AIAA JOURNAL
Vol. 32, No. 6, June 1994

Edge-Based Finite Element Scheme for the Euler Equations

Hong Luo* and Joseph D. Baum†
Science Applications International Corporation, McLean, Virgina 22102
and
Rainald Löhner‡
George Washington University, Washington, D.C. 20052

This paper describes the development, validation, and application of a new finite element scheme for the solution of the compressible Euler equations on unstructured grids. The implementation of the numerical scheme is based on an edge-based data structure, as opposed to a more traditional element-based data structure. The use of this edge-based data structure not only improves the efficiency of the algorithm but also enables a straightforward implementation of upwind schemes in the context of finite element methods. The algorithm has been tested and validated on some well-documented configurations. A flow solution about a complete F-18 fighter is shown to demonstrate the accuracy and robustness of the proposed algorithm.

I. Introduction

In recent years, significant progress has been made in the development of numerical algorithms for the solution of the compressible Euler and Navier-Stokes equations. The use of unstructured meshes for computational fluid dynamics problems has become widespread due to their ability to discretize arbitrarily complex geometries and due to the ease of adaptation in enhancing the solution accuracy and efficiency through the use of adaptive refinement techniques. However, any unstructured algorithm requires the storage of the mesh connectivity, which implies the increase of computer memory and the use of indirect addressing to retrieve nearest neighbor information. These requirements, in turn, mean that any numerical algorithm will run slower on an unstructured grid than on a structured grid. To reduce indirect addressing, new edge-based finite element schemes[1–4] have been recently introduced. In addition, even more sophisticated data structures such as stars, superedges, and chains were recently developed by Löhner.[5] The use of an edge-based data structure has been shown to result in remarkable computational savings for three-dimensional problems.

In the last few years, extensive research has been done on upwind-type algorithms for the solution of the Euler and Navier-Stokes equations on unstructured meshes.[6–9] A significant advantage of upwind discretization is that it is naturally dissipative, in contrast with central-difference discretizations, and consequently does not require any problem-dependent parameters to adjust. So far, all upwind schemes implemented as either node-centered or cell-centered discretizations on unstructured meshes have used the finite volume approach where the control volume must be constructed first. In terms of computational efficiency, node-centered schemes are preferable to their cell-centered counterparts. In the node-centered approach,[6,8] the control volume is typically taken to be part of the neighboring cells that have a vertex at that node. In two dimensions, the part of the cells taken is determined by connecting the centroid of the cell and the midpoints of the two edges that share the node. In three-dimensions, the part of the cells taken is determined by a surface constructed in a similar way. However, this is somewhat complicated geometrically to do in three dimensions. The switching from an element-based to an edge-based data structure renders the implementation of upwind schemes trivial and straightforward in the context of the finite element approach; this is especially attractive for three-dimensional application, since there is no need to construct control volumes explicitly and geometrically.

The authors have recently developed some high-accuracy schemes for the solution of the Euler and Navier-Stokes equations on unstructured grids by using an edge-based data structure.[1] This paper describes the development, validation, and application of an upwind finite element algorithm to the simulation of three-dimensional compressible flows around complex aerodynamic configurations. In this scheme, the spatial discretization is accomplished by an edge-based finite element formulation using Roe's flux-difference splitting. A MUSCL approach is used to achieve higher order accuracy. Solutions are advanced in time by a multistage Runge-Kutta time-stepping scheme. Convergence is accelerated using local time stepping and implicit residual smoothing. The algorithm has been tested and validated on some well-documented configurations. A solution of the flow around a complete F-18 fighter is presented to demonstrate the accuracy and robustness of the proposed algorithm.

II. Governing Equations

The Euler equations governing unsteady compressible inviscid flows can be expressed in the conservative form as

$$\frac{\partial U}{\partial t} + \frac{\partial F^j}{\partial x_j} = 0 \tag{1}$$

where the summation convention has been employed and

$$U = \begin{pmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ \rho e \end{pmatrix}, \quad F^j = \begin{pmatrix} \rho u_j \\ \rho u_1 u_j + p\delta_{1j} \\ \rho u_2 u_j + p\delta_{2j} \\ \rho u_3 u_j + p\delta_{3j} \\ u_j(\rho e + p) \end{pmatrix} \tag{2}$$

Presented as Paper 93-2933 at the AIAA 24th Fluid Dynamics Conference, Orlando, FL, July 6–9, 1993; received July 14, 1993; revision received Oct. 28, 1993; accepted for publication Nov. 26, 1993.

*Research Scientist, Applied Physics Operations, 1710 Goodridge Drive, MS 2-3-1. Member AIAA.

†Senior Research Scientist, Applied Physics Operations, 1710 Goodridge Drive, MS 2-3-1. Associate Fellow AIAA.

‡Associate Research Professor, CMEE, School of Engineering and Applied Science. Member AIAA.

where ρ, p, and e denote the density, pressure, and specific total energy, respectively, and u_j; $j = 1,2,3$ is the velocity of the flow in the coordinate direction x_j; $j = 1,2,3$ of a cartesian coordinate system. This set of equations is completed by the addition of the equation of state

$$p = (\gamma - 1)\rho\,[e - 1/2\,(u_1^2 + u_2^2 + u_3^2)\,] \tag{3}$$

which is valid for perfect gas, where γ is the ratio of the specific heats.

In the sequel, we assume that Ω is the flow domain, Γ its boundary, and n_j the unit outward normal vector to the boundary.

III. Variational Formulation and Finite Element Approximation

Let $\mathfrak{T}$ be a trial function space and $\mathfrak{W}$ a weighting function space, both defined to consist of all suitably smooth functions. An equivalent variational formulation of Eq. (1) is given by

$$\begin{cases} \text{find} \quad U \in \mathfrak{T} \quad \text{such that} \quad W \in \mathfrak{W} \forall \\ \displaystyle\int_\Omega \frac{\partial U}{\partial t} W \, d\Omega - \int_\Omega F^j \frac{\partial W}{\partial x_j} \, d\Omega + \int_\Gamma F^j n_j W \, d\Gamma = 0 \end{cases} \tag{4}$$

Assuming Ω_h is a classical triangulation of Ω with the nodes numbered from 1 to n and Γ_h the boundary of Ω_h, we approximate the trial and weighting spaces $\mathfrak{T}$ and $\mathfrak{W}$ by their subspaces of finite dimension $\mathfrak{T}_h$ and $\mathfrak{W}_h$, which, respectively, are defined by

$$\begin{aligned} \mathfrak{T}_h &= \left\{ U_h(x,t) \,|\, U_h(x,t) = \sum_{I=1}^{n} U_I(t)\, N_I(x) \right\} \\ \mathfrak{W}_h &= \left\{ W_h(x) \,|\, W_h(x) = \sum_{I=1}^{n} \alpha_I N_I(x) \right\} \end{aligned} \tag{5}$$

where N_I is the standard linear finite element shape function associated with node I, U_I is the value at node I, and α_I is a constant. The Galerkin finite element approximation is then given by

$$\begin{cases} \text{find} \quad U_h \in \mathfrak{T}_h \quad \text{such that for each} \quad N_I \,(1 \le I \le n) \\ \displaystyle\int_{\Omega_h} \frac{\partial U_h}{\partial t} N_I \, d\Omega_h = \int_{\Omega_h} F^j(U_h) \frac{\partial N_I}{\partial x_j} \, d\Omega_h - \int_{\Gamma_h} F^j(U_h)\, n_j N_I \, d\Gamma_h \end{cases} \tag{6}$$

