Finite Element Methods for Transonic Flow Analysis

Kenneth E. Tatum*
McDonnell Douglas Corporation, St. Louis, Missouri

A finite element method (FEM) for solution of the conservative full potential flow equation is presented as a means for improving the computation of transonic flows about complex geometries. The method does not require an orthogonal mesh, thus removing a major constraint in grid generation. A standard Galerkin finite element formulation is used in conjunction with the artificial compressibility method which stabilizes the algorithm in transonic flow and allows the capture of embedded shock waves. Solution comparisons, including one test case which stresses the capabilities of state-of-the-art finite difference methods, verify the FEM accuracy and geometric capability.

I. Introduction

WHILE the computation of inviscid two- and three-dimensional flowfields by finite difference methods has now become commonplace in industry, one of the most important technological deficiencies is in the area of geometric modeling, particularly for transonic flows. The finite element method (FEM) is presented as a way of alleviating some problems of geometric modeling which limit the effectiveness of current finite difference methods. Certain specific advantages of the FEM, if fully exploited, may improve greatly the computational state-of-the-art.

Since the governing equations of inviscid transonic potential flow are nonlinear, the method of superposition (i.e., panel methods) is insufficient to yield a proper solution, even though solutions have been obtained by a method combining surface singularities and field differences.[1] Also, because of the nonlinearity, similarity laws do not exist which allow subsonic flows to be related to transonic flows. Field methods must be used, typically finite difference or finite element methods, requiring a grid of points to be defined, both on the body surface and in the flowfield, on which solutions to the governing equations are approximated.

Finite difference method (FDM) computer codes generally are tailored to a specific grid topology. Body geometries that fit this topology are analyzed accurately by the corresponding FDM computer code. However, severe loss of grid-line orthogonality often occurs as a grid is body fitted about a new geometry not suited to the particular topology. Some loss of orthogonality occurs about all but the simplest geometries. The effects of nonorthogonality may be reduced by increasing the number of grid points or by redesigning the grid for each new complex geometry. The first approach is very costly, while the second is inconvenient at best, possibly requiring a separate computer code for each grid configuration.

Clearly, the orthogonality constraint causes the definition of a general body-conforming grid for an arbitrary geometry to be a very complicated task. The transonic FEM described herein allows the use of nonorthogonal grids for transonic computations, thereby easing the task of grid generation about complex geometries.

The FEM is a method of approximating the solution of a partial-differential equation (PDE) by specific simple functions defined over small local domains. An integral formulation is used to describe a weak solution to the original PDE. The integration is carried out over finite-sized elements using an assumed general form of the solution. The potential values which scale the element solutions are determined by satisfying boundary conditions and degrees of continuity across element boundaries. Being an integral method, no fundamental constraint of orthogonality is imposed on the grid defining the elements.

A transonic analysis code employing the FEM has been developed based on a Galerkin integral formulation of the full potential equation in strong conservation law form. The density is defined as a function of the gradient of the potential. The method is fundamentally stable for subsonic flow where the equations are elliptic, and the artificial compressibility method of Hafez et al.[2] stabilizes the computations in transonic flow.

The algorithm for computing upwind components of the density is the only part of the new algorithm that might require near orthogonality. However, this modification to the basic FEM has been developed carefully to avoid a need for alignment of grid lines with stream or coordinate directions and is a generalization of rotated upwind finite differencing on a nonorthogonal grid. Thus, skewed grids are not necessarily undesirable for FEM computations as they are for finite difference computations.[3,4] Thus, without a requirement of orthogonality, the two-dimensional transonic FEM code has increased capabilities for computing solutions about complex bodies.

II. Governing Equations

The FEM algorithm is based on the exact compressible potential flow equation written in strong conservation law form as

$$\nabla \cdot (\rho \nabla \Phi) = 0 \tag{1}$$

where Φ is the total velocity potential, ρ the local density, and ∇ the gradient operator. The density is further defined by the isentropic relation

$$\rho = (M_\infty^2 a^2)^{\frac{1}{\gamma-1}} = \left[q_\infty^2 - \frac{\gamma-1}{2} M_\infty^2 (q^2 - q_\infty^2)\right]^{\frac{1}{\gamma-1}} \tag{2}$$

where M, a, q, and γ are the Mach number, local speed of sound, velocity magnitude, and ratio of specific heats. The subscript ∞ refers to freestream conditions. The velocity is defined as $\bar{q} = \nabla\Phi$ due to the assumption of irrotationality. (Vector quantities are denoted by a bar over the variable.) These equations are independent of the coordinate system. Hereafter, the freestream velocity magnitude is normalized to unity.

Presented as Paper 82-0930 at the AIAA/ASME Third Joint Thermophysics, Fluids, Plasma and Heat Transfer Conference, St. Louis, Mo., June 7-11, 1982; submitted June 17, 1982; revision received Nov. 29, 1982.

*Senior Engineer-Technology, Aerodynamics, McDonnell Aircraft Company. Member AIAA.

III. FEM Formulation

A standard Galerkin formulation of Eq. (1) yields

$$\iint_R \nabla \cdot (\rho \nabla \Phi)\psi \mathrm{d}A = 0 \tag{3}$$

where ψ is an arbitrary function defined over a domain R with area A. Similar formulations have been employed previously by Deconinck and Hirsch,[5] Ecer and Akay,[6] and Habashi and Hafez.[7] By Green's theorem, Eq. (3) may be rewritten as

$$\iint_R \rho \nabla \Phi \cdot \nabla \psi \mathrm{d}A = \oint_{\partial R} \rho \frac{\partial \Phi}{\partial n} \psi \mathrm{d}s \tag{4}$$

where s and n denote local coordinates tangent and normal to the boundary ∂R. The application of Green's Theorem reduces the order of differentiation of Φ, thus allowing a reduction in the order of the approximation functions. Also, the line integral simplifies the application of Neumann boundary conditions.

