Implicit Flux-Split Euler Schemes for Unsteady Aerodynamic Analysis Involving Unstructured Dynamic Meshes

John T. Batina*

NASA Langley Research Center, Hampton, Virginia 23665

Improved algorithms for the solution of the time-dependent Euler equations are presented for unsteady aerodynamic analysis involving unstructured dynamic meshes. The improvements have been developed recently to the spatial and temporal discretizations used by unstructured grid flow solvers. The spatial discretization involves a flux-split approach, which is naturally dissipative and captures shock waves sharply with at most one grid point within the shock structure. The temporal discretization involves an implicit time-integration scheme using a Gauss-Seidel relaxation procedure, which is computationally efficient for either steady or unsteady flow problems. For example, very large time steps may be used for rapid convergence to steady state, and the step size for unsteady cases may be selected for temporal accuracy rather than for numerical stability. Steady and unsteady flow results are presented for the NACA 0012 airfoil to demonstrate applications of the new Euler solvers. The unsteady results were obtained for the airfoil pitching harmonically about the quarter chord. The resulting instantaneous pressure distributions and lift and moment coefficients during a cycle of motion compare well with experimental data. The paper presents a description of the Euler solvers along with results and comparisons that assess the capability.

Introduction

CONSIDERABLE progress has been made over the past two decades on developing computational fluid dynamics (CFD) methods for aerodynamic analysis.[1,2] Recent work in CFD has focused primarily on developing algorithms for the solution of the Euler and Navier-Stokes equations. For unsteady aerodynamic and aeroelastic analysis, these methods generally require that the mesh move to conform to the instantaneous position of the moving or deforming body under consideration. Many of the methods that are currently being developed assume that the mesh moves rigidly or that the mesh shears as the body deforms. These assumptions consequently limit the applicability of the procedures to rigid-body motions or small-amplitude deformations. Furthermore, these methods of solution typically assume that the computational grid has an underlying geometrical structure. As an alternative, algorithms have been developed recently that make use of unstructured grids.[3-12] In two dimensions these grids are typically made up of triangles, and in three dimensions they consist of an assemblage of tetrahedra. The unstructured grid methods have distinct advantages over structured grid methods in that they can easily treat the most complex of geometric configurations as well as flow conditions, and that the unstructured grid can be moved to treat realistic motions and structural deformations of these configurations.[10-12]

The results presented by the author in Refs. 11 and 12 demonstrated that 1) the methods produce solutions of comparable accuracy to results obtained using structured grid flow solvers,[11] and 2) the unstructured grid methodology can easily

analyze complex aircraft geometries undergoing structural deformation.[12] The methods of Refs. 11 and 12, however, use a spatial discretization based on central differencing with explicit artificial dissipation, and use a temporal discretization involving explicit time-marching based on a multistage Runge-Kutta time integration. The explicit artificial dissipation used in such schemes tends to smear shock waves over several grid cells and requires the tuning of free parameters that scale the dissipation. Also, the explicit Runge-Kutta time-integration has a step size that is limited by the Courant-Friedrichs-Lewy (CFL) condition to very small values. Consequently, thousands (and occasionally tens of thousands) of time steps are required to obtain steady-state solutions, and thousands of steps per cycle of motion are required for unsteady solutions. Therefore, the purpose of this paper is to report on improvements that have been developed recently to the spatial and temporal discretizations of the unstructured grid flow solvers, which resolve the numerical issues described above. The spatial discretization now involves a so-called flux-split approach, which is similar to discretizations presented in Refs. 10, 13, and 14 based on either the flux-vector splitting (FVS) of van Leer[15] or the flux-difference splitting (FDS) of Roe.[16] These flux-split discretizations account for the local wave-propagation characteristics of the flow and they capture shock waves sharply with at most one grid point within the shock structure. A further advantage is that these discretizations are naturally dissipative and consequently do not require additional artificial dissipation terms or the adjustment of free parameters to control the dissipation. Furthermore, the temporal discretization has been changed to an implicit time-integration scheme involving a Gauss-Seidel relaxation procedure similar to discretizations presented in Refs. 17 and 18. This relaxation scheme is unconditionally stable and thus allows the selection of the step size based on the temporal accuracy dictated by the problem being considered, rather than on the numerical stability of the algorithm. Consequently, very large time steps may be used for rapid convergence to steady state, and an appropriate step size may be selected for unsteady cases, independent of numerical stability issues. Steady and unsteady results are presented for the NACA 0012 airfoil to demonstrate applications of the new Euler solvers. The unsteady flow results were obtained for the airfoil pitching harmonically about the quarter chord. The paper presents a description of the Euler solvers along with results and comparisons that assess the capability.

Euler Equations

In the present study, the flow is assumed to be governed by the two-dimensional time-dependent Euler equations that may be written in integral form as

$$\frac{\partial}{\partial t}\iint_{\Omega} Q \, dx \, dy + \int_{\partial\Omega}(F \, dy - G \, dx) = 0 \tag{1}$$

where the vector of conserved variables Q and the convective fluxes F and G are given by

$$Q = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{Bmatrix} \tag{2a}$$

$$F = \begin{Bmatrix} \rho u \\ \rho Uu + p \\ \rho Uv \\ (e+p)U + x_t p \end{Bmatrix} \tag{2b}$$

$$G = \begin{Bmatrix} \rho V \\ \rho Vu \\ \rho Vv + p \\ (e+p)V + y_t p \end{Bmatrix} \tag{2c}$$

The contravariant velocities U and V are defined by

$$U = u - x_t \qquad V = v - y_t \tag{3}$$

where x_t and y_t are the grid speeds in the x and y directions, respectively, and the pressure p is given by the equation of state for a perfect gas

$$p = (\gamma - 1)\left[e - \tfrac{1}{2}\rho(u^2 + v^2)\right] \tag{4}$$

These equations have been nondimensionalized by the freestream density ρ_∞ and the freestream speed of sound a_∞.

