Accelerated Convergence of Euler Solutions Using Time Inclining

John F. Dannenhoffer III*
United Technologies Research Center, East Hartford, Connecticut 06108
and
Michael B. Giles†
Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

The use of *time inclining* for accelerating the convergence to steady state of inviscid-flow computations is examined. This technique, which was originally developed for the efficient computation of time-accurate rotor-stator interactions, is based upon a local inclination in time of the computational cells. For convergence acceleration, these inclinations are chosen so as to balance the time-step restrictions for upwind- and downwind-running pressure waves. The inclusion of time inclining into Ni's Lax-Wendroff-type integration scheme is discused in detail, both in terms of the additional transformations required as well as the selection of inclining parameters for near-optimal convergence rates. The technique is very easy to implement due to its local character and thus makes it an ideal candidate for structured as well as unstructured grid calculation procedures. The effectiveness of time inclining is demonstrated in two dimensions with freestream Mach numbers ranging from 0.2 to 3.0. The results show that although the steady-state accuracy is not affected by inclining, the convergence rate can be more than doubled.

Introduction

It is well known that the steady-state solution of a compressible fluid in transonic flows is difficult to compute due to the mixed types which the steady-state Euler equations exhibit; the equations are elliptic in subsonic regions of the flow and hyperbolic in supersonic regions. In order to circumvent the numerical difficulties associated with the above type-switching, numerical analysts have frequently relied on "time-marching" techniques, in which the time-asymptotic solution of the unsteady form of the equations is obtained. This approach is much simpler because the unsteady form of the equations is always hyperbolic.

The time-asymptotic (steady-state) solution is obtained when all of the errors associated with the initial conditions, evolution nonlinearities, and boundary reflections have been propagated out of the computational domain. Unfortunately, the waves do not all propagate at the same rate and hence convergence to steady state is limited by the traversal of the slowest waves.

In recent years, numerical analysts have developed techniques to hasten the propagation of the slowest wave. The most notable of these are the multiple-grid[1] and multigrid[2] techniques, wherein solutions are transferred amongst a series of nested computational grids of varying densities. By propagating the fine grid residuals on the coarse grids, the speeds of all waves are increased because of the larger time step allowed by the coarser cells. This technique has proven to be very effective over a large range of geometric and flow conditions. Unfortunately, the need for overlayed grids of varying densities makes these techniques difficult to implement, especially on unstructured grids.[3]

This paper discusses an alternative (or supplemental) approach, *time inclining*, which is an adaptation of a technique originally introduced by Giles[4] for the computation of time-accurate, rotor-stator interactions. This new approach can be viewed as either a local inclination in time of the computational cells, or as a local variable transformation as discussed below. In either case, the effect of inclining time is that the relative speeds of the characteristic waves are modified, and hence the convergence of a numerical integration scheme can be improved with a judicious choice of inclination parameter.

As will be discussed below, time inclining can be implemented very simply by viewing it as a local transformation, and hence it is very easy to retrofit into an existing Euler flow code, whether it uses a structured or unstructured grid.

Another advantage of time inclining over multigrid is that by suitable postprocessing, it is possible to extract time-accurate results from the time-marching solution, as long as constant inclining and time stepping is used. In fact, this feature of time inclining was the major reason that it was originally developed for rotor-stator interaction predictions.

This paper begins with a section which describes the evolution of quasi-one-dimensional Euler flows, both without and then with time inclining. Specific attention is focused on the propagation of the various characteristic waves, both analytically and through numerical examples. The next section considers one-dimensional flow with an algorithm for choosing optimal inclination parameters and the results of numerical experiments which demonstrate the effectiveness of employing this optimal inclination.

The remainder of the paper deals with the extension of the one-dimensional results to two dimensions, with the appropriate theoretical extensions given in detail. The paper concludes with a summary of the effectiveness of the technique over a wide Mach number and grid aspect ratio range.

Solution of One-Dimensional Euler Equations

Basic Scheme

The unsteady, one-dimensional Euler equations for compressible fluid flow can be expressed in conservation form as

$$\frac{\partial \boldsymbol{U}}{\partial t}+\frac{\partial \boldsymbol{F}}{\partial x}=0 \tag{1}$$

Received Jan. 23, 1989; revision received Oct. 12, 1989.

*Research Engineer, Computational Mathematics and Fluid Dynamics Group. Member AIAA.

†Assistant Professor, Department of Aeronautics and Astronautics. Member AIAA.

where

$$U = \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix}, \qquad F = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ (p + E)u \end{pmatrix} \tag{2}$$

and

$$p = (\gamma - 1)\left[E - \frac{1}{2}\rho u^2\right] \tag{3}$$

Here ρ, u, E, p, and γ are the density, velocity, total energy per unit volume, pressure, and ratio of specific heats of the fluid, respectively.

Alternatively, Eq. (1) can be written in the diagonal or characteristic form

$$\frac{\partial C}{\partial t} + \Lambda \frac{\partial C}{\partial x} = 0 \tag{4}$$

where $C = (c_1, c_2, c_3)^T$ and $\Lambda = \text{diag}[(u - a), u, (u + a)]$, and where $a = \sqrt{\gamma p/\rho}$ is the speed of sound. Because Λ is diagonal, this represents three uncoupled equations where the characteristic variables c_1, c_2, and c_3 (which represent the upstream-running pressure wave, entropy wave, and downstream-running pressure wave, respectively) propagate independently. From the definition of Λ, it can be seen that the characteristic variables travel at the speeds $dx/dt = (u - a)$, u, and $(u + a)$.

