# Implicit Finite-Difference Simulation of Flow about Arbitrary Two-Dimensional Geometries

Joseph L. Steger*
*NASA Ames Research Center, Moffett Field, Calif.*

**Finite-difference procedures are used to solve either the Euler equations or the "thin-layer" Navier-Stokes equations subject to arbitrary boundary conditions. An automatic grid generation program is employed, and because an implicit finite-difference algorithm for the flow equations is used, time steps are not severely limited when grid points are finely distributed. Computational efficiency and compatibility to vectorized computer processors is maintained by use of approximate factorization techniques. Computed results for both inviscid and viscous flow about airfoils are described and compared to viscous known solutions.**

## I. Introduction

THE current work is mainly concerned with two of the ingredients of computational fluid dynamics—the blending of an implicit finite-difference scheme with transformations that permit use of automatic grid generation techniques. The combination produces what should prove to be a good recipe for many practical flow calculations. Results in this paper are restricted to flow about airfoils, but the basic core routines apply to a variety of problems (e.g., internal flows, rotating machinery), and the extension of the algorithms to three dimensions is reasonably straightforward.

## II. Motivations

In any numerical simulation the choice of numerical algorithm and flowfield model is dictated by considerations of computer cost, the type of problem being solved, the flow regime, and whether a very precise solution or an approximate solution is needed. These dictates have spawned a vast array of simulation techniques (e.g., finite differences, finite elements, particle in cell, integral methods), the use of special approximations (e.g., incompressible or compressible, steady or unsteady, rotational or irrotational, boundary-layer or Navier-Stokes), and a variety of ways to impose geometric boundaries (e.g., thin airfoil theory, special boundary operators, curvilinear and conformal coordinates, finite elements).

It seems that, even if one has a definite problem to solve, it can be difficult or impossible to select a best or most efficient numerical method—for example, a procedure that runs very efficiently on a conventional serial computer processor may run poorly on a particular vector computer processor. However, as computer processors and numerical algorithms continue to improve, it is natural to try to develop general simulation routines that fit geometric boundaries exactly and make few physical approximations. The usual justification for a quite general simulation routine is that while it may be costly in comparison to a simplified algorithm, it can be readily adapted to a large class of problems. As a result a given solution might initially be obtained in fewer man hours, and of course, detailed solutions are needed as checkpoints for less expensive but more approximate solutions. A more pragmatic justification for a general routine is the suspicion that it may be easier in the long run to make a general program efficient than to make an efficient program more general.

Presented as Paper 77-665 at the AIAA 10th Fluid and Plasmadynamics Conference, Albuquerque, N. Mex., June 27-29, 1977; submitted Aug. 18, 1977. 

Index categories: Computational Methods; Transonic Flow.

*Research Scientist. Member AIAA.

Inadequate knowledge of turbulence modeling and lack of a sufficiently powerful computer to handle three dimensions insure the continued use of approximate methods, but various developments have come to the forefront to make general formulations more attractive. These are more versatile grid generation routines,[1-3] general transformations of the equations that maintain strong conservative form,[4-9] and reliable implicit finite-difference schemes.[10-13]

The thrust of the current work is toward combining general transformations, grid generation techniques, and implicit algorithms into a viable and versatile flow program. The use of implicit schemes removes much of the stiffness problem associated with locally refined grids, especially viscous grids, and of course, is ideally suited to the viscous terms themselves. Transformations and grid generation schemes permit arbitrary geometries to be solved by conventional finite-difference methods and permit study of various unsteady motions as well. Thus, considerable flexibility exists along with the promise that the core programs developed can be readily adapted to a variety of problems without considerable special tailoring.

The following sections detail the transformed equations, grid generation routines, and the implicit schemes. Finally, a variety of flowfields about airfoils are solved and are used to verify the numerical algorithm and illustrate its versatility.

## III. Transformation of Governing Equations

The strong conservation law form of the Navier-Stokes equations in Cartesian coordinates can be written in non-dimensional form as (see Peyret and Viviand[14]):

$$\partial_t q + \partial_x E + \partial_y F = R_e^{-1}(\partial_x R + \partial_y S) \tag{1}$$

where

$$q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix}, \quad E = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(e+p) \end{bmatrix}, \quad F = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(e+p) \end{bmatrix}$$

$$R = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ R_4 \end{bmatrix}, \quad S = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ S_4 \end{bmatrix}$$

with

$$\tau_{xx}=(\lambda+2\mu)u_x+\lambda v_y, \quad \tau_{xy}=\mu(u_y+v_x)$$

$$\tau_{yy}=(\lambda+2\mu)v_y+\lambda u_x$$

$$R_4=u\tau_{xx}+v\tau_{xy}+\kappa P_r^{-1}(\gamma-1)^{-1}\partial_x a^2$$

$$S_4=u\tau_{xy}+v\tau_{yy}+\kappa P_r^{-1}(\gamma-1)^{-1}\partial_y a^2$$

and

$$p=(\gamma-1)[e-0.5\rho(u^2+v^2)] \tag{2}$$

where $p$ is the pressure, $a$ is the sound speed, and $\gamma$ is taken as $-(2/3)\mu$, the Stokes' hypothesis. Note that while the non-dimensional reference quantities are arbitrary, the Reynolds' number ($R_e$) and Prandtl number ($P_r$) used in Eq. (1) are defined in terms of those reference values.

If new independent variables are introduced, a strong conservation law form of Eq. (1) can be maintained as shown, for example, by Lapidus,[4] Viviand,[5] and Vinokur[6] (see also the related finite volume methods[7-9]). Subject to the general transformation

$$\xi=\xi(x,y,t) \quad \eta=\eta(x,y,t), \quad \tau=t \tag{3}$$

the equations can be written (see Refs. 5 and 14) as

$$\partial_\tau\hat{q}+\partial_\xi\hat{E}+\partial_\eta\hat{F}=R_e^{-1}\{\partial_\xi[J^{-1}(\xi_x\boldsymbol{R}+\xi_y\boldsymbol{S})] + \partial_\eta[J^{-1}(\eta_x\boldsymbol{R}+\eta_y\boldsymbol{S})]\} \tag{4}$$

where

$$\hat{q}=q/J, \quad \hat{E}=(\xi_t q+\xi_x E+\xi_y F)/J,$$

$$\hat{F}=(\eta_t q+\eta_x E+\eta_y F)/J$$

and

$$\tau_{xx}=(\lambda+2\mu)(\xi_x u_\xi+\eta_x u_\eta)+\lambda(\xi_y v_\xi+\eta_y v_\eta)$$

etc., where derivatives such as $u_x$ are expanded by chain rule, $u_x=\xi_x u_\xi+\eta_x u_\eta$. Here $J$ is the transformation Jacobian

$$J=\xi_x\eta_y-\xi_y\eta_x=1/(x_\xi y_\eta-x_\eta y_\xi) \tag{5}$$

The metrics $\xi_t,\xi_x$, etc., are easily formed from the derivatives of $x_\tau$, $x_\xi$, etc., using the relations

$$\xi_x=Jy_\eta, \quad \xi_y=-Jx_\eta, \quad \xi_t=-x_\tau\xi_x-y_\tau\xi_y \tag{6a}$$

$$\eta_x=-Jy_\xi, \quad \eta_y=Jx_\xi, \quad \eta_t=-x_\tau\eta_x-y_\tau\eta_y \tag{6b}$$

