NASA Contractor Report 181699

ICASE REPORT NO. 88-47

ICASE

ADAPTIVE MESH GENERATION FOR VISCOUS
FLOWS USING DELAUNAY TRIANGULATION

(NASA-CR-181699) ADAPTIVE MESH GENERATION FOR VISCOUS FLOWS USING DELAUNAY TRIANGULATION (NASA) 16 p CSCL 01A N88-29749
Unclas
G3/02 0167389

Dimitri J. Mavriplis

Contract No. NAS1-18107
August 1988

INSTITUTE FOR COMPUTER APPLICATIONS IN SCIENCE AND ENGINEERING
NASA Langley Research Center, Hampton, Virginia 23665

Operated by the Universities Space Research Association

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665

ADAPTIVE MESH GENERATION FOR VISCOUS FLOWS USING DELAUNAY TRIANGULATION

Dimitri J. Mavriplis

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

ABSTRACT

A method for generating an unstructured triangular mesh in two dimensions, suitable for computing high Reynolds number flows over arbitrary configurations is presented. The method is based on a Delaunay triangulation, which is performed in a locally stretched space, in order to obtain very high aspect ratio triangles in the boundary layer and wake regions. It is shown how the method can be coupled with an unstructured Navier-Stokes solver to produce a solution adaptive mesh generation procedure for viscous flows.

This work was supported under the National Aeronautics and Space Administration under NASA Contract No. NAS1-18107 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

In recent years, the use of unstructured triangular and tetrahedral meshes in two and three dimensions has become more widespread for computational fluid dynamics problems. The advantages of unstructured meshes lie in their ability to deal with arbitrarily complex geometries, while providing a natural setting for the use of adaptive mesh enrichment techniques. On the other hand, the accuracy of unstructured mesh discretizations, and the efficiency of unstructured mesh solvers have generally fallen short of their structured mesh counterparts. The appearance of more efficient and accurate Euler solvers for unstructured meshes [1,2,3], combined with the benefits of adaptive meshing, and a general drive to problems of higher geometric complexity have combined to make unstructured meshes the preferred choice for many inviscid flow problems [4,5]. However, few attempts at solving high Reynolds number viscous flows about complex configurations with unstructured meshes are known. The efficient solution of such flows requires the generation of highly stretched elements in the thin boundary layer regions, where the resolution required in the direction normal to the layer can be several orders of magnitude greater than that in the streamwise direction. Present efforts at computing such flows have concentrated on the use of composite structured-unstructured meshes [6,7], where a thin structured triangular or quadrilateral mesh is placed in the boundary layer regions, and an unstructured mesh is used to fill the remainder of the domain. In this work, it is proposed to employ an unstructured mesh of triangles throughout the entire domain. This approach requires extreme stretching of the unstructured mesh in the boundary layer regions. However, it has the advantage of providing a completely automatic grid generation tool for arbitrary configurations, obviating the need for any human interaction, such as that required to define the structured-unstructured interface in the former approach. It also offers the possibility of obtaining a fully adaptive smoothly varying mesh throughout the viscous regions as well as in regions where the distance between neighboring boundaries may be smaller than the boundary layer thickness.

Of the various algorithms for generating triangular meshes in two dimensions, the advancing front method [8], and the Delaunay triangulation method [9,10] have been successfully applied to generate solution-adaptive evolving meshes. Sophisticated implementations of the advancing front method incorporate directional stretching and refinement by successive remeshing according to stretching factors and directions obtained from a flow solution on a previous mesh. While this technique has proven valuable for inviscid flow calculations, the stretchings obtained are still several orders of magnitude smaller than that required for resolving viscous boundary layer flows.

The first application of Delaunay triangulation to aerodynamic problems was performed by Weatherill [11]. The extension of this procedure to three dimensions has been pursued by Baker [12]. Delaunay triangulation is particularly well suited for adaptive meshing techniques, since it may be formulated as a sequential and local process. New points may be added and triangulated locally without the need for remeshing the domain in whole or in part. For a given set of data points, a Delaunay triangulation will produce the most equiangular triangles possible, and thus is not well suited for the generation of directionally refined meshes. In this paper, it will be shown how a Delaunay triangulation can be modified to accommodate directional stretching of any desired magnitude.

2. THE DELAUNAY TRIANGULATION

Given a set of points in two dimensions, there exists many ways of joining them together to form a set of non-overlapping triangles. A Delaunay triangulation represents a unique construction of this type, which obeys certain specific properties. The geometric dual of the Delaunay triangulation is known as the Dirichlet tessalation. It is constructed by associating with each data point the area of the plane which is closer to that point, in terms of Euclidien distances, than to any other point in the plane. These regions have polygonal shapes and the tessalation of a closed domain results in a set of non-overlapping convex polygons covering the entire domain. If all point pairs whose Dirichlet regions have a face in common are joined by straight line segments, the Delaunay triangulation of these points is obtained. Figure 1 depicts the Dirichlet regions and associated Delaunay triangulation for a small set of points.

A Delaunay triangulation obeys the circumcircle property, which states that no vertex from any triangle may lie within the circumcircle of any other triangle. This can also be shown [13] to be equivalent to the equiangular property, which states that a Delaunay triangulation is that which maximizes the minimum of the six angles in any pair of triangles of the mesh which make up a convex quadrilateral. Each of these properties may be used as the basis for a method of constructing a Delaunay triangulation.

2.1. Bowyer's Algorithm

Bowyer's algorithm [14], makes use of the cimcumcircle property to generate a Delaunay triangulation in a sequential manner. The mesh points are introduced one at a time into an existing triangulation. The triangles whose circumcircles are intersected by the new mesh point are flagged. These may be quickly determined by first locating the triangle which encloses the new point. The circumcircle of this triangle must be intersected by the new point, and so it is flagged. The neighbors of this triangle are then searched, and then their neighbors, thus proceeding outwards in a tree-search pattern, each leg of which terminates when a non-intersected triangle has been located. The union of the flagged triangles forms a convex polygonal region, and a new structure is defined in this region by joining the new point to all the vertices of the polygon. Proofs that the polygon is convex, and that the resulting triangulation is indeed a Delaunay structure can be found in the literature [12]. If an efficient search strategy is employed, Bowyer's algorithm exhibits linear computational complexity with the number of mesh points.

