*Computers Fluids* Vol. 23,
No. 1, pp. 1–21, 1994

# An Implicit Upwind Algorithm for Computing Turbulent Flows on Unstructured Grids.

W. Kyle Anderson and Daryl L. Bonhaus
NASA Langley Research Center
Hampton, Virginia 23665–5225

An implicit, Navier-Stokes solution algorithm is presented for the computation of turbulent flow on unstructured grids. The inviscid fluxes are computed using an upwind algorithm and the solution is advanced in time using a backward-Euler time-stepping scheme. At each time step, the linear system of equations is approximately solved with a point-implicit relaxation scheme. This methodology provides a viable and robust algorithm for computing turbulent flows on unstructured meshes.

Results are shown for subsonic flow over a NACA 0012 airfoil and for transonic flow over a RAE 2822 airfoil exhibiting a strong upper-surface shock. In addition, results are shown for 3–element and 4–element airfoil configurations. For the calculations, two one–equation turbulence models are utilized. For the NACA 0012 airfoil, a pressure distribution and force data are compared with other computational results as well as with experiment. Comparisons of computed pressure distributions and velocity profiles with experimental data are shown for the RAE airfoil and for the 3–element configuration. For the 4–element case, comparisons of surface pressure distributions with experiment are made. In general, the agreement between the computations and the experiment is good.

## 1. Introduction

For computing flows on complicated geometries such as multielement airfoils, the use of unstructured grids offers a good alternative to more traditional methods of analysis. This is primarily due to the promise of dramatically decreased time required to generate grids over complicated geometries. Also, unstructured grids offer the capability to locally adapt the grid to improve the accuracy of the computation without incurring the penalties associated with global refinement. While work remains to be done to fully realize their potential, much progress has been made in computing viscous flows on unstructured grids.

While several advances have been made for computing turbulent flow on unstructured grids (e.g. [1] [2]), probably the most mature and widely used code for computing two-dimensional turbulent viscous flow on unstructured grids is that of Mavriplis [3]. In this reference, the solution algorithm is a Galerkin finite-element discretization and a Runge-Kutta time-stepping algorithm is used in conjunction with multigrid to obtain very efficient solutions. The turbulence model predominantly utilized in this code is that of Baldwin and Lomax [4] although extensions have been made to include a two-equation turbulence model [5]. Other modifications to this code are presented in reference [6] in which backward-Euler time-differencing is used in conjunction with GMRES [7] to produce results which are competitive with multigrid for the cases considered.

The use of upwind differencing offers several advantages over a central-differencing formulation for computing viscous flows. For example, in references [8] and [9], it is clearly shown that with the flux-differencing scheme of Roe [10] the resolution of boundary layer details typically requires only half as many points as with a central-differencing code. As discussed in reference [11], the poor performance of the central-difference formulation is attributed to the scalar artificial dissipation formulas commonly used to damp odd-even oscillations and to provide non-linear stability.

For upwind calculations on unstructured grids, Barth [12] has described methodology for utilizing Roe's approximate Riemann solver [10] for the inviscid flux computations and a Galerkin formulation for the viscous terms. In this work, a sparse matrix solver is used in conjunction with a Runge-Kutta time-stepping algorithm for

updating the solution at each time step. The turbulence model included is that of Baldwin-Barth [13] and sample computations are shown over a single-element airfoil as well as a two-element airfoil in a wind tunnel.

For the current study, the flux-difference-splitting of Roe is used for computing the inviscid contribution to the flux and an implicit solver based on backward-Euler time differencing is utilized for updating the solution. At each time step, the linear system of equations is approximately solved with a red-black type relaxation procedure. This method circumvents the need to assemble large matrices and, therefore, significantly reduces the required memory. As in reference [12], the Baldwin-Barth turbulence model is used for computing flows at high Reynolds number. In addition, a recently developed turbulence model due to Spalart and Allmaras [14] is also utilized and comparisons between solutions obtained with each model are shown. Results are shown for subsonic flow over a NACA 0012 airfoil and for transonic flow over a RAE 2822 airfoil as well as for the flow over two multielement airfoils. Detailed comparisons are made with available experimental data. These comparisons include velocity profiles at particular locations along the surface as well as pressure and skin friction distributions.

## 2. Symbols

| | |
|---|---|
| $\mathbf{A}$ | matrix |
| $A$ | area of control volume. Also used in definition of numerical flux |
| $a$ | speed of sound |
| $\mathbf{b}$ | column vector for least squares |
| CFL | Courant-Friedichs-Lewy number |
| $C^*$ | constant used for Sutherlands law |
| $c$ | chord length |
| $c_{b_1}, c_{b_2}, c_{w_1}$ | constants used in Spalart-Allmaras turbulence model |
| $c_{\epsilon_1}, c_{\epsilon_2}$ | constants used in Baldwin-Barth turbulence model |
| $\mathbf{D}$ | diagonal components of $\mathbf{A}$ |
| $\mathrm{D_{ij}}$ | elements of $\mathbf{D}$ |
| $d$ | distance to nearest surface |
| $E$ | total energy per unit volume |
| $\vec{\mathbf{F}}$ | fluxes of mass, momentum, and energy |
| $\vec{\mathbf{F}}_\mathbf{i}$ | inviscid contribution to the fluxes |
| $\vec{\mathbf{F}}_\mathbf{v}$ | viscous contribution to the fluxes |
| $\mathbf{f}, \mathbf{g}$ | components of inviscid fluxes |
| $\mathbf{f_v}, \mathbf{g_v}$ | components of viscous fluxes |
| $f_2$ | function used in Baldwin-Barth turbulence model |
| $f_w, f_{t_1}, f_{t_2}$ | functions used in Spalart-Allmaras turbulence model |
| $\widetilde{\mathbf{k}}_\infty$ | thermal conductivity in freestream |

| | |
|---|---|
| $\kappa$ | Karman constant |
| $L$ | reference length |
| $l$ | distance between centroids adjacent to an edge |
| $l_d$ | length of edge in dual mesh |
| $M_\infty$ | freestream Mach number |
| $\mathrm{N}$ | number of edges meeting at a node |
| $\mathrm{N_d}$ | number of edges making up the boundary of the control volume |
| $\widehat{\mathbf{n}}$ | unit normal vector |
| $\vec{\mathbf{n}}$ | directed area vector |
| $\widehat{\mathbf{n}}_d$ | unit normal to a boundary for dual mesh |
| $\vec{\mathbf{n}}_L, \vec{\mathbf{n}}_R$ | directed areas formed by connecting the midpoint of an edge to the centroids of the triangles to the left and right, respectively |
| $\widehat{n}_\mathrm{x}, \widehat{n}_\mathrm{y}$ | $x$ and $y$ components of a unit normal |
| $\mathbf{O}$ | all off-diagonal components of $\mathbf{A}$ |
| $P$ | production of turbulent kinetic energy |
| $Pr$ | Prandtl number |
| $p$ | pressure |
| $\mathbf{Q}$ | conserved state vector, $\mathbf{Q} = [\rho \quad \rho u \quad \rho v \quad E\,]^\mathrm{T}$ . Also used to denote an orthogonal matrix |
| $\mathbf{q}$ | primitive state vector, $\mathbf{q} = [\rho \quad u \quad v \quad p\,]^\mathrm{T}$ |
| $\mathbf{q}^+, \mathbf{q}^-$ | primitive state vector on a cell boundary obtained from extrapolation |
| $\mathbf{q_L}, \mathbf{q_R}$ | primitive state vector at the nodes on either side of an edge |
| $\mathbf{q_{x_0}}, \mathbf{q_{y_0}}$ | $x$ and $y$ components of gradient at node |
| $q_x, q_y$ | components of heat flux |
| $\mathbf{R}$ | residual for a control volume. Also used to denote an upper triangular matrix |
| $Re$ | Reynolds Number |
| $\widetilde{R}_t$ | modified turbulent Reynolds Number |
| $RHS$ | Right hand side |
| $R^\pm$ | Riemann invariants |
| $\widetilde{S}$ | production term for Spalart-Allmaras model |
| $\vec{\mathbf{r}}$ | vector from vertex to an edge of the dual control volume |
| $r_{11}, r_{12}, r_{22}$ | components of upper triangular matrix $\mathbf{R}$ |
| $\mathrm{ref}$ | denotes reference condition |
| $S$ | entropy |

| | |
|---|---|
| $T$ | temperature |
| $t$ | time |
| $U$ | velocity normal to boundary of control volume |
| $u, v$ | Cartesian velocities in $x$ and $y$ directions |
| $W_i^x, W_i^y$ | least square weights for computing gradients |
| $x, y$ | Cartesian coordinates |
| $x_i, y_i$ | coordinates of mesh vertices |
| $x_0, y_0$ | coordinates of a central vertex |
| $\alpha$ | angle of attack |
| $\gamma$ | ratio of specific heats, taken as 1.4 |
| $\mu$ | laminar viscosity |
| $\mu_t$ | turbulent viscosity |
| $\nu$ | $\mu/\rho$ |
| $\nu_t$ | $\mu_t/\rho$ |
| $\widetilde{\nu}$ | dependent variable for Spalart-Allmaras turbulence model |
| $\rho$ | density |
| $\sigma$ | constant for Spallart-Allmaras turbulence model |
| $\sigma_\epsilon$ | constant for Baldwin-Barth turbulence model |
| $\tau_{xx}, \tau_{xy}, \tau_{yy}$ | shear stress terms |
| $\Phi$ | numerical flux |
| $\psi$ | flux limiter |
| $\Omega$ | boundary of cell |

## 3. Governing Equations

The governing equations are the time-dependent Reynolds-averaged Navier-Stokes equations. The equations are expressed as a system of conservation laws relating the rate of change of mass, momentum, and energy in a control volume of area $A$ to the fluxes of these quantities through the volume. The equations ( nondimensionalized by free stream density, $\widetilde{\rho}_\infty$, speed of sound, $\widetilde{a}_\infty$, temperature, $\widetilde{T}_\infty$, viscosity, $\widetilde{\mu}_\infty$, thermal conductivity, $\widetilde{k}_\infty$, and a reference length, $L$) are given as

$$A\frac{\partial \mathbf{Q}}{\partial t} + \oint_{\partial\Omega} \vec{\mathbf{F}} \cdot \hat{\mathbf{n}} dl - \oint_{\partial\Omega} \vec{\mathbf{F}}_v \cdot \hat{\mathbf{n}} dl = 0 \tag{1}$$

where $\widehat{\mathbf{n}}$ is the outward-pointing unit normal to the control volume. The vector of dependent variables $\mathbf{Q}$, and the flux vectors $\vec{\mathbf{F}}_\mathbf{i}$ and $\vec{\mathbf{F}}_\mathrm{v}$ are given as

$$\mathbf{Q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E \end{bmatrix} \tag{2}$$

$$\vec{\mathbf{F}}_{\mathbf{i}} = \mathbf{f}\,\hat{i} + \mathbf{g}\,\hat{j} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ (E+p)u \end{bmatrix} \hat{i} + \begin{bmatrix} \rho v \\ \rho v u \\ \rho v^2 + p \\ (E+p)v \end{bmatrix} \hat{j} \tag{3}$$

$$\vec{\mathbf{F}}_{\mathbf{v}} = \mathbf{f}_{\mathbf{v}}\,\hat{\mathrm{i}} + \mathbf{g}_{\mathbf{v}}\,\hat{\mathrm{j}} = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ u\tau_{xx} + v\tau_{xy} - q_x \end{bmatrix} \hat{\mathrm{i}} + \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ u\tau_{yx} + v\tau_{yy} - q_y \end{bmatrix} \hat{\mathrm{j}} \tag{4}$$

