https://ntrs.nasa.gov/search.jsp?R=19930020024 2020-03-17T04:49:01+00:00Z

DEVELOPMENT OF AN UNSTRUCTURED SOLUTION ADAPTIVE METHOD FOR THE QUASI-THREE-DIMENSIONAL EULER AND NAVIER-STOKES EQUATIONS

Yi-Tsann Jiang
William J. Usab, Jr.

School of Aeronautics and Astronautics
Purdue University
West Lafayette, IN

Final Technical Report for
NASA Research Grant No. NAG3-1127

Grant Title: Develop an Adaptive Solution Method for Viscous Flows Through Advanced Compressor and Turbine Cascades

Principal Investigator: W. J. Usab, Jr.

Report Period: January 2, 1990 to August 31, 1993

PRECEDING PAGE BLANK NOT FILMED

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor and graduate committee chairman, Professor William J. Usab, Jr., for his help and guidance throughout this project. His advice and support has been essential for the successful completion of this dissertation. I would also like to express my appreciation to the other members of my committee, Professors Joe D. Hoffman, John P. Sullivan and Marc H. Williams, for their many helpful comments.

I wish to thank Dr. David P. Miller (NASA Lewis) for his support of this work and providing help in obtaining and interpreting the experimental data used in our comparison. The use of the computer facilities at NASA Lewis research center is also appreciated.

I would like to thank Dr. Shih H. Chen (Rocketdyne) for his consideration and encouragement throughout these years. I would also like to thank fellow graduate student, M. Buckley for his time in reading this thesis and for helping me tune this thesis with his helpful suggestions. Special thanks are due to my friends at Purdue, especially, C. C. Chang, Dr. S. M. Swei, Dr. C. J. Jih and Dr. S. W. Lee, for their friendship and moral support.

I would like to express my sincerest appreciation to my wife, Teresa. Through her support, understanding, and love she made the completion of this work possible. Last, but certainly not least, I would like to thank my parents for their encouragement and support through the years I've been in school.

This work has been supported by the National Aeronautics and Space Administration at Lewis under grant NAG-3-1127, Dr. David P. Miller, technical monitor.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Figure | Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Page

Figure

NOMENCLATURE

m	Meridional distance
θ	Angular distance
r	Radius
z	Distance along turbomachine axis
h	Streamtube height
L	Referenced length scale
t	Time
ρ	Density
V_m, V_θ	m and θ components of absolute velocity
W_θ	Relative angular velocity
Ω	Rotational velocity
E	Internal energy per unit volume
p	Pressure
T	Temperature
$\vec{U}$	State variable vector
$\vec{F}, \vec{G}$	Inviscid m- and θ-flux vectors
$\vec{R}, \vec{S}$	Viscous m- and θ-flux vectors
$\vec{K}$	Source term in quasi-three-dimensional equations
M	Mach number
α	Angle of attack
β	Inlet and exit flow angles
σ	Shear stress
$\tau_{\mathbf{wall}}$	Wall shear stress
c_p	Specific heat at constant pressure

γ	Ratio of specific heats
Pr	Prandtl number
κ_T	Thermal conductivity coefficient
$\vec{R}$	Residual vector
$\vec{Q}$	Flux balance vector
Re_L	Reynolds number for a reference length L
$\vec{V}$	Velocity vector
a	Speed of sound
c_2	Viscosity law constant
μ	molecular viscosity
h_o	Stagnation enthalpy
∂	Partial derivative
Δt	Time step increment
Δm	Spatial increment in m-component
$\Delta\theta$	Spatial increment in θ-component
$\Delta\vec{U}$	Change in solution
ΔV	Volume of a computational grid cell
α_i	Runge-Kutta multi-stage coefficients
CFL	Courant-Friedrichs-Lewy number
λ	Eigenvalues of inviscid flux Jacobian matrix
I	Rothalpy
δ_{ij}	Kronecker delta function
∇	Gradient operator
∇^2	Laplacian operator
ϵ^2, ϵ^4	Artificial dissipation coefficients
ϕ	General scalar
$\mathcal{O}$	Order of magnitude
$\langle\rangle$	Ensemble-averaged operator
κ	Turbulence kinetic energy

ϵ	isotropic turbulent kinetic energy dissipation rate
y^+	Wall units in turbulence modeling parameters
Subscripts	
V	Volume integral
∂V	Boundary integral
b	Values at boundaries
ex	Quantities extrapolated from the domain
pr	Specified quantities
is	Isentropic quantities
j, k	Cell indices
n, s	Quantities in local normal and tangential directions
ξ, η	Quantities in curvilinear coordinate directions
x, y	Derivative with respect to spatial directions
∞	Conditions at infinity
in	Conditions at the inlet
exit	Conditions at the exit
wall	Conditions at wall boundary
turbulent	Turbulent quantities
laminar	Laminar quantities
e	Effective quantities
Superscripts	
n	Time level
$'$	Absolute total conditions
$''$	Relative total conditions
$\bar{()}$	Non-dimensional quantities
$\tilde{()}$	Favre-averaged quantities

ABSTRACT

Jiang, Yi-Tsann. Ph.D., Purdue University, May 1993. Development of an Unstructured Solution Adaptive Method for the Quasi-Three-Dimensional Euler and Navier-Stokes Equations. Major Professor: Dr. William J. Usab, Jr.

A general solution adaptive scheme based on a remeshing technique is developed for solving the two-dimensional and quasi-three-dimensional Euler and Favre-averaged Navier-Stokes equations. The numerical scheme is formulated on an unstructured triangular mesh utilizing an edge-based pointer system which defines the edge connectivity of the mesh structure. Jameson's four-stage hybrid Runge-Kutta scheme is used to march the solution in time. The convergence rate is enhanced through the use of local time stepping and implicit residual averaging. As the solution evolves, the mesh is regenerated adaptively using flow field information. Mesh adaptation parameters are evaluated such that an estimated local numerical error is equally distributed over the whole domain. For inviscid flows, the present approach generates a complete unstructured triangular mesh using the advancing front method. For turbulent flows, the approach combines a local highly stretched structured triangular mesh in the boundary layer region with an unstructured mesh in the remaining regions to efficiently resolve the important flow features. One-equation and two-equation turbulence models are incorporated into the present unstructured approach. Results are presented for a wide range of flow problems including two-dimensional multi-element airfoils, two-dimensional cascades, and quasi-three-dimensional cascades. This approach is shown to gain flow resolution in the refined regions while achieving a great reduction in the computational effort and storage requirements since solution points are not wasted in regions where they are not required.

1. INTRODUCTION

The flow in axial and radial compressors and turbines is complex both in terms of the geometry of advanced designs and in terms of the flow structures that are encountered. New designs incorporating radically shaped blades, splitter plates, and even multi-element blade configurations lead to even more complex flow problems. Early analysis of these flow problems was done mainly by experimental methods. Modern experimental facilities utilizing advanced measurement techniques can provide detailed flow information for these flow problems. However, it is costly to perform experimental studies in the design process. On the other hand computer technology has seen a rapid development and the cost has decreased dramatically over the last two decades. This has led to significant developments in the area of computational fluid dynamics (CFD). A variety of solution procedures have been proposed for the solution of the Euler and Navier-Stokes equations. Many of these methods are restricted to relatively simple geometries and are difficult to employ in turbomachinery applications. It is therefore of prime interest to establish a general solution scheme for turbomachinery applications.

The present research develops a general solution adaptive method for geometrically complex domains and complex flow structures. This approach provides a flexible framework for turbomachinery applications. It is important that this approach is accurate, efficient, and easily applicable to a wide range of designs. A general solution adaptive method involves a combination of mesh generation techniques, solution algorithms, and solution adaptive techniques. In the following section recent developments in these key areas are reviewed.

1.1 Background

In the area of CFD a majority of the flow solvers have been developed for body-fitted structured meshes. Efficient algorithms can be achieved using the body-fitted mesh line information. Many fast and efficient solution procedures have been proposed to the solution of flow for Euler [2, 14, 33, 37, 38, 65] and Navier-Stokes [19, 23, 53, 92] equations on structured meshes. However, applying these schemes to turbomachinery applications is difficult due to the problem of generating a structured mesh within a complex domain. The problem lies in the generation of a global body-fitted mesh which maps to a logical rectangle in computational space while satisfying a complex set of conflicting constraints in physical space.

Two approaches have been proposed to alleviate this difficulty. One is to keep the structured solver and simplify the mesh generation problem. Proposed techniques include the use of Cartesian meshes [11, 12], overlaid or composite meshes [5], and patched meshes [68, 72]. The use of a Cartesian mesh simplifies the problem of mesh generation by abandoning the requirement that mesh boundaries conform to body surfaces. This however increases the complexity of boundary condition formulations in the flow solver. It can also lead to clustering of mesh in uniform flow regions. In the overlapping approach several subdomain grids are overlaid together reducing the problem to a simpler mesh generation problem within each subdomain. The necessary interpolation between overlaid meshes requires a special data structure and increases the computing time. It is also difficult to automate such a procedure. In a patched approach the flow field is subdivided into a series of simpler subdomains with mesh generation performed on each block. This simplifies the mesh generation problem for particular geometries, but it does not eliminate the problem and is difficult to automate. The final mesh depends on the user's experience and skill. The work of planning block subdivisions becomes a major obstacle in the development of computational design tools. Although these techniques have been successfully used in many applications, for the complex flow structures encountered in advanced turbomachinery they are difficult to generalize and add to the complexity of the flow solver.

A second approach is to totally abandon the structured mesh and to formulate the problem using an unstructured triangular mesh. Since any distribution of points can be meshed with a triangular mesh, this approach eliminates any conflict between constraints imposed to implement boundary conditions and those required to resolve the complex flow structures. Two approaches are most commonly used for generation of unstructured meshes: Delaunay triangulation methods [31, 58, 87, 88] and advancing front methods [48, 64, 67]. Delaunay triangulation begins with a few super-large triangles covering the domain of interest. Mesh points are then added, one by one, with a retriangulation of the mesh. Retriangulation is performed using the criterion that any point can not fall inside of a circle determined from any other three points of the existing mesh. This particular method results in an "optimal" triangulation. However, the work to perform the mesh generation requires $\mathcal{O}(N^2)$ operations [87] because the sorting process for the triangulation is usually performed over all points. In addition, this method does not provide control over the mesh point distribution.

In the advancing front method, the mesh point locations are determined as part of the mesh generation process. Starting from an initial front defined by boundary segments, new points are added and triangulated into the front. This process is repeated until the complete domain is triangulated. Mesh parameters, such as mesh size, aspect ratio and stretched direction, can be specified over the domain in this approach. Triangles are generated using the criterion that a new triangle should not cross any given face of the front. This method provides a great deal of control over the resulting mesh distribution. For turbulent flow calculations a highly stretched mesh is used to achieve computing efficiency. Recently, Hassan *et al.* [30] noted that the advancing front method can only produce a maximum allowable mesh stretching of about 10 in order to preserve mesh quality.

Initial algorithm work for solving flow problems on unstructured meshes was done in the finite element community. In the finite element approach, numerical schemes are formulated on an element basis, with the element formulation generally requiring no information from other elements. Angrand *et al.* [3] and Morgan *et al.* [51] have demonstrated the use of unstructured finite-element flow solvers for two-dimensional flow problems. These schemes use a finite difference scheme in the temporal discretization and a finite element formulation in the spatial discretization.

A second approach to solving these equations, finite-volume methods, is based on a discrete approximation of the integral form of the governing equations. Jameson and Mavriplis [36] have demonstrated the extension of Jameson's multi-grid Runge-Kutta scheme [33] on regular triangles over a airfoil. Both finite-element and finite-volume approaches have been successfully demonstrated for the solution of the Euler equations using unstructured triangles in two dimensions [58, 64, 8, 90] and tetrahedras in three dimensions [35, 49, 66, 78]. Extension of unstructured schemes to the Navier-Stokes equations has recently been done for two-dimensional turbulent flow problems [7, 56, 59]. Finite-volume methods solve the physical conservation laws directly. For turbomachinery applications the accurate prediction of the mass flow is critical to obtaining an accurate solution. Therefore, a finite-volume formulation is more appropriate. Although the unstructured approach provides a simple and flexible framework for solving complex flow problems, it is computationally inefficient because of the need for the mesh connectivity information and they are very difficult to vectorize. It is also very difficult to implement solution acceleration methods because these techniques often assume a structured connection between mesh points.

Viscous flow problems are difficult to solve on an unstructured mesh due to the turbulence models by which closure of Favre or Reynolds averaged Navier-Stokes equations is achieved. For structured mesh solvers, algebraic or zero equation models are the simpliest and easiest models to implement. Algebraic models do not have numerical stability problems and work well for a wide range of engineering applications. Unfortunately, algebraic models require length scale information to compute turbulence quantities. The lack of body-fitted mesh line information on unstructured meshes makes it difficult to implement algebraic turbulence models. An overlaid mesh technique has been proposed to overcome this difficulty [56, 71]. In this approach local structured meshes are overlaid with a global unstructured mesh. The algebraic turbulence model is then solved on the structured mesh. The necessary interpolation between overlaid meshes requires a special data structure and increases the computing time. Moreover the use of local structured meshes restricts the flexibility of the method. In order to remove the structured mesh dependence, more complicated turbulence models which solve one or more transport equations for turbulence quantities can be used. Recently, Barth [7] has successfully demonstrated the Bladwin-Barth one-equation turbulence model [6] on a wide range of turbulent flow problems using the unstructured mesh approach. Among two-equation models, Chien's low Reynolds $k-\epsilon$ turbulence model [17] has been widely used in engineering applications [42].

Without a priori knowledge of the flow structure, neither the structured approach nor the unstructured approach can accurately and efficiently resolve the flow. While in principle a global fine mesh can be used to accurately resolve any flow structure, such an approach is impractical and computationally expensive. This has led to the development of solution adaptive methods. Solution adaptive methods can be divided into three general approaches: mesh refinement or enrichment, mesh movement, and mesh regeneration. Each type has advantages and disadvantages associated with it.

In the mesh refinement approach mesh points are added or removed from the solution domain either by subdivision or absorption of mesh elements. Dannenhoffer [21] has demonstrated this approach for a series of airfoil problems using quadrilateral unstructured meshes. Starting with an initial structured mesh, irregularly-shaped embedded mesh regions are generated by subdividing the cells in high gradient flow regions. This approach has been very successful in resolving complex flow structures. Although it is not necessary to have a good initial mesh, the use of a structured initial mesh constrains the problem. If skewed cells appear in the initial mesh, such properties will remain in the refined mesh. Löhner [47] on the other hand used mesh refinement on unstructured triangular meshes. This approach locally enrichs the mesh by subdividing triangular mesh elements. After refinement, any badly-formed cells are removed to improved the resulting mesh. The mesh refinement approach is very efficient, but it has the disadvantage of the significant bookkeeping involved in keeping track of modifications to the mesh. In addition, in both of the above formulations the adapted mesh has discontinuous variations in cell length scale since subdivisions are integer divisions of the original mesh.

In the solution adaptation method based on mesh movement, the mesh point connectivity is fixed and the points are moved as the solution evolves. This has the advantage that any existing solver can be applied with minimal modification. To move mesh points in the structured mesh Gnoffo [29] uses an equivalent spring analogy, in which the mesh edges are replaced by springs with a stiffness based on the local gradient of some flow property. Löhner [50] and Batina [9] extended this approach to unstructured meshes. The disadvantage of this technique is that the final mesh depends on the initial mesh connectivity.

In the mesh regeneration method, the mesh is regenerated periodically as the solution evolves. This may be expensive due to work required to generate the mesh but has many advantages. For structured meshes, mesh points are redistributed using structured mesh generation techniques as the solution evolves [22, 82]. In practice, the number of mesh points may be fixed so it has the same advantages as the mesh movement technique. This technique provides smooth distribution of mesh lines, but it is difficult to use on complex geometries. In the unstructured approach, Peraire *et al.* [67] introduced a remeshing process in which an unstructured mesh is regenerated using mesh parameters determined from the most recent solution. This approach provides smooth variation in mesh length scale and allows dense points to be placed in high gradient flow regions. Mavriplis [57] and

Holmes [31] demonstrated the use of Delaunay triangulation with mesh point insertion. Since Delaunay triangulation needs to search a retriangulation region when a new point is inserted into domain, for efficiency it requires special data structure to perform local retriangulation.

The preceding review shows that mesh generation for complex domains can be easily obtained through the use of unstructured meshes. An accurate prediction of complex flow problems can be effectively achieved using solution adaptive methods. Even for problems which can be solved using structured meshes (e.g., patched or overlaid meshes), there is often a significant reduction in the human effort required using unstructured meshes. An additional advantage of the unstructured approach is that it provides a convenient framework for implementing solution adaptive methods. This leads to a great reduction in the computational effort and storage requirements since solution points are not wasted in regions where they are not required.

1.2 Present Approach

The solution adaptive approach used in the present work is based on mesh regeneration, where the mesh is periodically regenerated as the solution evolves. While this is a more computationally intensive approach, it also has many advantages. There is very little bookkeeping required since the mesh structure is not being modified. Regeneration of the mesh results in a smoothly varying distribution of mesh points, which in turn should give better numerical solutions. Remeshing allows the opportunity to align the mesh with flow structures which, in turn, makes it possible to use different mesh scalings tangent and normal to a given flow structure. A shock wave is a good example of such a flow structure, since to accurately capture a shock wave the mesh scale normal to the shock wave must be small. The ability to align the mesh will also be very important in the extension of the present approach to viscous flow where the mesh may also be aligned with viscous shear layers and in the incorporation of flux splitting for improved resolution of shock waves.

For turbulent flow calculations the level of complexity of the models determines computing expense, so the turbulence models used are the one-equation and two-equation models. Incorporating such models in an unstructured approach is still a very new topic and only a few approaches have been proposed [7, 59]. In those proposed approaches, turbulence models are solved using an implicit scheme which usually requires an inversion of a large matrix system of equations. In the present study, the one-equation and two-equation turbulence models are discretized using the same explicit scheme employed on the mean flow equations.

In summary, the key elements in the present approach are the unstructured flow solver, the mesh generation scheme, and the adaptive remesh algorithm with associated refinement criteria. Jameson's four-stage Runge-Kutta cell-vertex finite-volume time-marching scheme [37] is used to solve the quasi-three-dimensional Favre averaged Navier-Stokes equations and turbulence transport equations. The convergence rate is enhanced through the use of local time stepping. The quasi-three-dimensional equations are chosen here because they provide a better approximation to the three-dimensional flow while retaining a two-dimensional form. In addition, quasi-three-dimensional equations can be simplified into standard two-dimensional equations. This provides a more universal and convenient model for general flow problems. The mesh generation method with which unstructured triangular meshes are generated is the advancing front scheme first formulated by Peraire *et al.* [67]. This particular approach has the advantage of being computationally efficient and also provides a convenient way of adapting the mesh distribution to the flow solution. For viscous flows in the present work local structured triangular meshes are generated around bodies to relieve the stretching limit on the unstructured mesh generator.

The quasi-three-dimensional flow model, the governing equations, and the Baldwin-Barth [6] one-equation and Chien's [17] two-equation turbulence models are introduced in Chapter 2. The unstructured flow solver using a multistage Runge-Kutta finite-volume time-marching scheme is described in Chapter 3. The stability criteria, local time-stepping, and implicit residual averaging are developed. A modified version of artificial dissipation for highly stretched meshes and boundary conditions for two-dimensional airfoil and quasi-three-dimensional cascade flow problems are also discussed. The mesh generation procedure for both inviscid and viscous flow problems is presented in Chapter 4. The solution remeshing scheme and mesh adaptation criteria are described in detail in Chapter 5. In Chapter 6 the numerical results of inviscid flow problems are presented.

Unstructured solution adaptive results for the two-dimensional Euler equations are presented for a model multi-element airfoil, a Sanz's supercritical compressor blade, and a Sanz's turbine blade. Computed solutions are compared to the analytic solutions. Quasi-three-dimensional Euler solutions are illustrated for the NACA Rotor 67 transonic fan operating at peak efficiency and the Allison tandem blade cascade. In Chapter 7 unstructured solution adaptive results for turbulent flow problems are presented. Two-dimensional turbulent flow solutions for a subsonic flat plate, a RAE2822 airfoil, a NLR two-element airfoil, and a VKI turbine blade configurations are presented. Numerical results are compared to the available experimental data. A quasi-three-dimensional Favre averaged solution is presented for the Allison tandem blade cascade.

2. GOVERNING EQUATIONS

This chapter reviews the quasi-three-dimensional flow approximation for turbomachinery applications. The governing equations are derived from the conservation of mass, momentum, and energy for viscous flows on a blade-to-blade stream surface. The mean flow equations and turbulence models for turbulent flow calculations are also presented. With the use of the equation of state, the constant Prandtl number approximation, and viscosity models, the complete set of governing equations is obtained. Initial and physical boundary conditions for the flows of interest are also described.

2.1 Quasi-Three-Dimensional Flow Model

In general, the flows in axial, radial, and mixed-flow turbomachinery designs are highly three-dimensional. In order to solve these three-dimensional flows in a relative simple manner, Wu [91] proposed the following simplification which allows these types of flows to be analyzed two dimensionally, but with more information provided. In this model, the three-dimensional flow field is described by the combination of two separate two-dimensional flow fields as sketched in Figure 2.1. These two separate flow fields are composed of surfaces located in the blade-to-blade direction (S1 surfaces) and surfaces lying in the hub-to-tip direction (S2 surfaces). In practice the flow is further assumed to follow an axisymmetric streamsurface as shown in Figure 2.2. The solution to the coupled two-dimensional flow fields requires iteration between the solution of the throughflow problem on a mean S2 streamsurface and several quasi-three-dimensional solutions on axisymmetric S1 streamsurfaces, since the solution for either surface requires the knowledge of the shape of the other surface. In the present research, only the flows within the quasi-three-dimensional S1 surfaces are investigated. The radial location and thickness of a streamsurface is assumed to be known as a function of streamwise distance along the surface. This information is obtained from an axisymmetric through-flow analysis (see [41]).

2.2 Quasi-Three-Dimensional Navier-Stokes Equations

The quasi-three-dimensional viscous compressible flow along the S1 streamsurface is expressed in terms of an axisymmetric coordinate system (m, θ) (see Fig. 2.2) which rotates with the blade row and is given by

$$dm^2 = dz^2 + dr^2 \tag{2.1}$$

and

$$\theta = \theta' - \Omega t \tag{2.2}$$

where θ' is fixed in space and Ω is the angular velocity of the blade row. In this coordinate system, with radius r and streamsurface thickness h taken as known functions of m, the Navier-Stokes equations may be expressed in the following form [18]:

$$\frac{\partial(rh\vec{U})}{\partial t} + \frac{\partial(rh\vec{F})}{\partial m} + \frac{\partial(h\vec{G})}{\partial \theta} - (\frac{\partial(rh\vec{R})}{\partial m} + \frac{\partial(h\vec{S})}{\partial \theta}) = rh\ \vec{K} \tag{2.3}$$

where

$$\vec{U} = \begin{pmatrix} \rho \\ \rho V_m \\ \rho V_\theta r \\ \rho E \end{pmatrix}, \quad \vec{F} = \begin{pmatrix} \rho V_m \\ \rho V_m^2 + p \\ \rho V_m V_\theta r \\ V_m(\rho E + p) \end{pmatrix}, \quad \vec{G} = \begin{pmatrix} \rho W_\theta \\ \rho V_m W_\theta \\ (\rho V_\theta W_\theta + p)r \\ W_\theta(\rho E + p) + r\Omega p \end{pmatrix}$$

$$\vec{R} = \begin{pmatrix} 0 \\ \sigma_{11} \\ \sigma_{12} r \\ R_4 \end{pmatrix}, \quad \vec{S} = \begin{pmatrix} 0 \\ \sigma_{12} \\ \sigma_{22} r \\ S_4 \end{pmatrix}, \quad \vec{K} = \begin{pmatrix} 0 \\ K_2 \\ 0 \\ 0 \end{pmatrix}$$

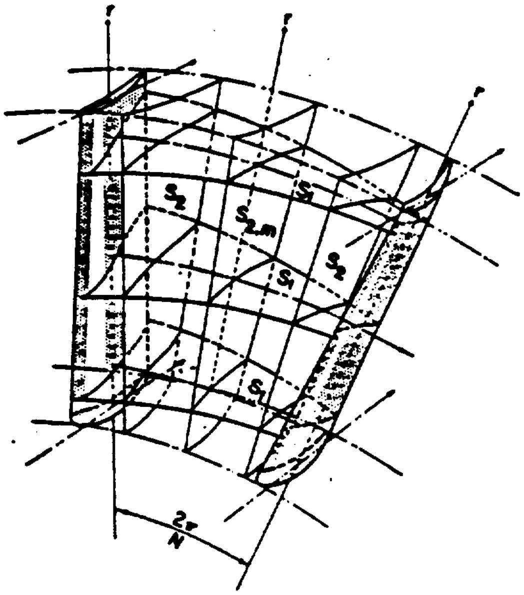


Figure 2.1 S1 and S2 streamsurfaces for Wu's quasi-three-dimensional flow model (Wu [91]).

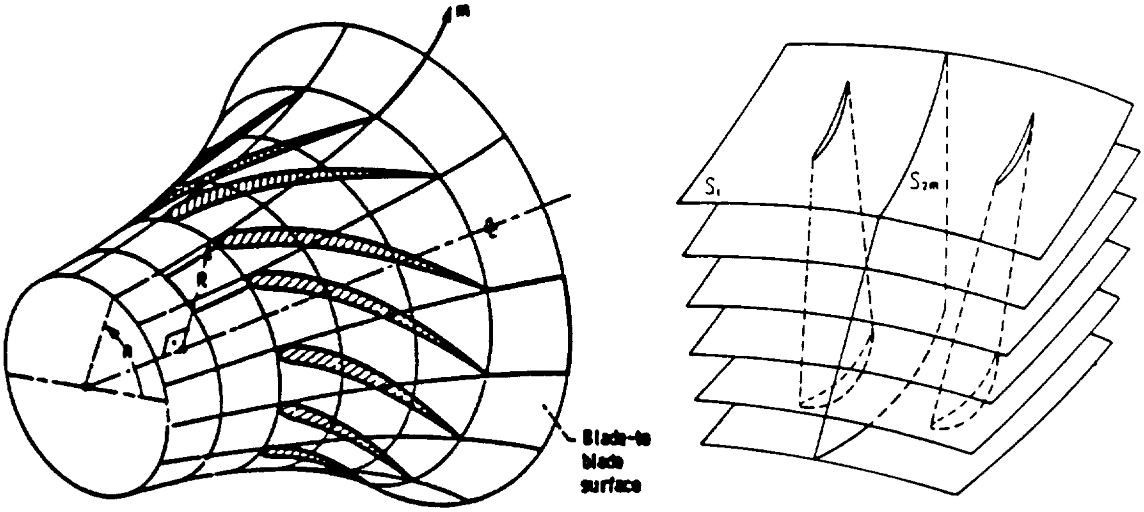


Figure 2.2 Quasi-three-dimensional coordinate system (m, θ) and streamsurface (Katsanis [41] and Wang [86]).

where

$$W_\theta = V_\theta - r\Omega$$

$$K_2 = (\rho V_\theta^2 + p - \sigma_{22})(\frac{1}{r}\frac{dr}{dm}) + (p - \sigma_{33})(\frac{1}{h}\frac{dh}{dm})$$

In these equations ρ, p, E, V_m, and V_θ are density, pressure, energy, m- and θ- absolute velocity components, respectively, and W_θ is the relative tangential velocity. The pressure p is defined by the equation of state for a perfect gas

$$p = (\gamma - 1)\rho[E - \frac{1}{2}(V_m^2 + V_\theta^2)] \tag{2.4}$$

where γ is the ratio of specific heats. The viscous terms in the energy equation are defined as follows:

$$\begin{aligned} R_4 &= (\kappa_\tau \partial_m T + V_m \sigma_{11} + V_\theta \sigma_{12}) \\ S_4 &= (\frac{\kappa_\tau}{r}\partial_\theta T + V_m \sigma_{12} + V_\theta \sigma_{22}) \end{aligned} \tag{2.5}$$

The coefficient of thermal conductivity, κ_τ, may be expressed as:

$$\kappa_\tau = \frac{C_p \mu}{p_r}$$

The shear stress terms are given by:

$$\begin{aligned} \sigma_{11} &= 2\mu \partial_m V_m + \lambda \nabla \cdot \vec{V} \\ \sigma_{22} &= 2\mu(\partial_\theta V_\theta + V_m \frac{dr}{dm})/r + \lambda \nabla \cdot \vec{V} \\ \sigma_{33} &= 2\mu V_m (\frac{1}{h}\frac{dh}{dm}) + \lambda \nabla \cdot \vec{V} \\ \sigma_{12} &= \mu(\partial_m V_\theta - V_\theta(\frac{1}{r}\frac{dr}{dm}) + \frac{1}{r}\partial_\theta V_m) \end{aligned}$$

In the above expressions, λ and μ are the two coefficients of viscosity which based on Stokes' hypothesis, $\lambda = -2\mu/3$, T is the static temperature, C_p is the specific heat at constant pressure, and p_r is the Prandtl number. The dilitation is given by

$$\lambda \nabla \cdot \vec{V} = \frac{-2\mu}{3}[\partial_m V_m + V_m(\frac{1}{r}\frac{dr}{dm} + \frac{1}{h}\frac{dh}{dm}) + \frac{1}{r}\partial_\theta V_\theta] \tag{2.6}$$

The molecular viscosity is obtained using Sutherland's viscosity law [76]:

$$\mu = \frac{C_1 T^{\frac{3}{2}}}{T + C_2}$$

where the constants p_r, C_1 and C_2 for air at moderate temperature are:

$$p_r = 0.72, \qquad C_1 = 1.458\text{x}10^{-6}\text{kg}/(\text{m} - \text{s} - \text{K}^{1/2}), \qquad C_2 = 110.4\text{K}$$

The governing equations may be nondimensionalized with respect to chosen reference quantities. In this work a reference length, L, and reference flow properties, ρ_∞, V_∞, T_∞, and μ_∞, are used to define the nondimensional parameters. The normalized variables will be expressed here as bar variables, which are

$$\bar{t} = \frac{t}{L/V_\infty}, \quad \bar{m} = \frac{m}{L}, \quad \bar{r} = \frac{r}{L}, \quad \bar{h} = \frac{h}{L}$$

$$\bar{\rho} = \frac{\rho}{\rho_\infty}, \quad \bar{V}_m = \frac{V_m}{V_\infty}, \quad \bar{V}_\theta = \frac{V_\theta}{V_\infty}, \quad \bar{r\Omega} = \frac{r\Omega}{V_\infty}$$

$$\bar{p} = \frac{p}{\rho_\infty V_\infty^2}, \quad \bar{T} = \frac{T}{T_\infty}, \quad \bar{e} = \frac{e}{\rho_\infty V_\infty^2}, \quad \bar{\mu} = \frac{\mu}{\mu_\infty}$$

The nondimensional equations resulting from substituting the above nondimensional variables into Eq.(2.3) are similar to their original dimensional form except a constant, Re_L, appears in the viscous terms. Dropping the bar notation, we may rewrite the governing equations as:

$$\frac{\partial(rh\vec{U})}{\partial t} + \frac{\partial(rh\vec{F})}{\partial m} + \frac{\partial(h\vec{G})}{\partial \theta} - Re_L^{-1}(\frac{\partial(rh\vec{R})}{\partial m} + \frac{\partial(h\vec{S})}{\partial \theta}) = rh\vec{K} \tag{2.7}$$

where the reference Reynolds number, Re_L, is defined as:

$$Re_L = \frac{\rho_\infty V_\infty L}{\mu_\infty}$$

and the coefficient of thermal conductivity, κ_T, is given as:

$$\kappa_T = \frac{\mu}{Pr(\gamma - 1)M_\infty^2}$$

Also note that the source term, K_2, now becomes

$$K_2 = (\rho V_\theta^2 + p - Re_L^{-1}\sigma_{22})(\frac{1}{r}\frac{dr}{dm}) + (p - Re_L^{-1}\sigma_{33})(\frac{1}{h}\frac{dh}{dm})$$

For quasi-three-dimensional inviscid flow problems the governing equations are obtained by dropping the viscous terms in the Navier-Stokes equations. These equations reduce to the standard two-dimensional Navier-Stokes equations in conservation form by setting r and h equal to constants. In the present work, the two-dimensional flow problems are solved using these equations with $r = h = 1$.

2.3 Favre-Averaged Navier-Stokes Equations

The full Navier-Stokes equations provide the "exact" transport equations for compressible turbulent flows. With specification of appropriate initial and boundary conditions the equations may be solved directly. Unfortunately, turbulent flows always contain fluctuations at a wide range of frequencies and in three-dimensional applications it requires $\mathcal{O}(Re_{\tau}^{9/4})$ grid points [45] to resolve all the flow scales. Even with modern supercomputers direct numerical simulation of the full Navier-Stokes equations is still restricted to low Reynolds flows and to very simple boundary conditions. In most engineering applications, the mean flow properties are of primary interest. As a result, a modified form of the Navier-Stokes equations derived through averaging techniques is adopted for engineering calculations.

In the present work, the Favre-averaged technique is used to obtain the mean flow equations. The Favre averaging is a hybrid averaged method which uses density weighted averaging on all fluid properties except pressure and density. For compressible flows this averaging results in a simpler form of the mean flow equations than the Reynolds averaging. Performing the Favre averaging the quasi-three-dimensional Navier-Stokes equations (see appendix B) are

$$\frac{\partial(rh\langle\vec{U}\rangle)}{\partial t} + \frac{\partial(rh\langle\vec{F}\rangle)}{\partial m} + \frac{\partial(h\langle\vec{G}\rangle)}{\partial\theta} - Re_L^{-1}(\frac{\partial(rh\langle\vec{R}\rangle)}{\partial m} + \frac{\partial(h\langle\vec{S}\rangle)}{\partial\theta}) = rh\langle\vec{K}\rangle \quad (2.8)$$

where

$$\langle\vec{U}\rangle = \begin{pmatrix} \langle\rho\rangle \\ \langle\rho\rangle\widetilde{V_m} \\ \langle\rho\rangle\widetilde{W_\theta} \\ \langle\rho\rangle\widetilde{E} \end{pmatrix}, \quad \langle\vec{F}\rangle = \begin{pmatrix} \langle\rho\rangle\widetilde{V_m} \\ \langle\rho\rangle\widetilde{V_m^2} + \langle p\rangle \\ \langle\rho\rangle\widetilde{V_m}\widetilde{V_\theta}r \\ \widetilde{V_m}(\langle\rho\rangle\widetilde{E} + \langle p\rangle) \end{pmatrix}$$

$$\langle \vec{G} \rangle = \begin{pmatrix} \langle \rho \rangle \widetilde{W_\theta} \\ \langle \rho \rangle \widetilde{V_m} \widetilde{W_\theta} \\ (\langle \rho \rangle \widetilde{V_\theta} \widetilde{W_\theta} + \langle p \rangle) r \\ \widetilde{W_\theta}(\langle \rho \rangle \widetilde{E} + \langle p \rangle) + r\Omega \langle p \rangle \end{pmatrix}$$

$$\langle \vec{R} \rangle = \begin{pmatrix} 0 \\ \widetilde{\sigma_{11}} - \langle \rho V_m''^2 \rangle \\ (\widetilde{\sigma_{12}} - \langle \rho V_m'' V_\theta'' \rangle) r \\ \widetilde{V_m}(\widetilde{\sigma_{11}} - \langle \rho V_m''^2 \rangle) + \widetilde{V_\theta}(\widetilde{\sigma_{12}} - \langle \rho V_m'' V_\theta'' \rangle) + \kappa_\tau \frac{\partial \widetilde{T}}{\partial m} - \langle \rho V_m'' H'' \rangle \end{pmatrix}$$

$$\langle \vec{S} \rangle = \begin{pmatrix} 0 \\ \widetilde{\sigma_{12}} - \langle \rho V_m'' V_\theta'' \rangle \\ (\widetilde{\sigma_{22}} - \langle \rho V_\theta''^2 \rangle) r \\ \widetilde{V_m}(\widetilde{\sigma_{12}} - \langle \rho V_m'' V_\theta'' \rangle) + \widetilde{V_\theta}(\widetilde{\sigma_{22}} - \langle \rho V_\theta''^2 \rangle) + \frac{\kappa_\tau}{r} \frac{\partial \widetilde{T}}{\partial \theta} - \langle \rho V_\theta'' H'' \rangle \end{pmatrix}$$

$$\langle \vec{K} \rangle = \begin{pmatrix} 0 \\ \left[\langle \rho \rangle \widetilde{V_\theta^2} + \langle p \rangle - Re_L^{-1}(\widetilde{\sigma_{22}} - \langle \rho V_\theta''^2 \rangle)\right] (\frac{1}{r}\frac{dr}{dm}) + \left(\langle p \rangle - Re_L^{-1}\,(\widetilde{\sigma_{33}} - \langle \rho V_n''^2 \rangle)\right) (\frac{1}{h}\frac{dh}{dm}) \\ 0 \\ 0 \end{pmatrix}$$

The equation of state for perfect gas is

$$\langle p \rangle = (\gamma - 1)\langle \rho \rangle \left[\widetilde{E} - \frac{1}{2}\left(\widetilde{V_m^2} + \widetilde{V_\theta^2} + \widetilde{V_m''^2} + \widetilde{V_\theta''^2} \right) \right]$$

The Favre-averaged equations are similar in form to the full Navier-Stokes equations except for the presence of unknown stress terms in the mean momentum and energy equations. Among these new terms, $\langle \rho V_m''^2 \rangle$, $\langle \rho V_m'' V_\theta'' \rangle$, and $\langle \rho V_\theta''^2 \rangle$ are the so-called Reynolds stresses and $\langle \rho V_m'' H'' \rangle$ and $\langle \rho V_\theta'' H'' \rangle$ are the turbulence heat flux. These terms must be modeled to provide closure.

2.4 Turbulence Models

The Favre-averaged equations contain correlations of turbulence quantities. In order to solve for the mean flow properties these terms must be modeled. Various approaches for modeling the correlated terms have been reviewed in the literature [44, 85]. Although the Reynolds stress model provides general one-point correlation approach for turbulence quantities, it increases the level of complexity of equations and still additional unknowns need to be modeled. In the present work an eddy viscosity model is adopted. Using an eddy viscosity model the averaged equations are identical in form to the full Navier-Stokes equations, with the viscosity and thermal conductivity replaced with the corresponding effective values, $\tilde{\mu}_e$ and $(\tilde{\mu}/p_r)_e$, respectively.

2.4.1 Eddy Viscosity Hypothesis

In the eddy viscosity hypothesis turbulence quantities are related to mean flow properties through the use of the Boussinesq assumption, which assumes that the Reynolds stress tensor is linearly proportional to the mean strain rate tensor. In Cartesian coordinates this gives

$$\langle -\rho V_i''V_j''\rangle = \widetilde{\mu_t}\left[\left(\frac{\partial \tilde{V}_i}{\partial x_j} + \frac{\partial \tilde{V}_j}{\partial x_i}\right) - \frac{2}{3}\delta_{ij}\frac{\partial \tilde{V}_k}{\partial x_k}\right] - \frac{2}{3}\delta_{ij}\langle\rho\rangle\tilde{\mathrm{k}} \tag{2.9}$$

In the above expression, $\widetilde{\mu_t}$ is the turbulent viscosity, δ_{ij} is the Kronecker delta function, and $\tilde{\mathrm{k}}$ is the mean turbulent kinetic energy. Analogous to the Boussinesq assumption, a linear relationship between the turbulent heat flux and the enthalpy gradient is also adopted:

$$\langle \rho V_i''H''\rangle = -\frac{\tilde{\mu}_t}{p_{r_t}}\frac{\partial \widetilde{H}}{\partial x_i} = -\frac{C_p\tilde{\mu}_t}{p_{r_t}}\frac{\partial \tilde{T}}{\partial x_i} \tag{2.10}$$

where p_{r_t} is the turbulent Prandtl number, which is assumed to be 0.91 in the present work. Thus, the mean flow equations for turbulent flow can be expressed in a form similar to the original equations of motion by replacing the viscosity and thermal conductivity with the corresponding effective value defined as the sum of laminar and turbulent parts. The effective viscosity and conductivity may be written as:

$$\tilde{\mu}_e = \tilde{\mu}_{\mathbf{laminar}} + \tilde{\mu}_{\mathbf{turbulent}}$$

$$(\tilde{\mu}/p_r)_e = (\tilde{\mu}/p_r)_{\mathbf{laminar}} + (\tilde{\mu}/p_r)_{\mathbf{turbulent}} \tag{2.11}$$

The remaining turbulence closure problem is then to model the turbulent viscosity, μ_t, and/or the mean turbulent kinetic energy, $\tilde{\mathrm{k}}$.

