NASA Contractor Report 182035
ICASE Report No. 90–30

ICASE

ALGEBRAIC TURBULENCE MODELING FOR UNSTRUCTURED AND ADAPTIVE MESHES

Dimitri J. Mavriplis

Contract No. NAS1–18605
May 1990

Institute for Computer Applications in Science and Engineering
NASA Langley Research Center
Hampton, Virginia 23665–5225

Operated by the Universities Space Research Association

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665-5225

ALGEBRAIC TURBULENCE MODELING FOR UNSTRUCTURED AND ADAPTIVE MESHES

Dimitri J. Mavriplis

Institute for Computer Applications in Science and Engineering
NASA Langley Research Center
Hampton, VA

ABSTRACT

An algebraic turbulence model based on the Baldwin-Lomax model, has been implemented for use on unstructured grids. The implementation is based on the use of local background structured turbulence meshes. At each time-step, flow variables are interpolated from the unstructured mesh onto the background structured meshes, the turbulence model is executed on these meshes, and the resulting eddy viscosity values are interpolated back to the unstructured mesh. Modifications to the algebraic model were required to enable the treatment of more complicated flows, such as confluent boundary layers and wakes. The model is used in conjunction with an efficient unstructured multigrid finite-element Navier-Stokes solver in order to compute compressible turbulent flows on fully unstructured meshes. Solutions about single and multiple element airfoils are obtained and compared with experimental data.

This research was supported under the National Aeronautics and Space Administration under NASA Contract No. NAS1-18605 while the author was in residence at the Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, Hampton, VA 23665.

1. INTRODUCTION

The use of unstructured mesh techniques for computational fluid dynamics (CFD) problems has become more widespread in recent years, due to the flexibility they afford in discretizing arbitrarily complex geometries, and due to the possibility they offer in resolving highly localized flow phenomena through the use of adaptive meshing. However, research on unstructured mesh techniques for CFD has concentrated almost exclusively on the solution of the Euler equations in two or three dimensions. For viscous flow calculations about non-simple geometries, hybrid meshes have generally been employed [1,2,3] where a thin structured mesh is placed in the boundary-layer and wake regions, and an unstructured mesh is constructed in the outer inviscid region of the flow-field. Besides leading to an increase in coding complexity, this type of compromise limits the generality of the unstructured mesh approach in dealing with arbitrarily complex geometries, such as multiple body geometries with close tolerances, where confluent boundary layers or wakes may occur, and complicates the task of performing adaptive meshing in the inviscid as well as viscous regions of flow. It appears that the difficulties associated with generating highly stretched unstructured meshes, which are required for efficiently resolving viscous shear layers, as well as the efficient implementation of a turbulence model on such meshes, has generally impeded the use of fully unstructured meshes for viscous flows. The use of unstructured meshes throughout the entire flow-field is advocated in the present work. Previous work by the author has shown how a highly stretched unstructured mesh, suitable for high-Reynolds-number viscous flow calculations may be constructed [4], and has also discussed the development of an efficient unstructured Navier-Stokes solver for laminar flows [5]. This paper is thus concerned with the efficient implementation of a turbulence model for computing high-Reynolds-number turbulent flows using fully unstructured meshes.

The most widespread turbulence models in use currently are either of the multiple field-equation type, or of the algebraic type. Field-equation turbulence models (such as the K-ε model) are in principle more general than their algebraic counterparts, and appear well suited for use on unstructured meshes; the additional field-equations may be discretized and solved on the unstructured mesh in the same fashion as the governing flow equations. However, the solution of additional field-equations can be considerably expensive, especially in the thin boundary-layer regions near the wall, where the equations can become considerably stiff. Algebraic turbulence models, on the other hand, are relatively inexpensive to compute, and have demonstrated generally superior accuracy and reliability for limited classes of problems, such as high-Reynolds-number attached flows over streamlined bodies. However, such models typically require information concerning the distance of each mesh point from the nearest wall. Turbulence length scales, which are related to the local boundary-layer or wake thickness, are determined by scanning the appropriate flow values along specified streamwise stations. In the context of unstructured meshes, mesh points do not naturally occur at regular streamwise locations. Hence, the implementation of algebraic models on such meshes is not entirely straightforward. Davis and Dannenhoffer [6] have implemented an algebraic model for use on heavily adapted quadrilateral meshes. The meshes are semi-structured in nature, and only a subset of the wall boundary points can be identified with mesh lines spanning the entire shear layer. A method for constructing turbulence quantities at all points is employed, which makes use of flow variables interpolated from neighboring mesh lines. Kallinderis [7] employs a similar approach and describes how the process may be extended to unstructured triangular meshes. The first successful implementation of an algebraic turbulence model on unstructured meshes,

however, appears to be due to Rostand [8]. In his work, additional mesh lines normal to the wall, emanating from each boundary point, are constructed. The unstructured mesh flow variables are interpolated onto these lines, and the algebraic turbulence model is executed on each normal mesh line. Rostand's work, however, is confined to supersonic flows over ramp geometries, and thus lacks the generality required for more complex geometries.

To construct a turbulence model suitable for use on unstructured meshes, one must ensure that the inherent capabilities and advantages of unstructured mesh techniques are not hindered by the implementation of the turbulence model. Hence, an algebraic turbulence model capable of dealing with arbitrarily complex geometries and amenable to adaptive meshing techniques must be devised. Furthermore, the overhead incurred by the turbulence modeling routine must represent a small fraction of the overall computational effort, in order to ensure a competitive solution procedure. The contributions of this work are twofold. In a first part, a method for efficiently implementing an arbitrary equilibrium algebraic turbulence model on unstructured meshes is described. In a second part, modifications to the Baldwin-Lomax model [9], which enable the treatment of multiple shear layers, such as boundary-layer wake interactions, are described.

2. IMPLEMENTATIONAL PROCEDURE

The general procedure employed for implementing an algebraic turbulence model on an unstructured mesh consists of constructing local structured meshes about each geometry component or each wall boundary. Each time the turbulence routine is called, the current flow variables are interpolated from the unstructured mesh onto the multiple background structured turbulence meshes. The algebraic turbulence model is executed on these meshes, and the resulting eddy viscosity values are interpolated back to the unstructured mesh, for subsequent use in the flow solution phase. The local structured meshes are constructed using a hyperbolic mesh generator [10], which uses, as its initial condition, the boundary point distribution of the unstructured mesh on the geometry component being considered. The use of local hyperbolically generated meshes is akin to the construction of normal mesh lines emanating from each boundary point, as originally employed by Rostand [8], but with an additional degree of sophistication which prevents the cross-over of normal mesh lines in the vicinity of concave boundaries, ensures a smooth variation of mesh points in the normal direction, and enables the handling of more complex multiple-body geometries.

2.1. Construction of Local Turbulence Meshes

Local structured turbulence meshes are constructed using a hyperbolic mesh generator [10]. Hyperbolic mesh generators required an initial boundary point distribution, and a distribution of normal mesh spacings. For wall boundaries, the boundary points of the unstructured mesh are employed as the initial condition for the hyperbolic mesh generator. A normal mesh spacing is specified, which closely approximates the spacing of the unstructured mesh in the region near the wall. Thus, a close matching between the resolution of the local structured meshes and the unstructured mesh is achieved, and the local meshes share the same boundary points as the global unstructured mesh. For modeling turbulent wakes, a fictitious wake line may be drawn, which approximates the expected position of the computed wake. Points are then placed along this line, and a hyperbolic mesh is generated on either side of the wake line, closely approximating the local resolution of the underlying unstructured mesh. In the present work, which has been mainly concerned with multi-element airfoil geometries, a C-type hyperbolic mesh is generated about each airfoil element, which is used for the turbulence modeling

in the airfoil wall region as well as in the wake region. The wake lines are predetermined by solving the inviscid flow over the entire configuration using a panel method [11]. For multiple body geometries, structured turbulence mesh lines are terminated if they intersect a neighboring geometry component. In this manner, the eddy viscosity in any region of the flow-field is only determined by the turbulence stations emanating from boundary or wake points which are directly visible from that location (c.f. Figures 12 and 18).

2.2. Interpolation Procedure

In order to execute the turbulence model, flow variables must be interpolated from the unstructured mesh onto the local turbulence meshes, and the resulting eddy viscosity values must be interpolated back onto the unstructured mesh. Since linear interpolation is most easily performed using triangular elements, the local turbulence meshes are triangulated. The simplest way of triangulating a structured mesh is to subdivide each quadrilateral into two triangles. However, in anticipation of the subsequent use of adaptive meshing techniques, a Delaunay triangulation of the set of points constituting each local turbulence mesh is more desirable. Hence, after the initial triangulation of these meshes is performed, the edges are swapped using the edge swapping algorithm described in [4], according to the modified Delaunay criterion, in order to obtain a Delaunay triangulation.

The patterns for transferring variables back and forth between the global unstructured mesh and the local turbulence meshes are then determined using an efficient tree-search routine. This operation is similar to that described in [12], for transferring variables between sequences of unstructured triangular meshes, in the context of a multigrid algorithm. Using neighbor information, this type of search is capable of determining the enclosing triangle on one grid for each point of another grid. Once the enclosing triangle of a given point is known, the interpolation coefficients can be determined from geometrical considerations. The entire process for all grid points can be performed in 0(N) operations, where N is the total number of grid points. The process is performed once, as a preprocessing operation, and the transfer addresses and coefficients are stored for subsequent use in the flow solution phase.

Since each background turbulence mesh only covers a portion of the domain spanned by the global unstructured mesh, not all unstructured mesh points will be contained in some particular triangle of the local background meshes. To avoid failure of the tree-search algorithm, points which lie in the region not covered by the background meshes must first be determined, and omitted from the search in the interpolation routine. Furthermore, since two or more background turbulence meshes may overlap in various regions of the flow field, unstructured grid nodes may receive multiple eddy viscosity values, one from each of the overlapping turbulence meshes in that region. The final eddy viscosity value taken at such points consists of a weighted average of the multiple interpolated values, where each of the values is weighted by a factor proportional to the inverse of the distance between that point and the respective boundary point of the corresponding turbulence mesh station. This provides for a smooth distribution of eddy viscosity in regions such as between two neighboring walls, or between a wall and a neighboring wake line.