It is also convenient to define the velocities

$$U=\xi_t+\xi_x u+\xi_y v=\xi_x(u-x_\tau)+\xi_y(v-y_\tau) \tag{7a}$$

$$V=\eta_t+\eta_x u+\eta_y v=\eta_x(u-x_\tau)+\eta_y(v-y_\tau) \tag{7b}$$

which are the so-called contravariant velocities along the $\xi$ and $\eta$ coordinates. Using these defined velocities, $\hat{E}$ and $\hat{F}$ can be written in the compact form

$$\hat{E}=J^{-1}\begin{bmatrix}\rho U\\ \rho uU+\xi_x p\\ \rho vU+\xi_y p\\ (e+p)U-\xi_t p\end{bmatrix}, \quad \hat{F}=J^{-1}\begin{bmatrix}\rho V\\ \rho uV+\eta_x p\\ \rho vV+\eta_y p\\ (e+p)V-\eta_t p\end{bmatrix} \tag{8}$$

Note that once $U$ and $V$ are formed, the flux vectors $\hat{E}$ and $\hat{F}$ are not much more complex than $\boldsymbol{E}$ and $\boldsymbol{F}$.

One does not generally have sufficient computer power to resolve the viscous terms except in a thin layer near the body; consequently, a thin-layer approximation is used here. Viscous terms in $\xi$, which is the direction along the body (see Fig. 1), are neglected and only terms in $\eta$ are retained. For a boundary-layer-like coordinate system the viscous terms are thus simplified to

$$R_e^{-1}\partial_\eta J^{-1}\times$$

$$\begin{bmatrix}0\\ \mu(\eta_x^2+\eta_y^2)u_\eta+(\mu/3)\eta_x(\eta_x u_\eta+\eta_y v_\eta)\\ \mu(\eta_x^2+\eta_y^2)v_\eta+(\mu/3)\eta_y(\eta_x u_\eta+\eta_y v_\eta)\\ \kappa P_r^{-1}(\gamma-1)^{-1}(\eta_x^2+\eta_y^2)\partial_\eta a^2+\mu(\eta_x^2+\eta_y^2)(u^2+v^2)_\eta/2\\ \quad+(\mu/6)[\eta_x^2(u^2)_\eta+\eta_y^2(v^2)_\eta+2\eta_x\eta_y(uv)_\eta]\end{bmatrix}$$

$$\equiv R_e^{-1}\partial_\eta\hat{S} \tag{9}$$

and Eq. (4) is rewritten

$$\partial_\tau\hat{q}+\partial_\xi\hat{E}+\partial_\eta\hat{F}=R_e^{-1}\partial_\eta\hat{S} \tag{10}$$

The thin layer model is not a necessary step in this development, but it is a useful simplification. Although similar (or identical if $\xi$ is normal to $\eta$) viscous terms are dropped in boundary-layer theory, the "normal momentum" equation is retained in the thin-layer model and pressure can vary through the viscous layer. Consequently, the thin-layer model is devoid of the problems that would occur in matching an inviscid solution with a conventional boundary-layer solution and the separation point is not a mathematical singularity.

Along the body surface $\eta(x,y,t)=0$ (see Fig. 1), the condition of tangency in inviscid flow is

$$V=0 \text{ or } \begin{pmatrix}u\\ v\end{pmatrix}=J^{-1}\begin{bmatrix}\eta_y & -\xi_y\\ -\eta_x & \xi_x\end{bmatrix}\begin{pmatrix}U-\xi_t\\ -\eta_t\end{pmatrix} \tag{11}$$

while in viscous flow $U=0$ as well. The pressure on the body surface can be obtained from the momentum equations and one such relation is found for inviscid flow by simplifying $\eta_x^*(\xi-\text{momentum})+\eta_y^*(\eta-\text{momentum})$:

$$\rho[\partial_\tau\eta_t+u\partial_\tau\eta_x+v\partial_\tau\eta_y]-\rho U(\eta_x u_\xi+\eta_y v_\xi)$$

$$=(\eta_x\xi_x+\xi_y\eta_y)p_\xi+(\eta_x^2+\eta_y^2)p_\eta=\sqrt{\eta_x^2+\eta_y^2}\,p_n \tag{12}$$

where $n$ is the direction normal to the body surface. The same relation has been used in viscous flow with $U=0$.

Jacobian matrices used in the time linearization of $\hat{E}$, $\hat{F}$, and $\hat{S}$ are needed for the implicit algorithm to be defined later. The flux vectors $\hat{E}$ and $\hat{F}$ are both linear combinations of $\boldsymbol{q}$, $\boldsymbol{E}$, and $\boldsymbol{F}$

$$\hat{E} \text{ or } \hat{F}=(k_0\boldsymbol{q}+k_1\boldsymbol{E}+k_2\boldsymbol{F})J^{-1} \text{ with } k_0=\xi_t \text{ or } \eta_t, \text{ etc.} \tag{13}$$

so the Jacobian matrices $\hat{A}\equiv\partial\hat{E}/\partial\hat{q}$ and $\hat{B}\equiv\partial\hat{F}/\partial\hat{q}$ are given by

$$\hat{A} \text{ or } \hat{B}=k_0 I+k_1 A+k_2 B \tag{14}$$

where $A=\partial E/\partial q$ and $B=\partial F/\partial q$ are the usual Jacobian matrices of the Cartesian flux vectors. The $\hat{A}$ or $\hat{B}$ matrix is

$$\begin{bmatrix} k_0 & k_1 & k_2 & 0 \\ -u(k_1u+k_2v) & -(\gamma-2)k_1u & -(\gamma-1)k_1v & (\gamma-1)k_1 \\ +k_1\phi^2 & +k_0+k_1u+k_2v & +k_2u & \\ -v(k_1u+k_2v) & k_1v & -(\gamma-2)k_2v & (\gamma-1)k_2 \\ +k_2\phi^2 & -(\gamma-1)k_2u & +k_0+k_1u+k_2v & \\ (k_1u+k_2v)^* & [\gamma(e/\rho)-\phi^2]k_1 & [\gamma(e/\rho)-\phi^2]k_2 & \gamma(k_1u+k_2v) \\ [-\gamma(e/\rho)+2\phi^2] & -(\gamma-1)(uk_1+vk_2)u & -(\gamma-1)(uk_1+vk_2)v & +k_0 \end{bmatrix} \tag{15}$$

where $\phi^2 = 0.5(\gamma-1)(u^2+v^2)$.