Eddy viscosity models employing the mixing length theory [81], which is analogous to the molecular kinetic theory, give the turbulent viscosity, $\tilde{\mu}_t$, as

$$\tilde{\mu}_t = C\langle\rho\rangle\tilde{q}l \tag{2.12}$$

where $\tilde{q}$ and l represent the velocity and length scales of the turbulence. Depending on the type and the number of equations employed to evaluate these turbulence scales, the modeling equations are classified as algebraic, one-, and two-equation models. Although algebraic models are difficult to implement using unstructured meshes, the Cebeci-Smith [16] algebraic model is described here for completeness. In the present work, the Baldwin-Barth [6] one-equation and Chien [17] low Reynolds number two-equation models are employed.

2.4.2 Algebraic Turbulence Models

In algebraic turbulence models the turbulent velocity and length scales are modeled algebraically. The Cebeci-Smith [16] model is one common example. In the Cebeci-Smith [16] model, the velocity and length scales of turbulence are determined using a two-layer model. In the inner layer where $y \leq y_c$, the required scales and turbulent viscosity are obtained as:

$$\tilde{\mu}_t^i = \langle\rho\rangle l^2 |\tilde{\Omega}| \tag{2.13}$$

$$l = \mathcal{K} y \mathcal{D}$$

$$|\tilde{\Omega}| = \sqrt{(\frac{\partial\tilde{u}}{\partial y} - \frac{\partial\tilde{v}}{\partial x})^2 + (\frac{\partial\tilde{v}}{\partial z} - \frac{\partial\tilde{w}}{\partial y})^2 + (\frac{\partial\tilde{w}}{\partial x} - \frac{\partial\tilde{u}}{\partial z})^2}$$

where $\mathcal{K} = 0.41$ is the von Karman constant and $\mathcal{D}$ is the van Driest damping function which is used to account for near-wall viscous effects on turbulence.

$$\mathcal{D} = 1 - \exp(-y^+/A^+), \qquad A^+ = 26$$

The product of $l|\tilde{\Omega}|$ gives the velocity scale of turbulence. The wall distance unit, y^+, is defined as:

$$y^+ = \frac{\Delta y_{\mathrm{wall}}\sqrt{\tau_{\mathrm{wall}}/\rho_{\mathrm{wall}}}}{\nu_{\mathrm{wall}}} \tag{2.14}$$

where Δy_{wall} is the minimum physical distance from an interior node to the wall, τ_{wall}, ρ_{wall} and ν_{wall} are the wall shear stress, density and kinetic viscosity at the corresponding wall location, respectively.

In the outer layer, $y > y_c$, the turbulence quantities are given as:

$$\tilde{\mu}_t^o = \mathcal{C}_\alpha \langle\rho\rangle \tilde{U}_e \delta_1$$

$$\tilde{U}_e \delta_1 = \int_0^\infty (1 - \tilde{u}/\tilde{U}_e) dy \tag{2.15}$$

where $\mathcal{C}_\alpha = 0.0168$, δ_1 is the displacement thickness, and U_e is the edge velocity.

The condition which defines y_c is the continuity of the turbulent viscosity. In practice, the turbulent viscosity is determined as the minimum of the inner and outer turbulent viscosities. That is

$$\tilde{\mu}_t = \min(\tilde{\mu}_t^i, \tilde{\mu}_t^o) \tag{2.16}$$

Algebraic models do not require any transport equation for turbulence properties and therefore are the simpliest and easiest models to use. However, it has been noted [69] that algebraic models do not account for any transport and history effects of turbulence and hence are not adequate for complex separated flow problems. Incorporating zero-equation models into a unstructured approach requires a locally structured mesh to provide the length scale information (e.g., δ_1 in the Cebeci-Smith model).

2.4.3 One-Equation Models

Early one equation models, including those of Bradshaw *et al.* [13], Rubesin [73], and Mitcheltree *et al.* [62], were based on the transport equation of the turbulent kinetic energy, $\tilde{k}$, from which the velocity scale is evaluated. The turbulent length scale is then determined using an empirical description. Although the transport equation of the turbulent kinetic energy provides transport and history effects of turbulence, a length scale specified algebraically is not adequate for general flow problems. In the past these models have been more difficult to code and often only marginally better than algebraic models [69]. Further, the need of algebraic length scales limits the applicability of these models to unstructured meshes.

Recently, Baldwin and Barth [6] proposed a one-equation model which involves a transport equation for a turbulence field variable. Except in the near wall region this variable is proportional to the turbulent viscosity so algebraic length scales are not required for the evaluation of the turbulent viscosity. This model has been shown to be a significant improvement over algebraic models. Although this model accounts for near-wall viscous effects on turbulence it does not require a very fine mesh spacing for resolving the viscous layer. Baldwin and Barth noted that their model only requires a wall mesh spacing of $y^+ < 3.5$, which relieves the numerical stiffness problem. In addition, since this model does not require an algebraic specification of the turbulent length scales it is applicable to unstructured meshes. A complete derivation and discussion of this model can be found in Reference [6]. The resulting model is summarized below.

The turbulent viscosity is modeled as:

$$\tilde{\nu}_t = C_\mu \mathcal{D}_1 \mathcal{D}_2 \tilde{\nu} \tilde{R}_T \tag{2.17}$$

In Reference [6] the transport equation for the turbulence variable, $\tilde{\nu}\tilde{R}_T$, is given as:

$$\frac{D(\tilde{\nu}\tilde{R}_T)}{Dt} = (C_{\epsilon_2}\mathcal{F}_2 - C_{\epsilon_1})\sqrt{\tilde{\nu}\tilde{R}_T\mathcal{P}} + \left(\tilde{\nu} + \frac{\tilde{\nu}_t}{\sigma_\epsilon}\right)\nabla^2(\tilde{\nu}\tilde{R}_T) - \frac{1}{\sigma_\epsilon}\nabla\tilde{\nu}_t \cdot \nabla(\tilde{\nu}\tilde{R}_T) \tag{2.18}$$

where $\mathcal{P}$ is the production of the turbulent kinetic energy.

$$\mathcal{P} = \tilde{\nu}_t \left(\frac{\partial \tilde{V}_i}{\partial x_j} + \frac{\partial \tilde{V}_j}{\partial x_i} \right) \frac{\partial \tilde{V}_i}{\partial x_j} - \frac{2}{3} \tilde{\nu}_t \left(\frac{\partial \tilde{V}_k}{\partial x_k} \right)^2$$

The corresponding nondimensional quasi-three-dimensional form is obtained as

$$\frac{\partial (rh\tilde{\nu}\tilde{R}_T)}{\partial t} + \tilde{V}_m \frac{\partial (rh\tilde{\nu}\tilde{R}_T)}{\partial m} + \widetilde{W}_\theta \frac{\partial (h\tilde{\nu}\tilde{R}_T)}{\partial \theta} = Re_L^{-1} \Bigg\{ \left(\tilde{\nu} + \frac{\tilde{\nu}_t}{\sigma_\epsilon} \right) \\ \left[\frac{\partial (rh\langle R_B \rangle)}{\partial m} + \frac{\partial (h\langle S_B \rangle)}{\partial \theta} \right] - \frac{1}{\sigma_\epsilon} \left[\frac{\partial \tilde{\nu}_t}{\partial m} (rh\langle R_B \rangle) + \frac{\partial \tilde{\nu}_t}{\partial \theta} (h\langle S_B \rangle) \right] + rh\langle K_B \rangle \Bigg\} \tag{2.19}$$

where

$$\langle R_B \rangle = \frac{\partial (\tilde{\nu}\tilde{R}_T)}{\partial m}, \qquad \langle S_B \rangle = \frac{1}{r} \frac{\partial (\tilde{\nu}\tilde{R}_T)}{\partial \theta}$$

$$\langle K_B \rangle = (\mathcal{C}_{\epsilon_2} \mathcal{F}_2 - \mathcal{C}_{\epsilon_1}) \sqrt{\tilde{\nu}\tilde{R}_T \mathcal{P}}$$

The following functions are employed to account for near-wall viscous effects on turbulence.

$$\begin{aligned} \frac{1}{\sigma_\epsilon} &= (\mathcal{C}_{\epsilon_2} - \mathcal{C}_{\epsilon_1}) \sqrt{\mathcal{C}_\mu} / \mathcal{K}^2 \\ \mathcal{D}_1 &= 1 - \exp(-y^+/A^+) \\ \mathcal{D}_2 &= 1 - \exp(-y^+/A_2^+) \\ \mathcal{F}_2(y^+) &= \frac{C_{\epsilon_1}}{C_{\epsilon_2}} + (1 - \frac{C_{\epsilon_1}}{C_{\epsilon_2}})(\frac{1}{\mathrm{k}y^+} + \mathcal{D}_1\mathcal{D}_2) \Bigg[\sqrt{\mathcal{D}_1\mathcal{D}_2} \\ &\quad + \frac{y^+}{\sqrt{\mathcal{D}_1\mathcal{D}_2}} (\mathcal{D}_1'\mathcal{D}_2 + \mathcal{D}_1\mathcal{D}') \Bigg] \end{aligned}$$

$$\mathcal{D}_1' = \frac{1}{A^+} \exp(-y^+/A^+), \qquad \mathcal{D}_2' = \frac{1}{A_2^+} \exp(-y^+/A_2^+), \qquad y^+ = \sqrt{\frac{\tilde{\tau}_{\mathrm{wall}}}{\langle \rho \rangle_{\mathrm{wall}}}} \frac{\Delta y_{\mathrm{wall}}}{\tilde{\nu}_{\mathrm{wall}}}$$

where the modeling constants are given as:

$$\mathcal{K} = 0.41, \quad \mathcal{C}_{\epsilon_1} = 1.2, \quad \mathcal{C}_{\epsilon_2} = 2.0$$

$$\mathcal{C}_\mu = 0.09, \quad A^+ = 26, \quad A_2^+ = 10$$

2.4.4 Two-Equation Models

In most one-equation models the length scale of turbulence is described algebraically. It has been recognized that the use of an algebraic length scale is not adequate for general flow problems [44, 85]. In order to eliminate an algebraic description for the turbulence length scale, an additional transport equation for the length scale of turbulence has been proposed. A variety of two-equation models can be found in review articles [44, 85]. The most popular model among them is the $k-\epsilon$ model. Both high Reynolds number and low Reynolds number forms of this model have been used successfully in engineering applications. The high Reynolds number $k-\epsilon$ model is not valid in near-wall regions, so instead of integrating the mean flow and $k-\epsilon$ equations up to the wall, wall functions are employed to estimate near-wall flow properties. A detailed numerical implementation of this approach can be found in References [46, 59]. This approach removes the need of extremely fine meshes near the wall and has been applied to a large class of flows. It is only appropriate to use wall functions on flows where the logarithmic law of the wall region exists.

In complex flow problems the near-wall distribution of mean flow and turbulence properties might be different from the logarithmic law. To accurately predict the near-wall viscous effects on turbulence, it is necessary to integrate the mean flow and turbulence equations to the wall. Chien [17] modified the standard $k-\epsilon$ model by accounting for the near-wall viscous effects and proposed a low Reynolds number $k-\epsilon$ turbulence model which is valid up to the wall. Although this model requires a mesh wall spacing ($y^+ < 1$), it is coarser than the spacing ($y^+ < 0.2$) that the Jones-Launder [39] low Reynolds number $k-\epsilon$ model requires. Chien's model has been widely used in engineering applications. This model is outlined below.

The nondimensional turbulent viscosity is expressed by

$$\tilde{\mu}_t = Re_L\, C_\mu \mathcal{F}_\mu \langle \rho \rangle \frac{\tilde{k}^2}{\tilde{\epsilon}} \tag{2.20}$$

where $\widetilde{\mathrm{k}}$ is the turbulent kinetic energy and $\widetilde{\epsilon}$ is the isotropic dissipation rate of the turbulent kinetic energy.

$$\widetilde{\mathrm{k}} = \frac{\frac{1}{2}\langle \rho V_i'' V_i'' \rangle}{\langle \rho \rangle}, \qquad \widetilde{\epsilon} = \widetilde{\nu}\frac{\langle \rho V_{i,j}'' V_{i,j}'' \rangle}{\langle \rho \rangle}$$

The averaged transport equations for $\widetilde{\mathrm{k}}$ and $\widetilde{\epsilon}$ quantities in quasi-3D compressible flows are described by

$$\frac{\partial(rh\langle \vec{U}_{\mathrm{k}\epsilon}\rangle)}{\partial t} + \frac{\partial(rh\langle \vec{F}_{\mathrm{k}\epsilon}\rangle)}{\partial m} + \frac{\partial(h\langle \vec{G}_{\mathrm{k}\epsilon}\rangle)}{\partial \theta} = Re_L^{-1}\left(\frac{\partial(rh\langle \vec{R}_{\mathrm{k}\epsilon}\rangle)}{\partial m} + \frac{\partial(h\langle \vec{S}_{\mathrm{k}\epsilon}\rangle)}{\partial \theta} + rh\langle \vec{K}_{\mathrm{k}\epsilon}\rangle\right) \tag{2.21}$$

where

$$\langle \vec{U}_{\mathrm{k}\epsilon}\rangle = \begin{pmatrix} \langle\rho\rangle\widetilde{\mathrm{k}} \\ \langle\rho\rangle\widetilde{\epsilon} \end{pmatrix}, \quad \langle \vec{F}_{\mathrm{k}\epsilon}\rangle = \begin{pmatrix} \langle\rho\rangle\widetilde{V}_m\widetilde{\mathrm{k}} \\ \langle\rho\rangle\widetilde{V}_m\widetilde{\epsilon} \end{pmatrix}, \quad \langle \vec{G}_{\mathrm{k}\epsilon}\rangle = \begin{pmatrix} \langle\rho\rangle\widetilde{W}_\theta\widetilde{\mathrm{k}} \\ \langle\rho\rangle\widetilde{W}_\theta\widetilde{\epsilon} \end{pmatrix}$$

$$\langle \vec{R}_{\mathrm{k}\epsilon}\rangle = \begin{pmatrix} (\widetilde{\mu} + \frac{\widetilde{\mu}_t}{\sigma_{\mathrm{k}}})\frac{\partial\widetilde{\mathrm{k}}}{\partial m} \\ (\widetilde{\mu} + \frac{\widetilde{\mu}_t}{\sigma_\epsilon})\frac{\partial\widetilde{\epsilon}}{\partial m} \end{pmatrix}, \quad \langle \vec{S}_{\mathrm{k}\epsilon}\rangle = \begin{pmatrix} (\widetilde{\mu} + \frac{\widetilde{\mu}_t}{\sigma_{\mathrm{k}}})\frac{1}{r}\frac{\partial\widetilde{\mathrm{k}}}{\partial \theta} \\ (\widetilde{\mu} + \frac{\widetilde{\mu}_t}{\sigma_\epsilon})\frac{1}{r}\frac{\partial\widetilde{\epsilon}}{\partial \theta} \end{pmatrix}$$

$$\langle \vec{K}_{\mathrm{k}\epsilon}\rangle = \begin{pmatrix} \mathcal{P} - Re_L\langle\rho\rangle\widetilde{\epsilon} + \mathcal{E}_{\mathrm{k}} \\ \frac{\widetilde{\epsilon}}{\widetilde{\mathrm{k}}}(\mathcal{F}_1\mathcal{C}_1\mathcal{P} - Re_L\mathcal{F}_2\mathcal{C}_2\langle\rho\rangle\widetilde{\epsilon}) + \mathcal{E}_\epsilon \end{pmatrix}$$

In Reference [17], Chien proposed the following constants and functions for this model:

$$\sigma_{\mathrm{k}} = 1.0, \qquad \sigma_\epsilon = 1.3$$

$$\mathcal{C}_\mu = 0.09, \qquad \mathcal{C}_1 = 1.35, \qquad \mathcal{C}_2 = 1.8$$

$$\mathcal{F}_\mu = 1 - \exp(-0.0115y^+), \qquad \mathcal{F}_1 = 1, \qquad \mathcal{F}_2 = 1 - \frac{2}{9}\exp(-R_t^2/36)$$

$$\mathcal{P} = \widetilde{\mu_t}\left(\frac{\partial\widetilde{V}_i}{\partial x_j} + \frac{\partial\widetilde{V}_j}{\partial x_i}\right)\frac{\partial\widetilde{V}_i}{\partial x_j} - \frac{2}{3}\widetilde{\mu_t}\left(\frac{\partial\widetilde{V}_k}{\partial x_k}\right)^2 - \frac{2}{3}\langle\rho\rangle\widetilde{\mathrm{k}}\frac{\partial\widetilde{V}_k}{\partial x_k}$$

$$\mathcal{E}_{\mathrm{k}} = -\frac{2\widetilde{\mu}\widetilde{\mathrm{k}}}{\Delta y_{\mathrm{wall}}^2}, \qquad \mathcal{E}_\epsilon = -\frac{2\widetilde{\mu}\widetilde{\epsilon}}{\Delta y_{\mathrm{wall}}^2}\exp(-0.5y^+)$$

$$R_t = Re_L\frac{\langle\rho\rangle\widetilde{\mathrm{k}}^2}{\widetilde{\mu}\widetilde{\epsilon}}, \qquad y^+ = \sqrt{\frac{\widetilde{\tau}_{\mathrm{wall}}}{\langle\rho\rangle_{\mathrm{wall}}}}\frac{\Delta y_{\mathrm{wall}}}{\widetilde{\nu}_{\mathrm{wall}}}$$

The extra terms, $\mathcal{E}_{\mathrm{k}}$ and $\mathcal{E}_\epsilon$, are used to account for near-wall viscous effects on turbulence. Because near-wall effects needs to be resolved, this model requires a wall spacing $y^+ < 1$.

2.5 Initial and Boundary Conditions

The mean flow and turbulent transport equations presented in the preceding sections represent an initial-boundary value problem. In order to solve these equations it is necessary to impose initial and boundary conditions. Numerical implementation of these conditions will be given in the next chapter. Since only steady state solutions are of interest in the present work, initial conditions are not relevant to the solutions.

2.5.1 Physical Boundary Conditions

In the present work solutions for 2-D airfoils, 2-D and quasi-3D cascades are presented. For these cases the boundaries are either inflow/outflow or solid wall type. On solid walls the physical boundary conditions are the same for all problems. For inviscid flows the solid wall boundary condition is flow tangency condition.

$$(\tilde{\bar{\vec{V}}} - \tilde{\bar{\vec{V}}}_{\mathbf{wall}}) \cdot \vec{n}_{\mathbf{wall}} = 0 \tag{2.22}$$

For viscous flows the physical boundary condition is the no-slip condition with either specified wall temperature or heat flux. That is,

$$\begin{gathered} \tilde{\bar{\vec{V}}} = \tilde{\bar{\vec{V}}}_{\mathbf{wall}} \\ \tilde{T} = \tilde{T}_{\mathbf{wall}} \text{ or } \frac{\partial \tilde{T}}{\partial n} = [\frac{\partial \tilde{T}}{\partial n}]_{\mathbf{wall}} \end{gathered} \tag{2.23}$$

At inflow/outflow boundaries the physical boundary conditions are dependent on the type of problem solved. For the 2-D airfoil problem, the inflow/outflow boundary is the far field boundary at infinity. The corresponding physical boundary condition is the asymptotic state of a uniform freestream condition for either inviscid or viscous flow. For the 2-D or quasi-3D cascade case, the physical boundary conditions at the inlet are specified total pressure, total temperature and whirl speed rV_θ. The physical boundary condition for the outflow boundary is specified static pressure.

2.5.2 Turbulence Transport Equations

When a turbulence model based on differential equations of turbulence quantities is used, it requires additional boundary conditions for each equation. The physical boundary conditions are determined by the type of turbulence model used. For the Baldwin-Barth one-equation model, the physical boundary conditions are [6]:

- Solid Walls: Specify $\widetilde{R_T} = 0$
- Inflow ($\vec{V} \cdot \vec{n} < 0$): Specify $\widetilde{R_T} = (\widetilde{R_T})_\infty < 1$
- Outflow ($\vec{V} \cdot \vec{n} > 0$): Extrapolate $\widetilde{R_T}$ from interior values

For Chien's low Reynolds number $\mathrm{k} - \epsilon$ model, the physical boundary conditions required for $\tilde{\mathrm{k}}$ and $\tilde{\epsilon}$ are [43]:

- Solid Walls: Specify $\tilde{\mathrm{k}} = 0$, and $\tilde{\epsilon} = 0$
- Inflow ($\vec{V} \cdot \vec{n} < 0$): Specify $\tilde{\mathrm{k}} = \mathrm{k}_\infty$, and $\tilde{\epsilon} = \epsilon_\infty$
- Outflow ($\vec{V} \cdot \vec{n} > 0$): Extrapolate $\tilde{\mathrm{k}}$ and $\tilde{\epsilon}$ from interior values

It is important to realize that the total dissipation rate is not negligible at walls, even thought the isotropic dissipation goes to zero. The extra terms in the Chien $\mathrm{k} - \epsilon$ equations can be used to account for a nonzero dissipation rate near the wall.

3. UNSTRUCTURED FLOW SOLVER

This chapter presents the spatial and temporal discretizations of the governing equations. The Jameson's four-stage Runge-Kutta cell-vertex finite-volume time-marching scheme is used to solve the governing equations. The stability requirements for this scheme are reviewed along with some convergence acceleration techniques. Finally the boundary conditions are formulated for two-dimensional airfoil and quasi-three-dimensional cascade cases. Initial conditions used in the quasi-three-dimensional cascade cases are also discussed.

3.1 Finite-Volume Spatial Discretization

For turbulent flow calculations the turbulence transport equations and mean flow equations may be solved in either a coupled or a decoupled manner. Although the coupling approach gives a single system of equations which simultaneously governs the mean flow and the turbulence properties, it is noted in Reference [42] that there is no advantage to numerically coupling these equations in terms of either convergence rate or accuracy. On the other hand, the decoupled approach allows the two set of equations to be treated separately using different numerical schemes. In addition, different turbulence models can be easily incorporated into the mean flow solver without major program modification. Since the objective here is to develop a flexible and robust solution scheme for complex flow problems, the decoupled approach is employed. With the decoupled approach the mean flow equations are marched in time using turbulence quantities frozen from last time step. The turbulence transport equations are then integrated in time using frozen mean flow properties.

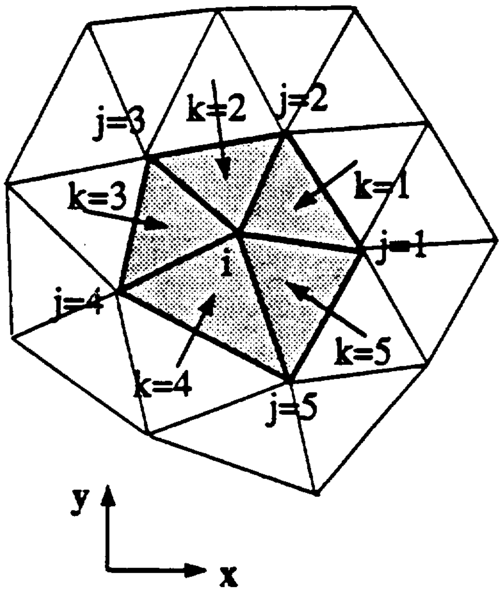


Figure 3.1 Control volume for the cell-vertex scheme.

Both the unsteady quasi-three-dimensional mean flow equations and turbulence transport equations are solved on an unstructured triangular mesh using Jameson's four-stage Runge-Kutta finite-volume time-marching scheme [58]. In the present work, the cell-vertex finite-volume spatial discretization for triangular meshes [36, 55] is used. The truncation error of this formulation has been shown to be second order for smooth grids and first order for general irregular meshes [70].

3.1.1 Mean Flow Equations

The quasi-three-dimensional mean flow equations can be expressed in an integral form for control volume $\mathcal{V}$ with a boundary surface ∂V as follows:

$$\frac{\partial}{\partial t}\iiint_{\mathcal{V}}\langle\vec{U}\rangle d\mathcal{V}+\oint_{\partial\mathcal{V}}(\langle\vec{F}\rangle,\langle\vec{G}\rangle)\cdot\vec{n}dS-Re_L^{-1}\oint_{\partial\mathcal{V}}(\langle\vec{R}\rangle,\langle\vec{S}\rangle)\cdot\vec{n}dS=\iiint_{\mathcal{V}}\langle\vec{K}\rangle d\mathcal{V} \quad (3.1)$$

In the cell-vertex finite-volume formulation flow properties are stored at the mesh points (triangle vertices) with the control volume for each point i being the union of all triangles with a common vertex at point i as shown in Figure 3.1. The volume integral is then approximated using the mean value theorem, and the boundary integral is approximated using trapezoidal integration over the bounding surface. This process results in the follow system of semi-discrete equations for each point i

$$\frac{d\langle\vec{U}_i\rangle}{dt} + R(\langle\vec{U}_i\rangle) = 0 \tag{3.2}$$

where $R(\langle\vec{U}_i\rangle)$ is the residual

$$R(\langle\vec{U}_i\rangle) = \frac{Q(\langle\vec{U}_i\rangle)}{\mathcal{V}_i} - \langle\vec{K}\rangle \tag{3.3}$$

and

$$\mathcal{V}_i = \sum_{i=1}^{N_i}(rhm)_k\Delta\theta_k = -\sum_{i=1}^{N_i}(rh\theta)_k\Delta m_k$$

$$\begin{aligned} Q(\langle\vec{U}_i\rangle) &= \sum_{k=1}^{N_i}\left[(rh\langle\vec{F}\rangle)_k\Delta\theta_k - (h\langle\vec{G}\rangle)_k\Delta m_k\right] \\ &- Re_L^{-1}\sum_{k=1}^{N_i}\left[(rh\langle\vec{R}\rangle)_k\Delta\theta_k - (h\langle\vec{S}\rangle)_k\Delta m_k\right] \end{aligned} \tag{3.4}$$

In the above expressions the subscript k refers to the k^{th} segment of line bounding the control volume, N_i denotes the total number of vertices of the control volume, $\mathcal{V}_i$ is the area of the control volume, $\Delta\theta_k$ and Δm_k are the increments along segment k, and the components of flux vectors, $\langle\vec{F}_k\rangle$, $\langle\vec{G}_k\rangle$, $\langle\vec{R}_k\rangle$ and $\langle\vec{S}_k\rangle$, are taken as the average of the nodal values for each end of the segment. An entire flux balance throughout the flow field can be computed in one simple pass over all edges using an edge-based data structure. The velocity gradient in $\langle R_k\rangle$ and $\langle S_k\rangle$ are computed using Gauss's theorem [61]:

$$\int_{\mathcal{V}}\nabla\phi d\mathcal{V} = \oint_{\partial\mathcal{V}}\phi\vec{n}dS \tag{3.5}$$

where $\vec{n}$ is the unit outward normal vector to dS. Using the same control volume $\mathcal{V}$, the above equation can be expressed as:

$$\frac{\partial\phi}{\partial m} = \frac{1}{\mathcal{V}_i}\sum_{k=1}^{N_i}(rh\phi)_k\Delta\theta_k, \qquad \frac{1}{r}\frac{\partial\phi}{\partial\theta} = -\frac{1}{\mathcal{V}_i}\sum_{k=1}^{N_i}(h\phi)_k\Delta m_k \tag{3.6}$$

Replacing ϕ with $\widetilde{V_m}$ and $\widetilde{V_\theta}$ gives the velocity gradient at each node. This completes the discretization for the mean flow equations.

3.1.2 Turbulence Transport Equations

Differential transport equations are integrated in time using the same explicit finite volume Runge-Kutta time integration scheme as the mean flow equations. This is accomplished by first taking a volume integral of the transport equations. Where possible the divergence theorem is then used to transform volume integrals into corresponding boundary integrals. The resulting integral equations are then discretized in space using the same cell-vertex finite-volume approximation as used for the mean flow equations.

3.1.2.1 Baldwin-Barth One-Equation Model

The finite-volume discretization of the Baldwin-Barth one-equation model, Eq. (2.19), involves two gradient operators, $\nabla(\tilde{\nu}\widetilde{R_T})$ and $\nabla\tilde{\nu}_t$, which results in a complicated discretization. In order to simplify the discretization for the one-equation model, the last two terms on the right-hand side of the equation are reformulated using vector identities. This yields

$$\left(\tilde{\nu}+\frac{\tilde{\nu}_t}{\sigma_\epsilon}\right)\nabla^2(\tilde{\nu}\widetilde{R_T})-\frac{1}{\sigma_\epsilon}\nabla\tilde{\nu}_t\cdot\nabla(\tilde{\nu}\widetilde{R_T})=\left(\tilde{\nu}+2\frac{\tilde{\nu}_t}{\sigma_\epsilon}\right)\nabla^2(\tilde{\nu}\widetilde{R_T})-\frac{1}{\sigma_\epsilon}\nabla\cdot\tilde{\nu}_t\nabla(\tilde{\nu}\widetilde{R_T}) \tag{3.7}$$

Substituting Eq. (3.7) into Eq. (2.19) and integrating over volume $\mathcal{V}$ gives the following integral form:

$$\begin{aligned}\iiint_{\mathcal{V}}\frac{D(\tilde{\nu}\widetilde{R_T})}{Dt}d\mathcal{V} \;&+\; \oint_{\partial\mathcal{V}}\frac{\tilde{\nu}_t}{\sigma_\epsilon}\nabla(\tilde{\nu}\widetilde{R_T})\cdot\vec{n}dS-\left(\tilde{\nu}+2\frac{\tilde{\nu}_t}{\sigma_\epsilon}\right)\oint_{\partial\mathcal{V}}\nabla(\tilde{\nu}\widetilde{R_T})\cdot\vec{n}dS\\ &=\; \iiint_{\mathcal{V}}(C_{\epsilon_2}\mathcal{F}_2-C_{\epsilon_1})\sqrt{\tilde{\nu}\widetilde{R_T}\mathcal{P}}\,d\mathcal{V}\end{aligned} \tag{3.8}$$

Using the mean value theorem for the volume integrals and trapezoidal integration for the boundary integrals yields

$$\frac{d(\tilde{\nu}\widetilde{R_T})_i}{dt}+R(\tilde{\nu}\widetilde{R_T})_i=0 \tag{3.9}$$

where the residual, $R(\tilde{\nu}\widetilde{R_T})_i$ is expressed as:

$$\begin{aligned} R(\widetilde{\tilde{\nu} R_T})_i &= \left(\tilde{V}_m\langle R_B\rangle + \widetilde{W}_\theta\langle S_B\rangle\right)_i - (C_{\epsilon_2}\mathcal{F}_2 - C_{\epsilon_1})\sqrt{C_\mu \mathcal{D}_1 \mathcal{D}_2}\mathcal{S}(\widetilde{\tilde{\nu} R_T})_i \\ &\quad -\frac{1}{Re_L \mathcal{V}_i}\left\{\left(\tilde{\nu} + 2\frac{\tilde{\nu}_t}{\sigma_\epsilon}\right)_i \sum_{k=1}^{N_i}\left(\langle R_B\rangle rh\Delta\theta - \langle S_B\rangle h\Delta m\right)_k \right. \\ &\quad \left. - \frac{1}{\sigma_\epsilon}\sum_{k=1}^{N_i}\tilde{\nu}_{t_k}\left(\langle R_B\rangle rh\Delta\theta - \langle S_B\rangle h\Delta m\right)_k\right\} \end{aligned} \tag{3.10}$$

and

$$\tilde{\nu}_t = \mathcal{C}_\mu \mathcal{D}_1 \mathcal{D}_2(\widetilde{\tilde{\nu} R_T})$$

In the above expression the production term, $\mathcal{P}$, is simplified using an approximation suggested in Reference [26]. In Cartesian coordinates, it is described as

$$\mathcal{P} = \tilde{\nu}_t S^2 = \tilde{\nu}_t(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x})^2 \tag{3.11}$$

In the quasi-three-dimensional coordinate system (m, θ), S is given as

$$\mathcal{S} = \frac{\partial V_\theta}{\partial m} + \frac{1}{r}\frac{\partial V_m}{\partial \theta} - V_\theta\left(\frac{1}{r}\frac{dr}{dm}\right) \tag{3.12}$$

The values of $\langle R_B\rangle$ and $\langle S_B\rangle$ are computed by replacing ϕ with $(\widetilde{\tilde{\nu} R_T})$ in Eq. (3.6).

For the Baldwin-Barth one-equation turbulence model, wall units or wall distance is used to account for near-wall viscous effects on turbulence. In addition to employing mean flow quantities, the wall distance, Δy_{wall}, or wall units, y^+, on which damping functions depend have to be evaluated in order to compute the turbulence quantities.

The definition of the wall units is given by

$$y^+ = \frac{\Delta y_{\text{wall}}\sqrt{\tau_{\text{wall}}/\rho_{\text{wall}}}}{\nu_{\text{wall}}} \tag{3.13}$$

where Δy_{wall} is the minimum distance from a vertex point to the wall. Other properties are evaluated at the nearest wall location. Based upon the procedure suggested by Barth [7], the evaluation of y^+ is implemented as follow:

Step 1. Compute the minimum distance from each vertex point to boundary edges.

Step 2. Store information concerning the corresponding boundary edge and weight factor for interpolation of the physical quantities on the boundary edge.

The local wall shear stress, density and viscosity can be interpolated using information stored in step 2. From this information and the minimum distance obtained in step 1, y^+ can be evaluated at each point.

3.1.2.2 Chien Low Reynolds Number k $-\ \epsilon$ Model:

In a similar manner, the finite-volume discretization of Chien's k $-\ \epsilon$ equations, Eqs. (2.21), gives

$$\frac{d\langle\vec{U}_{k\epsilon}\rangle_i}{dt} + R(\langle\vec{U}_{k\epsilon}\rangle)_i = 0 \tag{3.14}$$

where $R(\langle\vec{U}_{k\epsilon}\rangle)_i$ is the residual of the k $-\ \epsilon$ equations

$$R(\langle\vec{U}_{k\epsilon}\rangle)_i = \frac{Q(\langle\vec{U}_{k\epsilon}\rangle)_i}{\mathcal{V}_i} - Re_L^{-1}\langle\vec{K}_{k\epsilon}\rangle_i \tag{3.15}$$

and

$$\begin{aligned} Q(\langle\vec{U}_{k\epsilon}\rangle)_i &= \sum_{k=1}^{N_i}\left[(rh\langle\vec{F}_{k\epsilon}\rangle)_k\Delta\theta_k - (h\langle\vec{G}_{k\epsilon}\rangle)_k\Delta m_k\right] \\ &- Re_L^{-1}\sum_{k=1}^{N_i}\left[(rh\langle\vec{R}_{k\epsilon}\rangle)_k\Delta\theta_k - (h\langle\vec{S}_{k\epsilon}\rangle)_k\Delta m_k\right] \end{aligned} \tag{3.16}$$

The definitions of $\langle\vec{U}_{k\epsilon}\rangle_i, \langle\vec{F}_{k\epsilon}\rangle_k, \langle\vec{G}_{k\epsilon}\rangle_k, \langle\vec{R}_{k\epsilon}\rangle_k, \langle\vec{S}_{k\epsilon}\rangle_k$, and $\langle\vec{K}_{k\epsilon}\rangle_i$ can be obtained in Eqs. (2.21). The wall distance, Δy_{wall}, and wall units, y^+, which are used to account for the near-wall viscous effect on turbulence are evaluated using the same procedures described for the Baldwin-Barth one-equation model.

3.2 Artificial Dissipation

The finite-volume spatial discretization used here is equivalent to a central difference scheme. As with other central difference schemes applied to inviscid flow problems, additional artificial dissipation is required to damp out the high frequency error modes. In viscous flow computations, artificial dissipation is required in convective dominated regions of the flow field. Even though the viscous terms provide physical dissipation in the viscous flow regions, a small mesh length scale is required in all directions to provide enough resolution for the shear stress terms. In practical applications fine mesh spacing is only employed in the direction across to the boundary layer, so the artificial dissipation is still necessary in the streamwise direction for viscous flows.

In inviscid flow calculations, the adaptive artificial dissipation introduced by Jameson and Mavriplis [36, 58] is used to suppress odd-even point decoupling. This model is composed of a blend harmonic(Laplacian) and biharmonic operators which provide a weak but sufficient dissipative term throughout the smooth regions of the flow and a stronger dissipation to suppress oscillations in the regions near shock waves.

For the control volume shown in Figure 3.1 the conservative form of the dissipation operator is given by

$$\begin{aligned}
D(\langle\vec{U}_i\rangle) &= \sum_{j=1}^{N_i} \lambda_{ji}\left[\epsilon_j^2(\langle\vec{U}_j\rangle - \langle\vec{U}_i\rangle) + \epsilon_j^4(\nabla^2\langle\vec{U}_j\rangle - \nabla^2\langle\vec{U}_i\rangle)\right] \qquad (3.17)\\
\nabla^2\langle\vec{U}_i\rangle &= \sum_{j=1}^{N_i}(\langle\vec{U}_j\rangle - \langle\vec{U}_i\rangle)\\
\lambda_{ji} &= (\lambda_j + \lambda_i)/2, \qquad \lambda_i = \frac{\mathcal{V}_i}{\Delta t_i^*}\\
\epsilon_j^2 &= \epsilon' \frac{\sum_{j=1}^{N_i}(\langle p_j\rangle - \langle p_i\rangle)}{\sum_{j=1}^{N_i}(\langle p_j\rangle + \langle p_i\rangle)}, \qquad \epsilon_j^4 = \max(0, \epsilon'' - \epsilon_j^2)
\end{aligned}$$

In the above dissipation formula, λ_{ji}, the average spectral radius of the Euler Jacobian matrix at the cell face is estimated using the value of $\mathcal{V}/\Delta t^*$, $\mathcal{V}_i$ is the volume of the control volume, Δt^* is the time step limit for a Courant number of unity, and ϵ' and ϵ'' are constants. λ_{ji} is chosen to provide a proper scaling in these equations. ϵ_j^2 is proportional to a Laplacian of the pressure. ϵ_j^2 and ϵ_j^4 provide the desired switch between the harmonic and biharmonic operators.

In viscous flow computations, high aspect ratio meshes are usually used to resolve shear layers. Because the mesh length scales in the normal and tangential directions to the body are so different, the nearly isotropic scaling, λ_{ji}, used in inviscid flow calculations may produce excessive dissipation in the tangential direction and therefore reduce accuracy of the viscous calculations. To alleviate this problem, an anisotropic scaling has been introduced to provide different scales for different mesh coordinate directions on high aspect ratio cells.

3.2.1 Eigenvalue Scaling

For structured mesh calculations, the eigenvalue scaling approach [15] gives

$$\lambda_{ji} = \begin{cases} \lambda_\xi, & \text{for } \xi - \text{direction} \\ \lambda_\eta, & \text{for } \eta - \text{direction} \end{cases} \tag{3.18}$$

where λ_ξ and λ_η are the maximum eigenvalues for the Euler Jacobian matrices in each mesh coordinate direction:

$$\lambda_\xi = (|V_\xi| + a)\Delta\eta, \qquad \lambda_\eta = (|V_\eta| + a)\Delta\xi \tag{3.19}$$

Since there is no apparent mesh coordinate direction for unstructured meshes, a scaling is performed on each edge of a control volume in order to extend this approach to the present unstructured scheme. Consider a local coordinate system (n, s) on which s is tangential to the edge (ji), and n is normal to the edge. Then the directional eigenvalue along edge (ji) direction can be estimated as:

$$\lambda_s = (|V_s| + a)\Delta n \tag{3.20}$$

In the above expression, V_s is the s-component velocity, Δn is the length scale in the tangential direction to edge (ji), and a is the sound speed. Using information at points i and j we can estimated all the required parameters in Eq. (3.20) except for Δn. Further information is required to estimate Δn.

In the edge-based data structure, an additional triangle on either side of the edge can be constructed. the distance between centroids of these triangles is used to estimate the normal length scale Δn. Replacing the λ_{ji} with the directional eigenvalue λ_s gives the anisotropic scaling for large aspect ratio unstructured mesh cells.

Although the anisotropic scaling gives appropriate dissipation on large aspect ratio meshes, various investigators have found that such a scaling may not produce enough dissipation when mesh becomes highly stretched. This leads to slow convergence. To overcome this problem, a new scaling based on the combination of isotropic and anisotropic scaling was introduced by Martinelli [52]. On a structured mesh, the scaling becomes

$$\bar{\lambda}_\xi = \lambda_\xi \left(1 + \left(\frac{\lambda_\eta}{\lambda_\xi}\right)^\alpha\right), \quad \bar{\lambda}_\eta = \lambda_\eta \left(1 + \left(\frac{\lambda_\xi}{\lambda_\eta}\right)^\alpha\right) \tag{3.21}$$

On the present unstructured mesh, it is modified as

$$\lambda_{ji} = \lambda_s \left(1 + \left(\frac{\lambda_n}{\lambda_s}\right)^\alpha\right) \tag{3.22}$$

In the above expression, α is a constant. λ_n is the eigenvalue in the direction normal to the edge and is given as

$$\lambda_n = (|V_n| + a)\Delta s \tag{3.23}$$

where V_n is the n-component velocity and Δs is computed as the distance between points i and j.

3.2.2 Local Velocity Scaling

For viscous computations, the physical dissipation in near wall regions produces enough dissipation in the direction normal to a surface, because a fine mesh spacing is used in this direction. This allows the artificial dissipation term in the normal direction to be reduced. In the present approach, a local velocity scaling is incorporated into the directional eigenvalue scaling. That is

$$\lambda_{ji} = \left[q^2 \sin^2\theta + \cos^2\theta\right] \lambda_s \left(1 + \left(\frac{\lambda_n}{\lambda_s}\right)^\alpha\right) \tag{3.24}$$

where q is the local total velocity and θ is the angle between edge (ij) and a vector tangential to the nearest wall location.