For application to two-dimensional flow about an airfoil, the domain R is chosen as depicted in Fig. 1. The boundary ∂R is defined by the curve ABCDEF. Along the airfoil surface ABC, Neumann boundary conditions are applied while Dirichlet conditions of uniform flow are imposed directly on the far field DEF, rather than enforcing Eq. (4) where $\partial\Phi/\partial n$ is unknown. The vortex sheet surfaces CD and AF are artificial boundaries across which the total potential is discontinuous but the pressure is continuous. The potential Φ is periodic along any closed curve about the airfoil with the jump determined by the Kutta condition for lifting flows. An O grid is defined within R with a periodic coordinate ξ forming closed curves about the airfoil and a transverse coordinate η extending from the airfoil to the far field. Based on this domain definition the total potential Φ may be defined as the sum of a reduced potential Φ, and a vortex

$$\Phi = \phi + \Gamma\xi \tag{5}$$

where Γ is a measure of the vortex strength and ξ the periodic coordinate about the airfoil. The reduced potential is continuous across the vortex sheet when the vortex strength equals the circulation required by the trailing edge Kutta condition. Substitution of Eq. (5) into Eq. (4) yields

$$\iint_R \rho \nabla \phi \cdot \nabla \psi \mathrm{d}A + \iint_R \rho \Gamma \nabla \xi \cdot \nabla \psi \mathrm{d}A = \oint_{\partial R} \rho \frac{\partial \Phi}{\partial n} \psi \mathrm{d}s \tag{6}$$

Since Dirichlet and periodic boundary conditions are imposed on the far field and vortex sheets, respectively, the line integral is evaluated only on the airfoil surface where Neumann conditions are applied. Written in Cartesian coordinates, we now have

$$\iint_R \rho\phi_x\psi_x + \rho\phi_y\psi_y \mathrm{d}x\mathrm{d}y = -\iint_R \rho\Gamma[\xi_x\psi_x + \xi_y\psi_y]\mathrm{d}x\mathrm{d}y + \oint_{\partial R} \rho \frac{\partial \Phi}{\partial n}\psi \mathrm{d}s \tag{7}$$

for the reduced potential.

The domain is treated as a summation of small discrete subdomains called elements. The elements e_i are unique, nonoverlapping, simple regions (e.g., triangles or quadrilaterals) such that

$$R = \sum_{i=1}^{M} e_i \tag{8}$$

The corners of the elements are called nodes, or mesh points, designated P_i. There are M elements in R.

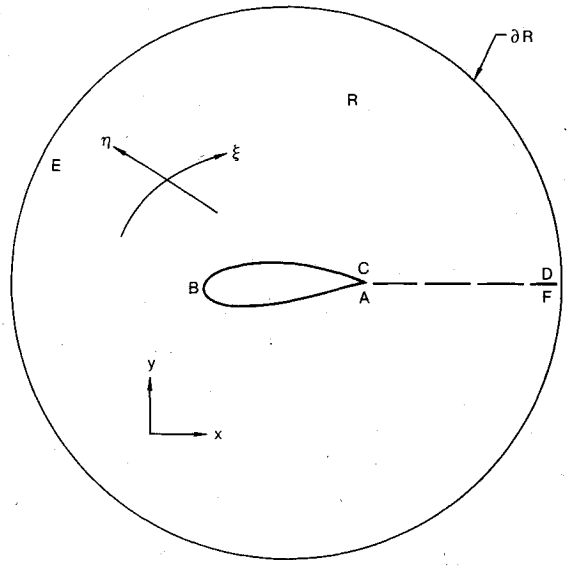

Fig. 1 Flowfield defined about an airfoil for two-dimensional FEM calculations.

The arbitrary function ψ is defined to be the linear combination of simple test functions ψ_i, each of which is locally nonzero only over some particular small finite area, thus,

$$\psi = \sum_{i=1}^{N} \psi_i(x,y) \tag{9}$$

Specifically, each ψ_i is unity at a single node P_i and decreases monotonically to zero over each adjacent (neighbor) element (elements containing P_i). Over all other elements ψ_i is defined to be identically zero. Based on this definition of ψ_i there exists only one nonzero ψ_i at each node, while on each element there overlap k nonzero ψ_i's where there are k corners (nodes) per element.

The reduced potential and density are approximated by $\tilde{\phi}$ and $\tilde{\rho}$ such that

$$\phi \sim \tilde{\phi} = \sum_{i=1}^{N} G_i\psi_i(x,y) \qquad \rho \sim \tilde{\rho} = \sum_{i=1}^{N} D_i\psi_i(x,y) \tag{10}$$

where there are N nodes in R and on ∂R. The G_i and D_i are constants while the ψ_i are interpolation functions defined to be identical to the test functions. For evaluation of $\tilde{\phi}$ and $\tilde{\rho}$ at an arbitrary point (x,y) within an element e_j, the summations of Eq. (10) involve only those nonzero ψ_i defined on e_j. The G_i and D_i are the approximated reduced potential and density at the nodal points.

Employing Eqs. (9) and (10) the integrals of Eq. (7) are evaluated over each element within the domain. The contributions of all M elements are then combined into a global system of algebraic equations through a summation, or assembly, algorithm. The line integral of Eq. (7) is evaluated only on element boundaries coinciding with the airfoil surface since Dirichlet and periodicity conditions replace the formulation on all other boundaries. In standard summation and differentiation notation, utilizing Eqs. (9) and (10), Eq. (7) is

$$D_kG_jI_l = -\Gamma D_kI_2 + \oint_{\partial R} \rho\Phi_n\psi_i \mathrm{d}s \tag{11}$$

where

$$I_l = \iint_e [\psi_{j,x}\psi_{i,x} + \psi_{j,y}\psi_{i,y}]\psi_k \mathrm{d}x\mathrm{d}y$$

and

$$I_2 = \iint_e [\xi_x\psi_{i,x} + \xi_y\psi_{i,y}]\psi_k \mathrm{d}x\mathrm{d}y$$

and the integrals I_1 and I_2 are over the local element e. For tangential flow on the airfoil surface, $\rho\Phi_n = 0$ and the line integral disappears. Equation (11) represents a nonlinear system of algebraic equations for the nodal values of the reduced potential and the density, G_j and D_k. The coefficients are functions of the basis functions, the circulation, and the normal flow on the boundary.