2.3. Adaptive Meshing Capability

One of the major advantages afforded by the use of unstructured meshes is the ability to easily perform adaptive meshing. The most efficient adaptive techniques are based on local mesh enrichment and restructuring, rather than global mesh regeneration. Thus, the current turbulence model implementation must be compatible with such adaptive meshing strategies. This

can be achieved by locally refining each background turbulence mesh at each stage when the global unstructured mesh is adapted, in such a manner that the resolution of the background turbulence meshes closely follows the evolving resolution of the adapted global unstructured mesh. For example, assuming an unstructured mesh and a set of background turbulence meshes have been constructed, and the turbulent flow solved for on these meshes, a new global unstructured mesh may be constructed by adding points to the existing mesh in regions of high flow gradients, and locally retriangulating [4]. A refinement field vector may be constructed, prior to the retriangulation step, by assigning the value 1.0 to each vertex of a triangle or edge which is to be refined, and the value 0.0 to all other nodes. This so-called refinement field vector can now be interpolated onto the background turbulence meshes and used to determine the regions of the turbulence meshes which require refinement. These background turbulence meshes may thus, in turn, be adaptively refined, but only in such a way as to preserve their original structure, i.e., as a set of normal mesh lines emanating from boundary points. Thus, the interpolated refinement field values are scanned along each normal mesh line and new points are added along the mesh line in regions where the refinement field values approach unity, thus increasing the normal resolution of the existing turbulence mesh lines. In regions where new unstructured mesh boundary points have been added, a new turbulence mesh station is constructed, which extends from the new boundary point out to the outer boundary of the local turbulence mesh. The normal distribution of points along this new station is determined by averaging the location of points from the two neighboring mesh stations. By allowing for only these two types of refinement, normal refinement of existing stations, and the generation of entire new stations, the structure of the turbulence meshes is preserved. However, the requirement of generating new stations which extend out to the outer boundary of the turbulence meshes may have the effect of adding mesh points to the turbulence meshes in regions where the underlying unstructured mesh is not refined, and may result in a more rapid growth of the number of turbulence mesh points than the unstructured mesh points. However, this has not been found to be a problem generally, probably due to the fact that the normal resolution of the turbulence meshes is very sparse in the far-field. Since the background turbulence meshes have previously been triangulated, the new adaptively refined turbulence meshes may be constructed by inserting each new mesh point into the existing triangulation and restructuring locally using Bowyer's Delaunay triangulation algorithm [4], thus generating the Delaunay triangulation of the new set of points. The interpolation patterns for transferring variables back and forth between the newly adapted global mesh and the new local turbulence meshes are then recomputed, and the flow solution process is resumed. It should be noted that the initial and subsequent adaptive triangulation of the turbulence meshes is only required for the determination of the interpolation transfer patterns, and is not required for the actual execution of the turbulence model.

2.4. Data Structures

The efficient implementation of the present turbulence model on unstructured meshes depends heavily on the use of adequate preprocessing of the turbulence mesh quantities, and the use of suitable data structures for storing and accessing the relevant information. Once a global unstructured mesh has been generated, the local background turbulence meshes are automatically generated from the boundary point distribution of the unstructured mesh. These meshes are then arranged into stations, by constructing a list of points for each normal turbulence mesh line, augmented by some additional directives to be employed in the turbulence model. The transfer coefficients and addresses for interpolating back and forth between the

global unstructured mesh and the local turbulence meshes are then computed and stored. This point marks the end of the preprocessing stage, as all the information required for the flow solver and turbulence model is presently at hand. This information is then dumped to a file which is used as input to the flow solver. Thus a turbulent flow mesh file consists of the following information:

1) Connectivity of the global unstructured mesh.
2) List of global unstructured mesh and coordinates.
3) List of the turbulence mesh normal stations, with each station pointing to the nodes which constitute that station, as well as several turbulence modeling directives particular to that station.
4) List of turbulence mesh nodes and their coordinates.
5) Interpolation coefficients and addresses for transfer of variables back and forth between the global unstructured mesh and the local turbulent meshes.

At this point, the information no longer resembles a series of structured local turbulence meshes with an overlaid global unstructured mesh. In fact the connectivity of the turbulence meshes is not stored, and the turbulence mesh points and stations are not associated with any particular turbulence mesh, nor are they ordered in any regular fashion. This constitutes the minimum amount of information required for executing the flow solver and turbulence model, and as such can be viewed as the definition of a preprocessed unstructured turbulent-flow mesh.

The data structure employed for the turbulence mesh stations is depicted in Figure 1. Since algebraic models are in general equilibrium turbulence models, the stations may appear in random order and no information concerning neighboring stations is required. Each station contains an integer list, which is dimensioned as JLMAX+3, where JLMAX represents the maximum number of mesh points in any given station. The first JL entries in this list point to the addresses of the mesh points which constitute the turbulence mesh station. This list is ordered, beginning with the wall or wake boundary point, and terminating with the outer boundary point. The JLMAX+1 entry contains the number of points JL for that station. Hence, when JL is less than JLMAX, empty entries occur in the list. The JLMAX+2 entry indicates to the turbulence model whether transition has occurred. For turbulence flow this entry is set to 1, but in the laminar flow region, prior to transition, is set to 0. This entry is generally determined manually, for cases where forced transition is desired. The final JLMAX+3 entry indicates the presence of a wall station (= 0) or a wake station. In the later case, this entry points to the address of the station which lies directly on the opposite side of the wake line, such that the two stations may be paired together to form a complete wake cut in the turbulence modeling routine.

When adaptive meshing is to be employed, additional information is required concerning the triangular connectivity of the turbulence meshes. This information is thus appended to the turbulence flow mesh file, although it is not used by the flow solver.

3. DESCRIPTION OF THE ALGEBRAIC TURBULENCE MODEL

In the previous section, a method for implementing an arbitrary algebraic-type turbulence model on unstructured grids has been described. The specifics of the algebraic model employed in this work will now be described. The model adopted is based on the Baldwin-Lomax model [9], which is a two-layer algebraic model. The inner-layer eddy viscosity is computed as

$$(\mu_T)_{inner} = \rho\, l^2\, |\omega|$$

where ρ is the fluid density, ω the vorticity, and l is a length scale which is proportional to the distance from the wall (scaled by a damping factor). In the outer region, the eddy viscosity is given by

$$(\mu_T)_{outer} = K\, \rho\, F_{WAKE}\, F_{KLEB}(y)$$

where K is a constant and

$$F_{WAKE} = function\ of\ (y_{MAX}, F_{MAX})$$

$$F_{KLEB} = function\ of\ (\frac{y_{MAX}}{y})$$

y is the distance from the wall, or wake centerline, and F is proportional to the moment of vorticity:

$$F = y\, |\omega| * damping\ factor$$

F_{MAX} thus represents the maximum value of F along a given turbulence station, and y_{MAX} is the y-distance at which this maximum occurs. The turbulence length scales are thus determined by l in the inner layer, and y_{MAX} in the outer layer. For fully attached or midly separated flows over simple geometries, a single well defined peak exists in the function F, along given streamwise stations. However, for flows over more complex configurations, where boundary layers and wakes may interact and larger separation regions may occur, the function F may exhibit multiple local maxima, and various methods for determining the appropriate length scales have been proposed [13,14]. In general, it is found that a distinction needs to be made between wall turbulence and wake turbulence.

3.1. Wall Turbulence

The Baldwin-Lomax model performs adequately for simple turbulent wall flows. However, in cases where additional wakes or neighboring wall boundary layers are present, such that these structures are traversed by a wall turbulence mesh station, multiple peaks in the F function are observed. Figure 2 depicts the case where a neighboring wake is traversed by a wall turbulence mesh station. Since the vorticity becomes non-zero and y, which is measured from the wall, is large in the wake region, the moment of vorticity exhibits a large secondary peak in this region. Selection of this peak leads to an inappropriate length scale. The proper length scale is that associated with the primary peak of F, which is located in the region of the wall boundary layer. Thus, the search for the maximum value of F must be limited to this region. For an isolated wall boundary layer, the vorticity is largest at the wall, and vanishes monotonically and asymptotically as the far-field is approached. When a neighboring wake or boundary layer is approached, the vorticity becomes negative, as the velocity begins to decrease near the edge of this shear layer. Thus, by locating the zeros of the vorticity distribution along a given station, and limiting the search for the peak of F to the region between the wall and the first zero in the vorticity, the appropriate length scale is obtained.

3.2. Wake Turbulence

In order to compute wake turbulence length scales, the location of the wake centerline must first be determined. This cannot in general be assumed to coincide with the turbulence-mesh wake-line boundary, since these merely represent initial guesses to the actual wake locations, provided in this case by a panel method solution. Since length scales are computed as distances away from the wake centerline, these can be significantly misrepresented if the proper wake centerline location is not employed. To locate the wake centerline and compute turbulent wake length-scale quantities, a single station traversing the entire wake is required, rather than

two individual stations, each on opposite sides of the wake line. Thus, the two corresponding stations on either side of the wake line are loaded into a single temporary array, which represents a complete station traversing the entire wake. The wake centerline corresponds to the location of minimum (local) velocity or zero vorticity as shown in Figure 2. Thus the zeros of the vorticity distribution along the cut are located, and the zero closest to the turbulence mesh wake-line is identified as the location of the wake centerline. The two neighboring zeros of the vorticity, one on each side of the centerline, are then employed to limit the search for the maximum of F, the moment of vorticity, in the wake region, as described in the previous section.

3.3. Limitations

These modifications to the standard model enable more complex geometries to be handled, and in principle, enable the treatment of confluent or merging boundary layers and wakes. For example, as a boundary layer and a wake gradually merge, as shown in Figure 2, the location of the zeros of the vorticity will be shifted, thus enabling the turbulence model to track the process. However, the success of this method rests on the assumption of a relatively smooth vorticity distribution and on the ability to generate a reasonable estimate of the location of the wake centerline. For example, if the turbulence mesh wake-line does not fall in the same vicinity as the actual wake centerline, then the vorticity zero closest to the mesh wake line may not correspond to the location of the wake centerline. Furthermore, since vorticity represents a difference in velocity, it tends to be somewhat noisy and may exhibit rather large spurious oscillations. Thus, a filtering technique is employed to smooth the vorticity distribution and remove any spurious oscillations. In the present work, two passes of a simple Laplacian smoothing operator are applied along each turbulence mesh station to filter the vorticity distribution. More sophisticated smoothing techniques based on Fourier methods may also be experimented with in the future. However, it is important to realize that filtering is only applied to the vorticity distribution in order to locate the zeros of the distribution, which in turn determine the extent of the search regions, and the location of the wake centerline. The unsmoothed vorticity values are employed in the calculation of all other turbulence modeling quantities.

4. RESULTS

The present algebraic turbulence model implementation is used in conjunction with the unstructured multigrid Navier-Stokes solver, previously described in [5], to compute the steady turbulent compressible flow over single and multiple-element airfoil geometries. In the context of a multigrid strategy, the turbulence model is only executed on the finest mesh of the sequence, and thus only background turbulence meshes corresponding to the finest unstructured mesh need be constructed. Within each multigrid cycle, the turbulence modeling routine is executed on the finest grid, and the resulting eddy viscosities are interpolated up to the coarser grids, where they are used in the multigrid correction equations. The whole process is very efficient, and in general, the entire turbulent modeling routine, including the interpolation procedures, requires only 10% of the total time within a multigrid cycle. Memory requirements are, however, increased by about 50% since extra variables and transfer coefficients must be stored for the turbulence mesh stations. When adaptive meshing is employed, new finer unstructured meshes are generated by adding new points and locally restructuring the previous coarser mesh. These new meshes are then added to the multigrid sequence. The background turbulence meshes corresponding to this new finer level are generated by adaptively refining

and preprocessing the background meshes from the previous level, which themselves are now discarded.

4.1. Single-Element Airfoil Results

As an initial test case, the turbulent flow over an RAE 2822 airfoil has been computed. The freestream Mach number is 0.729, the Reynolds number is 6.5 million, the corrected incidence is 2.31 degrees, and transition is fixed at 0.03 chords. This constitutes a well documented test case (Case 6) for turbulent transonic flow [15], which can be used to validate the present solver. The unstructured mesh employed for this case is depicted in Figure 3. This mesh contains 13,751 points of which 210 are on the airfoil surface. The average normal spacings of the triangles on the airfoil surface is 0.00001 chords, resulting in cell aspect ratios of the order of 1000:1 near the wall. The background turbulence mesh stations employed for computing the algebraic turbulence model, which contain a total of 13,372 points, are depicted in Figure 4. The computed surface pressure distribution and skin friction distribution are displayed in Figures 5 and 6, respectively, where they are compared with experimental data from [15]. Both quantities are seen to compare favorably with the experimental results, and the computed lift coefficient of 0.7403 is well within the range reported in previously published computational solutions, using structured meshes [16]. A total of five meshes were employed in the multigrid sequence, with the coarsest mesh containing only 98 points. The convergence rate for this case, as measured by the decrease in the RMS average of the density residuals throughout the flow-field, versus the number of multigrid cycles, is depicted in Figure 7. An average residual reduction of 0.955 per multigrid cycle is achieved on the finest grid, resulting in a decrease of the residuals by 4 orders of magnitude over 200 cycles. Furthermore, the lift and drag coefficients were converged to four significant figures within 90 cycles. Since each multigrid cycle requires roughly 1.4 CPU seconds on a single processor of the CRAY-YMP computer, engineering solutions could thus be obtained in approximately 2 minutes for this case.