Flux-Vector Splitting

The spatial discretization based on flux-vector splitting is a cell-centered scheme where the flow variables are stored at the centroid of each triangle and the control volume is simply the triangle itself. The boundary integral of Eq. (1) is approximated by using the flux-vector splitting of van Leer.[15] In this method the flux vectors are split into forward and backward contributions, which are continuously differentiable even at sonic and stagnation points. The scheme is derived as follows. For each edge of a given triangle, the fluxes are first rotated into a locally Cartesian coordinate system $\bar{x}$-$\bar{y}$ with the principal direction being perpendicular to the edge. The flux in this direction is defined as

$$H\Delta s = T(F\Delta y - G\Delta x) = \begin{Bmatrix} \rho\bar{U} \\ \rho\bar{U}\bar{u} + p \\ \rho\bar{U}\bar{v} \\ e\bar{U} + p\bar{u} \end{Bmatrix} \tag{5}$$

where the transformation matrix T is given by

$$T = \frac{1}{\Delta s}\begin{bmatrix} \Delta s & 0 & 0 & 0 \\ 0 & \Delta y & -\Delta x & 0 \\ 0 & \Delta x & \Delta y & 0 \\ 0 & 0 & 0 & \Delta s \end{bmatrix} \tag{6}$$

In Eqs. (5) and (6), Δx and Δy are the directed lengths of the edge in the x and y coordinate directions, respectively, and $\Delta s^2 = \Delta x^2 + \Delta y^2$. Also, $\bar{u}$ and $\bar{v}$ are the Cartesian velocity components perpendicular and parallel to the edge defined by

$$\bar{u} = u\frac{\Delta y}{\Delta s} - v\frac{\Delta x}{\Delta s} \tag{7a}$$

$$\bar{v} = u\frac{\Delta x}{\Delta s} + v\frac{\Delta y}{\Delta s} \tag{7b}$$

and $\bar{U}$ and $\bar{V}$ are the corresponding contravariant velocities

$$\bar{U} = (u - x_t)\frac{\Delta y}{\Delta s} - (v - y_t)\frac{\Delta x}{\Delta s} \tag{8a}$$

$$\bar{V} = (u - x_t)\frac{\Delta x}{\Delta s} + (v - y_t)\frac{\Delta y}{\Delta s} \tag{8b}$$

The flux vector H is split in a one-dimensional fashion into forward (H^+) and backward (H^-) vectors for $|\bar{U}| \le a$ as

$$H = H^+ + H^- \tag{9}$$

where

$$H^\pm = \begin{Bmatrix} h^\pm_{\text{mass}} \\ h^\pm_{\text{mass}}\left[(-\bar{U} \pm 2a)/\gamma + \bar{u}\right] \\ h^\pm_{\text{mass}}\bar{v} \\ h^\pm_{\text{energy}} \end{Bmatrix} \tag{10}$$

and

$$h^\pm_{\text{mass}} = \pm(\rho/4a)(\bar{U} \pm a)^2 \tag{11a}$$

$$h^\pm_{\text{energy}} = h^\pm_{\text{mass}}\left[\frac{-(\gamma-1)\bar{U}^2 \pm 2(\gamma-1)\bar{U}a + 2a^2}{\gamma^2 - 1} \right. $$
$$\left. + \frac{\bar{u}^2 + \bar{v}^2}{2} + \left(x_t\frac{\Delta y}{\Delta s} - y_t\frac{\Delta x}{\Delta s}\right)(-\bar{U} \pm 2a)/\gamma\right] \tag{11b}$$

The resulting split fluxes are finally rotated back into the original coordinate system so that

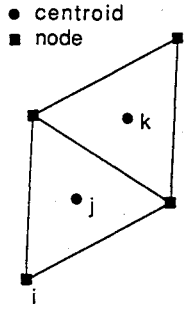$$F\Delta y - G\Delta x = T^{-1}\left[H^+(q^-) + H^-(q^+)\right] \tag{12}$$

where the notation $H^+(q^-)$ and $H^-(q^+)$ indicates that the fluxes $H^\pm$ are evaluated using upwind-biased interpolations of the primitive variables q. For a given triangle j, for example, and considering the diagram in Fig. 1a, the upwind-biased interpolation for q^- along the edge between triangles j and k is defined by

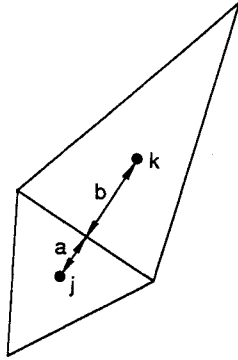$$q^- = q_j + \tfrac{1}{4}\left[(1-\kappa)\Delta_- + (1+\kappa)\Delta_+\right] \tag{13}$$

where

$$\Delta_+ = q_k - q_j \tag{14a}$$

$$\Delta_- = q_j - q_i \tag{14b}$$

• centroid
■ node

a) Centroids and nodes used in construction of upwind-biased flow variables

b) Distances between centroids and midpoint of edge used in Eqs. (15) and (16)

Fig. 1 Diagrams illustrating details of flux-split Euler algorithm implementation.

In Eqs. (13) and (14), q_j and q_k are the vectors of primitive variables at the centroids of triangles j and k, respectively, and q_i, the vector of primitive variables at node i, is determined by an average of the flow variables in the triangles surrounding node i. The upwind-biased interpolation for q^+ along this edge is determined similarly using the flow variables at centroids j and k and the flow variables at node i. The parameter κ in Eq. (13) controls a family of difference schemes by appropriately weighting Δ_- and Δ_+. On structured meshes it is easy to show that $\kappa = -1$ yields a fully upwind scheme, $\kappa = 0$ yields Fromm's scheme, and $\kappa = 1$ yields central differencing. The value $\kappa = \frac{1}{3}$ leads to a third-order-accurate upwind-biased scheme, although third-order accuracy is strictly correct only for one-dimensional calculations. Nevertheless, $\kappa = \frac{1}{3}$ was used in the calculations presented herein.