The explicit numerical integration in time of hyperbolic systems of the form given by Eqs. (1) and (4) are restricted (for considerations of numerical stability) to obey the CFL (Courant-Friedrichs-Lewy) condition, which states simply that the numerical domain of dependence must be at least as large as the physical domain of dependence. In other words, the CFL condition states that the numerical scheme must be such that the solution at some point i at time level $t + \Delta t$ must depend on *all* points at time level t whose characteristic variables affect point i. This then puts a limit on the size of time step Δt which may be taken.

The latter argument can be shown geometrically in Fig. 1. The three computational nodes A, B, and C at time level t are shown as the three solid circles along the horizontal axis. Shown superimposed on the space-time figure are the characteristic trajectories emanating from nodes A and C. (The slopes of these trajectories are consistent with subsonic $M = 0.50$ flow.) According to the CFL condition, any point within the shaded area can be stably computed, given just the conditions between A and C. The intersection of this shaded region with the vertical axis shows that the maximum stable time step Δt corresponds to point D.

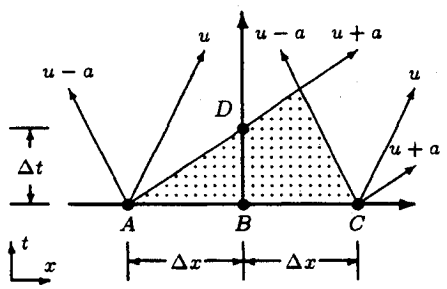

Fig. 1 Geometric interpretation of the CFL condition. Un-inclined computational plane ($\lambda = 0$) and subsonic flow ($M = 0.50$).

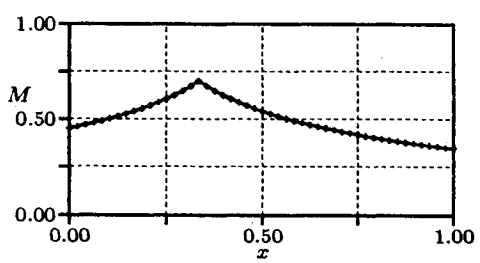

Fig. 2 Steady-state Mach number distribution for a quasi-one-dimensional, subsonic test case. Line represents numerical solution; symbols are generated analytically.

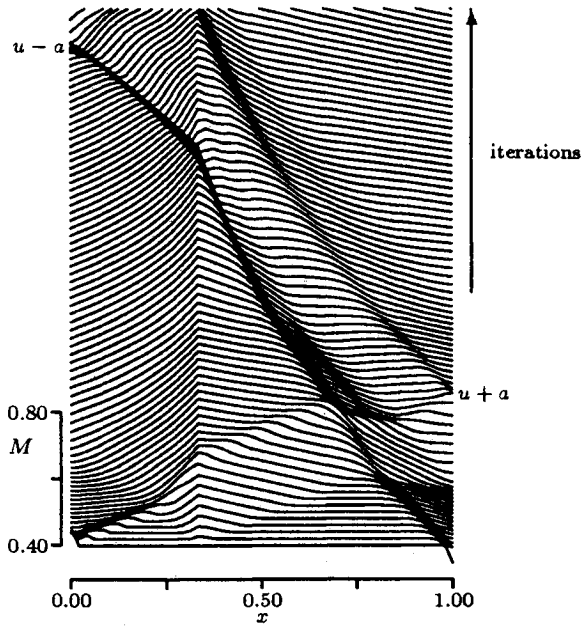

Fig. 3 Mach-number distribution evolution for quasi-one-dimensional, subsonic test case ($\lambda = 0$). Note: Every other iteration is plotted.

One can also note from the figure that in the time Δt, the downstream-running pressure wave $(u + a)$ moves one computational cell Δx, whereas the entropy u and upstream-running pressure wave $(u - a)$ each move $\Delta x/3$ (for $M = 0.50$); the degradation for each is in direct proportion to the ratio of its speed to the fastest wave speed $(u + a)$.

To demonstrate the effect of this, consider the numerical integration of Eq. (1) in a quasi-one-dimensional converging-diverging duct whose area ratio is given by $A = 1 + 0.96|x - 0.333|$. The flow, which is modeled by 49 equally distributed computational nodes, has the steady-state subsonic flow shown in Fig. 2. The numerical integration scheme used here is the one-step, Lax-Wendroff-type, finite-volume scheme developed by Ni,[1] with inlet and exit boundary conditions which preserve the specified reservoir conditions. In all calculations shown, local time stepping is employed to enhance convergence.

Shown in Fig. 3 is the Mach-number distribution evolution for this case. Each line represents the Mach number plotted vs the axial position in the duct at a given iteration; the solution for every other iteration is plotted (with its origin offset up slightly). The figure clearly shows that the forward-running $(u + a)$ wave which begins at the duct inlet reaches the duct exit in about 50 iterations, which is expected since the wave travels one computational cell in each iteration. The figure also shows that the upstream-running $(u - a)$ wave which begins at the duct exit reaches the duct inlet in about 150 iterations. Thus, this wave propagates approximately one-third of a cell per iteration, which was predicted above based upon an average flow Mach number of $M = 0.50$.

A similar analysis can be made for supersonic flow. The geometric description of the CFL condition for such a case, which is given in Fig. 4, clearly shows that only the $(u + a)$ characteristic from upstream (point A) affects the time step Δt. As before, the $(u + a)$ wave moves Δx in one time step,

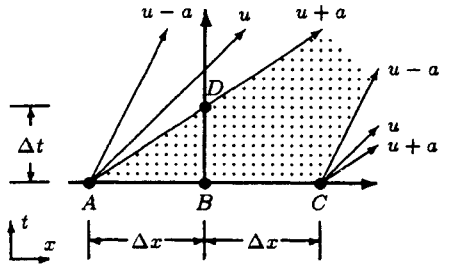

Fig. 4 Geometric interpretation of the CFL condition. Un-inclined computational plane ($\lambda = 0$) and supersonic flow ($M = 2.00$).