Adding the modified artificial dissipation to Eqs. (3.3), (3.9), and (3.14) results in the final form of the finite-volume spatial discretization:

$$R(\langle\vec{U}_i\rangle) = \frac{1}{V_i}[Q(\langle\vec{U}_i\rangle) - D(\langle\vec{U}_i\rangle)] - K(\langle\vec{U}_i\rangle) \tag{3.25}$$

The dissipation term adds a third-order error in smooth regions of the flow, and a first-order error near shock waves, so a second-order accurate finite-volume scheme is preserved except in regions close to shock waves.

The implementation of the dissipative terms requires two loops over all the edges in the domain. In the first loop a harmonic (Laplacian) operator is computed at each node. In the second loop the blended adaptive artificial dissipation is obtained by simultaneously collecting the Laplacian terms needed for the biharmonic operator and the terms for the harmonic operator.

3.3 Runge-Kutta Time-Integration Scheme

The semi-discrete equations are integrated in time using the following q-stage Runge-Kutta scheme:

$$\begin{aligned}
\langle\vec{U}\rangle^{(0)} &= \langle\vec{U}\rangle^{n} \\
\langle\vec{U}\rangle^{(1)} &= \langle\vec{U}\rangle^{(0)} - \alpha_1\Delta t R(\langle\vec{U}\rangle^{(0)}) \\
&\vdots \\
\langle\vec{U}\rangle^{(q-1)} &= \langle\vec{U}\rangle^{(0)} - \alpha_{q-1}\Delta t R(\langle\vec{U}\rangle^{(q-2)}) \\
\langle\vec{U}\rangle^{(q)} &= \langle\vec{U}\rangle^{(0)} - \alpha_{q}\Delta t R(\langle\vec{U}\rangle^{(q-1)}) = \langle\vec{U}\rangle^{n+1}
\end{aligned} \tag{3.26}$$

where the superscripts in parentheses represent the particular stage of the scheme. $\langle\vec{U}\rangle^n$ denotes the value of $\langle\vec{U}\rangle$ at time step n, $\alpha_1, \cdots, \alpha_q$ are the coefficients of the particular multi-stage scheme, and $R(\langle\vec{U}\rangle)$ is the residual of the spatial discretization as given in Eq. (3.25). Runge-Kutta schemes were originally developed for high temporal accuracy. However, for steady-state problems, time accuracy is not as important as efficiency of the scheme. This has led to the use of hybrid multistage schemes which allow the convective and dissipative terms to be evaluated separately.

For the results presented here a standard four-stage Runge-Kutta scheme with coefficients $\alpha_1 = 1/4$, $\alpha_2 = 1/3$, $\alpha_3 = 1/2$ and $\alpha_4 = 1$ is used. The viscous operator $(\langle\vec{R}\rangle, \langle\vec{S}\rangle)$ and the artificial dissipation operator $D(\langle\vec{U}_i\rangle)$ are evaluated in the first stage step and frozen for subsequent stages. In this way the CPU time required for the evaluation of the dissipative term is reduced by a factor of four.

3.3.1 Stability Criteria

The time step for all explicit time marching schemes is restricted by some stability limit usually expressed in terms of the *CFL* number. For the Euler equations the maximum allowable time step is described as:

$$\Delta t = CFL \cdot \Delta t_c \tag{3.27}$$

where Δt_c is the convective time step. For the Navier-Stokes equations the time step limits [54] due to both convective and diffusive effects must be considered, which may be expressed as:

$$\Delta t = CFL \frac{\Delta t_c \cdot \Delta t_v}{\Delta t_c + \Delta t_v} \tag{3.28}$$

where *CFL* is the Courant number and Δt_v is the diffusive time step.

The Courant number for the multi-stage scheme is obtained by performing the von Neumann stability analysis for the one-dimensional model problem:

$$u_t + u_x + \mu \Delta x^3 u_{xxxx} = 0 \tag{3.29}$$

The stability region for the hybrid multi-stage scheme depends on the discrete spatial operator, the coefficients, α_i, the number of stages, q, the manner of evaluations of the artificial dissipation, and the smoothing coefficient, μ. Although for a m-stage scheme a maximum allowable stability limit, $CFL \leq m-1$, can be obtained through optimizing the coefficients, α_i, such a scheme is not necessarily optimal in terms of computational effort or accuracy. The present four-stage Runge-Kutta scheme, using the centered difference operator with the dissipation computed in the first stage and frozen for subsequent stages, has been shown to be stable with a Courant number $CFL \leq 2.6$ for the one-dimensional model problem with $\mu = 1/32$. Note that this analysis only provides a reference value of the maximum Courant number for the present scheme. In practical applications values less than the maximum Courant number might be needed to assure stability.

The values of Δt_c and Δt_v are obtained by performing the von Neumann stability analysis on the finite difference equations. Since it is not possible to perform a stability analysis directly on the present unstructured scheme, the stability limit for the multi-stage scheme will be inferred from the corresponding structured formulation. The time steps Δt_c and Δt_v for a general two dimensional structured mesh [43] are given as:

$$\Delta t_c = \frac{\mathcal{V}_i}{\lambda_{c_\xi} + \lambda_{c_\eta}}, \quad \Delta t_v = \frac{\mathcal{V}_i}{\lambda_{v_\xi} + \lambda_{v_\eta}} \tag{3.30}$$

where

$$\begin{aligned}
\lambda_{c_\xi} &= (|V_\xi| + a)\Delta\eta, \quad \lambda_{c_\eta} = (|V_\eta| + a)\Delta\xi \\
\lambda_{v_\xi} &= \frac{1}{Re_L \mathcal{V}_i}\left[\frac{4\gamma\tilde{\mu}_e}{\langle\rho\rangle(p_r)_e}\Delta\xi^2 + \frac{\tilde{\mu}_e}{3\langle\rho\rangle}\Delta\xi\Delta\eta\right] \\
\lambda_{v_\eta} &= \frac{1}{Re_L \mathcal{V}_i}\left[\frac{4\gamma\tilde{\mu}_e}{\langle\rho\rangle(p_r)_e}\Delta\eta^2 + \frac{\tilde{\mu}_e}{3\langle\rho\rangle}\Delta\xi\Delta\eta\right] \\
\Delta\xi &= \sqrt{x_\eta^2 + y_\eta^2}, \quad \Delta\eta = \sqrt{x_\xi^2 + y_\xi^2}
\end{aligned}$$

Although the edge-based approach provides local mesh coordinate information for evaluating the above equations, it is expensive to compute all the information for each edge. In order to reduce the work of evaluating the time steps in the present unstructured scheme, the above equations are simplified using time steps in the ξ and η directions.

$$\Delta t_c = \min\{\Delta t_{c_\xi}, \Delta t_{c_\eta}\}, \quad \Delta t_v = \min\{\Delta t_{v_\xi}, \Delta t_{v_\eta}\} \tag{3.31}$$

After neglecting the cross terms in λ_v, the time steps in the ξ and η directions might be described as:

$$\Delta t_{c_\xi} = \frac{\Delta\xi}{|V_\xi| + a}, \quad \Delta t_{c_\eta} = \frac{\Delta\eta}{|V_\eta| + a}$$

$$\Delta t_{v_\xi} = Re_L \Delta\eta^2 \frac{\langle\rho\rangle(p_r)_e}{4\gamma\tilde{\mu}_e}, \quad \Delta t_{v_\eta} = Re_L \Delta\xi^2 \frac{\langle\rho\rangle(p_r)_e}{4\gamma\tilde{\mu}_e}$$

For the cell-vertex unstructured scheme, it is not necessary to compute the time steps in both tangential and normal directions for every edge, because the tangential direction of an edge can be approximated by a normal direction of another edge when looping over all the edges of a control volume. In the present work, the time step on the direction normal to a edge is used. Therefore, the time steps corresponding to unit Courant number on each edge may be rewritten as:

$$\Delta t_{c_k} = \{\frac{\Delta r_n}{|\vec{V} \cdot \vec{n}| + a}\}_k, \qquad \Delta t_{v_k} = \{\frac{Re_L \langle\rho\rangle (p_r)_e \Delta r_s^2}{4\gamma \tilde{\mu}_e}\}_k \tag{3.32}$$

In the above expression subscript k denotes the k^{th} edge surrounding the control volume, Δr_n is the normal distance from interior point i to the edge, Δr_s is the distance between two end points of the edge, velocity $\vec{V}$ and speed of sound a are taken as averaged values along edge k, $\vec{n}$ is the unit vector normal to the edge, and $(|\vec{V} \cdot \vec{n}| + a)_k$ is the maximum wave propagation speed along the direction normal to the edge.

In order to advance the solution in time with the maximum allowable speed, the maximum *CFL* number is always applied in the estimation of time steps.

3.3.2 Local Time-Stepping

If only the steady state solution is of interest, convergence to the steady state solution may be accelerated by sacrificing the time accuracy and marching the equations at each node in time by the maximum permissible time step, as determined by a local stability analysis. Based on the above edge-based approach the permissible time step at node i may be taken as the minimum of the allowable time steps from all surrounding edges to that node.

$$\Delta t_{c_i} = \min_{k \in N_i}\{\Delta t_{c_k}\}, \qquad \Delta t_{v_i} = \min_{k \in N_i}\{\Delta t_{v_k}\}$$

$$\Delta t_i = CFL_{\max} \frac{\Delta t_{c_i} \cdot \Delta t_{v_i}}{\Delta t_{c_i} + \Delta t_{v_i}} \tag{3.33}$$

3.3.3 Implicit Residual Averaging

Another simple method of increasing the allowable time step is implicit residual averaging. The concept of implicit residual averaging is to increase the stability limit of the basic time-integration scheme by implicitly smoothing the residuals. The implementation is performed by implicitly solving the equation:

$$\bar{R}_i = R_i + \epsilon \nabla^2 \bar{R}_i \tag{3.34}$$

where $\bar{R}'s$ are the smoothed residuals, and ϵ is a smoothing coefficient. The stability limit of the smoothed scheme can be estimated using the following inequality [34]:

$$\epsilon \geq \frac{1}{4}\left(\frac{CFL^2}{CFL^{*2}} - 1\right) \tag{3.35}$$

where CFL^* is the Courant number of the unsmoothed scheme.

For the unstructured formulation, a Jacobi iteration method has been suggested [58] for solving the implicit residual smoothed equation. Due to the fact that the computational work involved in solving Eq. (3.34) should not outweight the gain in convergence rate, two iterations were employed. With the same Laplacian operator as in the artificial dissipative term, the Jacobi iteration scheme gives

$$\bar{R}_i^{n+1} = \frac{R_i + \epsilon \sum_{k=1}^{N_i} \bar{R}_k^n}{1 + N_i \epsilon} \tag{3.36}$$

It is important to note that the Jacobi iteration method converges slowly unless the matrix is strongly diagonally dominant. This implies that a small value of ϵ is required and this restricts the increase of the stability limit. In the present work, it has been found that using two Jacobi iterations with $\epsilon = 0.5$ saves about 20 % in CPU time.

In order to improve the performance of the residual averaging method, an alternate iterative method using an approximate factorization is employed to enhance the convergence in solving Eq. (3.34). In matrix form, Eq. (3.34) gives

$$[M_{ij}]\bar{R}_i = R_i \tag{3.37}$$

$[M_{ij}]$ can be factorized using the corresponding diagonal lumped mass matrix [63], that is

$$[M_{ij}] = [D_{ij}] + ([M_{ij}] - [D_{ij}]) \tag{3.38}$$

where

$$D_{ij} = \sum_{j=1}^{N} |M_{ij}| \, \delta_{ij}$$

Substituting Eq. (3.38) into Eq. (3.37) gives

$$[D_{ij}] \, \delta \bar{R}_i = R_i - [M_{ij}] \, \bar{R}_i^n$$

$$\bar{R}_i^{n+1} = \bar{R}_i^n + \delta \bar{R}_i \tag{3.39}$$

The above equation can be simplified as:

$$\bar{R}_i^{n+1} = \frac{R_i + \epsilon N_i \bar{R}_i^n + \epsilon \sum_{k=1}^{N_i} \bar{R}_k^n}{1 + 2 N_i \epsilon} \tag{3.40}$$

This method requires about the same computing work as the Jacobi iteration method. The advantage in this iteration scheme is that the value of ϵ can be increased up to 1.0 with a resulting 40 % savings in CPU time. This method is used in the present work.

3.4 Initial Conditions

As mentioned previously, the mean flow equations and turbulence transport equations represent an initial-boundary-value problem. Appropriate initial and boundary conditions are needed to solve these equations. For steady state solutions, initial conditions are used only as a starting point for time-marching and only affect the convergence rate to the steady state solution. However, the imposed conditions should not be so inconsistent that the time-marching scheme diverges.

3.4.1 Mean Flow Equations

For external flows initial conditions are not critical to the iteration, and uniform freestream conditions are generally imposed. For turbomachinery applications, there is usually high loading and high turning of the flow. It is not generally possible to start a solution with constant initial conditions. In the present work the quasi-one-dimensional solution introduced by Chima [18] is imposed to provide a smooth variation of initial flow conditions.

For quasi-one-dimensional flows with area change the continuity and energy equations which state the conservation of mass and rothalpy are

$$\dot{m} = \rho r h \Delta\theta W \cos\alpha = \text{constant} \tag{3.41}$$

$$I = CpT' - r\Omega V_\theta = CpT'' - \frac{1}{2}r^2\Omega^2 = \text{constant} \tag{3.42}$$

where $()'$ and $()''$ values denote absolute and relative total conditions, respectively, $\Delta\theta$ is the blade spacing, and I is the rothalpy. For flow outside the blade row the free vortex flow assumption gives

$$rV_\theta = \text{constant} \tag{3.43}$$

From Eq. (3.41) the relative total velocity can be described as

$$W^2 = (\frac{\dot{m}}{\rho r h \Delta\theta})^2 + W_\theta^2 \tag{3.44}$$

From the isentropic relations we have

$$\frac{\rho}{\rho''} = (\frac{T}{T''})^{\frac{1}{\gamma-1}}$$

$$\frac{T}{T''} = 1 - \frac{W^2}{2CpT''}$$

So,

$$\rho = \rho''(1 - \frac{W^2}{2CpT''})^{\frac{1}{\gamma-1}} \tag{3.45}$$

Using these relations Eq. (3.44) may be rewritten as

$$W^2 - \phi^2(1 - \frac{W^2}{2CpT''})^{\frac{-2}{\gamma-1}} - W_\theta^2 = 0 \tag{3.46}$$

where

$$\phi = \frac{\dot{m}}{\rho'' r h \Delta\theta}$$

Within the blade row the free vortex assumption is replaced with the assumption that the flow angle α varies linearly through the blade row. The relative total velocity can be described as

$$W^2 - \phi(1 - \frac{W^2}{2CpT''})^{\frac{-2}{\gamma-1}} = 0 \tag{3.47}$$

where

$$\phi = \frac{\dot{m}}{\rho'' r h \Delta\theta \cos\alpha}$$

After the inflow conditions and the relative flow angle at the trailing edge are specified, the solution of the quasi-one-dimensional flow can be obtained from upstream to downstream. For each point upstream of the blade the relative total velocity is obtained using a Newton iteration method to solve Eq. (3.44). Other flow properties are then obtained thru Eq. (3.43), constant rothalpy, and the isentropic relations. Within the blade row Eq. (3.47) is solved for W. Once the flow condition are computed at trailing edge, Eq. (3.46) is used again for the downstream region.

The quasi-one-dimensional solution is not necessarily consistent with the flow tangency or the no-slip condition. To avoid an abrupt change of flux at a near wall cell, the flow tangency or no-slip condition is enforced at the blade surface and a smoothing operator is then used to smooth the flow quantities near blade edges.

3.4.2 Turbulence Transport Equations

For the Baldwin-Barth one-equation model, the turbulence field variable is initialized using specified freestream turbulent viscosity:

$$\widetilde{R_T} = \widetilde{R_{T_\infty}} < 1$$

For the Chien low Reynolds number k $-$ ϵ model, the initial distribution of $\tilde{k}$ and $\tilde{\epsilon}$ are based on specified freestream turbulent viscosity and turbulence intensity. The turbulence intensity is defined as

$$\mathcal{T}_\infty = \frac{\sqrt{\frac{2}{3}\tilde{k}_\infty}}{V_\infty} \tag{3.48}$$

In the present work, the values of $\mathcal{T}_\infty$ and $\tilde{\nu}_{t\infty}$ are specified as

$$\mathcal{T}_\infty < -0.1\ \%, \qquad \tilde{\nu}_{t\infty} < 1 \tag{3.49}$$

Substituting Eq. (3.49) into Eq. (3.48) yields the freestream turbulent kinetic energy. Then, the freestream dissipation rate can be obtained as

$$\tilde{\epsilon}_\infty = C_\mu \frac{\tilde{k}^2_\infty}{\tilde{\nu}_{t\infty}}$$

3.5 Boundary Conditions

For a cell-vertex scheme flow properties are required at boundary nodes. In the two-dimensional or quasi-three-dimensional mean-flow equations four boundary conditions are required along each boundary of the domain. In addition to the physical boundary conditions mentioned in the previous chapter, additional numerical boundary conditions might be needed to close the set of equations. When differential turbulence transport equations are used for the closure of the Favre-averaged Navier-Stokes equations, additional boundary conditions are required on each differential transport equation. The boundary conditions depend on the type of turbulence models used.

3.5.1 Inflow/Outflow Boundary Conditions

At inflow and outflow boundaries, a nonreflecting or radiation boundary condition based on a characteristic analysis is implemented to allow the numerical errors to propagate out of the domain. The idea of a characteristic formulation is try to correctly capture the physics of the flow in terms of a wave propagation problem by considering the characteristic waves in incoming and outgoing directions separately. For the incoming waves, the characteristic variables are held at prescribed values, while for outgoing waves the variables are extrapolated from the interior.

The characteristic formulation is derived in a local coordinate system tangential and normal to the boundary. Assuming the normal variation to be much larger than the tangential variation of state variables U, the Euler equations may be expressed as

$$\frac{\partial \vec{W}}{\partial t} + \Lambda \frac{\partial \vec{W}}{\partial n} = 0 \tag{3.50}$$

where

$$\Lambda = \mathrm{Diag}\{\lambda_1,\ \lambda_2,\ \lambda_3,\ \lambda_4\}$$

$$\vec{W} = \begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ W_4 \end{bmatrix} = \begin{bmatrix} \rho - p/\bar{a}^2 \\ q_s \\ (q_n + p/\bar{\rho}\bar{a})/\sqrt{2} \\ (-q_n + p/\bar{\rho}\bar{a})/\sqrt{2} \end{bmatrix}, \quad \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} \bar{q}_n \\ \bar{q}_n \\ \bar{q}_n + \bar{a} \\ \bar{q}_n - \bar{a} \end{bmatrix} \tag{3.51}$$

where W_i is the characteristic variable, λ_i is the associated wave speed, n is the local unit inward normal vector to the boundary, and q_n and q_s are the normal and tangential velocities, respectively. Barred values are evaluated using values frozen from the last Runge-Kutta stage step.

Along each characteristic line defined by the corresponding wave speed, Eq. (3.50) reduces to a set of ordinary differential equations. That is,

$$\frac{dW_i}{dt} = 0, \qquad \text{along } \frac{dn}{dt} = \lambda_i \tag{3.52}$$

Employing Eq. (3.52) along the boundaries of the computational domain gives characteristic formulations for solving for the flow properties at boundary nodes. The discrete form of Eq. (3.52) involves the characteristic variables propagating from outside the domain as well as from the interior. The characteristic variable W_i must be specified if the corresponding wave speed λ_i originates from outside the domain. Otherwise, it is extrapolated from the interior. Based on the local velocity normal to the boundary (see Figure 3.2), there are four possible boundary formulations summarized as follows:

For subsonic inflow ($0 < \bar{q}_n < \bar{a}$), there are three incoming waves corresponding to λ_1, λ_2 and λ_3, and one outgoing wave corresponding to λ_4. Therefore, the characteristic variables, W_1, W_2, and W_3 are prescribed and W_4 is extrapolated from the interior. This leads to the following system of equations.

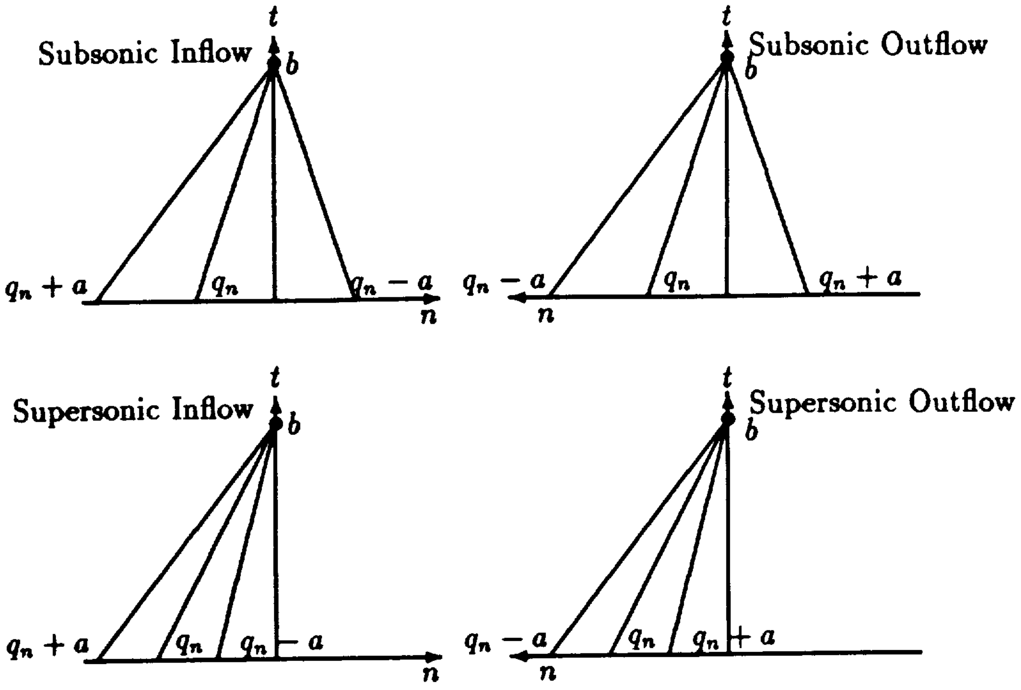


Figure 3.2 Characteristics at far field boundaries.

$$\begin{aligned}
\rho_b - p_b/\bar{a}^2 &= \rho_{pr} - p_{pr}/\bar{a}^2 \\
q_{s_b} &= q_{s_{pr}} \\
q_{n_b} + p_b/\bar{\rho}\bar{a} &= q_{n_{pr}} + p_{pr}/\bar{\rho}\bar{a} \\
-q_{n_b} + p_b/\bar{\rho}\bar{a} &= -q_{n_{ex}} + p_{ex}/\bar{\rho}\bar{a}
\end{aligned} \tag{3.53}$$

where the subscripts b, pr, and ex represent the boundary, prescribed, and extrapolated values, respectively. Solving Eq. (3.53) for the boundary values gives

$$\begin{aligned}
q_{s_b} &= q_{s_{pr}} \\
p_b &= \frac{1}{2}\left[p_{ex} + p_{pr} + \bar{\rho}\bar{a}(q_{n_{pr}} - q_{n_{ex}})\right] \\
\rho_b &= \rho_{pr} + (p_b - p_{pr})/\bar{a}^2 \\
q_{n_b} &= q_{n_{pr}} + (p_{pr} - p_b)/\bar{\rho}\bar{a}
\end{aligned} \tag{3.54}$$

For supersonic inflow ($0 < \bar{a} < \bar{q}_n$), all four waves (λ_i) come inward, so all four characteristic variables (W_i) are prescribed. This is equivalent to prescribing all four flow properties.

$$\begin{aligned} q_{s_\mathrm{b}} &= q_{s_\mathrm{pr}} \\ p_\mathrm{b} &= p_\mathrm{pr} \\ \rho_\mathrm{b} &= \rho_\mathrm{pr} \\ q_{n_\mathrm{b}} &= q_{n_\mathrm{pr}} \end{aligned} \tag{3.55}$$

For subsonic outflow $(-\bar{a} < \bar{q}_n < 0)$, there is only incoming wave corresponding to λ_3, and thus the corresponding characteristic variable W_3 is prescribed. Using the same procedure as for the subsonic inflow case the boundary values are obtained as

$$\begin{aligned} q_{s_\mathrm{b}} &= q_{s_\mathrm{pr}} \\ p_\mathrm{b} &= \frac{1}{2}\left[p_\mathrm{ex} + p_\mathrm{pr} + \bar{\rho}\bar{a}(q_{n_\mathrm{pr}} - q_{n_\mathrm{ex}})\right] \\ \rho_\mathrm{b} &= \rho_\mathrm{ex} + (p_\mathrm{b} - p_\mathrm{ex})/\bar{a}^2 \\ q_{n_\mathrm{b}} &= q_{n_\mathrm{ex}} + (p_\mathrm{b} - p_\mathrm{ex})/\bar{\rho}\bar{a} \end{aligned} \tag{3.56}$$

For supersonic outflow $(\bar{q}_n < -\bar{a})$, all four waves λ_i propagate from the interior, therefore all four characteristic variables W_i are extrapolated from the interior. This is equivalent to extrapolating all flow properties from the interior.

$$\begin{aligned} q_{s_\mathrm{b}} &= q_{s_\mathrm{ex}} \\ p_\mathrm{b} &= p_\mathrm{ex} \\ \rho_\mathrm{b} &= \rho_\mathrm{ex} \\ q_{n_\mathrm{b}} &= q_{n_\mathrm{ex}} \end{aligned} \tag{3.57}$$

When the prescribed and the extrapolated values are given, these relations, Eqs. (3.54) to (3.57), determine the primitive variables ρ, q_n, q_s, and p at the boundary. Energy is obtained from the equation of state.

For the two-dimensional airfoil case the far field boundary is, in practice, placed at a finite distance from the airfoil. In order to accurately approximate the asymptotic state of a uniform freestream condition the far field boundary must be placed at a distance of O(100) airfoil chord lengths from the airfoil. For subsonic flows the distance from the outer boundary of the domain to the airfoil can be reduced by replacing the uniform freestream far field condition with the vortex far field boundary condition. Based on the work of Usab [83] the vortex far field boundary condition is defined by the combination of freestream and a compressible point vortex centered at the airfoil quarter chord. Use of the point vortex correction allows a reduction in the distance to the far field boundary by a factor of 10 [21].

For turbomachinery applications the total pressure, total temperature and whirl rV_θ are specified at the inlet with exit pressure imposed at the outlet. In order to explicitly prescribe these quantities, an alternate boundary formulation based on Riemann invariants is used at the inlet boundary. At the outlet boundary a characteristic formulation is used.

The Riemann invariant formulation is based on the work of Chima [18]. In this formulation, an upstream-running Riemann invariant is extrapolated from the domain interior:

$$\partial_m R^- = -\frac{1}{V_m - a}\left[(V_\theta^2 + aV_m)(\frac{1}{r}\frac{dr}{dm}) + aV_m(\frac{1}{h}\frac{dh}{dm})\right] \tag{3.58}$$

where

$$R^- = V_m - \frac{2a}{\gamma - 1}$$

is the upstream-running Riemann invariant.

For subsonic flow at the inlet the Riemann invariant is extrapolated using backward-differencing in Eq. (3.58). Based on R^-, the isentropic relation, the specified total temperature and whirl rV_θ, the velocity V_m at the boundary is given by:

$$V_m = \frac{(\gamma-1)R^- + \sqrt{(\gamma+1)(4CpT' - 2V_\theta^2) - 2(\gamma-1)(R^-)^2}}{(\gamma+1)} \tag{3.59}$$

Density and pressure are computed using the isentropic relations and the specified inlet conditions. The nondimensional specified total pressure and total temperature yield the total density:

$$\rho' = (\gamma M_\infty^2)\frac{p'}{T'}$$

From R^-, V_m, and rV_θ, the total velocity and speed of sound at inlet boundary are found:

$$\begin{aligned} Q_b &= \sqrt{V_m^2 + V_\theta^2} \\ a_b &= \frac{\gamma - 1}{2}(V_m - R^-) \end{aligned} \tag{3.60}$$

Then, the isentropic relations and equation of state give:

$$\begin{aligned} p_b &= p'\left[1 + \frac{\gamma - 1}{2}\left(\frac{Q_b}{a_b}\right)\right]^{\frac{-\gamma}{\gamma-1}} \\ \rho_b &= \rho'\left[\frac{p_b}{p'}\right]^{\frac{1}{\gamma}} \\ e_b &= \frac{p_b}{\gamma - 1} + \frac{\rho_b}{2}(V_m^2 + V_\theta^2) \end{aligned} \tag{3.61}$$

For supersonic inflow all four quantities are specified.

At the outlet, the characteristic formulation with prescribed exit pressure are used for subsonic flow, while all four variables are extrapolated from the interior on supersonic flow.

3.5.2 Solid Wall Boundary Conditions

The cell-vertex scheme flux balance is evaluated using flow properties at node points. This requires boundary values to be updated at each time step. For inviscid flow calculations, the flow tangency condition is enforced at wall. Along solid wall boundaries the imposition of the no flux condition in flux balance calculations does not guarantee that the flow tangency condition will be satisfied. To enforce the flow tangency condition the solid wall boundary conditions are implemented in a predictor-corrector manner using a characteristic analysis. With $q_n = 0$ in Eq. (3.51) the characteristic formulation shows one outgoing wave requiring one prescribed condition.

The state variables are first predicted based on the Runge-Kutta scheme:

$$\vec{U}_p = \vec{U}^n + \Delta\vec{U} \tag{3.62}$$

Substituting the condition $q_n = 0$ and the other variables from the predicted state into Eq. (3.51) results in a characteristic formulation of the solid wall boundary conditions. The boundary values are then corrected using the following formulation.

$$\begin{aligned} q_{n_c} &= 0 \\ q_{s_c} &= q_{s_p} \\ p_c &= p_p + q_{n_p}\bar{\rho}\bar{a} \\ \rho_c &= \rho_p + q_{n_p}\bar{\rho}/\bar{a} \end{aligned} \tag{3.63}$$

where the subscripts p and c represent the predicted and corrected values, respectively. Again energy is computed using the equation of state.

For viscous flow calculations, the no slip condition with either adiabatic or isothermal wall condition is specified at the wall. For the cell-vertex scheme an additional numerical boundary condition is required to update all boundary flow properties. In practice a zero normal pressure gradient is imposed at the wall. The boundary conditions imposed at the wall are

$$\begin{aligned} \vec{V} &= \vec{V}_{\text{wall}} \\ T &= T_{\text{wall}}, \quad \text{or} \quad (\frac{\partial T}{\partial n})_{\text{wall}} = 0 \\ (\frac{\partial p}{\partial n})_{\text{wall}} &= 0 \end{aligned} \tag{3.64}$$

Density is computed using the isentropic relations and equation of state.

3.5.3 Periodic Boundary Conditions

For a single-blade-passage a spatial periodic boundary condition is imposed on the upper and lower boundaries between which an equal pitch spacing is maintained from inlet to exit. In the present approach the points are placed on boundaries and no imaginary cells are placed outside periodic boundaries. Along the periodic boundaries the flux balance is performed in the same way as at the interior points. The net change at a boundary point is simply the sum of the partial sums from the corresponding points on the upper and lower boundaries (see Figure 3.3).

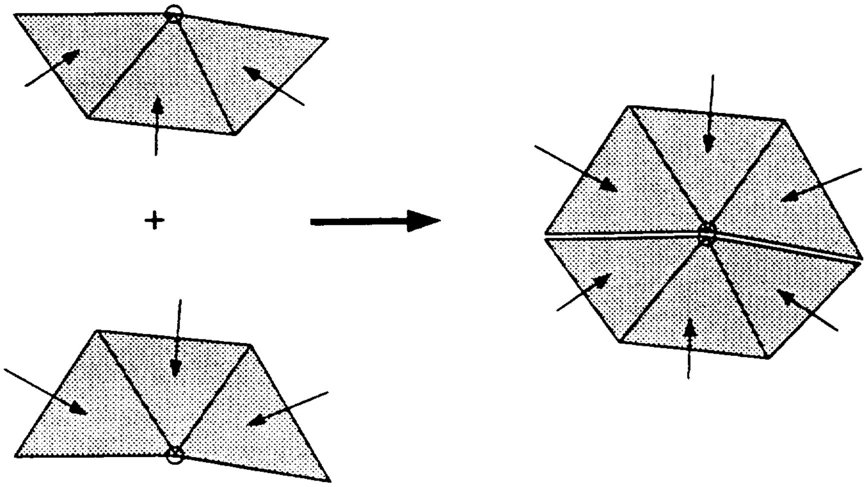


Figure 3.3 Flux balance implementation for periodic boundary points.

3.5.4 Special Boundary Conditions

For turbomachinery applications special boundary conditions are required. Blades designed on an analytical basis provide good test cases for the validation of numerical methods. Many of these cases have an open profile at the blade trailing edge. Changing the blade shape to close the trailing edge often results in a very different solution. In order to obtain an accurate numerical solution, the nonclosed trailing edge needs to be correctly modeled. One example of a nonclosed profile is the "viscous" blade profile obtained by adding the boundary layer displacement thickness to a physical blade. The inviscid solution computed using this "viscous" profile is equivalent to an inviscid solution about the true blade with surface injection used to account for the boundary layer displacement effects. Based on this analogy the nonclosed trailing edge is modeled as a free jet which accounts for the boundary layer displacement inviscidly.

In the present work, the nonclosed trailing edge is modeled as a uniform free jet at the blade trailing edge. The uniform flow properties are evaluated using the averaged values of the upper and lower blade surface properties at the trailing edge (see Figure 3.4):

$$\begin{aligned}
\rho_{\mathrm{j}} &= (\rho^L + \rho^U)/2 \\
u_{\mathrm{j}} &= (u^L + u^U)/2 \\
v_{\mathrm{j}} &= (v^L + v^U)/2 \\
e_{\mathrm{j}} &= (e^L + e^U)/2 \qquad (3.65)
\end{aligned}$$

where the subscript j denotes the jet flow and the superscripts L and U represent the lower and the upper end points, respectively. Pressure is obtained from the isentropic relations and equation of state.

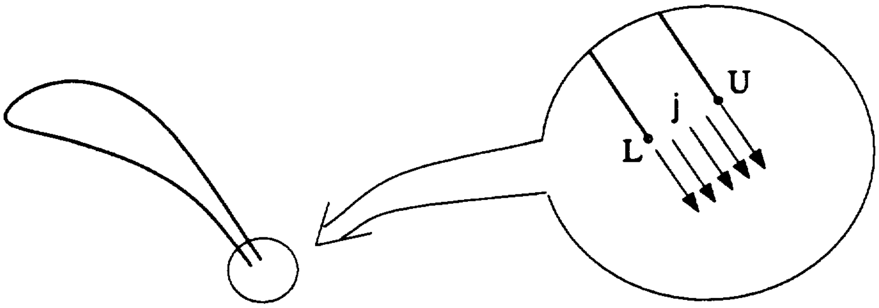


Figure 3.4 Uniform jet assumption at the nonclosed trailing edge.

For turbomachinery applications the exit pressure is set or measured in an averaged (mass- or area-averaged) sense. For a uniform outlet flow this is numerically equivalent to specifying a constant pressure at the exit. However, there are situations in which the exit flow is not uniform. For example, a turbine blade operating at a high speed it is likely to have oblique shock or expansion waves which propagate to the exit boundary. If the axial exit velocity is subsonic, one boundary condition is still required. Because the exit flow is not uniform, it is physically inappropriate to specify a uniform exit pressure at the exit. Therefore, nonuniform exit boundary conditions are required.

In order to account for the nonuniformity the exit pressure is imposed in a predictor/corrector fashion. In the predictor step the state variables are extrapolated from the interior and a mass- or area-averaged exit pressure is evaluated using the extrapolated value. The exit pressure is then corrected by subtracting the difference between the specified and the predicted averaged exit pressure from the extrapolated value:

$$\begin{aligned} \Delta p &= p_{\text{exit}} - p_{p_{ave}} \\ p_c &= p_p + \Delta p \end{aligned} \tag{3.66}$$

where the subscripts c and ex denote the predicted and corrected state, respectively, p_{ave} is a predicted averaged value, and exit represents the specified exit value.

A variation of this boundary condition is that the exit pressure is only known at one boundary point. Since there is no averaged exit pressure available, the nonuniform exit pressure is imposed using a slight modification of the above procedure. In Eq. (3.66), the averaged pressure difference is replaced with the pressure difference computed at one boundary node.

3.5.5 Boundary Conditions for the Turbulence Transport Equations

For the Baldwin-Barth one-equation model, the boundary conditions stated in Reference [6] are used.

- Solid Walls: Specify $\widetilde{R_T} = 0$
- Inflow ($\vec{V} \cdot \vec{n} < 0$): Specify $\widetilde{R_T} = (\widetilde{R_T})_\infty < 1$
- Outflow ($\vec{V} \cdot \vec{n} > 0$): Extrapolate $\widetilde{R_T}$ from interior values

For Chien's low Reynolds number $\mathrm{k} - \epsilon$ model, the boundary conditions listed in Reference [43] are used.

- Solid Walls: Specify $\tilde{\mathrm{k}} = 0$, and $\tilde{\epsilon} = 0$
- Inflow ($\vec{V} \cdot \vec{n} < 0$): Specify $\tilde{\mathrm{k}} = \mathrm{k}_\infty$, and $\tilde{\epsilon} = \epsilon_\infty$
- Outflow ($\vec{V} \cdot \vec{n} > 0$): Extrapolate $\tilde{\mathrm{k}}$ and $\tilde{\epsilon}$ from interior values

4. MESH GENERATION

In the present work an adaptive remeshing procedure is applied to the resolution of complex flow problems. With a solution adaptive method based on remeshing, the mesh is recomputed periodically as the solution evolves. Therefore, the speed at which a mesh can be generated is critical to the overall performance. For this reason both the initial and adapted triangular meshes are generated using the advancing front method developed by Peraire et al. [67]. This particular scheme also allows a significant amount of control over local mesh properties with specification of the local length scale, aspect ratio and orientation of triangles. As shown in Reference [67] these mesh parameters provide a means of directionally adapting the mesh. The present unstructured mesh generation scheme follows the work described in References [67] and [48].

4.1 Advancing Front Method

The advancing front mesh generation scheme begins with the specification of the domain boundaries and a background mesh on which nodal values of mesh parameters δ, α and s are prescribed. Referring to Figure 4.1, these mesh parameters define the characteristics of the elements to be generated: δ defines the node spacing; s defines the stretching or aspect ratio; and α defines the direction of stretching. These three mesh parameters are assumed to have piecewise linear spatial distribution over triangular elements of the background mesh. When a new triangle is formed, the mesh parameters are first evaluated through interpolation from the background mesh. Initially, the background mesh is constructed using a

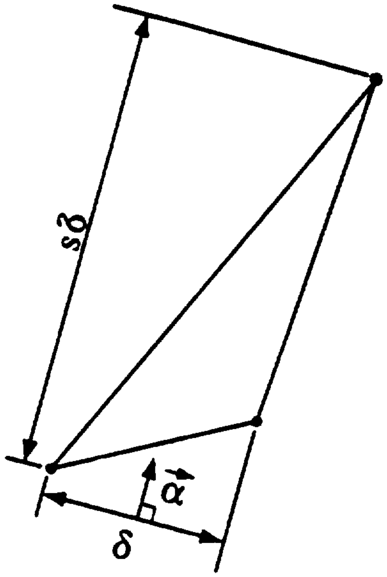


Figure 4.1 Definition of mesh parameters.

very coarse hand triangulation of the domain (e.g. see Figure 4.2). If the domain is being remeshed the last mesh is used.

The first step of the mesh generation process involves generation of mesh points on the boundary of the domain. This step is done in a predictor/corrector fashion with a spline-fitting technique providing the boundary surface location. The two-step process is described below with the example problem shown in Figure 4.3.