On highly stretched meshes, the formula for Δ_+ is modified to be

$$\Delta_+ = [2a/(a+b)](q_k - q_j) \tag{15}$$

where a and b are the distances from the midpoint of the edge to the centroids of triangles j and k, respectively, as shown in Fig. 1b. This formula weights the flow variables in the interpolation formula [Eq. (13)] differently to account for the stretching of the mesh. For example, by substituting Eq. (15) into Eq. (13) and letting $\kappa = 1$ yields

$$q^- = [b/(a+b)]q_j + [a/(a+b)]q_k \tag{16}$$

For the case shown in Fig. 1b, Eq. (16) clearly gives more weight in the calculation of q^- to the flow variables at centroid j than to the flow variables at centroid k, since $b > a$. Furthermore, in calculations involving upwind-biased schemes, oscillations in the solution near shock waves are expected to occur. To eliminate these oscillations flux limiting is usually required. The flux limiter modifies the upwind-biased interpolations for q^- and q^+ such that, for example

$$q^- = q_j + (s/4)[(1-\kappa s)\Delta_- + (1+\kappa s)\Delta_+] \tag{17}$$

where s is the flux limiter. In the present study, the continuously differentiable flux limiter of Ref. 21 was employed, which is defined by

$$s = \frac{2\Delta_- \Delta_+ + \epsilon}{\Delta_-^2 + \Delta_+^2 + \epsilon} \tag{18}$$

where ϵ is a very small number to prevent division by zero in smooth regions of the flow.

Flux-Difference Splitting

The spatial discretization based on flux-difference splitting is a cell-centered scheme where the flow variables are stored at the centroid of each triangle and the control volume is simply the triangle itself. The flux balance becomes

$$\sum T^{-1}H\Delta s = \frac{1}{2}\sum T^{-1}[H(q^-) + H(q^+)]\Delta s$$

$$- \frac{1}{2}\sum T^{-1}|\tilde{A}|(Q^+ - Q^-)\Delta s \tag{19}$$

where the first term on the right-hand side is simply the average of the original flux H evaluated using the flow variables on each side of the edge. The second term on the right-hand side represents a flux difference since it involves the product of the difference in flow variables $(Q^+ - Q^-)$ and the flux Jacobian A, which is defined as

$$A = \frac{\partial H}{\partial Q} \tag{20}$$

In this context the flux Jacobian can be rewritten by a similarity transformation as

$$A = R\Lambda R^{-1} \tag{21}$$

where Λ is a diagonal matrix whose diagonal elements are the characteristic speeds $\bar{U}$, $\bar{U}$, $\bar{U} + a$, and $\bar{U} - a$. The variable $\bar{U}$ is the contravariant velocity normal to the edge being considered and a is the local speed of sound. The notation $|\tilde{A}|$ indicates that the flux Jacobian is evaluated by taking the absolute value of the characteristic speeds and by using so-called Roe-averaged[16] flow variables (indicated by the tilde).

In the FDS scheme, the conserved flow variables on either side of a given edge, Q^+ and Q^-, are determined by first calculating the upwind-biased primitive flow variables q^- and q^+ and then converting them from primitive to conserved. The interpolation formula for q^-, for example, is identical to Eq. (17) including the flux limiter defined by Eq. (18). Therefore, the same parameter κ in Eq. (17) controls a family of difference schemes, ranging from fully upwind to central differencing, by appropriately weighting Δ_- and Δ_+ as in the FVS scheme. The calculations presented herein used $\kappa = \frac{1}{3}$.

Implicit Temporal Discretization

The implicit relaxation algorithm is formulated by first approximating the time derivative in the Euler equations by

$$\frac{\partial Q}{\partial t} \approx \frac{2+\phi}{2}\frac{\Delta Q}{\partial t} + \frac{2+\phi}{2}\frac{Q^* - Q^n}{\Delta t} - \frac{\phi}{2}\frac{Q^n - Q^{n-1}}{\Delta t} \tag{22}$$

where $\Delta Q = Q^{n+1} - Q^*$ and where the parameter ϕ controls the temporal order of accuracy. For example, the scheme is first-order-accurate in time if $\phi = 0$ and the scheme is second-order-accurate in time if $\phi = 1$. For an implicit temporal discretization, the flux H must be treated at time level $(n+1)$, which is accomplished by linearizing according to

$$H^{n+1} = H^* + \frac{\partial H}{\partial Q}\bigg|_{Q=Q^*}\Delta Q \tag{23}$$

where $\partial H / \partial Q$ is the flux Jacobian A as discussed earlier. Also, Eqs. (22) and (23) involve Q^*, the vector of flow variables at an iterate level (*), which is normally taken to be time level (n). For unsteady applications, however, subtractions may be performed to drive Q^* to Q^{n+1} and thus minimize linearization and relaxation errors.

For FVS the forward and backward fluxes are linearized for a given triangle j as

$$\sum T^{-1} \left[H^+(q^-) + H^-(q^+) \right]^{n+1} \Delta s$$

$$= \sum T^{-1} \left[H^+(q^-) + H^-(q^+) \right]^* \Delta s$$

$$+ \left[\sum T^{-1} A^+ \Delta s \right] \Delta Q_j + \sum_{m=1}^{3} T^{-1} A^- \Delta s \Delta Q_m \qquad (24)$$

In this equation the last summation on the right side involves ΔQ_m, the change in the flow variables in the three triangles adjacent to triangle j. Also, the exact Jacobians A^+ and A^- are determined by differentiation of H^+ and H^- by the conserved variables Q. By combining Eqs. (22) and (24), the Euler equations are discretized as

$$\left[\frac{2+\phi}{2} \frac{\text{area}}{\Delta t} I + \sum T^{-1} A^+ \Delta s \right] \Delta Q_j + \sum_{m=1}^{3} T^{-1} A^- \Delta s \Delta Q_m$$

$$= -\frac{2+\phi}{2} \text{area} \frac{Q^* - Q^n}{\Delta t} + \frac{\phi}{2} \text{area} \frac{Q^n - Q^{n-1}}{\Delta t}$$

$$- \sum T^{-1} \left[H^+(q^-) + H^-(q^+) \right]^* \Delta s \qquad (25)$$

where I is the identity matrix and "area" is the area of triangle j. Direct solution of the system of simultaneous equations, which results from application of Eq. (25) for all triangles in the mesh, requires the inversion of a large matrix with large bandwidth that is computationally expensive. Instead, a Gauss-Seidel relaxation approach is used to solve the equations whereby the summation involving ΔQ_m is moved to the right-hand side of Eq. (25). The terms in this summation are then evaluated for a given time step using the most recently computed values for ΔQ_m. The solution procedure then involves only the inversion of a 4×4 matrix [represented by the terms in square brackets on the left-hand side of Eq. (25)] for each triangle in the mesh. Also, although the procedure is implemented for application on (randomly ordered) unstructured meshes, it is not a point Gauss-Seidel procedure. The method is in fact more like line Gauss-Seidel since the list of triangles that make up the unstructured mesh is re-ordered from upstream to downstream, and the solution is obtained by sweeping two times through the mesh as dictated by stability considerations. The first sweep is performed in the direction from upstream to downstream and the second sweep is from downstream to upstream. For purely supersonic flows the second sweep is unnecessary.