The integrals appearing here are evaluated in the standard finite element form by summing individual element and boundary surface contributions; the compact support of the shape function N_I means that the equation can be written as

$$\sum_{e \in I} \int_{\Omega_e} \frac{\partial U_h}{\partial t} N_I \, d\Omega_h = \sum_{e \in I} \int_{\Omega_e} F^j(U_h) \frac{\partial N_I}{\partial x_j} \, d\Omega_h - \sum_{b \in I} \int_{\Gamma_b} F^j(U_h)\, n_j N_I \, d\Gamma_h \tag{7}$$

where the summation extends over those elements e and boundary surfaces b that contain node I. Inserting the assumed form for U_h in Eq. (7), the left-hand side integral can be evaluated exactly to give

$$\begin{aligned} \sum_{e \in I} \int_{\Omega_e} \frac{\partial U_h}{\partial t} N_I \, d\Omega_h &= \sum_{e \in I} \left(\int_{\Omega_e} N_I N_J \, d\Omega_h \right) \frac{dU_J}{dt} \\ &= M_{IJ} \frac{dU_J}{dt} \end{aligned} \tag{8}$$

where M denotes the finite element consistent mass matrix. For steady-state computations, M can be replaced by the lumped (diagonal) mass matrix, denoted by M_L.

IV. Edge-Based Upwind Finite Element Scheme

It is shown in the Appendix that, for any interior node, Eq. (7) can be written as

$$\left(M_L \frac{dU}{dt} \right)_I = \sum_{IJ}^{m_I} C_{IJ}^j (F_I^j + F_J^j) \tag{9}$$

where m_I is the number of edges connected to the node I, and

$$C_{IJ}^j = -\sum_{e \in IJ} \frac{\Omega_e}{4} \left. \frac{\partial N_I}{\partial x_j} \right|_e \tag{10}$$

in three dimensions. The coefficient C_{IJ} denotes the weight applied to the average value of the flux on the edge that connects nodes I and J, to obtain the contribution made by the edge to node I, whereas C_{JI} denotes the weight applied to the same quantity to obtain the contribution made by the edge to node J. It can be easily verified that these weights possess the properties, for all I,

$$\sum_{IJ}^{m_I} C_{IJ}^j = 0 \tag{11}$$

and for all I and J,

$$C_{IJ}^j = -C_{JI}^j \tag{12}$$

For notational convenience, we define the vector C_{IJ} by the expression

$$C_{IJ} = (C_{IJ}^1, C_{IJ}^2, C_{IJ}^3) \tag{13}$$

and let L_{IJ} denote the modulus and k_{IJ} denote a unit vector in the direction of C_{IJ}; then Eq. (9) can be written as

$$\left(M_L \frac{dU}{dt} \right)_I = \sum_{IJ}^{m_I} L_{IJ} (F_I + F_J) = \sum_{IJ}^{m_I} L_{IJ} F_{IJ} \tag{14}$$

where

$$F_I = (F_I^1, F_I^2, F_I^3) \cdot k_{IJ} \tag{15}$$

$$F_J = (F_J^1, F_J^2, F_J^3) \cdot k_{IJ} \tag{16}$$

The alternative procedure for obtaining the discrete form of the equations is now apparent. Whereas the element-based data structure gathers information from all of the nodes of each element, operates on the element, and then scatters it back to the nodes of the element, the edge-based algorithm gathers information from all of the nodes of each edge, operates it on the edge, and then scatters it back to the nodes of the edge. The property of conservation in the numerical scheme is guaranteed by the asymmetry of edge coefficients as expressed in Eq. (11). This edge-based data structure not only improves the efficiency of the algorithm,[1] but also enables a straightforward implementation of upwind schemes in the context of finite element methods. It is clear that Eq. (14) is nothing but a classic Galerkin finite element scheme, which is equivalent to a central-difference-type scheme. By using the results of Eq. (12), this scheme allows for the appearance of checkerboarding modes and thus suffers from numerical instabilities unless some type of

numerical dissipation in the form of artificial viscosity is introduced. A stable scheme can be constructed, for example, using Roe's flux difference splitting,[10] to replace the actual flux function F_{IJ} in Eq. (14) by Roe's numerical flux formula $\mathfrak{F}_{IJ}$:

$$\mathfrak{F}_{IJ} = F_I + F_J - |A_{IJ}|\Delta U \tag{17}$$

where $\Delta U = U_J - U_I$ and

$$|A_{IJ}|\Delta U = |\Delta \bar{F}_1| + |\Delta \bar{F}_4| + |\Delta \bar{F}_5| \tag{18}$$

with

$$|\Delta \bar{F}_1| = |\bar{\lambda}_1| \left[\left(\Delta \rho - \frac{\Delta p}{\bar{c}^2} \right) \begin{pmatrix} 1 \\ \bar{u}_1 \\ \bar{u}_2 \\ \bar{u}_3 \\ \bar{q}^2/2 \end{pmatrix} + \bar{\rho} \begin{pmatrix} 0 \\ \Delta u_1 - k_1 \Delta q_k \\ \Delta u_2 - k_2 \Delta q_k \\ \Delta u_3 - k_3 \Delta q_k \\ \bar{u}_1 \Delta u_1 + \bar{u}_2 \Delta u_2 + \bar{u}_3 \Delta u_3 - \bar{q}_k \Delta q_k \end{pmatrix} \right] \tag{19}$$

$$|\Delta \bar{F}_{4,5}| = |\bar{\lambda}_{4,5}| \left(\frac{\Delta p \pm \bar{\rho} \bar{c} \Delta q_k}{2\bar{c}^2} \right) \begin{pmatrix} 1 \\ \bar{u}_1 \pm k_1 \bar{c} \\ \bar{u}_2 \pm k_2 \bar{c} \\ \bar{u}_3 \pm k_3 \bar{c} \\ \bar{h} \pm \bar{q}_k \bar{c} \end{pmatrix} \tag{20}$$

where $\bar{q}^2 = \bar{u}_1^2 + \bar{u}_2^2 + \bar{u}_3^2$, $\Delta q_k = \Delta u_1 k_1 + \Delta u_2 k_2 + \Delta u_3 k_3$, and $\bar{q}_k = \bar{u}_1 k_1 + \bar{u}_2 k_2 + \bar{u}_3 k_3$. The bar designates Roe-averaged quantities, which are defined by

$$\bar{\rho} = \sqrt{\rho_I \rho_J}$$

$$\bar{h} = (h_I + h_J \sqrt{\rho_J/\rho_I}) / (1 + \sqrt{\rho_J/\rho_I})$$

with $\bar{u}_1$, $\bar{u}_2$, and $\bar{u}_3$ defined in the same way as $\bar{h}$ and

$$\bar{c} = \sqrt{(\gamma - 1)(\bar{h} - 0.5 * \bar{q}^2)}$$

Furthermore, the eigenvalues of A are $\bar{\lambda}_1 = \bar{q}_k$ and $\bar{\lambda}_{4,5} = \bar{q}_k \pm \bar{c}$.