Assume background mesh information along the boundary of interest is known (see Figure 4.3). The spline function is given in terms of an arc length tangent to the boundary ranging from 0 to $r_{\max}$. The coordinates (x, y) and the mesh parameters are expressed as function of r. Marching around the boundary, a new list of boundary points with spacing based on the local value of δ is generated iteratively. The iteration process is described as follows.

$$r^n_{i+1} = r_i + \delta^n_{i+1/2}$$

Starting with initial guess $\delta^0_{i+1/2} = \delta_i$, r^0_{i+1} is computed. The local value of δ^0_{i+1} is interpolated from background mesh. Then new value of $\delta^{n+1}_{i+1/2}$ is computed as a

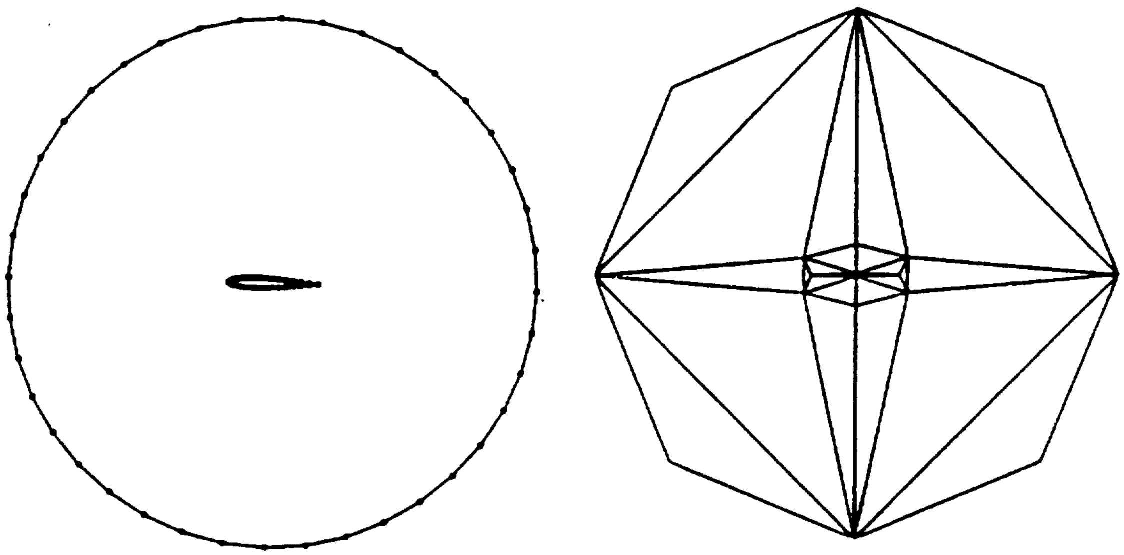


Figure 4.2 Boundary information and background mesh for NACA0012 airfoil.

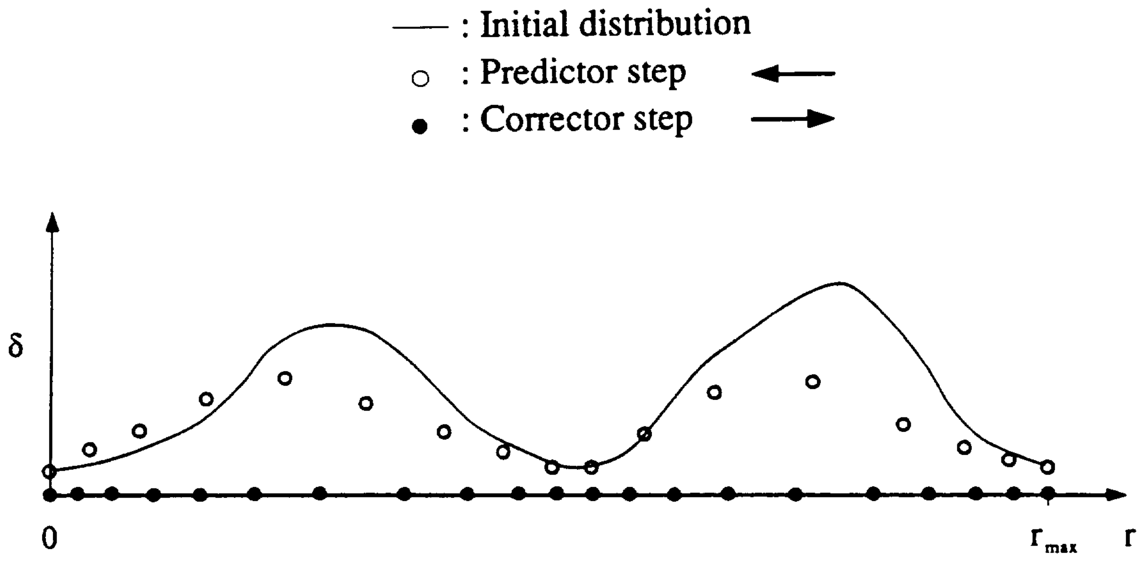


Figure 4.3 Boundary node generation using predictor-corrector process.

mean of δ_i and δ^n_{i+1}

$$\delta^{n+1}_{i+1/2} = 0.5(\delta^n_{i+1} + \delta_i)$$

The above iteration is continued until convergence. In order to generate a smooth distribution the following restriction is imposed:

$$\delta^{n+1}_{i+1} = \min(1.15\delta_i, \delta^{n+1}_{i+1})$$

Then, if a local minimum δ exists within node i and $i+1$, the location r and the value δ replaces the values at point $i+1$.

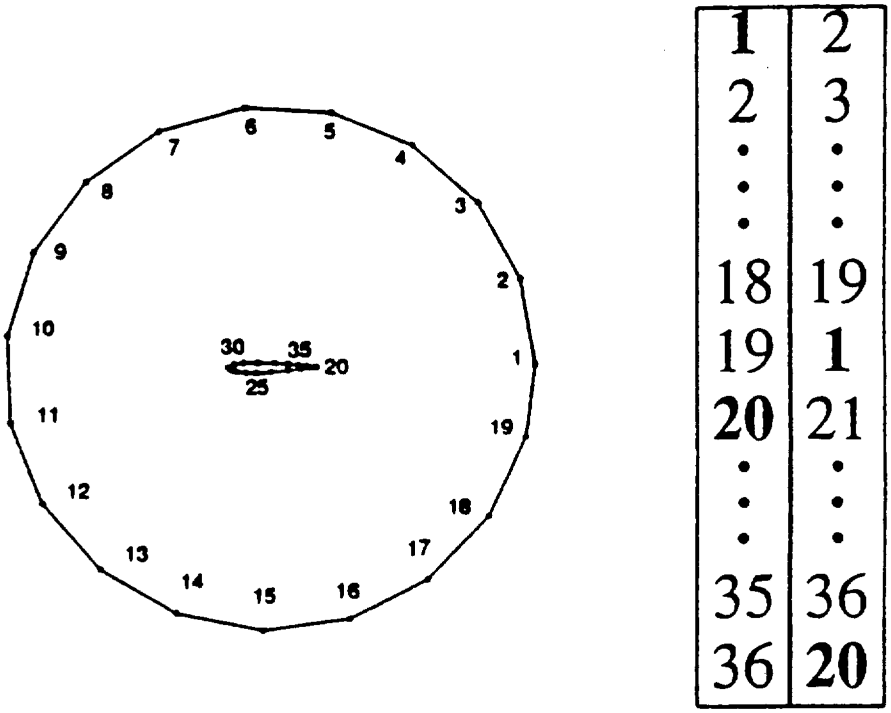
The above procedure produces a smooth mesh spacing distribution if the value of δ is in ascending order corresponding to the marching direction. In order to obtain a smooth variation for a general distribution of δ, the boundary points and mesh parameter distribution are first predicted marching backward around the domain boundary using the above iterative scheme. Then, the boundary points distribution is corrected marching forward around the domain boundary placing points on the boundary using the mesh parameters obtained at the predictor step. There is no need for iteration in the corrector step, because the predictor step provides a good prediction of the boundary mesh space distribution δ.

These boundary points define the initial front list, a set of straight line segments which connect consecutive boundary points as shown in Figure 4.4. Step by step new triangles are then added alone the front and then absorbed into the front. Starting with the shortest segment of the front, a new node is added within the domain at a point determined by the local values of the mesh parameters and specified mesh quality constraints (see [67]).

$$\sigma = \begin{cases} \max(\sigma_i,\ 0.75\sigma_{\text{fr}}) & \text{if } \sigma_i < \sigma_{\text{fr}} \\ \min(\sigma_i,\ 1.3\sigma_{\text{fr}}) & \text{if } \sigma_i > \sigma_{\text{fr}} \end{cases} \tag{4.1}$$

where σ represents the mesh parameters, δ, s, and α, and the subscripts i and fr denote local value and value of the active front, respectively.

Based upon the local values of the stretching parameter, the Cartesian coordinates (x, y) are transformed into elliptic coordinates (x_e, y_e), in which the triangles



1	2
2	3
•	•
•	•
•	•
18	19
19	1
20	21
•	•
•	•
•	•
35	36
36	20

Figure 4.4 Initial front set up.

satisfying the stretching conditions will look "equilateral". This transformation is described as

$$\begin{pmatrix} x_e \\ y_e \end{pmatrix} = \begin{bmatrix} 1/s & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{4.2}$$

where s and α are the local values of mesh stretching and mesh orientation. In elliptic space, a list of candidate nodes for the triangulation is created searching the front nodes which lie inside of a circle formed by the new point and a specified radius. The radius used in the present work is set equal to 1.6δ. The candidate points are then sorted in ascending order according to their distance from the new point. The new point is placed at the first place in the list if the distance from the new point to the end points of the active front are less than 1.6δ. Otherwise, it is placed at the end of the list.

A new triangle is formed using the active front face and the first node from the candidate list which satisfies a criterion that any face of the new triangle can not cross any existing front segment. The front list is then modified to include the faces of the new triangle in the front, while removing the triangle face adjoining the previous front. After the new triangle is created the elliptic coordinates are transformed back to the Cartesian coordinates. This process is repeated until there are no segments in the front list, at which time the domain has been completely filled with triangular elements (e.g., see Figure 4.5).

To improve the quality of the flow solutions computed on the new mesh the following three additional operations are performed in the present implementation. First each node within the domain which is a vertex for less than 5 triangles is removed and the mesh is retriangulated in the region of the removed node. Second, diagonal swapping is performed to remove any obtuse triangle. If the angle between any two sides of a triangle is greater than some specified angle α_{sw} the diagonal of two adjacent triangles are to be swapped. In the present work $\alpha_{sw} = 170\,\text{deg}$ is used. Last, the mesh is smoothed to remove any nonuniformities in the mesh using a Laplacian type operator:

$$\vec{X}_i^{m+1} = \vec{X}_i^m + \frac{\epsilon}{n}\sum_{j=1}^{n}(\vec{X}_j^m - \vec{X}_i^m) \tag{4.3}$$

where $\vec{X}^{m+1}$ is the new location of the point $\vec{X}_i$, n is the number of the neighboring points and ϵ is the smoothing coefficient. In the present work two smoothing passes with $\epsilon = 1$ are used.

4.2 Structured Triangular Mesh Algorithm

The advancing front technique provides a flexible way to generate unstructured triangular meshes for complex geometries. However, Hassan *et al.* [30] have recently noted that the advancing front technique can only produce a maximum allowable mesh stretching of about 10 in order to preserve mesh quality. In a preliminary study of the directional mesh generation for viscous flow problems it has been found that the unstructured mesh generation scheme used here can not produce a good quality mesh when mesh stretching is greater than 20. For turbulent flow calculations highly stretched meshes, where aspect ratios are on the order of 100

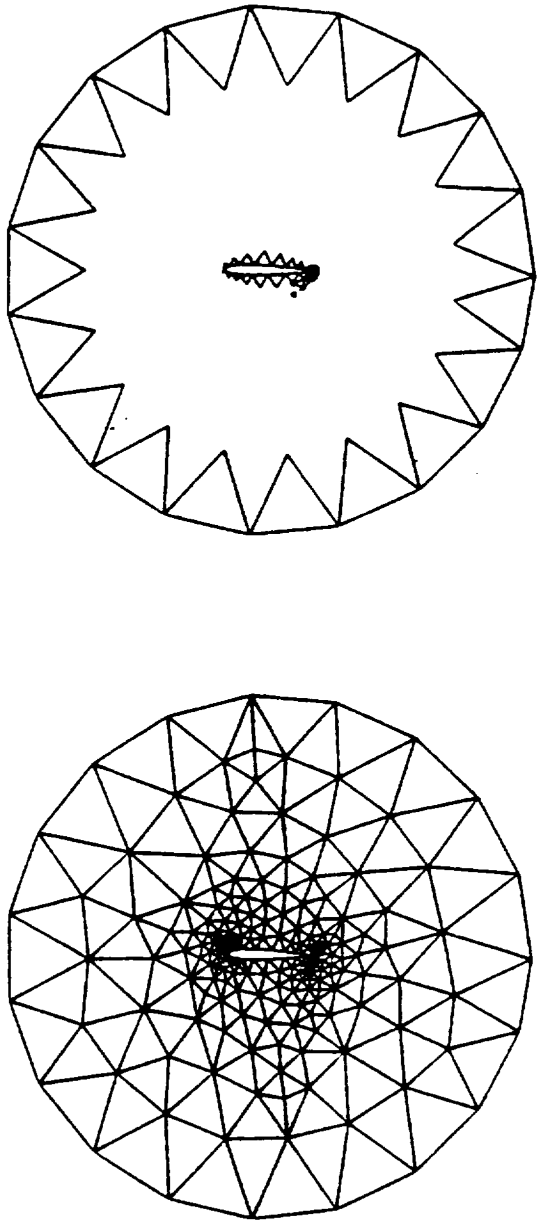


Figure 4.5 Intermediate and final mesh of NACA0012.

to 1,000, are usually employed to resolve the shear layers. It is obvious that a stretching of O(10) is not sufficient.

In the present approach a structured triangular mesh is generated around bodies to achieve high aspect ratio meshes within the boundary layer. Depending on the shape of the trailing edge two types of meshes are used in the present work. A C-type mesh is used if a blade has a wedged or cusped trailing edge and an O-type mesh is employed when the blade has a rounded trailing edge.

4.2.1 O-Mesh Approach

For blades with rounded trailing edges, once the blade boundary nodes have been determined the normal vector at every boundary node is defined. Radial running mesh lines are generated using these normal vectors. A fixed number of mesh points are then placed along each mesh line based on an algebraic stretching algorithm. That is

$$\Delta S_i = S_i - S_{i-1} = \delta_{y^+}\, r_o^{i-1}, \qquad i = 1, N \tag{4.4}$$

where S_i indicates the normal distance from point i to the solid surface with the initial value $S_0 = 0$, and r_o and N denote the algebraic stretching factor and total number of points in the normal direction, respectively. The minimum normal mesh scale, δ_{y^+}, is defined as

$$\delta_{y^+} = \Delta y_{\mathrm{wall}}\, y^+_{\mathrm{pr}} \tag{4.5}$$

In the above expression Δy_{wall} is the physical wall distance corresponding to $y^+ = 1$ and y^+_{pr} represents the allowable y^+ value at the first layer off the wall.

The value of N is specified such that the aspect ratio of all cells in the outer layer are in the range of 10 to 20. The total thickness of the structured mesh at each station can be computed as:

$$\Delta Y_{\max} = \delta_{y^+}\, \frac{r_o^N - 1}{r_o - 1} \tag{4.6}$$

The above mesh is usually referred as an O-type mesh.

4.2.2 C-Mesh Approach

At a wedged or cusped trailing edge the normal vectors on both upper and lower surfaces are different. It is appropriate to use the corresponding normal vector to construct mesh lines on the upper and lower surfaces. This leads to a C-type mesh. The mesh point location is determined using the same procedure as for the O-mesh. In order to reduce the aspect ratio off the trailing edge, this mesh is extended in the near-body wake region. First a mesh line is constructed along the bisector vector of the trailing edge in the downstream direction. Then a number of grid points are placed on this line using the algebraic stretching distribution

$$\Delta S_{x_i} = \Delta X_0 \ \min\{\Delta X_{\max}, \frac{r_x^i - 1}{r_x - 1}\}, \qquad i = 1, M \tag{4.7}$$

where ΔX_0 is the streamwise mesh size at the trailing edge, and $\Delta X_{\max}$ and M are user-specified constants which are in the range of 4–8 and 15–20, respectively. The value of $\Delta X_{\max}$ is specified to limit the growth of the streamwise mesh size at wake regions.

After grid points are placed on the bisector line, mesh lines are generated on the upper and lower sides of the bisector line at each node. The following distribution is employed to smooth the angle between the bisector vector and the upper or lower mesh line:

$$\theta_i^U = \theta_{\mathrm{TE}}^U + \left(\theta_{\mathrm{BI}}^L - \theta_{\mathrm{TE}}^U\right) \frac{i}{M} \tag{4.8}$$

$$\theta_i^L = \theta_{\mathrm{TE}}^L + \left(\theta_{\mathrm{BI}}^L - \theta_{\mathrm{TE}}^L\right) \frac{i}{M} \tag{4.9}$$

In these expressions, the superscripts U and L represent the upper and lower surfaces, respectively, the subscripts TE and BI denote the trailing edge and the bisector line, respectively, the subscript i denotes the $i-th$ mesh line, M is the total number of mesh lines, and θ is the orientation of mesh lines.

Mesh points are then placed on these mesh lines using a distribution defined as:

$$\Delta S_{y_j} = \min\{\Delta Y_{\max}, Y_{0_i}\frac{r_y^j - 1}{r_y - 1}\}, \qquad j = 1, N \tag{4.10}$$

$$Y_{0_i} = \delta_y + r_x^{i-1}, \quad i = 1, M \qquad (4.11)$$

where Y_{0_i} is used to reduce the mesh aspect ratios on the wake regions. The stretching factors, r_x and r_y, are in the range of 1.1–1.35. Notice that the combination of Eq. (4.10) and Eq. (4.11) might produce a different number of mesh points on the adjacent mesh lines. The use of quadrilateral meshes can only connect part of the mesh points. Additional triangles are therefore placed to connect the remaining mesh points.

After the quadrilateral meshes are constructed a structured triangular mesh is generated by diagonally dividing each quadrilateral mesh cell into two triangles (e.g., see Figure 4.6 and Figure 4.7). The initial front list is then updated to include the faces of the outer layer of structured triangles in the front, while removing the previous fronts which connect consecutive body boundary points. With the new front list, the advancing front method is used to mesh the remaining domain. Because the algebraic stretching algorithms, Eqs. (4.4), (4.10) and (4.11), relieve the stretching ratio on the outer layer of the structured quadrilateral meshes, the advancing front method can be employed without difficulty.

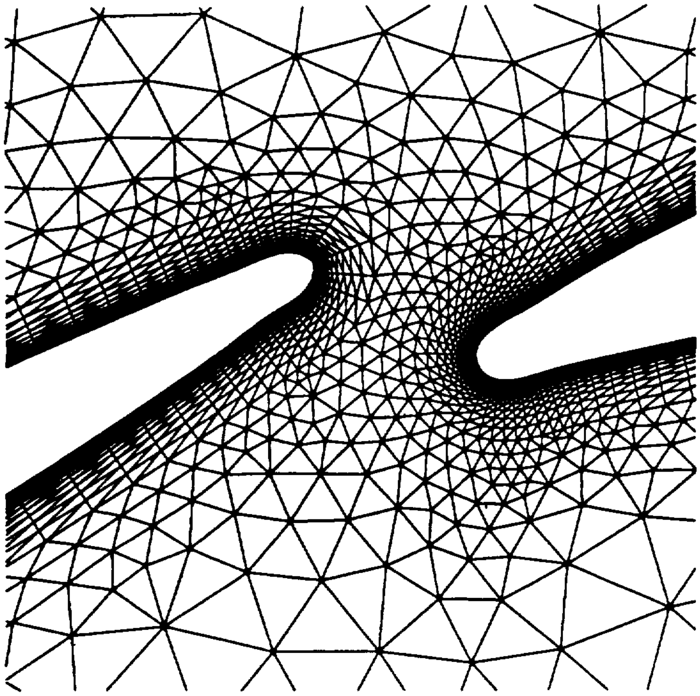


Figure 4.6 O-type structured triangular mesh.

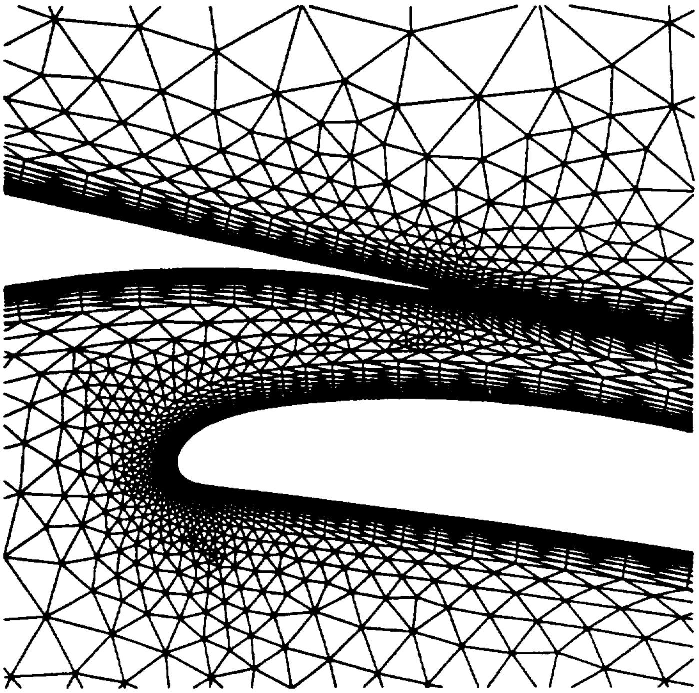


Figure 4.7 C-type structured triangular mesh.

5. SOLUTION ADAPTATION ALGORITHM

This chapter presents a solution adaptive algorithm for both inviscid and turbulent flow problems. Key elements in this algorithm are the definition of mesh adaptation parameters, the convergence criteria for the flow solver and the convergence criteria for the overall adaptation loop. The mesh adaptation parameters are defined in terms of geometric and flow field information. For turbulent flow problems additional length scale information is used to define the local structured meshes.

5.1 Adaptive Remeshing Algorithm

In the present work, the flow solver and the advancing front mesh generator are coupled together using a solution adaptation scheme which periodically remeshes the solution domain to resolve the flow structure as the solution evolves. The basic remeshing algorithm is shown schematically in Figure 5.1. The initialization step generates an extremely coarse "hand-triangulation" of the solution domain to be used as the initial background mesh. Mesh adaptation parameters are then computed at nodes on this background mesh. Since no flow field information is available at this point, the mesh adaptation parameters are set to constants or are computed based on geometric information. With this information, a coarse initial mesh is generated and the flow solver is called.

The solution is marched in time on the initial mesh until a prescribed steady-state convergence criteria is met. Based on this initial solution, the solution adaptation parameters are computed using flow field and geometric information on the initial mesh. The mesh is then regenerated and a new solution is computed. The

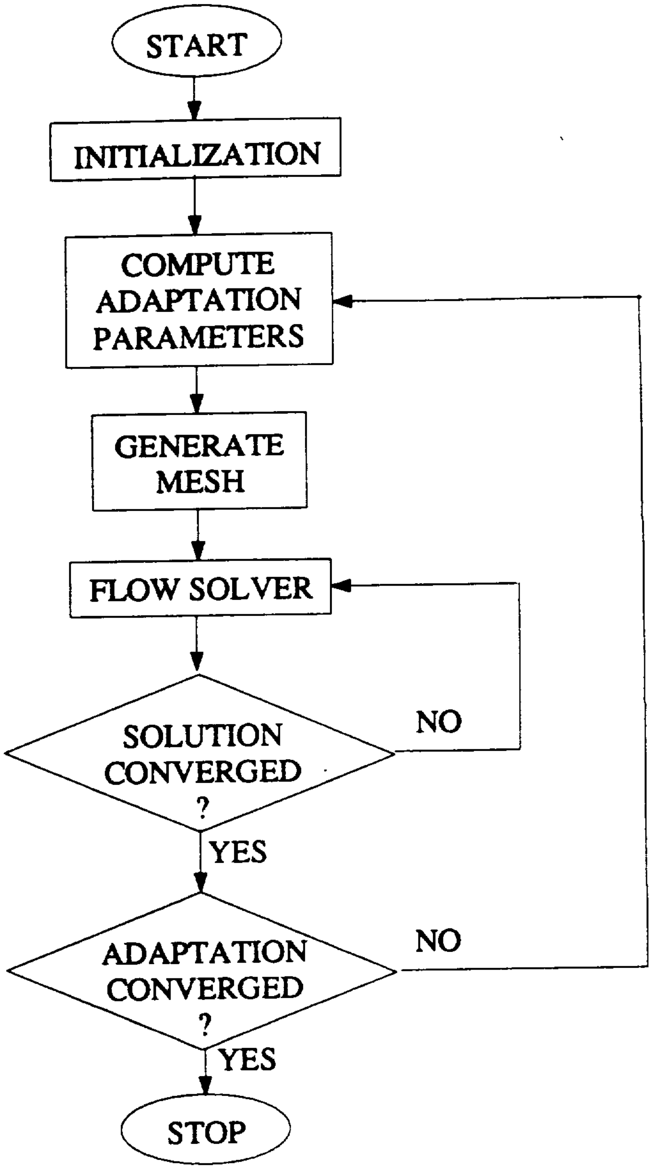


Figure 5.1 Flow chart of adaptive remeshing algorithm.

current solution is then compared to the solution on the previous mesh to check for overall convergence of the adaptation cycle. The adaptation cycle (outer loop in Figure 5.1) is repeated until the difference between the current and previous solutions drops below a prescribed convergence criterion.

5.2 Mesh Adaptation Parameters

In the advancing front method, the mesh adaptation parameters required for generation of a new mesh are the mesh length scale, δ, the stretching parameter, s, and the stretching direction denoted by α. These mesh adaptation parameters are computed using one or more chosen refinement parameters determined by the current solution and the geometry of the problem. The mesh length scale parameter, δ, is defined in terms of two refinement parameters, one based on surface geometry information and the second based on flow field information. Combined, these two refinement parameters define the variation of δ throughout the solution domain. The mesh stretching parameter, s, and the stretching direction, α, are defined in terms of additional flow field information such that the mesh orientation and stretching align with detected flow features.

5.2.1 Geometric Refinement Parameter

Along solid boundaries of the solution domain a refinement parameter based on the local surface curvature is used. This parameter has been found to be very important in transonic fan and compressor applications, where an accurate representation of the blade geometry is critical to accurately setting up the flow structure within the solution domain. This in turn improves the overall convergence of the solution adaptive method. The geometric length scale parameter, δ_s, is defined in terms of the local radius of curvature of the surface, R_i, as

$$(\delta_s)_i = \frac{2\pi R_i}{(N_s - 1)}. \tag{5.1}$$

where N_s is the number of points equally spaced on the arc, and $(\delta_s)_i$ is the distance of the arc length between any two neighboring points on the circumference of a circle with radius R_i. For the solutions presented in the following sections, N_s is chosen in the range of 10 to 40 depending on the resolution desired.

5.2.2 Flow Field Refinement Parameter

The goal of a solution adaptive scheme is to achieve an optimal mesh on which the numerical error is uniformly distributed. Because the flow solution is not known in advance, a quantitative estimate of numerical errors must be developed. These errors can be estimated in many different ways. A review of various approaches is given by Morgan and Peraire [63]. Morgan and Peraire note that solution refinement parameters based on either first- or second-derivatives work well and are very economical. In the present work two sets of solution refinement parameters have been investigated.

5.2.2.1 Second-Derivative Refinement Parameters

In Reference [63] Morgan and Peraire introduce refinement parameters based on a local interpolation error analysis for a piecewise linear discrete approximation. In one dimension, the interpolation error is estimated by

$$\delta^2|\partial^2\phi/\partial x^2|$$

where δ is the mesh spacing and ϕ is some flow variable. An optimal mesh is determined by the requirement that the distribution of local numerical errors is uniform. This gives

$$\delta^2|\frac{\partial^2\phi}{\partial x^2}| = \text{constant} \tag{5.2}$$

For two-dimensional problems, the discretization errors are represented by the tensor of second derivatives which is given by

$$\begin{bmatrix} \partial^2\phi/\partial x^2 & \partial^2\phi/\partial x\partial y \\ \partial^2\phi/\partial x\partial y & \partial^2\phi/\partial y^2 \end{bmatrix}$$

This is a real symmetric tensor. For a real symmetric tensor, it can be shown that the eigenvalues are all real and the corresponding eigenvectors are orthogonal [4].

The coordinates parallel to the eigenvectors are referred to as the principal axes. Morgan and Peraire invoke the orthogonality of the principal axes and introduce refinement parameters based on the second derivatives computed along the local principal directions, $\vec{X}_1$ and $\vec{X}_2$. This is equivalent to computing the eigenvalues of the local second derivative tensor.

$$\lambda_1 = \frac{\partial^2\phi}{\partial X_1^2}, \qquad \lambda_2 = \frac{\partial^2\phi}{\partial X_2^2}, \qquad |\lambda_1| > |\lambda_2| \tag{5.3}$$

To achieve an optimal two-dimensional mesh, the requirement of uniform distribution of local numerical errors is applied to all directions. This leads to

$$\delta_1^2|\lambda_1| = \delta_2^2|\lambda_2| = \delta_{\min}^2\lambda_{\max} = \text{constant} \tag{5.4}$$

where δ_1 and δ_2 are the mesh spacings in the local principal directions $\vec{X}_1$ and $\vec{X}_2$, respectively, $\delta_{\min}$ is an user-specified minimum mesh spacing, and $\lambda_{\max}$ is the maximum value of $|\lambda_1|$ over the whole domain. The aspect ratio of mesh cells is determined as the ratio of the two mesh spacings:

$$s = \delta_2/\delta_1 = \sqrt{|\lambda_1|/|\lambda_2|} \tag{5.5}$$

The mesh orientation is simply defined as the angle between the principal axes and the Cartesian coordinates. That is

$$\alpha = \frac{1}{2}\tan^{-1}\left(2\frac{\partial^2\phi}{\partial x\partial y}/\left(\frac{\partial^2\phi}{\partial x^2} - \frac{\partial^2\phi}{\partial y^2}\right)\right) \tag{5.6}$$

The advantage of this approach is that the discretization error is evaluated on a theoretical basis and all three mesh parameters are uniquely defined. However, the second derivatives tend to produce large errors when there is a perturbation in the solution domain. For example, for transonic flow problems, a standard flow solver smears shock waves over several points and also produces Gibbs phenomena near shock wave locations. Computing second derivatives based on this solution gives the largest value of λ_1 on either side of the shock wave and a small value at the center of the shock wave. When a new mesh is generated, small mesh cells are constructed on either side of the shock wave and bigger mesh cells are constructed at the center of the shock wave. Thus, the new mesh fails to accurately resolve the shock wave. This problem can be alleviated by using either a high resolution flow solver or a refinement parameter based on lower-order derivatives. In the present work, the latter is used.

5.2.2.2 First-Derivative Refinement Parameters

For the present adaptive solution method, a refinement parameter based on the gradient of a certain flow property is developed. The refinement parameter, the absolute value of the gradient of a specified flow property, ϕ, is computed at every node.

$$g_i = \mid \nabla\phi \mid = \sqrt{(\frac{\partial\phi}{\partial x})^2 + (\frac{\partial\phi}{\partial y})^2} \tag{5.7}$$

A quantitative local numerical error is estimated by

$$\delta \mid \nabla \phi \mid = \delta\, g_i$$

The variation of δ within the solution domain is defined such that the above quantity is constant over the whole domain:

$$\delta \mid \nabla \phi \mid = \delta\, g_i = \delta_{\mathbf{min}} g_{\mathbf{max}} = \text{constant} \tag{5.8}$$

where $\delta_{\mathbf{min}}$ is a user-specified minimum mesh spacing, and $g_{\mathbf{max}}$ is the maximum value of $\mid \nabla \phi \mid$ over the whole domain. The mesh orientation is computed based on the gradient direction.

$$\alpha = \tan^{-1}\left(\frac{\partial\phi}{\partial y} / \frac{\partial\phi}{\partial x}\right) \tag{5.9}$$

$\nabla\phi$ only is used to define two mesh parameters. Although second-order derivative terms can be used to estimate the mesh stretching parameter, s, this leads to the original problem addressed in the previous section. For complicated flow structures the combined effect of the mesh stretching parameter and stretching direction can produce very distorted meshes. To eliminate the potential for such highly distorted meshes, only the mesh length scale, δ, is used. The stretching parameter, s, is set to one and the stretching direction α is taken as the direction perpendicular to the generating segment. Specifying s and α in this fashion results in grids composed of roughly equilateral triangles with varying length scales.

The computed g_i is smoothed to remove numerical noise and spread the high gradient regions. Even with this smoothing, it is necessary to limit the allowable range of g_i. A cutoff level for g_i is determined as follows. First a referenced value of the gradient of ϕ is defined by the difference between the maximum and minimum values of ϕ divided by the minimum length scale of the previous mesh. That is

$$g_{\text{ref}} = \frac{\phi_{\text{max}} - \phi_{\text{min}}}{\delta_{\text{min}}^{\text{last}}} \tag{5.10}$$

where $\delta_{\text{min}}^{\text{last}}$ is the minimum length scale of the previous mesh. The cutoff value for g is then set at

$$g_{\text{cut}} = \begin{cases} g_{\text{ref}}/C_1, & \text{if } g_{\text{ref}}/g_{\text{min}} > C_2 \\ g_{\text{ref}}, & \text{otherwise} \end{cases} \tag{5.11}$$

where C_1 and C_2 are constant, typically in the range of 10–30 and 50–100, respectively. Finally, the limited value of the gradient at each point is given by

$$\bar{g}_i = \frac{g_{\text{max}} g_{\text{cut}} g_i}{g_{\text{max}} g_{\text{cut}} + (g_{\text{max}} - g_{\text{cut}}) g_i} \tag{5.12}$$

This function has a maximum value of g_{cut} and limits g in a smooth fashion. The mesh length scale is now determined by

$$\delta_i = \min(\delta_{\text{max}}, \frac{g_{\text{cut}}}{\bar{g}_i} \delta_{\text{min}}) \tag{5.13}$$

The minimum and maximum values of δ must be specified for each mesh. The maximum length scale, δ_{max}, defines the size of the largest triangles in the domain. Within the solution domain, the minimum length scale, δ_{min}, is sequentially decreased from δ_{max} on the initial mesh down to a value which leads to a solution satisfying the adaptive convergence criteria. The goal is to refine subsequent meshes as quickly as possible, without introducing wasted points as the solution evolves. Currently, this sequence is set in terms of the ratio of δ_{min} of the last mesh divided by δ_{min} of the new mesh.

5.2.2.3 Flow Feature Indicator

Without a proper choice of the flow feature indicator, neither the second-order nor the first-order error estimator can accurately and efficiently resolve the flow. The location of flow structures is not known in advance. An indicator related directly to an earlier solution must be developed. The performance of various indicators for inviscid transonic flows has been investigated by Dannenhoffer [21]. With the inclusion of viscous flow features in his results, the performance of different indicators for viscous flows is summarized in Table 5.1. Although a combination of different indicators have been proposed for the resolution of viscous transonic flow features [40], in the present work, an indicator based on the local flow speed has been found to be sufficient.

5.2.2.4 Refinement Parameters for Local Structured Mesh

The mesh spacing, δ, computed from Eq. (5.4) or Eq. (5.13) depends on a single mesh scale $\delta_{\min}$. For inviscid flow calculations, the single length scale criterion is sufficient for resolving all flow features. Viscous flows generally involve two or more different length scales. For example, for turbulent transonic airfoil calculations the length scale required inside the boundary layer is usually on the order of $\mathcal{O}(Re_L^{-1})$, while outside the boundary layer the length scale is on the order of the chord. Although one may argue that since the thickness of shock waves is on the order of a molecular free path, it should be reasonable to use a viscous length scale in the region of the shock wave. However, such a length scale increases the computing expense and is not required in practice.

For the turbulent flow calculations presented here, the mesh scales for inviscid and viscous regions are considered separately. In the inviscid flow region the mesh spacing is computed using Eq. (5.13). In the boundary layer the mesh scale in the normal direction to the wall is based on the wall distance or wall units, y^+, while the mesh spacing in the tangential direction to the wall is estimated using the inviscid mesh scale. These two mesh scales usually results in a highly stretched mesh. In

Table 5.1 Expected performance of different indicators for viscous transonic flow problems.

Feature Indicator	Feature Type				
	Shock Wave	Expansion Wave	Stagnation Zone	Slip Line	Viscous Shear Layer
ρ	○	○	⊙	○	○
$\nabla\rho$	●	●	⊙	⊙	⊙
$\nabla^2\rho$	●	⊙	⊙	●	⊙
p	○	○	●	○	○
∇p	●	●	●	○	○
$\nabla^2 p$	●	⊙	●	●	○
P_t	○	○	○	○	●
∇P_t	●	○	○	●	●
$\nabla^2 P_t$	●	○	○	●	●
q	○	○	○	○	⊙
∇q	●	●	●	●	●
$\nabla^2 q$	●	⊙	●	●	⊙

Note: ● ⇒ Flow feature is well detected.

⊙ ⇒ Flow feature is somewhat detected.

○ ⇒ Flow feature is not detected.

C-2

order to achieve such high aspect ratio mesh cells, a structured triangular mesh is employed in the present work. For the structured triangular mesh algorithm, Eq. (4.4), the streamwise and the normal mesh scales are specified along the wall. Then, a number of layers of structured mesh are generated from the wall boundary to the interior. The use of a structured mesh simplifies the multiple length scale problem in that the streamwise mesh scale specified on the wall surface is sufficient to define the streamwise mesh scale in the boundary layer.

For the initial mesh the wall distance is estimated using the local skin friction for a turbulent flat plate flow [76].

$$\frac{\tilde{\tau}_{\text{wall}}}{\langle\rho\rangle U_\infty} = 0.0296\, Re_x^{-1/5} \tag{5.14}$$

The wall distance corresponding to $y^+ = 1$ at the end of the plate is then obtained as

$$\Delta y_{\text{wall}} = 5.81\, Re_L^{-9/10} \tag{5.15}$$

Specifying the wall units y^+_{pr} gives the wall distance of the first structured mesh line off the wall:

$$\delta_{y^+} = \Delta y_{\text{wall}}\, y^+_{\text{pr}} \tag{5.16}$$

where the value of δ_{y^+} is specified in the range of 2.5–3.5 for the Baldwin-Barth one-equation model. The streamwise mesh scale is taken directly from the initial mesh scale distribution on the background mesh. From these two mesh scales, the structured mesh is then constructed.

As the flow solution evolves, the local wall distance is recomputed at each boundary node using Eq. (2.14) and the minimum value of the local wall distance is used to estimated the new δ_{y^+}. The local streamwise mesh scale is then estimated from the inviscid flow information. In the inviscid flow region the Bernoulli equation gives

$$\frac{\partial p}{\partial s} + \rho q \frac{\partial q}{\partial s} = 0 \tag{5.17}$$

where q is the flow speed, s is in the streamwise direction, and p is the static pressure. After extrapolating the inviscid flow speed from the boundary layer edge to the wall, the streamwise gradient of the flow speed at the wall is obtained as

$$g_s = |\frac{\partial q}{\partial s}| = |\frac{1}{\rho q}\frac{\partial p}{\partial s}| \tag{5.18}$$

where s is now replaced with the arc length scale along the wall. Employing Eq. (5.12) to smooth the distribution of $|\partial q/\partial s|$ gives the streamwise mesh spacing along the wall.

$$\delta_{s_i} = \min(\delta_{\mathrm{max}}, \frac{g_{\mathrm{cut}}}{\bar{g}_{s_i}}\delta_{\mathrm{min}}) \tag{5.19}$$

In the present work two additional mesh scales are used to define the surface mesh scale:

$$\delta_w = \min\{\delta_c,\ \delta_{s_i},\ (\delta_{y^+} A_w),\ (\delta_{y^+} r_0^{N-1} A_e)\} \tag{5.20}$$

where A_w is the mesh aspect ratio along the wall and A_e is the mesh aspect ratio along the outer edge of local structured meshes. The value of A_w is in the range of 300–3,000 and A_e is in the range of 10–20. From the mesh scales δ_{y^+} and δ_w, the local structured mesh is regenerated.

5.3 Convergence Criteria

To complete the description of the present adaptive remeshing scheme the convergence criteria for the solver and adaptive remeshing cycle must be defined. Convergence for the flow solver is based on three properties of the solution: the average of the absolute value of the local change in ρV_m divided by the local time step, which is a measure of the convergence of the solution to steady-state; and two global norms defined by the difference between the highest and lowest values of the normal force coefficient, C_{f_n}, and the tangential force coefficient, C_{f_t}, over the last 50 iterations divided by reference quantities, K_1 and K_2, respectively. That is

$$\Delta C_{f_n} = [(C_{f_n})_{\mathrm{max}} - (C_{f_n})_{\mathrm{min}}]/K_1 \tag{5.21}$$

and

$$\Delta C_{f_t} = [(C_{f_t})_{\mathrm{max}} - (C_{f_t})_{\mathrm{min}}]/K_2 \tag{5.22}$$

with the reference quantities K_1 and K_2 defined as

$$K_1 = \max(1, (C_{f_n})_{\mathrm{max}}), \quad \text{and} \quad K_2 = \max(1, (C_{f_t})_{\mathrm{max}}) \tag{5.23}$$

The first quantity, $\delta(\rho V_m)/\delta t$ is inversely proportional to the local time step. As the mesh is refined δt decreases which makes this norm more restrictive as the mesh is adapted. The reference quantities K_1 and K_2 are used to normalize the global norms. Since the flow properties are normalized with freestream values in the present calculations, it is appropriate to normalize the integral quantities with a value of 1 if the integral quantities are much less than 1. The use of windowing of integral quantities to define convergence of the force coefficients was developed by Dannenhoffer [21] for an unstructured quadrilateral based adaptive mesh scheme. When any one of these three coefficients drops below the user defined constant, ϵ_1, the solution on the current mesh is converged.