For FDS the exact Jacobian A is too expensive to compute and thus an approximate Jacobian is normally used. There are several ways to accomplish this, two of which are described as follows: The first way is to simply use the forward and backward Jacobians from the FVS scheme as in Eq. (25), except of course, the residual is computed using FDS. The second way is to construct approximate Jacobians using Eq. (21) and the fact that the forward and backward Jacobians should have nonnegative and nonpositive eigenvalues (characteristic speeds), respectively. This is done to produce a diagonally dominant system of equations for numerical stability and is accomplished by defining

$$A^+ = R \Lambda^+ R^{-1} \qquad A^- = R \Lambda^- R^{-1} \qquad (26)$$

where

$$\Lambda^+ = (\Lambda + |\Lambda|)/2 \qquad \Lambda^- = (\Lambda - |\Lambda|)/2 \qquad (27)$$

The results presented herein were obtained using the first of these two approaches.

Boundary Conditions

To impose the flow tangency boundary condition along the surface of the airfoil, the flow variables are set within dummy cells that are effectively inside the geometry being considered. The velocity components within a dummy cell are determined from the values in the cell that is adjacent to the surface by requiring that the normal velocity component be zero. Also, pressure and density within the dummy cell are set equal to the values in the adjacent cell. Alternatively, the pressure can be extrapolated from the interior and the density determined by enforcing a zero normal entropy derivative across the surface, but this had no appreciable effect on the solution.

In the far field, a characteristic analysis based on Riemann invariants is used to determine the values of the flow variables on the outer boundary of the grid. This analysis correctly accounts for wave propagation in the far field, which is important for rapid convergence to steady state and serves as a "nonreflecting" boundary condition for unsteady applications.

Results and Discussion

To assess the new Euler solvers, calculations were performed for the NACA 0012 airfoil. These results were obtained using the unstructured grid shown in Fig. 2, which was generated using the advancing front method.[6,19] The grid has 3300 nodes, 6466 triangles, and extends 20 chordlengths from the airfoil with a circular outer boundary. Also there are 110 points that lie on the airfoil surface and the mesh was used for both steady and unsteady applications. This is the same mesh that was used to obtain the results that were presented in Ref. 11.

Steady Flow Results

Steady-state calculations were performed for the NACA 0012 airfoil at a freestream Mach number of $M_\infty = 0.8$ and an angle of attack of $\alpha_0 = 1.25$ deg. Results were obtained using both the implicit time-marching of the present study and the explicit four-stage Runge-Kutta time-marching of Ref. 11. Results are presented first using flux-vector splitting and then using flux-difference splitting for the spatial discretization. The explicit time-marching results were obtained using a CFL number of 2.5 (since the CFL limit is approximately 2.8) and the implicit time-marching results were obtained using a CFL number of 100,000. Such a large value was used for the implicit results since the relaxation scheme has maximum damping and hence fastest convergence for very large time steps. This is in contrast with implicit approximate factorization schemes that have maximum damping for CFL numbers on the order of 10.
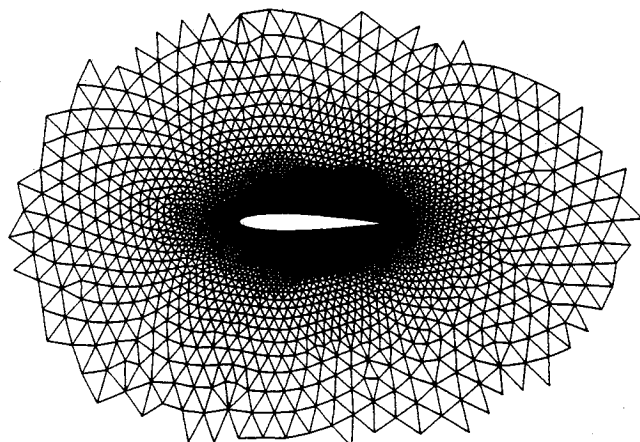

Fig. 2   Partial view of unstructured grid of triangles about the NACA 0012 airfoil.

a) Convergence histories

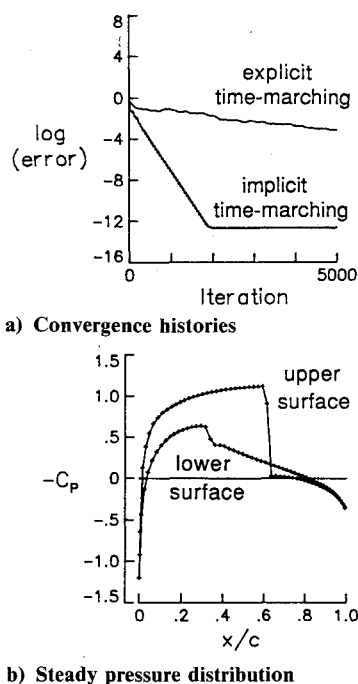b) Steady pressure distribution

Fig. 3   Comparison of steady-state results for the NACA 0012 airfoil at $M_\infty = 0.8$ and $\alpha_0 = 1.25$ deg computed using flux-vector splitting.

a) Convergence histories
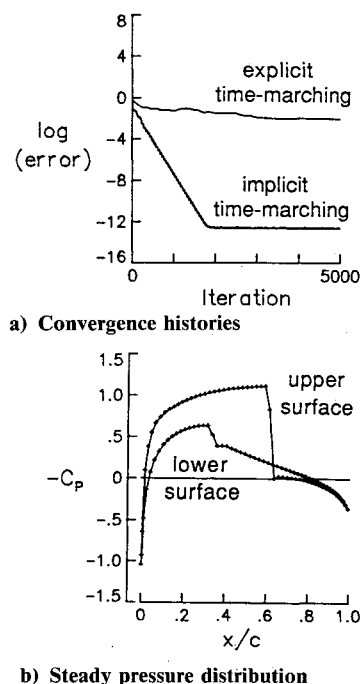

b) Steady pressure distribution

Fig. 4   Comparison of steady-state results for the NACA 0012 airfoil at $M_\infty = 0.8$ and $\alpha_0 = 1.25$ deg computed using flux-difference splitting.