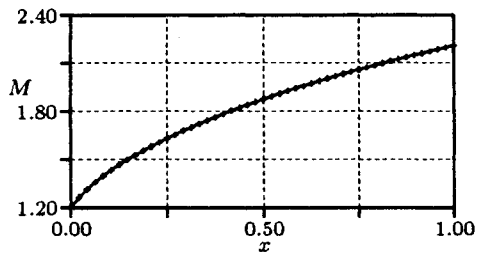

Fig. 5 Steady-state Mach number distribution for a quasi-one-dimensional, supersonic test case. Line represents numerical solution; symbols are generated analytically.

whereas the u and $(u - a)$ waves travel $\Delta x_u = 0.67\Delta x$ and $\Delta x_{u-a} = 0.33\Delta x$ per iteration (for $M = 2.0$).

The effect of this can again be demonstrated numerically by considering the supersonic flow in a diverging duct whose area ratio is given by $A = 1 + 0.96x$. The steady-state Mach number distribution for this flow, as computed with 49 equally spaced points, is shown in Fig. 5.

The Mach-number distribution evolution for this case is shown in Fig. 6, where again every other iteration is shown. The figure clearly shows the $(u + a)$ wave from the inlet reaching the exit in about 50 iterations, whereas the $(u - a)$ wave requires in excess of 200 iterations.

In summary, in both the subsonic and supersonic cases above, the size of the time step is restricted by the propagation of the fastest wave $(u + a)$, whereas the propagation of errors through and ultimately out of the domain are limited by the slowest wave $(u - a)$. Hence, it is the ratio of these speeds, $(u - a)/(u + a)$, which ultimately governs the convergence rate.

Scheme with Time Inclining

As discussed in the Introduction, standard time-marching schemes employ computational cells which are bounded by lines of constant time t and constant space x. The time-inclining technique on the other hand employs computational cells which are bounded by lines of constant pseudotime t' and constant space x. In this section, the consequences of this inclination will be discussed.

Consider a coordinate transformation from (x,t), the physical coordinates, to (x',t'), the computational coordinates in which t' is constant on each computational time level. The equations defining the transformation are

$$x' = x, \qquad t' = t - \lambda x \tag{5}$$

Applying this transformation to the Euler Eq. (1) results in the modified equations

$$\frac{\partial \boldsymbol{U}}{\partial t'} + \left(-\lambda \frac{\partial \boldsymbol{F}}{\partial t'} + \frac{\partial \boldsymbol{F}}{\partial x'} \right) = 0 \tag{6}$$

where now the independent variables are x' and t' and the inclination parameter is $\lambda \equiv \mathrm{d}t/\mathrm{d}x$. Equation (6) can be cast in the same form as Eq. (1) if the conservation state variables are changed from $\boldsymbol{U}$ to $\boldsymbol{Q} \equiv \boldsymbol{U} - \lambda \boldsymbol{F}$, giving

$$\frac{\partial \boldsymbol{Q}}{\partial t'} + \frac{\partial \boldsymbol{F}}{\partial x'} = 0 \tag{7}$$

For the Euler equations with the perfect-gas assumption, there are closed-form relations between $\boldsymbol{U}$ and $\boldsymbol{Q}$ as described in Refs. 4 and 5; for the Navier-Stokes equations, these relations still can be applied with the complication being the introduction of errors which are inversely proportional to the Reynolds number.[6] Notice that at steady state, $\partial \boldsymbol{F}/\partial x' = 0$, which is the same as the steady-state solution of the original Eq. (1).

Because Eqs. (7) and (1) are of the same form, numerical schemes developed to solve Eq. (1) can be used to solve Eq. (7) with very minor changes. For the standard Ni implementation of the Lax-Wendroff method used here,[1] only a few modifications are required.

The first step of Ni's scheme involves the calculation of the total flux out of each cell. This determines the first-order change $\Delta \boldsymbol{U}$ in each cell in the standard method, but in the modified algorithm this gives $\Delta \boldsymbol{Q}$ which can simply be converted to $\Delta \boldsymbol{U}$ as shown in Ref. 5. The next step in Ni's scheme involves the calculation of the changes in the flux terms $\Delta \boldsymbol{F}$, which are then distributed along with the first-order changes to the nodes. In the standard scheme, the distribution can be written as $\delta \boldsymbol{U} = (\Delta t/\Delta x)[\Delta \boldsymbol{U} \pm \Delta \boldsymbol{F}]$, whereas in the time-inclined scheme, it takes the form $\delta \boldsymbol{Q} = (\Delta t/\Delta x)[\Delta \boldsymbol{Q} \pm \Delta \boldsymbol{F}]$. Again the $\delta \boldsymbol{Q}$ have to be converted to $\delta \boldsymbol{U}$ before being added to the original values of $\boldsymbol{U}$ at the nodes to obtain the new flow solution. The total increase in CPU time per iteration associated with these conversions is approximately 15%.