Convergence for the adaptive solution cycle is based on the change in force coefficients from one mesh solution to the next. When the relative change in C_{f_n} is less than ϵ_2 and the relative change in C_{f_t} is less than ϵ_3 the adaptive mesh cycle is converged. The relative change is defined as

$$\Delta C_f = |(C_f)_{\mathrm{last}} - (C_f)_{\mathrm{new}}| / \max\left(1, \max(|(C_f)_{\mathrm{last}}|, |(C_f)_{\mathrm{new}}|)\right) \tag{5.24}$$

The values of ϵ_1, ϵ_2, and ϵ_3 are user specified constants. Specifying smaller values of ϵ_2 and ϵ_3 will increase the number of mesh adaptation cycles. This in turn leads to a smaller mesh scale and improves the accuracy of the solution. Therefore, by choosing these constants a user can specify the accuracy of computed solution. Note the value of ϵ_1 should be at least one-order less than the values of ϵ_2 and ϵ_3. In the present work, ϵ_1 is set to 0.0001 for both inviscid and turbulent flow problems. For inviscid flow problems, both ϵ_2 and ϵ_3 are set to 0.002. For turbulent flow problems, these values are raised to 0.005 to reduce the number of mesh adaptations.

6. INVISCID FLOW RESULTS

The present adaptive solution scheme has been formulated to solve both two-dimensional planar flows and quasi-three-dimensional axial-, radial- and mixed-type flows. In the development of this approach a range of inviscid problems has been solved. To validate the present approach the test cases presented here include a two-dimensional multi-element airfoil, the Sanz supercritical compressor and subcritical turbine cascades, and the Denton supersonic staggered wedge cascade. Each of these test cases is two dimensional and has a known analytic solution. These test cases cover a wide range of Mach numbers and each has different flow structures which must be resolved for an accurate solution. To demonstrate the adaptive solution scheme in practical turbomachinery applications, the quasi-three-dimensional analysis has been used to analyze the NASA Rotor 67 low-aspect-ratio transonic fan and the Allison tandem blade cascade.

6.1 Multi-Element Airfoil Case

The first case is a model three-element airfoil operating in a high lift configuration. This case illustrates the ease with which flow problems in arbitrary multiply-connected regions can be solved. Reference [80] provides the airfoil geometry and an analytic incompressible potential flow solution for an angle of attack of 20°. In order to make a direct comparison with the analytic solution, this test case should be run with a very low freestream Mach number. Since the stability criteria for the present explicit scheme is inversely proportional to the speed of sound, which will be very large, low Mach number flows require very small time steps. Further, very low Mach number flows imply very small convection speeds. Both these conditions result in a very slow convergence rate to the steady state solution and in turn large computation time. As a compromise between the computation time and the compressibility effects, this test case was run with freestream Mach number of 0.125. The flow conditions and airfoil characteristics are summarized in Table 6.1. The far field and near field meshes are shown in Figures 6.1 and 6.2, respectively. The far field boundary is set at a radius of 5 chords from the airfoil. At this boundary the vortex far field boundary condition [83] is used.

The initial mesh is refined near the airfoil through the specification of the length scale parameter along the surface ($N_s = 40$). In the far field the mesh is uniform with a length scale of $\delta_{\max}$= 0.5 chord. Within the solution domain the minimum length scale $\delta_{\min}$ is sequentially decreased, from $\delta_{\max}$ on the initial mesh, down to a value which leads to a solution satisfying the adaptive convergence criteria. The refinement sequence is set in terms of the ratio of $\delta_{\min}$ of the last mesh divided by $\delta_{\min}$ of the new mesh. For the airfoil case presented here this ratio equals 32 for the first adaptation, 4 for the second and the third adaptations and 2 for the last refinement. The sequence of solution adapted meshes are shown in Figures 6.2 to 6.6. Mesh statistics and force coefficient information for each mesh are summarized in Table 6.2.

Mach number contours for the final solution are shown in Figure 6.7. Note that the maximum surface Mach number on the leading edge slat is about 0.6, which is well out of the range of what can be considered incompressible. Figure 6.8 compares the converged surface pressure coefficient distribution over each element

Table 6.1 Flow conditions and airfoil characteristics for the three-element airfoil.

M_∞	α	δ_{slat}	δ_{flap}	Cl_{exact}
0.125	20.0°	45.0°	15.0°	5.136
Note: Incompressible analytic solution is available in Reference [80].				

with the incompressible analytic solution. Although the flow structure of this test case involves high gradients in flow properties, they are only concentrated in small regions where the surface curvature is high. Therefore, plots of intermediate mesh solutions are almost exactly the same. Most of the error in the initial solution is concentrated in the region of the suction peak on the slat.

To illustrate the importance of compressibility in this case, a compressibility correction to the analytic solution using the Karman-Tsien rule is considered in Figure 6.9. The compressible presure coefficient using the Karman-Tsien rule is given as:

$$C_p = \frac{C_{p,0}}{\sqrt{1 - M_\infty^2} + \left[M_\infty^2 / \left(1 + \sqrt{1 - M_\infty^2}\right)\right] C_{p,0}/2} \qquad (6.1)$$

where $C_{p,0}$ is the incompressible pressure coefficient, M_∞ is the freestream Mach number, and C_p is the corrected pressure coefficient. Since the high Mach number flows are concentrated in the suction surface of the slat portion and the leading edge suction surface of the airfoil, the corrected surface pressure coefficients are lower than the incompressible values in these regions. Otherwise, the corrected surface pressure coefficients are essentially the same as the incompressible values. Figure 6.9 shows a slight difference between the computed pressure coefficient and the corrected pressure coefficient in the high Mach number flow regions. This is due to the extremely high flow gradients in these regions. These regions can be accurately resolved with further mesh adaptation.

Figures 6.10, 6.11, and 6.12 show the surface total pressure errors for the initial, 3rd adapted, and final meshes, respectively. The total pressure error is defined as the difference between local total pressure and the upstream value. That is,

$$P_{T_{error}} = 1 - \frac{P_{T_i}}{P_{T_0}}$$

Since this flow is isentropic the total pressure error should be zero. These plots clearly show a reduction in error as the mesh is refined in the slat region. There is a 40 % reduction in the peak total pressure error in the final mesh solution relative to the 3rd adapted mesh solution. The final mesh length scale in the slat leading edge region is about 60 % of the 3rd adapted mesh length scale in the same region. This indicates that the present scheme is first-order.

The computed lift coefficient is 3 % less than the Karman-Tsien corrected lift coefficient which is about 5.279. It is believed that part of this error is due to the location of the far field boundary. Based on the results of Usab [83] and Dannenhoffer [21], the lift coefficient increases about 1.5 to 2.0 % as the far field boundary is moved from 5 chord to 50 chord. It should also be noted that for low Mach number flows, the total pressure error has a significant effect on the pressure distribution. Even though the total pressure error in the final mesh is very small, if all the total pressure error is assumed to reduce the surface static pressure this would lead to a 5 % error in the predicted lift coefficient. Although a direct relation between the lift coefficient and total pressure error is not known, it is reasonable to expect a 1 to 2 % reduction in the lift coefficient due to this total pressure error.

Finally, the convergence histories for the lift coefficient and average residual, $|\delta(\rho u)/\delta t|$, are shown in Figures 6.13 and 6.14, respectively. Note the reduction in the convergence rate of the flow solver on the latter meshes. This is due to the decreasing mesh length scale in the fine mesh region where the explicit time step is very small.

Table 6.2 Three-element airfoil: $M_\infty = 0.125$, $\alpha = 20°$, and $(C_l)_{\text{incomp}} = 5.136$.

MESH	0	1	2	3	4
nodes	2,034	2,092	2,911	4,643	8,506
elements	3,769	3,888	5,517	8,927	16,567
C_l	4.9892	4.9847	5.0630	5.1061	5.1165
C_d	0.0570	0.0444	0.0286	0.0201	0.0145
δ_{max}/ chord	0.5	0.5	0.5	0.5	0.5
$\delta_{\text{max}}/\delta_{\text{min}}$	1	32	128	512	1,024
Total CPU: 1,307 sec in Cray-YMP.					

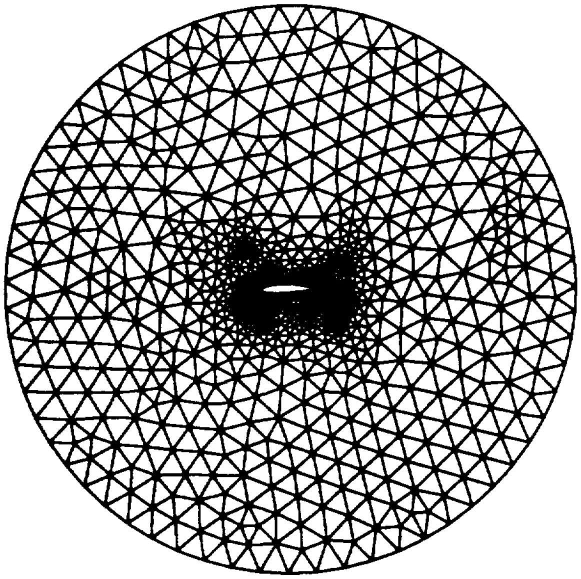


Figure 6.1 Multi-element airfoil initial mesh: far field view.

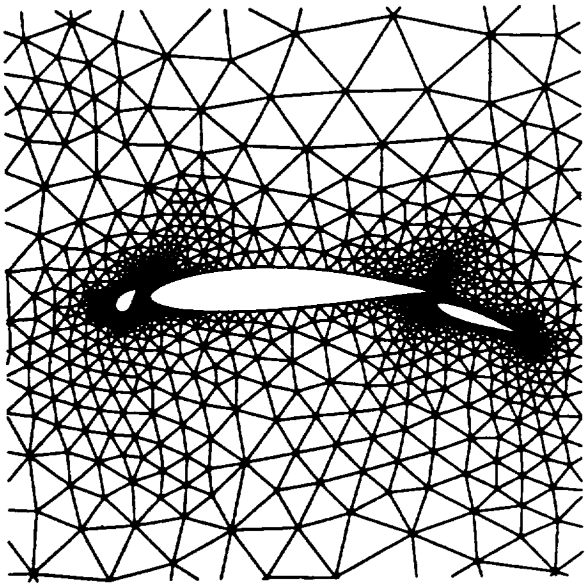


Figure 6.2 Multi-element airfoil initial mesh: 2,034 nodes and 3,769 elements.

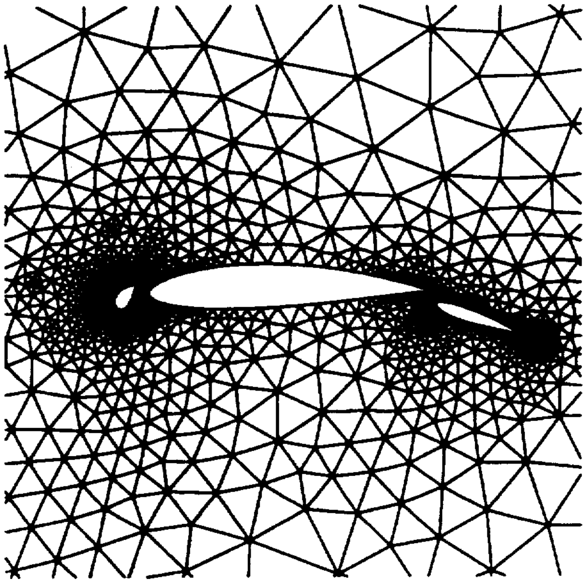


Figure 6.3 Multi-element airfoil 1st adapted mesh: 2,092 nodes and 3,888 elements.

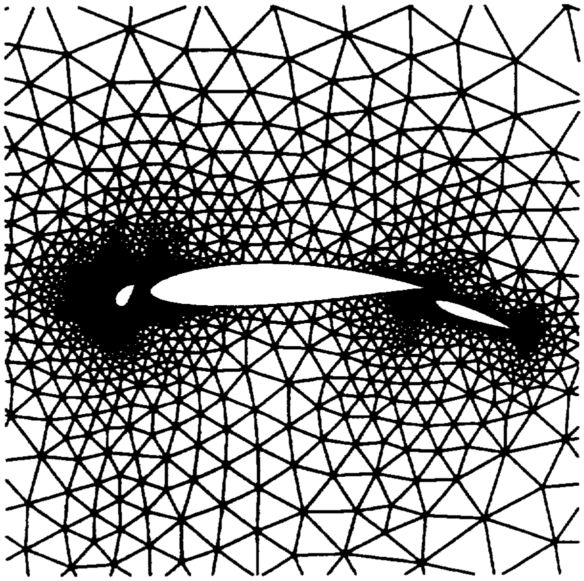


Figure 6.4 Multi-element airfoil 2nd adapted mesh: 2,911 nodes and 5,517 elements.

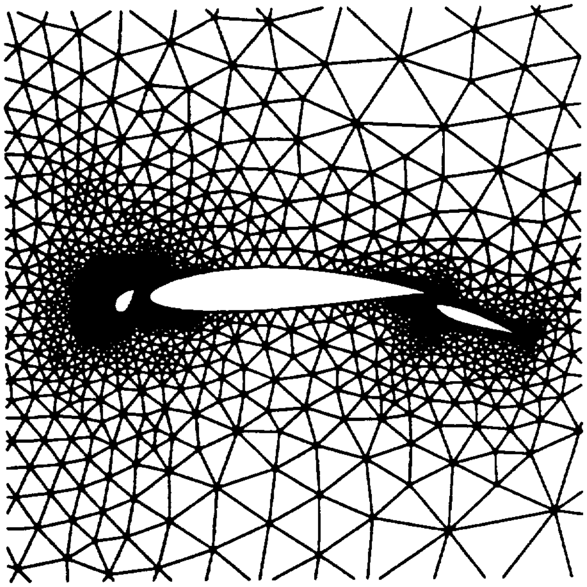


Figure 6.5 Multi-element airfoil 3rd adapted mesh: 4,643 nodes and 8,927 elements.

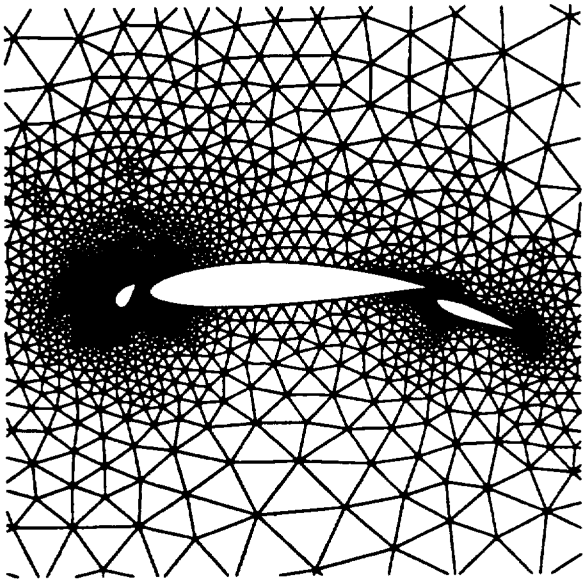


Figure 6.6 Multi-element airfoil 4th (final) adapted mesh: 8,506 nodes and 16,567 elements.

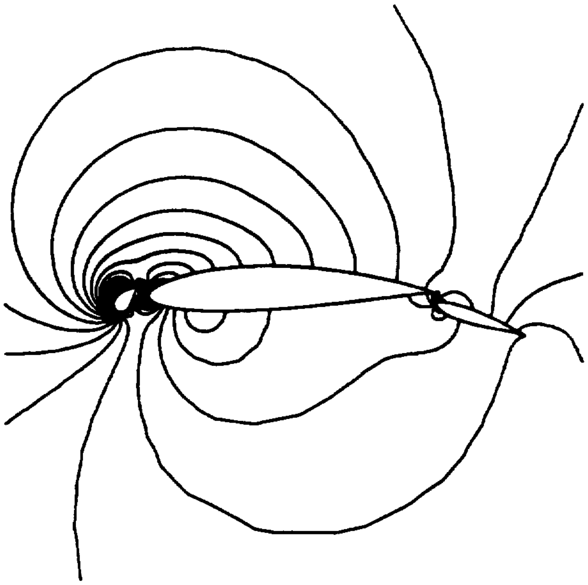


Figure 6.7 Multi-element airfoil Mach number contours on the final mesh: cmin = 0.0, cmax = 0.60, and inc = 0.02.

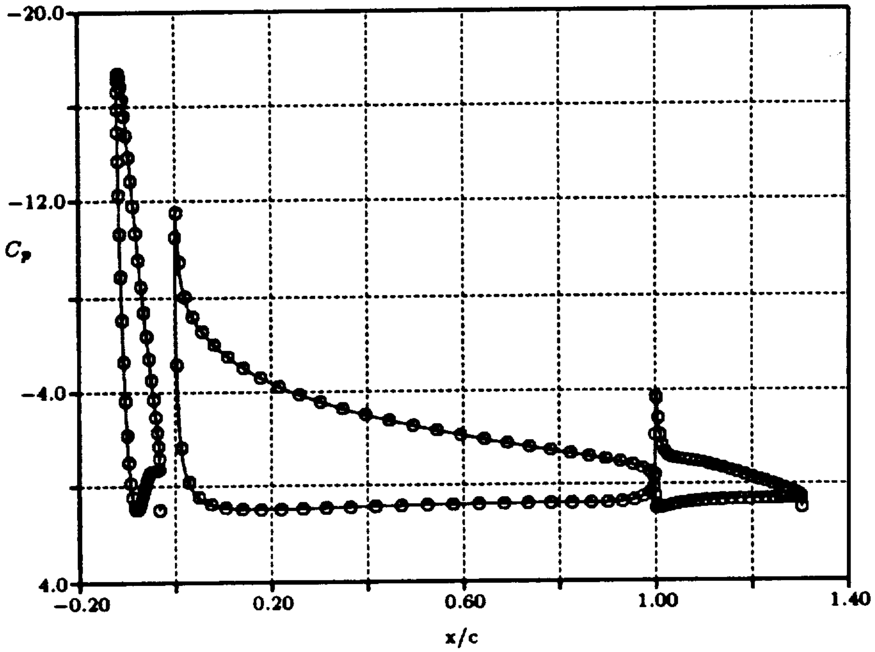


Figure 6.8 Multi-element airfoil surface pressure coefficient for the final mesh: solid line - numerical solution and symbol - analytic solution.

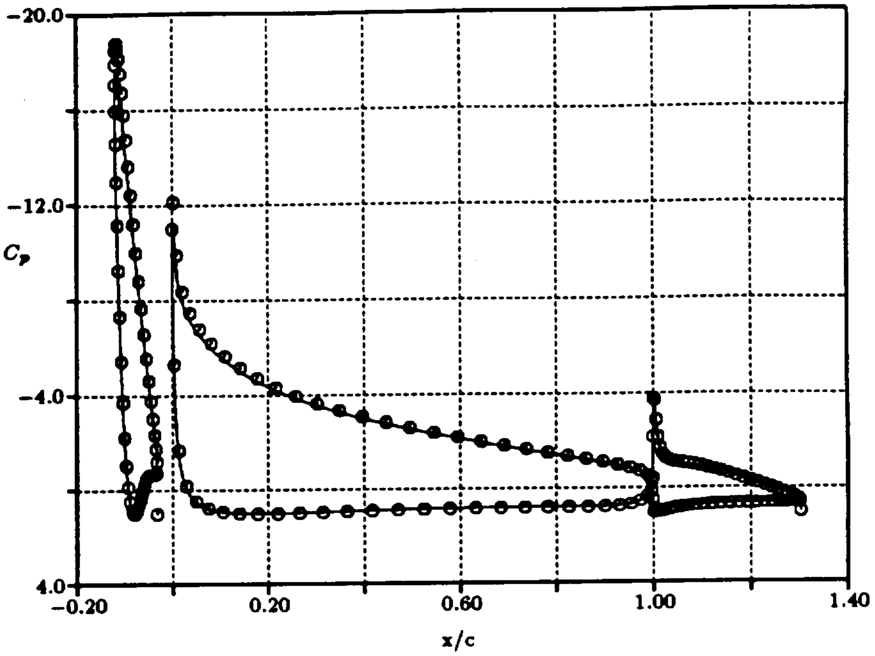


Figure 6.9 Multi-element airfoil surface pressure coefficient for the final mesh: solid line - numerical solution and symbol - compressibility correction using the Karman-Tsien rule.

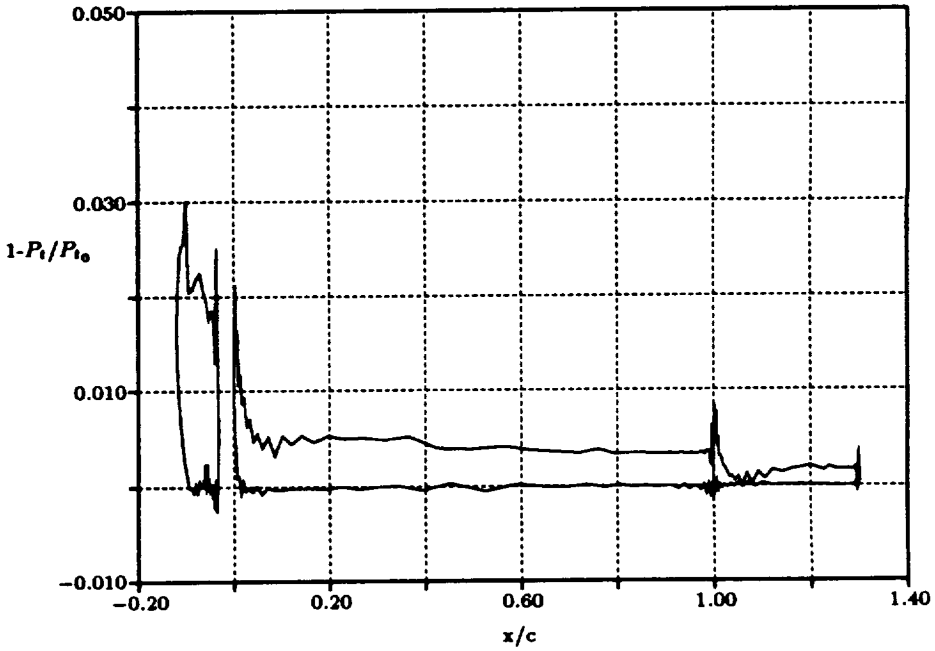


Figure 6.10 Multi-element airfoil surface total pressure error: initial mesh.

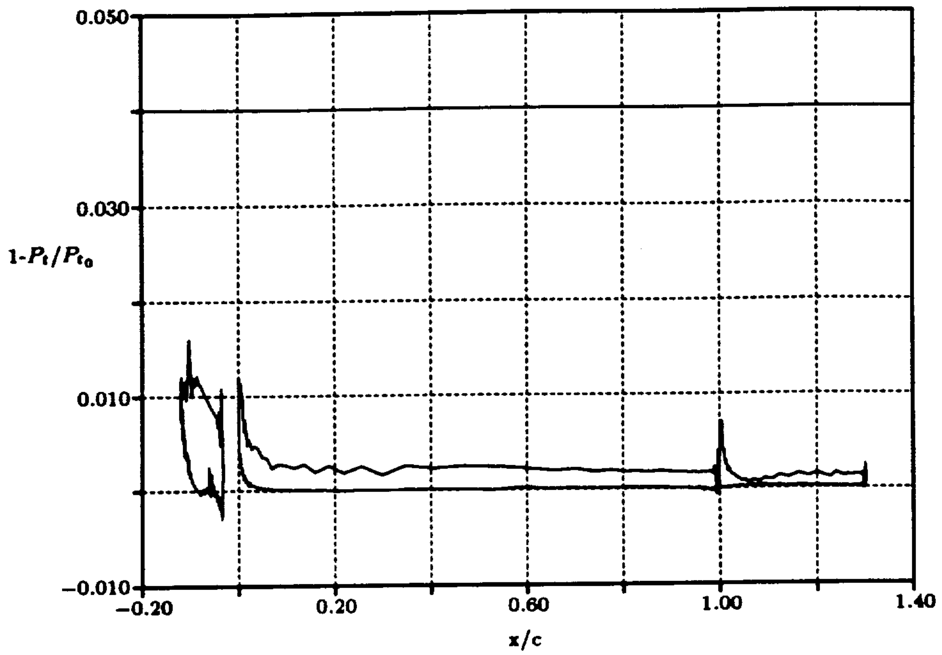


Figure 6.11 Multi-element airfoil surface total pressure error: 3rd adapted mesh.

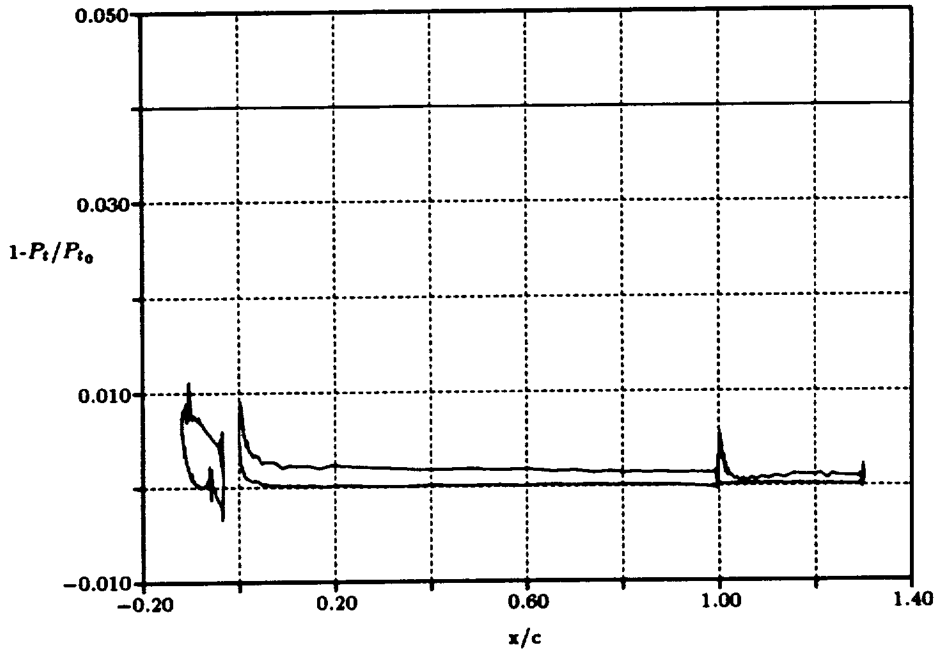


Figure 6.12 Multi-element airfoil surface total pressure error: final mesh.

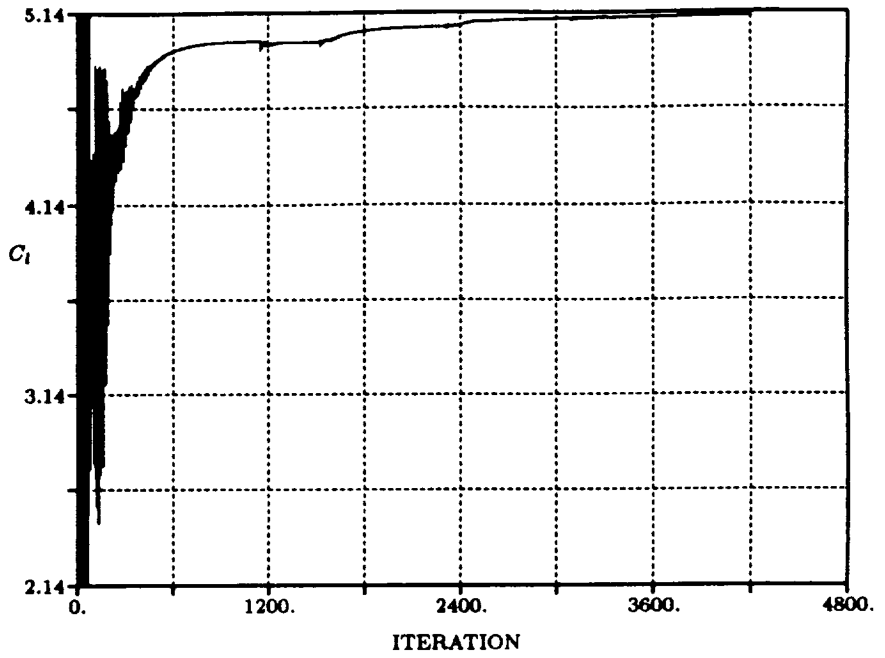


Figure 6.13 Multi-element airfoil: C_l verses iteration.

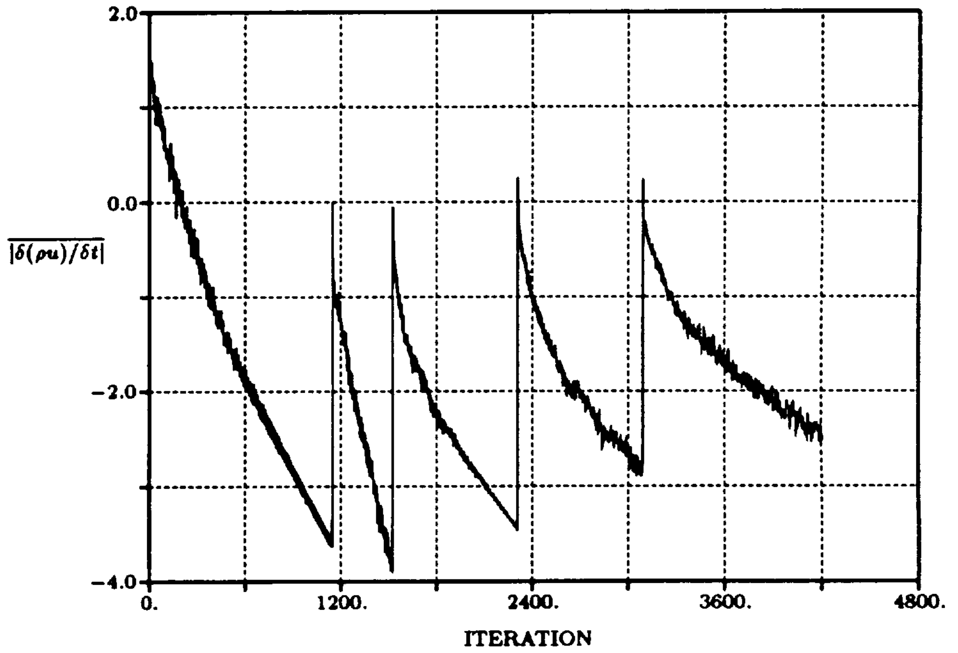


Figure 6.14 Multi-element airfoil: average $|\delta(\rho u)/\delta t|$ verses iteration.

6.2 Sanz Supercritical Compressor Cascade

This case is a shock free supercritical compressor cascade designed by Sanz [75] using the hodograph method. This case is described in the AGARD report [25] as test case A/CA-1. The hodograph method used to design this cascade results in a blade section with a nonclosed trailing edge. As noted in Reference [25], any attempt to close the blade trailing edge changes the blade section, resulting in a different solution. The flow conditions and blade characteristics are listed in Table 6.3. M_{exit} is the specified isentropic exit Mach number which sets the ratio of the exit static pressure to the inlet total pressure. This ratio is given as

$$\frac{p_{\text{exit}}}{P_{t_{\text{in}}}} = \left(1 + \frac{\gamma - 1}{2} M^2_{\text{exit}}\right)^{-\frac{\gamma}{\gamma-1}} \tag{6.2}$$

The ratio is 0.8177 for the current value of M_{exit}. The inflow and outflow boundaries for the mesh are placed one half an axial chord from the blade leading and trailing edges, respectively. At the inflow boundary total pressure, total temperature, and absolute flow angle are specified. At the outflow boundary the exit pressure is set. Flow tangency is enforced on the blade surface and periodicity is imposed at the upper and lower boundaries of the domain. At the open trailing edge a uniform jet flow condition is specified.

The sequence of solution adapted meshes is shown in Figures 6.15 to 6.16. Because the flow structure for this test case does not involve high gradients it only

Table 6.3 Flow conditions and blade characteristics for the Sanz supercritical compressor cascade.

<table>
<tr><th>M_{in}</th><th>M_{exit}</th><th>β_{in}</th><th>β_{exit}</th><th>Pitch/chord</th></tr>
<tr><td>0.711</td><td>0.544</td><td>30.81°</td><td>−0.35°</td><td>1.034</td></tr>
<tr><td colspan="5">Note: • Hodograph solution is available in Reference [25].
• Nonclosed trailing edge present in the physical profile.</td></tr>
</table>

Table 6.4 Sanz supercritical compressor cascade: $M_{\text{in}} = 0.711$ and $\beta_{\text{in}} = 30.81°$.

MESH	0	1
nodes	820	947
elements	1,469	1,717
C_{f_n}	0.9044	0.9039
C_{f_t}	-0.2817	-0.2825
$\delta_{\max}/$ pitch	0.095	0.095
$\delta_{\max}/\delta_{\min}$	1	4
Total CPU:	34 sec in Cray-YMP.	

takes one remesh cycle to converge. The mesh and loading coefficient information are given in Table 6.4. Mach number contours for the final solution are shown in Figure 6.17. A comparison of the converged Mach number distribution over the blade with the hodograph solution is shown in Figure 6.18. The agreement between the numerical and analytic solutions is excellent except in a small region near the sonic point on the suction surface. Note that the supercritical blade geometry is an isolated shock free design and that very small variations in the geometry will lead to the formation of shock waves. Since the blade coordinates given in Reference [25] have gaps in the sonic regions of the blade, the blade geometry in these regions is defined through interpolation. The difference between the computed solution and the analytic solution is the result of the difference between the interpolated and the unknown correct blade geometry in these regions. Convergence histories for the normal force coefficient and average residual are shown in Figures 6.19 and 6.20, respectively.

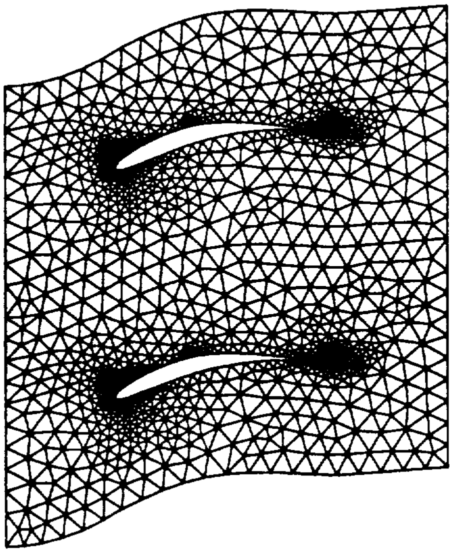


Figure 6.15 Sanz supercritical compressor cascade initial mesh: 820 nodes and 1,469 elements.

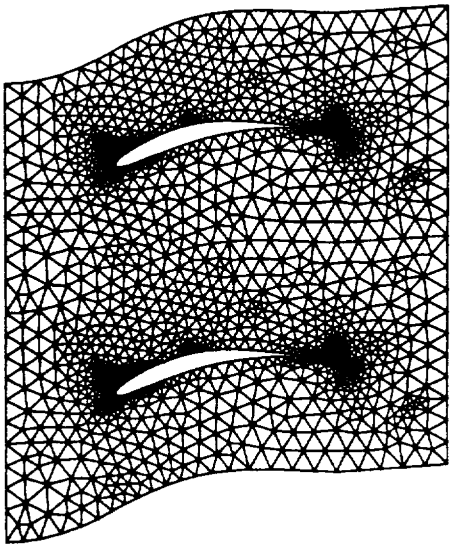


Figure 6.16 Sanz supercritical compressor cascade 1st (final) adapted mesh: 947 nodes and 1,717 elements.

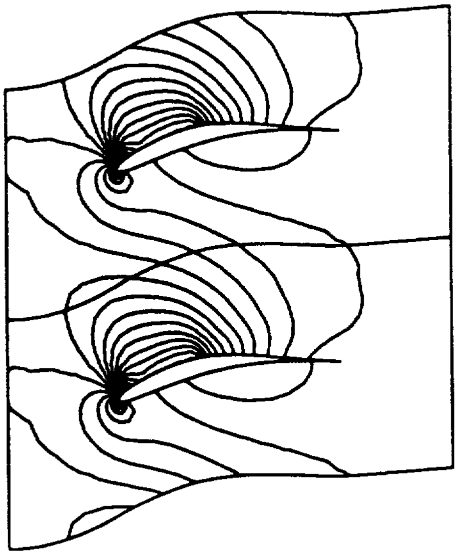


Figure 6.17 Sanz supercritical compressor cascade Mach number contours on the final mesh: cmin = 0.0, cmax = 1.30, and inc = 0.05.

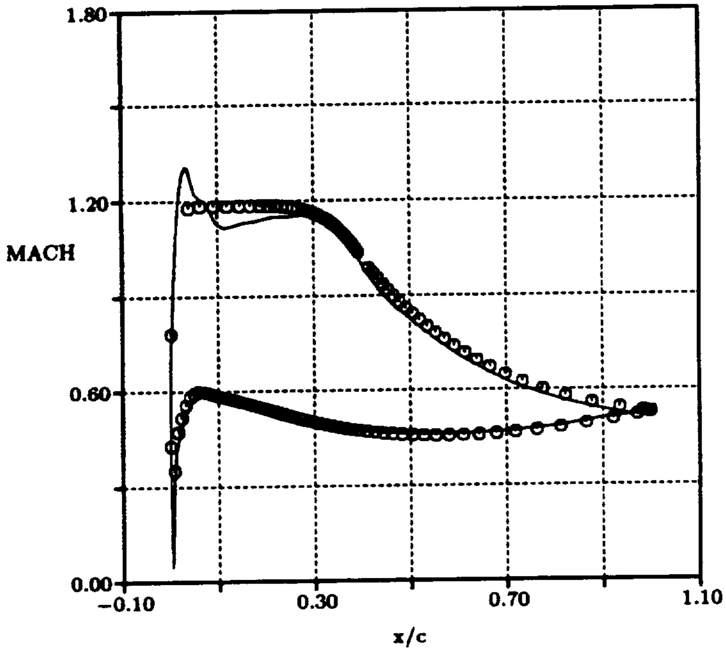


Figure 6.18 Sanz supercritical compressor cascade surface Mach number for the final mesh: solid line - numerical solution and symbol - analytic solution.

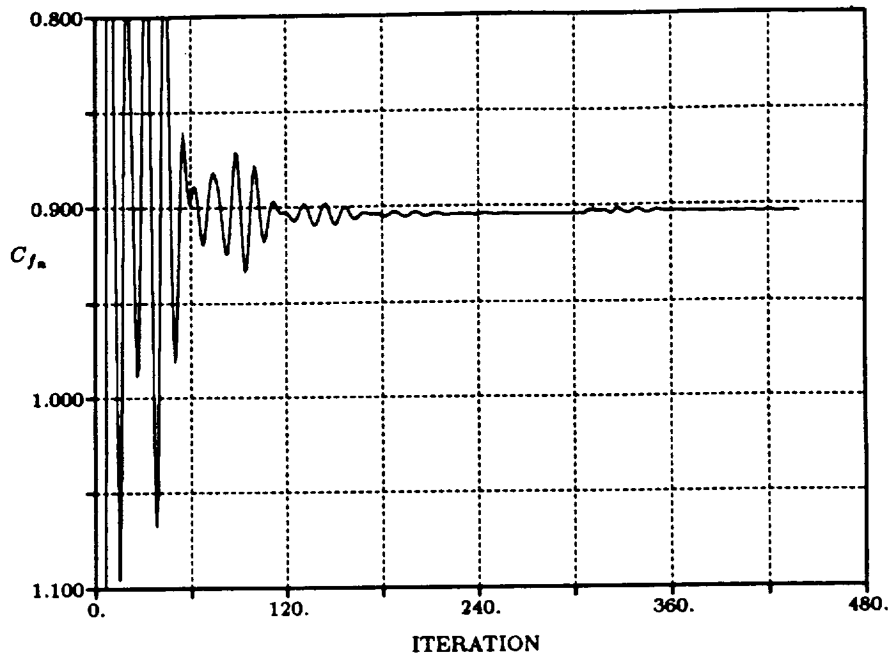


Figure 6.19 Sanz supercritical compressor cascade: C_{f_n} verses iteration.