To prevent entropy violation, an entropy fix is imposed. When an eigenvalue $\bar{\lambda}$ reduces to zero, a smoothed value $|\bar{\lambda}|^*$ is defined to replace $|\bar{\lambda}|$.

$$|\bar{\lambda}|^* = \begin{cases} |\bar{\lambda}|, & \text{if } |\bar{\lambda}| \geq \varepsilon \\ \dfrac{|\bar{\lambda}|^2 + \varepsilon^2}{2\varepsilon}, & \text{if } |\bar{\lambda}| \leq \varepsilon \end{cases} \tag{21}$$

where $\varepsilon = K \max(\lambda_J - \lambda_I, 0)$. K is a small constant.

It can be shown that this scheme is equivalent to the first-order finite volume upwind cell-vertex scheme based on a dual mesh. There are many different ways to achieve higher order accuracy. In the present study, a scheme of higher order accuracy is achieved by using upwind-biased interpolations of the solution U via the MUSCL approach.[11] This leads to the flux function

$$\mathfrak{F}_{IJ} = F_I^+ + F_J^- - |A(U_I^+, U_J^-)|(U_J^- - U_I^+) \tag{22}$$

where

$$F_I^+ = F(U_I^+), \quad F_J^- = F(U_J^-) \tag{23}$$

The upwind-biased interpolations for U_I^+ and U_J^- are defined by

$$U_I^+ = U_I + (1/4)\left[(1-k)\Delta_I^- + (1+k)(U_J - U_I)\right] \tag{24}$$

$$U_J^- = U_J - (1/4)\left[(1-k)\Delta_J^+ + (1+k)(U_J - U_I)\right] \tag{25}$$

where the forward and backward difference operators are given by

$$\Delta_I^- = U_I - U_{I-1} = 2(\nabla U)_I \cdot \boldsymbol{l}^{IJ} - (U_J - U_I) \tag{26}$$

$$\Delta_J^+ = U_{J+1} - U_J = 2(\nabla U)_J \cdot \boldsymbol{l}^{IJ} - (U_J - U_I) \tag{27}$$

where $\boldsymbol{l}^{IJ} = \boldsymbol{x}_J - \boldsymbol{x}_I$ is the length vector of this edge.

The parameter k can be chosen to control a family of difference schemes in the interpolation. On structured meshes it is easy to show that $k = -1$ yields a fully upwind scheme, $k = 0$ yields a semi-upwind approximation (Fromm's scheme), and $k = 1$ yields central differencing. The value $k = 1/3$ leads to a third-order-accurate upwind-biased scheme, although third-order accuracy is strictly correct only for one-dimensional linear problems. Nevertheless, $k = 1/3$ was used in the calculations presented herein. With higher order spatial accuracy, spurious oscillations in the vicinity of shock waves are expected to occur. Some form of limiting is usually required to eliminate these numerical oscillations of the solution and to provide some kind of monotonicity property. The flux limiter modifies the upwind-biased interpolation U_I and U_J and the Eqs. (24) and (25) are replaced, respectively, by

$$U_I^+ = U_I + (s_I/4)\left[(1-ks_I)\Delta_I^- + (1+ks_I)(U_J - U_I)\right] \tag{28}$$

$$U_J^- = U_J - (s_J/4)\left[(1-ks_J)\Delta_J^+ + (1+ks_J)(U_J - U_I)\right] \tag{29}$$

where s is the flux limiter. The van Albada limiter employed in this study acts in a continuously differentiable manner and is defined by

$$s_I = \max\left[0, \frac{2\Delta_I^+(U_J - U_I) + \varepsilon}{(\Delta_I^+)^2 + (U_J - U_I)^2 + \varepsilon}\right] \tag{30}$$

$$s_J = \max\left[0, \frac{2\Delta_J^+(U_J - U_I) + \varepsilon}{(\Delta_J^+)^2 + (U_J - U_I)^2 + \varepsilon}\right] \tag{31}$$

where ε is a very small number to prevent division by zero in smooth regions of the flow. Three options exist concerning the choice of interpolation variables: conservative variables, primitive variables, and characteristic variables. Using limiters on characteristic variables seems to give the best results. However, the primitive variables are used in this study for the sake of computational efficiency.

The treatment of boundary conditions is very important for rapid convergence to steady state and serves as nonreflecting boundary conditions for unsteady computations. On the solid walls, the slip condition is assumed and the normal velocity vanishes. In the farfield, a characteristic analysis based on the one-dimensional Riemann invariants is used to determine the values of the flow variables.

V. Temporal Discretization

Equation (14) represents the time evaluation of the unknown vector $U_I(t)$ at node I of the grid. Assuming that the nodal values U_I^n are known at time t_n, the solution is advanced over a time step Δt, to time t_{n+1} by an explicit multistage Runge-Kutta time-stepping scheme given by

$$U_I^{(0)} = U_I^n$$

$$\vdots$$

$$U_I^{(p)} = U_I^{(0)} - \alpha_p \Delta t \, (M_L)_I^{-1} R_I\left[U_I^{(p-1)}\right] \qquad p = 1, 2, \ldots, m$$

$$\vdots$$

$$U_I^{n+1} = U_I^{(m)}$$

with the parameters α_p assigned appropriate values. The scheme is second order accurate in time. For steady-state computations, implicit residual smoothing and local time stepping are used to accelerate convergence to steady state. The residual smoothing allows the use of larger Courant-Friedrichs-Lewy (CFL) numbers than the one dictated by the stability of the original scheme. This is accomplished by averaging implicitly the residual with values at neighboring grid points. These implicit equations are solved approximately by using several Jacobi iterations. The local time stepping uses separately a maximum allowable step size for each node according to the local stability analysis.

VI. Numerical Results

All of the grids used here were generated by the advancing front technique.[12] All computations used a three-stage Runge-Kutta time-stepping scheme with local time stepping and implicit residual smoothing. The computations were started with uniform flow and advanced with a CFL number of 4. The L_2 norm of density residual is taken as a criterion to test the convergence history.