The only other modification required to the standard scheme occurs in the calculation of the maximum permissible time step. Because of the inclination of the time planes, the speeds of the three characteristic waves are now governed by

$$\left.\frac{\Delta t'}{\Delta x'}\right|_{u+a} = \frac{1}{u + a} - \lambda \tag{8a}$$

$$\left.\frac{\Delta t'}{\Delta x'}\right|_{u} = \frac{1}{u} - \lambda \tag{8b}$$

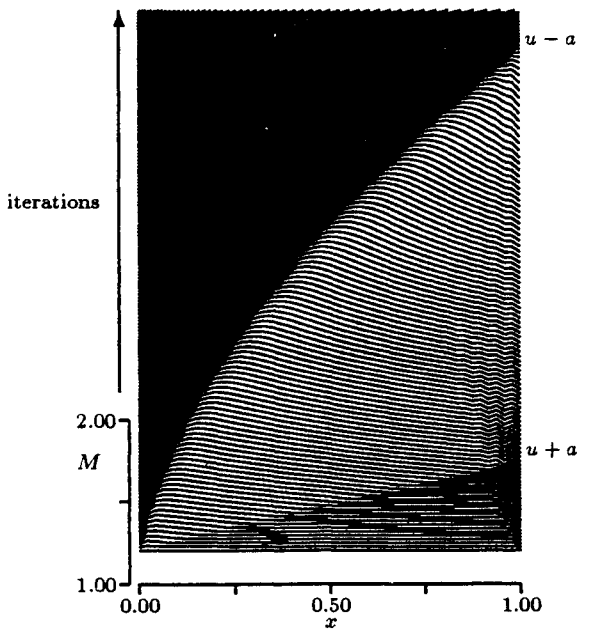

Fig. 6 Mach-number distribution evolution for quasi-one-dimensional, supersonic test case ($\lambda = 0$). Note: Every other iteration is plotted.

$$\left.\frac{\Delta t'}{\Delta x'}\right|_{u-a} = \frac{\mp 1}{u-a} \pm \lambda \tag{8c}$$

where the upper sign in the third equation applies to subsonic and the lower sign applies to supersonic conditions. From these equations, one can determine the maximum allowable time step

$$\frac{\Delta t'}{\Delta x'} = \min\left\{\frac{1}{u+a} - \lambda, \frac{\mp 1}{u-a} \pm \lambda\right\} \tag{9}$$

Note that the u wave can be ignored in the above since its effect is always the same as that of the $(u+a)$ wave, but it is always less restrictive.

To examine the effect of time inclining on the propagation of waves, consider again the subsonic test case shown earlier in Figs. 1–3. If the pseudotime planes t' are inclined with $\lambda = dt/dx = -0.50$, the revised CFL condition is as shown in Fig. 7. The figure shows that the $(u+a)$ wave from upstream still restricts the time step at D, but that the time step has improved over the un-inclined case (Fig. 1) due to the elevation (in time) of node A over node B. This improvement can be calculated by

$$\mathcal{T} = \frac{\Delta t}{\Delta t|_{\lambda=0}} = \frac{\dfrac{1}{u+a} - \lambda}{\dfrac{1}{u+a}} \tag{10}$$

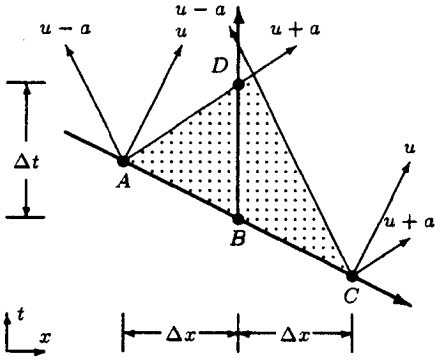

Fig. 7 Geometric interpretation of the CFL condition. Inclined computational plane ($\lambda a = -0.50$) and subsonic flow ($M = 0.50$).

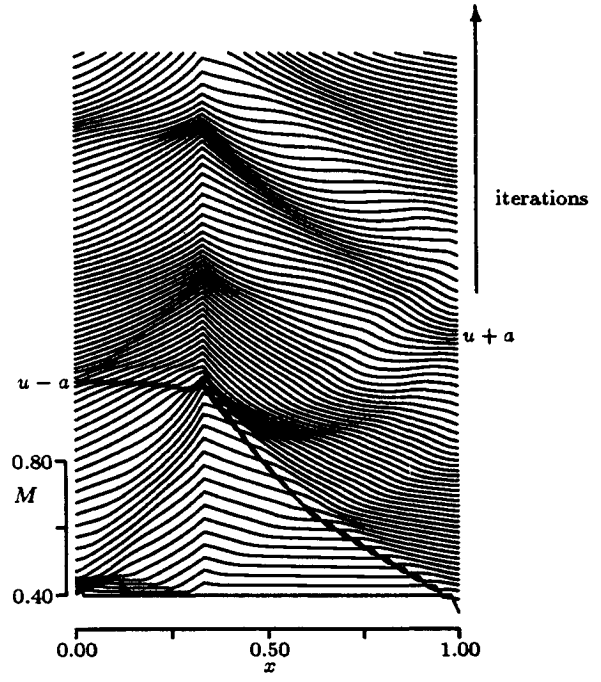

Fig. 8 Mach-number distribution evolution for quasi-one-dimensional, subsonic test case ($\lambda a = -0.50$). Note: Every other iteration is plotted.

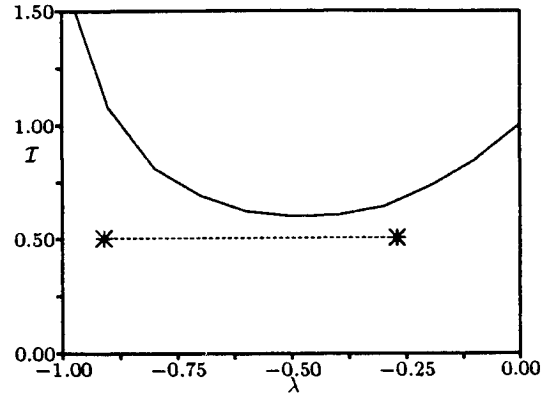

Fig. 9 Normalized number of iterations to convergence as a function of inclination parameter for the quasi-one-dimensional subsonic test case. Solid line is fixed inclination parameter; dashed line is optimal inclination parameter.