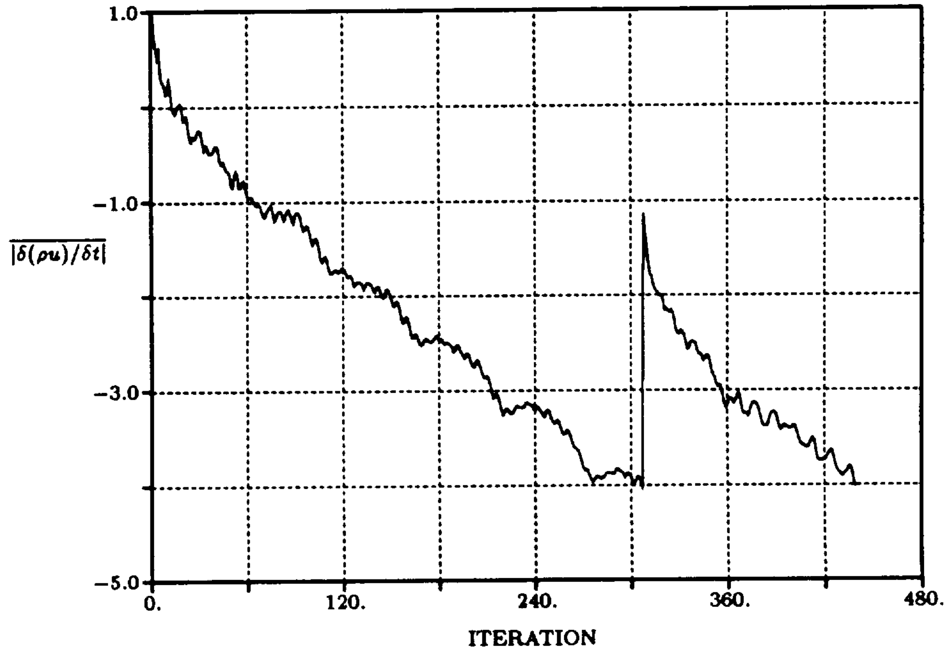


Figure 6.20 Sanz supercritical compressor cascade: average $|\delta(\rho u)/\delta t|$ verses iteration.

6.3 Sanz Subcritical Turbine Cascade

This subcritical turbine cascade, AGARD test case A/CA-3 [25], was also designed using a hodograph method. The blade section has a nonclosed trailing edge. The flow conditions and blade characteristics are tabulated in Table 6.5. The upstream and downstream mesh boundaries are placed one half an axial chord from the blade edges. The numerical boundary conditions are the same as those used in the supercritical compressor cascade case. The ratio of the exit static pressure to the inlet total pressure is 0.6788 for the present value of M_{exit}.

Figures 6.21 to 6.24 present the sequence of solution adapted meshes. Since the gradients are higher in this case due to the rapid acceleration of the flow through the turbine blade passage, it takes three mesh adaptation cycles to converge. Mesh properties and force coefficients for this case are summarized in Table 6.6. Mach number contours for the final solution are shown in Figure 6.25. Figures 6.26 and 6.27 compare the computed surface Mach number distribution with the hodograph solution for the initial and final meshes, respectively. These plots clearly show the improvement in the numerical solution as the mesh is refined.

For inviscid flows, the total pressure error is also a good indicator of numerical error in the solution scheme. Since this flow is isentropic the total pressure error should be zero. The plots of surface total pressure error shown in Figures 6.28

Table 6.5 Flow conditions and blade characteristics for the Sanz subcritical turbine cascade.

M_{in}	M_{exit}	β_{in}	β_{exit}	Pitch/chord
0.343	0.765	36.0°	−57.35°	0.566
Note:	• Hodograph solution is available in Reference [25]. • Nonclosed trailing edge present in the physical profile.			

and 6.29 demonstrate the improved accuracy of numerical solution as the mesh is refined. The maximum surface total pressure error is decreased from 4 % on the initial mesh down to 2 % on the final mesh. Convergence histories for the normal force coefficient and average residual are given in Figures 6.30 and 6.31, respectively.

Table 6.6 Sanz subcritical turbine cascade: $M_{\text{in}} = 0.343$ and $\beta_{\text{in}} = 36.0°$.

MESH	0	1	2	3
nodes	722	1,018	1,620	2,336
elements	1,244	1,799	2,962	4,367
C_{f_n}	1.9172	1.9239	1.9260	1.9279
C_{f_t}	1.7264	1.7218	1.7212	1.7216
$\delta_{\text{max}}/\text{pitch}$	0.075	0.075	0.075	0.075
$\delta_{\text{max}}/\delta_{\text{min}}$	1	2	4	8
Total CPU: 181 sec in Cray-YMP.				

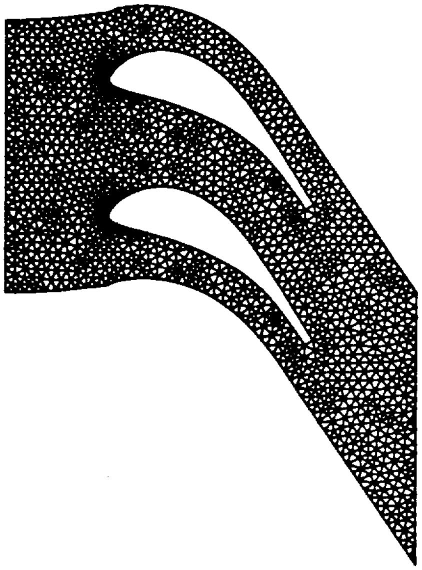


Figure 6.21 Sanz subcritical turbine cascade initial mesh: 722 nodes and 1,244 elements.

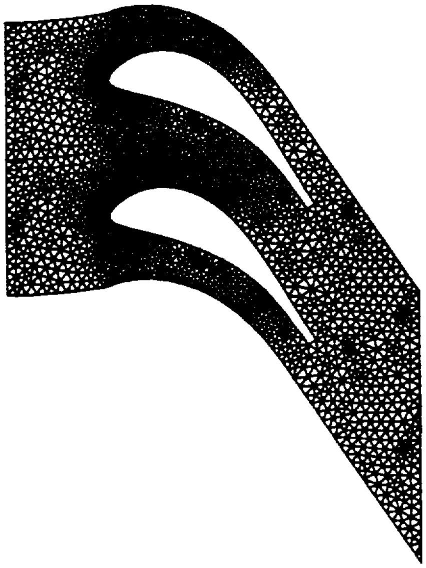


Figure 6.22 Sanz subcritical turbine cascade 1st adapted mesh: 1,018 nodes and 1,799 elements.

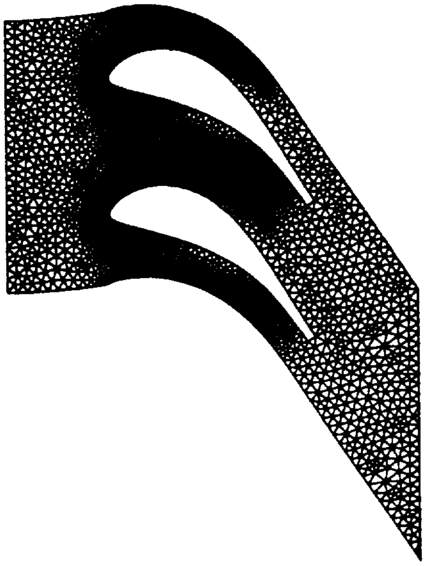


Figure 6.23 Sanz subcritical turbine cascade 2nd adapted mesh: 1,620 nodes and 2,962 elements.

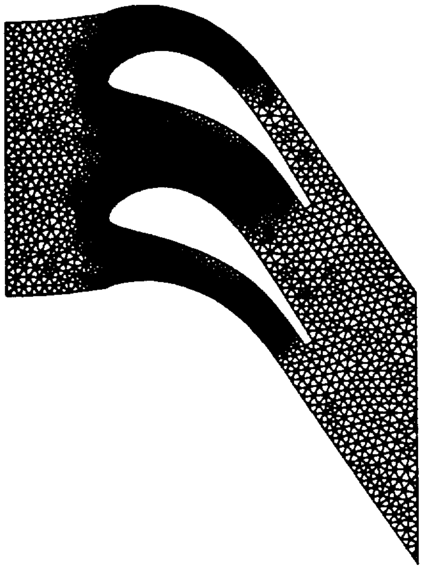


Figure 6.24 Sanz subcritical turbine cascade 3rd (final) adapted mesh: 2,336 nodes and 4,367 elements.

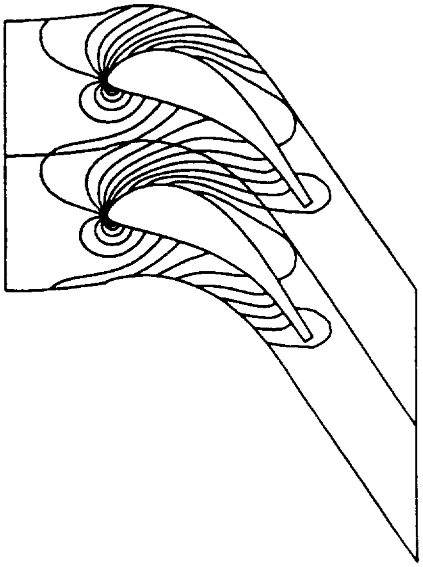


Figure 6.25 Sanz subcritical turbine cascade Mach number contours on the final mesh: cmin = 0.0, cmax = 0.80, and inc = 0.05.

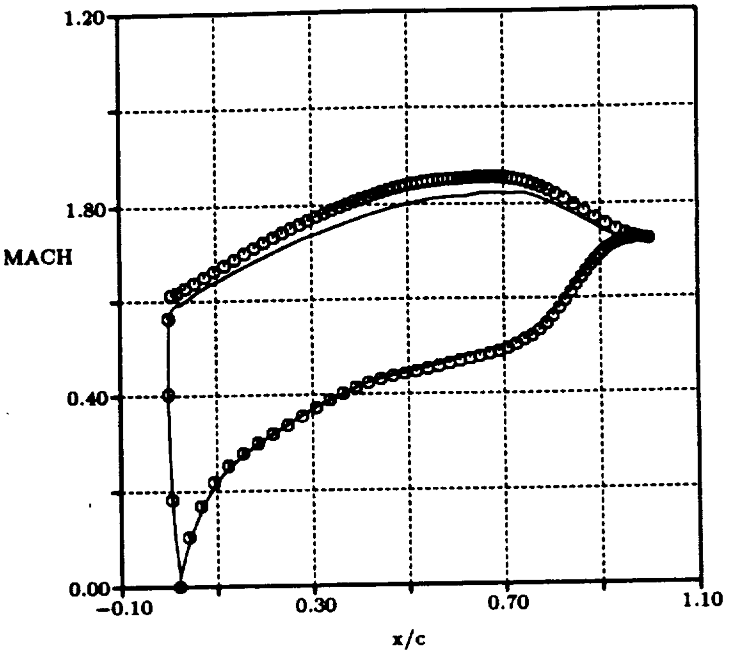


Figure 6.26 Sanz subcritical turbine cascade surface Mach number for the initial mesh: solid line - numerical solution and symbol - analytic solution.

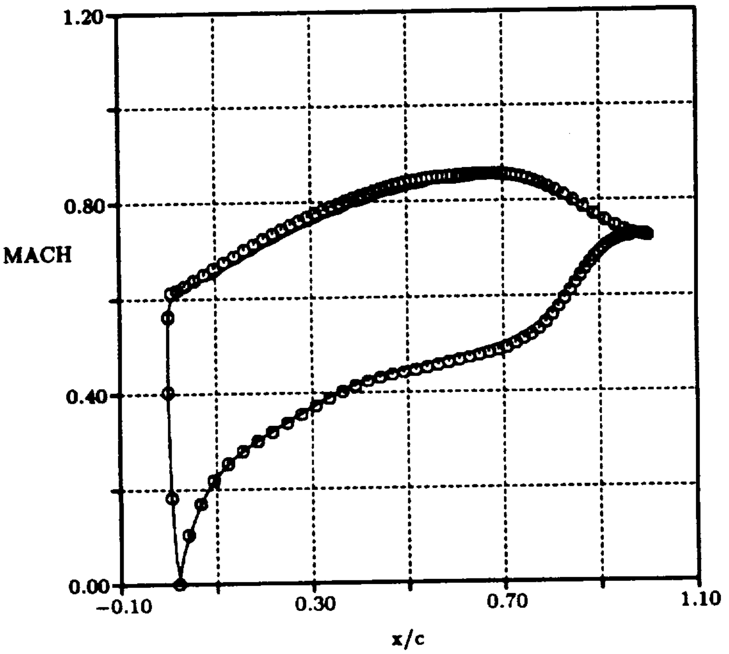


Figure 6.27 Sanz subcritical turbine cascade surface Mach number for the final mesh: solid line - numerical solution and symbol - analytic solution.

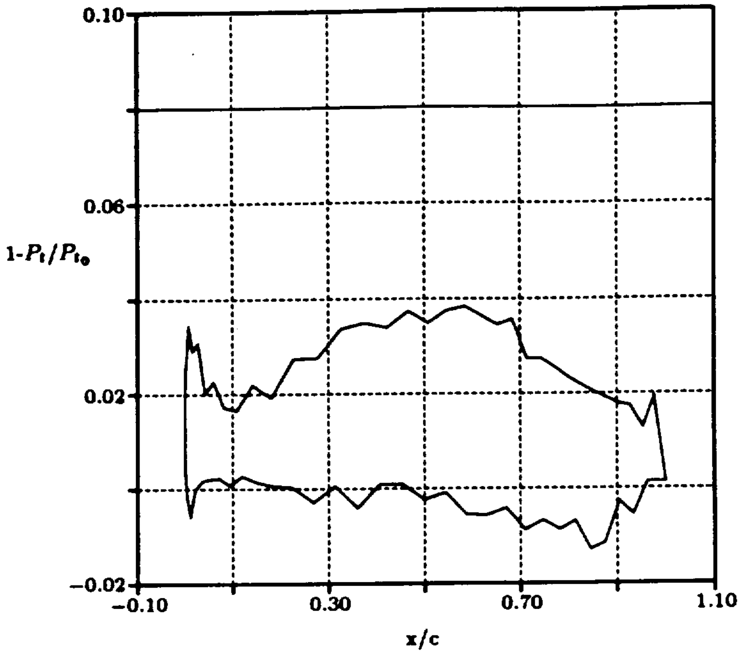


Figure 6.28 Sanz subcritical turbine cascade surface total pressure error: initial mesh.

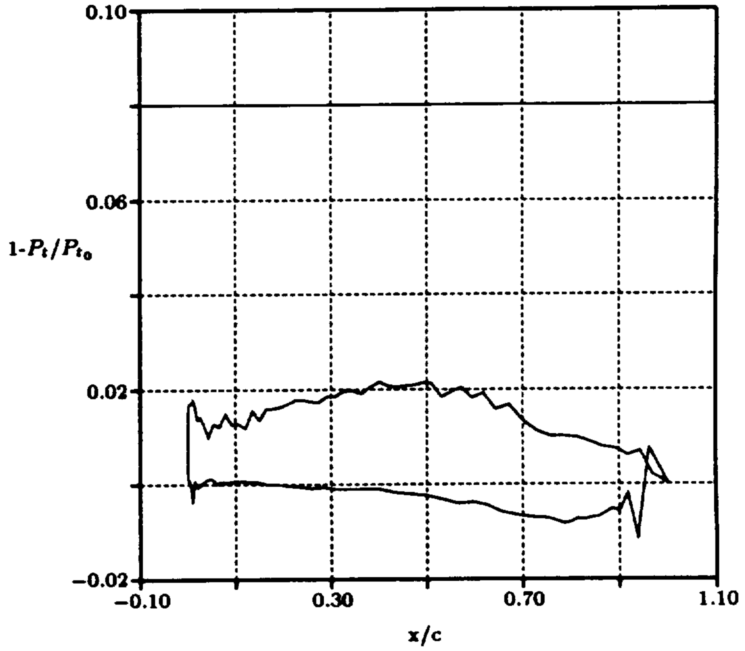


Figure 6.29 Sanz subcritical turbine cascade surface total pressure error: final mesh.

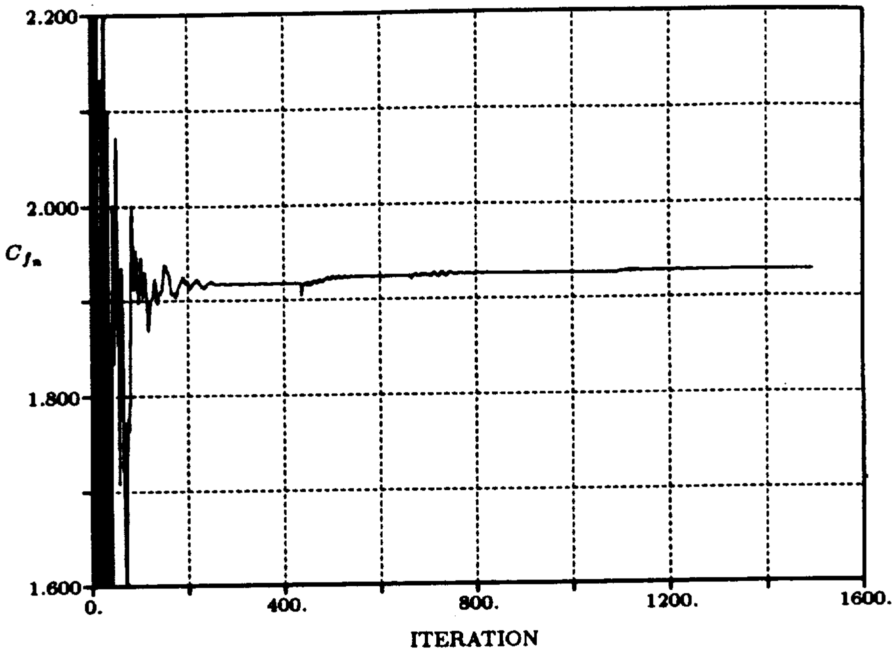


Figure 6.30 Sanz subcritical turbine cascade: C_{f_n} verses iteration.

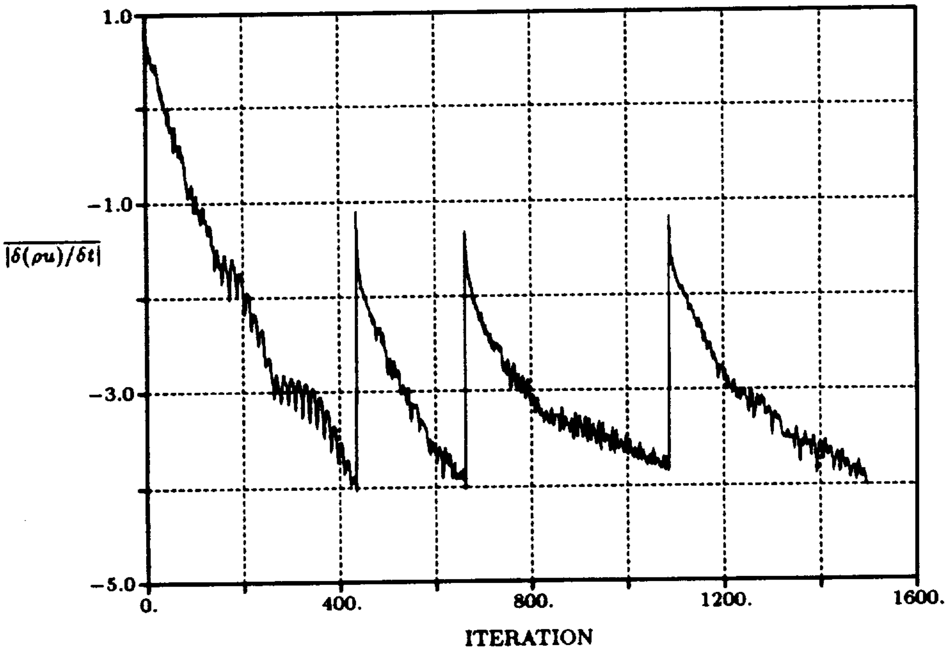


Figure 6.31 Sanz subcritical turbine cascade: average $|\delta(\rho u)/\delta t|$ verses iteration.

6.4 Denton Supersonic Staggered Wedge Cascade

The Denton wedge cascade, AGARD test case A/CA-2 [25], is a compressor cascade operating in a fully supersonic flow. This staggered wedge cascade has been carefully designed so that the reflected oblique shock wave is exactly cancelled resulting in an uniform flow between the two parallel surfaces. In order to close the cascade profile, another wedge is introduced on the pressure side resulting in an expansion off the downstream corner. Figure 6.38 presents the exact solution derived from shock/expansion theory. This test case illustrates the capability of the present solution adaptive scheme in capturing complicated shock wave structures. The flow conditions and blade characteristics are listed in Table 6.7.

The initial mesh is shown in Figure 6.32. The upstream and downstream mesh boundaries are located one half an axial chord from the blade leading and trailing edges, respectively. At the inflow boundary total pressure, total temperature, and absolute flow angle are specified as boundary conditions. At the outflow boundary the nonuniform static pressure boundary condition is imposed (see section 3.5.4).

Figures 6.32 to 6.35 present the mesh sequence during the solution adaptation process. The mesh characteristics are summarized in Table 6.8. Comparing the initial mesh Mach number contours shown in Figure 6.36 with the final mesh Mach number contours in Figure 6.37 shows an improved shock wave resolution for the

Table 6.7 Flow conditions and blade characteristics for the Denton supersonic staggered wedge cascade.

M_{in}	M_{exit}	β_{in}	β_{exit}	Pitch/chord
1.6	1.401	60.0°	60.0°	0.47985
Note: • Analytical solution is available in Reference [25]. • Nonuniform pressure present at the exit.				

Table 6.8 Denton supersonic staggered wedge cascade: $M_{\text{in}} = 1.60$, $\beta_{\text{in}} = 60.0^\circ$, $(C_{f_t})_{\text{exact}} = 0.03774$, and $(C_{f_n})_{\text{exact}} = -0.0654$.

MESH	0	1	2	3
nodes	1,231	1,982	3,625	7,526
elements	2,196	3,562	6,762	14,440
C_{f_n}	0.0411	0.0375	0.0384	0.0382
C_{f_t}	-0.0722	-0.0647	-0.0667	-0.0664
δ_{max}/ pitch	0.08	0.08	0.08	0.08
$\delta_{\text{max}}/\delta_{\text{min}}$	1	3.1	12.5	25
Total CPU: 880 sec in Cray-YMP.				

adapted solution. Mach number contours for the final and theoretical solutions are shown in Figures 6.37 and 6.38, respectively. Even though the shock wave structure is complex, the remeshing procedure accurately resolves the flow. The computed surface Mach number distributions on the initial and final meshes are compared with the theoretical solution in Figures 6.39 and 6.40, respectively. These plots show a great improvement in the accuracy of numerical results. Since the present scheme is not a monotonic scheme, overshoots are shown in the shock wave regions. Comparing the computed loading coefficients C_{f_n} and C_{f_t} to the analytical values in Table 6.8 shows a reduction in the error as the mesh is refined with the exception of the 1st adapted mesh solution. This indicates that it is difficult to determine the improvement of solution based on the loading coefficients only. Since the loading coefficients are integral quantities, numerical errors may cancel after integrating over the blade surface.

Convergence histories are given in Figures 6.41 and 6.42. It is noted that the CPU time this case requires is five times longer than that of the Sanz subcritical turbine cascade case. This is due to the increasing number of mesh points and decreasing mesh length scale as the mesh is adapted to the solution.

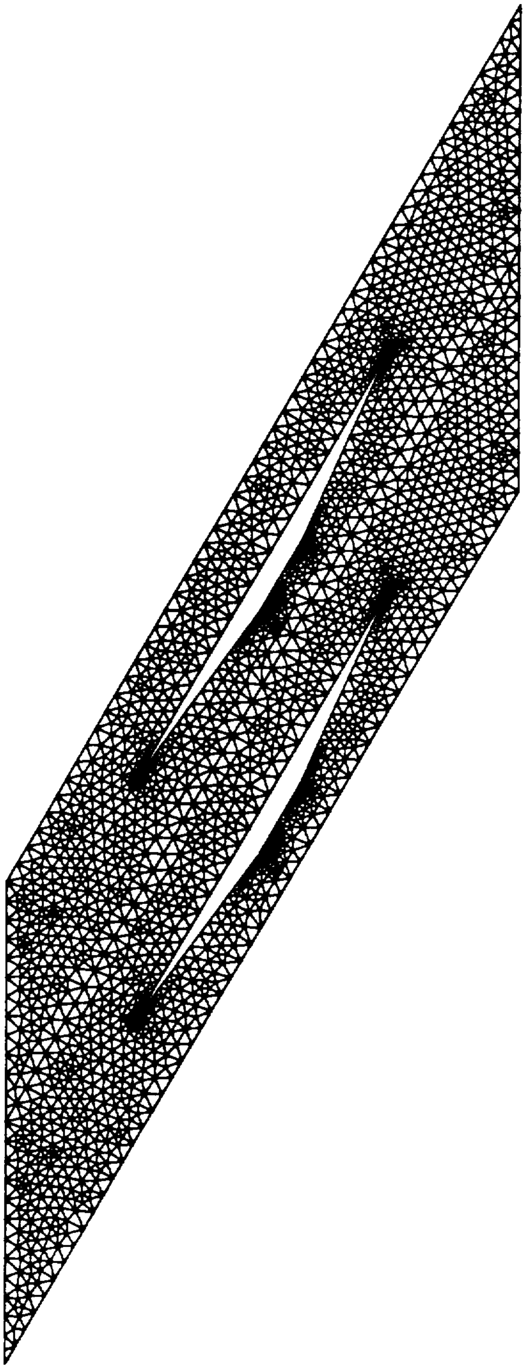


Figure 6.32 Denton supersonic staggered wedge cascade initial mesh: 1,231 nodes and 2,196 elements.

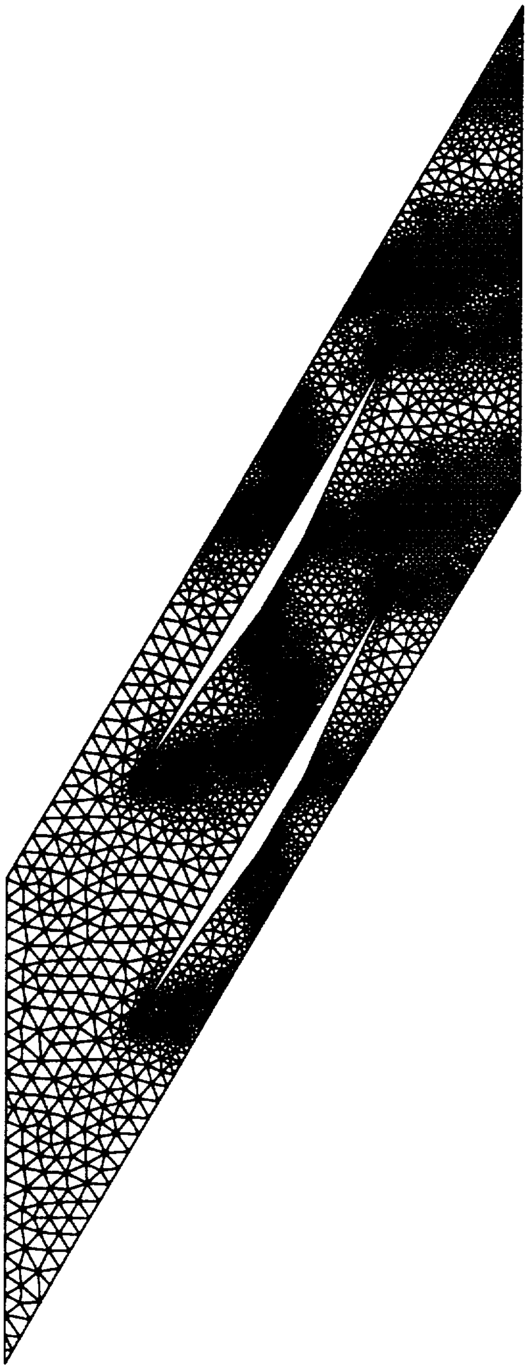


Figure 6.33 Denton supersonic staggered wedge cascade 1^{st} adapted mesh: 1,982 nodes and 3,562 elements.

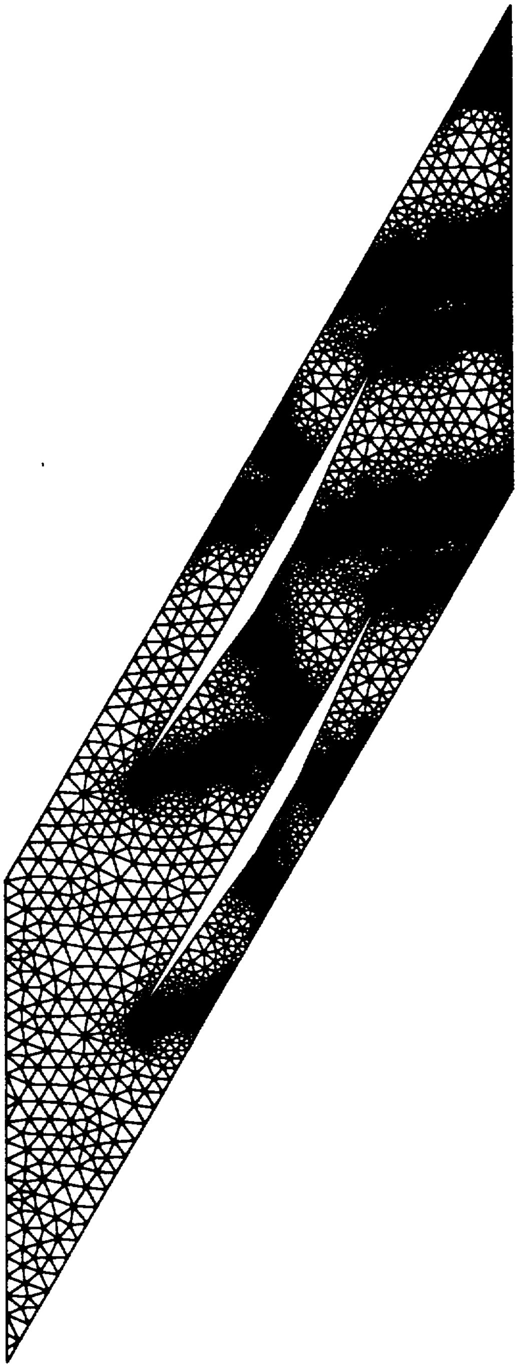


Figure 6.34 Denton supersonic staggered wedge cascade 2$^{\text{nd}}$ adapted mesh: 3,625 nodes and 6,762 elements.

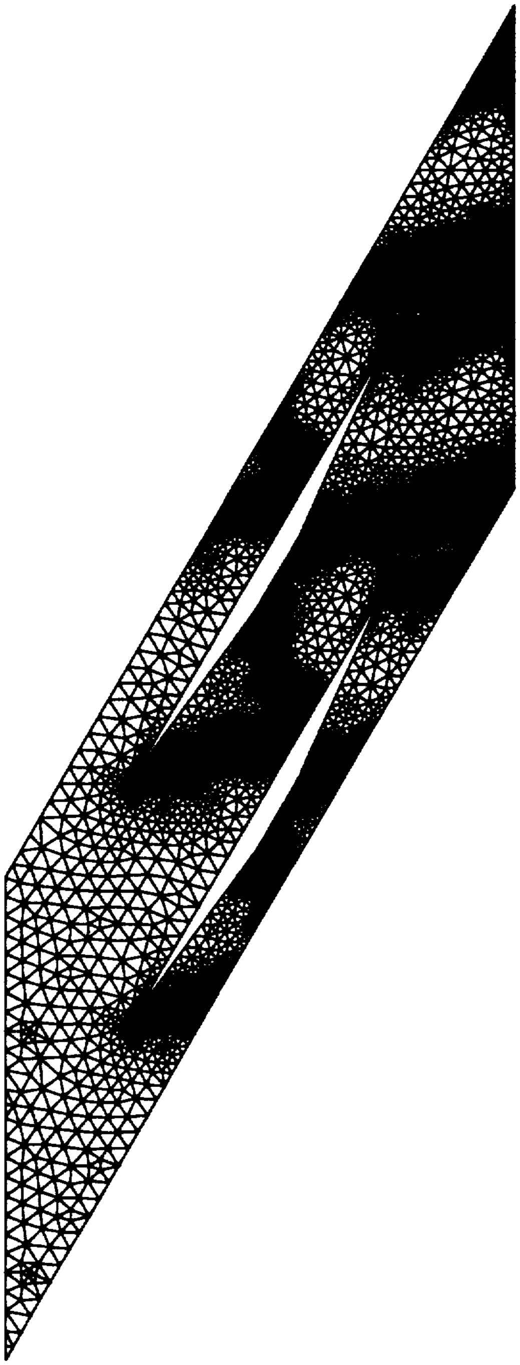


Figure 6.35 Denton supersonic staggered wedge cascade 3rd (final) adapted mesh: 7,562 nodes and 14,440 elements.

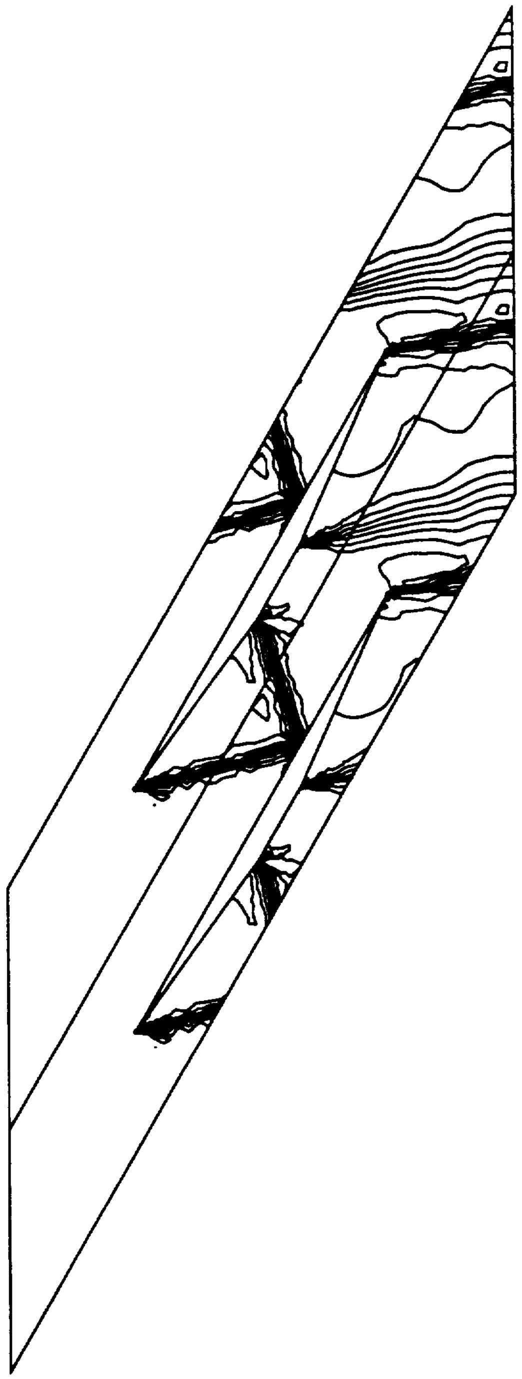


Figure 6.36 Denton supersonic staggered wedge cascade Mach number contours on the initial mesh: cmin = 1.17, cmax = 1.62, and inc = 0.03.

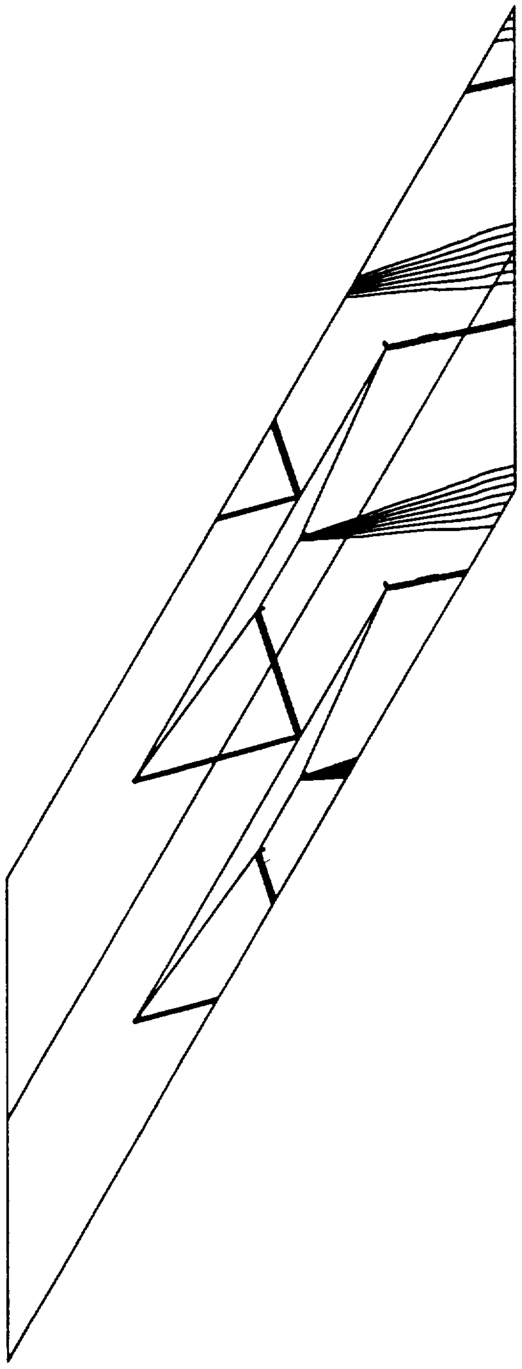


Figure 6.37 Denton supersonic staggered wedge cascade Mach number contours on the final mesh: cmin = 1.17, cmax = 1.62, and inc = 0.03.

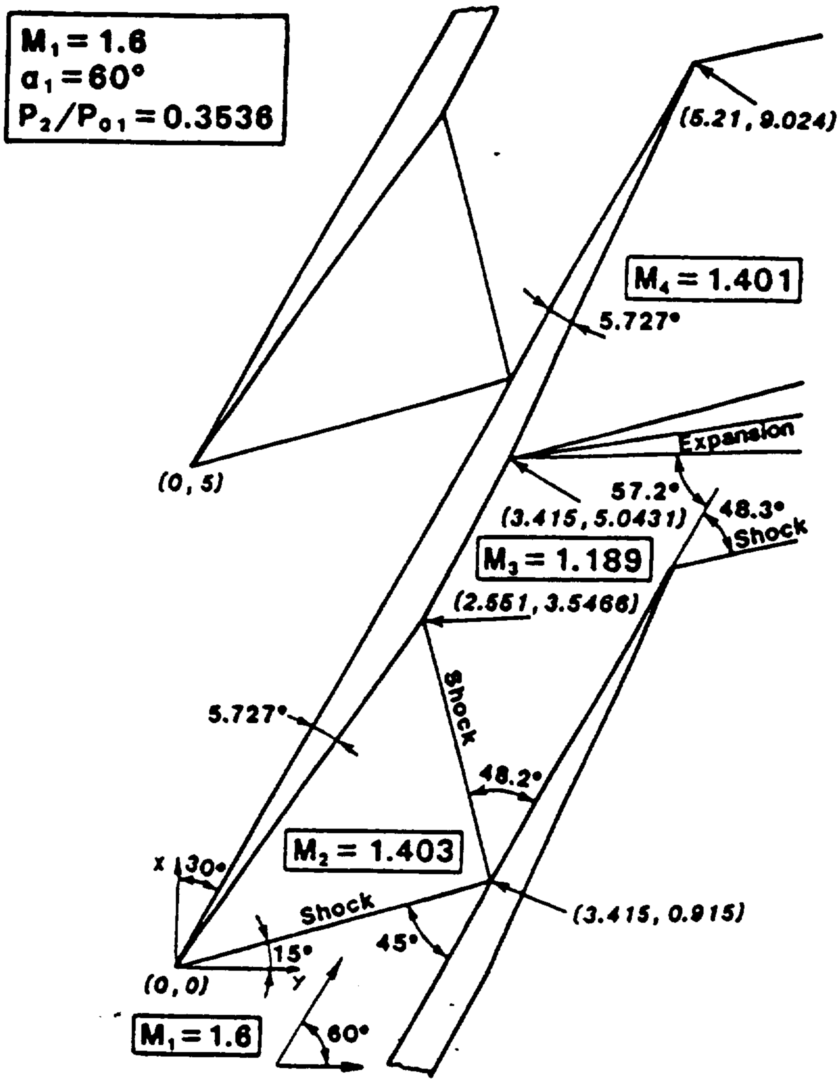


Figure 6.38 Denton supersonic staggered wedge cascade Mach number contours: analytic solution (Denton *et al.* [25]).

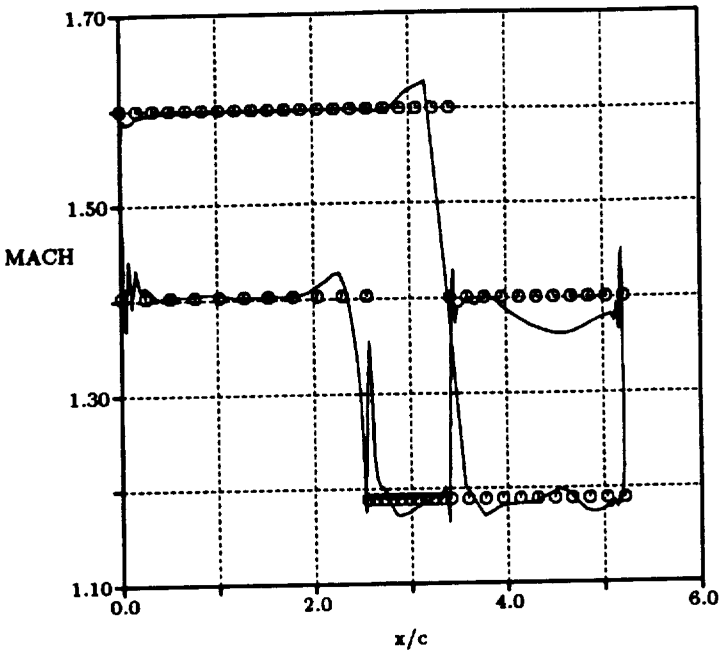


Figure 6.39 Denton supersonic staggered wedge cascade surface Mach number for the initial mesh: solid line - numerical solution and symbol - analytic solution.