Here, $\vec{\mathbf{F}}_{\mathbf{i}}$ and $\vec{\mathbf{F}}_{\mathbf{v}}$ are the inviscid and viscous flux vectors respectively; the shear stress and heat conduction terms are given as

$$\tau_{xx} = (\mu + \mu_t)\frac{M_\infty}{Re}\frac{2}{3}[2u_x - v_y] \tag{5}$$

$$\tau_{yy} = (\mu + \mu_t)\frac{M_\infty}{Re}\frac{2}{3}[2v_y - u_x] \tag{6}$$

$$\tau_{xy} = (\mu + \mu_t)\frac{M_\infty}{Re}[u_y + v_x] \tag{7}$$

$$q_x = \frac{-M_\infty}{Re(\gamma - 1)}\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial a^2}{\partial x} \tag{8}$$

$$q_y = \frac{-M_\infty}{Re(\gamma - 1)}\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial a^2}{\partial y} \tag{9}$$

The equations are closed with the equation of state for a perfect gas

$$p = (\gamma - 1)\left[E - \rho\left(u^2 + v^2\right)/2\right] \tag{10}$$

and the laminar viscosity is determined through Sutherland's law

$$\frac{\mu}{\mu_\infty} = \frac{(1 + C^*)}{(T/T_\infty + C^*)}(T/T_\infty)^{3/2} \tag{11}$$

where $C^* = \frac{198.6}{460.0}$ is Sutherland's constant divided by a free stream reference temperature which is assumed to be $460^{\circ}$ Rankine.

## 4. Solution Algorithm

The flow solver is an implicit, upwind-differencing algorithm in which the inviscid fluxes are obtained on the faces of each control volume using the flux-difference-splitting technique of Roe [10]. For the current scheme, a node-based algorithm is used in which the variables are stored at the vertices of the mesh and the equations are solved on non-overlapping control volumes surrounding each node. The viscous terms are evaluated with a finite-volume formulation which is equivalent to a Galerkin-type approximation and results in a central-difference-type formulation for these terms. The solution at each time step is updated using the linearized backward-Euler, time-differencing scheme. At each time step, the linear system of equations is approximately solved with a subiterative procedure in which the unknowns are divided into two groups (colors) according to whether the node number is even or odd. For each subiteration, the solution is obtained by solving for all the unknowns in one color before proceeding to the next one. This corresponds to a red-black type of iterative algorithm for solving the linear system.

### 4.1. Finite Volume Scheme

The solution is obtained by dividing the domain into a finite number of triangles from which control volumes are formed that surround each vertex in the mesh. Equation 1 is then numerically integrated over the closed boundaries of the control volumes surrounding each node. These control volumes are determined by connecting the centroid of each triangle to the midpoint of the edges as shown in figure 1. These non-overlapping control volumes combine to completely cover the domain and are considered to form a mesh which is dual to the mesh composed of triangles formed from the vertices.

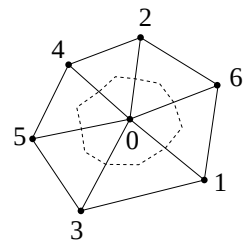


Figure 1. Control Volume Surrounding Node

The numerical evaluation of the surface integrals in equation 1 is conducted separately for the inviscid and viscous contributions. For a finite volume formulation, the inviscid contribution can be approximated using midpoint integration of the fluxes over each edge of the dual mesh that defines the boundary of the control volume. i.e.

$$\oint_{\partial\Omega} \vec{\mathbf{F}} \cdot \hat{\mathbf{n}}_d dl_d = \oint_{\partial\Omega} \vec{\mathbf{F}} \cdot d\vec{\mathbf{n}}_d \approx \sum_{\mathrm{i}=1}^{N_d} \Phi\left(\mathbf{q}^+, \mathbf{q}^-; \widehat{\mathbf{n}}_{\mathrm{d_i}}\right) \times l_{d_i} \tag{12}$$

Here $N_d$ is the number of edges from the dual mesh that make up the surface of the control volume and $l_{d_i}$ is the length of the edge. Also, $\Phi(\mathbf{q}^+, \mathbf{q}^-; \widehat{\mathbf{n}}_{d_i})$ is a numerical flux formed from data on the left ($\mathbf{q}^+$) and right sides ($\mathbf{q}^-$) of the face which are determined by extrapolation from the surrounding data. Details of this procedure are presented in a later section. Note that the distinction between the left and right hand side of a face is determined by the direction of the normal which has been specified a priori and is considered to point from the left side of the face to the right. The flux calculations for a node are made by distributing the contributions from each of the edges. Since the associated normal vector is directed outward from the control volume on the left, the contribution of each edge to the integral in equation 12 is added to the control volume to the left and subtracted from that on the right.

A simpler and computationally more efficient process than the one described above can be achieved by replacing the two directed areas from the dual mesh that join at the midpoint of an edge in the triangular mesh with a single directed area. For example, in figure 2, the average directed area is defined as

$$\vec{\mathbf{n}} = \vec{\mathbf{n}}_{\mathbf{L}} + \vec{\mathbf{n}}_{\mathbf{R}} \tag{13}$$

This is identical to the directed area vector normal to the line formed by connecting the cell centroids of the adjoining cells. Observe that for this choice of directed area, if the numerical flux on the face is formed from the arithmetic average of the fluxes at the two nodes,

$$\Phi\left(\mathbf{q}^+, \mathbf{q}^-; \widehat{\mathbf{n}}\right) = \frac{1}{2}\left(\vec{\mathbf{F}}(\mathbf{q}_{\mathbf{L}}) + \vec{\mathbf{F}}(\mathbf{q}_{\mathbf{R}})\right) \cdot \widehat{\mathbf{n}} \tag{14}$$

the resulting scheme is equivalent to a Galerkin finite-element method [12]. Note that in equation 14, $\mathbf{q}^+$ is simply taken to be the data at the left node ($\mathbf{q_L}$) and $\mathbf{q}^-$ represents the data to the right ($\mathbf{q_R}$). The inviscid boundary

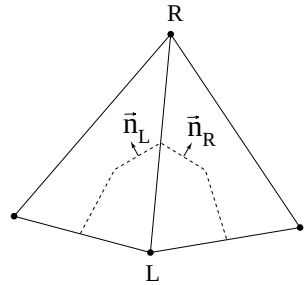


Figure 2. Computing an Average Normal

integral contribution can now be written as

$$\oint_{\partial\Omega} \vec{\mathbf{F}} \cdot \hat{\mathbf{n}} dl = \oint_{\partial\Omega} \vec{\mathbf{F}} \cdot d\vec{\mathbf{n}} \approx \sum_{\mathrm{i}=1}^{N} \Phi\left(\mathbf{q}^+, \mathbf{q}^-; \widehat{\mathbf{n}}\right) \times l_i \tag{15}$$

where $N$ is the number of edges in the triangular mesh incident to the node under consideration and $\Phi(\mathbf{q}^+, \mathbf{q}^-; \widehat{\mathbf{n}})$ represents the numerical flux on the newly defined face.

For obtaining an upwind scheme, the numerical fluxes on the edges of the control volume are obtained using Roe's approximate Riemann solver [10]. These fluxes are formed from data on either side of the face as

$$\mathbf{\Phi} = \frac{1}{2}\left(\vec{\mathbf{F}}\left(\mathbf{q}^+;\widehat{\mathbf{n}}\right) + \vec{\mathbf{F}}\left(\mathbf{q}^-;\widehat{\mathbf{n}}\right)\right) - \frac{1}{2}|\mathbf{A}(\widehat{\mathbf{q}};\widehat{\mathbf{n}})|\left(\mathbf{q}^+ - \mathbf{q}^-\right) \tag{16}$$

Here, $\vec{\mathbf{F}}(\mathbf{q}^+;\widehat{\mathbf{n}})$ and $\vec{\mathbf{F}}(\mathbf{q}^-;\widehat{\mathbf{n}})$ are the inviscid flux vectors given by equation 3 formed from the data on the left side ($\mathbf{q}^+$) and right side ($\mathbf{q}^-$) of the face, respectively. The matrix $|\mathbf{A}(\widehat{\mathbf{q}};\widehat{\mathbf{n}})|$ is formed from the variables on the cell face which are determined using an averaging procedure described in reference [10]. Note that an equally valid alternative to this formulation is to form the numerical flux on the face as

$$\mathbf{\Phi} = \frac{1}{2}\left(\vec{\mathbf{F}}(\mathbf{q_L}) + \vec{\mathbf{F}}(\mathbf{q_R})\right) \cdot \widehat{\mathbf{n}} - \frac{1}{2}|\mathbf{A}(\widehat{\mathbf{q}};\widehat{\mathbf{n}})|\left(\mathbf{q}^+ - \mathbf{q}^-\right) \tag{17}$$

where $\vec{\mathbf{F}}(\mathbf{q_L})$ and $\vec{\mathbf{F}}(\mathbf{q_R})$ are the flux vectors evaluated from the values at the nodes on either side of the edge instead of from data extrapolated to the edge. As discussed above and in [12], with the matrix term neglected the formation of the flux in this manner yields a Galerkin discretization. Although results are only shown below in which inviscid fluxes are obtained using equation 16, solutions obtained using equation 17 have also been obtained and no observable difference in pressures, skin frictions, or velocity vectors have been seen.

For first-order-accurate differencing, the data on the left and right sides of the cell face ($\mathbf{q}^+$ and $\mathbf{q}^-$) are set equal to the data at the nodes lying on either side of the cell face. For higher-order differencing, the primitive variables are extrapolated to the boundaries of the control volumes using a Taylor series expansion about the central vertex so that the data on the face is given by

$$\mathbf{q}_{\mathrm{face}} = \mathbf{q}_{\mathrm{node}} + \psi \bigtriangledown \mathbf{q} \cdot \vec{\mathbf{r}} \tag{18}$$

where $\bigtriangledown\mathbf{q}$ represents the gradient of the variables at the node and $\vec{\mathbf{r}}$ is the vector extending from the vertex to the midpoint of each edge. In the above equation, $\psi$ is a variable that ranges from zero to one and is used to control oscillations that may occur at steep gradients. For the current study, $\psi$ is determined using the flux-limiting procedure described in reference [15]. Note that although the data on either side of the cell face ($\mathbf{q}^+$ and $\mathbf{q}^-$) will be discontinuous, a single-valued flux is obtained through equation 16 or 17.