2.2. Diagonal Swapping Algorithm

This algorithm, originally proposed in [15], and reviewed in [13], makes use of the equiangular property. Assuming we have an arbitrary triangulation of a given set of points, we may proceed to transform it into a Delaunay triangulation by repeatedly swapping the positions of the edges in the mesh in accordance with the equiangular property. Hence, each pair of triangles which constitute a convex quadrilateral are examined. The two possible configurations for the diagonal interior to the quadrilateral are examined, as shown in Figure 2, and the one which maximizes the minimum of the six interior angles of the quadrilateral is chosen. Each time an edge swap is performed, the triangulation becomes more equiangular. Multiple edge swapping passes through the entire mesh are then effected, until the most equiangular (Delaunay) triangulation is obtained. Although this algorithm is guaranteed to converge, it has a much higher complexity than Bowyer's algorithm, and is only useful for constructing an equiangular triangulation either when the initial mesh is coarse, or when it represents a small deviation from a Delaunay triangulation.

3. STRETCHING FACTORS

Equiangular triangulations, which have been termed "best fit or optimal triangulations" may be desirable if one wishes to tessalate or subdivide a domain in a uniform manner. However, by its very nature, such triangulations are ill-suited for the generation of meshes with directional stretchings. In fact, even if the mesh points are distributed sparsely in one direction, and compactly in the perpendicular direction, the Delaunay construction will generally produce low aspect-ratio triangles of widely varying size. Thus, the standard Delaunay construction must be modified to accommodate directional mesh stretching.

We proceed by defining a stretching vector, i.e. a direction and magnitude, at each point of the mesh. It is important to note that this stretching is a local property, but that it must vary smoothly throughout the domain. Because a Delaunay triangulation is a local construction, we may thus map a local region of the mesh onto the stretched space defined by the stretching vector in that region, perform the triangulation in this space, and then project the triangulation back into physical space. A stretching vector at a point is given by its magnitude s and its direction θ, where

$$s > 1 \quad \text{and} \quad -\frac{\pi}{2} < \theta < \frac{\pi}{2} \tag{1}$$

When s=1, no stretching occurs. If a stretching smaller than unity is encountered, it is replaced by the inverse stretching in the perpendicular direction. Equation (1) also reflects the fact that two stretchings of equal magnitudes in opposite directions are equivalent.

The locally mapped space is obtained by considering a two-dimensional control surface in three dimensional space, as described in [16]. If d represents a Euclidien distance in physical space:

$$d^2 = \Delta x^2 + \Delta y^2 \tag{2}$$

then, in the mapped space, it is replaced by

$$\delta^2 = \Delta x^2 + \Delta y^2 + \Delta z^2 \tag{3}$$

where

$$\Delta z = [\ \Delta x \sin\theta - \Delta y \cos\theta\ ]\ (s-1) \tag{4}$$

For a stretching of unity, Δz vanishes and the mapped space and the physical space become identical. If for example, a stretching of s is applied in the x-coordinate direction, as would be required to resolve a boundary layer aligned with this direction, then, taking $\theta = 0$, an increment in the x direction maps to

$$\delta_x = \Delta x \tag{5}$$

and an increment in the y direction maps to

$$\delta_y = \Delta y\ [1 + (s-1)^2]^{\frac{1}{2}} \tag{6}$$

which, for large values of s, approaches

$$\delta_y = s\,\Delta y \tag{7}$$

Thus, if we have a distribution of mesh points in the physical space which is closely packed in the y-direction, and sparse in the x-direction, then, in the mapped space, this mesh point distribution becomes more uniform in both directions. By triangulating this mapped mesh point distribution, we obtain an equiangular triangulation in the stretched space, and a directionally stretched mesh in the physical space. If the diagonal swapping algorithm is to be used for constructing the triangulation, then the six interior angles of each convex quadrilateral in the locally stretched space must be considered. On the other hand, when Bowyer's algorithm is employed, the procedure can be thought of as the construction of a modified Delaunay triangulation, where the circumcircles of the triangles in the stretched space correspond to ellipses in the physical space.

The necessary condition for this stretched triangulation to succeed requires that the local variation of the stretching vectors in space be small compared with the average local cell size. Thus, in regions were the stretching values vary rapidly, a fine mesh resolution is needed so that, on the scale of the local mesh cells, the assumption of a constant stretching can be made, and the stretched space appears locally planar or Euclidien.

4. INITIAL MESH GENERATION

In the above discussion, it has been assumed that an initial triangulation with adequate resolution in highly stretched regions exists, and that all further modifications or refinements of the mesh are of a purely local nature. However, the construction of an initial mesh is a global procedure, and thus local stretching values cannot be directly accommodated at this initial stage. To circumvent this difficulty, a regular Delaunay triangulation is first generated in physical space using Bowyer's algorithm, while disregarding the stretching values. Although this step involves non-local procedures, it results in a discretization of the physical space upon which purely local procedures may now be performed. The edge swapping algorithm is then employed to transform the mesh from an equiangular triangulation in physical space, to an equiangular triangulation in the stretched space. Because the initial mesh need only be coarse, and since no edge swapping is required in regions where the stretching values are small, the edge swapping algorithm can be expected to converge rapidly. Once this initial coarse stretched mesh is obtained, it can be adaptively refined by adding points and retriangulating locally following Bowyer's algorithm in the locally stretched space.