To demonstrate the advantages offered by adaptive meshing techniques, the same test case has been recomputed using a sequence of adaptively generated meshes. The first four meshes are identical to the four coarse meshes employed previously. The final two meshes are obtained by successive adaptive refinements of the previous coarser mesh. The criterion for adaptive refinement is based on a combination of the undivided difference of Mach number, and the undivided difference in pressure. When the difference of one of these variables along a particular mesh edge is found to be larger than the average of the differences across all mesh edges, a new point is added midway along that edge. The finest adaptive mesh is shown in Figure 8. This mesh contains 12,829 points, slightly less than the number of points of the mesh of Figure 3, but exhibits twice the maximum resolution of the non-adapted mesh. The newly adapted background turbulence mesh stations corresponding to this refined mesh are depicted in Figure 9. A total of 13,011 mesh points are now contained in these stations. The Mach contours of the solution computed on the unstructured mesh of Figure 8, using the corresponding turbulence mesh stations of Figure 9, are shown in Figure 10, exhibiting a sharper shock and boundary layer resolution using fewer mesh points than the previously non-adapted solution. A total of six meshes (including two adaptive refinements) have been used in the multigrid sequence to compute this case, yielding a fine grid convergence rate roughly equivalent to that observed for the non-adapted case in Figure 7.

4.2. Two-Element Airfoil Results

In the next test case, transonic flow over a two-element airfoil has been computed. The configuration consists of a main airfoil with a leading edge slat. The freestream Mach number is 0.5, the Reynolds number is 4.5 million, and the incidence is 7.5 degrees. A sequence of five unstructured meshes was employed for this case, with the finest mesh containing 28,871 points. The main airfoil and the slat contain 208 and 228 surface points respectively. The normal height of the triangles at the wall is 0.00002 chords, based on the chord of each individual airfoil. In figure 11, a global view of the second finest mesh (7,272 points), and a close-up view of the finest mesh are shown. A global view of the finest mesh is omitted, due to the difficulties associated with plotting such dense grids. The background turbulence meshes corresponding to the finest unstructured mesh level contain a total of 28,256 points. In Figure 12, the background turbulence meshes corresponding to the second finest level (7,415 turbulence mesh points), which exhibit the same topology as the finer level meshes, are shown, prior to the triangulation operation. The computed Mach contours for this case are depicted in Figure 13. At these conditions, the flow is supercritical and a shock is formed on the upper surface of the slat, as seen in the figure. The slat boundary layer thickens appreciably as it passes through the shock, and a small region of separated flow is formed at the foot of the shock. The wakes from both airfoils appear to be resolved reasonably well in the present calculation. The flow-field distribution of eddy viscosity is illustrated in the contour plot of Figure 14. The turbulence model is seen to yield a smooth distribution of eddy viscosity in the gap region, and between the main airfoil upper surface, and the wake of the slat, where the two background turbulence meshes overlap, and where these two shear layers merge. The computed surface pressure distribution for this case is compared with experimental wind-tunnel data [17], in Figure 15. Good overall agreement is observed, including the prediction of the height of the suction peaks and the shock strength and location. The convergence history for this case is shown in Figure 16, where the fine grid residuals were reduced by 5 orders of magnitude in 350 multigrid cycles, resulting in an average residual reduction rate of 0.968 per multigrid cycle. The lift and drag coefficients could be converged to four significant figures in approximately 75 cycles, requiring roughly 3.5 CPU minutes on a single processor of the CRAY-YMP.

4.3. Four-Element Airfoil Case

The final test case consists of a four-element airfoil configuration. This represents a truly complex geometry which is not easily amenable to structured mesh techniques and is of considerable practical interest, as it relates to the design of high-lift devices for commercial aircraft. This particular configuration has been the subject of extensive wind-tunnel testing [18], and thus provides a suitable code validation test case. A multigrid sequence of five meshes was employed to compute the flow about this configuration. The finest mesh of the sequence contains 62,076 points of which 294 are on the surface of the slat, 211 on the surface of the main airfoil, and 254 on both the vane and the trailing edge flap. The average width of the elements at the wall is 0.00002 chords for each airfoil, resulting in cell aspect ratios of the order of 1000:1 in these regions. The background turbulence meshes are based on the same boundary point resolution as the global unstructured mesh, and contain a total of 56,168 points. Figure 17 provides a global view of the coarser level unstructured mesh (15,896 points), and a close-up view in the region of the leading-edge slat of the finest mesh level. The background turbulence meshes corresponding to the coarser level (16,886 turbulence mesh points) are depicted in Figure 18, prior to the triangulation and preprocessing operations. For this case, the

freestream Mach number is 0.2, the Reynolds number is 2.83 million, based on the chord of the nested flap configuration, and the incidence is 8.18 degrees. At these conditions, the flow is entirely subcritical, although compressibility effects remain important, since local Mach numbers greater than 0.6 are achieved in the suction peaks. The computed Mach contours are shown in Figure 19. The flow is mostly attached and a good resolution of the boundary layers and wakes is achieved about all four airfoil elements. A comparison of the computed surface pressure distribution with the experimental wind-tunnel data is given in Figure 20. Computed and experimental values are seen to agree favorably in all regions, demonstrating a good prediction of the suction peaks and lift on all airfoil elements. This solution required roughly 14 Mwords of memory and 15 minutes of CPU time on a single processor of the CRAY-YMP, which corresponds to 150 multigrid cycles on the finest grid, during which the residuals were reduced by approximately two and a half orders of magnitude.

5. CONCLUSIONS

An algebraic turbulence model, suitable for non-simple flows and geometries, has been implemented successfully for use on unstructured and adaptive meshes. By combining a highly-stretched unstructured-mesh generation method, a multigrid finite-element Navier-Stokes solver, and the present algebraic turbulence model, turbulent flow fields may be computed using fully unstructured meshes. The method is efficient in that solutions may be obtained using on the order of 100 multigrid cycles, and the turbulence model consumes less than 10% of the total time required to compute a solution. In order to accommodate complex flow situations, such as confluent boundary layers and wakes, modifications to the basic algebraic model have been devised. These, however, are not foolproof, and further work in this area may be required to increase the reliability of the model. For more complex flows, and flows with massive separation, multiple field-equation turbulence models may appear to be more suitable. However, much research remains to be done before physically accurate as well as numerically efficient field models can be routinely employed.

REFERENCES

1. Nakahashi, N., "FDM-FEM Zonal Approach for Viscous Flow Computations Over Multiple Bodies", *AIAA paper 87-0604*, January, 1987.

2. Holmes D. G., and Connell, S., "Solution of the 2-D Navier-Stokes Equations on Unstructured Adaptive Grids", *AIAA paper 89-1932, Proc. of the AIAA 9th Computational Fluid Dynamics Conference, Buffalo, NY*, June, 1989.

3. Thareja, R. R., Prabhu, R. K., Morgan, K., Peraire, J., Peiro, J., and Soltani, S., "Applications of an Adaptive Unstructured Solution Algorithm to the Analysis of High Speed Flows", *AIAA paper 90-0395*, January 1990.

4. Mavriplis, D. J., "Adaptive Mesh Generation for Viscous Flows Using Delaunay Triangulation" *ICASE Rep. 88-47, NASA CR 181699*, To appear in Journal of Comp. Physics, 1990.

5. Mavriplis, D. J., Jameson, A., and Martinelli L., "Multigrid Solution of the Navier-Stokes Equations on Triangular Meshes", *AIAA paper 89-0120*, January, 1989.

6. Davis, R. L., and Dannenhoffer, J. F., "Adaptive Grid Embedding Navier-Stokes Technique for Cascade Flows", *AIAA paper 89-0204*, January, 1989.

7. Kallinderis, I., "Adaptation Methods for Viscous Flows" *Ph.D Dissertation, Massachusetts Institute of Technology, CFDL-TR-89-5*, May 1989.

8. Rostand, P. "Algebraic Turbulence Models for the Computation of Two-Dimensional High Speed Flows Using Unstructured Grids", *ICASE Rep 88-63*, Submitted to Numerical Methods in Fluids, November 1988.

9. Baldwin, B. S., Lomax, H., "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows", *AIAA paper 78-275*, 1978

10. Cordova, J. Q., and Barth, T. J., "Grid Generation for General 2-D Regions Using Hyperbolic Equations", *AIAA paper 88-0520*, January, 1988.

11. Wigton, L., *Private Communication*, The Boeing Company.

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

13. Degani, D., Schiff, L. B., and Levy, Y., "Physical Considerations Governing Computation of Turbulent Flows over Bodies at Large Incidence", *AIAA paper 90-0096*, January, 1990.

14. Fritz, W., "Numerical Simulation of 2-D Turbulent Flow Fields with Strong Separation", *ICAS paper ICAS-88-4.6.4*, August, 1988.

15. Cook, P. H., McDonald, M. A., and Firmin, M. C. P., "Aerofoil RAE 2822 Pressure Distributions and Boundary Layer and Wake Measurements", *AGARD Advisory Report No. 138*, May 1979.

16. Martinelli, L., and Jameson, A., "Validation of a Multigrid Method for the Reynolds Averaged Equations" *AIAA paper 88-0414*, January 1988.

17. Volpe, G., "A Multigrid Method for Computing the Transonic Flow Over Two Closely-Coupled Airfoil Components", *ICAS paper 84-1.4.3*, Paper presented at the 14th ICAS Congress, Toulouse, France, September, 1984.

18. Omar, E. , Zierten, T., Hahn, M., Szpizo, E., and Mahal, A., "Two-Dimensional Wind-Tunnel Tests of a NASA Supercritical Airfoil with Various High-Lift Systems, Volume II - Test Data", *NASA CR-2215*, April, 1977.