For evaluating the gradient, $\nabla \mathbf{q}$, a least squares procedure is used in which the data surrounding each node is assumed to behave linearly. Referring to figure 3 as an example, the data at each node surrounding the center node may be expressed as

$$\mathbf{q_i} = \mathbf{q}_0 + \mathbf{q_{x_0}}(\mathbf{x_i} - \mathbf{x_0}) + \mathbf{q_{y_0}}(\mathbf{y_i} - \mathbf{y_0}) \tag{19}$$

By expressing the data in a like manner at each of the $N$ surrounding nodes, an $N \times 2$ system of equations is

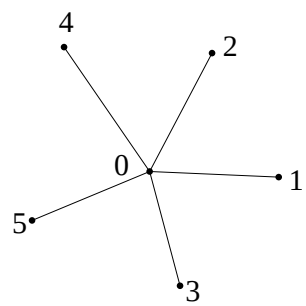


Figure 3. Nodes For Least Square Reconstruction of Data

formed which can be solved to obtain the gradients at the nodes

$$\begin{bmatrix} \Delta x_1 & \Delta y_1 \\ \Delta x_2 & \Delta y_2 \\ \vdots & \vdots \\ \Delta x_N & \Delta y_N \end{bmatrix} \begin{Bmatrix} \mathbf{q_{x_0}} \\ \mathbf{q_{y_0}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{q_1} - \mathbf{q_0} \\ \mathbf{q_2} - \mathbf{q_0} \\ \vdots \\ \mathbf{q_N} - \mathbf{q_0} \end{Bmatrix} \tag{20}$$

This represents an over-determined system of linear equations, $\mathbf{Ax} = \mathbf{b}$ which may be solved in a least squares sense using the normal equation approach in which both sides are multiplied by the transpose of the coefficient matrix, $\mathbf{A}$, so that a $2 \times 2$ system of equations is obtained.

$$\mathbf{A^T A x} = \mathbf{A^T b} \tag{21}$$

Unfortunately, the sensitivity of the solution obtained using this technique is dependent on the square of the condition number of $\mathbf{A}$ [16]. For problems on grids which are highly stretched, the accuracy of the process can be severely compromised.

Therefore, in the current study, a Gram-Schmidt process is used in which the system of equations is solved by decomposing the $\mathbf{A}$ matrix into a product of an orthogonal matrix $\mathbf{Q}$, and an upper triangular matrix $\mathbf{R}$, i.e.

$$\mathbf{A} = \mathbf{QR} \tag{22}$$

so that the solution is obtained by

$$\mathbf{x} = \mathbf{R^{-1} Q^T b} \tag{23}$$

A similar procedure has been used for computing inviscid flows on unstructured meshes in references [17] and [18]. Further details of least squares procedures can be found in reference [16].

With this procedure, the numerical difficulties associated with solving linear systems with near rank deficiency is significantly reduced over the use of normal equations. Further improvement may be achieved through the use of Householder transformations [16] or singular value decompositions[16]. The Gram-Schmidt process, however, allows for the easy precomputation and storage of weights so that the gradients at each node can be calculated by

"looping" over the edges in the mesh and distributing the contribution of each edge to each of the nodes. The resulting formulas for calculating the gradients at the center node in figure 3 are given as

$$\mathbf{q_{x_0}} = \sum_{\mathrm{i}=1}^{\mathrm{N}} \mathrm{W_i^x}(\mathbf{q_i} - \mathbf{q_0}) \qquad \mathbf{q_{y_0}} = \sum_{\mathrm{i}=1}^{\mathrm{N}} \mathrm{W_i^y}(\mathbf{q_i} - \mathbf{q_0}) \tag{24}$$

where the summation is over all the edges that connect to the node (N=5 in figure 3) and the weights are given by

$$W_i^x = \frac{x_i - x_0}{r_{11}^2} - \frac{r_{12}}{r_{11} r_{22}^2}\left[(y_i - y_0) - (x_i - x_0)\frac{r_{12}}{r_{11}}\right] \tag{25}$$

$$W_i^y = \frac{1}{r_{22}^2}\left[(y_i - y_0) - (x_i - x_0)\frac{r_{12}}{r_{11}}\right] \tag{26}$$

where

$$r_{11} = \left[\sum_{i=1}^{N} (x_i - x_0)^2\right]^{1/2} \tag{27}$$

$$r_{12} = \frac{\sum_{i=1}^{N} (x_i - x_0)(y_i - y_0)}{r_{11}} \tag{28}$$

$$r_{22} = \left[\sum_{i=1}^{N} \left((y_i - y_0) - (x_i - x_0)\frac{r_{12}}{r_{11}}\right)^2\right]^{1/2} \tag{29}$$

Note that equation 20 yields an unweighted least squares procedure in which all the data surrounding the central node are given equal consideration. Numerical experiments have been conducted which indicate that for reconstructing nonlinear data on highly stretched meshes using equation 18, the unweighted formulation is far superior to either inverse distance weighting or the use of gradients calculated with Green's theorem. It was found, however, that for computing the actual values of gradients, inverse distance weighting and Greens theorem give very similar results, both of which are more accurate than unweighted least squares. Their failure to accurately reproduce surrounding data via equation 18 is attributable to the flawed assumption of linearly varying data. Therefore, for reconstructing data on boundaries of control volumes, the unweighted least squares procedure is used. When actual gradients are required, as in the production terms for the turbulence models, Green's theorem is utilized.

For computing the viscous contribution to the residual, $\int_{\partial\Omega} \widehat{\mathbf{F}}_{\mathbf{v}} \cdot \widehat{\mathbf{n}}\mathrm{dl}$, a finite volume approach is followed. For example, the surface integrals of the viscous contributions involve terms such as the two terms below

$$\int_{\partial\Omega} [(\mu + \mu_t)u_x] \cdot \widehat{\mathrm{n}}_x \, dl \qquad \int_{\partial\Omega} [(\mu + \mu_t)v_y] \cdot \widehat{\mathrm{n}}_x \, dl \tag{30}$$

Recall that the boundary of the control volume is formed by connecting the centroid of each triangle to the midpoint of each edge as shown in figure 1. By assuming a linear distribution in each triangle, gradients along these boundaries may be considered as constant. Therefore, quantities such as those in brackets in 30 above are first evaluated in each triangle of the mesh where $\mu + \mu_t$ are computed from an average of the surrounding nodes and the gradients are computed using Green's theorem. Although this approach is from a finite-volume point of view, the computation of the viscous terms in this manner leads to identical formulas as a Galerkin approximation as given in reference [12].

Note that for calculating viscous flows, the aspect ratio of the triangles in the boundary layer can be very large. Babuska [19] has shown in two dimensions that high aspect ratio triangles are not always detrimental as long as no angle is too close to 180 degrees. In the present study, all the grids have been generated using the method described in reference [20] which triangulates a set of points derived from structured grids. The resulting meshes tend to have very few cells with angles larger than 120 degrees and therefore the accuracy of the computations is not severely degraded.

### 4.2. Time Advancement Scheme

The time advancement algorithm is based on the linearized backward-Euler time-differencing scheme which yields a system of linear equations for the solution at each step given by

$$[\mathbf{A}]^{\mathrm{n}}\{\Delta\mathbf{Q}\}^{\mathrm{n}} = \{\mathbf{R}\}^{\mathrm{n}} \tag{31}$$

where

$$[\mathbf{A}]^{\mathrm{n}} = \frac{A}{\Delta t}\mathrm{I} + \frac{\partial \mathbf{R}^{\mathrm{n}}}{\partial \mathbf{Q}} \tag{32}$$

The solution of this system of equations is obtained by a relaxation scheme in which $\{\Delta\mathbf{Q}\}^{n}$ is obtained through a sequence of iterates, $\{\Delta\mathbf{Q}\}^{i}$ which converge to $\{\Delta\mathbf{Q}\}^{n}$. Note that several variations of classic relaxation procedures have been used in the past for solving the Euler equations on unstructured grids (see for example [21], [18], [1] and [22]). To clarify the scheme, $[\mathbf{A}]^{\mathrm{n}}$ is first written as a linear combination of two matrices representing the diagonal and off-diagonal terms

$$[\mathbf{A}]^{\mathrm{n}} = [\mathbf{D}]^{\mathrm{n}} + [\mathbf{O}]^{\mathrm{n}} \tag{33}$$

The simplest iterative scheme for obtaining a solution to the linear system of equations is a Jacobi type method in which all the off-diagonal terms (i.e. $[\mathbf{O}]^{\mathrm{n}}\{\boldsymbol{\Delta}\mathbf{Q}\}$), are taken to the right-hand side of equation 31 and are evaluated using the values of $\{\Delta\mathbf{Q}\}^{\mathrm{i}}$ from the previous subiteration level i. This scheme can be represented as

$$[\mathbf{D}]^{\mathrm{n}}\{\Delta\mathbf{Q}\}^{\mathrm{i}+1} = \left[\{\mathbf{R}\}^{\mathrm{n}} - [\mathbf{O}]^{\mathrm{n}}\{\Delta\mathbf{Q}\}^{\mathrm{i}}\right] \tag{34}$$

The convergence rate of this process can be slow but it may be accelerated somewhat by using the latest values of $\{\Delta\mathbf{Q}\}$ as soon as they are available. This can be achieved by adopting a red-black type of strategy in which all the even-numbered nodes are updated first, followed by the solution of the odd-numbered ones. This procedure can be represented as

$$[\mathbf{D}]\{\Delta\mathbf{Q}\}^{\mathrm{i}+1} = \left[\{\mathbf{R}\}^{\mathrm{n}} - [\mathbf{O}]^{\mathrm{n}}\{\Delta\mathbf{Q}\}^{\mathrm{i+1}}_{\mathrm{i}}\right] \tag{35}$$

where $\{\mathbf{Q}\}^{\mathrm{i+1}}_{\mathrm{i}}$ is the most recent value of $\mathbf{Q}$ and will be at subiteration level i+1 for the even numbered nodes which have been previously updated and will be at level i for the odd numbered nodes.