The elements are chosen to be isoparametric quadrilaterals as described in Ref. 8 and bilinear functions of the isoparameteric coordinates define the basis functions ψ_i. The quadrilaterals are positioned along the (ξ,η) coordinate lines (Fig. 1) in an O grid configuration. Thus, the local isoparametric coordinates coincide with (ξ,η) and the basis functions are known bilinear functions of (ξ,η). Also, the vortex flow of Eq. (5) is quasi-one-dimensional through each element. Utilizing the isoparametric transformations, the integrals I_1 and I_2 may be rewritten in terms of (ξ,η) and evaluated numerically by means of Gaussian quadrature to yield two matrices for each element. These matrices are the building blocks for the algebraic equations to be solved.

The integrands of I_1 and I_2, when written explicitly as functions of ξ and η, are quotients of a cubic function and a linear function in both ξ and η variables. Since an n-point Gaussian quadrature exactly integrates polynomials of order $2n$-1 or less,[8] the integrals require at least two-point quadrature rules in each coordinate direction as opposed to the one-point rule of current finite volume methods. The one-point rule decouples the equations for adjacent nodes and extra terms must be introduced to recouple the equations.[9] Experience has shown that two-point integration yields optimum error vs computational effort.

IV. Grid Generation

A simple combination of algebraic and conformal mapping techniques is employed to generate the O grids on which the FEM is applied. A detailed description of the method is given in Ref. 10. The method of grid generation is entirely independent of the FEM solution algorithm and other grid generation schemes may also be used. However, for two-dimensional airfoil calculations the current method is very flexible and inexpensive to use. No attempt has been made to maintain grid line orthogonality.

The grid generation method is based on separate conformal (complex sine function) mappings of a rectangular region to a semioval region for the upper and lower surfaces. Figure 2 illustrates the conformal mapping. A sequence of shearing and stretching transformations, both prior to and subsequent to the conformal mapping, shape the line $\overline{E'F'A'}$ in Fig. 2 to that of one surface of the airfoil. This procedure is used twice to form two mesh regions as shown in Fig. 3, one about the upper surface and one about the lower surface. The airfoil is defined with the forwardmost and aftmost points at A and B as shown in Fig. 3 and the two regions are designed to match along the lines $\overline{BD}$ and $\overline{AC}$. Points along $\overline{BD}$ are merged while points along $\overline{AC}$ are doubly specified, thus creating a cut on which the trailing vortex sheet is defined.

The method can easily generate finite element meshes about complex geometries with sharp corners, including sharp leading edges, since the problem of locating a singular point inside the leading edge, as needed by parabolic mappings, is avoided. Also, the method can be used to generate meshes of either triangular or quadrilateral elements. Sample meshes are shown in Figs. 4 and 5 to illustrate the method's versatility.

While both triangular and quadrilateral elements have been tested, the quadrilateral has been shown to be more accurate and versatile than the triangle. Further discussions in this text will be restricted to quadrilateral elements.

V. Special Solution Techniques

The following paragaraphs discuss several algorithms designed to satisfy specific requirements in the solution of the system of equations generated by Eq. (11).

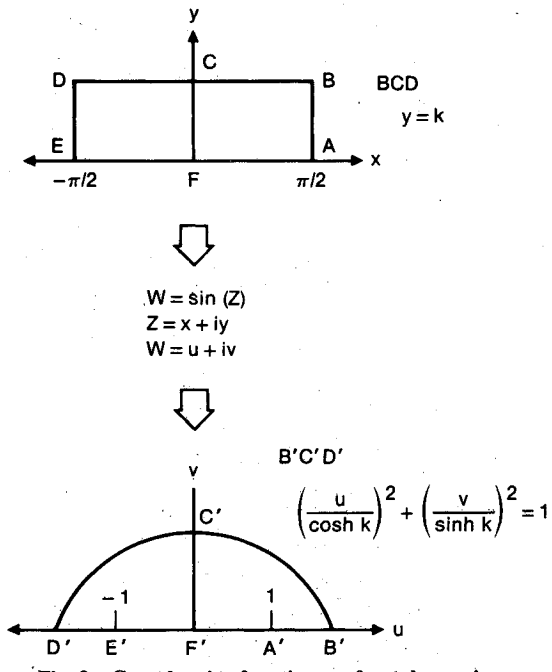

Fig. 2 Complex sine function conformal mapping.

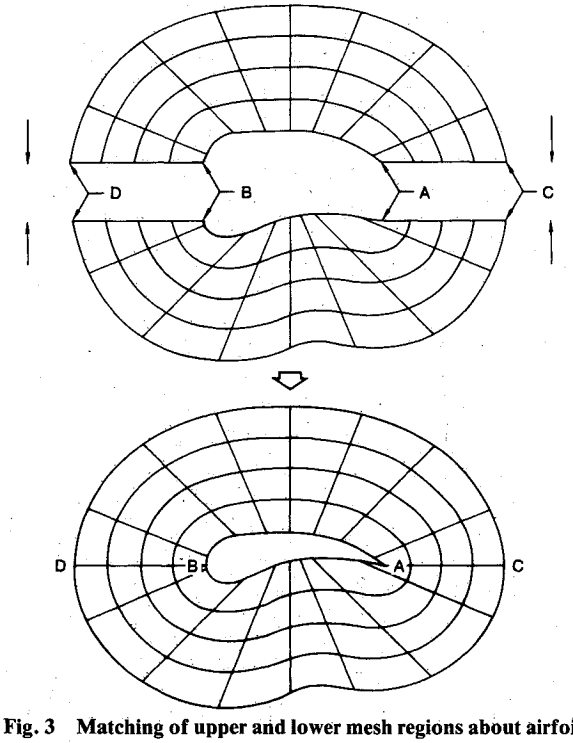

Fig. 3 Matching of upper and lower mesh regions about airfoil.