5. GENERAL PROCEDURE FOR MULTI-ELEMENT AIRFOILS

To generate a stretched Delaunay mesh around a multi-element airfoil configuration, a set of mesh points and their associated stretching vectors must first be defined. This is achieved by generating a standard quadrilateral C-mesh around each airfoil element, thus resulting in a set of overlapping structured meshes. Stretching vectors are then defined at each mesh point, by taking their magnitude s as

$$s = \max \left(\frac{\Delta\xi}{\Delta\eta} , \frac{\Delta\eta}{\Delta\xi} \right) \tag{8}$$

and their direction equal to the direction of the ξ or η structured mesh lines, depending on whether the first or second values are chosen for s in equation (8). Here, $\Delta\xi$ and $\Delta\eta$ represent the local spacing of the structured C-mesh in the two mesh coordinate directions.

The mesh point distribution resulting from the set of overlapping C-meshes can now be used as the basis for a Delaunay triangulation. An initial triangulation is set up by joining the trailing edge point of the main airfoil to all the outer boundary points. The points on the surface of the airfoils are then introduced and triangulated into the existing structure using Bowyer's algorithm. Because neighboring surface points on a given airfoil are much closer to each other than they are to any points on other airfoils, or to the far-field boundary points, the resulting triangulation is body conforming. That is, the triangulation contains elements inside the regions defined by the airfoils, as well as in the exterior of these regions, and the interior triangles all have a face aligned with the airfoil surfaces. These interior triangles are then identified and protected, thus preventing their structure from being broken when new mesh points are introduced, and hence preserving the integrity of the boundaries. The remaining mesh points are then introduced and triangulated into the existing structure. It can be seen that, in the initial stages of this construction, a small number of triangles will cover the entire domain, and thus Bowyer's algorithm can no longer be considered to be a purely local process. Hence local stretching values cannot be taken into account at this stage, and a Delaunay triangulation in the physical space is generated. However, once this triangulation has been constructed, the space is sufficiently discretized so that all further operations may be effected in a purely local manner. The next step consists of applying a Laplacian-type averaging procedure to the local stretching vectors over the existing Delaunay triangulation, thus ensuring a smoothly varying distribution of the stretching throughout the domain. The edge swapping routine is then applied in conjunction with the smoothed local stretching values to obtain a Delaunay triangulation in the stretched space. This mesh may finally be smoothed by slightly repositioning the points according to a Laplacian filtering operation described previously [1].

Once this initial stretched triangulation has been generated, it may be combined with a flow solver to produce an adaptive mesh refinement procedure, where the mesh point distribution is defined by the evolving solution of the flow field. For steady state calculations, this corresponds to converging the solution on the initial coarse mesh, adding points in regions where the computed flow gradients are large, and retriangulating these points into the existing structure using Bowyer's algorithm in the locally stretched space. When new points are introduced, they are assigned stretching values taken as the average of the stretchings of the neighboring points, thus maintaining a smooth distribution of stretching throughout the domain. When points are added on the surface of an airfoil, they must be repositioned onto the original surface definition of the airfoil, which, for curved surfaces, will not in coincide with the position determined by linear interpolation between the two adjacent surface points. The new surface points are then triangulated by joining them to the two neighboring surface points, and to the vertices of all triangles exterior to the airfoil whose circumcircles are intersected. The flow solution is then interpolated onto the new finer mesh, and the entire solution-adaptation process is repeated.

6. SINGLE AIRFOIL MESH

As a first example, an adaptive Navier-Stokes triangular mesh has been generated about a single NACA 0012 airfoil. An initial mesh, containing 1360 points, is first generated by constructing a structured C-mesh around the airfoil with a hyperbolic grid generator, triangulating this point distribution, and then swapping the diagonals. The full Navier-Stokes equations are then discretized and solved for on this mesh. The mesh is then adaptively refined according to the local gradient of density. The difference in density along each edge of the mesh is examined. When this value is larger than the average difference of the density across all the mesh edges, a new point is added midway along that edge. These new points are then triangulated into the existing mesh using Bowyer's algorithm in the locally stretched space. The refined mesh is then smoothed out by slightly repositioning the mesh points according to a Laplacian filtering operation [1], to ensure a smooth distribution of elements. The refined mesh, which contains a total of 2316 points, is depicted in Figure 3, where the refinement occurring in the boundary layer regions and at the leading and trailing edges is apparent. The Figure illustrates the topology of the mesh in the vicinity of the leading and trailing edges, where a smooth transition from an essentially regular, highly stretched triangulation near the body, to a random unstructured triangulation further out in the flow-field is observed.

7. TWO-ELEMENT AIRFOIL MESH

The second configuration consists of a main airfoil with a leading edge slat. The initial mesh point distribution is obtained by constructing a structured C-mesh around each airfoil. The C-mesh about the main airfoil extends out to the far-field boundary, while the C-mesh about the slat is truncated less than one chord out from the surface of the slat, and also beyond the region where the wake lines from the slat impinge upon the main airfoil. This point distribution is then triangulated, the stretching values are smoothed, the edges swapped, and finally the point distribution is smoothed. The resulting mesh, which contains 5856 points is depicted in Figure 4. The stretching of the mesh in the boundary layer and wake regions of both airfoils is apparent, and a smooth transition of the elements is observed in the gap region, between the main airfoil and the leading edge slat. The steady state solution of the Navier-Stokes equations was obtained on this mesh, for a Mach number of 0.5, an incidence of 3°, and a Reynolds number of 5000. A plot of the Mach contours in the flow field is given in Figure 5, where the boundary layer regions are evident, and a recirculation region is observed near the trailing edge on the upper surface of the main airfoil. A region of low velocity fluid is also seen to occur in the gap region between the main airfoil and the slat. A plot of the velocity vectors in this region, as given in Figure 6, clearly shows the boundary layer profiles on both airfoils, as well as the region of recirculating flow on the lower surface of the slat. This solution was then used to adaptively refine the mesh according to the local gradient of the Mach number. The refined mesh, depicted in Figure 7, contains 11377 points. Refinement is seen to occur in the boundary layer regions, as well as in a portion of the gap region. However, as expected, the regions of recirculating flow are left unrefined. This mesh contains triangles of aspect ratio of the order of 1000:1 in the wake regions, and up to 100:1 midway along the surface of the main airfoil, and on the upper surface of the slat.