For Laplace's equation, the use of a red-black strategy offers a factor of two improvement in the asymptotic convergence rate over point Jacobi [23]. Numerical experiments for the Euler and Navier-Stokes equations indicate that a factor of approximately 1.6 is attained in practice. While the use of the red-black algorithm offers improvement over the Jacobi iteration strategy, the convergence of the linear system can still be slow, particularly on grids with highly stretched cells necessary for turbulent Navier-Stokes calculations. Fortunately, it is not necessary to fully converge the linear system to provide a robust algorithm that remains stable at time steps much larger than an explicit scheme can provide.

Although the use of high CFL numbers is desirable for convergence of the nonlinear system, the greatest benefits of large time steps are only obtained when the linear system is solved well at each time step. Unfortunately, large time steps are generally detrimental to the convergence of the subiterations. Namely, the convergence of the subiterative procedure is greatly enhanced by smaller time steps resulting from larger diagonal contributions. Therefore, a reasonable compromise must be made to allow CFL numbers small enough for good convergence of the linear system but large enough to also provide good convergence of the nonlinear system. In the current study, it has been found that although computations with CFL numbers of 500 or more remain stable, the best convergence for the Navier-Stokes equations on large grids is achieved for moderate CFL numbers between 100 and 200.

To enhance the convergence to steady state, local time-stepping is used. The time-step calculation is based strictly on inviscid stability considerations for each node as given by

$$\Delta t = \mathrm{CFL}\frac{\mathrm{A}}{\int\limits_{\partial\Omega}(|\mathrm{U}| + \mathrm{a})\,dl} \tag{36}$$

where $U$ is the velocity normal the boundary of the control volume.

### 4.3. Boundary Conditions

The boundary conditions on the body correspond to no-slip and a prescribed wall temperature. They are enforced by modifying the matrix terms in equation 35 to appropriately reflect the desired boundary conditions. To more clearly demonstrate the procedure, a slightly expanded representation of one of the rows in equation 35 is given by

$$\begin{bmatrix} D_{11} & D_{12} & D_{13} & D_{14} \\ D_{21} & D_{22} & D_{23} & D_{24} \\ D_{31} & D_{32} & D_{33} & D_{34} \\ D_{41} & D_{42} & D_{43} & D_{44} \end{bmatrix} \begin{Bmatrix} \Delta\rho \\ \Delta\rho u \\ \Delta\rho v \\ \Delta E \end{Bmatrix} = \begin{Bmatrix} RHS_1 \\ RHS_2 \\ RHS_3 \\ RHS_4 \end{Bmatrix} \tag{37}$$

where $RHS$ represents both the residual and off-diagonal terms on the right hand side of equation 35 and $D_{ij}$ represents the individual components of one of the diagonal blocks in $[\mathbf{D}]$. The density can be determined from the continuity equation during the solution process from the first row of equation 37. However, the contribution to the continuity residual along the boundary involves an integration around the dual mesh surrounding the node and a segment of the body surface as depicted in figure 4. The contribution from the surface, assuming zero velocity at the wall is identically zero. The second and third rows are modified so the solution of equation 37 maintains a zero velocity at the nodes on the solid boundaries. Further, the fourth row is altered to preserve a constant temperature which is set to the adiabatic wall temperature [24]

$$\frac{T_{wall}}{T_\infty} = 1 + \sqrt{Pr}\frac{\gamma - 1}{2}M_\infty^2 \tag{38}$$

The constant wall temperature assumption is used to relate the change in energy at the wall to the change in density

$$\Delta E_{wall} = \frac{T_{\text{wall}}}{\gamma(\gamma - 1)}\Delta\rho \tag{39}$$

The resulting matrix now reflects the enforcement of appropriate boundary conditions at the wall and is given by

$$\begin{bmatrix} D_{11} & D_{12} & D_{13} & D_{14} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\frac{T_{wall}}{\gamma(\gamma-1)} & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} \Delta\rho \\ \Delta\rho u \\ \Delta\rho v \\ \Delta E \end{Bmatrix} = \begin{Bmatrix} RHS_1 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \tag{40}$$

In the far-field, the data at the nodes are not explicitly set but are obtained through the solution process in much the same manner as the points interior to the flowfield. The only distinction between a far-field boundary node and an interior node comes from the fact that the enforcement of the boundary condition is reflected in the residual calculation. Referring to figure 4, the flux along the boundary between $\mathrm{i}$ and $\mathrm{i} + \frac{1}{2}$ is calculated from a weighted average of a characteristic reconstruction of the data at $\mathrm{i}$ and $\mathrm{i} + 1$. Note that the characteristic reconstructions at these nodes are only utilized to form the flux on the boundary; the actual values at the nodes are updated through equation 35.

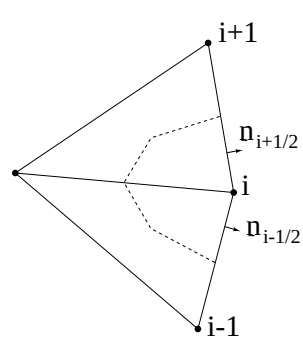


Figure 4. Geometry for Calculating Flux on Solid and Far-Field Boundaries

For obtaining the characteristic reconstruction in the far-field, the flow-field is assumed to be essentially inviscid so that quantities needed for the computation of the flux along the outer boundary are obtained from two locally one-dimensional Riemann invariants. While this assumption is not strictly valid, the perturbations from inviscid flow are generally small so that the assumption of inviscid flow is acceptable. The Riemann invariants are considered constant along characteristics defined normal to the outer boundary and are given as

$$R^{\pm} = U \pm \frac{2a}{\gamma - 1} \tag{41}$$

For subsonic flow, $R^-$ can be evaluated locally from free stream conditions taken to be outside the computational domain and $R^+$ is evaluated locally from the interior of the domain and is taken to correspond to the present value of the data at each node on the outer boundary. The local normal velocity and speed of sound on the boundary are calculated using the Riemann invariants as

$$U_{\text{boundary}} = \frac{1}{2}\left(R^+ + R^-\right) \tag{42}$$

$$a_{\text{boundary}} = \frac{\gamma - 1}{4}\left(R^+ - R^-\right) \tag{43}$$

The Cartesian velocities are determined on the outer boundary by decomposing the normal and tangential velocity vectors into components yielding

$$\begin{aligned} u_{\text{boundary}} &= u_{\text{ref}} + \widehat{n}_{\text{x}}\left(U_{\text{boundary}} - U_{\text{ref}}\right) \\ v_{\text{boundary}} &= v_{\text{ref}} + \widehat{n}_{\text{y}}\left(U_{\text{boundary}} - U_{\text{ref}}\right) \end{aligned} \tag{44}$$

where the subscript, ref, represents values obtained from free stream values for inflow and from current data at the boundary node for outflow.

The entropy is determined using either the free stream value or the value at the boundary node depending on whether the boundary is an inflow or outflow boundary. Once the entropy is known, the density on the far-field boundary is calculated from the entropy and speed of sound as

$$\rho_{\text{boundary}} = \left(\frac{a^2_{\text{boundary}}}{\gamma S_{\text{boundary}}}\right)^{\frac{1}{\gamma-1}} \tag{45}$$

The energy is then calculated from the equation of state and the flux is then calculated using these characteristic reconstructions.

### *4.4. Turbulence Modeling*

For the current study, two different turbulence models are considered. At each time step, the equation for the turbulent viscosity is solved separately from the flow equations resulting in a loosely coupled solution process that allows for the easy interchange of new turbulence models. The turbulence models considered in the current study are that of Baldwin and Barth [13] and Spalart and Allmaras [14]. The governing equation for each of these models is given below although reference to the original publications should be made for the precise meaning of all the terms and for details related to the implementation.

The first model is that of Baldwin and Barth, which is a one-equation advection-diffusion model derived using the $k-\epsilon$ equations. The dependent variable for the model is a field quantity , $\nu\widetilde{R}_t$, from which the turbulent viscosity can be obtained from an algebraic relation. The partial differential equation for $\nu\widetilde{R}_t$ is given in non-dimensional terms as

$$\begin{aligned} \frac{D\left(\nu\widetilde{R}_t\right)}{Dt} &= \frac{M_\infty}{R_e}\left[\left(\nu + \frac{2\nu_t}{\sigma_\epsilon}\right)\nabla^2\left(\nu\widetilde{R}_t\right) - \frac{1}{\sigma_\epsilon}\nabla\cdot\left(\nu_t\nabla\left(\nu\widetilde{R}_t\right)\right)\right] \\ &\quad + \left(c_{\epsilon_2}f_2 - c_{\epsilon_1}\right)\sqrt{\nu\widetilde{R}_t P} \end{aligned} \tag{46}$$

Here, the left hand side of the equation is the substantial derivative

$$\frac{D(\cdot)}{Dt} = \frac{\partial(\cdot)}{\partial t} + u\frac{\partial(\cdot)}{\partial x} + v\frac{\partial(\cdot)}{\partial y} \tag{47}$$

and the right hand side of the equation represents diffusion and production terms respectively. The details of each of the terms can be found in reference [13].

In addition to the turbulence model of Baldwin and Barth, the model of Spalart and Allmaras [14] has also been utilized. This is also a one–equation turbulence model but its derivation is much more heuristic in nature, relying heavily on empiricism and dimensional analysis. The form of this equation is similar to that of the previous model and is given by

$$\begin{aligned}\frac{D(\widetilde{\nu})}{Dt} &= c_{b_1}[1 - f_{t_2}]\widetilde{S}\widetilde{\nu} + \frac{M_\infty}{\sigma R_e}\left[\nabla \cdot \left((\nu + (1 + c_{b_2})\widetilde{\nu})\nabla\widetilde{\nu} - c_{b_2}\widetilde{\nu}\nabla^2\widetilde{\nu}\right)\right] \\ &\quad - \frac{M_\infty}{Re}\left[c_{w_1} f_w - \frac{c_{b_1}}{\kappa^2} f_{t_2}\right]\left[\frac{\widetilde{\nu}}{d}\right]^2 + \frac{Re}{M_\infty} f_{t_1}\Delta U^2\end{aligned} \tag{48}$$

where again, the original reference should be consulted for a precise explanation of each of the terms. Note however, that this model includes a destruction term which is absent in the previous model as well as a source term used for specifying transition locations.

For both the Baldwin-Barth and the Spalart-Allmaras turbulence models, the equations are solved using a backward-Euler time-stepping-scheme similar to that used for the flow variables. For both models, the dependent variable ($\nu\widetilde{R}_t$ for Baldwin-Barth and $\widetilde{\nu}$ for Spalart-Allmaras) is specified to be zero on the body and is therefore not solved for during the solution process. In the far-field, the dependent variable is taken to be free stream for inflow and is extrapolated from the interior for outflow. For the discretization of the terms in each model, the convective terms are evaluated with first-order upwind differencing and the higher-order derivatives are evaluated in the same manner as described for the flow solver.