Nodal Approximations

Corresponding to the bilinear definition of the basis functions ψ_i, the approximations of ϕ and ρ given in Eq. (10) are bilinear in local (ξ,η) coordinates on each element. The gradients of the approximations are thus linear functions of (ξ,η) on each element. This definition results in a global solution that is continuous but not necessarily smooth. Generally, the first derivatives are discontinuous at element boundaries, including nodes.

Given a current approximation to the potential, velocity components may be computed at any point in an element using the linear gradient definition. Then, based on Eq. (2), the density distribution is also defined on the same element.

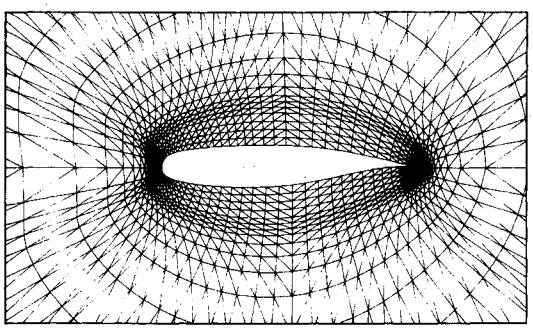

Fig. 4 16.3% thick NLR 7301 airfoil; 72×17 mesh, triangular elements.

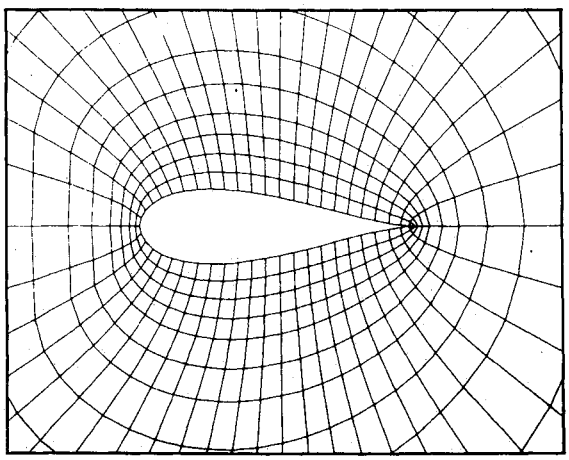

Fig. 5 25.7% thick symmetric Joukowsky airfoil; 48×12 mesh, quadrilateral elements.

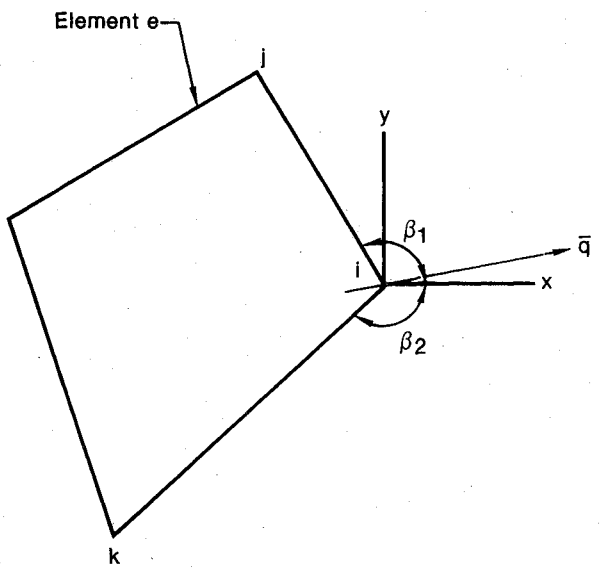

Fig. 6 Relationship of angles β_1 and β_2 to upwind nodes on an arbitrary element.

However, the linearization of Eq. (11) requires values of density at the nodes where the derivatives and, therefore, the density, are discontinuous. To obtain the required nodal values, a nodal velocity is first computed by a weighted average of the element velocity components on elements containing the particular node (neighbor elements). The element velocity components are computed at the element centroid ($\xi=0$, $\eta=0$) for simplicity and the weightings are taken proportional to the element areas. The nodal density is then computed as a function of the nodal velocity by Eq. (2).

Artificial Compressibility

The Galerkin FEM algorithm is well-posed for all domains in which the governing equation is totally elliptic (i.e., subcritical flow). To preserve the proper domain of dependence of the algorithm in local supersonic flow regions, where the governing equation is hyperbolic, the artificial compressibility method of Hafez et al.[1] is used. The method involves the addition of special terms to the definition of the density to account for the upwind shift of the zone of influence in supersonic flow. The original governing equation is written as

$$\nabla \cdot (\rho' \nabla \phi) = 0 \tag{12}$$

where the isentropic density ρ is replaced by an artificial density defined as

$$\rho' = \rho - \mu \rho_{s \leftarrow \Delta s} \tag{13}$$

The finite difference $\rho_{s \leftarrow \Delta s}$ is computed by an upwind formula (indicated by the arrow) along the local streamline s, and μ is an artificial viscosity coefficient.

The upwind differencing is accomplished by a special algorithm designed to avoid any dependence on mesh orthogonality. The element immediately upstream of a given node is defined by the direction of the nodal velocity vector and the isentropic density is determined at the corner nodes of this element. The difference is then computed by

$$-\rho_{s \leftarrow \Delta s} = (\rho_i - \rho_j)\left[\frac{u}{q}\cos\beta_1 + \frac{v}{q}\sin\beta_1\right] + (\rho_i - \rho_k)\left[\frac{u}{q}\cos\beta_2 + \frac{v}{q}\sin\beta_2\right] \tag{14}$$

where the points i, j, and k and the angles β_1 and β_2 are defined in the schematic of Fig. 6. The x and y components of the velocity magnitude q are u and v. If the mesh is orthogonal, the angles β_1 and β_2 are related directly and the expression reduces to a simple difference expression on a rotated coordinate system. Each individual difference, $\rho_i - \rho_j$, is proportionately scaled by the local mesh spacing relative to an average element dimension to eliminate dependence of the upwind differencing on the local mesh spacing.