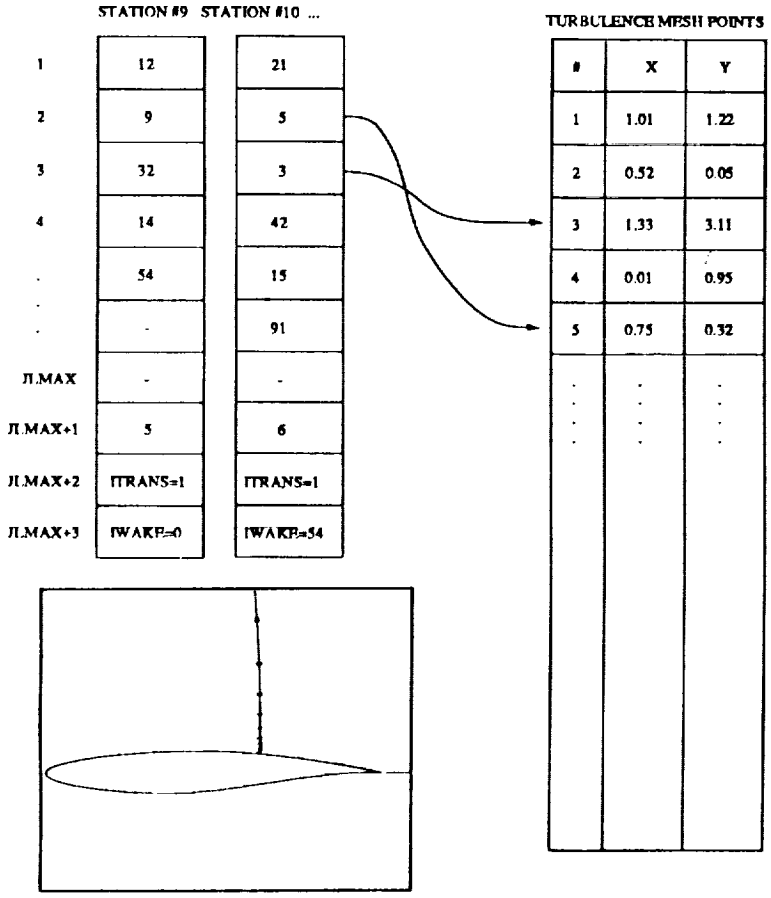

Figure 1
Data-Structure for Turbulence Mesh Stations

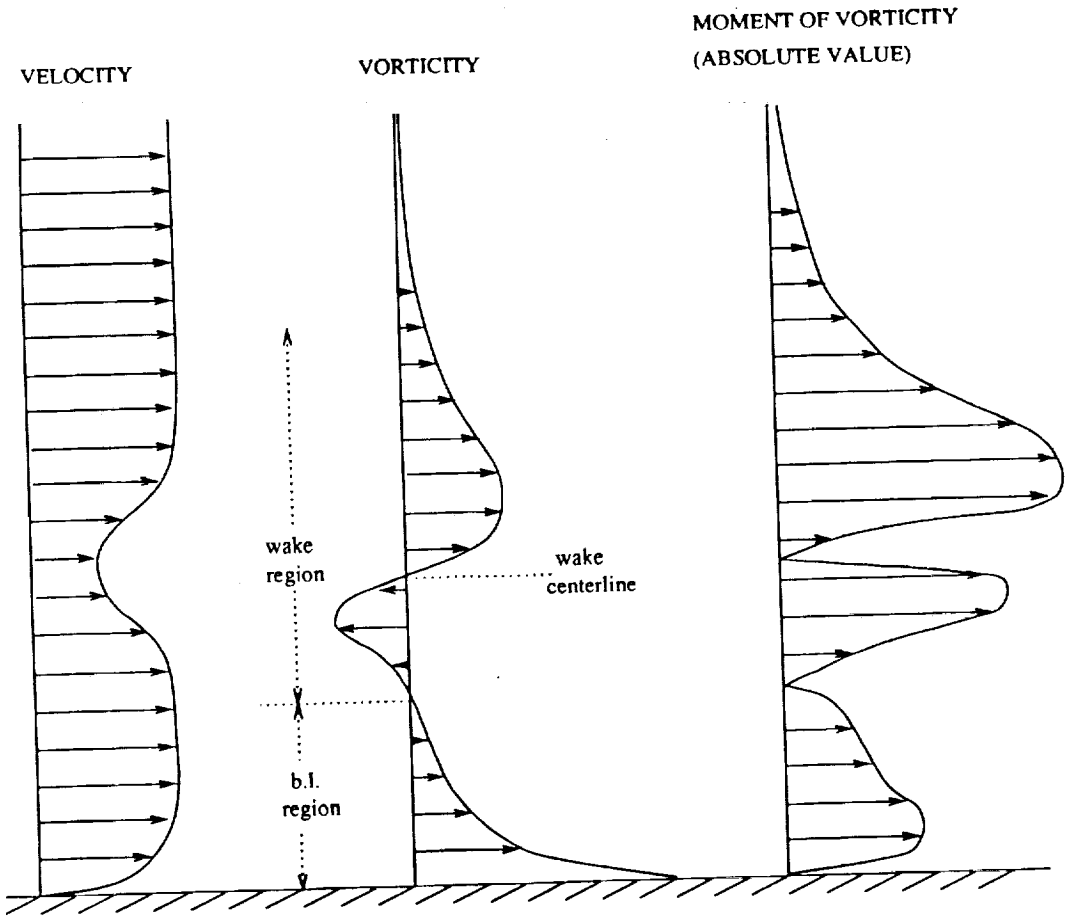

Figure 2
Delimitation of Boundary-Layer and Wake Search Regions for the Algebraic Turbulence Model

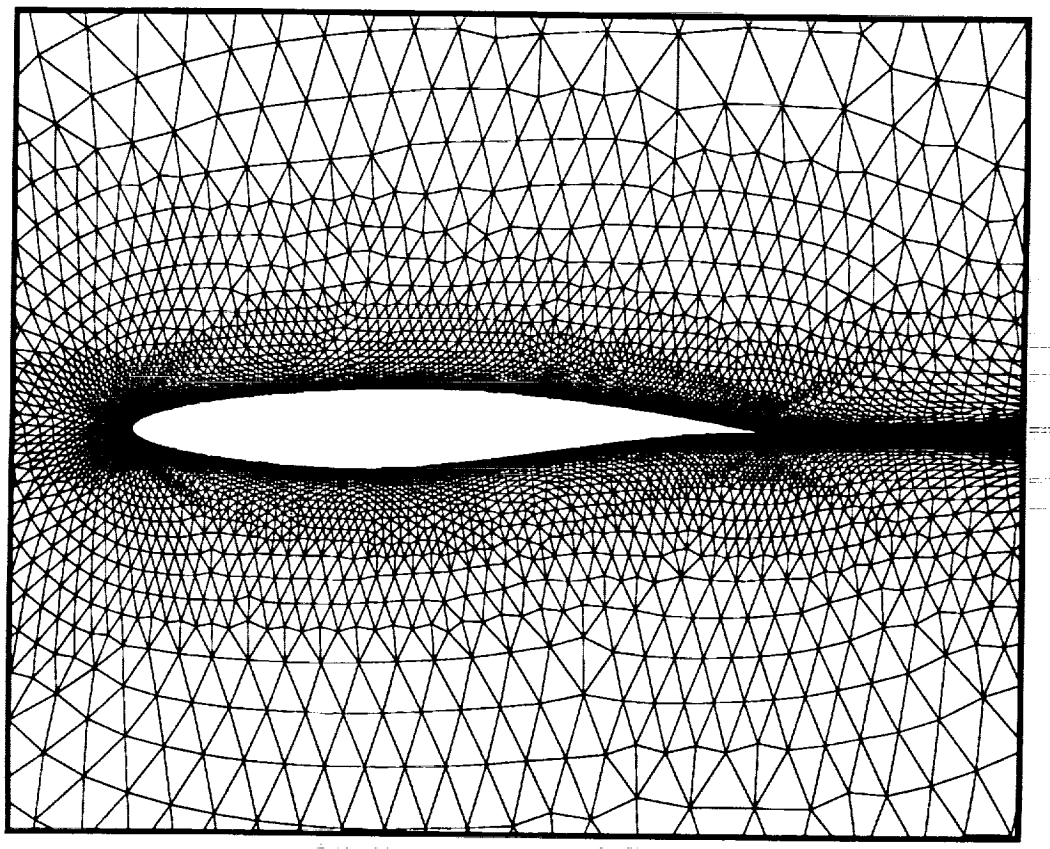

Figure 3
Fully Unstructured Mesh with High Stretching Employed for Computing Turbulent Flow Past an RAE 2822 Airfoil (Number of Points = 13751)

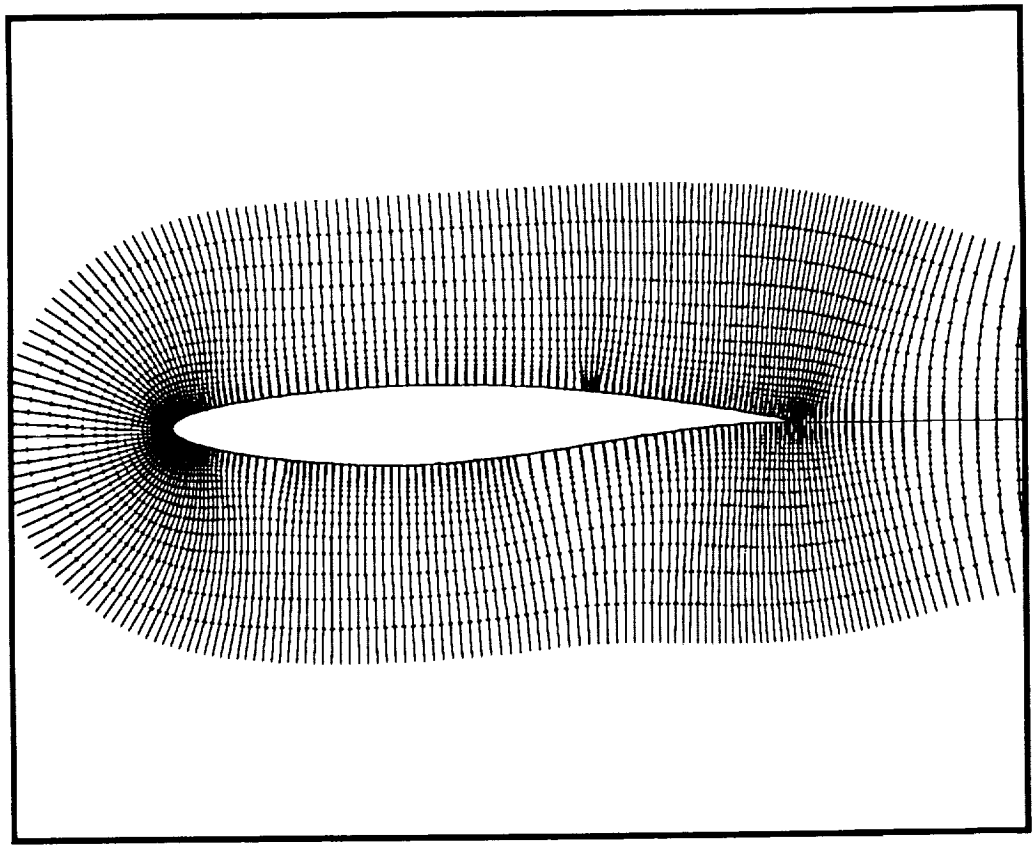

Figure 4
Turbulence Stations Employed for Computing Flow Past an RAE 2822 Airfoil
(Total Number of Points = 13372)

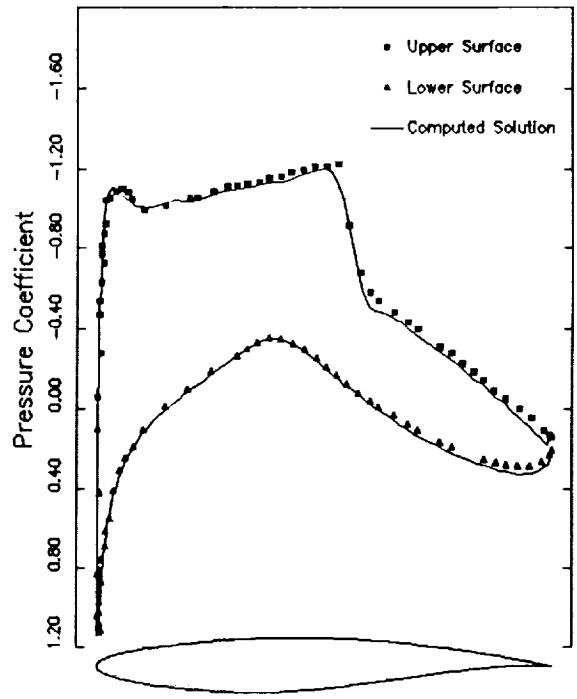

Figure 5
Comparison of Computed Surface Pressure with Experimental Measurements for Flow over an RAE 2822 Airfoil (Mach = 0.729, Re=6.5 million, Incidence = 2.31 degrees)