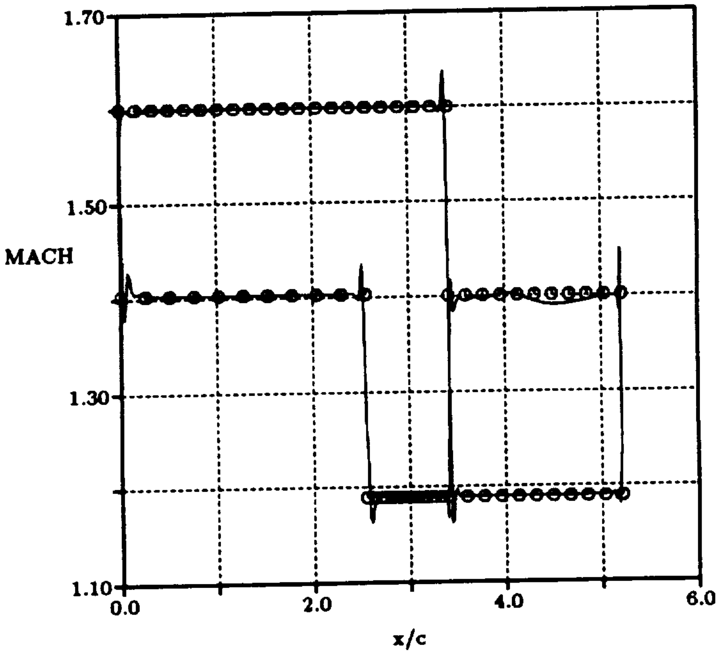


Figure 6.40 Denton supersonic staggered wedge cascade surface Mach number for the final mesh: solid line - numerical solution and symbol - analytic solution.

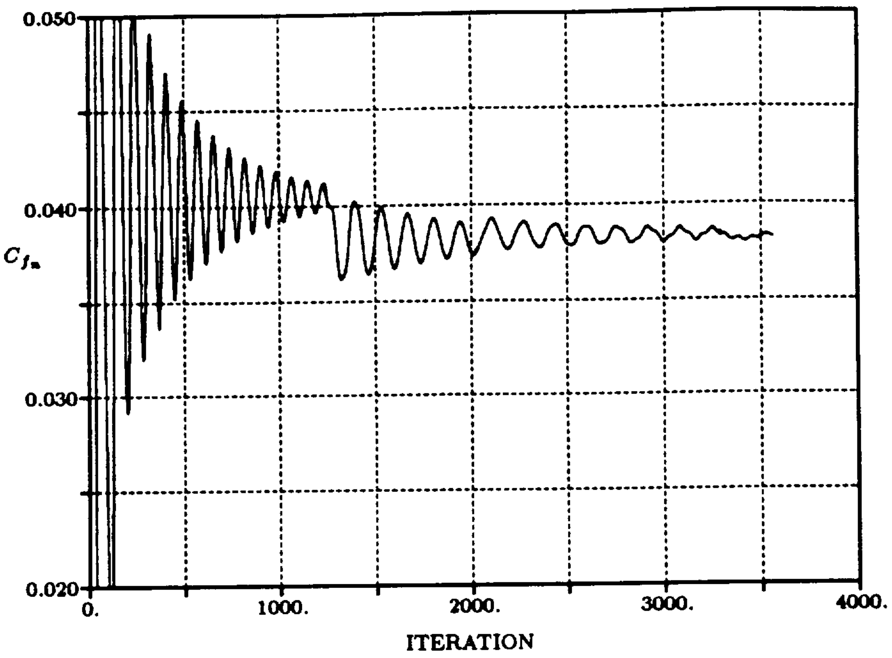


Figure 6.41 Denton supersonic staggered wedge cascade: C_{f_n} verses iteration.

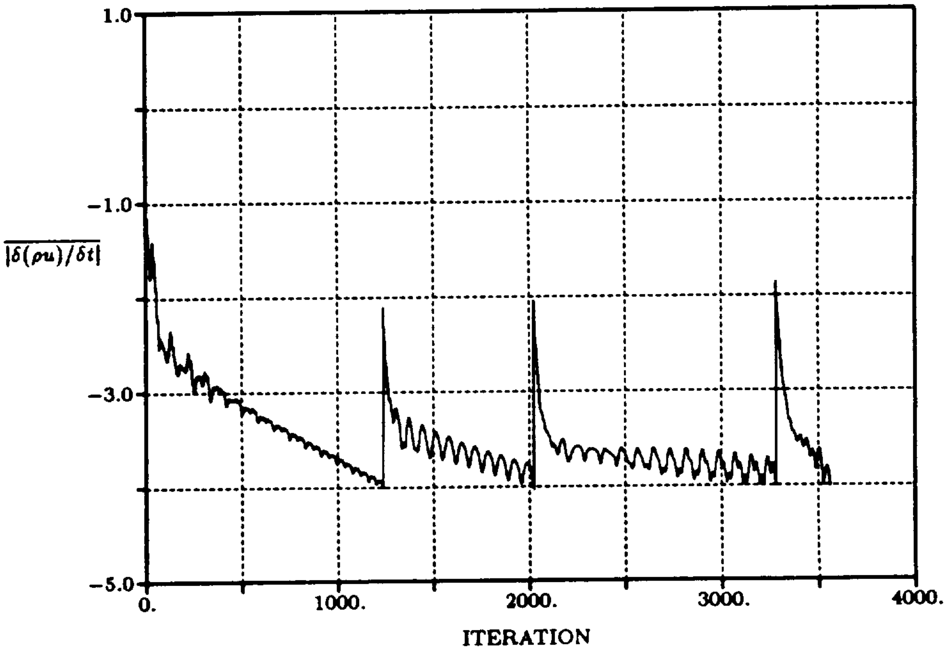


Figure 6.42 Denton supersonic staggered wedge cascade: average $|\delta(\rho u)/\delta t|$ verses iteration.

6.5 NASA Rotor 67 Transonic Fan

NASA rotor 67 is a low-aspect-ratio transonic axial-flow fan rotor. Laser anemometer surveys of the flowfield were made for operating conditions near peak efficiency and near stall by Strazisar *et al.* [79]. The fan tip relative Mach number is 1.38. The experiments were conducted in a rotor-only configuration making them good test cases for the three-dimensional flow solver. Only the near peak efficiency test condition will be considered here. Since the flow in this machine is three-dimensional, axisymmetric through-flow information is required as input to the present quasi-three-dimensional analysis. The streamsurface location and thickness data used in the present calculations were obtained from Reference [60]. It is important to note that the accuracy of the present results depends on the accuracy of the streamsurface data, which is difficult to validate. This test case is important in illustrating how the adaptive solution procedure resolves realistic flow structures.

The adaptive mesh solution is computed for the 30 %-span streamsurface (measured from the shroud). In this case the inlet boundary for the mesh is placed three quarters an axial chord from the blade leading edge, and the exit boundary is located half an axial chord from the blade trailing edge. At the inflow boundary total pressure, total temperature, and absolute flow angle are specified as boundary conditions. At the outflow boundary the exit pressure is imposed. Since there is high turning involved in this flow, it is not possible to start a solution with constant initial conditions. Therefore, the quasi-one-dimensional solution is used to provide a smooth variation of initial flow conditions (see section 3.4). The inflow and outflow condition as determined from the through-flow analysis are summarized in Table 6.9.

This span station is particularly interesting because the upstream relative Mach number is 1.20. With a blunt leading edge, a bow shock wave stands away from the blade leading edge. To accurately predict this flow the leading edge must be well resolved. Referring to Figures 6.50 and 6.52 it is clear that a refinement parameter

Table 6.9 NASA Rotor 67 far field flow conditions.

Case	$P_{t_{in}}$ (psi)	$T_{t_{in}}$ (R)	$r\,v_\theta$	$p_{exit}/P_{t_{in}}$
30 % span	14.7	518.7°	0	1.21
50 % span	14.7	518.7°	0	1.25
70 % span	14.7	518.7°	0	1.16

based on surface curvature does a good job in resolving the leading edge region. Even on the initial mesh the bow shock wave is located in the proper location. Without this refinement the adaptive procedure improperly locates the bow shock wave. This changes the flow conditions within the blade passage, resulting in a very poor initial solution. It then takes many mesh adaptation cycles to converge to the correct solution. The present adaptation criteria leads to a converged solution after three remeshes (see Figures 6.43 to 6.46). Mesh properties and force coefficients are summarized in Table 6.10.

The relative Mach number contours for the final mesh are shown in Figure 6.47. The bow shock reflection on the blade suction surface is not well resolved, because the shock strength is much weaker than the normal shock. Figure 6.48 presents contours of the experimentally measured relative Mach number for the 30 %-span station. Even though a very large amount of experimental data was taken, the experimental data used in making these contour plots are still very sparse. This fact makes it very difficult to evaluate the overall accuracy of the computed solution. The compute results show a good agreement with the experimental data in the upstream of the blade leading edge. Note the experimental data shows a passage shock wave located further upstream than the shock wave location computed here. Boundary layer blockage plays an important role in determining the location of the passage shock wave. Since this is an inviscid solution the shock wave is located downstream of the correct experimental location.

Figures 6.51 and 6.52 show a blowup of the leading edge region of the blade. There are about 20 mesh points along the half circle of the leading edge. The bow shock wave is well resolved. Plots of the surface pressure coefficient are shown in Figures 6.53 and 6.54 for the initial and final meshes, respectively. Convergence histories are shown in Figures 6.55 and 6.56. The normal force coefficient (force in the θ-direction) converges rapidly to a constant value. The adaptation cycle has been stopped at the 3rd adapted mesh, since further refinement will lead to a large number of mesh points which in turn increases the computation time. The relative change in the force coefficient between last two meshes is within 1 %.

Table 6.10 NASA Rotor 67 operating at peak efficiency: 30 % span station.

MESH	0	1	2	3
nodes	1,327	2,031	5,138	7,245
elements	2,375	3,700	9,877	14,074
C_{f_n}	1.2149	1.1919	1.1950	1.2073
C_{f_t}	-1.6292	-1.5763	-1.5931	-1.6270
$\delta_{\mathbf{max}}/\text{pitch}$	0.067	0.067	0.067	0.067
$\delta_{\mathbf{max}}/\delta_{\mathbf{min}}$	1	4	16	32
Total CPU: 553 sec in Cray-YMP.				

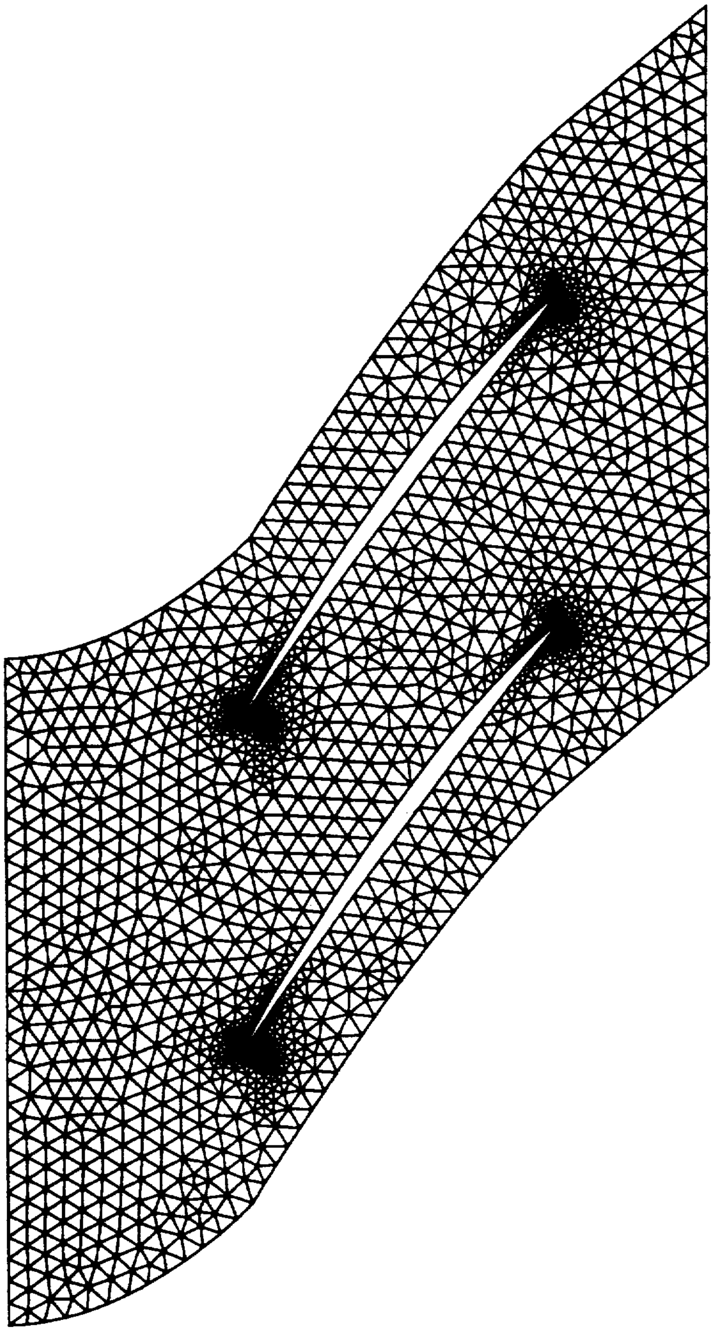


Figure 6.43 NASA Rotor 67 30 % span from shroud initial mesh: 1,327 nodes and 2,375 elements.

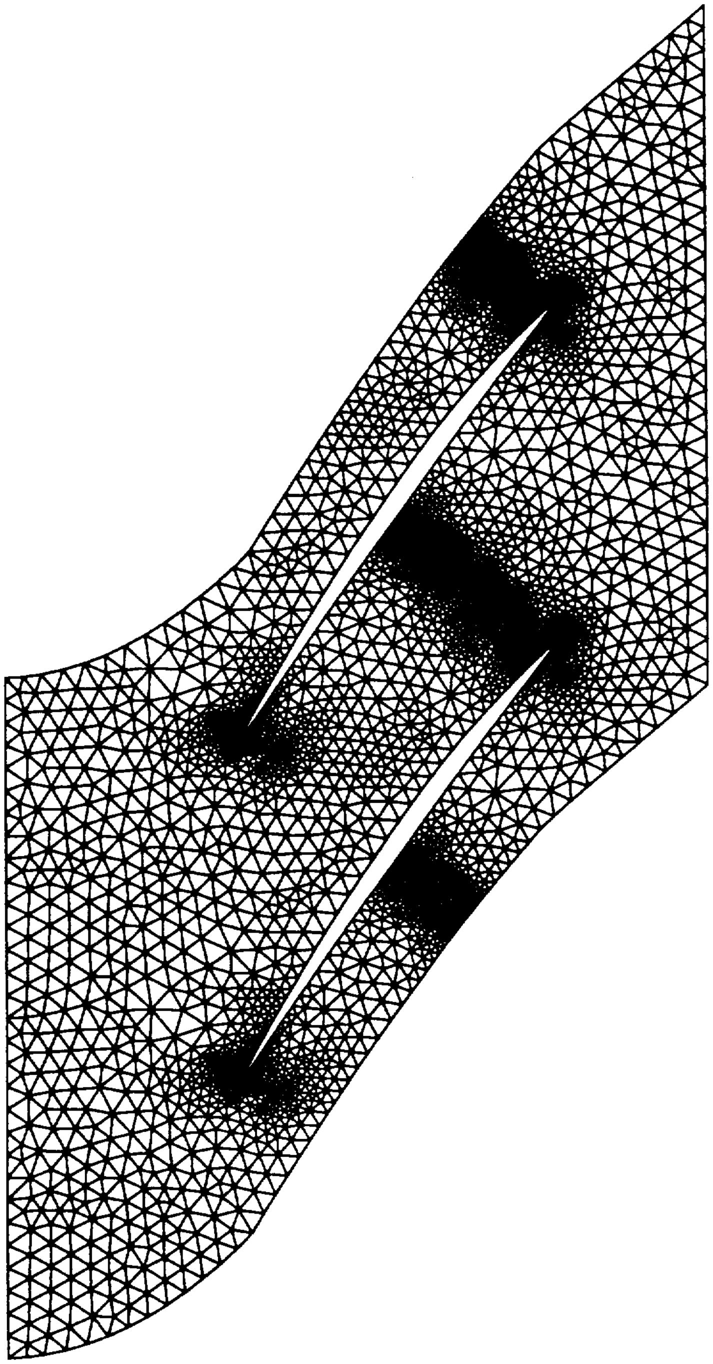


Figure 6.44 NASA Rotor 67 30 % span from shroud 1^{st} adapted mesh: 2,031 nodes and 3,700 elements.

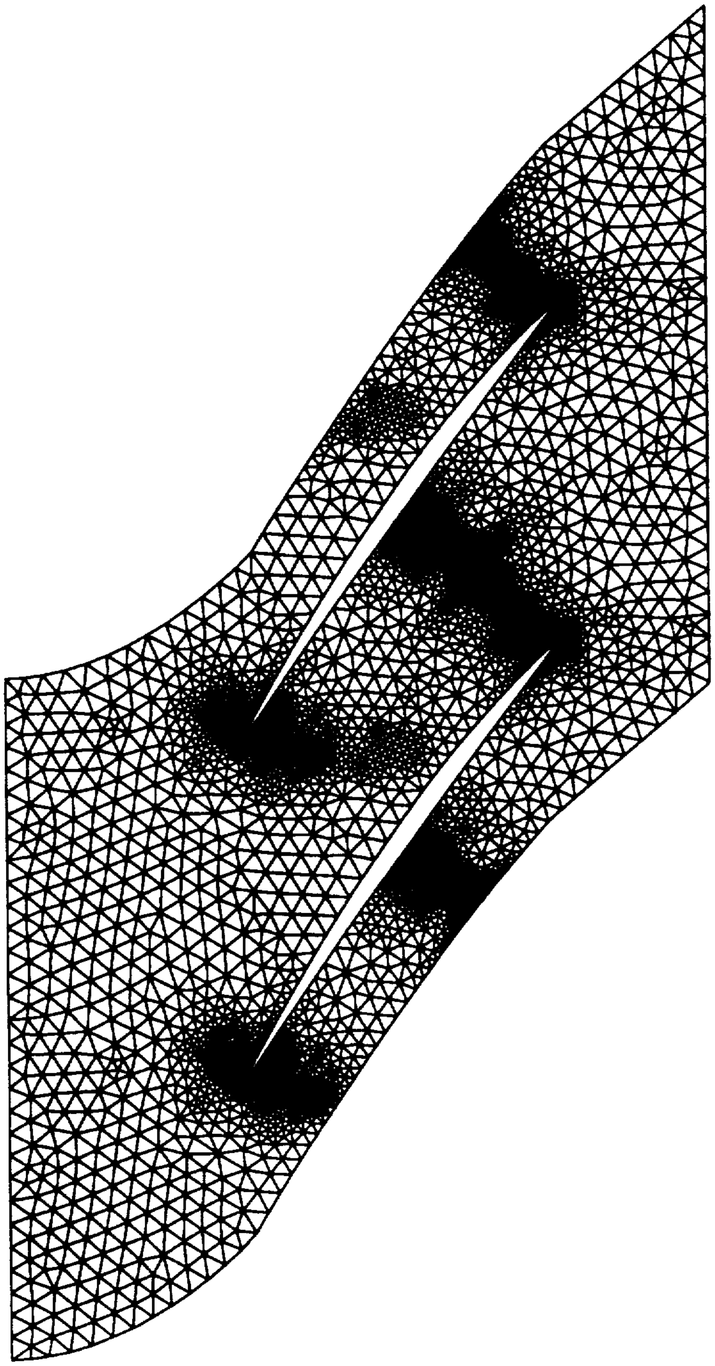


Figure 6.45 NASA Rotor 67 30 % span from shroud 2nd adapted mesh: 5,138 nodes and 9,877 elements.

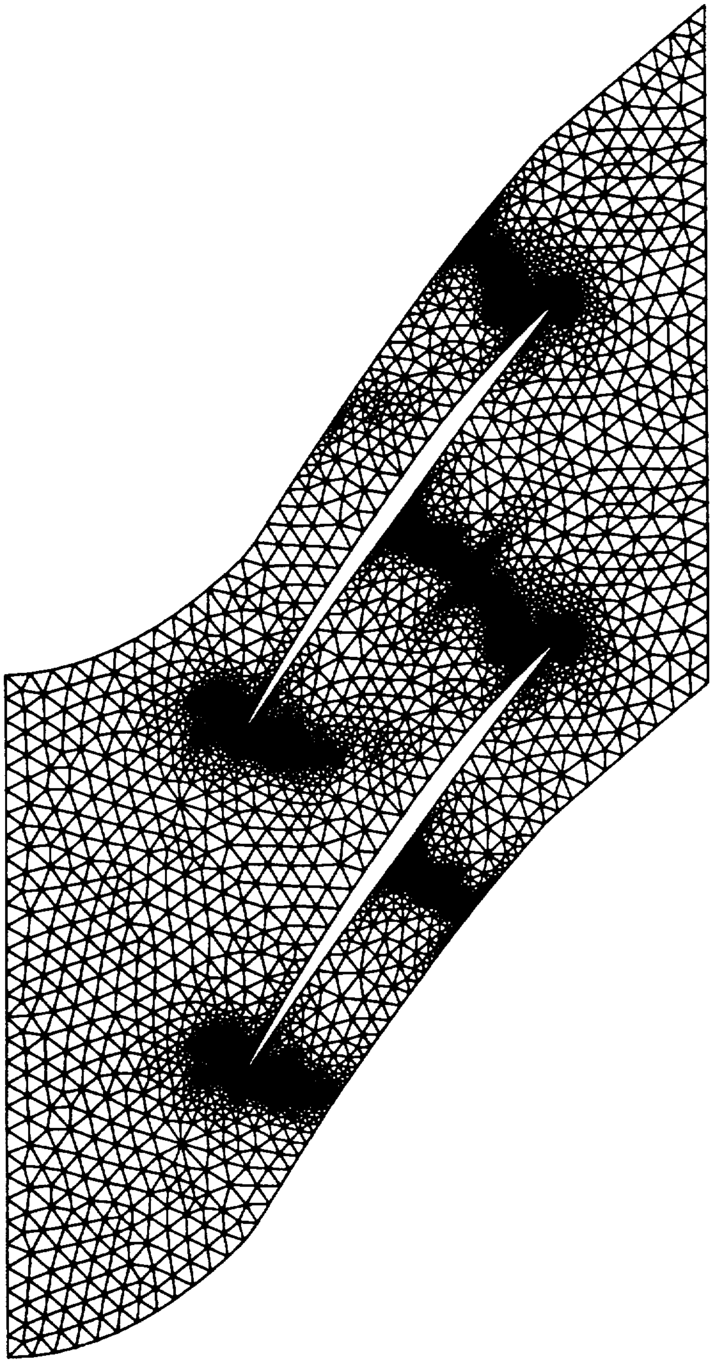


Figure 6.46 NASA Rotor 67 30 % span from shroud 3rd (final) adapted mesh: 7,245 nodes and 14,074 elements.

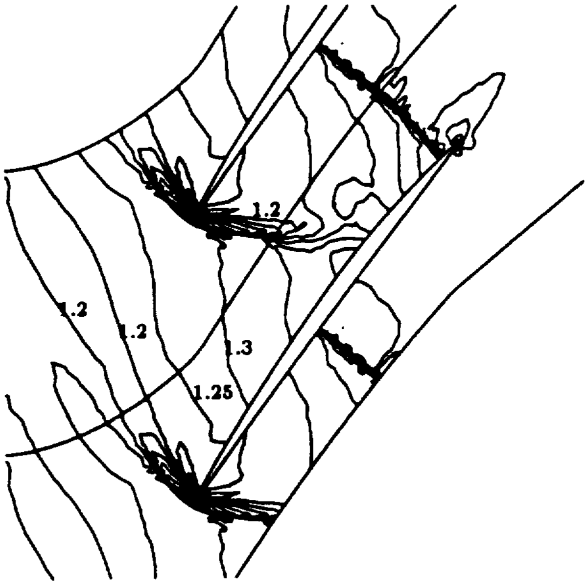


Figure 6.47 NASA Rotor 67 30 % span from shroud relative Mach number contours on the final mesh: cmin = 0.0, cmax = 1.70, and inc = 0.05.

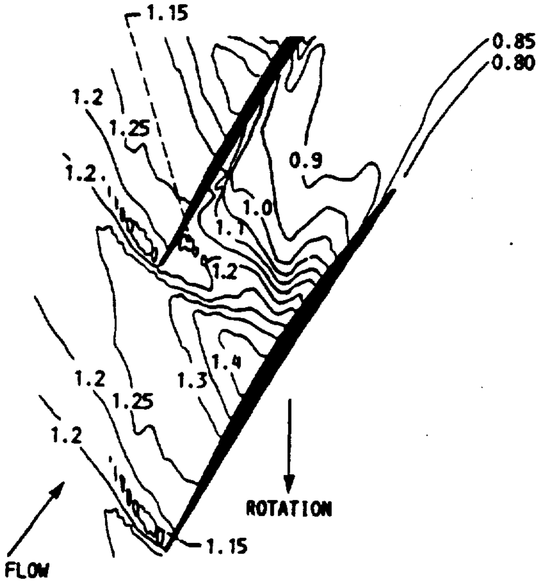


Figure 6.48 NASA Rotor 67 30 % span from shroud relative Mach number contours: experimental data (Strazisar, *et al.* [79]).

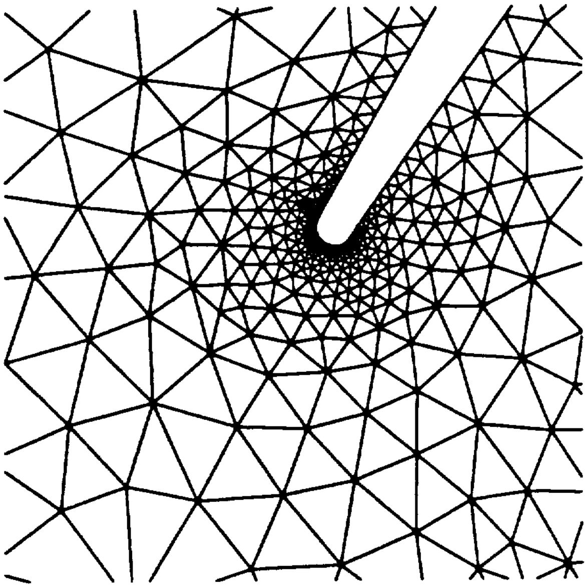


Figure 6.49 NASA Rotor 67 30 % span from shroud: leading edge blowup of the initial mesh.

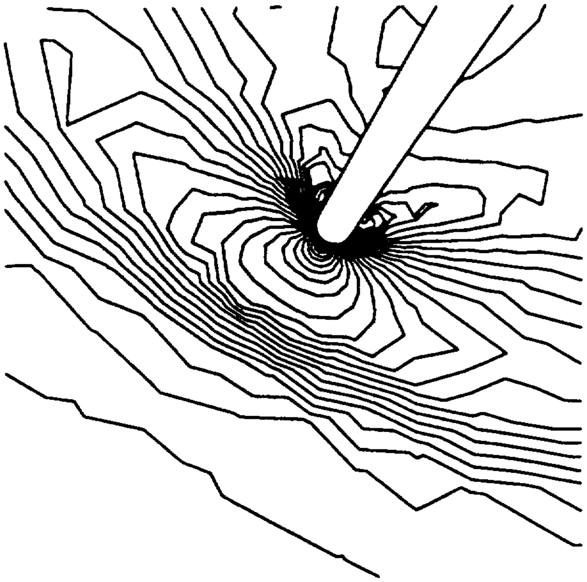


Figure 6.50 NASA Rotor 67 30 % span from shroud relative Mach number contours (cmin = 0.0, cmax = 1.70, and inc = 0.05): initial mesh leading edge region.

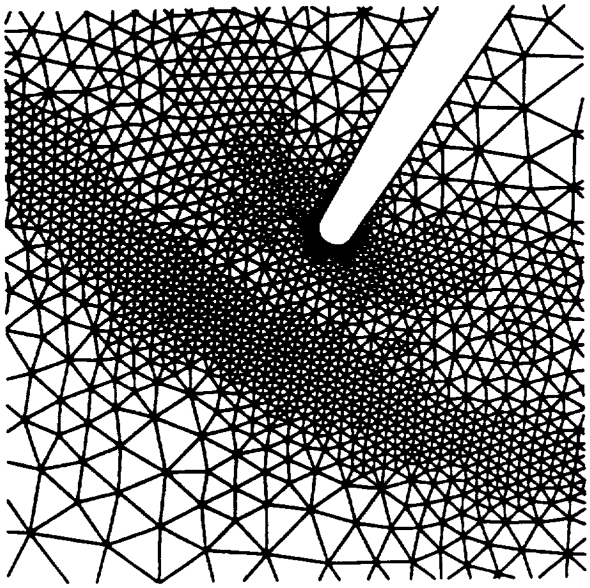


Figure 6.51 NASA Rotor 67 30 % span from shroud: leading edge blowup of the final mesh.

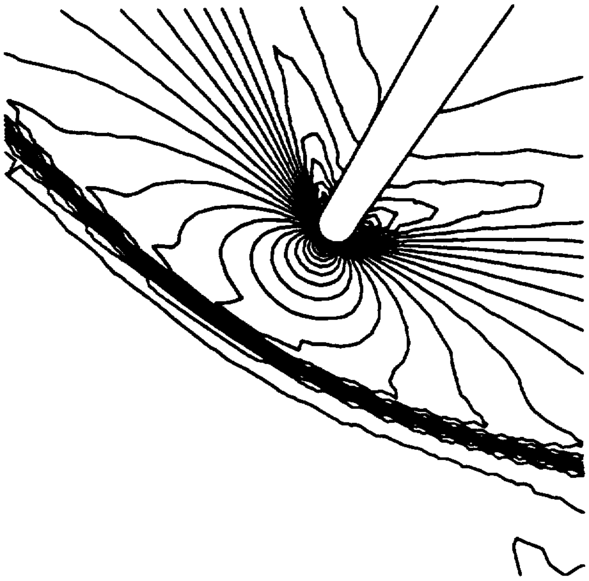


Figure 6.52 NASA Rotor 67 30 % span from shroud relative Mach number contours (cmin = 0.0, cmax = 1.70, and inc = 0.05): final mesh leading edge region.

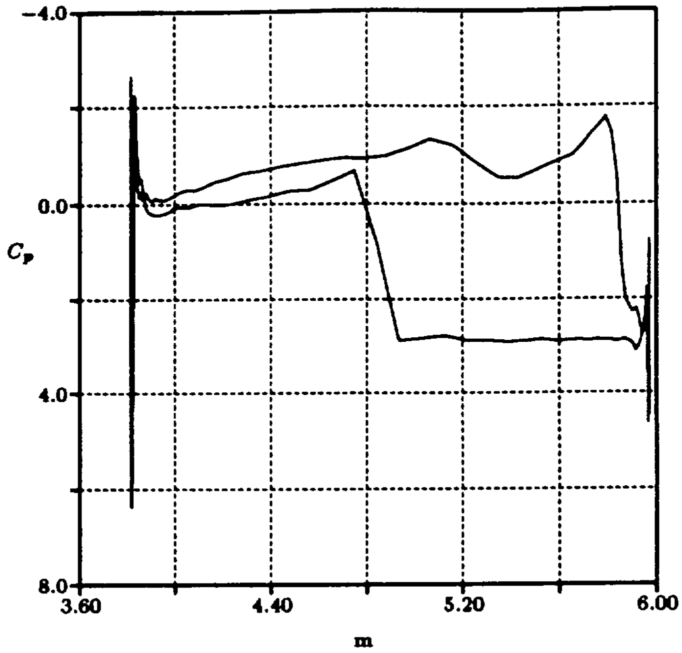


Figure 6.53 NASA Rotor 67 30 % span from shroud surface pressure coefficient: initial mesh.

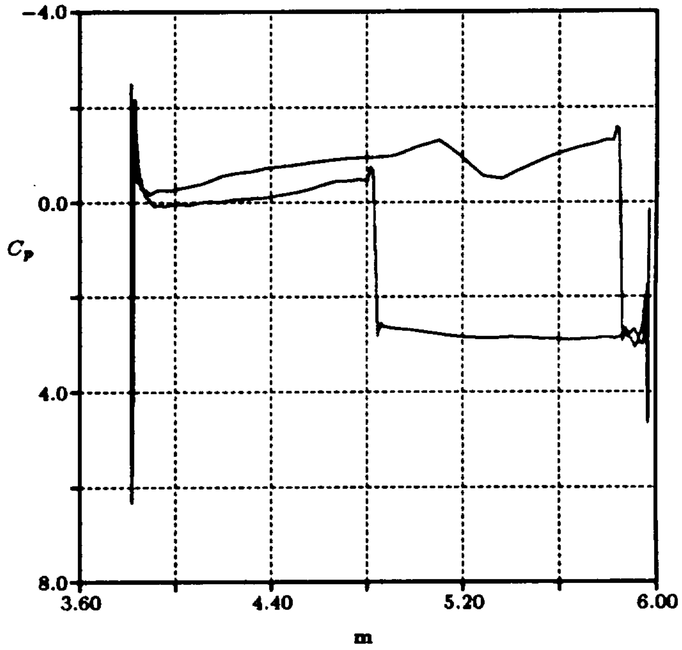


Figure 6.54 NASA Rotor 67 30 % span from shroud surface pressure coefficient: final mesh.

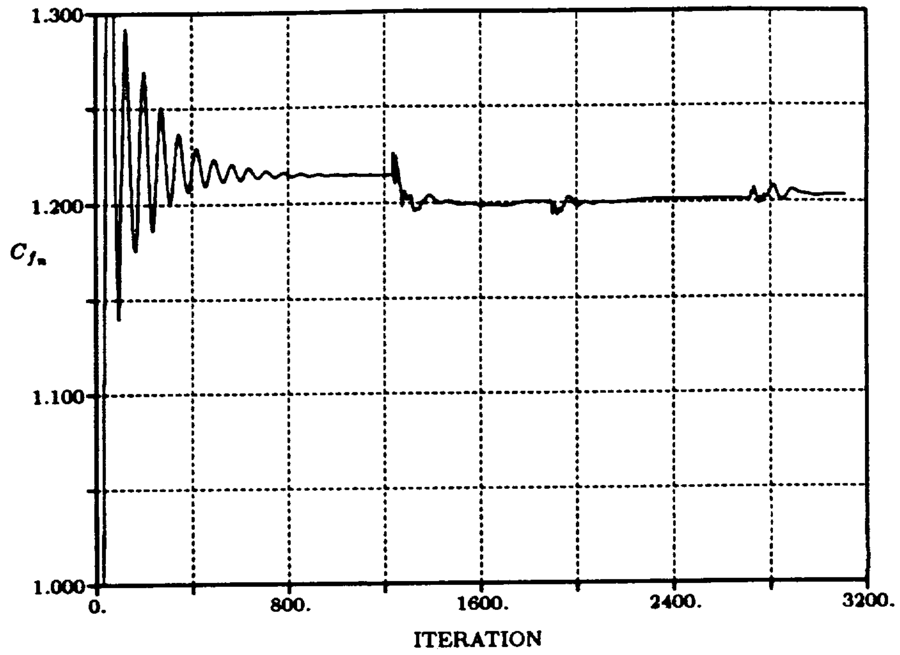


Figure 6.55 NASA Rotor 67 30 % span from shroud: C_{f_n} verses iteration.

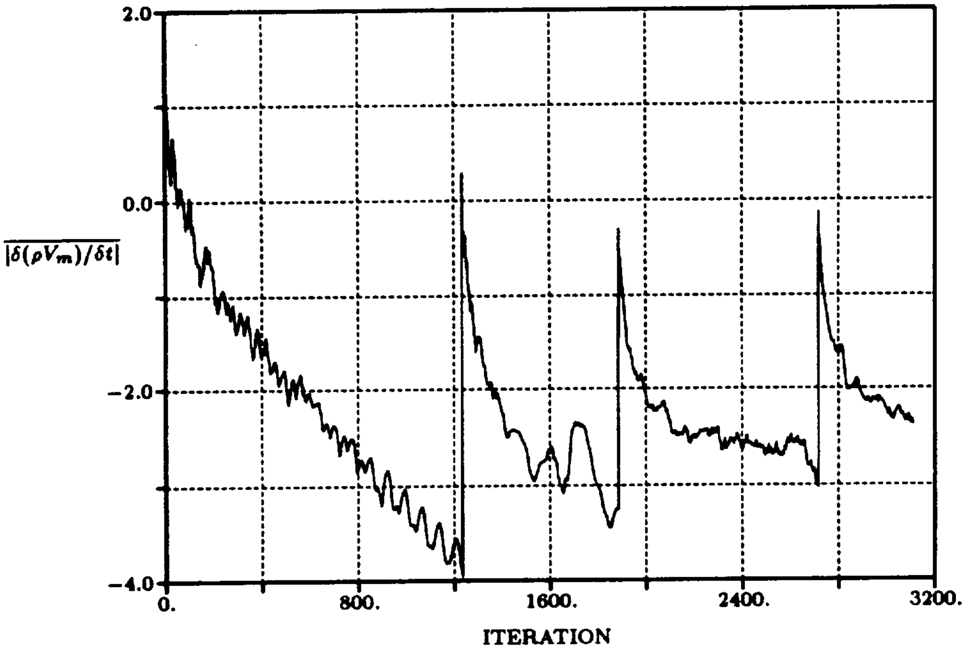


Figure 6.56 NASA Rotor 67 30 % span from shroud: average $|\delta(\rho V_m)/\delta t|$ verses iteration.

6.6 Allison Tandem Blade Cascade

The Allison tandem blade cascade is an axial-flow compressor stator configuration. This is a preliminary design and there are no experimental data available for this case. This case demonstrates the ease with which arbitrary multiply-connected regions can be solved. Quasi-three-dimensional through-flow information is used in the present quasi-three-dimensional analysis. The streamsurface location and thickness data used in the present calculation are obtained from Reference [60].

The adaptive mesh solution is computed for the 70 %-span streamsurface. The far field and near field meshes are shown in Figures 6.57 and 6.58, respectively. In this case the far field boundary for the mesh is placed one axial chord from the blade edges. Numerical boundary conditions are the same as those used for the NASA Rotor 67 case. Uniform initial flow conditions are used. The inflow and outflow conditions determined from the through-flow analysis are summarized in Table 6.11.

Table 6.11 Allison tandem blade cascade far field flow conditions.

Case	$P_{t_{in}}$(psi)	$T_{t_{in}}$ (R)	$r\,v_\theta$	$p_{exit}/P_{t_{in}}$
30 % span	74.5	872.7°	0	0.791
50 % span	74.5	872.7°	0	0.793
70 % span	74.5	872.7°	0	0.796

The remesh sequence is shown in Figures 6.57 to 6.60. These plots illustrate that flow problems of arbitrary multiply-connected regions are no more difficult to solve than flow over a single blade. Mesh statistics and loading coefficient information for this case are given in Table 6.12. This case takes two adaptation cycles to converge. Mach number contours for the final solution are shown in Figure 6.61. The inviscid solution shows shock waves in both leading edge regions. This is due to the rapid acceleration of the flow around a small finite radius leading edge. Viscous effects will slow the acceleration and may even result in a laminar leading edge separation bubble. These shock waves will not be present in a viscous flow calculation. Figures 6.62 to 6.65 show a blowup of the leading edge region and the gap region between the tandem blades. These plots show a smooth variation of mesh length scale in the gap region. This kind of mesh resolution is very difficult, if not impossible, to achieve using a structured mesh flow solver. Figures 6.66 and 6.67 show velocity vectors for the leading edge and the gap regions, respectively. Due to the high flow incidence relative to the second blade there is a rapid acceleration of the flow through the gap region.

The plots of the surface pressure coefficient are shown in Figures 6.68 and 6.69 for the initial and final meshes, respectively. The spikes in the leading edge regions are caused by the rapid acceleration of the flow. Convergence histories are shown in Figures 6.70 and 6.71. The complete procedure converged in 3,000 iterations.

Table 6.12 Allison tandem blade cascade: 70 % span station.

MESH	0	1	2
nodes	1,822	2,415	4,361
elements	3,292	4,445	8,279
C_{f_n}	0.8141	0.7802	0.7885
C_{f_t}	-0.3674	-0.3689	-0.3699
$\delta_{\mathbf{max}}$/pitch	0.096	0.096	0.096
$\delta_{\mathbf{max}}/\delta_{\mathbf{min}}$	1	4	16
Total CPU: 478 sec in Cray-YMP.			