While the definition of μ is fairly arbitrary, Holst[11] has shown that an efficient form is

$$\mu_i = \max[0, C \cdot (M^2_{i-\frac{1}{2}} - 1)] \tag{15}$$

where C is an arbitrary constant normally set to unity. The subscript $i-\frac{1}{2}$ refers to ½ of an element dimension in the local upstream direction. In practice this value is taken at the centroid of the local upstream element.

Relaxation Scheme

The nonlinear equation system defined by Eq. (11) is linearized by assuming a known density distribution at each step of the solution process. A linear algebraic equation system results for the unknown nodal values of the reduced potential G_j. Solution of this system yields an approximation to the potential which is then used to obtain updated values of velocity and density. The process is repeated until the solution converges.

The linear system for the G_j is solved by a successive-line over-relaxation (SLOR) scheme and the density is updated after each SLOR sweep. For solution of subsonic flow problems the SLOR algorithm may be replaced by a direct Gaussian elimination algorithm. However, for transonic solutions including embedded shock waves, the direct solution method advances in pseudo-time too rapidly and is unstable. For shock-free transonic solutions the direct method is stable but the solutions tend to exhibit waves about corresponding

exact hodograph solutions. A similar linearization/direct-solution method has been shown applicable in transonic small perturbation problems[12,13] but a special shock-point operator as defined by Murman and Cole[14] was utilized to capture the shock wave. The artificial compressibility algorithm, used in conjunction with a relaxation scheme such as SLOR, avoids the need for special handling of shock waves and captures shocks as large gradients in the pressure field.

Vortex Strength Specification

The strength of the vortex defined in Eq. (5) is determined by application of the Kutta condition at the airfoil trailing edge. After a user-specified number of iterations the velocity is computed on the airfoil trailing edge elements on both the upper and lower surfaces. A simple fraction of the difference between these velocity magnitudes determines the amount that the circulation strength is changed for the next iteration. The algorithm has been designed to gradually increase the fraction as convergence proceeds. Numerical experimentation has shown that it should not exceed approximately ¼ for smooth convergence in most cases. The sign of the change in circulation is determined by the sign of the trailing edge velocity magnitude difference.

Programming Applications

The present FEM approach has been programmed for execution on a virtual addressing computer. Virtual memory allows storage of the extensive element matrices I_1 and I_2 of Eq. (11). These matrices (computed by 2×2 Gaussian quadrature integration) require 80 words of memory per element for two-dimensional programs employing quadrilateral elements. Efficient use of virtual memory makes this rather extensive storage requirement preferable to recomputing each matrix every time it is needed. Also, a more efficient version of the FEM based on an elemental, rather than a nodal, density definition reduces the storage to 20 words per quadrilateral element. For three-dimensional applications, the storage per hexahedral element increases from 20 to 72 words.

For triangular elements the element matrices I_1 and I_2 may be evaluated analytically rather than numerically. Thus, storage of all the matrices is less efficient than recomputing them as needed. However, as previously stated, triangular elements have been shown to be less accurate than quadrilaterals. The root-mean-square (rms) error convergence with decreasing mesh size is identical for both triangle and quadrilateral methods, as well as with the second-order finite difference method of Holst.[15] However, the quadrilateral FEM shows a smaller magnitude of rms error for a given average mesh size than either Holst's code or the triangular FEM.

At present, the code runs between one and two thousand SLOR iterations on a single grid depending on the severity of the Mach/alpha inputs. Little effort has been expended in making the code cost effective through such features as grid refinements, etc. A multigrid method would be extremely useful in reducing costs because most iterations take place on coarser meshes where the number of element matrices is much smaller. This would eliminate much of the paging through virtual address space that occurs within each iteration on a fine grid.

VI. Results and Comparisons

To verify the geometric capability of the method, test cases were computed on a number of widely varying geometries. Figures 7-9 are subsonic, transonic nonlifting, and transonic lifting solutions by the FEM code TPOT2 (for transonic potential two-dimensional) compared with two commonly used finite difference codes, FLO-36 (Ref. 16) and TAIR.[15] The NLR 7301 airfoil of Fig. 7 is a thick, blunt supercritical shape with considerable trailing edge loading and a cusped trailing edge. Figures 8 and 9 show solutions for a NACA 0012 airfoil. The only discrepancy between the solutions is in

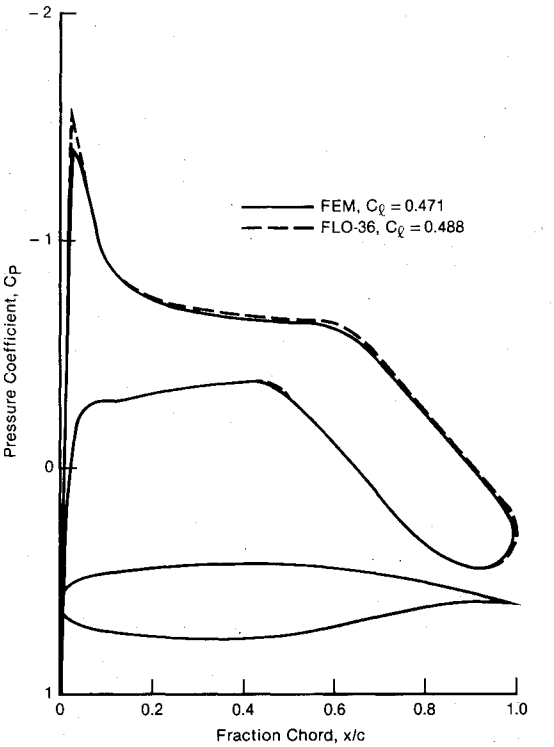

Fig. 7 Potential flow solutions by finite element method and finite difference method-subcritical case, NLR 7301 airfoil; $M_\infty = 0.502$, $\alpha = 0.391$ deg.