## 5. Numerical Results

Results are shown below for several test cases. For each case, the CFL number has been linearly ramped from 20 to 100 over 100 iterations and 15 subiterations were done at each time step to obtain an approximate solution of the linear system. While this strategy is in no way optimal for every case, it has been found to yield satisfactory results. As previously mentioned, all the grids used in the current study have been generated with the code described in reference [20] which triangulates point sets derived from structured grids generated around individual components. Consequently, few large angles appear in the mesh; typically fewer than two percent of the angles are larger than 120 degrees.

All the figures shown below have been obtained assuming fully turbulent flow by specifying a small level of turbulent viscosity in the free stream. Although not shown, studies with the Spalart-Allmaras turbulence model to specify transition locations and to examine the effects of varying the level of free stream turbulence have also been conducted. For the cases considered, neither effect yielded major differences in the flowfields.

### *5.1. NACA 0012 Airfoil*

The first test case is flow over the NACA airfoil at a free stream Mach number of 0.7, an angle of attack of 1.49 degrees, and a Reynolds number of 9.0 million. Computations for this case have been computed using a wide range of codes with a compilation of results being reported in reference [25]. The grid used for the present computations consists of 28,126 nodes, of which 202 lie on the airfoil surface. The average normal spacing at the wall for this grid is $2 \times 10^{-6}$.

The convergence history for this case is shown in figure 5 for the computation with the Baldwin-Barth turbulence model both with and without the use of the flux limiter; results obtained with the Spalart-Allmaras model are very similar. Without the limiter, the residual is reduced approximately 4 orders of magnitude in 1500 iterations and the

final lift is obtained in less than 700 iterations. With the use of the limiter, the residual is reduced only one and one-half orders of magnitude before entering into a limit cycle oscillation. The lift, however, exhibits the same convergence behavior as when the limiter is not used.

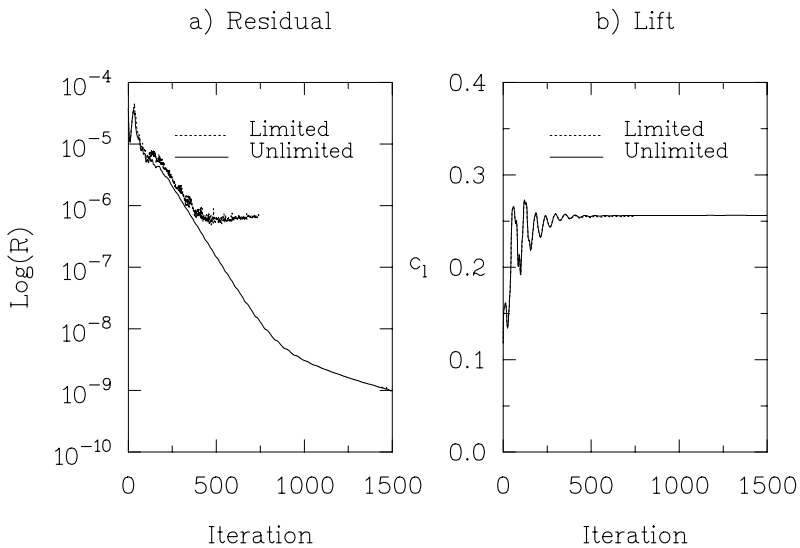


Figure 5. Convergence History for NACA 0012

Since the flux limiter used in the present calculations is non-differentiable and is sensitive to perturbations on the order of truncation error [26], its use yields residuals which are typically reduced only one or two orders of magnitude before entering into a limit cycle oscillation. For cases in which the limiter is used, convergence is determined primarily by monitoring the lift coefficient and other physical parameters such as velocity profiles and skin frictions. While alternate flux limiters exists for structured grids which lessen this tendency, further work is required to extend their use for unstructured grids. Most limiters developed for structured grids limit gradients in a one-dimensional fashion, limiting the gradients normal to a cell face independently in each direction. The task of reconstructing data on cell faces for the current work is currently handled in a more “multidimensional” fashion in which a single gradient for each variable is calculated from adjoining data and is used for extrapolating data to each of the boundaries of the control volume.

A comparison of the computed surface pressure distribution with the experimental data of Harris [27] is shown in figure 6 using both turbulence models. In the figure, the results denoted as B-B have been obtained with the Baldwin-Barth turbulence model while those denoted as S-A are obtained with the Spalart-Allmaras turbulence model. The agreement between the computations and the experiment are good over most of the airfoil although an overexpansion near the leading edge is evident in the computations which does not appear in the experiment. This difference is typical of the results compiled in reference [25] and may be symptomatic of a Mach number or angle-of-attack correction.

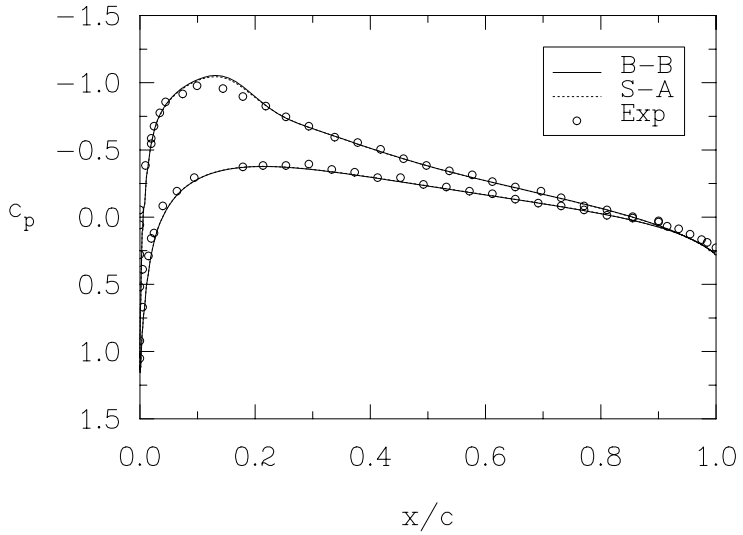


Figure 6. Comparison of Calculated and Experimental Pressure Distribution for NACA 0012 Airfoil

At this same Mach number and Reynolds number, an angle of attack sweep has been conducted to obtain a drag polar. Results are shown in figure 7 comparing the computational and experimental results. The shaded area in the figure indicates the range of results reported in reference [25]. The current results are within the range reported in the reference. Although not directly identified in the figure, the current results closely follow the results from the workshop obtained by both King [28] and Coakley [29] using the Johnson-King turbulence model[30].

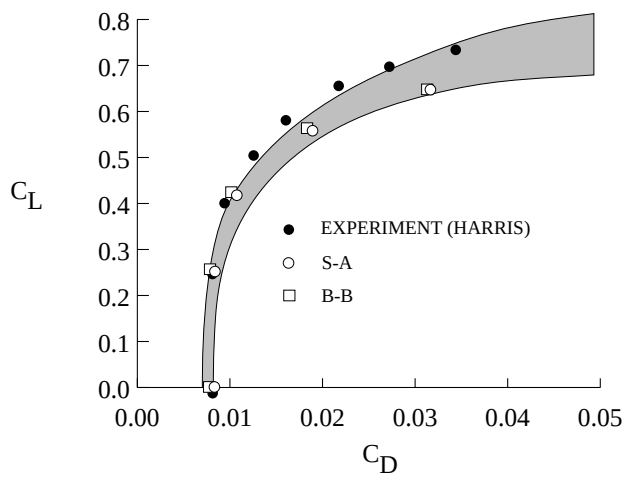


Figure 7. Computed Drag Polar Compared with Experiment and with Results from Reference [25]

### *5.2. RAE 2822 Airfoil*

The next case is flow over the RAE 2822 airfoil at a free stream Mach number of 0.75, an angle of attack of 2.81 degrees, and a Reynolds number of 6.2 million. This corresponds to Case 10 in reference [31]. The grid

used for this case has been obtained using the computer code described in reference [20]. It consists of 13,385 nodes, of which 208 lie on the airfoil surface.

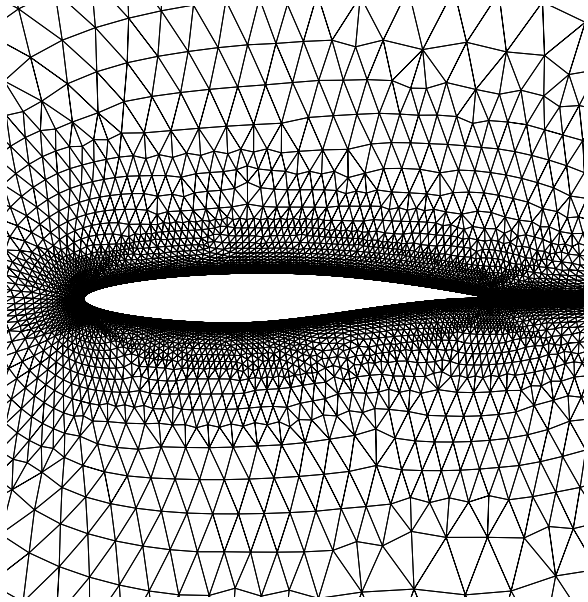

Figure 8. Near-Field View of RAE 2822 Grid

A comparison of the computed surface pressure distribution with experimental data is shown in figure 9. The shock location obtained using both the Spalart-Allmaras and the Baldwin-Barth turbulence models agrees well with the experimental data. Note that experience on structured grids [32], [25] indicates that solutions obtained with an equilibrium model such as Baldwin-Lomax [4] gives a shock location on the upper surface which is significantly aft of the experimental data.

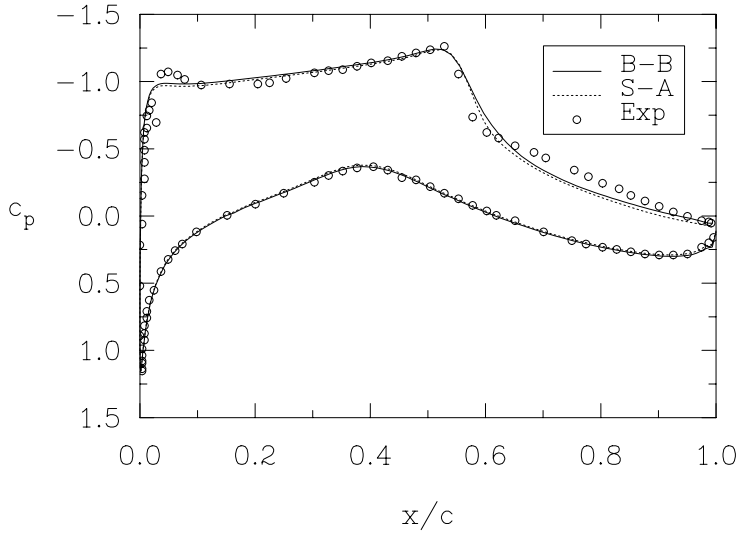


Figure 9. Comparison of Calculated and Experimental Pressure Distribution for RAE 2822 Airfoil

An examination of the skin friction coefficients in figure 10 indicates that the flow separates immediately downstream of the shock. The flow computed with the Spalart-Allmaras model re-attaches downstream of the

shock before separating at the trailing edge whereas the Baldwin-Barth model predicts separated flow all the way to the trailing edge. Note that computations with the Spalart-Allmaras model for this case are reported in reference [14] and indicated that the flow remained separated after the shock. Those calculations, although made on very fine meshes, were not done at the same angle of attack as the present calculation. In reference [14], the computations were made at an angle of attack to give a prescribed lift coefficient of 0.743; the present computations are done at the experimental angle of attack (corrected) and yield a lift coefficient of 0.715 for the Spallart-Allmaras model and 0.744 for the Baldwin-Barth model.