8. CONCLUSION

A method for adaptively generating unstructured triangular meshes with highly stretched elements, suitable for Navier-Stokes computations has been demonstrated. The method is efficient, in that once an initial coarse stretched triangulation has been set up, all further refinements are of a local nature and have a near linear computational complexity. For the two-element airfoil mesh of the previous section, the initial coarse mesh required approximately 210 CPU secs to generate on a Convex C-1 computer. Roughly 1/3 of this time was required to construct the initial Delaunay triangulation in physical space, and the remainder was required by the edge swapping algorithm, which converged in 13 passes through the mesh. The adaptive refinement of this grid, which effectively doubled the number of mesh points, required roughly 80 secs of CPU time.

In the examples presented in this work, the stretching factors for the mesh were determined at the outset, and held fixed throughout the adaptive refinement procedure. In future work, it may be possible to adaptively modify these stretchings according to the evolving flow solution.

The extension of this technique to three dimensions is not entirely straight forward. Although Bowyer's algorithm extends readily to higher dimensions, the equiangular property applies only in two dimensions, and thus no straightforward counterpart to the edge swapping algorithm exists in three dimensions. However, all the other concepts apply in three dimensions, thus if an initial coarse stretched mesh can be generated in three dimensions, then it may easily be adaptively refined.

ACKNOWLEDGEMENTS

The help of G. Erlebacher of the Computational Methods Branch at NASA Langley is greatly appreciated for his sharing of knowledge and ideas in differential geometry.

REFERENCES

1. MAVRIPLIS, D. J., and JAMESON, A., "Multigrid Solution of the 2-D Euler Equations on Unstructured Triangular Meshes", *AIAA paper 87-0353*, January, 1987.
2. GILES, M., "Accuracy of Node-Based Solutions on Irregular Meshes", *Proc. of the 11th Int. Conf. on Num. Meth. in Fluid Dynamics*, Eds D. Dwoyer, Y. Hussaini, R. Voight, Springer Verlag, June, 1988
3. LOHNER, R., MORGAN, K., PERAIRE, J., VAHDATI, M., "Finite Element Flux-Corrected Transport for the Euler and Navier-Stokes Equations", *Int. J. Num. Meth. Fluids* Vol. 7, 1987, pp. 1093-1109
4. JAMESON, A., BAKER, T. J., and WEATHERILL, N. P., "Calculation of Inviscid Transonic Flow over a Complete Aircraft", *AIAA paper 86-0103*, January 1986.
5. STOUFFLET, B., PERIAUX, J., FEZOUI, F., and DERVIEUX, A., "Numerical Simulation of 3-D Hypersonic Euler Flows Around Space Vehicles Using Adapted Finite Elements", *AIAA paper 87-0560* January 1987.
6. WEATHERILL, N. P., JOHNSTON, L. J., PEACE, A. J., and SHAW, J. A., "Development of a Method for the Solution of the Reynolds-Averaged Navier-Stokes Equations on Triangular Grids", *Aircraft Research Association Ltd. Report 70*, Bedford, UK, December, 1986.
7. NAKAHASHI, N., "FDM-FEM Zonal Approach for Viscous Flow Computations Over Multiple Bodies", *AIAA paper 87-0604*, January, 1987.
8. PERAIRE, J., VAHDATI, M., MORGAN, K., and ZIENKIEWICZ, O. C., "Adaptive Remeshing for Compressible Flow Computations", *J. Comp. Phys.*, Vol 72, October, 1987, pp. 449-466
9. MAVRIPLIS, D. J., "Accurate Multigrid Solution of the Euler Equations on Unstructured and Adaptive Meshes", *Proceedings of the First National Fluid Dynamics Congress, AIAA paper 88-3706*, July, 1988.
10. HOLMES, D. G., and LAMSON, S. H., "Adaptive Triangular Meshes for Compressible Flow Solutions", *Proc. of the Int. Conf. on Numerical Grid Generation in Computational Fluid Dynamics*, Edited by J. Hauser and C. Taylor, Pineridge Press Ltd., July 1986, pp. 413-424
11. WEATHERILL, N. P., "The Generation of Unstructured Grids Using Dirichlet Tessalations" *Princeton University MAE Report No. 1715*, July 1985
12. BAKER, T. J., "Three Dimensional Mesh Generation by Triangulation of Arbitrary Point Sets", *Proc. of the AIAA 8th Comp. Fluid Dyn. Conf., AIAA paper 87-1124*, June, 1987
13. SIBSON, R., "Locally Equiangular Triangulations", *The Computer Journal*, Vol 21, No. 3, 1978, pp. 243-254
14. BOWYER, A., "Computing Dirichlet Tessalations", *The Computer Journal*, Vol. 24, No. 2, 1981, pp. 162-166
15. LAWSON, C. L., "Generation of a Triangular Grid with Application to Contour Plotting", *Cal. Tech. Jet Propulsion Lab. Tech Memorandum 299*, 1972
16. EISEMAN, P., AND BOCKELIE, M., "Adaptive Grid Solution for Shock-Vortex Interaction", *Proc. of the 11th Int. Conf. on Num. Meth. in Fluid Dynamics*, Eds D. Dwoyer, Y. Hussaini, R. Voight, Springer Verlag, June, 1988