A. Channel with a Circular Bump on the Lower Wall

The first test case is the well-known Ni's test case. It is a transonic flow in a channel with a 10% thick circular bump on the bottom. The length of the channel is 3, its height 1, and its width 0.5. The inlet Mach number is 0.675. This is a three-dimensional simulation of a two-dimensional flow. This simple test case is chosen to assess both accuracy and convergence of the numerical scheme and to validate the implementation of the code. The mesh, which contains 13,891 grid points, 68,097 elements, and 4442 boundary points, is depicted in Fig. 1b. The convergence history is shown in Fig. 1a, where a monotone convergence to computer machine zero is observed. Figure 1c displays the computed pressure contours in the flowfield. The Mach number distribution on the lower wall, shown in Fig. 1d, indicates that there is only one grid point within the shock structure; this demonstrates the sharp shock-capturing ability of Roe's approximate Riemann solver for the solution of steady problems.

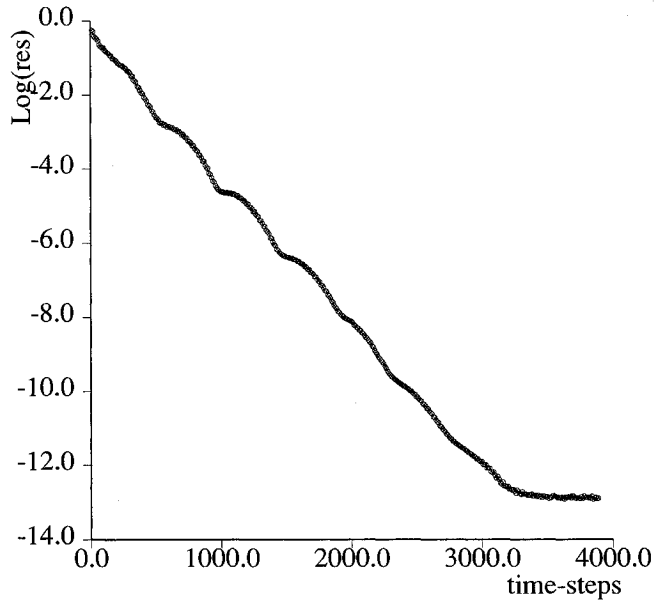

Fig. 1a Convergence history.

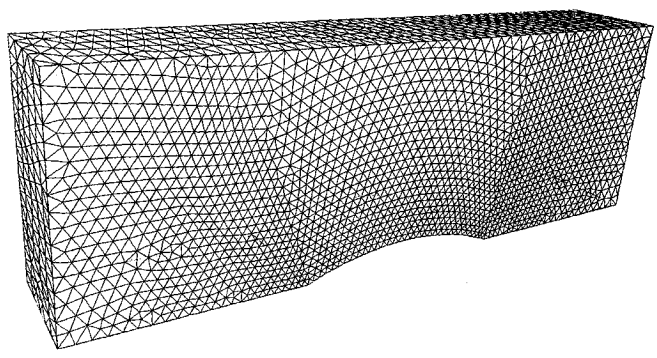

Fig. 1b Surface mesh for the channel.

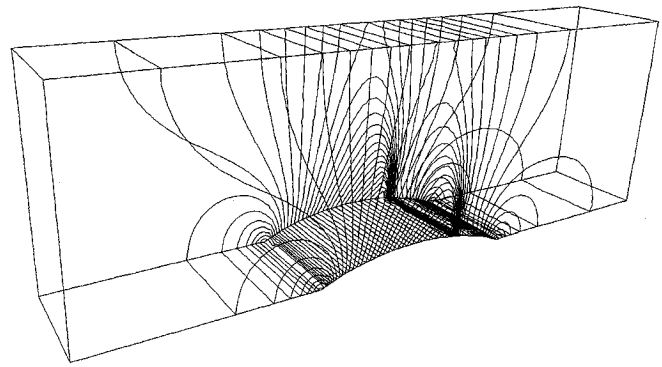

Fig. 1c Computed pressure contours.

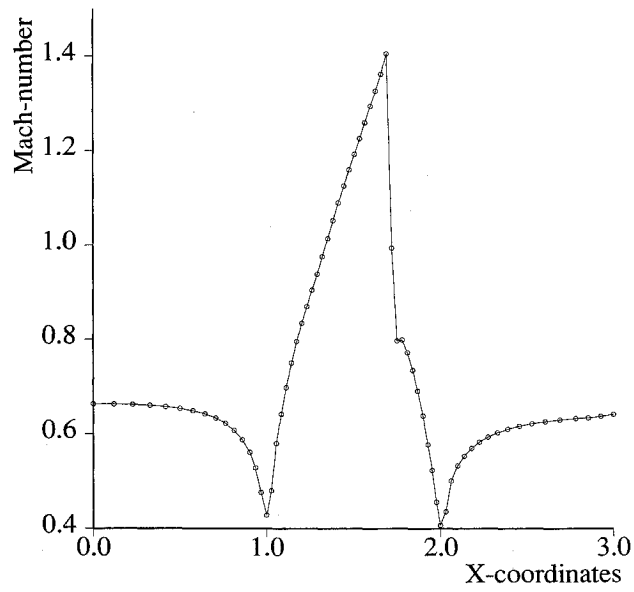

Fig. 1d Mach number distribution.

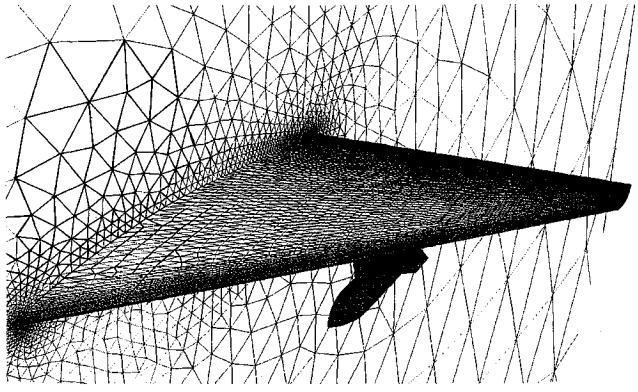

Fig. 2a Surface mesh for the wing/pylon/store.

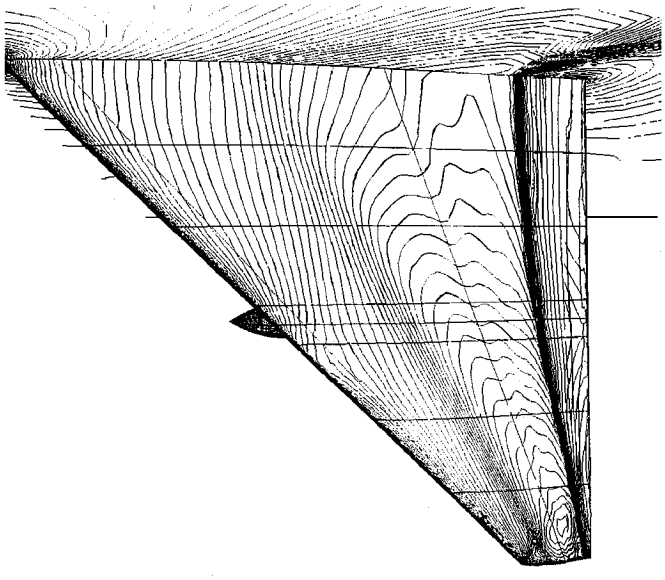

Fig. 2b Computed pressure contours on the upper surface at M_∞ = 0.95 and α = 0 deg.