Warming et al.[15] found the eigenvalues of $k_1A+k_2B$:

$$\sigma(k_1A+k_2B) = k_1u+k_2v,\ k_1u+k_2v,\ k_1u+k_2v + a\sqrt{k_1^2+k_2^2},\ k_1u+k_2v-a\sqrt{k_1^2+k_2^2} \tag{16}$$

so the roots of $\hat{A}$ or $\hat{B}$ are real and are simply

$$\sigma(\hat{A} \text{ or } \hat{B}) = k_0+\sigma(k_1A+k_2B) \tag{17}$$

The eigenvalues $\sigma(\hat{A})$, for example, are $U, U, U+a\sqrt{\xi_x^2+\xi_y^2}$, $U-a\sqrt{\xi_x^2+\xi_y^2}$.

The elements of $\hat{S}$ are of the general form [see Eq. (9)]

$$f_1(x,y,t,q) = \alpha_i(x,y,t,q)\partial_\eta\beta_i(x,y,t,q) \tag{18}$$

so each element linearizes in time (with the metrics fixed at $n+\frac{1}{2}$) as

$$f_i^{n+1} \doteq f_i^n + \alpha_i^n\partial_\eta\left\{\sum_{\ell=1}^{4}\left[\frac{\partial\beta_i}{\partial q_\ell}\Big|_n(q_\ell^{n+1}-q_\ell^n)\right]\right\} + \left\{\sum_{\ell=1}^{4}\left[\frac{\partial\alpha_i}{\partial q_\ell}\Big|_n(q_\ell^{n+1}-q_\ell^n)\right]\right\}\partial_\eta\beta_i^n \tag{19a}$$

For simplicity, $\mu$ and $\kappa$ will not be taken as functions of $q$ (as this would require numerical evaluation of the elements) and $\partial\alpha/\partial q = 0$. Furthermore, $\beta$ is a vector that is homogenous in $q$ of degree zero, and consequently

$$\sum_{\ell=1}^{4}\left[\frac{\partial\beta_i}{\partial q_\ell}q_\ell\right] = 0$$

and the linearization drops to

$$f_i^{n+1} = f_i^n + \alpha_i^n\delta_\eta\left[\sum_{\ell=1}^{4}\left(\frac{\partial\beta_i}{\partial q_\ell}\Big|_n q_\ell^{n+1}\right)\right] \tag{19b}$$

Using this linearization in Eq. (9)

$$R_e^{-1}\partial_\eta\hat{S}^{n+1} = R_e^{-1}\partial_\eta(\hat{S}^n + J^{-1}\hat{M}^n\hat{q}^{n+1}) \tag{20}$$

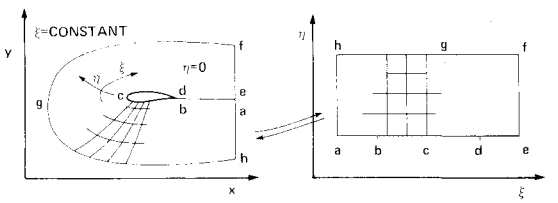


**Fig. 1 Physical and transfomed computational planes.**

where $M$ is defined by

$$M = \begin{bmatrix} 0 & 0 & 0 & 0 \\ m_{21} & \alpha_1\partial_\eta(\rho^{-1}) & \alpha_2\partial_\eta(\rho^{-1}) & 0 \\ m_{31} & \alpha_2\partial_\eta(\rho^{-1}) & \alpha_3\partial_\eta(\rho^{-1}) & 0 \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix} \tag{21}$$

$$m_{21} = -\alpha_1\partial_\eta(u/\rho) - \alpha_2\partial_\eta(v/\rho)$$

$$m_{31} = -\alpha_2\partial_\eta(u/\rho) - \alpha_3\partial_\eta(v/\rho)$$

$$m_{41} = \alpha_4\partial_\eta[-(e/\rho^2)+(u^2+v^2)/\rho] - \alpha_1\partial_\eta(u^2/\rho) - 2\alpha_2\partial_\eta(uv/\rho) - \alpha_3\partial_\eta(v^2/\rho)$$

$$m_{42} = -\alpha_4\partial_\eta(u/\rho) - m_{21}$$

$$m_{43} = -\alpha_4\partial_\eta(v/\rho) - m_{31}$$

$$m_{44} = \alpha_4\partial_\eta(\rho^{-1})$$

$$\alpha_1 = \mu[(4/3)\eta_x^2+\eta_y^2]$$

$$\alpha_2 = (\mu/3)\eta_x\eta_y$$

$$\alpha_3 = \mu[\eta_x^2+(4/3)\eta_y^2]$$

$$\alpha_4 = \gamma\kappa P_r^{-1}(\eta_x^2+\eta_y^2)$$

and $\hat{M}$ differs from $M$ insofar that it is formed using elements of $\hat{q}$ not $q$.

For practical calculations a turbulence model must also be supplied. In the present calculations a two layer algebraic eddy viscosity model patterned after that used by Cebeci[16] was supplied by Barrett Baldwin Jr. of the Ames Research Center. A detailed description of the model will be published separately by Baldwin.

## IV. Advantages of the Transformed Equations, Grid Generation, and Accuracy

The transformed equations are somewhat more complicated than the original Cartesian form but offer several significant advantages. The main advantage is that boundary surfaces in the physical plane can be mapped onto rectangular surfaces in the transformed plane. Unsteady body motion has also been incorporated into the equations. Another significant aspect of the transformation is that grid points can be concentrated in region that experience rapid change in the flowfield gradients, eventually allowing dynamic remeshing.

### Grid Generation

To take advantage of the generality of the transformed equations, one need a fairly automatic method of generating

smoothly varying grids that fit arbitrary bodies and allow grid point clustering. While several automatic grid solvers have been developed[1-3,17-20] one that is appealing for the present application is the scheme advocated by Thompson et al.[1] and Thames et al.[2] In their method the grid in the physical plane is defined by the solution of a Laplace or a Poisson equation. Grid points are arbitrarily specified on the body boundaries, so even if the Laplace equation is used the generated grid is not orthogonal. The capability to select the location of boundary node points is one of the desirable features of the scheme and Eqs. (3) and (4) do not assume orthogonality.