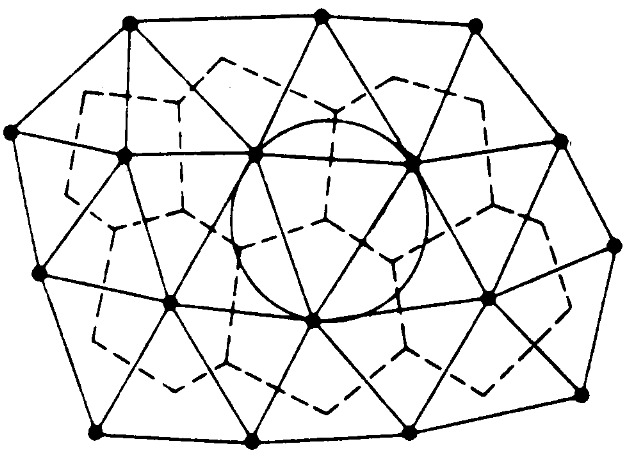

Figure 1
Dirichlet Tessalation and Delaunay Triangulation of a Set of Points
Showing the Circumcircle of One of the Triangles

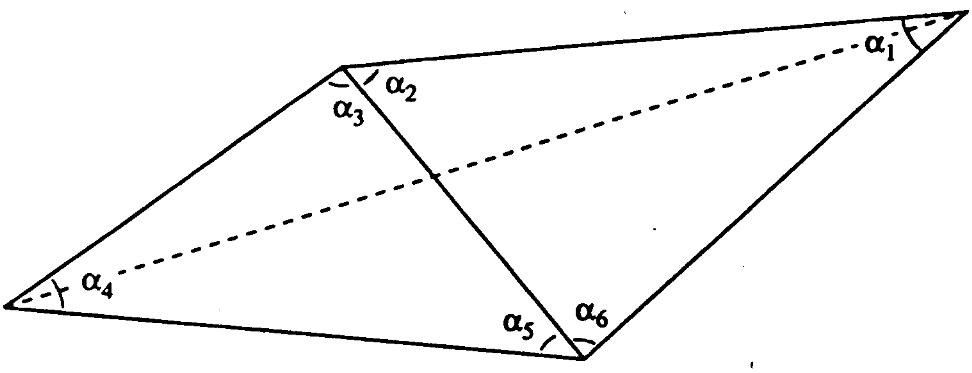

Figure 2
The Two Possible Configurations for the Diagonal in a Convex Quadrilateral
and the Six Angles Associated with the Most Equiangular
Configuration (Solid Line Diagonal)

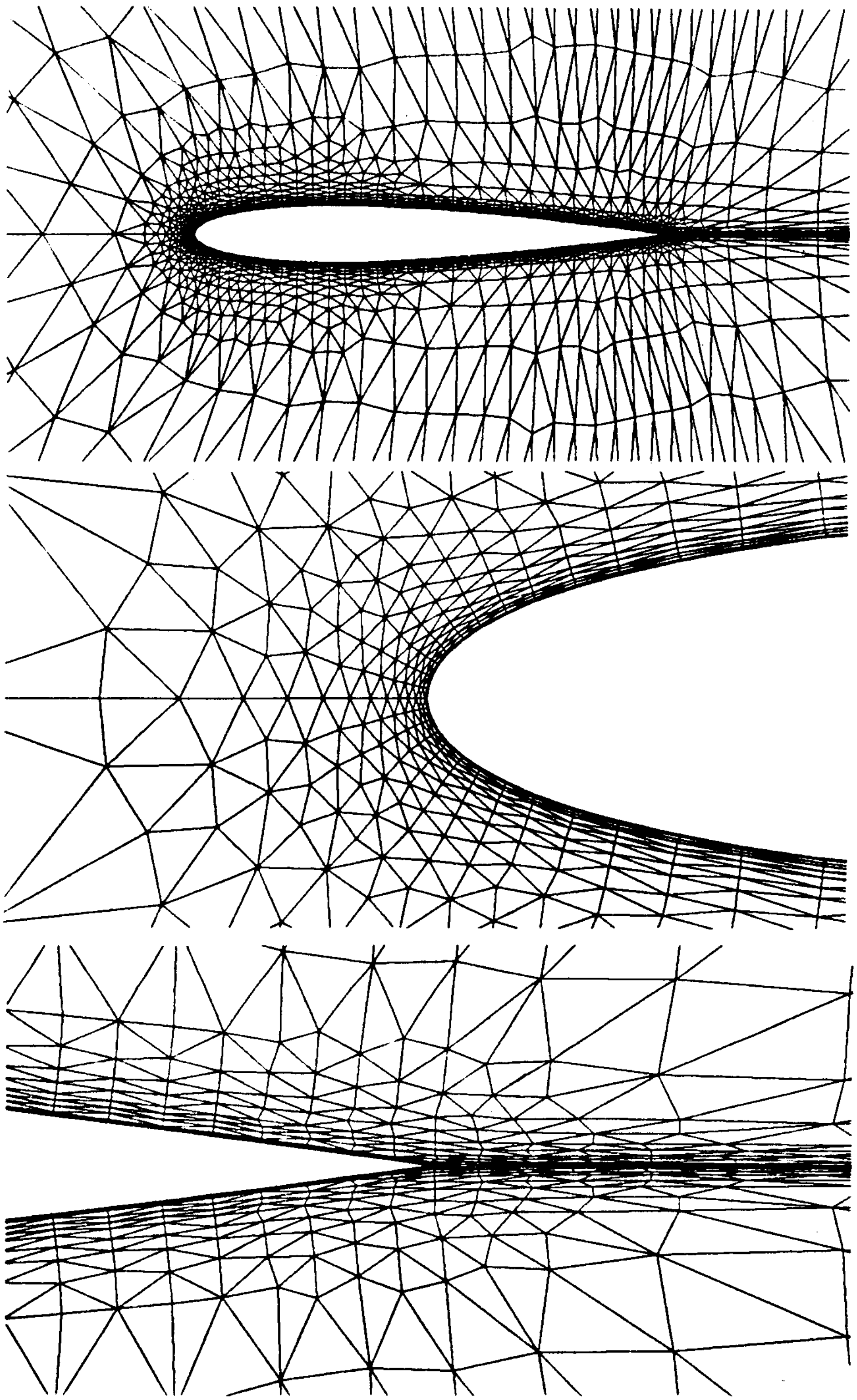

Figure 3
Illustration of the Adaptively Generated Stretched Triangulation about
a NACA 0012 Airfoil Including Details at the Leading and Trailing Edges
(Number of Points = 2316)