A comparison of the convergence histories between explicit and implicit time-marching for flux-vector splitting is shown in Fig. 3a. The "error" in the solution was taken to be the L_2 norm of the density residual. As shown in Fig. 3a, the explicit solution is very slow to converge. This solution takes approximately 10,000 time steps to become converged to engineering accuracy, which is taken to be a four-order-of-magnitude reduction in solution error. In contrast, the implicit solution is converged to four orders of magnitude in only approximately 500 steps and is converged to machine zero in less than 2000 steps. The implicit solution costs approximately 75% more per time step than the explicit solution because of the increased number of operations required to evaluate the flux Jacobians. This increase in CPU time is far out-weighed by the faster convergence to steady state in that a converged solution is obtained with the implicit relaxation scheme with an order of magnitude less CPU time than the explicit scheme. A converged steady-state solution on the mesh of Fig. 2 can now be obtained with the implicit scheme with less than 10 min CPU time on a Cray-2 supercomputer. The resulting steady pressure distribution is shown in Fig. 3b. For this case there is a relatively strong shock wave on the upper surface of the airfoil near 62% chord and a relatively weak shock wave on the lower surface near 30% chord. The pressure distributions indicate that there is only one grid point within the shock structure, on either the upper or lower surface of the airfoil, due to the sharp shock capturing ability of flux-vector splitting. Furthermore, the steady pressure results of Fig. 3b are of comparable accuracy in comparison with the numerous published results for this case such as those reported in Ref. 20.

A comparison of the convergence histories between explicit and implicit time marching for flux-difference splitting is shown in Fig. 4a. Similar to the solutions obtained using flux-vector splitting, the explicit solution here is very slow to converge. However, the implicit solution is again converged to four orders of magnitude in only approximately 500 steps and is converged to machine zero in less than 2000 steps. These solutions, with either implicit or explicit time-marching, cost approximately the same as the corresponding solutions involving flux-vector splitting. The resulting steady pressure distribution is shown in Fig. 4b. The pressure distribution again indicates that there is only one grid point within the shock structures due to the flux-difference splitting, and the shocks

appear to be slightly more sharply captured in comparison with the shocks from the solution obtained using flux-vector splitting. This is because the FDS scheme has less dissipation than the FVS scheme.

Unsteady Flow Results

Unsteady calculations were performed for the NACA 0012 airfoil pitching harmonically about the quarter chord with an amplitude of $\alpha_1 = 2.51$ deg and a reduced frequency based on semichord of $k = 0.0814$ at $M_\infty = 0.755$ and $\alpha_0 = 0.016$ deg. In these calculations the mesh was rotated rigidly about the quarter chord to continuously conform to the pitching airfoil. The results were obtained initially without performing subiterations and using 250, 1000, and 2500 steps per cycle of motion with the implicit time-marching and flux-vector splitting to determine the appropriate step size to ensure temporal accuracy for this case. Three cycles of motion were computed to obtain periodic solutions. The effects of step size on the instantaneous pressure distribution at $k\tau = 69$ deg in the third cycle, which corresponds to an instantaneous pitch angle of $\alpha(\tau) = 2.34$ deg, are shown in Fig. 5. This angle was selected for this assessment since it lies in the most sensitive part of the cycle. The results of Fig. 5 clearly indicate that with as few as 250 steps per cycle the upper and lower surface shocks both have inaccurate strength and location in comparison with the results obtained using 2500 steps per cycle. The results of Fig. 5 also indicate that the appropriate step size for this case is a time step corresponding to between 1000 and 2500 steps per cycle of motion. This finding is consistent with the temporal refinement study of Ref. 22, where similar results were obtained using implicit approximate factorization solutions of the transonic small-disturbance and Euler equations. It is noted, however, that for easier cases involving higher reduced frequencies and smaller amplitudes of motion, as few as two or three hundred steps per cycle of motion are sufficient for temporal accuracy.

The effects of performing subiterations per time step on the instantaneous pressure distributions at $k\tau = 69$ deg in the cycle corresponding to an (instantaneous pitch angle of $\alpha(\tau) = 2.34$ deg are shown in Fig. 6. The calculations were performed using 250 steps per cycle of motion with the implicit time-marching and flux-vector splitting and parallel results were obtained

using 0, 5, and 10 subiterations per time step. As discussed previously, the purpose of the subiterations is to minimize linearization and relaxation errors, similar to that which is done with approximate factorization schemes to minimize linearization and factorization errors.[22] As shown in Fig. 6, as the number of subiterations is increased the inaccuracies in shock strength and location are decreased. With 10 subiterations, for example, the instantaneous pressure distributions resemble closely those of Fig. 5, obtained using 2500 steps per cycle of

motion and no subiterations. Therefore, there is a compromise between running large time steps with subiterations and running smaller time steps with no subiterations, since the CPU time is approximately the same.

