addition, the following lower limit is sometimes imposed:

$$\epsilon_2 = \chi \frac{\mu_\infty \sqrt{\max\left(\xi_x^2+\xi_y^2, \eta_x^2+\eta_y^2\right)}}{\rho u_\infty} \tag{33}$$

where χ is the second free parameter associated with the preconditioner. This prevents the preconditioned pseudo-acoustic velocities from becoming too small relative to the local diffusion velocity, $\nu/\Delta x$. When these limits are combined, the determination of ϵ is implemented as follows:

$$\epsilon = \min\left[1, \max\left(\mathrm{Ma}^2, \epsilon_1, \epsilon_2\right)\right] \tag{34}$$

The Choice of Δt

The choice of the local time step, Δt, is critical to exploit the gains of local preconditioning. As the Weiss-Smith preconditioner brings the acoustic speed in the flow to the same order as the convective flow speed, the characteristic speeds, while uniform across equations, vary widely between stagnation and freestream conditions. For this reason, a constant Courant number is imposed that sets the *minimum* Courant number at each node to the user specified value. However, in low Reynolds number regions this can lead to a time step which is too large. Hence a maximum von Neumann number $(\mu\Delta t/\rho\Delta x^2)$ of unity is also imposed. This leads to the following definition of Δt:

$$\Delta t = \min\left[\frac{\rho}{\mu_\infty \max\left(\xi_x^2+\xi_y^2, \eta_x^2+\eta_y^2\right)}, \mathrm{CFL} \times \max\left(\frac{1}{\sigma_\xi}, \frac{1}{\sigma_\eta}\right)\right] \tag{35}$$

where σ_ξ and σ_η are determined from eq. 28, and CFL is user specified.

Parametric Studies

The following parameters are considered:

- The value of the CFL number in the time step definition, eq. 35.
- The values of ϕ and χ in the determination of ϵ, eq. 34.

The choice of the CFL number affects only the convergence rate, while the values of ϕ and χ affect both the convergence rate and the accuracy of the converged solution through their impact on the artificial dissipation.

The convergence rate can be assessed by studying either the behavior of a residual or the actual convergence of lift and drag. A CFL number equal to 11 gives near optimal drag convergence for a broad range of cases. For more difficult cases involving high angles of attack or extremely fine grids, CFL values in the range 7-9 are preferable.

With ϕ and χ, the situation is more complicated because there can be a trade-off between convergence and accuracy. From an accuracy perspective, one would like to keep both ϕ and χ as small as possible. If they are too small, however, the code may not converge. Hence there is a degree of flexibility in selecting these parameters, which are, in any case, problem dependent.

The influence of ϕ is demonstrated for a test case consisting of the NACA 0012 airfoil at an angle of attack of two degrees and a Reynolds number of 2.88 million. The grid used has 249 $\times$ 49 nodes with an off-wall spacing of 2×10^{-6} chords. Experiments with finer grids suggest that the drag coefficient should be roughly 0.0085 for this case with little dependence on the freestream Mach number for values below about 0.2. Figure 1 shows the dependence of the computed drag coefficient on ϕ for freestream Mach numbers ranging from 0.01 to 0.20, demonstrating that increasing ϕ increases the error. Good convergence rates are often obtained with ϕ equal to unity. Reducing ϕ further improves drag accuracy but increases the number of iterations required to achieve convergence of the drag. More difficult cases can require increased values of ϕ, up to a maximum of 3. For example, a value of ϕ equal to 3 is required to obtain convergence for the same conditions at an angle of attack of 12 degrees.

The dependence of the computed drag coefficient in χ is shown in Figure 2. It can be seen that increasing χ increases the error, particularly at $\mathrm{Ma}_\infty = 0.01$. For the cases studied, a nonzero value of χ did not produce an improvement in the convergence rate; hence $\chi = 0$ is recommended for aerodynamic computations.