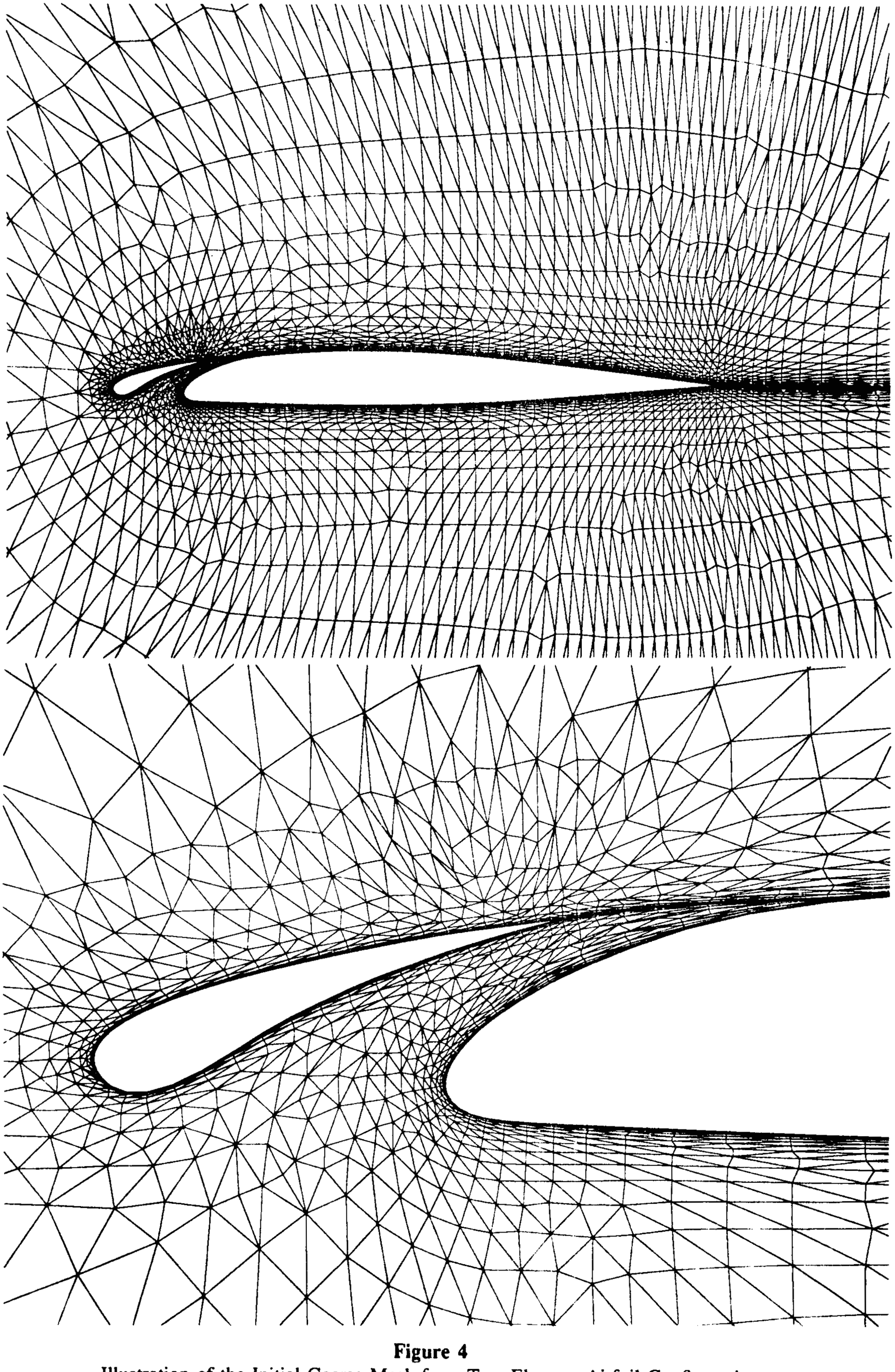

Figure 4
Illustration of the Initial Coarse Mesh for a Two-Element Airfoil Configuration
Including Details of the Mesh in the Gap Region
(Number of Points = 5856)