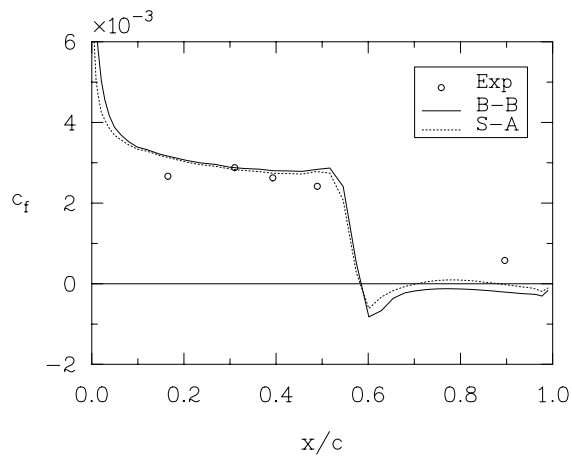


Figure 10. Skin Friction Coefficients for RAE 2822 Airfoil

A comparison of the computed velocity profiles with experimental data at two locations along the chord is shown in figure 11. The velocity profiles at $x/c = 0.75$ are in poor agreement with experimental data. However, in reference [25], many computations for this case are presented using a wide variety of turbulence models which indicate similar inaccuracies. The computed profiles at $x/c = 0.90$ agree significantly better with the experimental data than those at the previous location although the comparisons remain poor very near the wall.

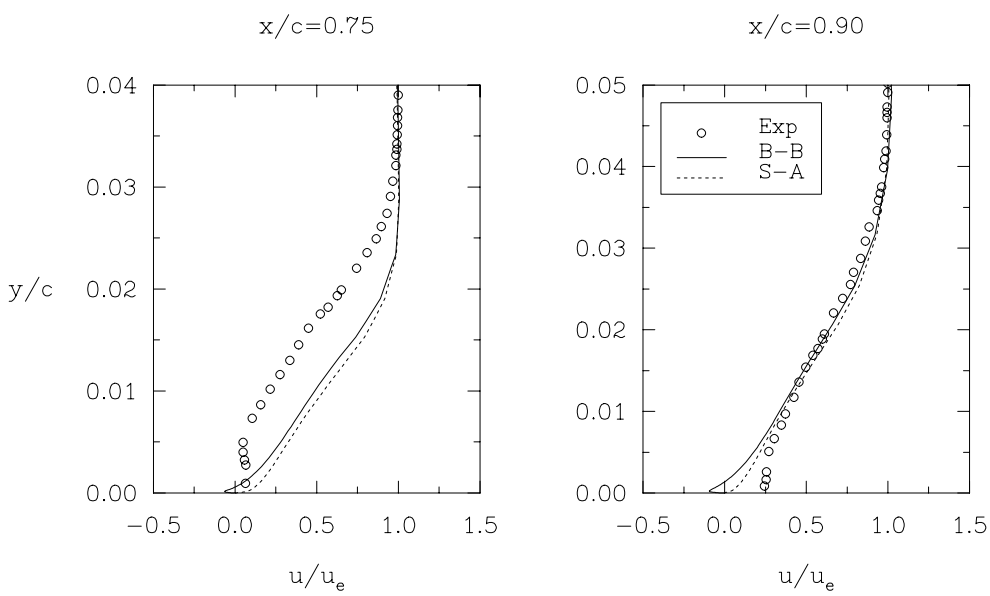


Figure 11. Comparison of Calculated and Experimental Velocity Profiles for RAE 2822 Airfoil

### *5.3. 3–Element Airfoil*

The next case considered is flow over a 3–element airfoil. For this configuration, depicted in figure 12, an extensive amount of experimental data are available [33]. The experiments have been performed in the Low Turbulence Pressure Tunnel (LTPT) located at the NASA Langley Research Center. The experimental data include boundary-layer details such as velocity profiles, as well as pressure distributions over the surface.

Figure 12. Geometry for 3–Element Airfoil

The case chosen for study is at a free stream Mach number of 0.2, an angle of attack of 10 degrees, and a Reynolds number of 3 million; this condition is denoted in reference [33] as Case A. The grid, shown in figure 13, has been generated using the computer code described in reference [20] and consists of 45,902 nodes. The average

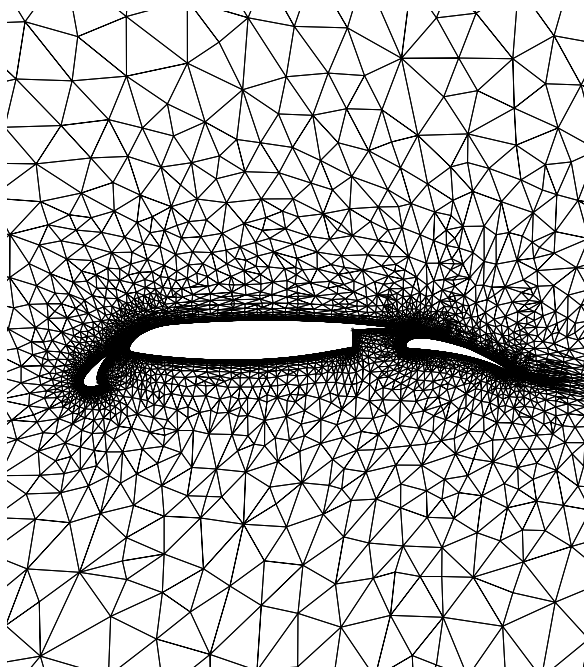

Figure 13. Near View of Grid for 3–Element Airfoil

spacing normal to the body is approximately $2 \times 10^{-5}$ and was determined based on obtaining a $y^+$ of about 2.0 for the point next to the wall at the rear of a unit length flat plate. Note that for this configuration, the trailing edges on each element have a finite thickness as is the tip of the cusped region of the slat where the upper surface and

the cove join. For the calculation, the modelling of these edges has been faithfully reproduced; an example of the grid in the vicinity of the cusp on the slat where the upper surface and cove regions coalesce is shown in figure 14.

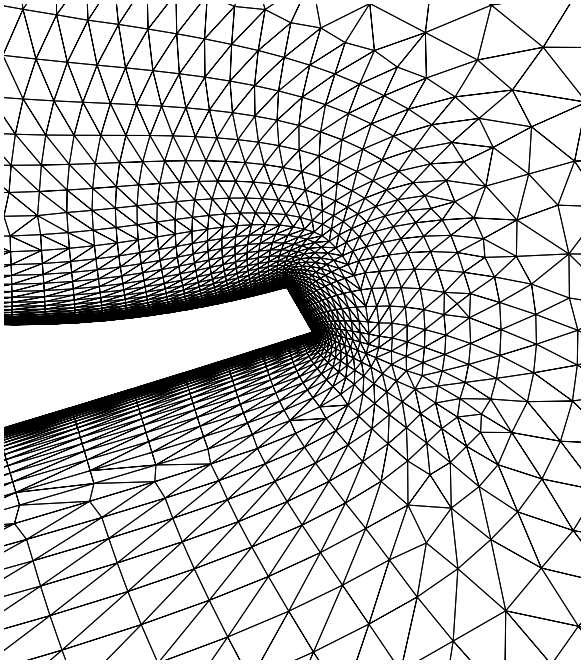

Figure 14. Close-Up View of Grid on Slat Where Upper Surface and Cove Region Coalesce

Although not shown, the calculations with both the Baldwin-Barth and the Spalart-Allmaras turbulence models exhibit a small sinusoidal oscillation in the lift coefficient of about $1\%$. While examination of the flow adjacent to the upper surfaces of each element indicated no unsteadiness, this flow-field contains numerous regions of separated or recirculating flow. For example, it is evident from stream function contours in figure 15 that the flow in the cove regions behind the slat and the main element exhibit large regions of vortical flow. Also, since the trailing edges of each element have finite thickness, small regions of recirculation exist in these areas as well. It should be pointed out that the accuracy of the turbulence models for this type of flow is largely unknown and further studies are required to completely assess their capabilities and range of applicability.

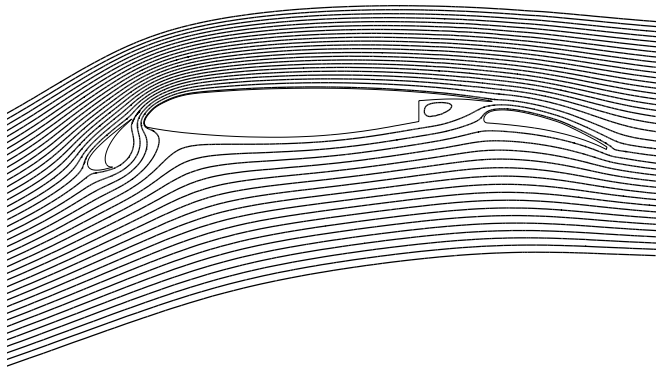

Figure 15. Stream Function Contours for 3–Element Airfoil

Comparisons between the computed data and experimental data are shown in figures 16, 17, and 18. The computed pressure distributions are compared with experiment in figure 16 and the overall agreement is seen to be good over the slat and main element. The comparisons are not as favorable aft of about mid-chord on the flap, although the "hump" in the experimental data is also evident to a lesser extent in the computations.