To actually solve the grid generation equations, the Poisson equation is itself transformed to the specified transform plane and is solved on the same rectangular grid on which the flow equations are solved (see Fig. 1). In the grid generation method of Thompson et al. this entails finding values of $x,y$ on the known $\xi,\eta$ grid by solution of

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} = -J^2 (P x_\xi + Q x_\eta)$$

$$\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} = -J^2 (P y_\xi + Q y_\eta) \qquad (22)$$

where

$$\alpha = x_\eta^2 + y_\eta^2, \quad \beta = x_\xi x_\eta + y_\xi y_\eta, \quad \gamma = x_\xi^2 + y_\xi^2$$

$$P = -\sum_{m=1}^{M} a_m \frac{\xi - \xi_m}{|\xi - \xi_m|} e^{(-c_m|\xi - \xi_m|)} - \sum_{n=1}^{N} b_n \frac{\xi - \xi_n}{|\xi - \xi_n|} e^{(-d_n\sqrt{(\xi-\xi_n)^2 + (\eta-\eta_n)^2})}$$

$$Q = -\sum_{m=1}^{M} a_m \frac{\eta - \eta_m}{|\eta - \eta_m|} e^{(-c_m|\eta - \eta_m|)} - \sum_{n=1}^{N} b_n \frac{\eta - \eta_n}{|\eta - \eta_n|} e^{(-d_n\sqrt{(\xi-\xi_n)^2 + (\eta-\eta_n)^2})}$$

and where $a_m$, $b_n$, $c_m$ and $d_n$ are arbitrary positive coefficients. Values of $x$ and $y$ on the $\xi$ and $\eta$ domain boundaries are known and correspond to the specified $x,y$ grid points in the physical domain. The transformed Poisson equation is nonlinear but it remains elliptic and is easily solved by conventional relaxation methods used, for example, for subsonic potential flow.

The grid generated is smoothly continuous, and grid points can be clustered along the body as desired. Clustering of grid points to the body is seldom adequate if the Laplace equation ($P$ and $Q=0$) is used as the original grid generating equations. The problem that is encountered can be observed in Fig. 2, which illustrates a grid generated about an airfoil. In this case points are specified on the airfoil, a cut behind the airfoil, and all outer boundaries as indicated in the schematic illustration shown in Fig. 1. For these boundary conditions the interior points are poorly clustered in the radial-like $\eta$ direction, away from the trailing edge. To obtain good clustering in this direction, one can adjust the $P$ and $Q$ source terms of Eq. (22) or perhaps some other terms. Alternately, the grid point distribution along a line of constant $\xi$ or $\eta$ can be discarded and reclustered by simple stretching relations (see Ref. 21 for details). Because of its simplicity this is the approach taken here. Figure 3 illustrates this clustering where the $x,y$ distribution of grid points along each line of constant $\xi$ shown in Fig. 2 is redistributed by means of a stretching relation. An expanded portion of the grid about the airfoil is detailed in Fig. 4. Note that the highly skewed lines in the wake occur because $x$ and $y$ are specified on the cut $ab$ and $de$ (see Fig. 1). Floating these values removes this skewness, but the grid spacing along the cut must then be controlled in another way, for example, the source term $P$.

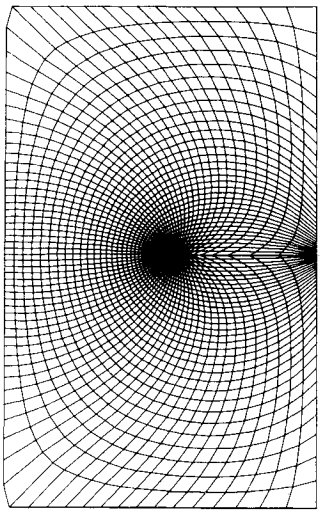

**Fig. 2 Grid generated about an airfoil with $P$ and $Q=0$.**

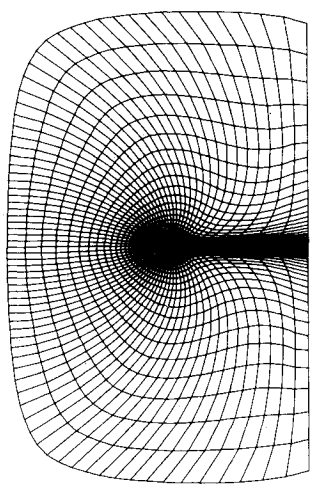

**Fig. 3 Computational grid generated by combining simple stretching in $\eta$ with the Thompson, Thames, and Mastin method.**

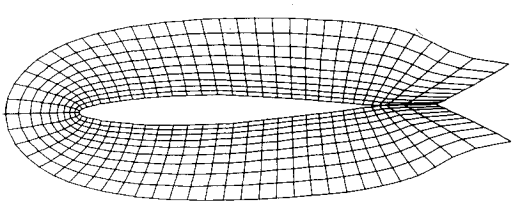

**Fig. 4 Grid detail near body.**

### Accuracy

Once a grid is generated it is a simple matter to difference the $xy$ values in the transformed plane to form $x_\xi$, $x_\eta$, $y_\xi$, $y_\eta$ and from this information form the metrics $\xi_x$, $\xi_y$, $\eta_x$, $\eta_y$, and $J$ using Eq. (6). The metrics themselves should be evaluated with sufficient accuracy that they correctly carry information about geometry; however, an exact evaluation of the metrics would not necessarily lead to the minimum error. Consider, for example, the continuity equation in steady freestream with $v=0$. From Eq. (4)

$$\partial_\xi (J^{-1}\xi_x \rho_\infty u_\infty) + \partial_\eta (J^{-1}\eta_x \rho_\infty u_\infty) = 0 \qquad (23)$$

or using Eq. (6) and taking $\rho_\infty u_\infty$ outside the operator

$$\rho_\infty u_\infty (\partial_\xi y_\eta - \partial_\eta y_\xi) = 0 \qquad (24)$$

If the terms $y_\eta$ and $y_\xi$ are exactly evaluated but the operators $\partial_\xi$ and $\partial_\eta$ are approximated by central differences, then Eq. (24) is not identically zero but is zero to second-order accuracy. In the far field, in which $y$ may vary rapidly, this error

can be appreciable. However, if $y_\eta$ and $y_\xi$ are also centrally differenced, the difference equations cancel identically for constant values of $\rho u$ for any grid stretching.