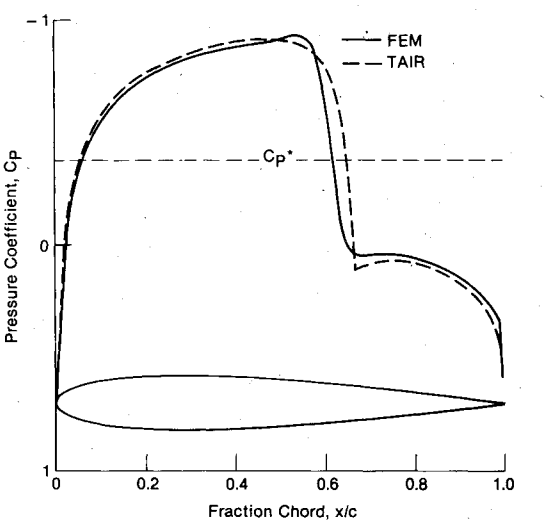

Fig. 8 Potential flow solutions by finite element method and finite difference method-nonlifting case, NACA 0012 airfoil; $M_\infty = 0.82$, $\alpha = 0$ deg.

prediction of shock-wave location (consistently about 3-5% chord). However, no exact solution exists to verify the accuracy of either solution.

To further test the geometric capability of the FEM, solutions were computed on a 6% thick biconvex (circular arc) airfoil. Both the leading and trailing edges of this airfoil are sharp. Two test cases were computed: 1) $M_\infty = 0.806$, angle of attack = 1 deg, and 2) $M_\infty = 0.857$, angle of attack = 1 deg. Case 1 is primarily subsonic while case 2 is transonic with a strong shock wave on the upper surface. Both cases are lifting and exhibit supersonic expansions about the sharp leading edge.

Six solutions were computed by the FEM code TPOT2 for each test case. Only the grid was varied as the Mach number and angle of attack were held constant. No geometric ap-

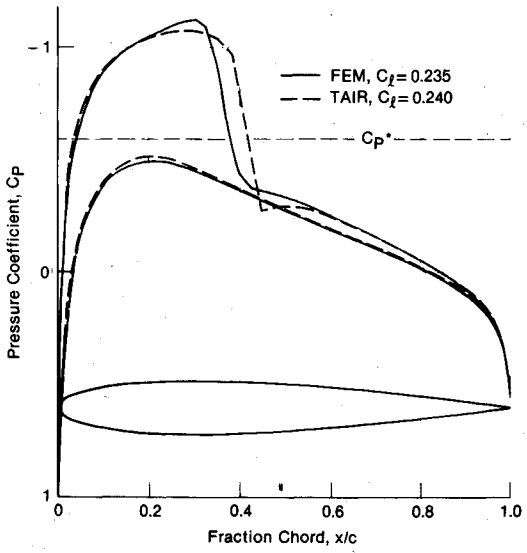

Fig. 9 Potential flow solutions by finite element method and finite difference method-lifting case, NACA 0012 airfoil; $M_\infty = 0.75$, $\alpha = 1$ deg.

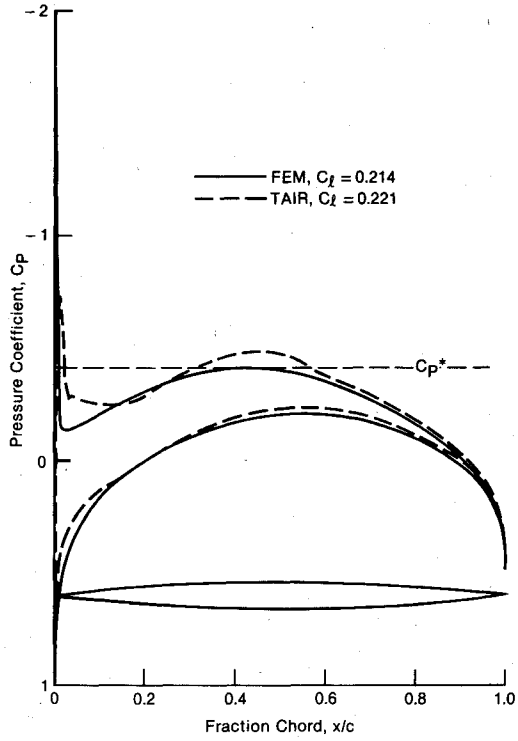

Fig. 10 Potential flow solutions by finite element method and finite difference method-sharp leading edge, 6% thick biconvex airfoil; $M_\infty = 0.806$, $\alpha = 1$ deg.

proximation was made for the sharp leading edge, and all convergence parameters were held constant at the same values as those used in the solutions of Figs. 7-9. The outer grid boundary was fixed at five chord lengths radius about the airfoil midchord location for all grids. The six grid variations included two grid densities and three stretchings of the nodal spacing on the airfoil surface. The fine O grid had dimensions of 96×24 elements and the coarse grid was 48×12. The stretchings varied the degree of nodal clustering about the airfoil leading and trailing edges.

The TPOT2 solutions were considered converged when the maximum residual had been reduced by five orders of magnitude. All solutions for each test case converged smoothly to this criterion independent of grid variations. Fine

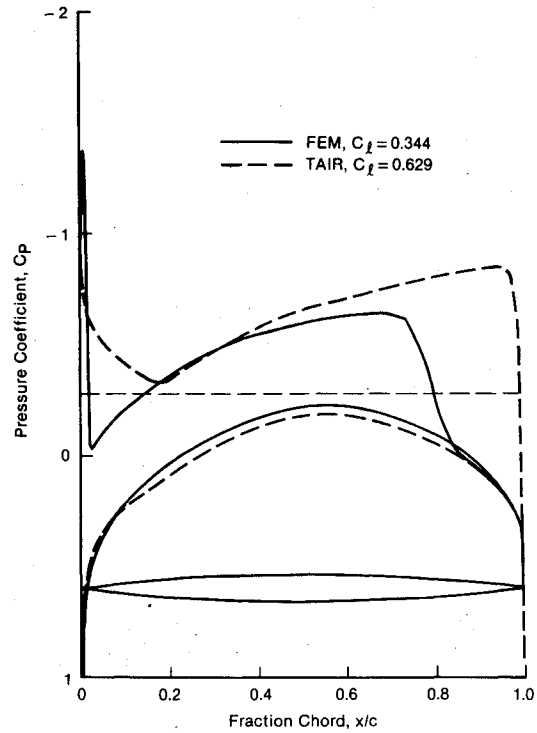

Fig. 11 Potential flow solutions by finite element method and finite difference method-sharp leading edge, 6% thick biconvex airfoil; $M_\infty = 0.857$, $\alpha = 1$ deg.