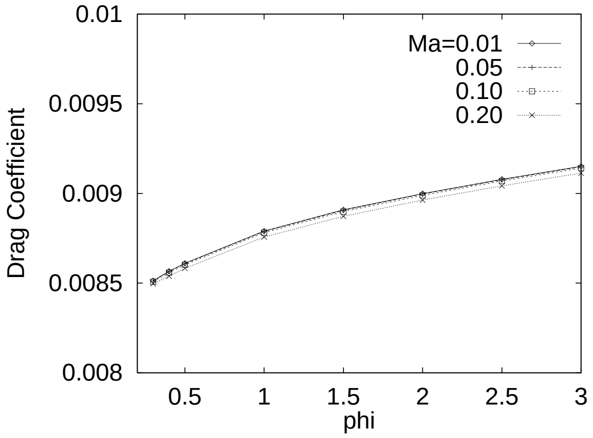

Figure 1: The variation of drag with ϕ for CFL = 11 and $\chi = 0$.

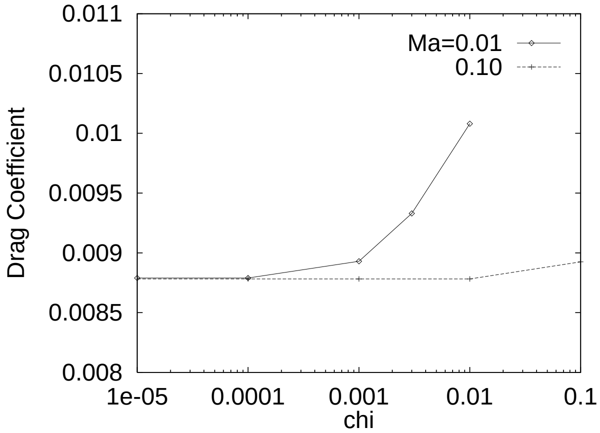

Figure 2: The variation of drag with χ for CFL = 11 and $\phi = 1.0$.

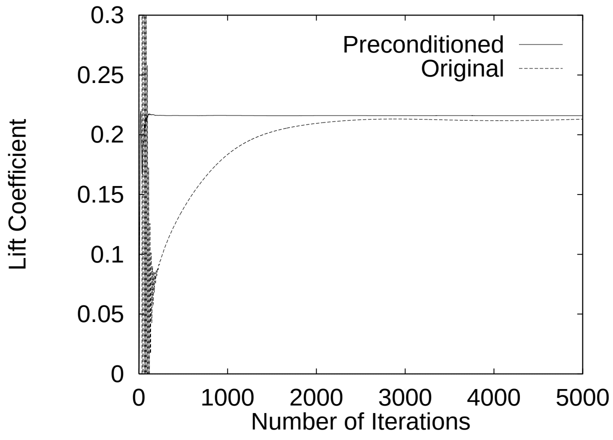

Figure 3: The convergence of lift with $\mathrm{Ma}_\infty = 0.01$, $\mathrm{CFL} = 11$, $\phi = 1$, $\chi = 0$.

Results

In this section, we compare the performance of the preconditioned algorithm with that of the original algorithm, using the parameter values selected above. In each case, the unpreconditioned algorithm has been run using local time stepping with a time step choice which is optimal for that particular case. The preconditioned algorithm was used with $\mathrm{CFL} = 11$, $\phi = 1$, and $\chi = 0$.

Figures 3 and 4 show the convergence of lift and drag for the airfoil, grid, and Reynolds number as above with $\alpha = 2^o$, $\mathrm{Ma}_\infty = 0.01$. At this Mach number, the benefits of local preconditioning are enormous. The preconditioned algorithm produced lift and drag coefficients within 0.1% of the final converged values in 250 and 247 iterations, respectively. The original algorithm required over 5,000 iterations to achieve the same degree of convergence. The preconditioned algorithm is also much more accurate, as shown in Figure 5, which displays the computed surface pressure distributions. The original algorithm produces visible aphysical oscillations which are completely removed by the preconditioning. The drag coefficient computed without the preconditioning is 0.02604, roughly three times that computed with the preconditioning, 0.00879, as shown in Figure 4, with the latter much closer to the grid-independent result. These results clearly show that the preconditioned algorithm is capable of accurately and efficiently computing flows which are effectively incompressible.