In fact, the transformed differenced equations are exactly balanced on any deformed rectangular grid for constant values of $\rho$, $\rho u$, $\rho v$, and $e$ if identical central difference operators are used to evaluate both the metric terms and the spatial derivatives of the governing equations. Such difference operators are used throughout this work. Once the dependent variables depart from constant or freestream values, the grid in the physical plane must meet additional constraints if good solution accuracy is to be obtained. In particular, the dependent variables should vary smoothly with $\xi$ and $\eta$ and the metrics should be evaluated with sufficient accuracy.

## V. Implicit Algorithm

A time-implicit numerical algorithm is used to solve the equations because in many flowfield problems it is desirable to take a larger time step than that permitted by a conventional explicit scheme. Such a situation may occur if the dependent variables experience a more rapid variation with space than with time or if the time accuracy is controlled by boundary conditions which act as forcing functions.

In the Beam-Warming[10,11] delta-form approximate-factorization (AF) algorithm, which is used here, the main computational work is contained in the solution of block tridiagonal systems of equations. As a consequence, the transformed flowfield equations (including the viscous terms) are not much more costly to solve than the equations in Cartesian coordinates. The Beam-Warming implicit algorithm has been described elsewhere in various applications[10,11,13,22] and is presented here without elaboration. For either trapezoidal or Euler temporal implicit differencing the delta-form algorithm is given by

$$\begin{aligned}&(I+h\delta_\xi \hat{A}^n - J^{-1}\alpha h\nabla_\xi\Delta_\xi J)\,(I+h\delta_\eta \hat{B}^n - J^{-1}\alpha h\nabla_\eta\Delta_\eta J\\&\quad - R_e^{-1}h\delta_\eta J^{-1}\hat{M}^n)\,(\hat{q}^{n+1}-\hat{q}^n)\\&\quad = -\Delta t(\delta_\xi \hat{E}^n + \delta_\eta \hat{F}^n - R_e^{-1}\delta_\eta \hat{S}^n)\\&\quad - \alpha h J^{-1}[(\nabla_\xi\Delta_\xi)^2 + (\nabla_\eta\Delta_\eta)^2]J\hat{q}^n \qquad (25)\end{aligned}$$

where for the convection terms $\delta_\xi$ and $\delta_\eta$ are second-order central difference operators, $h=\Delta t$ or $\Delta t/2$ for first-order or second-order two-level time differencing, and for convenience the spatial indices are deleted throughout. For second-order accuracy all of the metric terms should be evaluated at time level $n+\frac{1}{2}$ and $\hat{A}$, $\hat{B}$, and $\hat{M}$ are defined in Sec. III. Fourth-order dissipation terms $(\nabla_\xi\Delta_\xi)^2$ and $(\nabla_\eta\Delta_\eta)^2$ are explicitly appended to the right-hand side to control numerical stability where, for example, $\nabla_\xi$ is the conventional backward difference, $\nabla_\xi = I - E_\xi^{-1}$ and $E_\xi^{\pm 1}q_{jk} = q_{j\pm 1,k}$. For consistency, these terms are multiplied by $\Delta t$ and here $\alpha = 0(1)$. To alleviate restrictive stability bounds ($\alpha\Delta t < 1/16$), second differences operating on $\hat{q}^{n+1}-\hat{q}^n$ are inserted into the implicit factors. By a linear stability test with $h\equiv\Delta t$ one finds that $\alpha\Delta t \le \frac{1}{2}$. Implicit use of fourth-order differences as dissipation terms would permit any value of $\alpha\Delta t$ but would require block-pentadiagonal inversions. Note that the dissipation terms have been scaled by the Jacobian determinate to maintain a freestream solution, so the numerical dissipation terms are not conservatively differenced.

The viscous terms are central-differenced in the usual way. Any one term has the form $\partial_\eta\alpha\partial_\eta\beta$ and is differenced as

$$\begin{aligned}\delta_\eta\alpha\delta_\eta\beta = &[(\alpha_{j,k+1}+\alpha_{j,k})(\beta_{j,k+1}-\beta_{j,k})\\&-(\alpha_{j,k}+\alpha_{j,k-1})(\beta_{j,k}-\beta_{j,k-1})]/[2(\Delta\eta)^2] \qquad (26)\end{aligned}$$

For transonic flow cases, upwind second-order trapezoidal (or Padé) spatial differencing is used in the $\xi$ direction for the last several points prior to the shock wave in order to prevent upstream shock capturing oscillations. The operator is transitioned as described by Beam and Warming[10] and the $\xi$-direction dissipation term must be removed in regions of upwind differencing. In the freestream, the Padé upwind difference operator does not lead to perfect difference cancellation as does the central difference operator, but in the current application it is employed only in those portions of the grid that gradually vary.

Finally, it is remarked that higher-order spatial accuracy for the convection terms is simply achieved in the steady state by using fourth-order accurate central differences for the right-hand side operators $\delta_\xi$ and $\delta_\eta$. However, this differencing has not been combined with upwind differencing and, according to linear stability theory, it is unstable when $h=\Delta t/2$.

To solve Eq. (25) one first forms the right-hand side terms, beginning with the smoothing operator and then the steady part of the partial differential equation. These values are temporarily stored in $\hat{q}^{n+1}$. The block tridiagonals in $\xi$ are then formed and solved with the result again stored in $\hat{q}^{n+1}$. Finally, the block tridiagonals in $\eta$ are formed, solved, and the correct value of $\hat{q}^{n+1}$ is found by adding $\hat{q}^n$ to the result.

Throughout the inversion process values of $\hat{q}^{n+1}$ are assumed to equal $\hat{q}^n$ on the flowfield boundaries. This approximation results in a first-order accurate error in time on the boundary, but leads to a simple and flexible scheme. In the present formulation new values of $\hat{q}$ are obtained on the body boundary at the start of each time step by linear extrapolation of the flowfield for $\rho$ and $U$ and using Eq. (11) to find new values of $u$ and $v$. Updated values for surface pressure are obtained from Eq. (12) by central differencing $p_\xi$, forward differencing $p_\eta$, and solving a tridiagonal system of equations for $p$ along the body surface. These data are then converted into values of $\rho$ $\rho u$, $\rho v$, and $e$. For the particular problem shown in Fig. 1, values of $\rho$, $\rho u$, and $\rho v$ are found on the downstream boundaries *ef* and *ah* by extrapolation and $e$ is found from Eq. (2) by maintaining $p=p_\infty$. Flowfield values along the common cut *de* and *ba* are found by averaging linear extrapolates of the variables from above and below. Finally, the numerical dissipation terms are dropped to second differences at points adjacent to the boundaries and are dropped altogether at body surfaces in viscous flows.