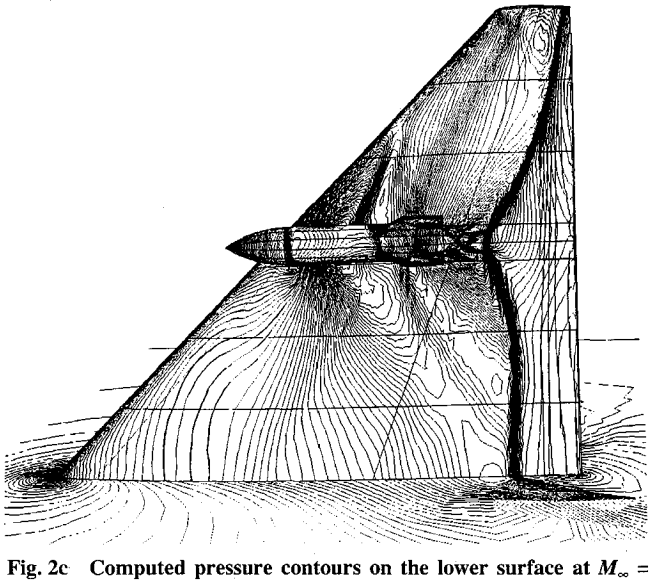

Fig. 2c Computed pressure contours on the lower surface at M_∞ = 0.95 and α = 0 deg.

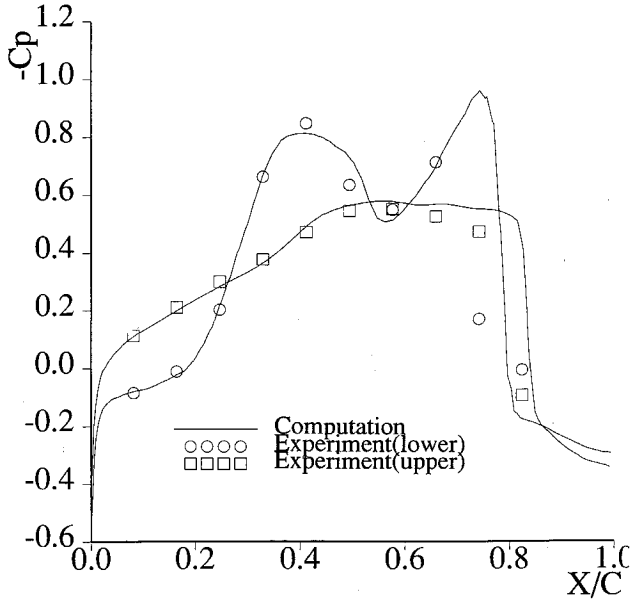

Fig. 2d Comparisons between computed and experimental surface pressure coefficient for wing section at 36% semispan.

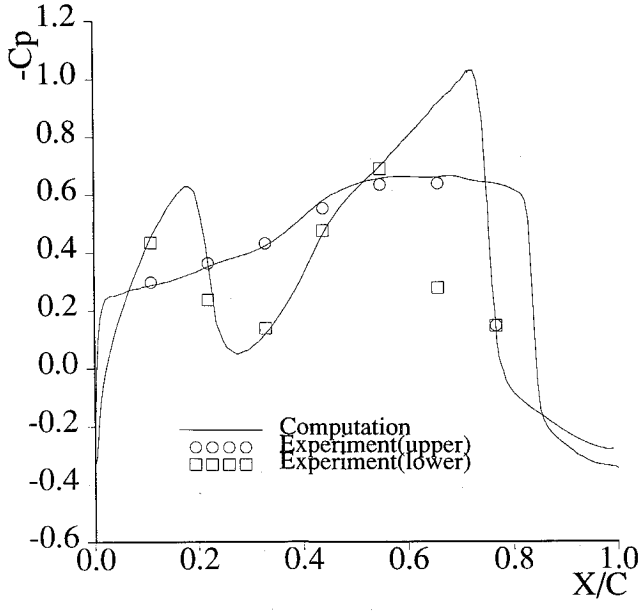

Fig. 2e Comparisons between computed and experimental surface pressure coefficient for wing section at 59% semispan.

B. Wing/Pylon/Finned-Store Configuration

The second test case is conducted for a wing/pylon/finned-store configuration reported in Ref. 13. The configuration consists of a clipped delta wing with a 45-deg sweep comprised from a constant NACA 64010 symmetric airfoil section. The wing has a root chord of 15 in., a semispan of 13 in., and a taper ratio of 0.134. The pylon is located at the midspan station and has a cross-section characterized by a flat plate closed at the leading and trailing edges by a symmetrical ogive shape. The width of the pylon is 0.294 in. The four fins on the store are defined by a constant NACA 0008 airfoil section with a leading-edge sweep of 45 deg and a truncated tip. The mesh used in the computation is shown in Fig. 2a. It contains 274,953 grid points, 1,518,770 elements, and 33,046 boundary points. The flow solutions are presented at a Mach number of 0.95 and an angle of attack of 0 deg. Figures 2b and 2c show the pressure contours on the upper and lower wing surface, respectively. The computed pressure coefficient distributions are compared with experimental data at two spanwise stations in Figs. 2d and 2e. The comparison with experimental data is excellent on both the upper and lower surface up to 70% chord. As expected from the Euler solution, the computation predicts a shock location that is downstream of that measured by the experiment due to the lack of viscous effect.

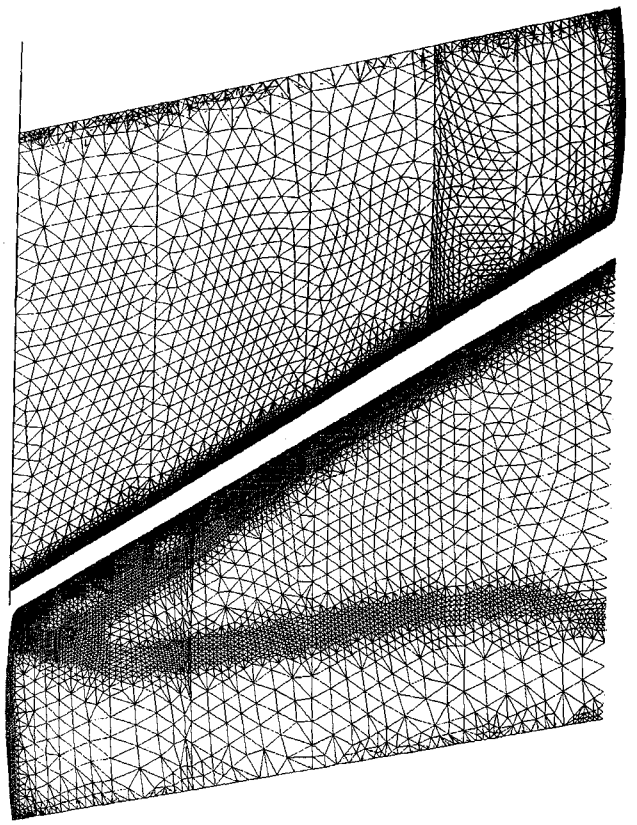

Fig. 3a Upper and lower surface meshes for M6 wing.