Instantaneous pressure distributions at eight points in time during the third cycle of motion from the 2500 steps per cycle solution using flux-vector splitting are shown in Fig. 7 for comparison with the experimental data of Ref. 21. In each pressure plot the instantaneous pitch angle $\alpha(\tau)$ and the angu-
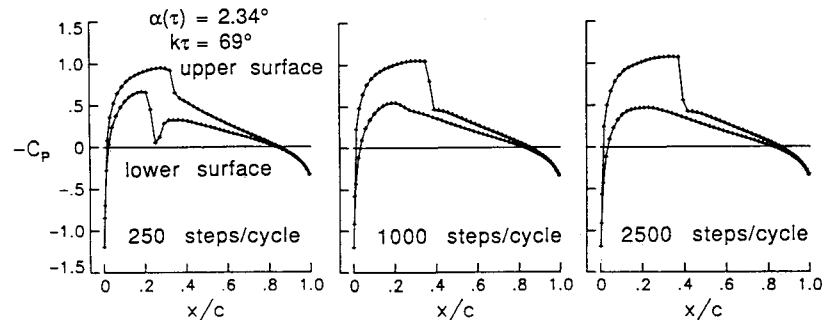

Fig. 5   Effects of step size on the instantaneous pressure distribution at $k\tau = 69$ deg and $\alpha(\tau) = 2.34$ deg during the third cycle of motion for the NACA 0012 airfoil pitching at $M_\infty = 0.755$, $\alpha_0 = 0.016$ deg, $\alpha_1 = 2.51$ deg, and $k = 0.0814$ computed using flux-vector splitting.
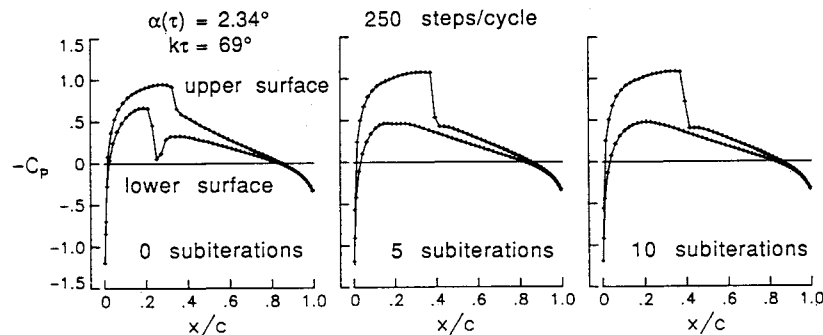

Fig. 6   Effects of performing subiterations per time step on the instantaneous pressure distribution at $k\tau = 69$ deg and $\alpha(\tau) = 2.34$ deg during the third cycle of motion for the NACA 0012 airfoil pitching at $M_\infty = 0.755$, $\alpha_0 = 0.016$ deg, $\alpha_1 = 2.51$ deg, and $k = 0.0814$ computed using flux-vector splitting.
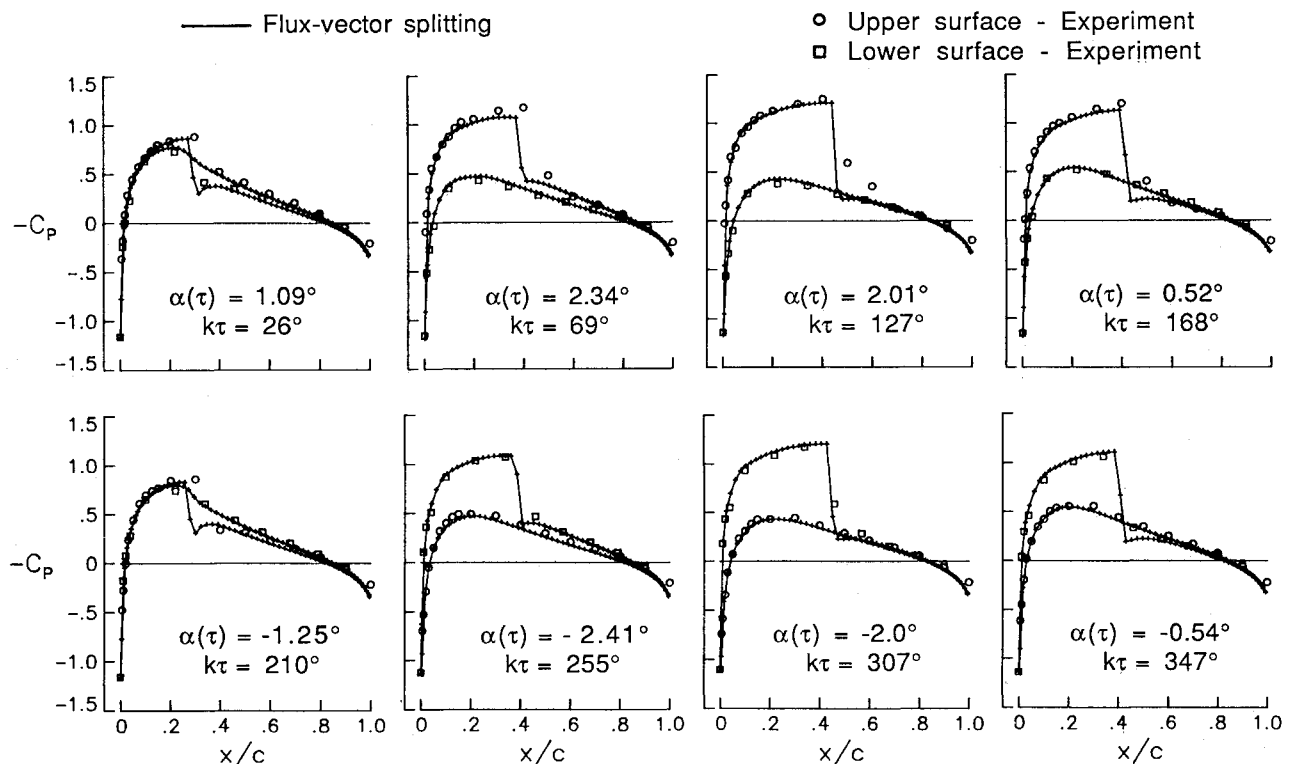

Fig. 7   Comparison of instantaneous pressure distributions for the NACA 0012 airfoil pitching at $M_\infty = 0.755$, $\alpha_0 = 0.016$ deg, $\alpha_1 = 2.51$ deg, and $k = 0.0814$ computed using flux-difference splitting.