All of the boundary conditions could have been at least partially incorporated into the implicit inversion process; doing so would have improved the time accuracy and perhaps allowed larger time steps to be taken without instability. However, the inclusion of a particular set of boundary conditions into the inversion process leads to a more complicated program that is not as readily converted to other flowfield problems.

## VI. Results

A variety of flowfields about airfoils have been computed to test the combination of numerical algorithm, grid mapping, and boundary conditions for the transformed flow equations. Inviscid and viscid, steady and unsteady cases have been run.

A subcritical flow solution and comparison are shown in Fig. 5. The flow was computed about a NACA 0012 airfoil at an angle of attack $\alpha$ of 2 deg using a grid similar to that shown in Fig. 3. Some 49 grid points were distributed over the airfoil and the minimum grid spacing in the $\eta$-direction was 0.01 chords. The grid was stretched smoothly to 8-12 chord lengths away from the body, depending on whether the outer boundary was horizontal or vertical to the airfoil. The present solution accuracy is satisfactory in comparison to the Sells[23] solution proposed by Lock[24] as a test case, although better leading-edge grid refinement is desired.

A transonic solution obtained with and without upwind differencing is shown for the same airfoil in Fig. 6 with

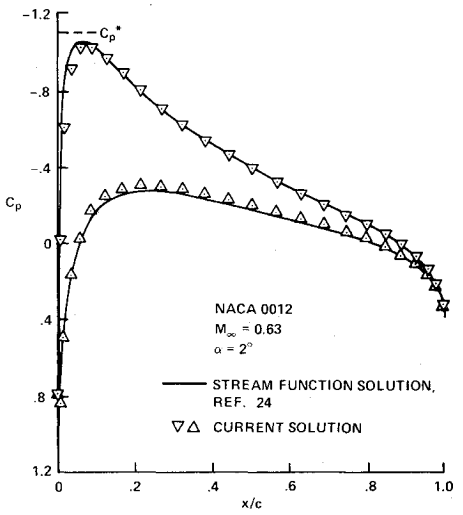


Fig. 5 Inviscid subcritical comparison.

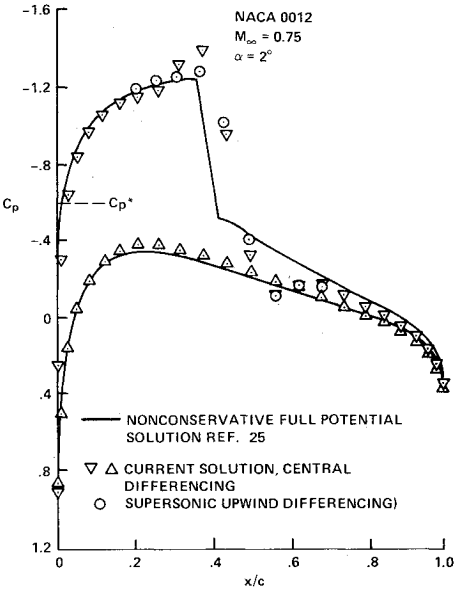


Fig. 6 Inviscid transonic comparison.

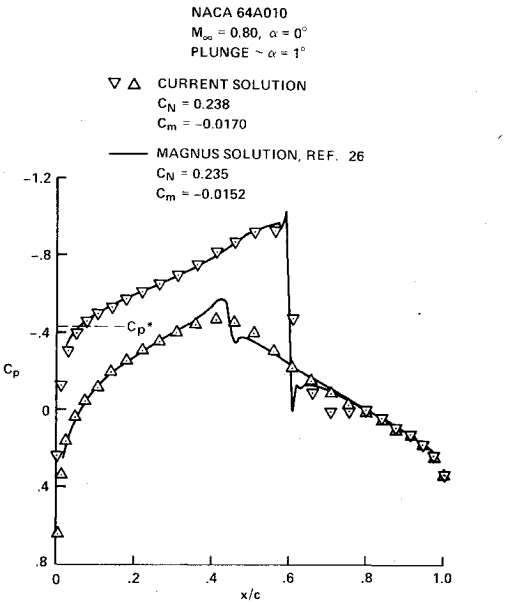


Fig. 7 Transonic airfoil solution at a constant plunge velocity.

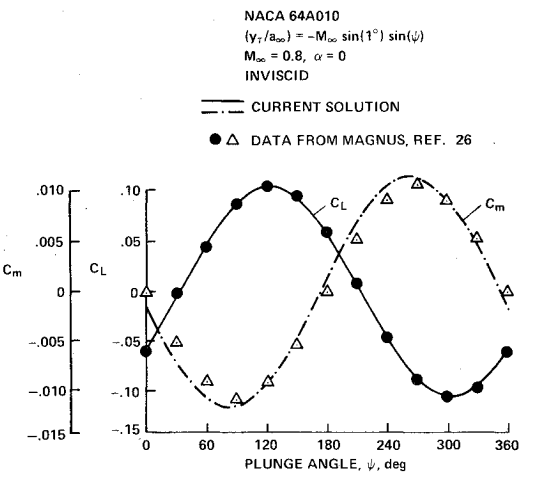


Fig. 8 $C_L$ and $C_m$ variation during the fourth cycle of sinusoidal plunging motion.

$M_\infty = 0.75$ and $\alpha = 2$ deg. For comparison, a nonconservative full potential solution[25] is shown which tends to place the shock wave too far upstream. The effect of angle of attack can also be achieved by plunging the airfoil at a constant rate by simply setting $\eta_t = -y_\tau \eta_y$ and $\xi_t = -y_\tau \xi_y$. A solution about a NACA 64A010 airfoil in a plunge equivalent to an angle of attack of 1 deg ($y_\tau = -a_\infty M_\infty \sin(\pi/180)$ where $a_\infty$ is the nondimensional reference speed) is compared to a very accurate calculation due to Magnus[26] in Fig. 7. Again, the solution accuracy is good although clearly the weak shock on the lower surface is highly diffused in the coarse grid. In the preceding calculations approximately 800 time steps are required to reach a steady state.

A sinusoidally plunging airfoil comparison was also made with the Magnus[26] program and in Fig. 8 both $C_L$ and $C_m$ comparisons during the fourth cycle are shown. The airfoil in this case is plunging between $\pm 1$ deg with a reduced frequency of 0.4 [where the reduced frequency is defined by $k = 2\pi/(t_p M_\infty)$ and $t_p$ is the nondimensional time period, $t_p \equiv t_p a_\infty$/(chord length)]. The lift comparison with Magnus is excellent, but the moment about the quarter chord differs somewhat in amplitude and phase. The moment, however, is very sensitive to shock smearing and this discrepancy can be attributed to the coarser grid used with the present method.