At higher Mach numbers, more relevant to aerodynamic flows, the benefits of local preconditioning are not quite as spectacular but nevertheless remain significant. Figures 6 and 7 show the convergence histories obtained for the same case with the same parameters and $\mathrm{Ma}_\infty = 0.2$, a Mach number which is fairly typical of high-lift flows. For this case, the preconditioned algo-

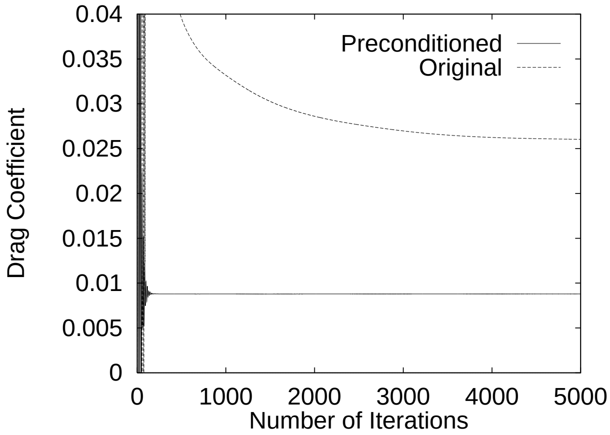

Figure 4: The convergence of drag with $\mathrm{Ma}_\infty = 0.01$, $\mathrm{CFL} = 11$, $\phi = 1$, $\chi = 0$.

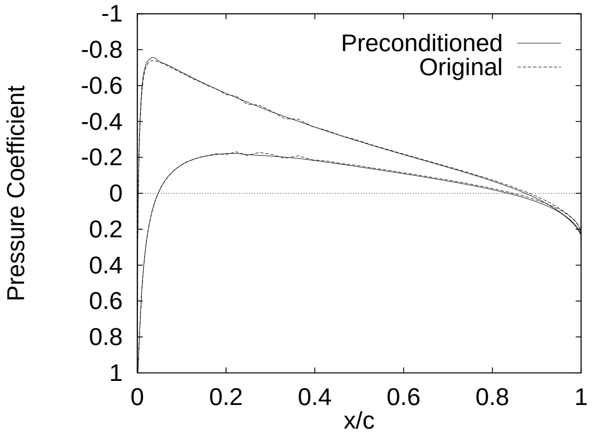

Figure 5: The pressure distribution with $\mathrm{Ma}_\infty = 0.01$, $\phi = 1$, $\chi = 0$.

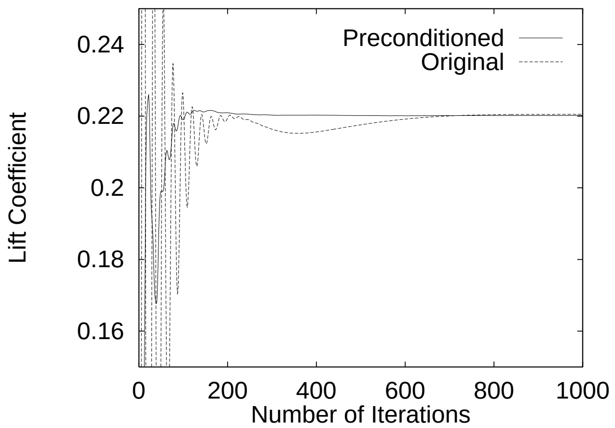

Figure 6: The convergence of lift with $\mathrm{Ma}_\infty = 0.2$, $\mathrm{CFL} = 11$, $\phi = 1$, $\chi = 0$.