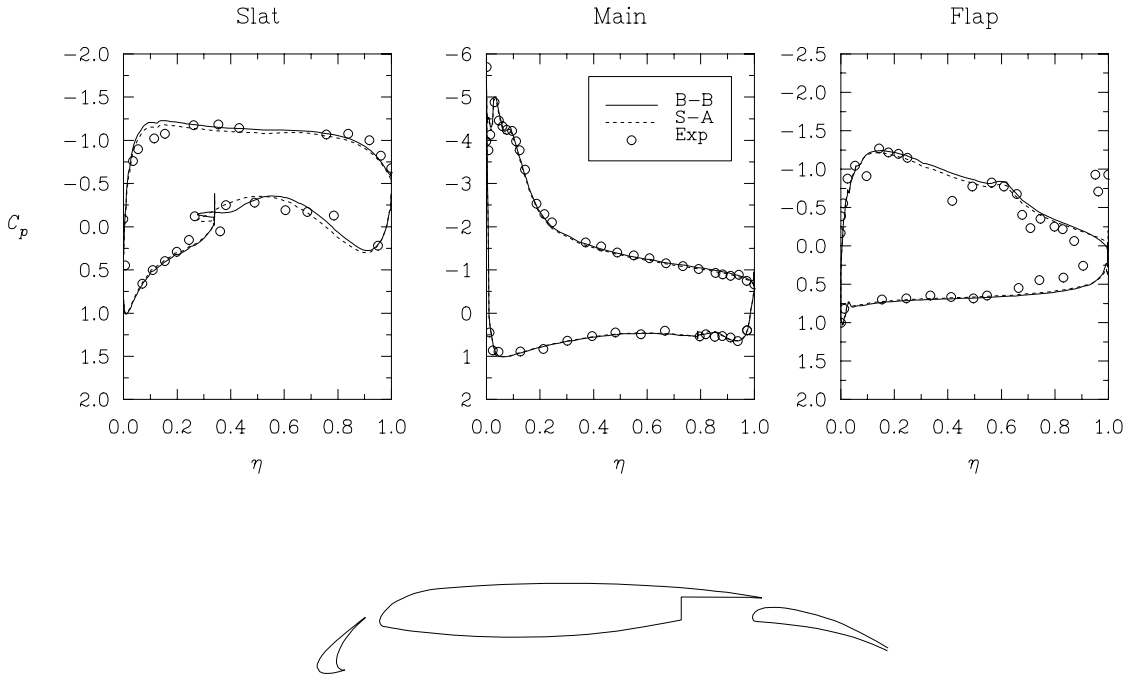


Figure 16. Comparison of Calculated and Experimental Pressures for 3–Element Airfoil

A comparison of velocity profiles at three locations on the airfoil is shown in figure 17. The locations at which the comparisons are made are shown in the figure along with the data. The first station is on the main element behind the slat, while the other two stations are on the flap: one behind the main element and one closer to the trailing edge.

Experimental data are shown which have been obtained using both a hot wire and a flattened Pitot tube. Note that the data obtained with the flattened Pitot tube extend somewhat closer to the surface that the hot wire data. However, as pointed out in reference [33], the data obtained with the flattened tube are not accurate when the pressure in the boundary layer deviates significantly from the pressure at the wall. A sample of experimentally obtained pressure normal to the airfoil is shown in figure 19 for the station on the flap in the near vicinity of the main element. It is apparent that a sharp deviation in the pressure from that at the wall occurs at about $y/c = 0.02$ due to the wake from the main element. Therefore, since the data above this height are not considered to be accurate, they are not shown in the figure. A similar procedure has been followed for each of the other locations on the airfoil. Also note that no velocity profile data on the slat were obtained experimentaly and are therefore not shown.

The comparisons of computed velocity profiles in figure 17 indicate an overall good agreement with the experimental data. However, above the wake from the main element, the experimental data on the flap show a slightly fuller profile than do the computations. A logarithm plot of the same data is shown in figure 18 which greatly magnifies the region near the wall. Again, this figure indicates an overall good agreement with the experimental data.

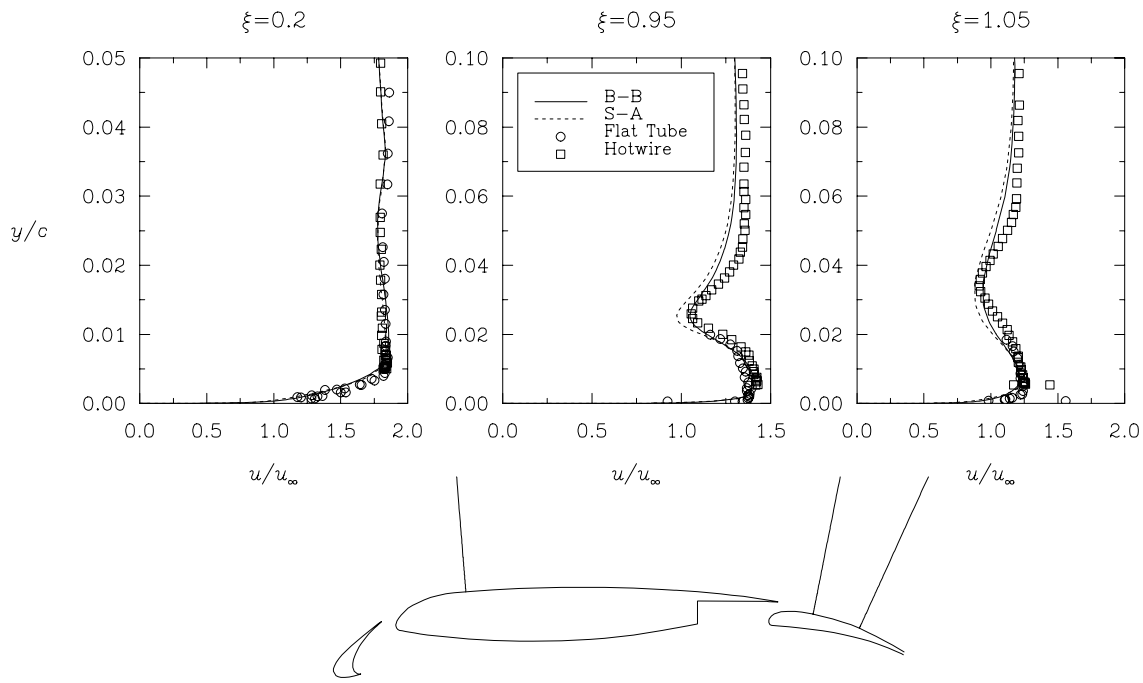


Figure 17. Comparison of Calculated and Experimental Velocity Profiles for 3–Element Airfoil

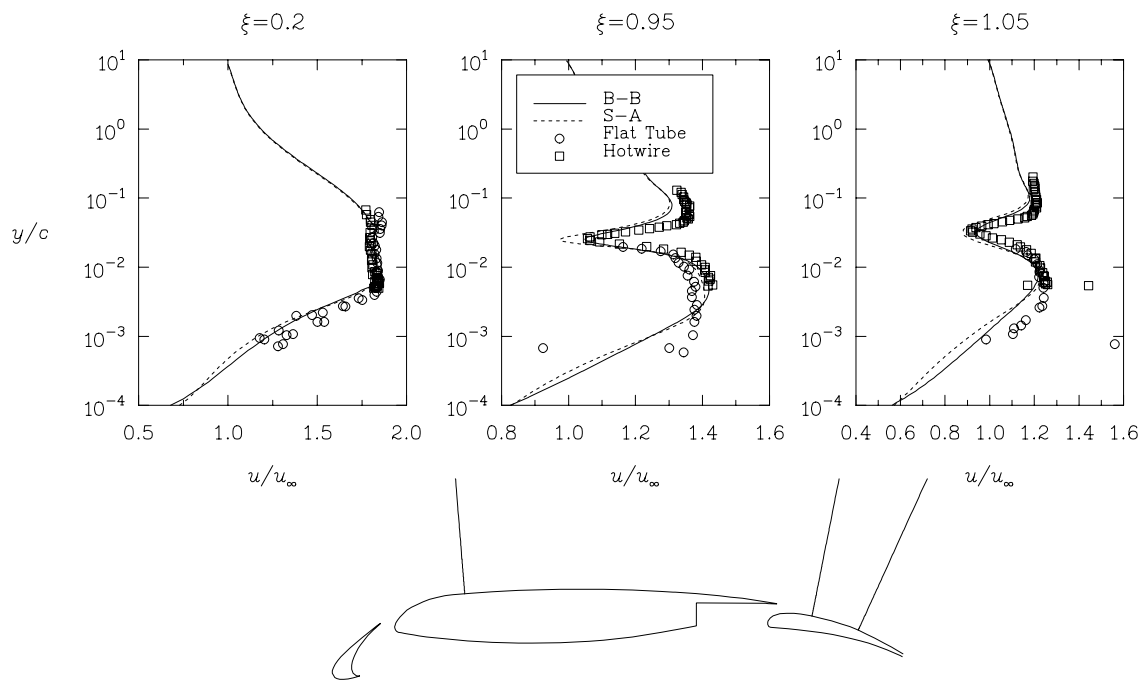


Figure 18. Comparison of Calculated and Experimental Velocity Profiles for 3–Element Airfoil (Logarithmic Coordinates)

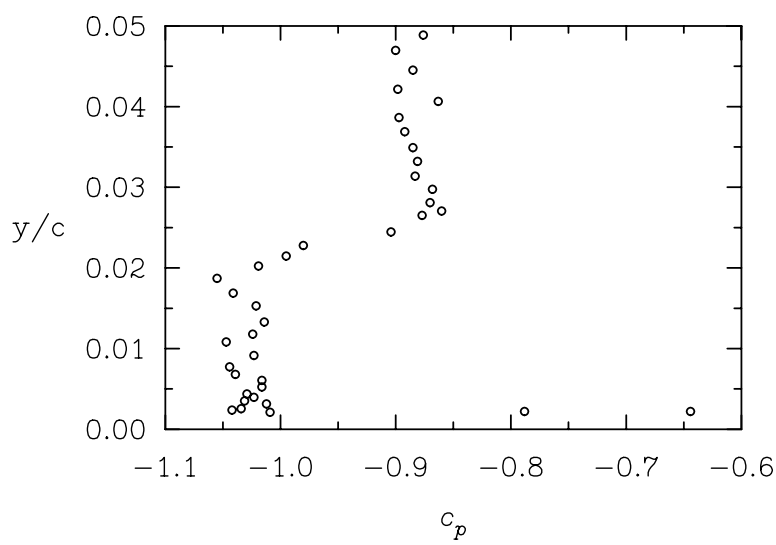


Figure 19. Experimental Pressure Coefficient Through the Boundary Layer at $\xi = 0.95$ on the Flap

### *5.4. 4–Element Airfoil*

The final test case is the flow over a 4–element airfoil which has also been tested in the LTPT [34] and is depicted in figure 20. The conditions given are a Mach number of 0.2, Reynolds number of 9 million, and an angle

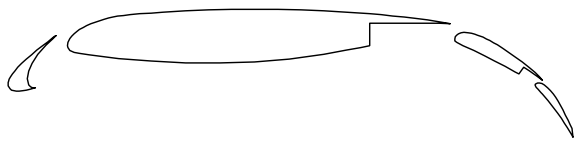

Figure 20. Geometry for 4–Element Airfoil

of attack of 20.318 degrees. Although not shown, this grid has been generated with the same methodology as before. Note, however, that the experimental model did not have finite-thickness trailing edges except for the auxiliary flap. For the computations, all trailing edges are sharp including that on the auxiliary flap which was closed by simply extending the upper and lower surfaces until their coordinates coincided. This grid consists of 59,788 nodes, 1070 of which lie on the surfaces of the airfoil, and has an average minimum spacing at the wall of $2.0 \times 10^{-5}$.