The inviscid calculations basically show the accuracy and flexibility of the code. Good results are obtained on a relatively coarse grid and with use of a finer grid should be excellent.

A series of viscous flow calculations was made with very fine grid spacings in the $\eta$ direction. For example, for an $Re = 11 \times 10^6$ simulation the minimum grid spacing near the body was 0.000025, but the grid is stretched exponentially away from the body. As mentioned previously, a two-layer eddy viscosity model was provided by Baldwin for turbulent flow simulations.

To substantiate the numerical algorithm a laminar flow calculation was made at low Mach number ($M_\infty = 0.2$) and in Fig. 9 is compared to one of Mehta's[27] excellent incompressible solutions for flow about an NACA 0012 airfoil in zero angle of attack and $Re = 10^4$. Like the inviscid solutions, the agreement is good for the grid size used, but the discontinuous change of body shape at the airfoil trailing edge does result in small spurious oscillations in that region. This error is accentuated at lower Mach numbers and for $M_\infty = 0(0.1)$ the numerical algorithm became generally inaccurate with very poor steady-state convergence.

A series of exploratory calculations was made with an 18% thick biconvex airfoil at zero angle of attack using a relatively

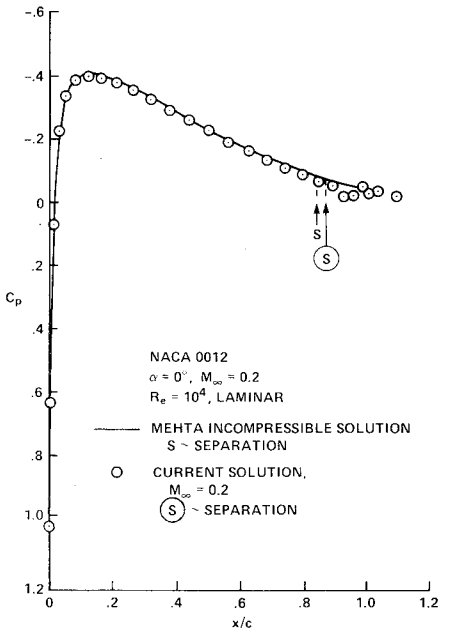


**Fig. 9 Laminar flow comparison.**

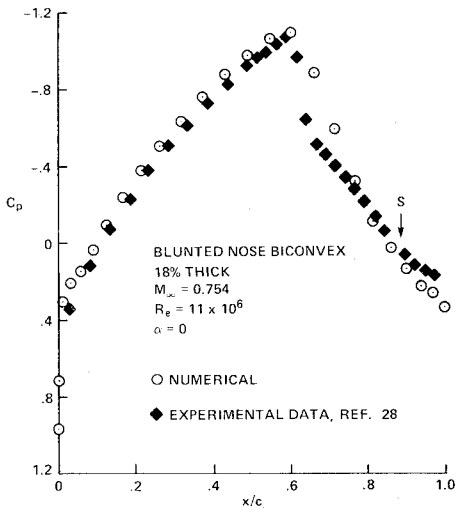


**Fig. 10 Computed pressure distribution over an 18% thick biconvex airfoil.**

coarse grid. Experimental data on this airfoil were taken by McDevitt et al.[28] in the high Reynolds number channel at Ames Research Center. Numerical calculations were performed by Deiwert[29,30] and, more recently, Levy[31] has been exploring the buffet range of this airfoil, using the Deiwert code with MacCormack's new modifications to improve efficiency.[32] In Levy's calculation, the leading edge of the biconvex airfoil was blunted slightly and the same airfoil was used in the following calculations. In the experiment, the airfoil buffets through the Mach number range 0.76 to 0.78, although this range can be extended depending on whether the tunnel Mach number is brought up or brought down from a previous Mach number value. In Levy's calculations, strong buffeting occurs at $M_\infty = 0.754$ with a reduced frequency $k = 0.81$. In the present calculation $M_\infty = 0.754$ is very steady and is in relative agreement with experiment (see Fig. 10). Even plunging the airfoil for a brief period failed to induce a buffet condition. However, buffeting does occur with the present code at $M_\infty = 0.783$ at a reduced frequency of 0.82 where the period of oscillation was found from the $C_L$ vs $\alpha$ curve shown in Fig. 11. Levy[31] gets essentially steady flow at this Mach number.

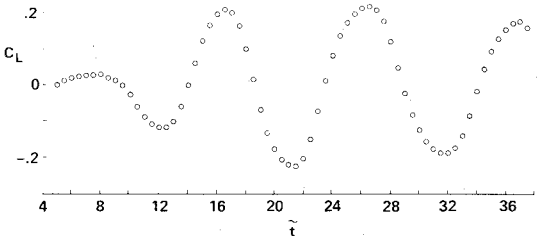


**Fig. 11 $C_L$ vs nondimensional time ($\tilde{t} = 1 \sim 0.783$ chords traveled) for an 18% thick biconvex airfoil at $M_\infty = 0.783$ and $\alpha = 0$ deg.**

Computer processor times for inviscid flow calculations using a $77 \times 27$ grid have been approximately 0.75 seconds per time step on a Control Data Corporation 7600 computer and approximately 1.25 second per time step for the viscous cases (with turbulence model) on a $71 \times 33$ grid. The present program is strictly a research code, and computer processor unit (CPU) times should be reduced by ¼ to ½ by careful reprogramming.

Even without assuming improvements to the numerical algorithm, significant improvements in CPU time will likely be obtained by exploiting the organized data structure of the overall numerical algorithm. This is because arbitrary geometries are mapped onto grid lines of a well-ordered rectangular grid (in contrast to some poorly ordered finite-element methods that use triangular elements) and the spatial operators have been factored into products of easily invertible, one-dimensional operators. Although not employed here, the algorithm can be readily modified to accept interior boundary surfaces. Consequently, the overall numerical algorithm is compatible with vectorized computer processors. At this writing, H. Lomax and H. E. Bailey of Ames Research Center have successfully adapted the basic algorithm (without, for example, upwind differencing) to the ILLIAC IV parallel computer processor.

## VII. Conclusions

An implicit finite-difference scheme has been combined with transformations that permit use of automatic grid generation techniques. The overall algorithm for the equations of motion in conservation law form has considerable flexibility and should be adaptable to a variety of problems. The algorithm is sufficiently robust and efficient for many unsteady flow problems and can be used to obtain steady-state solutions as well.