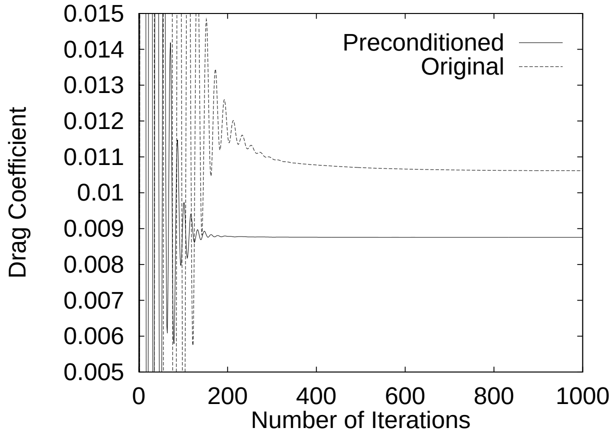

Figure 7: The convergence of drag with $\mathrm{Ma}_\infty = 0.2$, $\mathrm{CFL} = 11$, $\phi = 1$, $\chi = 0$.

rithm produces lift and drag coefficients within 0.1% of the final values in about 280 iterations, while the original algorithm requires roughly 800 iterations. Figure 8 shows the number of iterations required to achieve this level of convergence as a function of the freestream Mach number with $\phi = 1$, $\chi = 0$. Local preconditioning is beneficial up to about $\mathrm{Ma}_\infty = 0.3$.

At freestream Mach numbers ranging from 0.1 to 0.2, the improvements in accuracy resulting from local preconditioning are not quite as obvious as those shown in Figure 5. However, the solutions are indeed more accurate, and this is best seen by examining the drag coefficients. Figure 9 shows the variation of the computed drag coefficient with the freestream Mach number. The preconditioned results show the expected lack of dependence on the Mach number, while the results without preconditioning show large errors which increase as the Mach number is decreased.

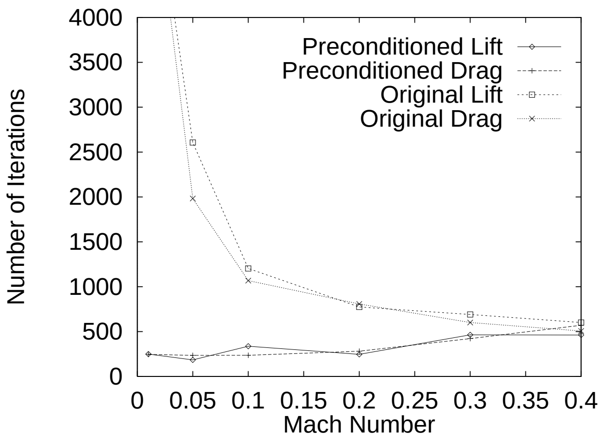

Figure 8: The number of iterations required to converge the lift and drag to within 0.1% of the converged values as a function of the freestream Mach number.

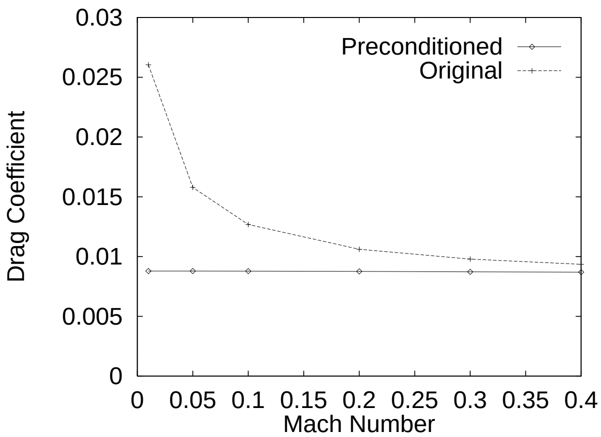

Figure 9: The variation of drag with Ma_∞ for $\mathrm{CFL} = 11$, $\phi = 1$, and $\chi = 0$.