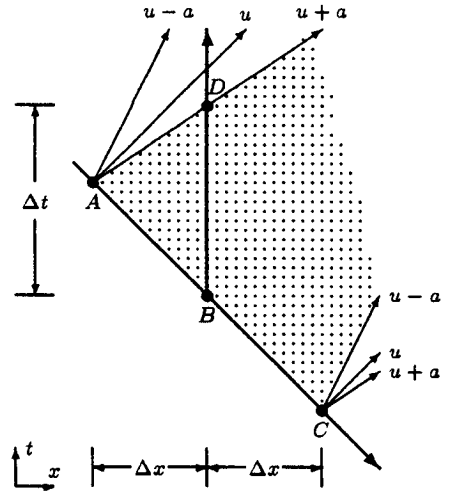

Fig. 10 Geometric interpretation of the CFL condition. Inclined computational plane ($\lambda a = -1.0$) and supersonic flow ($M = 2.00$).

yielding $\mathcal{T} = 1.75$ for $a\lambda = -0.50$ and $M = 0.50$. Additionally, the figure indicates that the $(u-a)$ wave now traverses nearly $0.8\Delta x$ in each time step.

This effect of inclining time is demonstrated by the Mach-number distribution evolution shown in Fig. 8. Here it is clear that the $(u-a)$ wave traverses the domain at only a slightly slower rate than the $(u+a)$ wave (as predicted above) yielding a much improved convergence rate.

By examining Eq. (9), one can see that the first term within the min-operator increases and that the second term decreases as $a\lambda$ becomes more negative. Therefore, the maximum value of the min-operator occurs when the magnitudes of these terms become equal, or

$$\frac{\Delta t_{\text{opt}}}{\Delta x} = \frac{1}{u+a} - \lambda_{\text{opt}} = -\frac{1}{u-a} + \lambda_{\text{opt}} \tag{11}$$

which yields

$$\lambda_{\text{opt}}|_{\text{subsonic}} = \frac{1}{a}\left[\frac{M}{M^2-1}\right] \tag{12}$$

The corresponding time-step improvement is given by

$$\mathcal{T}_{\text{opt}} = \frac{\Delta t_{\text{opt}}}{\Delta t_{\lambda=0}} = \frac{\dfrac{1}{u+a} - \lambda_{\text{opt}}}{\dfrac{1}{u+a}} = \frac{1}{1-M} \tag{13}$$

For subsonic ($M = 0.50$) flow, this yields $a\lambda_{opt} = -2/3$ and $\mathscr{T}_{opt} = 2.0$, while for nearly sonic ($M \to 1$) flow, this yields a nearly infinite ($\mathscr{T}_{opt} \to \infty$) time-step improvement.

To demonstrate the effectiveness of using the optimum inclination parameter, the subsonic test case was recomputed using λ_{opt} (which is a function of the *local* Mach number). Of course, local time stepping can be (and is) used in conjunction with the local time inclination. The wave pattern produced by this case closely resembles that shown in Fig. 8.

To quantitatively assess the effectiveness of the use of λ_{opt}, consider Fig. 9, where the abscissa represents the inclination parameter and the ordinate is *, the number of iterations needed for convergence (a five-order drop in the residual) normalized by the number of iterations required with no inclination ($\lambda = 0$). The computations represented by the solid line were made with differing levels of constant λ. The dashed line represents one run made with the optimal inclination parameter computed at each point in the domain. The horizontal extent of this dashed line shows the minimum and maximum values of λ throughout the domain. Note that the number of iterations to converge for this case is significantly fewer than that required for any case with constant λ.

Now consider the effect that time inclining has on the convergence rate of *supersonic* flows. For this purpose, the supersonic case shown previously in Figs. 4–6 is re-examined with an inclination parameter $\lambda = -1.0$. The resulting wave diagram, shown in Fig. 10, indicates that again the $(u + a)$ wave limits the time step, but that the time step has improved dramatically. Additionally, the $(u - a)$ and u waves now traverse a large fraction of a computational cell in each time step.

These effects are demonstrated in the Mach-number evolution given in Fig. 11. It is clear that the slowest $(u - a)$ wave now traverses the domain in only twice as many iterations as the fastest $(u + a)$ wave. The time-step improvement factor can be computed by Eq. (9); for $a\lambda = -1$ and $M = 2.0$, this yields $\mathscr{T} = 4.0$.

Examination of Eq. (9) for supersonic flows shows that the time step is always restricted by the $(u + a)$ wave, and that the restriction can be lessened by making $a\lambda$ more negative. In fact, one can see from Eq. (8) that the maximum time step occurs when

$$\lambda_{opt}|_{supersonic} \to -\infty \tag{14}$$

which yields $\Delta t_{opt} \to \infty$. In other words, no matter how large the time step, one can find a value of λ which will result in a stable integration. This is in contrast to the subsonic results derived above.