—— Flux-difference splitting   ○ Upper surface - Experiment
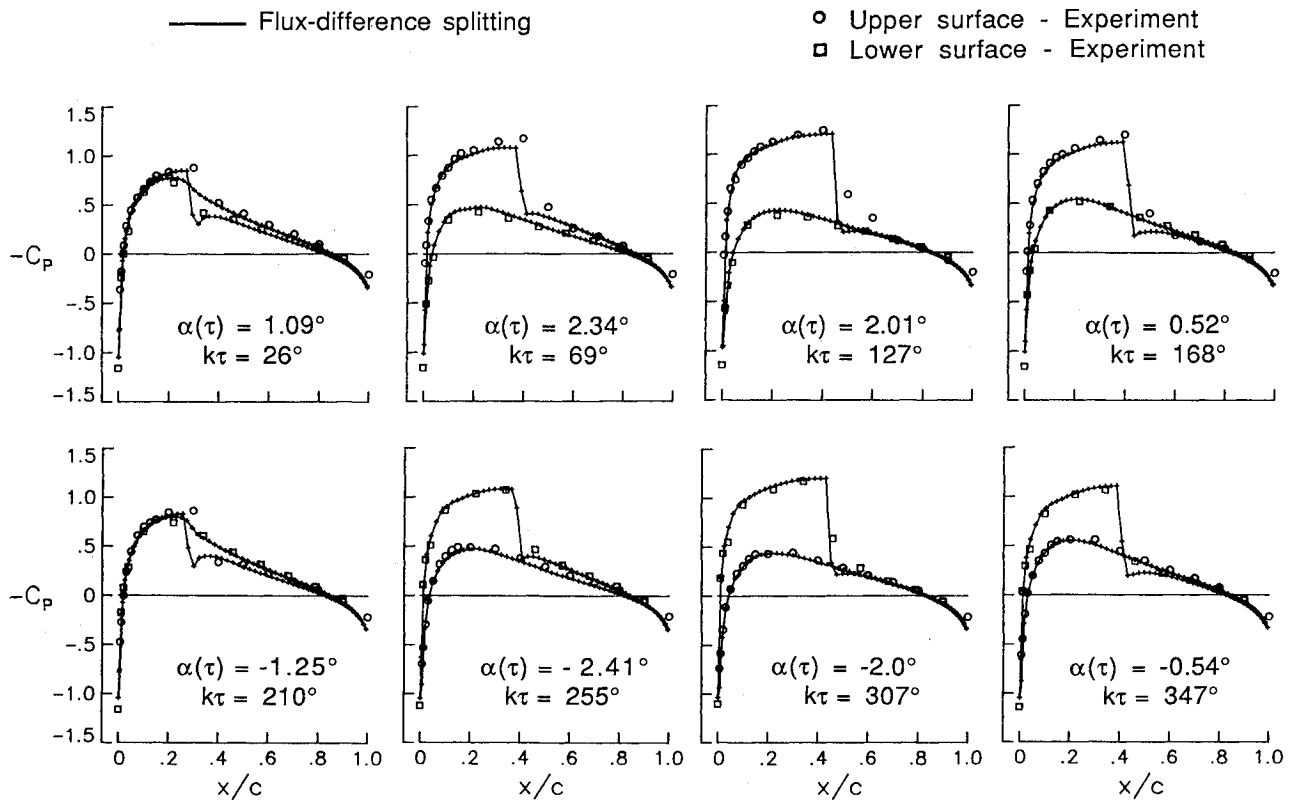   □ Lower surface - Experiment

Fig. 8 Comparison of instantaneous pressure distributions for the NACA 0012 airfoil pitching at $M_\infty = 0.755$, $\alpha_0 = 0.016$ deg, $\alpha_1 = 2.51$ deg, and $k = 0.0814$ computed using flux-vector splitting.

a) Lift coefficient   b) Moment coefficient

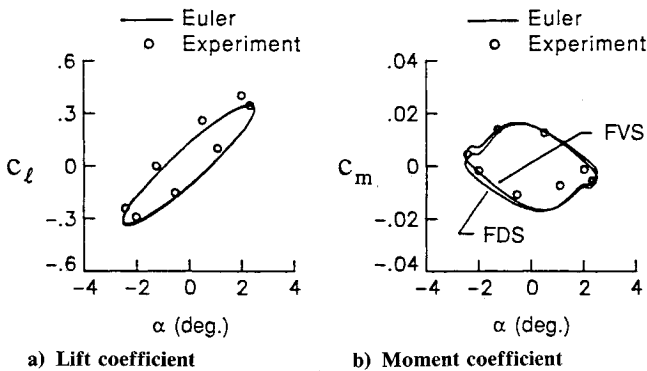Fig. 9 Comparison of coefficient vs instantaneous angle of attack for the NACA 0012 airfoil pitching at $M_\infty = 0.755$, $\alpha_0 = 0.016$ deg, $\alpha_1 = 2.51$ deg, and $k = 0.0814$.

lar position in the cycle $k\tau$ are noted. During the first part of the cycle there is a shock wave on the upper surface of the airfoil, and the flow over the lower surface is predominately subcritical. During the latter part of the cycle the flow about the upper surface is subcritical, and a shock forms along the lower surface. The pressure distributions indicate that the shock position oscillates over approximately 25% of the chord along each surface, and in general, that the two sets of calculated results compare well with the experimental data. Similar to the steady flow results, the shock waves are captured sharply with at most one grid point within the shock structure. The calculated results, however, show the expected symmetry in the flow, in that the upper surface pressure distribution during the first half of the cycle is very similar to the lower surface pressure distribution during the second half of the cycle. The experimental data therefore appears to have been obtained at a slightly higher effective steady-state angle of attack than that reported in Ref. 21. Furthermore, the unstructured grid results of Fig. 7 are of comparable accuracy in comparison with published results obtained using structured grid methods for this case, such as those reported in Ref. 22.

Similar comparisons between calculated and experimental instantaneous pressure distributions at the same eight points in time during the cycle are shown in Fig. 8. The calculated results were obtained using the implicit time-marching with 2500 steps per cycle of motion and the flux-difference splitting for the spatial discretization. The FDS pressures show similar features as the FVS pressures of Fig. 7 in that the shock waves are sharply captured with only one grid point within the shock structure. In general, the FDS pressure distributions also agree well with the experimental data.

Comparisons of calculated and experimental lift and moment coefficients vs the instantaneous angle of attack are presented in Fig. 9. The lift coefficient is shown in Fig. 9a, and the moment coefficient is shown in Fig. 9b. These coefficients show the variation as a function of angle of attack during a cycle of motion, and in general, the two sets of calculated results compare well with the experimental data. The comparisons of lift coefficient further indicate that the data was probably obtained at a higher effective steady-state angle of attack, since the experimental values are higher than the calculated values. Also, the largest difference between FDS and FVS coefficients that occur in the moment coefficient (Fig. 9b) are due to the sensitivity of the moment since the moment center is at the quarter-chord.