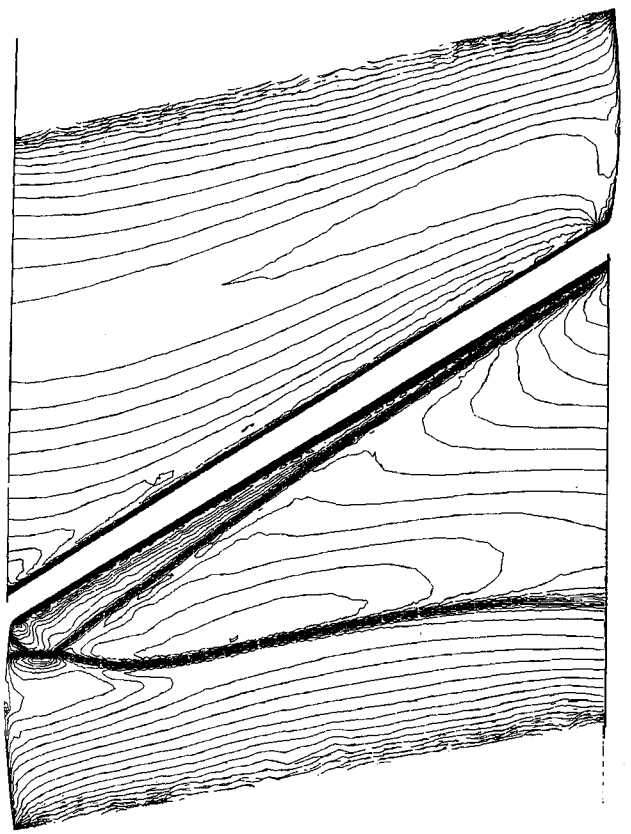

Fig. 3b Computed pressure contours on the upper and lower surface at $M_\infty = 0.84$ and $\alpha = 3.06$ deg.

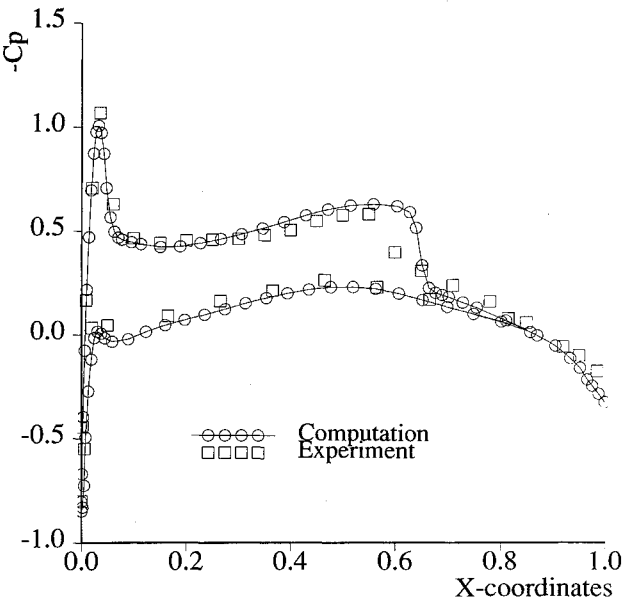

Fig. 3c Comparison between computed and experimental surface pressure coefficient for wing section at 20% semispan.

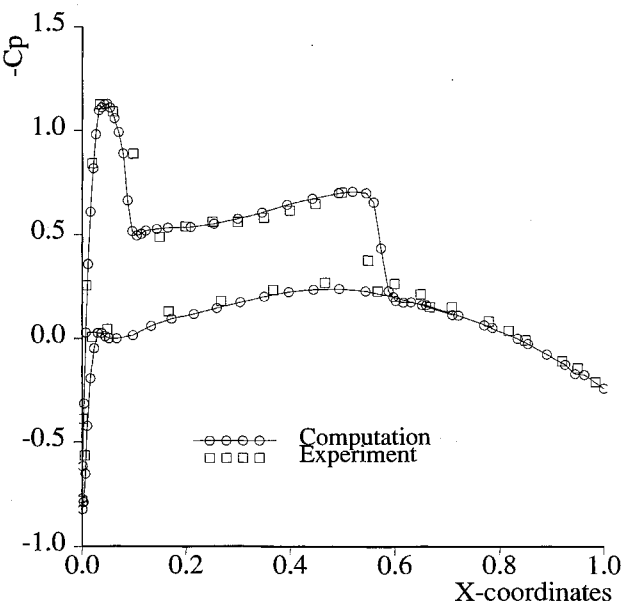

Fig. 3d Comparison between computed and experimental surface pressure coefficient for wing section at 44% semispan.

C. ONERA M6 Wing Configuration

The third, well-documented case is the transonic flow over the ONERA M6 wing configuration. The M6 wing has a leading-edge sweep angle of 30 deg, an aspect of 3.8, and a taper ratio of 0.562. The airfoil section of the wing is the ONERA "D" airfoil, which is a 10% maximum thickness-to-chord ratio conventional section. The flow solutions are presented at a Mach number of 0.84 and an angle of attack of 3.06. The grid adaption scheme was used for this test case.