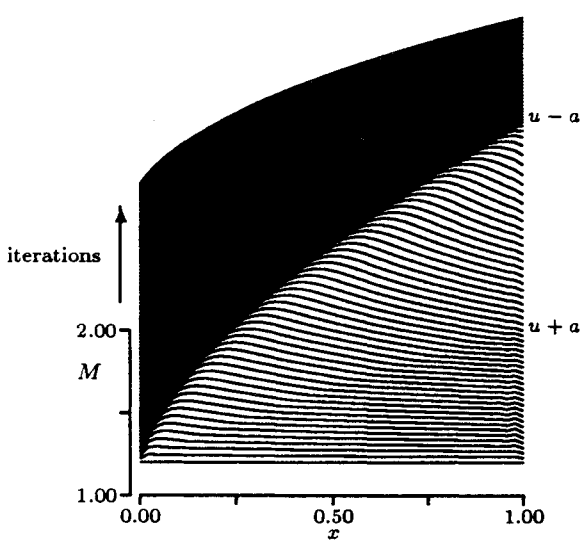

Fig. 11 Mach-number distribution evolution for quasi-one-dimensional, supersonic test case ($\lambda a = -1.0$). Note: Every other iteration is plotted.

The effectiveness of using λ_{opt} in supersonic flow is shown by again plotting the normalized number of iterations * vs the inclination parameter λ as shown in Fig. 12. Notice that choosing λ_{opt} (which was arbitrarily limited to -100) again results in a significant decrease in the number of iterations required to converge.

In summary, the optimal inclination parameter for subsonic flows is finite and given by Eq. (12), whereas in supersonic flow the optimal inclination is infinite as given by Eq. (14). In both cases, the use of optimal time inclining can reduce the number of iterations to convergence by at least a factor of two.

Application in Two Dimensions

The time-inclining concepts developed above in one dimension can easily be applied to two- and three-dimensional flows. In this section, the extension to two dimensions is described.

Formulation

In two dimensions, the inclined pseudotime planes are generated through the transformation

$$x' = x, \qquad y' = y, \qquad t' = t - \lambda_x x - \lambda_y y \tag{15}$$

where now the two inclination parameters, λ_x and λ_y, can be chosen independently. These yield the transformed governing equation

$$\frac{\partial \boldsymbol{Q}}{\partial t'} + \frac{\partial \boldsymbol{F}}{\partial x'} + \frac{\partial \boldsymbol{G}}{\partial y'} = 0 \tag{16}$$

where $\boldsymbol{Q} \equiv \boldsymbol{U} - \lambda_x \boldsymbol{F} - \lambda_y \boldsymbol{G}$. As in one dimension, there are closed-form expressions for $\boldsymbol{Q} = \boldsymbol{Q}(\boldsymbol{U})$ and $\boldsymbol{U} = \boldsymbol{U}(\boldsymbol{Q})$, which are described in Ref. 5.

Computational schemes written without time inclining can again be easily modified to include pseudotime planes by incorporating $\boldsymbol{Q}$-to-$\boldsymbol{U}$ transformations in exactly the same way as in the one-dimensional scheme discussed previously.

To examine the effect that inclined time has on the time-step restriction, it is instructive to first examine the time-step-restricting mechanism in two dimensions—without inclined time. Based upon a characteristic analysis, one finds that the time step is restricted both by pressure waves propagating in the streamwise direction as well as pressure waves propagating normal to the local flow direction. The analysis of these restrictions is facilitated by considering intrinsic orthogonal coordinates (s,n), where s is in the local streamwise direction. The time-step restrictions then take the form

$$\Delta t_s = \Delta s \min\left\{\frac{1}{q + a}, \frac{1}{|q - a|}\right\} \tag{17}$$

and

$$\Delta t_n = \Delta n \left\{\frac{1}{a}\right\} \tag{18}$$

where Δs and Δn are the dimensions of the computational cell in the s and n directions, and the velocity $q = \sqrt{u^2 + v^2}$ is defined in terms of the x and y components of the velocity (u and v). Of course, the time step which may be taken is $\Delta t = \min(\Delta t_s, \Delta t_n)$.

The inclusion of time inclining modifies these restrictions in an analogous way as in one dimension, yielding

$$\Delta t_s = \Delta s \min\left\{\frac{1}{q + a} - \lambda_s, \frac{\mp 1}{q - a} \pm \lambda_s\right\} \tag{19}$$

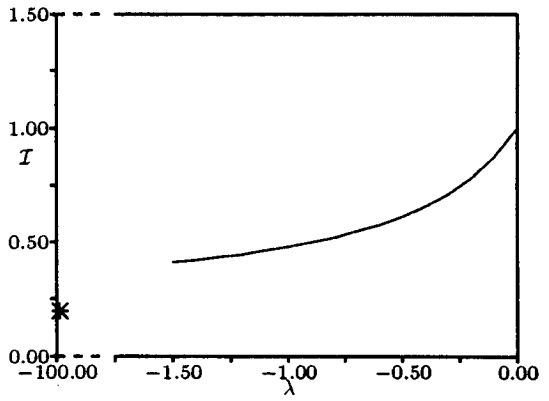

Fig. 12 Normalized number of iterations to convergence as a function of inclination parameter for the quasi-one-dimensional supersonic test case. Solid line is fixed inclination parameter; symbol is near-optimal inclination parameter.