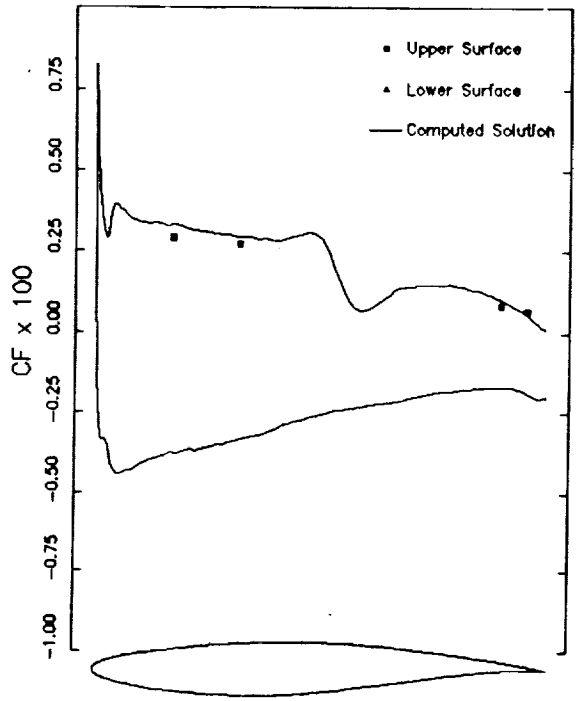

Figure 6
Comparison of Computed Skin Friction with Experimental Measurements for Flow over an RAE 2822 Airfoil (Mach = 0.729, Re=6.5 million, Incidence = 2.31 degrees)

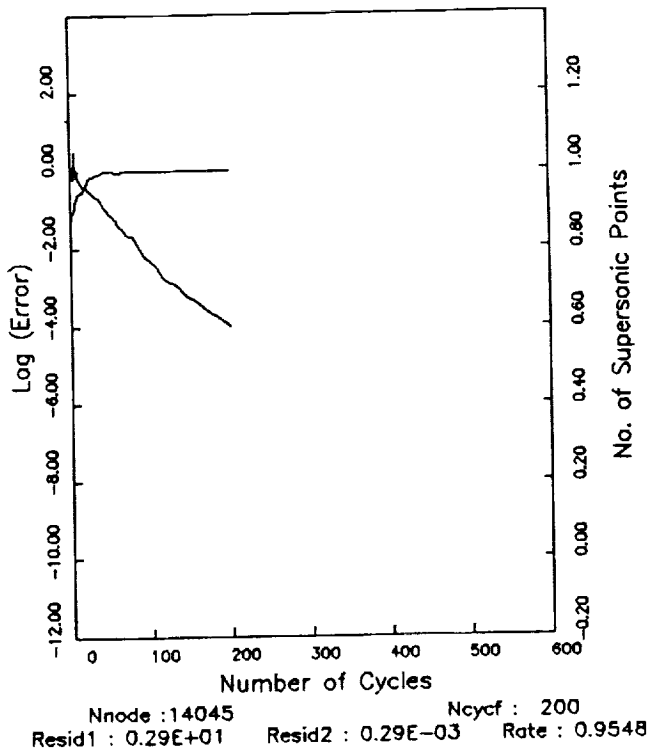

Figure 7
Convergence Rate for Flow over an RAE 2822 Airfoil as Measured by the RMS average of the Density Residuals versus the Number of Multigrid Cycles

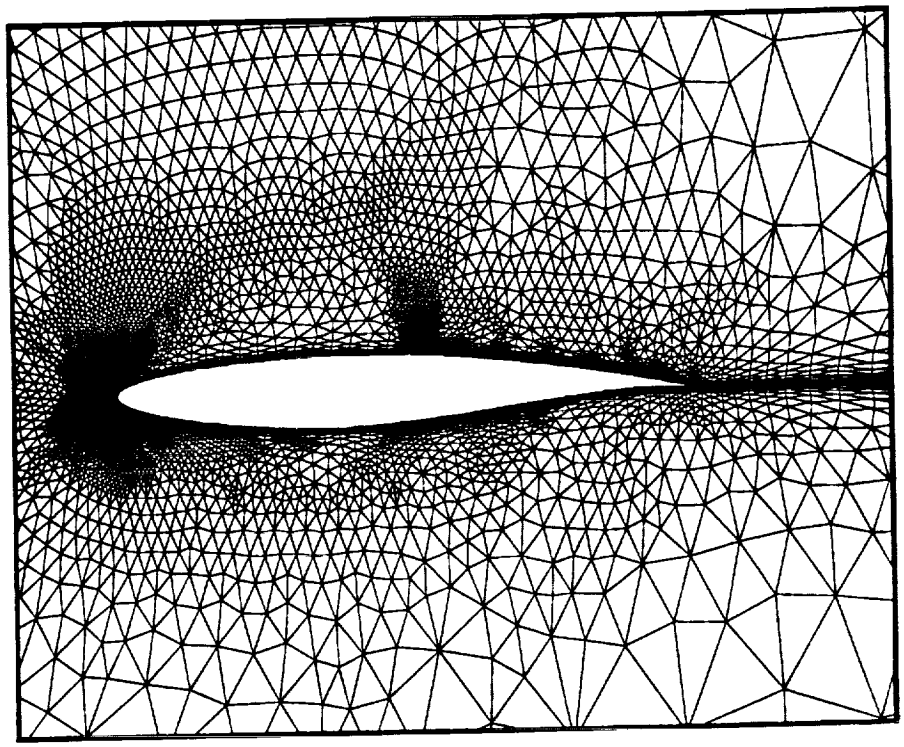

Figure 8
Adaptively Generated Unstructured Mesh for Computing Flow Past an RAE 2822 Airfoil (Number of Points = 12829)

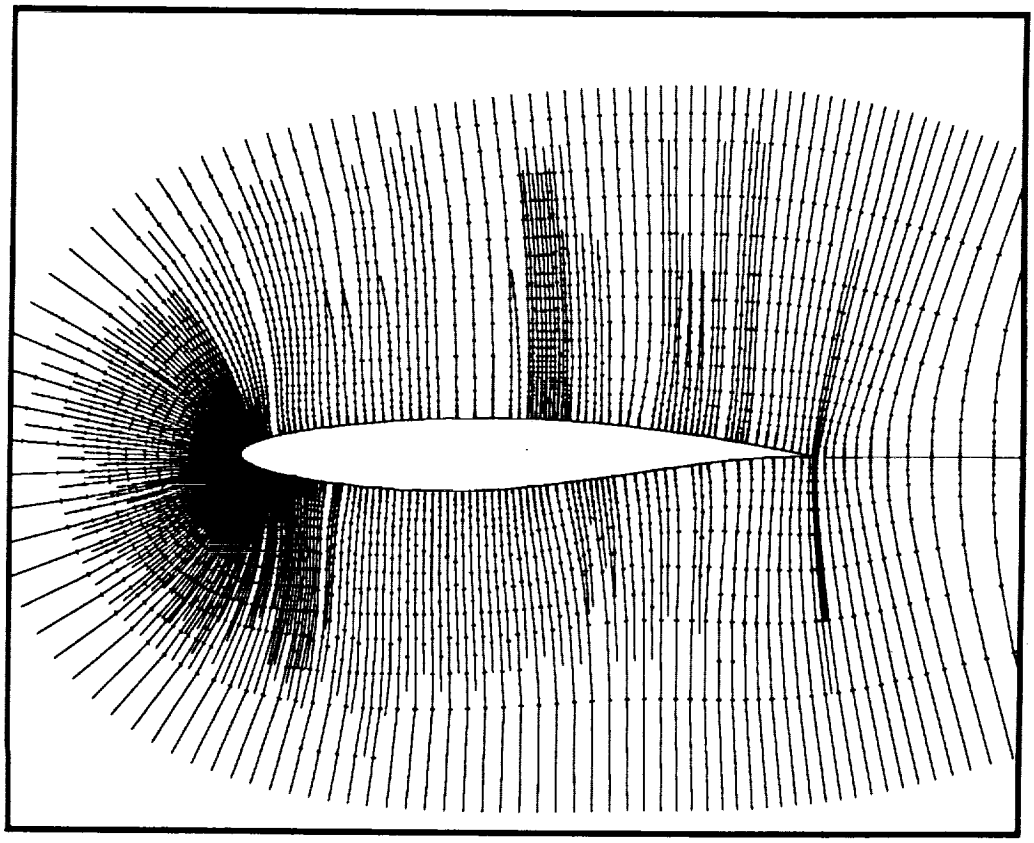

Figure 9
Adaptively Generated Turbulence Mesh Stations for Computing Flow
Past an RAE 2822 Airfoil (Number of Points = 13011)

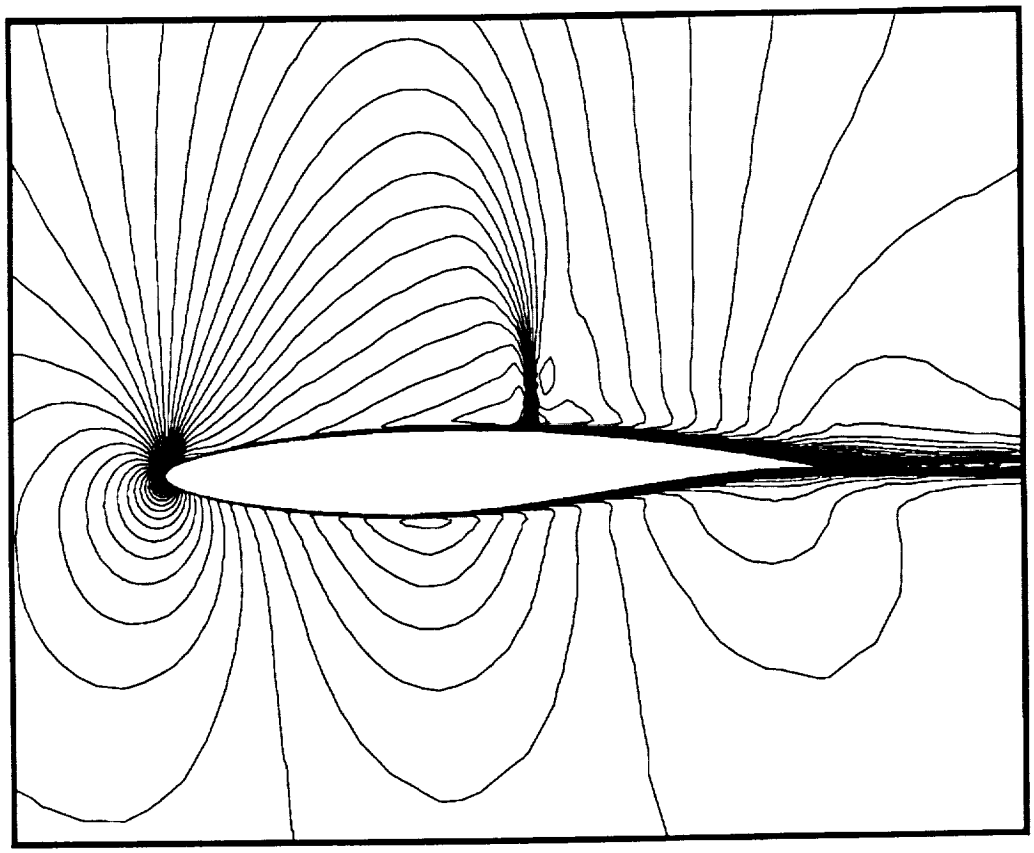

Figure 10
Computed Mach Contours for Flow Past an RAE 2822 Airfoil on the Adaptively Generated Unstructured Mesh