The final adapted mesh contains 133,206 grid points, 738,669 elements, and 17,155 boundary points after two levels of refinement. The refinement of high gradient regions such as shocks, leading edge, and wing tip is well captured. The final adapted upper and lower surface meshes are shown in Fig. 3a. The pressure contours on the upper wing surface and lower surface are displayed in Fig. 3b. The upper surface contours clearly show the sharply captured lambda-type shock structure formed by the two inboard shock waves, which merge together near 80% semispan to form the single strong shock wave in the outboard region of the wing. The computed pressure coefficient distributions are com-

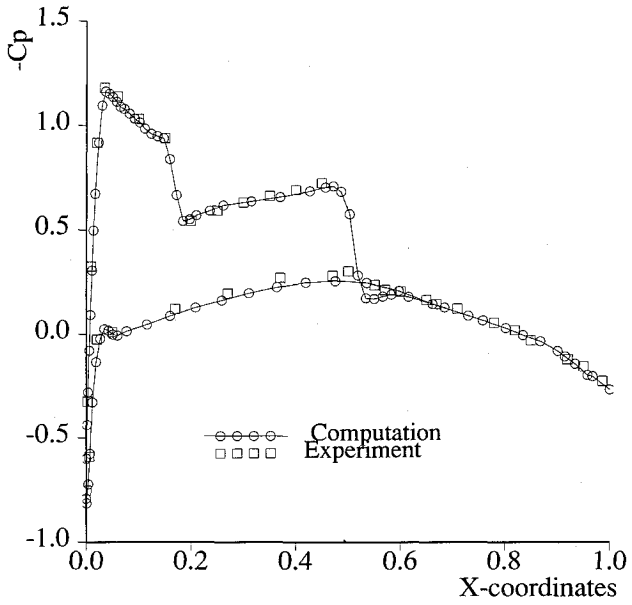

Fig. 3e Comparison between computed and experimental surface pressure coefficient for wing section at 65% semispan.

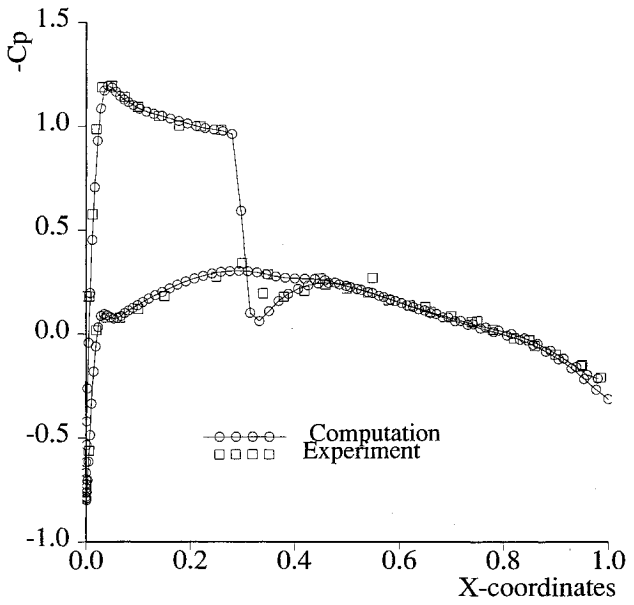

Fig. 3f Comparison between computed and experimental surface pressure coefficient for wing section at 90% semispan.

pared with experimental data[14] at four spanwise stations in Figs. 3c–3f. The results obtained compare closely with experimental data, except at the root stations, due to lack of viscous effects.

D. F-18 Fighter Configuration

The final case is a complete F-18 fighter configuration, which includes the wing, horizontal and vertical tails, and flow-through engine ducts. The mesh, which contains 93,642 grid points, 505,087 elements, and 15,421 boundary points for the half-span airplane, is shown in Fig. 4b. The computations were performed at a freestream Mach number of 0.9 and an angle of attack of 3 deg. The convergence history is depicted in Fig. 4a. The solution is converged to engineering accuracy (a decrease of four orders-of-magnitude in the L_2 norm of the density residual) in 1300 time steps. It required a total of 10 CPU h on a single processor Cray 2. The computed pressure contours on the surface of the airplane are shown in Fig. 3c. Some of the features occurring in this flow regime, such as the canopy and wing shocks, are well captured.

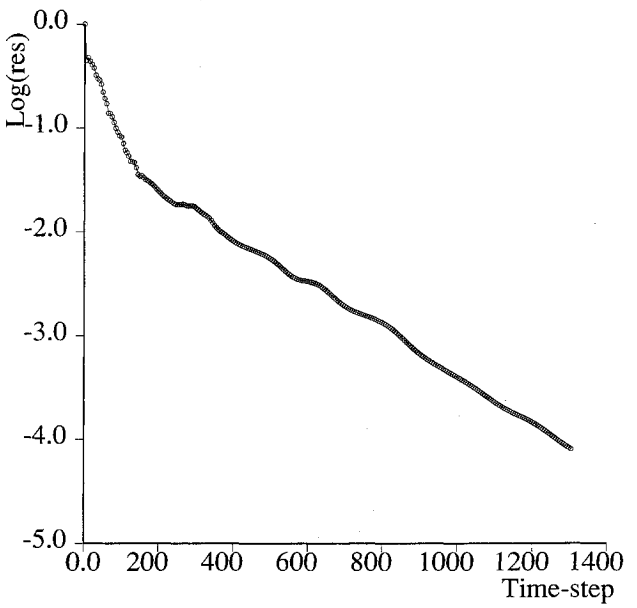

Fig. 4a Convergence history for the F-18 fighter.

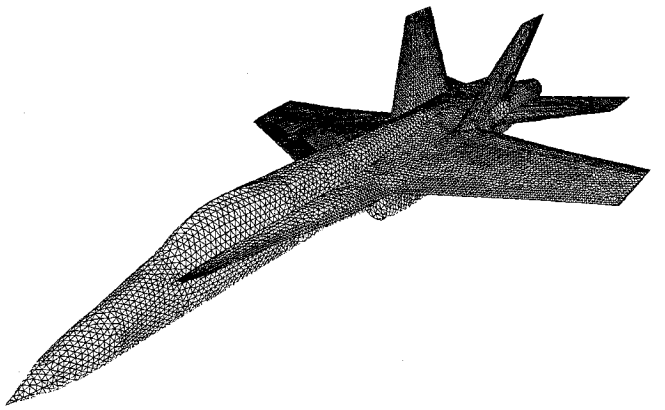

Fig. 4b Surface mesh of triangles for the F-18 fighter.

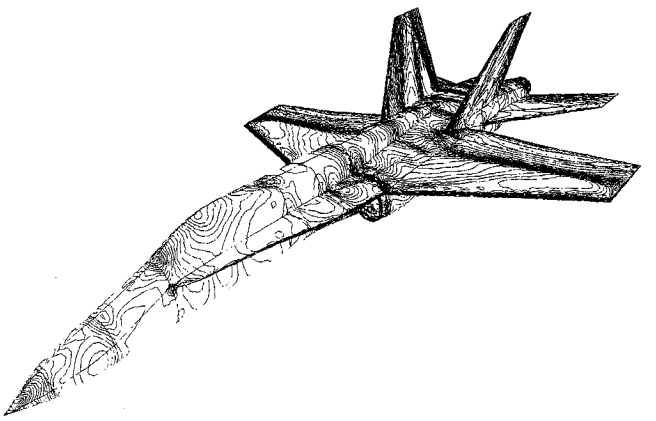

Fig. 4c Computed pressure contours on the F-18 fighter at $M_\infty = 0.9$ and $\alpha = 3$ deg.