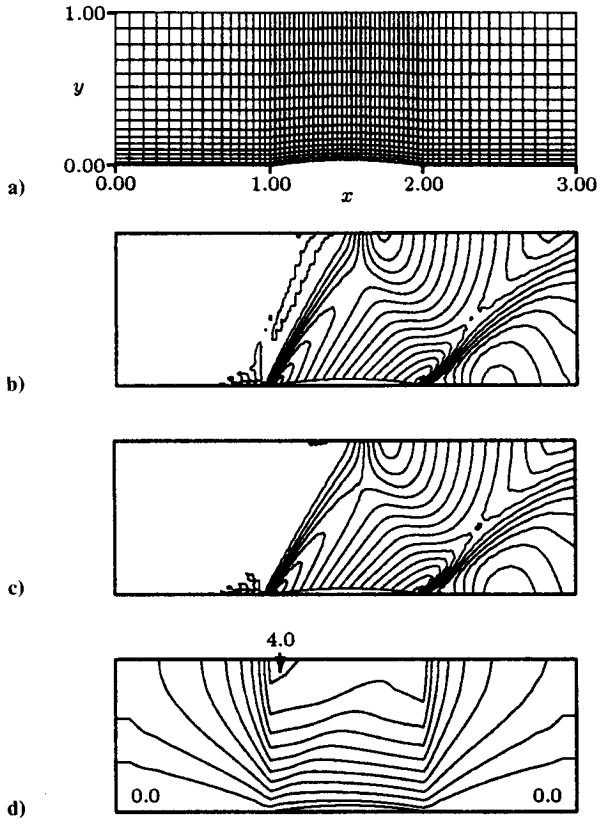

Fig. 13 Two-dimensional test case ($M_\infty = 1.40$): a) geometry and computational grid; b) Mach number contours for $\lambda = 0$ ($\Delta M = 0.04$); c) Mach number contours for $\lambda = \lambda_{opt}$ ($\Delta M = 0.04$); and d) inclination parameter magnitude contours ($\Delta\lambda = 0.40$).

and

$$\Delta t_n = \Delta n \min\left\{\frac{1}{a} - \lambda_n, \frac{1}{a} + \lambda_n\right\} = \Delta n\left\{\frac{1}{a} - |\lambda_n|\right\} \tag{20}$$

where λ_s and λ_n are given by the rotations

$$\lambda_s = (+u\lambda_x + v\lambda_y)/q \tag{21a}$$

$$\lambda_n = (-v\lambda_x + u\lambda_y)/q \tag{21b}$$

Thus, computing the maximum stable time step with time inclining in two dimensions is only slightly more complicated than calculating it without inclination.

The obvious method for computing λ_s and λ_n which maximizes the time step is to calculate them independently, based solely upon considerations in the s and n directions, respectively. Examination of Eq. (19) shows that the streamwise restriction is exactly the same as the one-dimensional restriction of Eq. (9), giving that the Δt_s is maximized when

$$\begin{aligned} \lambda_s &= \frac{1}{a}\frac{M}{M^2 - 1} && \text{for } M < 1 \\ &= -\infty && \text{for } M \geq 1 \end{aligned} \tag{22}$$

In an analogous way, Eq. (20) can be used for the crossflow inclination, giving that Δt_n is maximized when

$$\lambda_n = 0 \tag{23}$$

Unfortunately, the actual time step which may be taken in any computational cell is restricted by the smaller of Δt_s and Δt_n, and thus it is not necessary to increase Δt_s to its maximum value, but instead only to a value such that Δt_n is either equally or more restrictive. In other words, in two dimensions it is appropriate to choose λ_s to maximize Δt_s *only* if $\Delta t_s < \Delta t_n$; otherwise λ_s should be chosen so as to make $\Delta t_s = \Delta t_n$ (if possible). In equation form, this becomes

$$\begin{aligned} \lambda_s|_{opt} &= \min\left[0, \max\left\{\frac{1}{q+a} - \frac{\Delta n}{\Delta s}\frac{1}{a}, \frac{q}{q^2 - a^2}\right\}\right] && \text{for } M < 1 \\ &= \min\left[0, \frac{1}{q+a} - \frac{\Delta n}{\Delta s}\frac{1}{a}\right] && \text{for } M \geq 1 \end{aligned} \tag{24}$$

$$\lambda_n|_{opt} = 0 \tag{25}$$

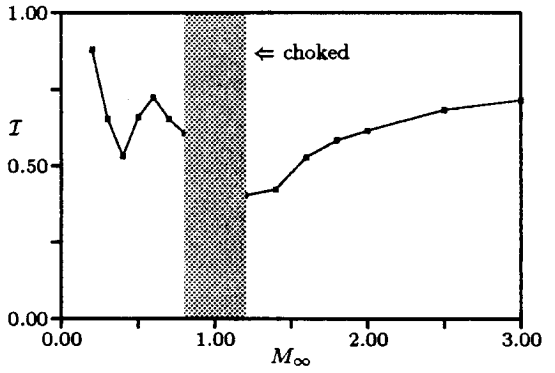

Fig. 14 Ratio of number of iterations required to converge (optimally inclined/noninclined) vs freestream Mach number.

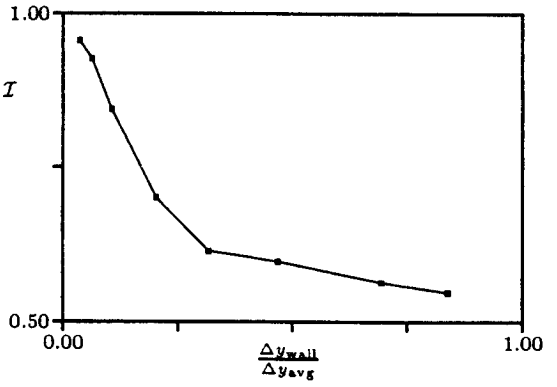

Fig. 15 Normalized number of iterations required to converge vs ratio of minimum-to-average cell size for $M_\infty = 2.0$ flow.