Concluding Remarks

Improved algorithms for the solution of the time-dependent Euler equations were presented for unsteady aerodynamic analysis involving unstructured dynamic meshes. The improvements have been developed recently to the spatial and temporal discretizations used by unstructured grid flow solvers. The improved spatial discretization involves a flux-split approach that is naturally dissipative and captures shock waves sharply with at most one grid point within the shock structure. The improved temporal discretization involves an implicit time-integration scheme using a Gauss-Seidel relaxation procedure, which is computationally efficient for either steady or unsteady flow problems. For example, very large time steps may be used for rapid convergence to steady state,

and the step size for unsteady cases may be selected for temporal accuracy rather than for numerical stability.

Steady and unsteady flow results were presented for the NACA 0012 airfoil to demonstrate applications of the new Euler solvers. The steady results showed that rapid convergence to steady state could be achieved with the implicit time-marching in comparison with results obtained using explicit time-marching. A factor of ten reduction in computational cost was obtained for the case that was presented. The unsteady results were obtained for the airfoil pitching harmonically about the quarter chord. The effects of step size and of performing subiterations per time step on the instantaneous pressure distributions during a cycle of pitching motion were demonstrated. These results indicated that the scheme was numerically stable for large time steps although smaller time steps were required to maintain temporal accuracy for the unsteady case that was considered. Also, the calculated instantaneous pressure distributions and lift and moment coefficients during a cycle of motion compared well with experimental data.

Acknowledgments

References

[1]Jameson, A., "Successes and Challenges in Computational Aerodynamics," AIAA Paper 87-1184, Jan. 1987.

[2]Edwards, J. W., and Thomas, J. L., "Computational Methods for Unsteady Transonic Flows," AIAA Paper 87-0107, Jan. 1987; also, *Unsteady Transonic Aerodynamics,* edited by D. Nixon, Vol. 120, Progress in Astronautics and Aeronautics, AIAA, Washington, DC, 1989, Chap. 5.

[3]Jameson, A., and Mavriplis, D. J., "Finite Volume Solution of the Two-Dimensional Euler Equations on a Regular Triangular Mesh," *AIAA Journal,* Vol. 24, No. 4, 1986, pp. 611–618.

[4]Mavriplis, D. J., "Multigrid Solution of the Two-Dimensional Euler Equations on Unstructured Triangular Meshes," *AIAA Journal,* Vol. 26, No. 7, 1988, pp. 824–831.

[5]Jameson, A., Baker, T. J., and Weatherill, N. P., "Calculation of Inviscid Transonic Flow Over a Complete Aircraft," AIAA Paper 86-0103, Jan. 1986.

[6]Morgan, K., and Peraire, J., "Finite Element Methods for Compressible Flow," von Kármán Institute for Fluid Dynamics Lecture Series 1987-04, *Computational Fluid Dynamics,* von Kármán Institute, Belgium, March 2–6, 1987.

[7]Lohner, R., "Finite Elements in CFD: What Lies Ahead," *International Journal of Numerical Methods in Engineering,* Vol. 24, 1987, pp. 1741–1756.

[8]Morgan, K., Peraire, J., Thareja, R. R., and Stewart, J. R., "An Adaptive Finite Element Scheme for the Euler and Navier-Stokes Equations," AIAA Paper 87-1172, June 1987.

[9]Peraire, J., Peiro, J., Formaggia, L., and Morgan, K., "Finite Element Euler Computations in Three Dimensions," AIAA Paper 88-0032, Jan. 1988.

[10]Batina, J. T., "Unsteady Euler Airfoil Solutions Using Unstructured Dynamic Meshes," AIAA Paper 89-0115, Jan. 1989.

[11]Rausch, R. D., Batina, J. T., and Yang, H. T., "Euler Flutter Analysis of Airfoils Using Unstructured Dynamic Meshes," AIAA Paper 89-1384, April 1989.

[12]Batina, J. T., "Unsteady Euler Algorithm with Unstructured Dynamic Mesh for Complex- Aircraft Aeroelastic Analysis," AIAA Paper 89-1189, April 1989.

[13]Barth, T. J., and Jespersen, D. C., "The Design and Application of Upwind Schemes on Unstructured Meshes," AIAA Paper 89-0366, Jan. 1989.

[14]Whitaker, D. L., and Grossman, B., "Two-Dimensional Euler Computations on a Triangular Mesh Using an Upwind, Finite-Volume Scheme," AIAA Paper 89-0470, Jan. 1989.

[15]Van Leer, B., "Flux-Vector Splitting for the Euler Equations," *Lecture Notes in Physics,* Vol. 170, 1982, pp. 507–512.

[16]Roe, P. L., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *Journal of Computational Physics,* Vol. 43, 1981, pp. 357–372.

[17]Thareja, R. R., Stewart, J. R., Hassan, O., Morgan, K., and Peraire, J., "A Point Implicit Unstructured Grid Solver for the Euler and Navier-Stokes Equations," AIAA Paper 88-0036, Jan. 1988.

[18]Struijs, R., Vankeirsbilck, P., and Deconinck, H., "An Adaptive Grid Polygonal Finite Volume Method for the Compressible Flow Equations," AIAA Paper 89-1959, June 1989.

[19]Lohner, R., "Some Useful Data Structures for the Generation of Unstructured Grids," *Communications in Applied Numerical Methods,* Vol. 4, 1988, pp. 123–135.

[20]Anderson, W. K, Thomas, J. L., and Van Leer, B., "Comparison of Finite Volume Flux Vector Splittings for the Euler Equations," *AIAA Journal,* Vol. 24, No. 9, 1986, pp. 1453–1460.

[21]Landon, R. H., "NACA 0012. Oscillating and Transient Pitching," Data Set 3 in AGARD-R-702, Compendium of Unsteady Aerodynamic Measurements, Aug. 1982.

[22]Anderson, W. K., and Batina, J. T., "Accurate Solutions, Parameter Studies, and Comparisons for the Euler and Potential Flow Equations," AGARD CP-437, Paper 14, May 1988.