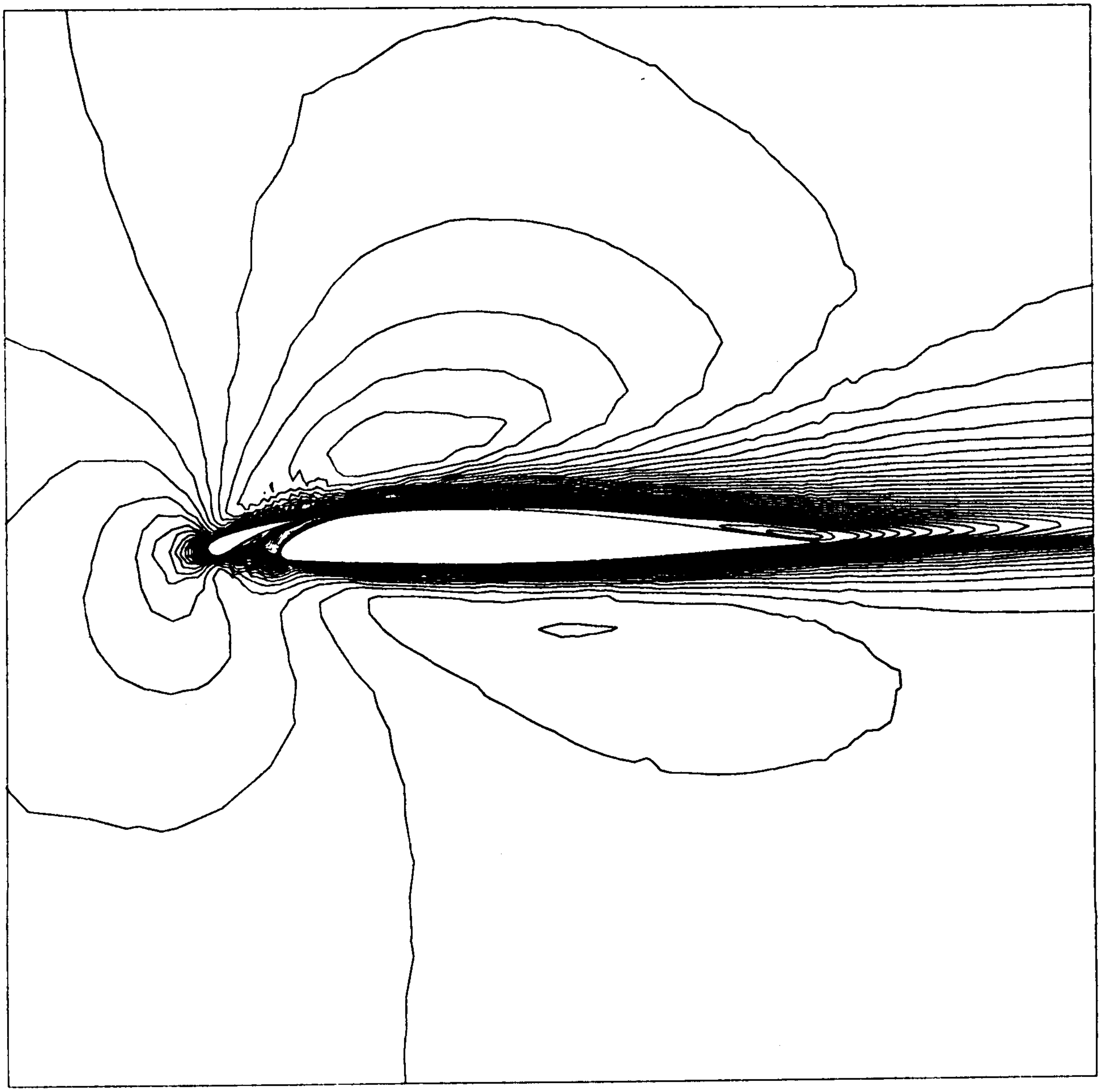

Figure 5
Mach Contours in the Flow-Field Obtained with the Navier-Stokes Solution
on the Initial Coarse Mesh for the Two-Element Airfoil Configuration
Mach = 0.5, Incidence = 3°, Re = 5000

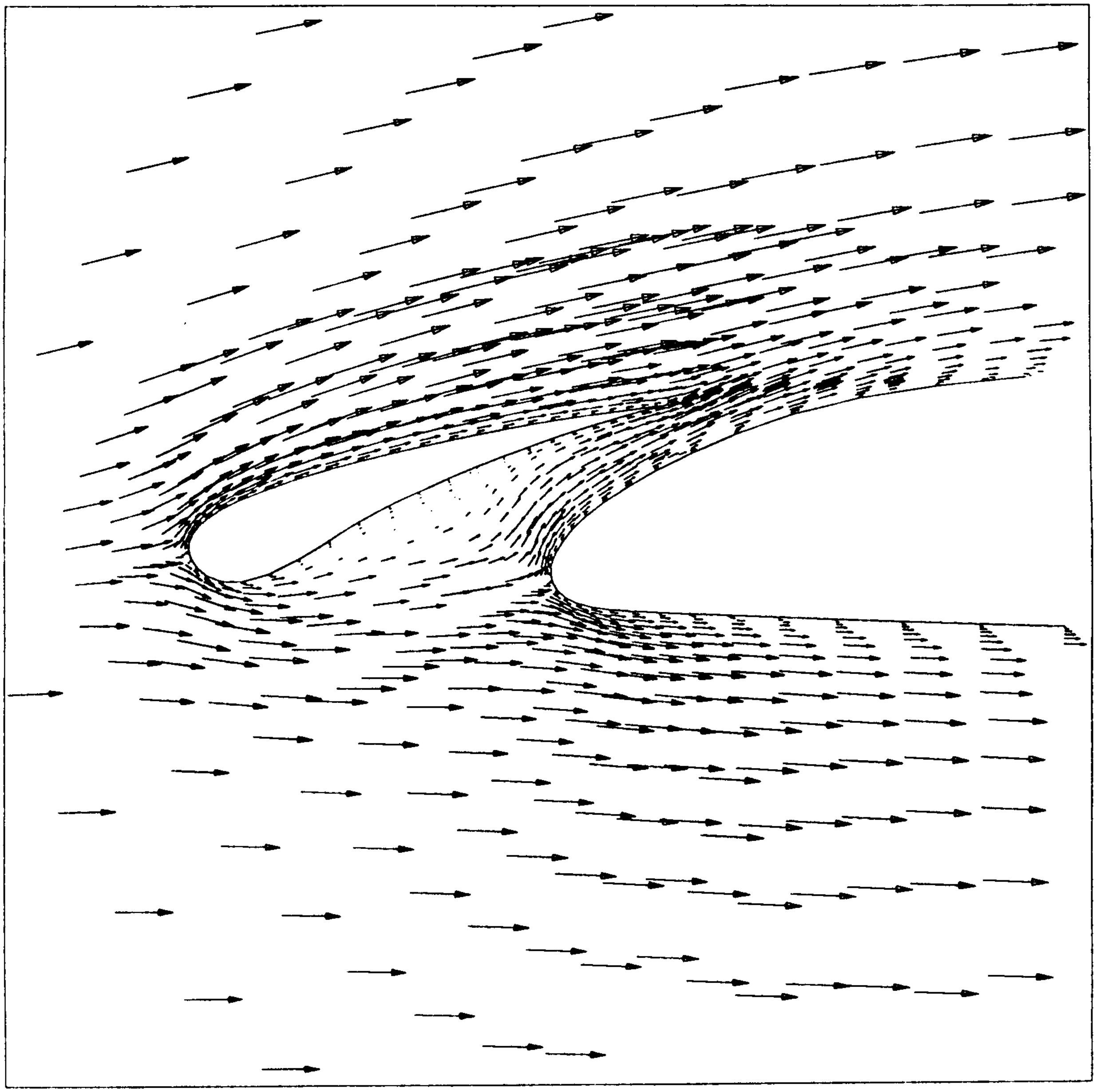

Figure 6
Vector Velocities in the Gap Region of the Two-Element Airfoil
Configuration Computed on the Initial Coarse Mesh
Mach = 0.5, Incidence = 3°, Re = 5000