Finally, we consider two flow cases involving the NACA 0012 airfoil at $\mathrm{Re} = 2.88 \times 10^6$, $\mathrm{Ma}_\infty = 0.16$, previously studied by Zingg [12] and Frew and Zingg. [13] For the first case, with $\alpha = 0^o$ and laminar-turbulent transition at 43% chord on both surfaces, Zingg [12] determined a reasonably grid-independent drag coefficient of 0.00579, using very fine grids and Richardson extrapolation. Using the scalar artificial dissipation model without preconditioning and the grid described above, a drag coefficient of 0.00823 is computed, an error of 24 counts or 42%. Applying the local preconditioning with $\phi = 0.5$ and $\chi = 0$, the drag coefficient is reduced to 0.00599, an error of only two counts or roughly 3.5%. Using a grid with the same characteristics but an off-wall spacing which is ten times smaller (2×10^7 chords), the unpreconditioned algorithm produces a drag coefficient of 0.00660, an error of 8 counts or 14%. The preconditioned algorithm requires $\phi = 1.7$, CFL = 9, and produces a drag coefficient of 0.00581, an error of 0.2 counts or less than 0.4%. It should be noted that the large errors resulting from the scalar artificial dissipation model without preconditioning can also be reduced using matrix dissipation or cell Reynolds number scaling. [13]

For the second case, with $\alpha = 6^o$ and transition at 5% and 80% chord on the upper and lower surfaces, respectively, a grid-independent drag coefficient of 0.00779 was determined. [12] Using the grid with the smaller off-wall spacing, a drag coefficient of 0.01018 is computed by the unpreconditioned algorithm, an error of 24 counts or 31%. With local preconditioning, using $\phi = 3$, CFL = 9, the computed drag coefficient is 0.00923, an error of 14 counts or 18%. The error reduction is not quite as dramatic because the error is not associated with overdissipation to the same degree as with $\alpha = 0^o$, as observed by Frew and Zingg. [13]

With our current implementation, the local preconditioning increases the CPU time per iteration by about 30%. This could easily be reduced further. Figure 8 shows that the local preconditioning ceases to provide improved *efficiency* at about $\mathrm{Ma}_\infty = 0.4$ and becomes detrimental at higher Mach numbers because of its extra cost. However, improvements in *accuracy* continue as the Mach number is increased. This results primarily from the fact that the local preconditioning greatly improves the accuracy of the spectral radius approximation in the artificial dissipation model near the airfoil surface, where the Mach number is low. Consequently errors in drag are reduced, especially friction drag, even at moderate freestream Mach numbers. Further study is required to examine the effectiveness of matrix artificial dissipation in conjunction with local preconditioning.

Conclusions

We have presented an implementation of the Weiss-Smith local preconditioner in an approximately-factored algorithm for the thin-layer Navier-Stokes equations using centered differences with a scalar artificial dissipation model. The preconditioned algorithm has been applied to a range of turbulent flows over airfoils. Studies have been presented showing the impact of various parameters, and appropriate values are suggested. The results show this preconditioner to be extremely effective in improving both the speed and the accuracy of computations at low Mach number. Convergence rates and accuracy are virtually independent of the freestream Mach number. It is noteworthy that significant improvements are obtained for Mach numbers as high as 0.2, which is certainly in the practical range for aerodynamic calculations, especially at high lift. One concern with these results is that the optimum values of ϕ and CFL are dependent on the angle of attack and the grid. Future research should attempt to remove this dependence.

References

1. Zingg, D.W., "Low Mach Number Euler Computations," *CASI J.*, Vol. 36, No. 3, pp. 146-152, 1990, (also NASA TM 102205, Oct. 1989).