## References

[1] Thompson, J. F., Thames, F. C., and Mastin, C. M., "Automatic Numerical Generation of Body-Fitted Curvilinear Coordinate System for Field Containing any Number of Arbitrary Two-Dimensional Bodies," *Journal of Computational Physics*, Vol. 15, July 1974, pp. 299-319.

[2] Thames, F. C., Thompson, J. F., and Mastin, C. M., "Numerical Solution of the Navier-Stokes Equations for Arbitrary Two-Dimensional Airfoils," NASA SP-347, Pt. I, March 1975, pp. 469-530.

[3] Ghia, U. and Ghia, K. N., "Numerical Generation of a System of Curvilinear Coordinates for Turbine Cascade Flow Analysis," Univ. of Cincinnati, Rept. No. AFL 75-4-17, 1975.

[4] Lapidus, A., "A Detached Shock Calculation by Second-Order Finite Differences," *Journal of Computational Physics*, Vol. 2, Nov. 1967, pp. 154-177.

[5] Viviand, H., "Conservative Forms of Gas Dynamic Equations," *La Recherche Aerospatiale*, No. 1, Jan.-Feb. 1974, pp. 65-68.

[6] Vinokur, M., "Conservation Equations of Gas-dynamics in Curvilinear Coordinate Systems," *Journal of Computational Physics*, Vol. 14, Feb. 1974, pp. 105-125.

[7] MacCormack, R. W. and Paullay, A. J., "The Influence of the Computational Mesh on Accuracy for Initial Value Problems with Discontinuous or Nonunique Solutions," *Computers and Fluids*, Vol. 2, Dec. 1974, pp. 339-361.

[8]Rizzi, A., "Transonic Solutions of the Euler Equations by the Finite Volume Method," *Proceedings of Symposium Transsonicum II*, edited by K. Oswatitsch and D. Rues, Springer-Verlag, 1975, pp. 567-574.

[9]Schiff, L. B., "A Numerical Solution of the Axisymmetric Jet Counterflow Problem," *Lecture Notes in Physics*, Vol. 59, Springer-Verlag, 1976.

[10]Beam, R. and Warming, R. F., "An Implicit Finite-Difference Algorithm for Hyperbolic Systems in Conservation-Law-Form," *Journal of Computational Physics*, Vol. 22, Sept. 1976, pp. 87-110.

[11]Beam, R. and Warming, R. F., "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," AIAA Paper 77-645, June 1977.

[12]Briley, W. R. and McDonald, H., "An Implicit Numerical Method for the Multi-Dimensional Compressible Navier-Stokes Equations," United Aircraft Research Laboratoties, Rept. M911363-6, 1973.

[13]Steger, J. L. and Kutler, P., "Implicit Finite-Difference Procedures for the Computation of Vortex Wakes," AIAA Paper 76-385, July 1976.

[14]Peyret, R. and Viviand, H., "Computation of Viscous Compressible Flows Based on the Navier-Stokes Equations," AGARD-AG-212, 1975.

[15]Warming, R. F., Beam, R., and Hyett, B. J., "Diagonalization and Simulations Symmetrization of the Gas-Dynamic Matrices," *Mathematics of Computation*, Vol. 29, Oct. 1975, pp. 1037-1045.

[16]Cebeci, T., "Calculation of Compressible Turbulent Boundary Layers with Heat and Mass Transfer," *AIAA Journal*, Vol. 9, June 1971, pp. 1091-1097.

[17]Barfield, W. D., "Numerical Method for Generating Orthogonal Curvilinear Meshes," *Journal of Computational Physics*, Vol. 5, Feb. 1970, pp. 23-33.

[18]Amsden, A. A. and Hirt, C. W., "A Simple Scheme for Generating Curvilinear Grids," *Journal of Computational Physics*, Vol. 11, March 1973, pp. 348-359.

[19]Chu, W. H., "Development of a General Finite Difference Approximation for a General Domain," *Journal of Computational Physics*, Vol. 8, Dec. 1971, pp. 392-408.

[20]Godunov, S. K. and Prokopov, G. P., "The Use of Moving Meshes in Gas-Dynamical Computations," *U.S.S.R. Computational Mathematics and Mathematical Physics*, Vol. 12, No. 2, 1972, pp. 182-195.

[21]Sorenson, R. and Steger, J. L., "Simplified Clustering of Nonorthogonal Grids Generated by Elliptic Partial Differential Equations," NASA TM-73,252, 1977.

[22]Kutler, P., Chakravarthy, S. R., and Lombard, C. K., "Supersonic Flow Over Ablated Nosetips Using a Unsteady Implicit Numerical Procedure," AIAA Paper 78-213, Jan. 1978.

[23]Sells, C. C. L., "Plane Subcritical Flow Past a Lifting Airfoil," Royal Aircraft Establishment, Tech. Rept. 67146, 1967.

[24]Lock, R. C., "Test Cases for Numerical Methods in Two Dimensional Transonic Flow," AGARD Rept. 575, 1960.

[25]Steger, J. L. and Lomax, H., "Numerical Calculation of Transonic Flow About Two-Dimensional Airfoils by Relaxation Procedures," *AIAA Journal*, Vol. 10, Jan. 1972, pp. 49-54.

[26]Magnus, R. J., "Computational Research on Inviscid, Unsteady, Transonic Flow Over Airfoils," Office of Naval Reseach, CASD/LVP 77-D10, Jan. 1977.

[27]Mehta, U., Private communication, NASA Ames Research Center, Moffett Field, Calif., 1976.

[28]McDevitt, J. B., Levy, L. L., Jr., and Deiwert, G. S., "Transonic Flow About a Thick Circular-Arc Airfoil," *AIAA Journal*, Vol. 14, May 1976, pp. 606-613.

[29]Deiwert, G. S., "Numerical Simulation of High Reynolds Number Transonic Flows," *AIAA Journal*, Vol. 13, Oct. 1975, pp. 1354-1359.

[30]Deiwert, G. S., "Computation of Separated Transonic Turbulent Flows," *AIAA Journal*, Vol. 14, June 1976, pp. 735-740.

[31]Levy, L. L., Jr., "An Experimental and Computational Investigation of the Steady and Unsteady Transonic Flowfield About an Airfoil and a Solid Wall Test Channel," AIAA Paper 77-678, July 1977.

[32]MacCormack, R. W., "An Efficient Numerical Method for Solving the Time Dependent Compressible Navier-Stokes Equations at High Reynolds Number," NASA TM X-73,129, July 1976.