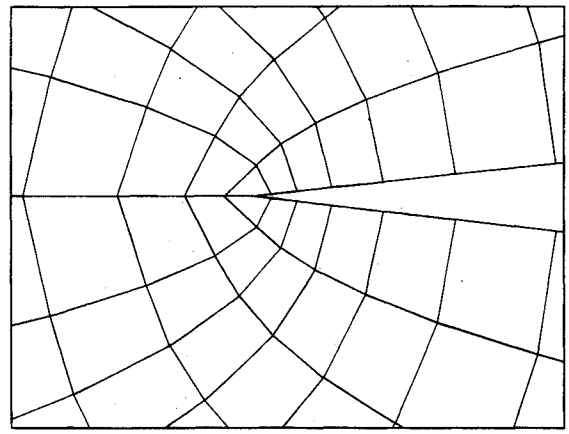

Fig. 12 Coarse FEM mesh about leading edge of 6% thick biconvex airfoil.

grid solutions for the two test cases are shown in Figs. 10 and 11. All other TPOT2 fine grid solutions agreed very closely with those shown, only varying in the degree of resolution of pressure peaks and shock waves. The coarse grid solutions converged just as smoothly as did the fine grid solutions. The resolution of certain features of the flowfield was slightly worse on the coarse grids, as expected, but the solutions were consistent with those on the fine grids. Figure 12 magnifies the coarse grid in the vicinity of the leading edge where the supersonic expansion occurs. On either the upper or lower surface the grid is relatively orthogonal but not directly upstream of the leading edge. The convergence of these coarse grid solutions well illustrates the orthogonality independence of the FEM.

Comparison solutions were computed by the FDM computer code TAIR to verify the FEM solutions and detemine the geometric modeling advantages of the FEM. The FLO-36 code cannot accept a sharp leading edge due to its use of a circle-plane mapping for grid generation. TAIR, however, has provisions built into the code to allow computations about a

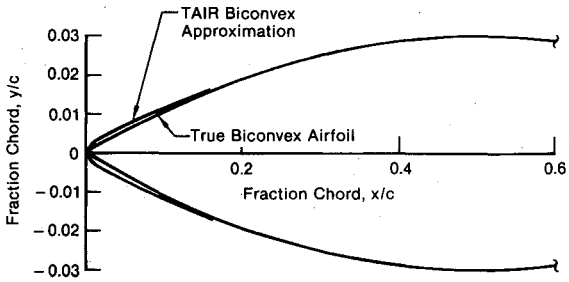

Fig. 13 Biconvex airfoil leading edge geometry, ordinate enlarged by factor of 5.

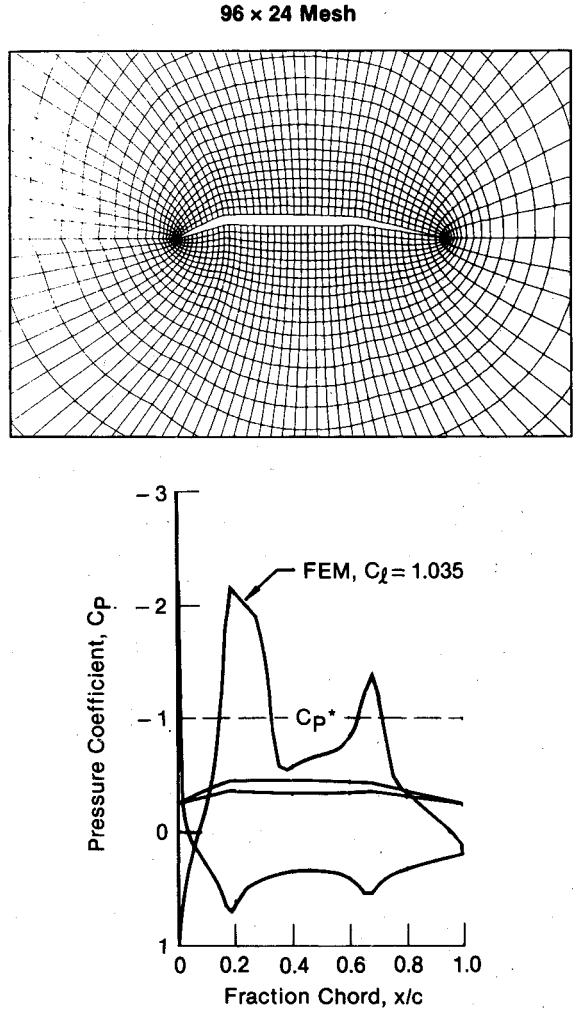

Fig. 14 Potential flow solution for supersonic fighter airfoil and corresponding mesh, 20 deg leading and 10 deg trailing edge flap deflections; $M_\infty = 0.65$, $\alpha = -0.50$ deg.

consistent parabolic approximation of the biconvex airfoil leading-edge shape. A parabola is matched both in thickness and slope at a given x/c location and the blunted airfoil is renormalized. Figure 13 compares the actual 6% biconvex airfoil with the TAIR approximation employing the x/c matching value of 0.2 (the computer code default value).