For this computation, solutions with both the Baldwin-Barth and Spalart-Allmaras turbulence model are obtained. The final solution for each is obtained in about 2000 iterations which corresponds to about 3.5 hours of computer time on a Cray YMP. The computed pressure distributions are shown compared to experimental data in figure 21. Although both computations agree reasonably well with experimental data, differences in the pressure distributions are seen over both flaps. Since more detailed data such as velocity profiles are not available for this case, it is not clear which computation is more accurate. Note that although the angle of attack is reasonably high and large vortices exist in the coves behind the slat and underneath the main element, the computed lift coefficients are steady.

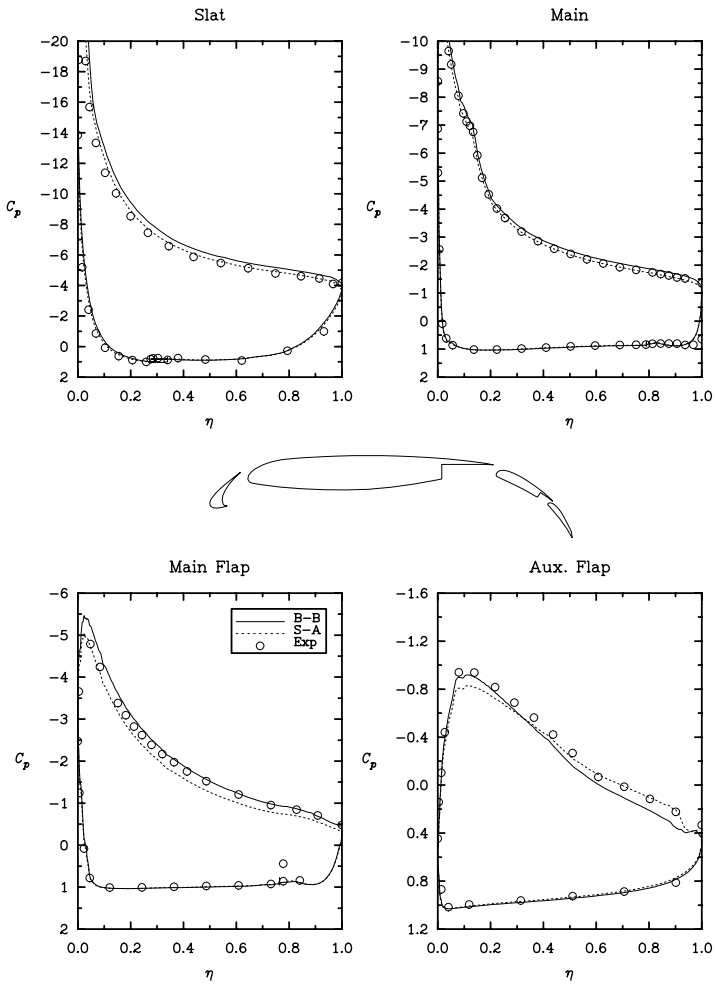


Figure 21. Comparison of Calculated and Experimental Pressure Distributions for 4–Element Airfoil

## 6. Concluding Remarks

A two-dimensional implicit flow solver has been developed for solving the turbulent Navier-Stokes equations on unstructured grids which utilizes the flux-difference-splitting technique of Roe [10]. At each time step, the linear system of equations which arises through linearization of the fluxes is approximately solved with a point-implicit scheme.

Several test cases have been computed and compared with experiment. These cases include subsonic flow over an NACA 0012 airfoil as well as a transonic flow over an airfoil with shock-induced separation. Also included are computations of subsonic flow over two multielement airfoils. The calculations are obtained with two recently developed one–equation turbulence models. Comparisons with experiment include pressure distributions and a limited set of skin frictions and velocity profiles. In general, the comparisons with experiment are good. However, the case with shock-induced separation indicates that although the correct shock location is obtained, the turbulence models remain inadequate for capturing the details of the velocity profiles in the separated region. The cases for the multielement airfoils indicate that good agreement between the computed and experimental pressures is obtained. Comparisons of computational and experimental velocity profiles are made at several stations on the upper surface of a three-element airfoil. At these locations, the flow is attached and the comparisons are good.

## 7. Acknowledgments

The authors would like to acknowledge Timothy J. Barth of NASA Ames Research Center, Christopher L. Rumsey, Jerry C. South and James L. Thomas of NASA Langley Research Center for many useful conversations pertaining to the present work.

# References

1. Karman, S. L., *Development of a 3D Unstructured CFD Method*. PhD thesis, The University of Texas at Arlington, 1991.
2. Marcum, D. L., and Agarwal, R., "A Three-Dimensional Finite Element Navier-Stokes Solver with k-$\epsilon$ Turbulence Model for Unstructured Grids," *AIAA 90–1652*, 1990.
3. Mavriplis, D. J., "Turbulent Flow Calculation Using Unstructured and Adaptive Meshes," *International Journal for Num. Meth. in Fluids*, vol. 13, pp. 1131–1152, 1991.
4. Baldwin, B. S., and Lomax, H., "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows," *AIAA paper 78–257*, 1978.
5. Mavriplis, D. J., and Martinelli, L., "Multigrid Solution of Compressible Turbulent Flow on Unstructured Meshes Using a Two-Equation Model," Tech. Rep. 91–11, ICASE, 1991.
6. Venkatakrishnan, V., and Mavriplis, D. J., "Implicit Solvers for Unstructured Meshes," *AIAA 91–1537CP*, 1991.
7. Saad, Y., and Schultz, M. H., "GMRES: A Generalized Minimum Residual Algorithm for Solving Nonsymetric Linear Systems," *SIAM J. Sci. Stat. Comp.*, vol. 7, no. 3, pp. 856–869, 1986.
8. Bonhaus, D. L., and Wornom, S. F., "Relative Efficiency and Accuracy of Two Navier-Stokes Codes for Simulating Attach Transonic Flow Over Wings," Tech. Rep. TP 3061, NASA, 1991.
9. Jou, W., Wigton, L. B., Allmaras, S. R., Spalart, P. R., and Yu, N. J., "Towards Industrial-Strength Navier-Stokes Codes," *Paper Presented at the Fifth Symposium on Numerical and Physical Aspects of Aerodynamic Flows*, Jan. 1992. Long Beach California.
10. Roe, P., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *J. of Comp. Phys.*, vol. 43, pp. 357–372, 1981.
11. van Leer, B., Thomas, J., Roe, P., and Newsome, R., "A Comparison of Numerical Flux Formulas for the Euler and Navier-Stokes Equations," *AIAA 87–1104*, 1987.
12. Barth, T. J., "Numerical Aspects of Computing Viscous High Reynolds Number Flows on Unstructured Meshes," *AIAA 91–0721*, 1991.
13. Baldwin, B. S., and Barth, T. J., "A One-Equation Turbulence Transport Model for High Reynolds Number Wall Bounded Flows," *NASA Technical Memorandum 102847*, Aug. 1991.
14. Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," *AIAA 92–0439*, 1991.
15. Barth, T., and Jespersen, D., "The Design and Application of Upwind Schemes on Unstructured Meshes," *AIAA 89–0366*, 1989.
16. Golub, G., and VAN LOAN , C., *Matrix Computations*. The Johns Hopkins University Press, 1991.
17. Barth, T. J., "A 3–D Upwind Euler Solver for Unstructured Meshes," *AIAA 91–1548CP*, 1991.
18. Anderson, W. K., "Grid Generation and Flow Solution Method for Euler Equations on Unstructured Grids," Tech. Rep. TM 4295, NASA, 1992.
19. Babuska, I., and Aziz, A. K., "On the Angle Condition in the Finite Element Method," *SIAM J. Numer. Anal.*, vol. 13, pp. 214–226, Apr. 1976.
20. Mavriplis, D., "Adaptive Mesh Generation for Viscous Flows using Delaunay Triangulation," *J. of Comp. Phys.*, vol. 90, pp. 271–291, 1990.

21. Batina, J. T., “Implicit Flux-Split Euler Schemes for Unsteady Aerodynamic Analysis Involving Unstructured Dynamic Meshes,” *AIAA 90–0936*, Apr. 1990.
22. Whitaker, D. L., “Solution Algorithms for the Two-Dimensional Euler Equations on Unstructured Meshes,” *AIAA 90–0697*, 1990.
23. Hageman, L. A., and Young, D. M., *Applied Iterative Methods*. Academic Press, 1981.
24. White, F. M., *Viscous Fluid Flow*. McGraw-Hill Book Company, 1974.
25. Holst, T., “Viscous Transonic Airfoil Workshop Compendium of Results,” *AIAA 87–1460*, 1987.
26. Anderson, W. K., Thomas, J. L., and van Leer, B., “A Comparison of Finite Volume Flux Vector Splittings for the Euler Equations,” *AIAA Journal*, vol. 24, pp. 1453–1460, Sept. 1986.
27. Harris, C., “Two-Dimensional Aerodynamic Characteristics of the NACA 0012 Airfoil in the Langley 8–Foot Transonic Pressure Tunnel,” Tech. Rep. TM 81927, NASA, 1981.
28. King, L. S., “A Comparison of Turbulence Closure Models for Transonic Flows About Airfoils,” *AIAA 87–0418*, Jan. 1987.
29. Coakley, T. J., “Numerical Simulation of Viscous Transonic Airfoil Flows,” *AIAA 87–0416*, Jan. 1987.
30. Johnson, D. A., and King, L. S., “A Mathematically Simple Turbulence Closure Model for Attached and Separated Turbulent Boundary Layers,” *AIAA*, vol. 23, pp. 1684–1692, Nov. 1985.
31. Cook, P., McDonald, M., and Firmin, M., “Airfoil RAE 2822 — Pressure Distributions and Boundary Layer Wake Measurement,” *AGARD AR-138 paper A6*, 1979.
32. Rumsey, C. L., Taylor, S. L., Thomas, J. L., and Anderson, W. K., “Application of an Upwind Navier-Stokes Code to Two-Dimensional Transonic Airfoil Flow,” *AIAA 87–0413*, Jan. 1987.
33. Nakayama, A., “Flowfield Survey Around High-Lift Model LB 546,” *Douglas Aircraft Company Report MDC J4827*, Feb. 1987.
34. Valarezo, W. O., Dominik, C. J., McGhee, R. J., Goodman, W. L., and Paschal, K. B., “Multi-Element Airfoil Optimization for Maximum Lift at High Reynolds Numbers,” *AIAA paper 91–3332–CP*, 1991.