2. Volpe, G., "On the Use and Accuracy of Compressible Flow Codes at Low Mach Numbers," AIAA Paper 91-1662, June 1991.

3. Turkel, E., "Review of Preconditioning Methods for Fluid Dynamics," *Applied Numerical Mathematics*, Vol. 12, pp. 257-284, 1993.

4. Choi, Y.H., and Merkle, C.L., "The Application of Preconditioning in Viscous Flows," *J. Comput. Phys.*, Vol. 105, pp. 207-233, 1993.

5. Turkel, E., "Preconditioned Methods for Solving the Incompressible and Low Speed Compressible Equations," *J. Comput. Phys.*, Vol. 72, pp. 277-298, 1987.

6. Van Leer, B., Lee, W.T., and Roe, P., "Characteristic Time-Stepping or Local Preconditioning of the Euler Equations," AIAA Paper 91-1552, June 1991.

7. Weiss, J.M., and Smith, W.A., "Preconditioning Applied to Variable and Constant Density Flows," *AIAA J.*, Vol. 33, No. 11, pp. 2050-2057, 1995.

8. Godfrey, A.G., "Steps Toward a Robust Preconditioning," AIAA Paper 94-0520, Jan. 1994.

9. Merkle, C.L., Venkataswaran, S., and Buelow, O.E., "The Relationship Between Pressure-Based and

Density-Based Algorithms," AIAA Paper 92-0425, Jan. 1992.

10. Chorin, A.J., "A Numerical Method for Solving Incompressible Viscous Flow Problems," *J. Comput. Phys.*, Vol. 2, pp. 12-26, 1967.

11. Allmaras, S.A., "Analysis of a Local Matrix Preconditioner for the 2-D Navier-Stokes Equations," AIAA Paper 93-3330, July 1993.

12. Zingg, D.W., "Grid Studies for Thin-Layer Navier-Stokes Computations of Airfoil Flowfields," *AIAA J.*, Vol. 30, No. 10, pp. 2561-2564, 1992.

13. Frew, K., and Zingg, D.W., "On Artificial Dissipation Models for Viscous Airfoil Computations," AIAA Paper 96-1970, June 1996.

14. Jespersen, D., Pulliam, T.H., and Buning, P., "Recent Enhancements to OVERFLOW," AIAA paper 97-0644, Jan. 1997.

15. Pulliam, T.H., "Efficient Solution Methods for the Navier-Stokes Equations," Lecture Notes for the Von Karman Institute for Fluid Dynamics Lecture Series, Brussels, Jan. 1986.

16. Barth, T.J., "Aspects of Unstructured Grids and Finite-Volume Solvers for the Euler and Navier-Stokes Equations," Lecture Notes for the Von Karman Institute for Fluid Dynamics Lecture Series, 1994-05, Brussels, 1994.

Appendix

The matrices relating the symmetry variables and the conservative variables are

$$M = \begin{bmatrix} \frac{\rho}{c} & 0 & 0 & -\frac{1}{c^2} \\ \frac{\rho u}{c} & \rho & 0 & -\frac{u}{c^2} \\ \frac{\rho v}{c} & 0 & \rho & -\frac{v}{c^2} \\ \rho\left(\frac{u^2+v^2}{2c} + \frac{c}{\gamma-1}\right) & \rho u & \rho v & -\frac{u^2+v^2}{2c^2} \end{bmatrix} \tag{36}$$

$$M^{-1} = \begin{bmatrix} (\gamma-1)\frac{u^2+v^2}{2\rho c} & (1-\gamma)\frac{u}{\rho c} & (1-\gamma)\frac{v}{\rho c} & \frac{\gamma-1}{\rho c} \\ -\frac{u}{\rho} & \frac{1}{\rho} & 0 & 0 \\ -\frac{v}{\rho} & 0 & \frac{1}{\rho} & 0 \\ \frac{\gamma-1}{2}\left(u^2+v^2\right) - c^2 & (1-\gamma)u & (1-\gamma)v & (\gamma-1) \end{bmatrix} \tag{37}$$