VII. Conclusions

An edge-based upwind finite element scheme has been developed for the solutions of the compressible Euler equations on unstructured grids. The numerical scheme has been tested and validated on some well-documented configurations. An example application is presented for a complete F-18 fighter configuration

to demonstrate the accuracy and robustness of the proposed algorithm.

Appendix

In this appendix, the evaluation of inviscid fluxes using an edge-based data structure is derived:

$$RHS(I) = -\int_{\Omega_h} F^j \frac{\partial N_I}{\partial x_j}\, d\Omega_h + \int_{\Gamma_h} F^j n_j N_I \, d\Gamma_h$$

$$- \int_{\Omega_h} (n\text{node} - 2) F_I^J N_I \frac{\partial N_I}{\partial x_j}\, d\Omega_h$$

$$+ \int_{\Omega_h} (n\text{node} - 2) F_I^J N_I \frac{\partial N_I}{\partial x_j}\, d\Omega_h \tag{A1}$$

where $n$node is the number of nodes on an element, and the last two integrals are artificially added to derive the desired formula. By using Green's formula, we obtain

$$\int_{\Omega_h} F_I^j N_I \frac{\partial N_I}{\partial x_j}\, d\Omega_h = \frac{1}{2}\int_{\Gamma_h} F^j n_j N_I N_I \, d\Gamma_h \tag{A2}$$

Using Eq. (A2), Eq (A1) can be written as

$$RHS(I) = -\int_{\Omega_h} \left[F^j + (n\text{node} - 2) F_I^j N_I\right] \frac{\partial N_I}{\partial x_j}\, d\Omega_h$$

$$+ \int_{\Gamma_h} F^j n_j N_I \, d\Gamma_h + \frac{1}{2}\int_{\Gamma_h} (n\text{node} - 2) F_I^j n_j N_I N_I \, d\Gamma_h \tag{A3}$$

For an interior point, the boundary integrals can be dropped and the right-hand side becomes

$$RHS(I) = -\int_{\Omega_h} \left[F^j + (n\text{node} - 2) F_I^j N_I\right] \frac{\partial N_I}{\partial x_j}\, d\Omega_h$$

$$= -\sum_{e \in I} \int_{\Omega_e} \left[F^j + (n\text{node} - 2) F_I^j N_I\right] \frac{\partial N_I}{\partial x_j}\, d\Omega_h$$

$$= -\sum_{e \in 1} \int_{\Omega_e} \left[\sum_{J=1}^{n\text{node}} N_J F_J^j + (n\text{node} - 2) F_I^j N_I\right] \frac{\partial N_I}{\partial x_j}\, d\Omega_h$$

$$= -\sum_{e \in 1} \int_{\Omega_e} \left[\sum_{\substack{J=1 \\ J \neq I}}^{n\text{node}} N_J F_J^j + (n\text{node} - 1) F_I^j N_I\right] \frac{\partial N_I}{\partial x_j}\, d\Omega_h$$

$$= -\sum_{e \in 1} \int_{\Omega_e} \sum_{\substack{J=1 \\ J \neq I}}^{n\text{node}} \left(N_J F_J^j + F_I^j N_I\right) \frac{\partial N_I}{\partial x_j}\, d\Omega_h \tag{A4}$$

Note that

$$\int_{\Omega_e} N_J \frac{\partial N_I}{\partial x_j}\, d\Omega_h = \int_{\Omega_e} N_I \frac{\partial N_I}{\partial x_j}\, d\Omega_h \tag{A5}$$

Then the following expression is obtained:

$$RHS(I) = -\sum_{\substack{J=1 \\ J \neq I}}^{n\text{node}} \left(\sum_{e \in I} \int_{\Omega_e} N_J \frac{\partial N_I}{\partial x_j}\, d\Omega_h\right)\left(F_J^j + F_I^j\right) \tag{A6}$$

which can be further simplified as

$$RHS(I) = \sum_{IJ}^{m_I} C_{IJ}^j \left(F_J^j + F_I^j\right) \tag{A7}$$

where m_I is the number of edges connected to the node I, and

$$C_{IJ}^j = -\sum_{e \in IJ} \frac{\Omega_e}{4} \frac{\partial N_I}{\partial x_j}\bigg|_e \tag{A8}$$

in three dimensions.

Acknowledgments

This research was sponsored by the Defense Nuclear Agency. Michael E. Giltrud served as the technical program monitor. Partial funding for the third author was also provided by the Air Force Office of Scientific Research. Leonidas Sakell served as the technical monitor.

References

[1]Luo, H., Baum, J. D., Löhner, R., and Cabello, J., "Adaptive Edge-Based Finite Element Schemes for the Euler and Navier-Stokes Equations on Unstructured Meshes," AIAA Paper 93-0336, Jan. 1993.

[2]Peraire, J., Peiro, J., and Morgan, K., "A 3D Finite Element Multigrid Solver for the Euler Equations," AIAA Paper 92-0449, Jan. 1992.

[3]Barth, T. J., "Numerical Aspects of Computing Viscous High Reynolds Number Flow on Unstructured Meshes," AIAA Paper 91-0721, Jan. 1991.

[4]Venkatakrishnan, V., and Mavriplis, D. J., "Implicit Solvers for Unstructured Meshes," AIAA Paper 91-1537, June 1991.

[5]Löhner, R., "Stars, Super Edges and Chains," George Washington Univ., GWU-CMEE Rept., 91/92-1, Washington, DC; see also *Computational Methods in Applied Mechanical Engineering* (submitted for publication).

[6]Billey, V., Périaux, J., Perrier, P., and Stoufflet, B., "2-D and 3-D Euler Computations with Finite Element Methods in Aerodynamic," *International Conference on Hypersonic Problems*, Saint-Etienne, Jan. 13-17, 1986.

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

[8]Whitaker, D. L., "Solution Algorithms for the Two-Dimensional Euler Equations on Unstructured Meshes," AIAA Paper 90-0697, Jan. 1990.

[9]Batina, J. T., "Three-Dimensional Flux-Split Euler Schemes Involving Unstructured Meshes," AIAA Paper 90-1649, Jan. 1990.

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

[11]Van Leer, B., "Towards the Ultimate Conservative Difference Scheme, II. Monotonicity and Conservation Combined in a Second Order Scheme," *Journal of Computational Physics*, Vol. 14, 1974, pp. 361–370.

[12]Löhner, R., and Parikh, P., "Three-Dimensional Grid Generation by the Advancing Front Method," *International Journal of Numerical Methods in Fluids*, Vol. 8, 1988, pp. 1135–1149.

[13]Ilelm, E. R., "CFD Wing/Pylon/Finned Store Mutual Interference Wind Tunnel Experiment," AEDC-TSR-91-P4, Arnold Engineering Development Center, Arnold AFB, TN, Jan. 1991.

[14]Schmitt, V., and Charpin, F., "Pressure Distributions on the ONERA M6-Wing at Transonic Mach Numbers," *Experimental Data Base for Computer Program Assessment*, AGARD AR-138, 1979.