The resulting time step is given by

$$\Delta t = \min\left\{\Delta s\left(\frac{1}{q+a} - \lambda_s\right), \Delta n\left(\frac{1}{a}\right)\right\} \tag{26}$$

One further problem arises in the computation of transonic flows, where the optimal inclination parameter for the subsonic cells is of order one whereas the optimal for the (adjacent) supersonic cells is a large negative number (for example, of order -1000). This disparity in neighboring cells yields nearly discontinuous pseudotime planes, which if not treated properly, can result in errors on the order of Δx. Fortunately in two-dimensional applications, the crossflow time-step restriction Δt_n tends to limit the streamwise inclination, and hence, λ_s is generally of order ten or less. Unfortunately, this alone does not produce a smooth enough distribution of λ_s, thereby necessitating an additional smoothing of λ_s over the domain. It should be noted that this smoothing does not affect the steady-state solution, but rather only increases the convergence rate.

Computed Solutions

The two-dimensional test case examined in this paper is an 8%-thick biconvex cascade with gap/chord ratio of 2.0. This geometry, and the 65×17 computational grid used, are shown in Fig. 13a. Notice that the computational cells are clustered near the lower wall using Roberts' grid-stretching function.[7]

Calculations were performed on this geometry with freestream Mach numbers in the range of $0.20 \rightarrow 0.80$ and $1.20 \rightarrow 3.00$; flows near $M_\infty \approx 1.0$ were choked and could not be computed, even without inclined time. In each case, the Mach-number distributions for the solutions with and without time inclining were essentially the same, as demonstrated by Figs. 13b and 13c in which the Mach-number contours are shown. In addition, contours of the time-inclination parameter $\lambda \equiv \sqrt{\lambda_x^2 + \lambda_y^2}$ are given in Fig. 13d for the optimally inclined case, showing the wide range of inclinations necessary.

The benefit of time inclining is summarized in Fig. 14, which shows $* =$ (optimally inclined iterations/noninclined iterations) vs the freestream Mach number. One can see that the major benefit occurs for moderate Mach numbers (between about 0.50 and 2.00). At both lower and higher Mach numbers, the speeds of the upwind and downwind pressure waves become nearly equal $[(u+a)/(u-a) \rightarrow \mp 1]$ and thus time inclining cannot improve the convergence rate by much. The local minimum in the curve in the vicinity of $M \approx 0.4$ is not currently understood. Even so, in all cases, the optimally inclined scheme converged faster than its noninclined counterpart.

Fortunately in all of the above calculations, the time step was mainly restricted by streamwise waves and thus time inclining was effective. This unfortunately is not always the case, especially in high-aspect-ratio cells which are aligned with the flow (as in the case of cells within the boundary layer of a Navier-Stokes computation). To demonstrate this, the $M_\infty = 2.0$ case was recomputed, with computational grids generated by modifying the Roberts' grid-stretching parameter. A summary of these results is shown in Fig. 15, where the ratio of iterations $*$ is plotted vs the ratio of minimum-to-average cell height. It clearly shows that as the cell-height ratio approaches zero (longer, narrower cells), the effectiveness of time inclining diminishes. Other techniques, such as semi-implicit formulations near the wall, may be used with time inclining to circumvent the problems with this type of cell.

Conclusions

This paper has discussed the use of time-inclined computational planes in order to accelerate the convergence rate of time-marching procedures. The major conclusions which can be drawn from this research include

1) Both subsonic and supersonic one-dimensional test cases demonstrate that a computation using optimal time inclining requires less than half the number of iterations to converge (as compared with a standard, noninclined scheme), with a very small increase in computational work per time step (less than 15%).

2) Two-dimensional subsonic, transonic, and supersonic test cases demonstrate that in all cases optimal time inclining reduces the number of iterations to convergence. The actual benefit for time inclining depends on a) *Mach number*: Time inclining works best for flows in the $0.50 \rightarrow 2.0$ Mach number range; and b) *Grid aspect ratio*: Computational cells which have time-step restrictions due to the streamwise characteristic waves can be helped through the use of time inclining, whereas those whose time steps are limited by the crossflow waves (such as boundary-layer cells) cannot be helped much by this scheme.

3) Time inclining is easy to retrofit into existing explicit and/or implicit codes due to the very local nature of the required transformations.

4) This technique can also be used to accelerate the computation of time-accurate flows, as long as the inclination parameter and time steps are held constant across the domain.

5) The extension of this technique to three dimensions should be relatively straightforward since the three-dimensional crossflow effects are no more severe than those in two dimensions.

References

[1]Ni, R-H., "A Multiple-Grid Scheme for Solving the Euler Equations," *AIAA Journal*, Vol. 20, Nov. 1982, pp. 1565–1571.

[2]Jameson, A., "Transonic Flow Calculations," MAE Rept. 1651, Dept. of Mechanical and Aerospace Engineering, Princeton Univeristy, NJ, 1983.

[3]Mavriplis, D. J., "Multigrid Solution of the Two-Dimensional Euler Equations on Unstructured Triangular Meshes," *AIAA Journal*, Vol. 26, July 1988, pp. 824–831.

[4]Giles, M. B., "Generalized Conservation Cells for Finite Volume Calculations," AIAA Paper 87-1118-CP, June 1987.

[5]Giles, M. B., "UNSFLO: A Numerical Method for Unsteady Inviscid Flow in Turbomachinery," GTL Rept. 195, Massachusetts Institute of Technology, MA, Oct. 1988.

[6]Giles, M. B., "Numerical Methods for Unsteady Turbomachinery Flow," CFDL-TR-89-3, Massachusetts Institute of Technology, MA, April 1989.

[7]Roberts, G. O., "Computational Meshes for Boundary-Layer Problems," *Lecture Notes in Physics*, Vol. 8, Springer-Verlag, New York, 1971, pp. 171–177.