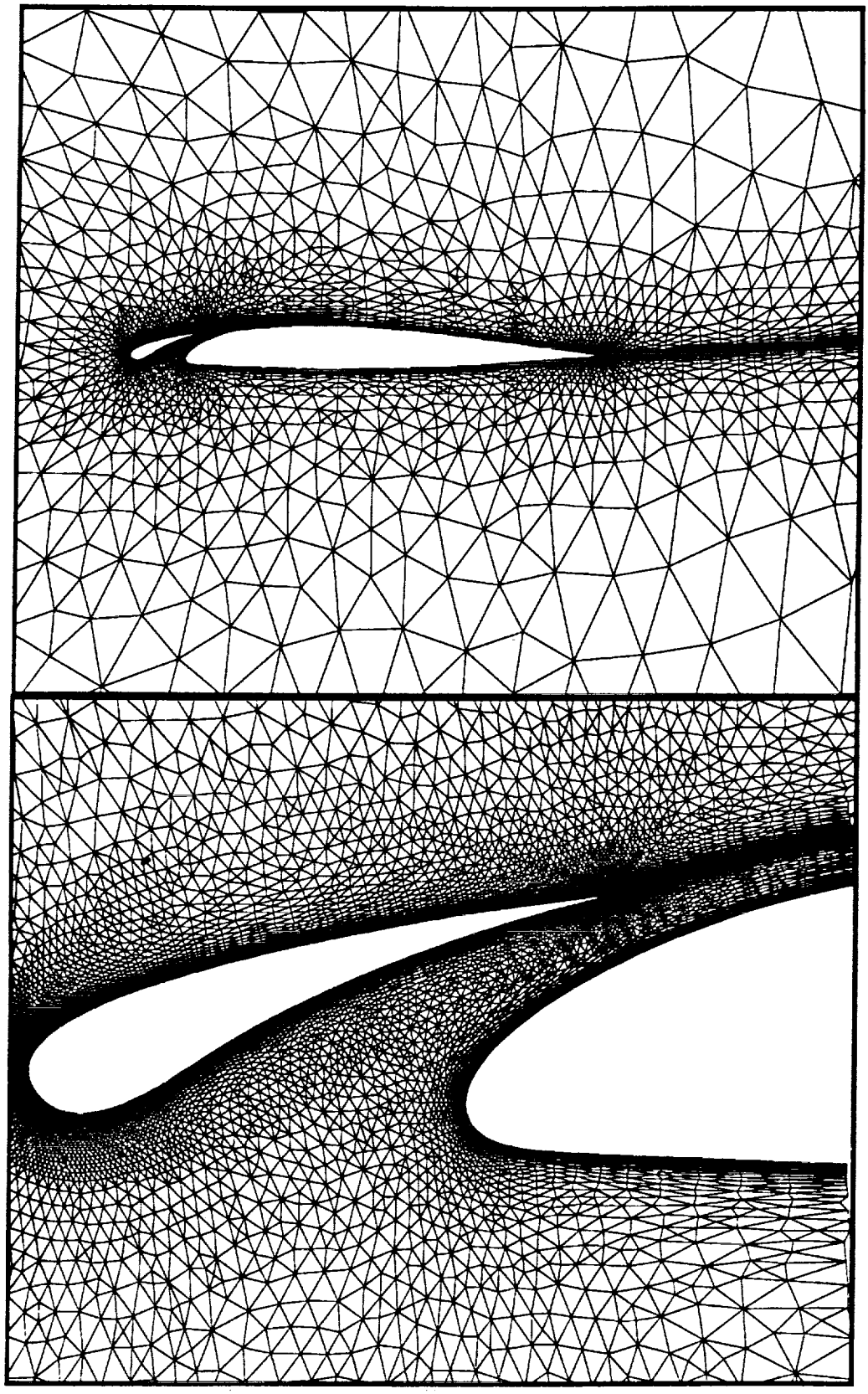

Figure 11
Global View of Coarse Unstructured Mesh and Close-Up View of Fine Unstructured Mesh Employed For Computing Flow Past a Two-Element Airfoil (Coarse Mesh Points = 7272, Fine Mesh Points = 28871)

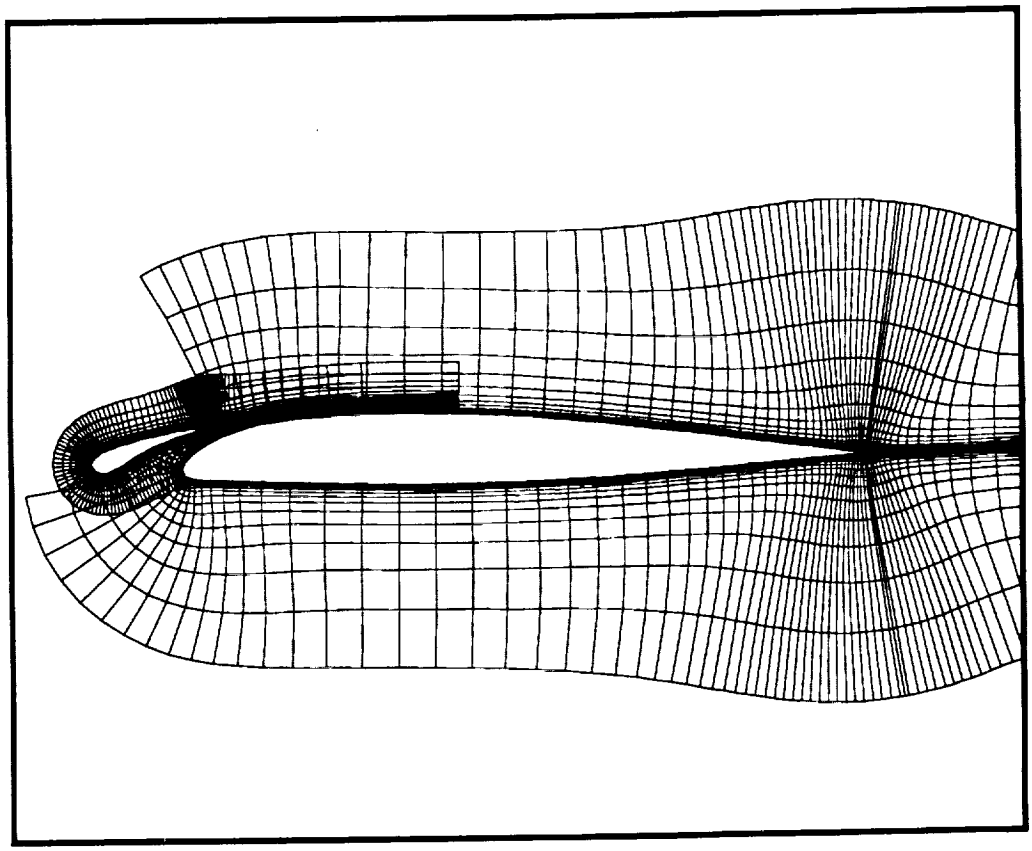

Figure 12
Illustration of Background Turbulence Meshes Employed in the Turbulence Modeling Routine for Computing Flow Past a Two-Element Airfoil

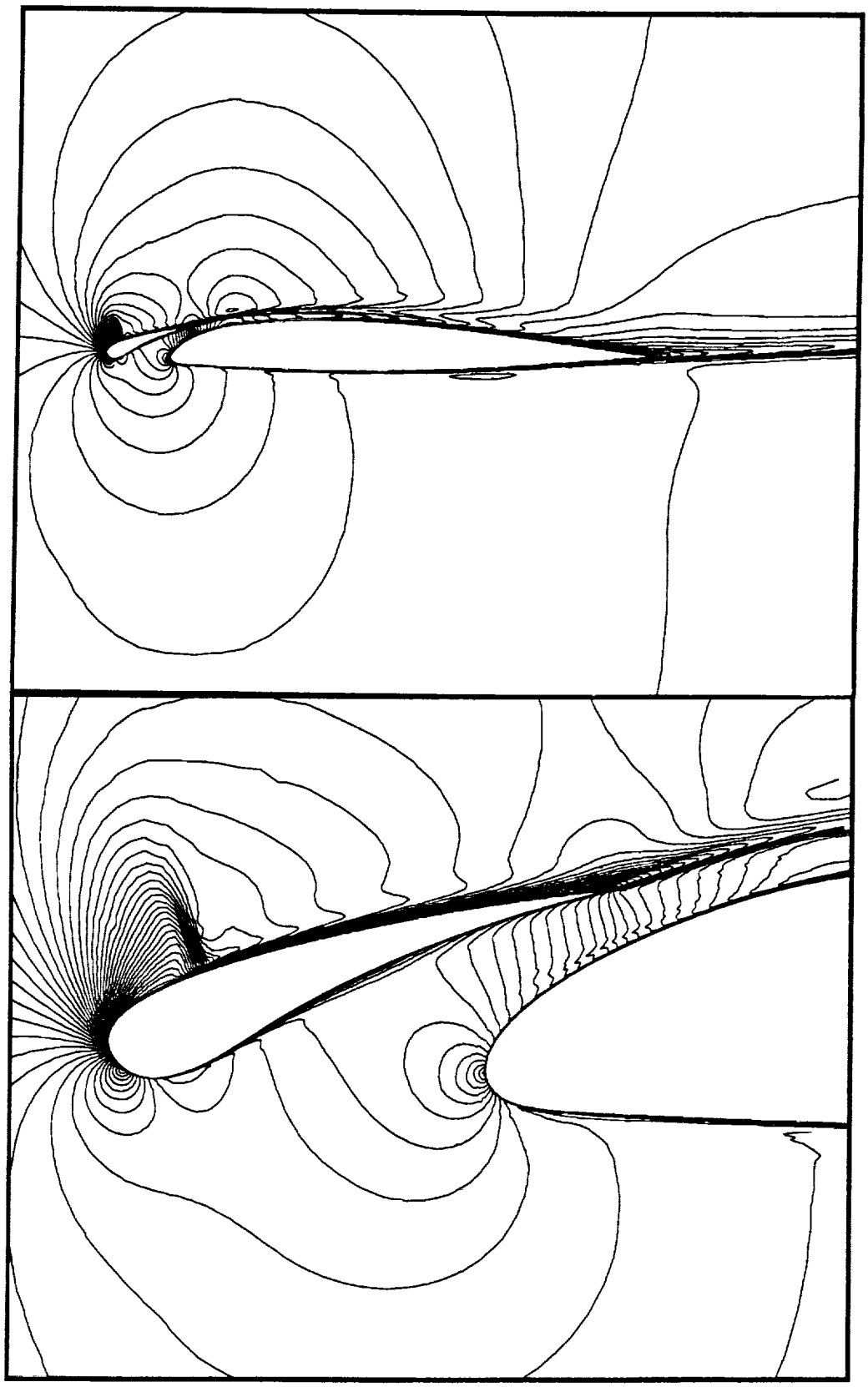

Figure 13
Computed Mach Contours for Flow Past a Two-Element Airfoil
(Mach = 0.5, Re = 4.5 million, Incidence = 7.5 degrees)