Biconvex airfoil solutions were attempted with TAIR using a variety of input combinations for each test case. The degree of leading-edge approximation was varied as were the grid dimensions and the airfoil surface node spacing. The O-grid dimensions were taken at 148×29 (TAIR default) and 96×24 cells, the latter matching the TPOT2 grid dimensions. The surface node spacings were varied in the same manner as in the TPOT2 grids. All grid generation calculations were converged one order of magnitude further than recommended by the user's manual[15] and the outer grid boundary was maintained the same as for the TPOT2 grids. The approximate factorization convergence parameters were also chosen conservatively for all TAIR runs.

The degree of leading-edge bluntness proved to be the most critical parameter affecting convergence of the TAIR solutions. For consistent convergence, regardless of the combination of grid dimension or node spacing, the leading edge had to be blunted by at least the amount shown in Fig. 13. For some fine grid cases where the degree of bluntness was only slightly less than that of Fig. 13, TAIR neither diverged nor converged within a large number of iterations. The maximum residual tended to level off after 20-40 iterations and could not be reduced to the usual levels accepted for convergence. In actuality, these solutions were consistent with the converged TAIR solutions. However, in typical cases where the solution is unknown, the attainment of small residuals is often the only criteria by which to judge convergence.

The best converged TAIR solutions are shown in Figs. 10 and 11 for the grid size matching that of TPOT2 along with the TPOT2 solution. While the solutions for case 1 are in relatively good agreement, the case 2 solutions differ drastically. In both cases, the leading edge expansion is resolved more sharply by TPOT2, probably due to the bluntness approximation employed in TAIR. The large difference in shock locations may be due to the difference in leading-edge resolution since TAIR fails to predict subsonic flow on the upper surface just aft of the leading edge.

One final test of the FEM is illustrated by Fig. 14. The airfoil shown is an existing supersonic prototype design having a sharp leading edge. For added geometric complexity, leading- and trailing-edge flaps were deflected 20 and 10 deg, respectively. The FEM code computed the solution at a transonic, high lift condition with no change required in any input variables other than Mach number and angle of attack. The mesh is also shown in the figure.

VII. Conclusions

An application of the finite element method to solution of the transonic potential flow equation has been presented. The artificial compressibility method stabilizes the elliptic FEM spatial discretization operator in transonic (mixed elliptic/hyperbolic) flow when used in conjunction with a relaxation algorithm which advances the solution in a pseudotime. The algorithm is very robust with respect to airfoil geometric variations of thickness, camber, or bluntness. No special treatment or approximation is required at sharp corners. Nor are there any convergence parameter constraints based on the airfoil geometry. Smooth convergence is obtained even on relatively coarse grids.

While transonic finite difference methods are accurate and reliable when applied to most common geometries, the application of such methods to more complex geometries requires special consideration. Loss of local grid orthogonality typically occurs in such cases and techniques for regaining local orthogonality are often expensive or inconvenient, particularly in extensions of current methods to three dimensions. The FEM algorithm described in this report, however, does not require special handling of complex geometries since there is no requirement for an orthogonal grid.

References

[1] Johnson, F. T., James, R. M., Bussoletti, J. E., and Woo, A. C., "A Transonic Rectangular Grid Embedded Panel Method," AIAA Paper 82-0953, June 1982.

[2] Hafez, M. M., Murman, E. M., and South, J. C., "Artificial Compressibility Methods for Numerical Solution of Transonic Full Potential Equation," AIAA Paper 78-1148, July 1978.

[3]Lee, K. D., "3D Transonic Flow Computations Using Grid Systems with Block Structure," AIAA Paper 81-0998, June 1981.

[4]Holst, T. L. and Brown, D., "Transonic Airfoil Calculations Using Solution-Adaptive-Grids," AIAA Paper 81-1010, June 1981.

[5]Deconinck, H. and Hirsch, C. H., "A Finite Element Method Solving the Full Potential Equation with Boundary Layer Interaction in Transonic Cascade Flow," AIAA Paper 79-0132, Jan. 1979.

[6]Ecer, A. and Akay, H. U., "Investigation of Transonic Flow in a Cascade Using an Adaptive Mesh," AIAA Paper 80-1430, July 1980.

[7]Habashi, W. G. and Hafez, M. M., "Finite Element Solutions of Transonic Flow Problems," *AIAA Journal*, Vol. 20, Oct. 1982, pp. 1368-1376.

[8]Chung, T. J., *Finite Element Analysis in Fluid Dynamics*, McGraw-Hill International Book Company, St. Louis, Mo., 1978, pp. 81-87, 202-203.

[9]Jameson, A. and Caughey, D. A., "A Finite Volume Method for Transonic Potential Flow Calculations," AIAA Paper 77-635, June 1977.

[10]Tatum, K. E., "Grid Generation for Two-Dimensional Finite Element Flowfield Computation," *Proceedings of Numerical Grid Generation Workshop*, NASA Langley Research Center, NASA CP 2166, Oct. 1980.

[11]Holst, T. L. and Albert, A., "An Implicit Algorithm for the Conservative, Transonic Full-Potential Equation with Effective Rotated Differencing," NASA TM 78570, April 1979.

[12]Tatum, K. E., "A Finite Element Solution of Transonic Flow in Two Dimensions," Master of Science Thesis, University of Tennessee, Dec. 1977.

[13]Tatum, K. E., "Finite Element Solution of Transonic Flow," AIAA Paper 78-0330, Feb. 1978.

[14]Murman, E. M., "Analysis of Embedded Shock Waves Calculated by Relaxation Methods," *Proceedings of AIAA Computational Fluid Dynamics Conference*, Palm Springs, AIAA CP-73G20, July 1973, pp. 27-40.

[15]Dougherty, F. C., Holst, T. L., Gundy, K. L., and Thomas, S. D., "TAIR—A Transonic Airfoil Analysis Computer Code," NASA TM 81296, May 1981.

[16]Jameson, A., "Acceleration of Transonic Potential Flow Calculations on Arbitrary Meshes by the Multiple Grid Method," AIAA Paper 79-1458, July 1979.