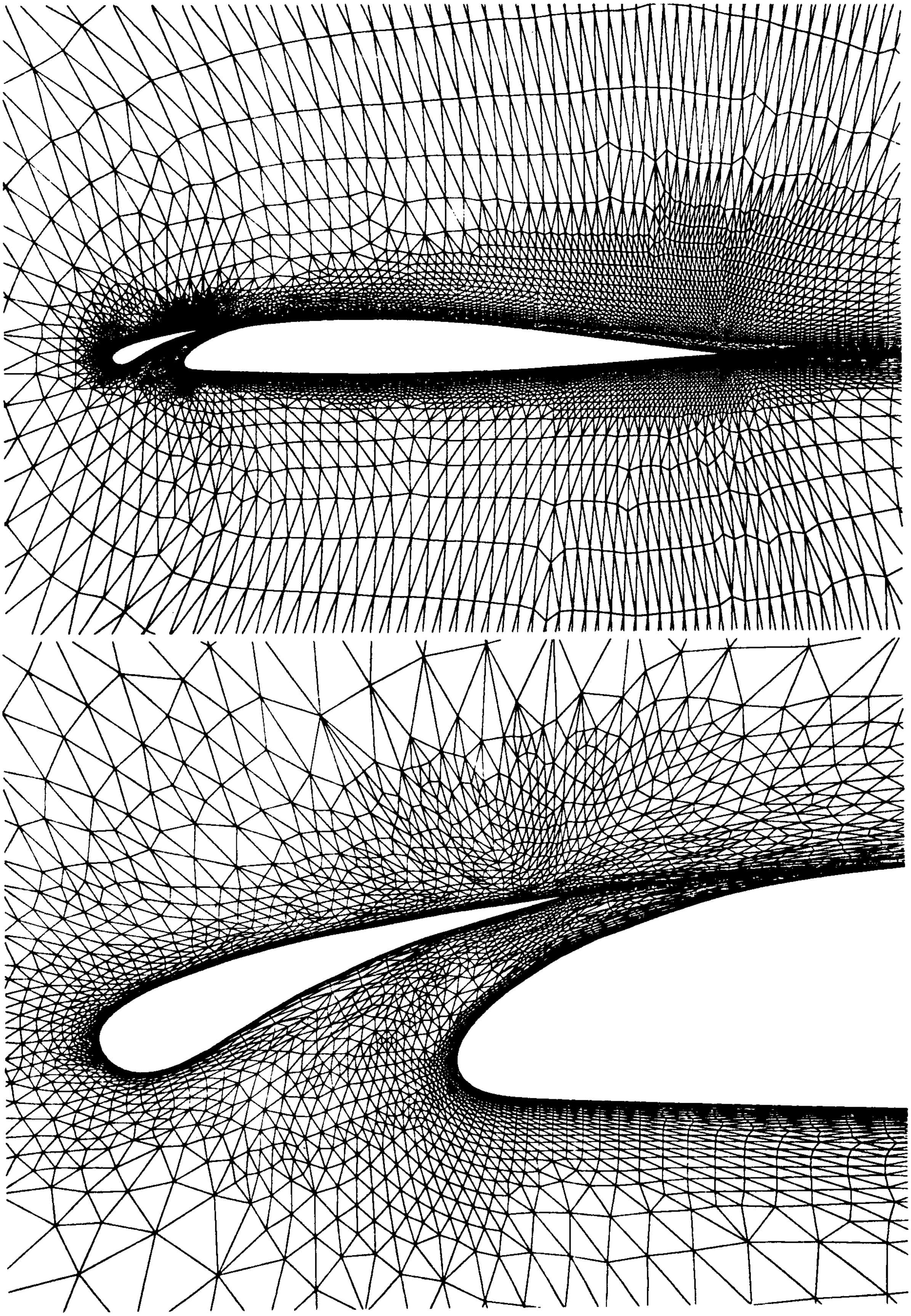

Figure 7
Illustration of the Adaptively Refined Mesh for the Two-Element Airfoil Configuration Including Details of the Mesh in the Gap Region (Number of Points = 11377)

Report Documentation Page

NASA

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-181699 ICASE Report No. 88-47		

4. Title and Subtitle

ADAPTIVE MESH GENERATION FOR VISCOUS FLOWS USING DELAUNAY TRIANGULATION

5. Report Date

August 1988

6. Performing Organization Code

7. Author(s)

Dimitri J. Mavriplis

8. Performing Organization Report No.

88-47

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.

NAS1-18107

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

Final Report

Submitted to Proc. of the Second International Conference on Numerical Grid Generation in Computational Fluid Dynamics

16. Abstract

A method for generating an unstructured triangular mesh in two dimensions, suitable for computing high Reynolds number flows over arbitrary configurations is presented. The method is based on a Delaunay triangulation, which is performed in a locally stretched space, in order to obtain very high aspect ratio triangles in the boundary layer and wake regions. It is shown how the method can be coupled with an unstructured Navier-Stokes solver to produce a solution adaptive mesh generation procedure for viscous flows.

ORIGINAL PAGE IS
OF POOR QUALITY

17. Key Words (Suggested by Author(s))

Delaunay triangulation, adaptive

18. Distribution Statement

02 - Aerodynamics
34 - Fluid Mechanics and Heat Transfer
Unclassified - unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	15	A02