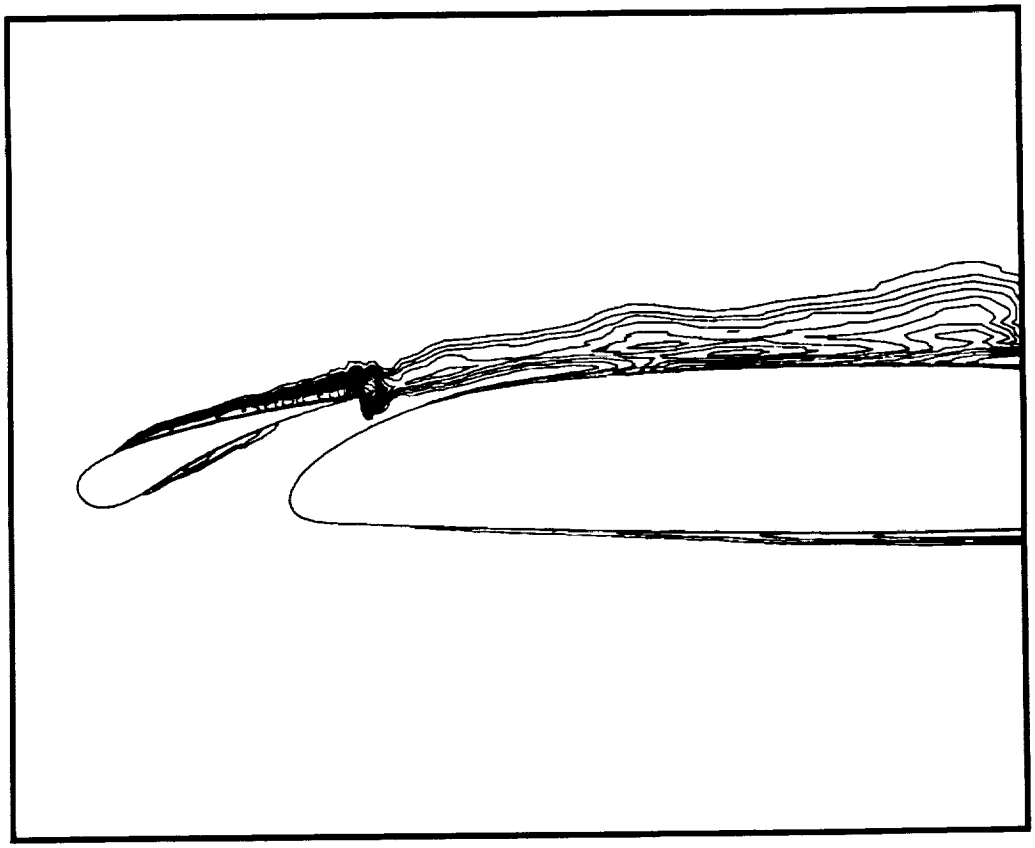

Figure 14
Contours of Eddy Viscosity Produced by the Algebraic Turbulence Model
for the Flow Past a Two-Element Airfoil

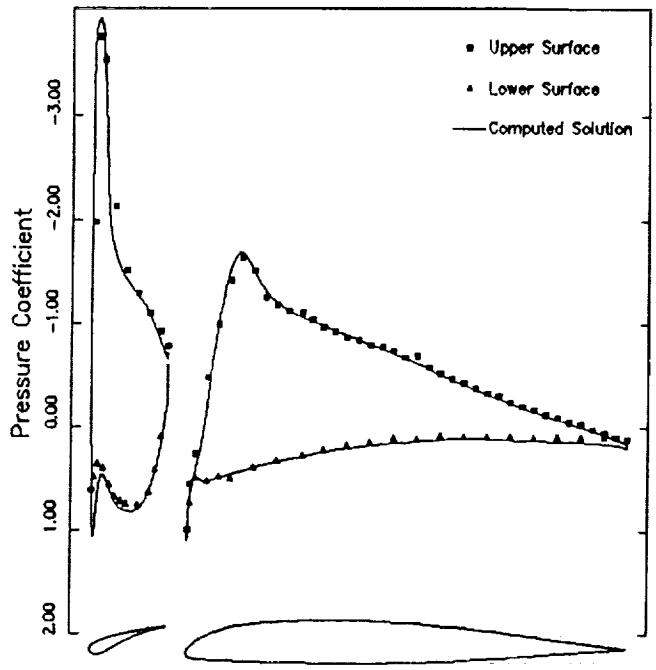

Figure 15
Comparison of Computed Surface Pressure Distribution with Experimental Wind-Tunnel Data for Turbulent Transonic Flow Past a Two-Element Airfoil (Mach = 0.5, Re = 4.5 million, Incidence = 7.5 degrees)

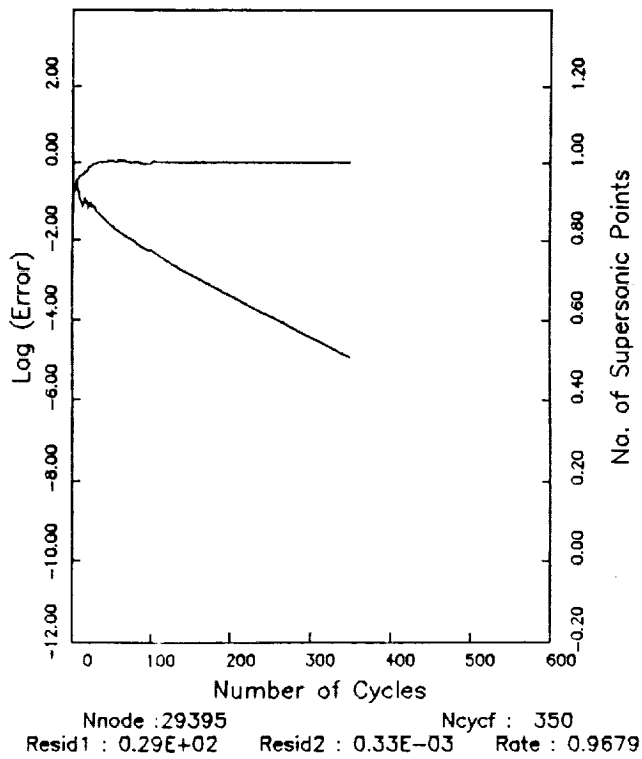

Figure 16
Convergence Rate as Measured by the RMS average of the Density Residuals versus the Number of Multigrid Cycles for Flow Past a Two-Element Airfoil

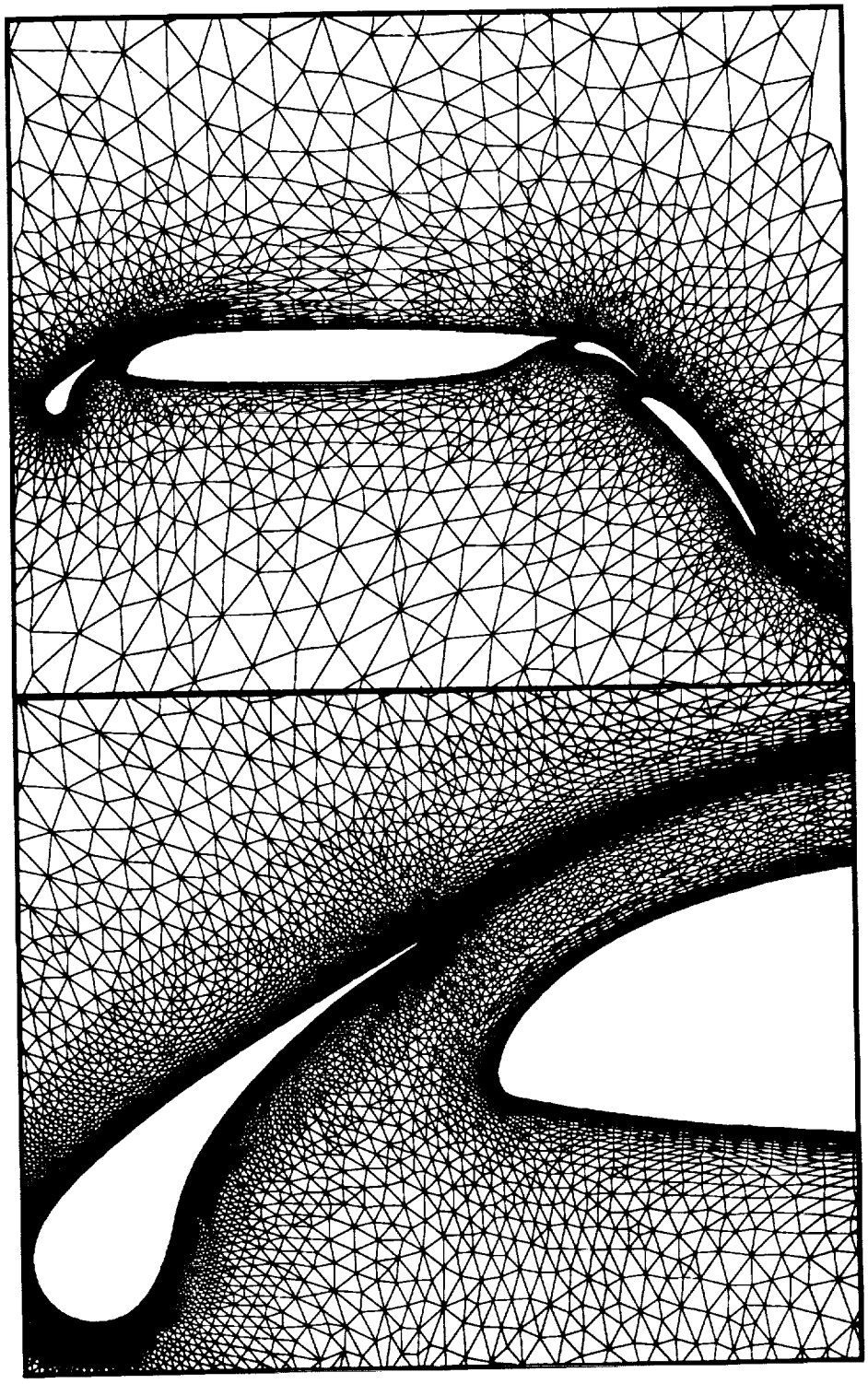

Figure 17
Global View of Coarse Unstructured Mesh and Close-Up View of Fine Unstructured Mesh Employed for Computing Flow Past a Four-Element Airfoil (Coarse Mesh Points = 15896, Fine Mesh Points = 62076)

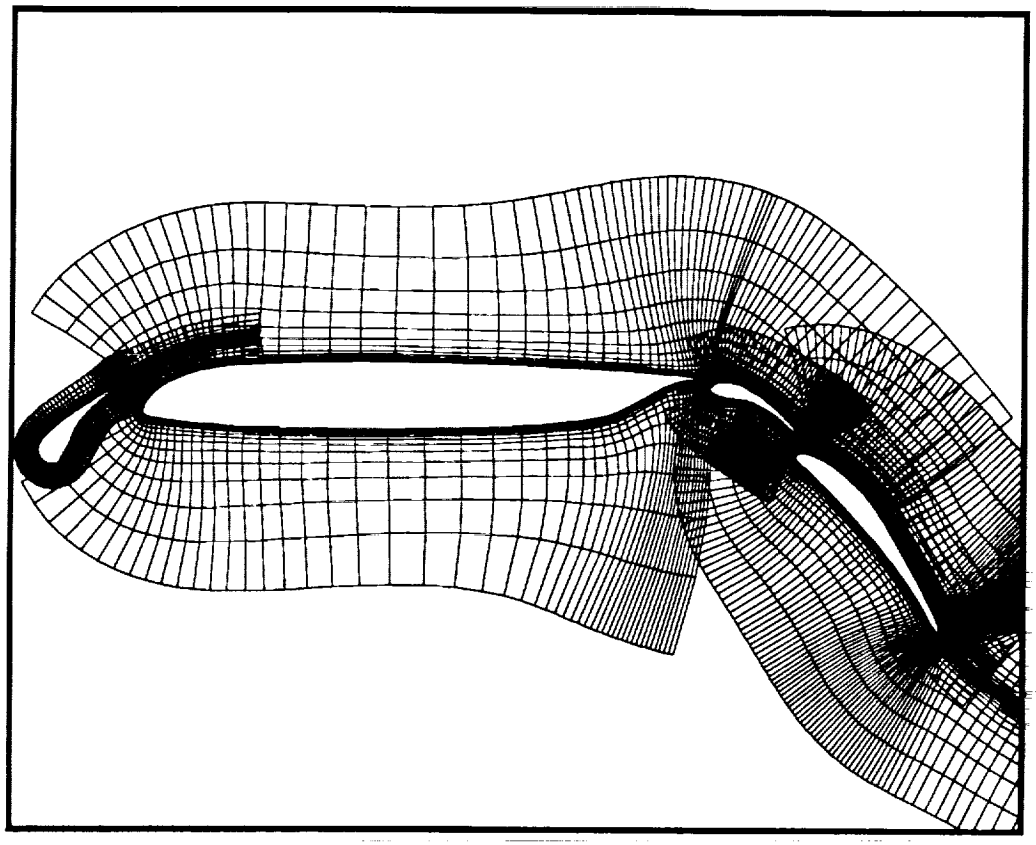

Figure 18
Illustration of Background Turbulence Meshes Employed for Computing Turbulent Flow Past a Four-Element Airfoil Configuration

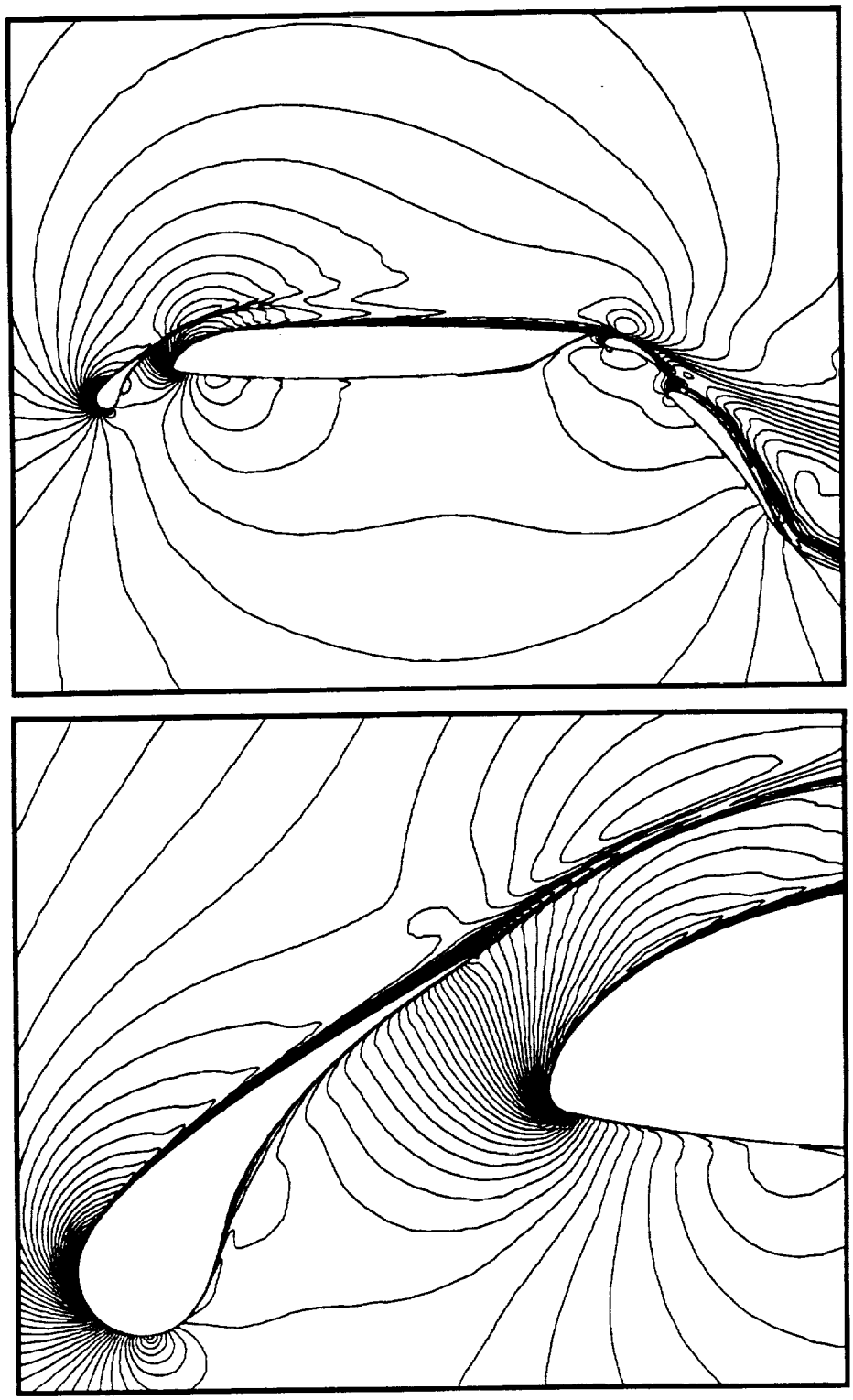

Figure 19
Computed Mach Contours for the Compressible Turbulent Flow
Past a Four-Element Airfoil Configuration
(Mach = 0.2, Re = 2.83 million, Incidence = 8.18 degrees)

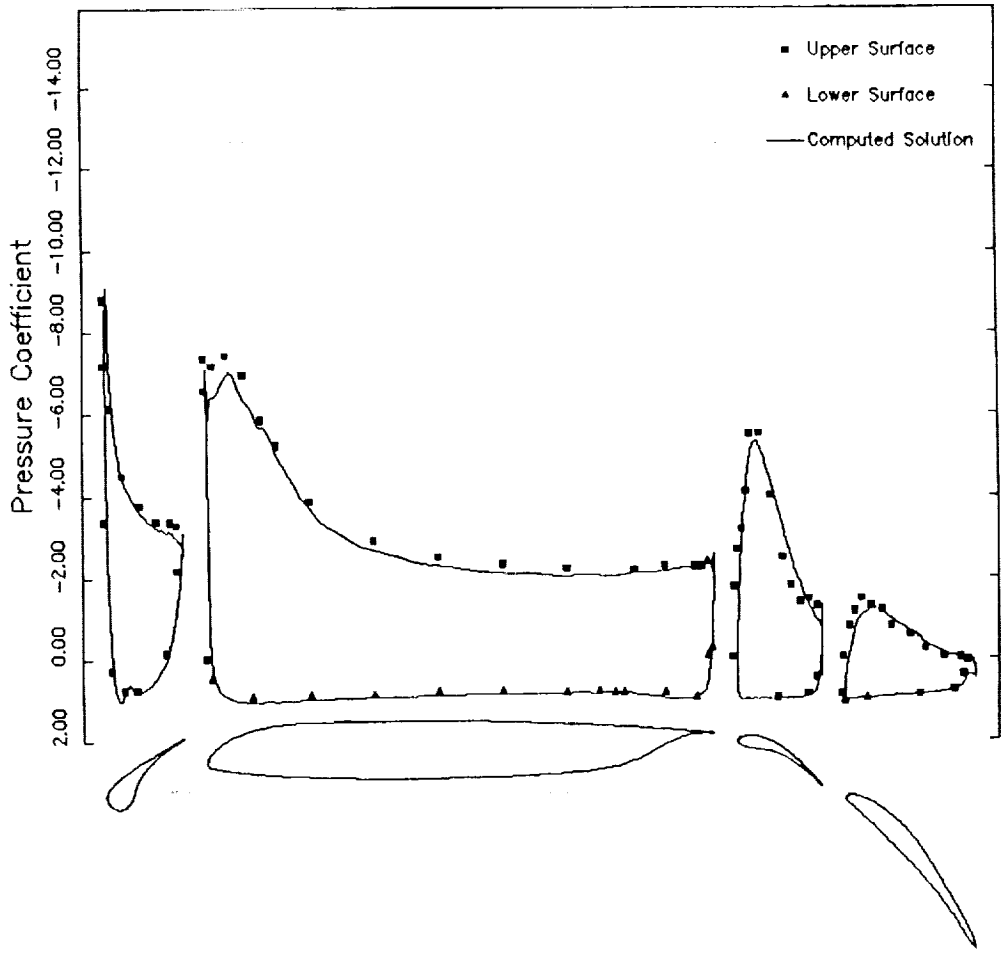

Figure 20
Comparison of Computed Surface Pressure with Experimental Wind-Tunnel Data for Flow Past a Four-Element Airfoil Configuration (Mach = 0.2, Re = 2.83 million, Incidence = 8.18 degrees)

NASA
National Aeronautics and Space Administration

Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-182035 ICASE Report No. 90-30		

4. Title and Subtitle

ALGEBRAIC TURBULENCE MODELING FOR UNSTRUCTURED AND ADAPTIVE MESHES

5. Report Date
May 1990

6. Performing Organization Code

7. Author(s)
Dimitri J. Mavriplis

8. Performing Organization Report No.
90-30

10. Work Unit No.
505-90-21-01

9. Performing Organization Name and Address
Institute for Computer Applications in Science and Engineering
Mail Stop 132C, NASA Langley Research Center
Hampton, VA 23665-5225

11. Contract or Grant No.
NAS1-18605

13. Type of Report and Period Covered
Contractor Report

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Langley Research Center
Hampton, VA 23665-5225

14. Sponsoring Agency Code

15. Supplementary Notes

Langley Technical Monitor:
Richard W. Barnwell

Submitted to AIAA Journal

Final Report

16. Abstract

An algebraic turbulence model based on the Baldwin-Lomax model, has been implemented for use on unstructured grids. The implementation is based on the use of local background structured turbulence meshes. At each time-step, flow variables are interpolated from the unstructured mesh onto the background structured meshes, the turbulence model is executed on these meshes, and the resulting eddy viscosity values are interpolated back to the unstructured mesh. Modifications to the algebraic model were required to enable the treatment of more complicated flows, such as confluent boundary layers and wakes. The model is used in conjunction with an efficient unstructured multigrid finite-element Navier-Stokes solver in order to compute compressible turbulent flows on fully unstructured meshes. Solutions about single and multiple element airfoils are obtained and compared with experimental data.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Turbulence; Unstructured; Adaptive	34 - Fluid Mechanics and Heat Transfer 02 - Aerodynamics Unclassified - Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	30	A03

NASA FORM 1626 OCT 86