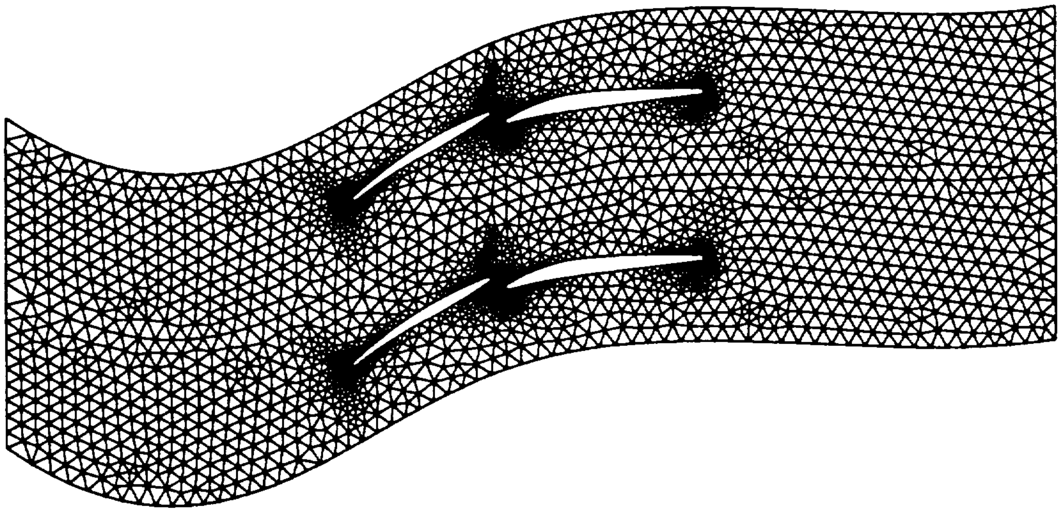


Figure 6.57 Allison tandem blade cascade 70 % span from shroud initial mesh: far field view.

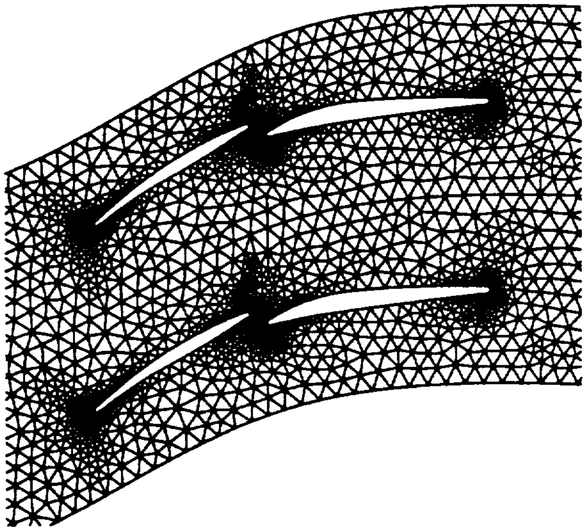


Figure 6.58 Allison tandem blade cascade 70 % span from shroud initial mesh: 1,822 nodes and 3,292 elements.

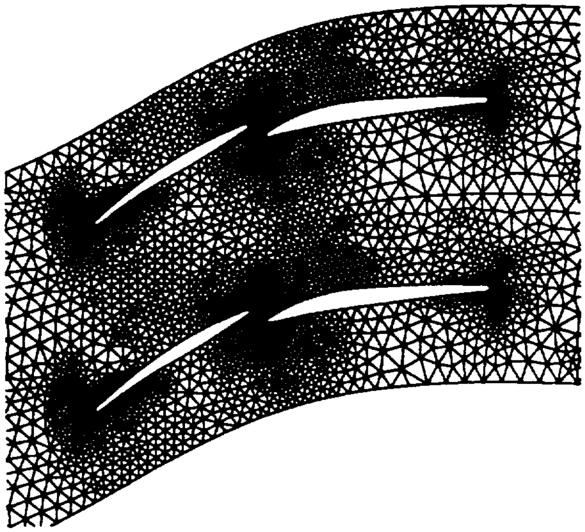


Figure 6.59 Allison tandem blade cascade 70 % span from shroud 1st adapted mesh: 2,415 nodes and 4,445 elements.

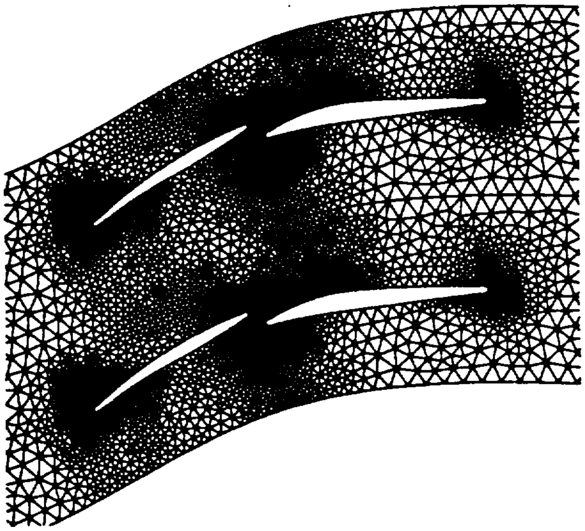


Figure 6.60 Allison tandem blade cascade 70 % span from shroud 2nd (final) adapted mesh: 4,361 nodes and 8,279 elements.

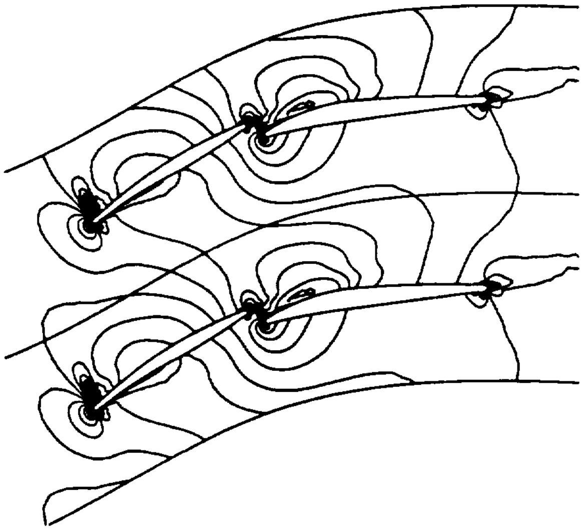


Figure 6.61 Allison tandem blade cascade 70 % span from shroud Mach number contours on the final mesh: cmin = 0.0, cmax = 1.60, and inc = 0.05.

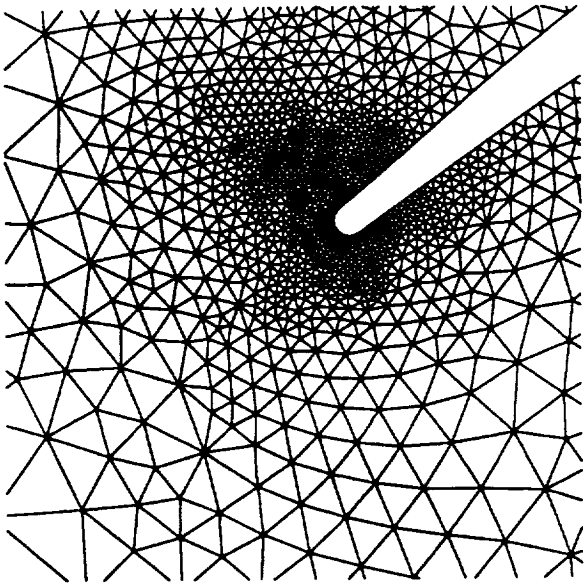


Figure 6.62 Allison tandem blade cascade 70 % span from shroud: leading edge blowup of the final mesh.

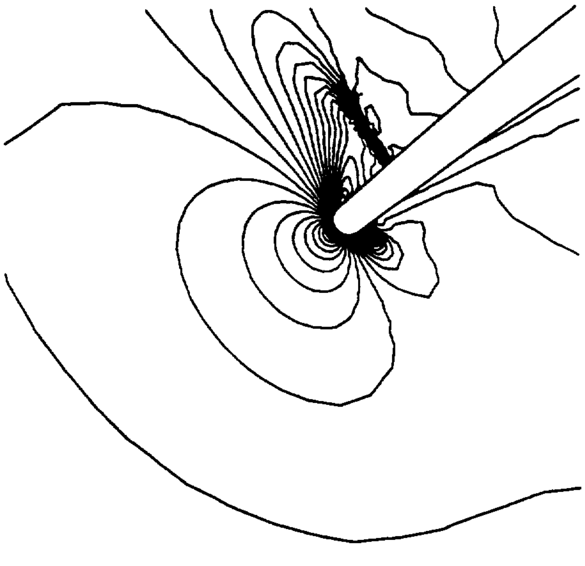


Figure 6.63 Allison tandem blade cascade 70 % span from shroud Mach number contours (cmin =0.0, cmax= 1.60, and inc = 0.05): final mesh leading edge region.

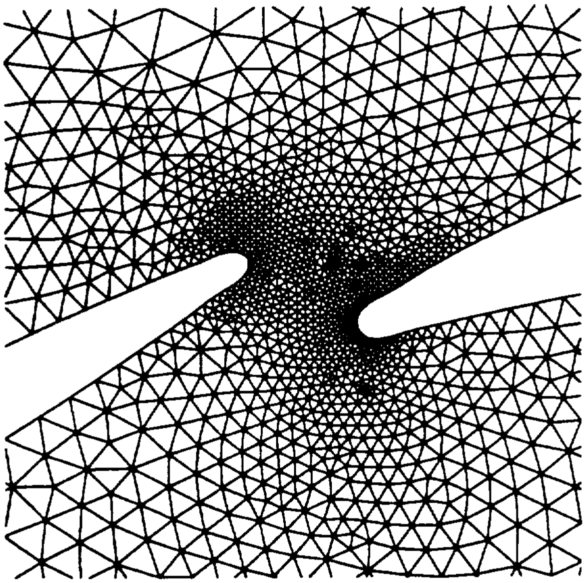


Figure 6.64 Allison tandem blade cascade 70 % span from shroud: tandem blade gap blowup of the final mesh.

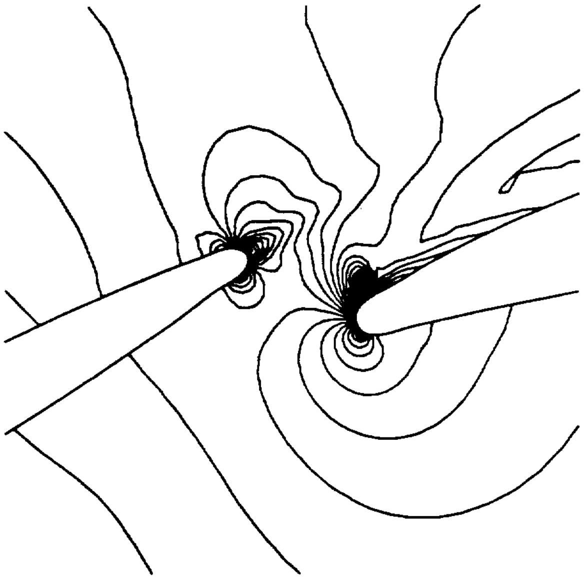


Figure 6.65 Allison tandem blade cascade 70 % span from shroud Mach number contours (cmin = 0.0, cmax = 1.60, and inc = 0.05): final mesh tandem blade gap region.

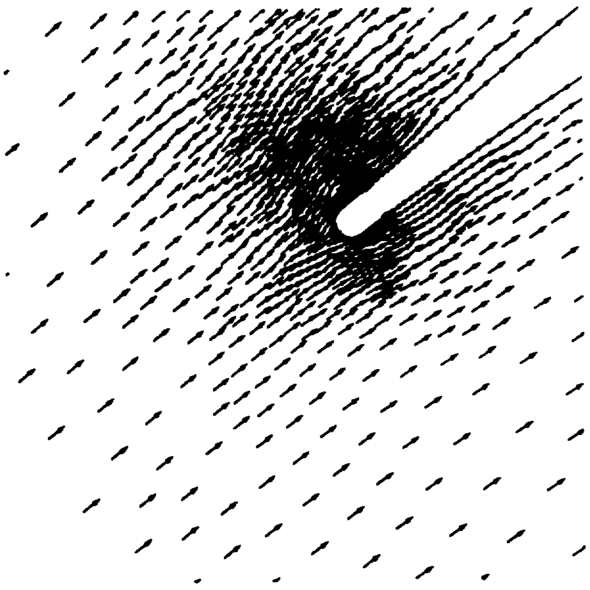


Figure 6.66 Allison tandem blade cascade 70 % span from shroud velocity vector: final mesh leading edge region.

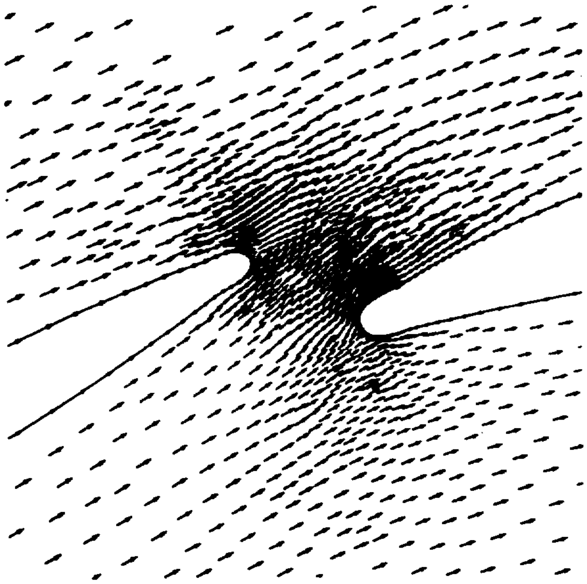


Figure 6.67 Allison tandem blade cascade 70 % span from shroud velocity vector: final mesh tandem blade gap region.

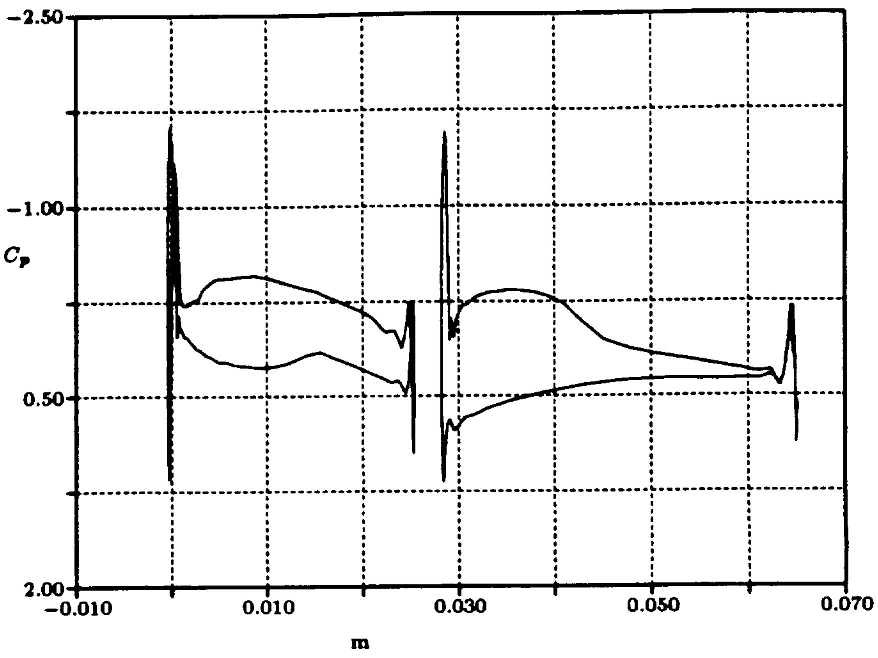


Figure 6.68 Allison tandem blade cascade 70 % span from shroud surface pressure coefficient: initial mesh.

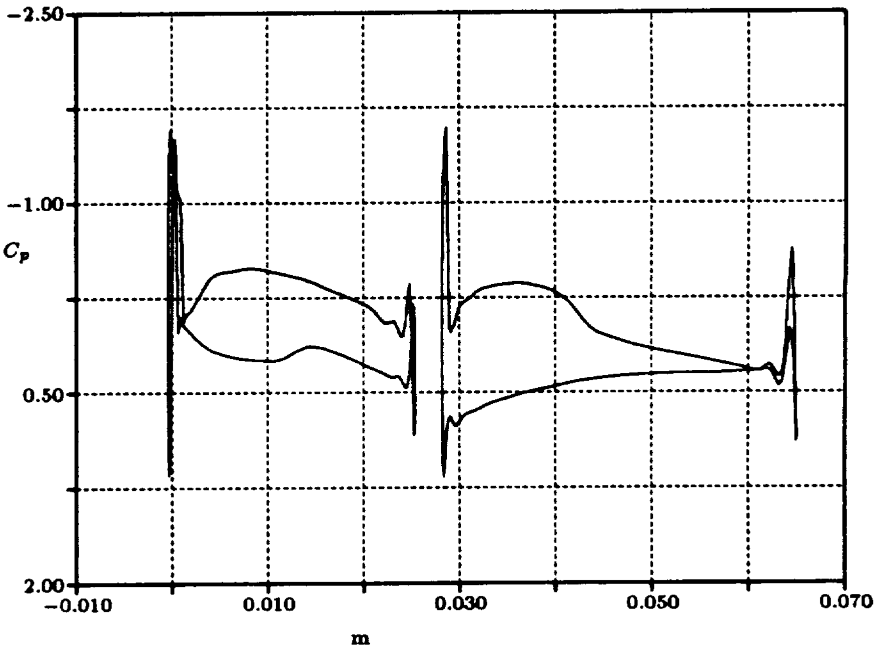


Figure 6.69 Allison tandem blade cascade 70 % span from shroud surface pressure coefficient: final mesh.

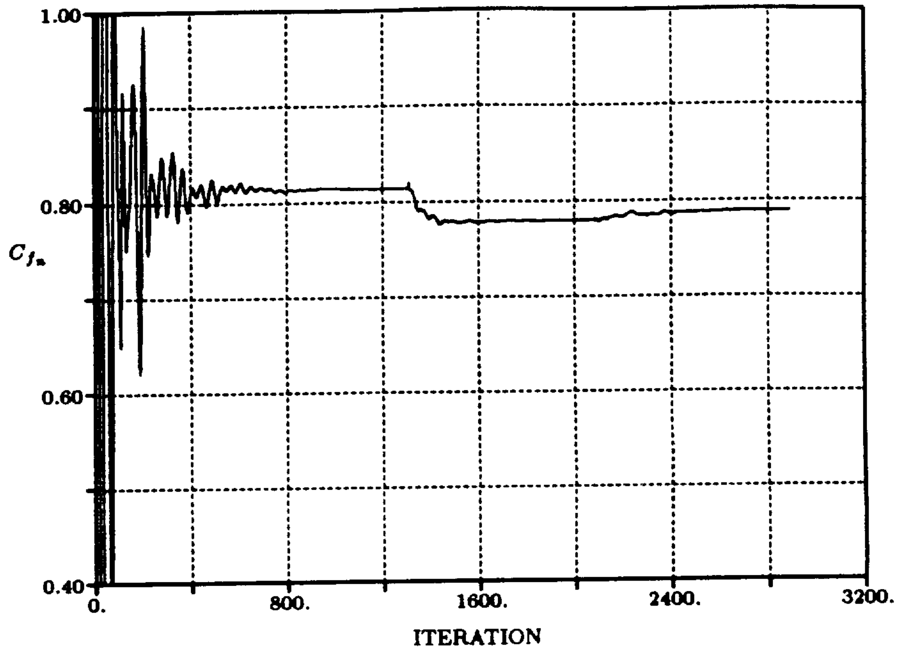


Figure 6.70 Allison tandem blade cascade 70 % span from shroud: C_{f_n} verses iteration.

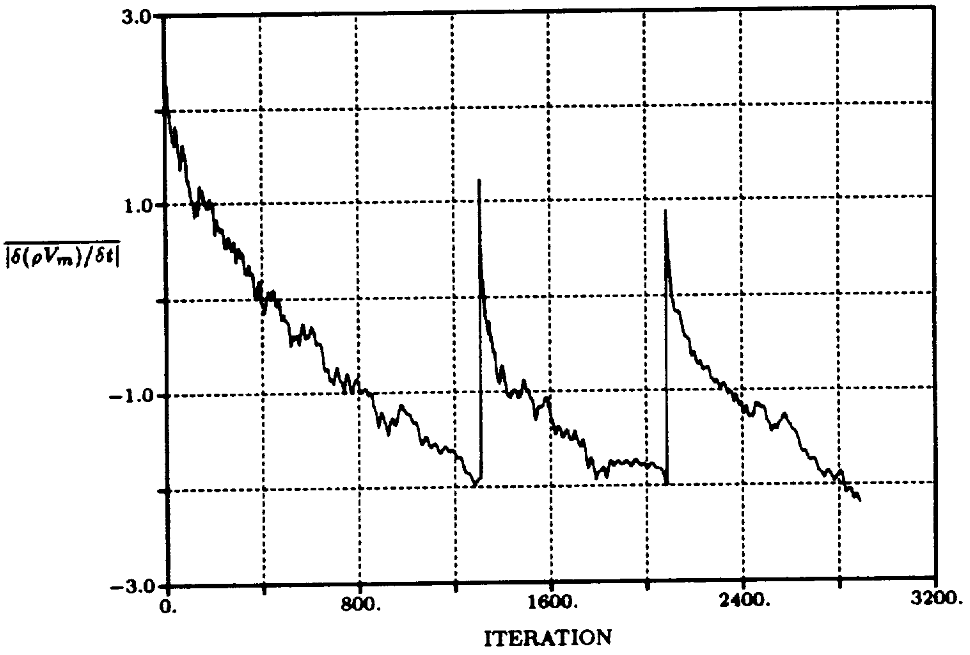


Figure 6.71 Allison tandem blade cascade 70 % span from shroud: average $|\delta(\rho V_m)/\delta t|$ verses iteration.

7. TURBULENT FLOW RESULTS

In this chapter the solution adaptive scheme is applied to turbulent flow problems. The following turbulent flow problems are considered: a subsonic flat plate; a RAE2822 airfoil; a NLR two-element airfoil; the VKI LS82-1 turbine cascade, and the Allison tandem blade cascade. Since experimental data for the first four cases is available, these test cases allow a validation of the present solution adaptive scheme. Numerical results for the Allison tandem blade cascade are presented to demonstrate the advantage of an adaptive unstructured turbulent flow solver for new advanced turbomachinery blade designs. The objective here is to illustrate the capability of the present solution adaptive scheme in resolving complex flow structures. Thus, the convergence criteria are set at a higher level to reduce the computing work.

7.1 Subsonic Flat Plate

The subsonic flat plate flow case is a standard test case for the validation of turbulent flow calculations. Because all the turbulence models were calibrated using the flat plate solution, an accurate prediction of this flow problem indicates a correct implementation of the turbulence model equations. The purpose of this test case here is to validate the numerical implementation of the present turbulent flow solver, therefore, adaptive remeshing is not performed. The triangular mesh used to compute the subsonic flat plate turbulent boundary layer solution is generated from a 72x80 structured quadrilateral mesh. The mesh, shown in Figure 7.1, is algebraically packed near the leading edge and solid surface with minimum increments of $\Delta x_{\min} = 0.0003$ and $\Delta y_{\min} = 0.000003$. The maximum aspect ratio in this mesh is about 40,000. The inlet boundary is placed at one half of the plate length from the leading edge and the exit boundary is located at the end of flat plate. Numerical boundary conditions for the inlet and exit boundaries are the same as those used in the inviscid cascade cases. The freestream static pressure is prescribed along the exit plane. A no-slip condition with adiabatic wall and zero normal pressure gradient are imposed along the solid surface. Flow tangency is enforced along the symmetric boundary which lies along the lower boundary upstream of the plate. The computed results are compared to the experimental data given in the 1968 AFOSR-IFP-Stanford Conference [89] for the Bladwin-Barth one-equation and the Chien $k - \epsilon$ turbulence models. The flow conditions for this case are given in Table 7.1.

In structured mesh approaches, it has been found that the accuracy of the viscous solutions is very sensitive to the effect of artificial dissipation. In order to determine a proper scaling of the artificial dissipation for the present unstructured flow solver, three different scaling of the artificial dissipation terms are investigated. The first scaling is the standard modified eigenvalue scaling, Eq. (3.22). The second scaling combines a local velocity scaling with the modified eigenvalue scaling (see Eq. (3.24)). Note that a local velocity scaling based on q^4 was also investigated in the present work. However, this scaling does not produce a stable solution. In the last scaling, the artificial dissipation is switched off if the local velocity is less than a specified cut-off flow speed, U_{cut}. This scaling is a modified velocity scaling which is described as

$$\lambda_{ji} = Q\,\lambda_s \left(1 + \left(\frac{\lambda_n}{\lambda_s}\right)^{\alpha}\right) \tag{7.1}$$

Table 7.1 Flow conditions for the subsonic flat plate.

M_∞	α	p_∞ (psi)	T_∞ (° K)	Re_L	L_{ref} (m)
0.25	0.0°	14.7	293.0	2.2x10^7	10
Note: Experimental data are available in Reference [89].					

where

$$Q = \min\left\{1, \max\left\{0, (q^2 - U_{\mathrm{cut}}^2)/(U_\infty^2 - U_{\mathrm{cut}}^2)\right\}\right\}$$

In the present calculation U_{cut} is taken as 50 % of freestream velocity. This scaling does not guarantee a stable solution. In the present work, the scaling, Eq. (7.1), is only used for the assessment of the accuracy of the turbulent flat plate flow solution. In order to achieve a stable solution, this scaling is used after the flat plate solution is obtained with the local velocity scaling, Eq. (3.24).

Figures 7.2 and 7.3 present the log-law velocity profile predicted by the Baldwin-Barth one-equation and the Chien low Reynolds number k $-$ ϵ turbulence models at Re_x = 1.13x10^7. These plots clearly show that the eigenvalue scaling does not produce a satisfactory log-law velocity profile. With local velocity scaling the results are great improved. Specifying zero artificial dissipation in the viscous layer further improves the predicted results for both turbulence models. Figures 7.4 and 7.5 show the nondimensional velocity profile predictions for each model at x=4.987 m. The nondimensional velocity is defined as u/U_∞, the ratio of local velocity to the freestream velocity. Results are compared to experimental data. Results for the velocity scaling and the switch-off approach agree quite well with experimental data. Figures 7.6 and 7.7 present the coefficient of friction prediction along the plate for both turbulence models. Again the local velocity scaling results are better than the modified eigenvalue scaling. Without adding artificial dissipation in the viscous layer the friction coefficient distribution is accurately predicted.

In the present turbulent flat plate flow case specifying zero artificial dissipation in the viscous layer produces an accurate prediction. However, for general flow problems such an approach can lead to numerical stability problems. In the remaining cases the local velocity scaling artificial dissipation is used. Both Baldwin-Barth one-equation and Chien low Reynolds number k $-$ ϵ turbulence models have been successfully demonstrated on the turbulent flat plate flow. It has been recognized that Chien's low Reynolds number k$-\epsilon$ model requires a finer wall spacing ($y^+ \approx 1$) than the Baldwin-Barth one-equation model ($y^+ \approx 3.5$). In addition, for the low

Reynolds number $k - \epsilon$ model it is very difficult to obtain a stable solution without a proper initialization of the turbulence quantities. In structured mesh schemes k and ϵ are usually initialized using an algebraic turbulence model solution [26]. However, such an initialization is not practical in an unstructured mesh scheme. Since the Bladwin-Barth one-equation is more robust and requires less mesh resolution in the near-wall regions, only the Baldwin-Barth one-equation model is used in the remaining turbulent flow problems.

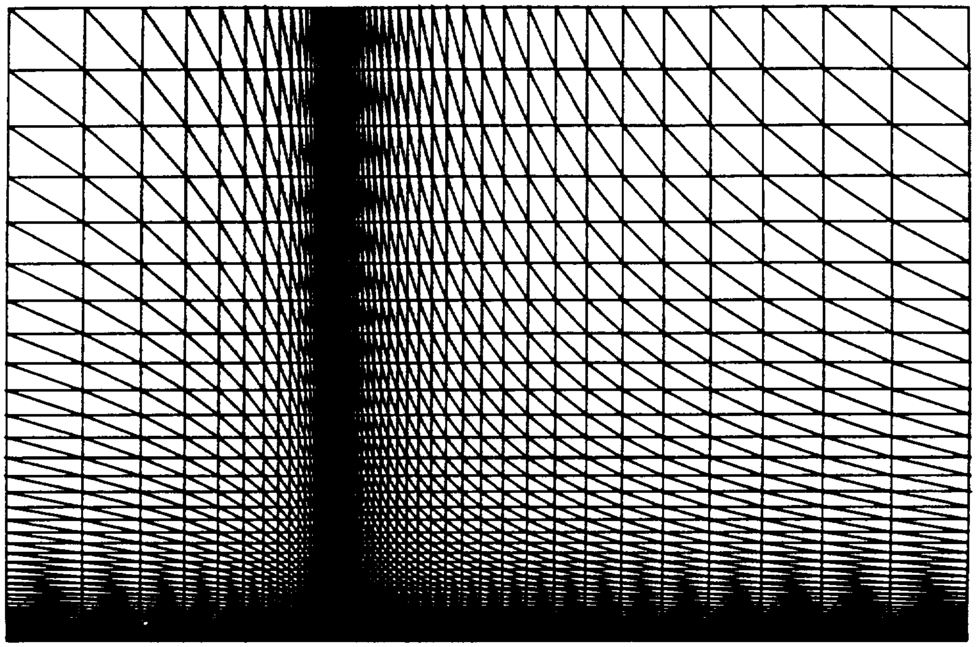


Figure 7.1 Triangular mesh for flat plate flow case: 5,760 nodes and 11,218 elements (10:1 magnification in y-direction).

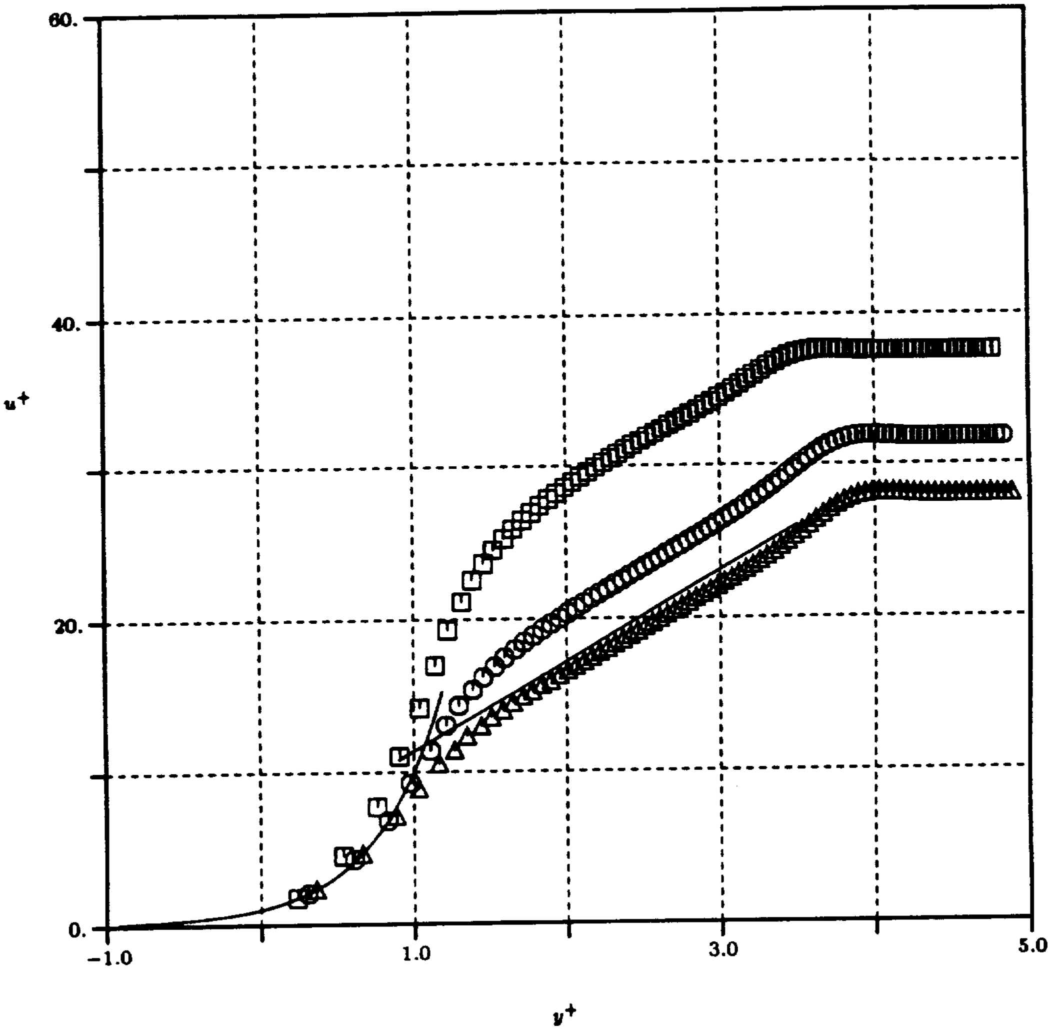


Figure 7.2 Flat plate log-law velocity profile at $Re_x = 1.13\text{x}10^7$ obtained with Baldwin-Barth one-equation turbulence model for different artificial dissipations: □ - eigenvalue scaling, ⊙ - local velocity scaling, △ - zero artificial dissipaton in the boundary layer and solid line - standard log-law profile
$u^+ = 5.85 \log_{10} y^+ + 5.56$.

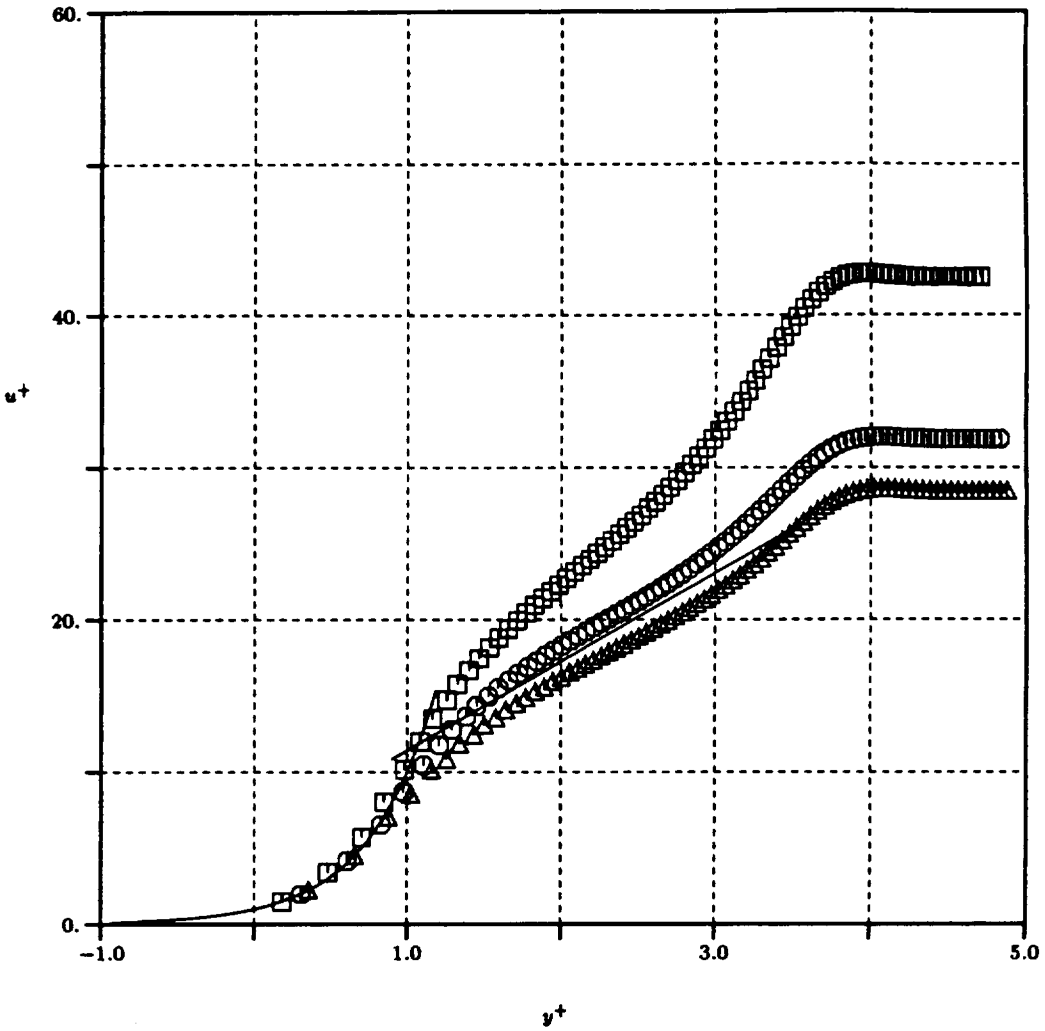


Figure 7.3 Flat plate log-law velocity profile at $Re_x = 1.13\text{x}10^7$ obtained with Chien's low Reynolds number $\kappa - \epsilon$ turbulence model for different artificial dissipations: □ - eigenvalue scaling, ⊙ - local velocity scaling, △ - zero artificial dissipaton in the boundary layer and solid line - standard log-law profile $u^+ = 5.85 \log_{10} y^+ + 5.56$.

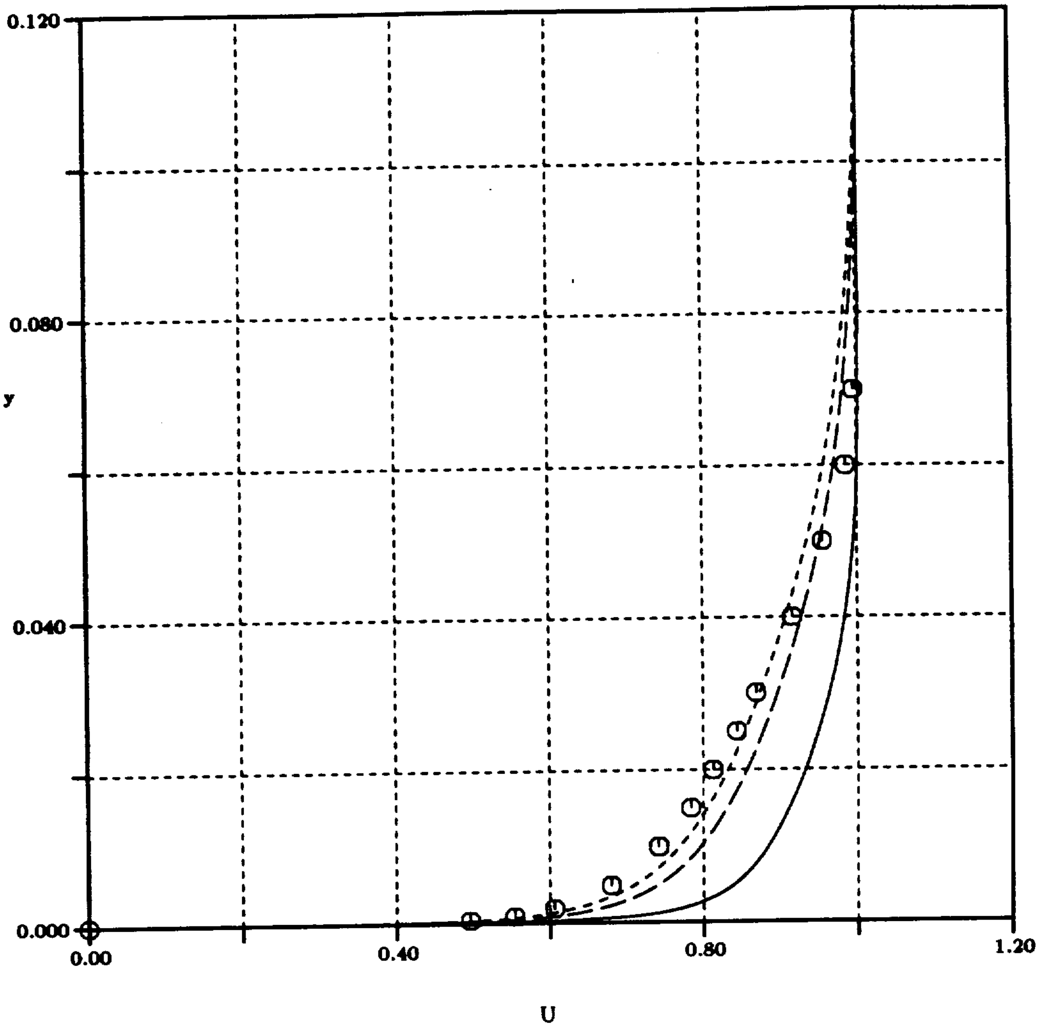


Figure 7.4 Flat plate nondimensional velocity profile at x=4.987 m obtained with Baldwin-Barth one-equation turbulence model for different artificial dissipations: solid line - eigenvalue scaling, dashed line - local velocity scaling, short dashed line - zero artificial dissipaton in the boundary layer and symbol - experimental data.

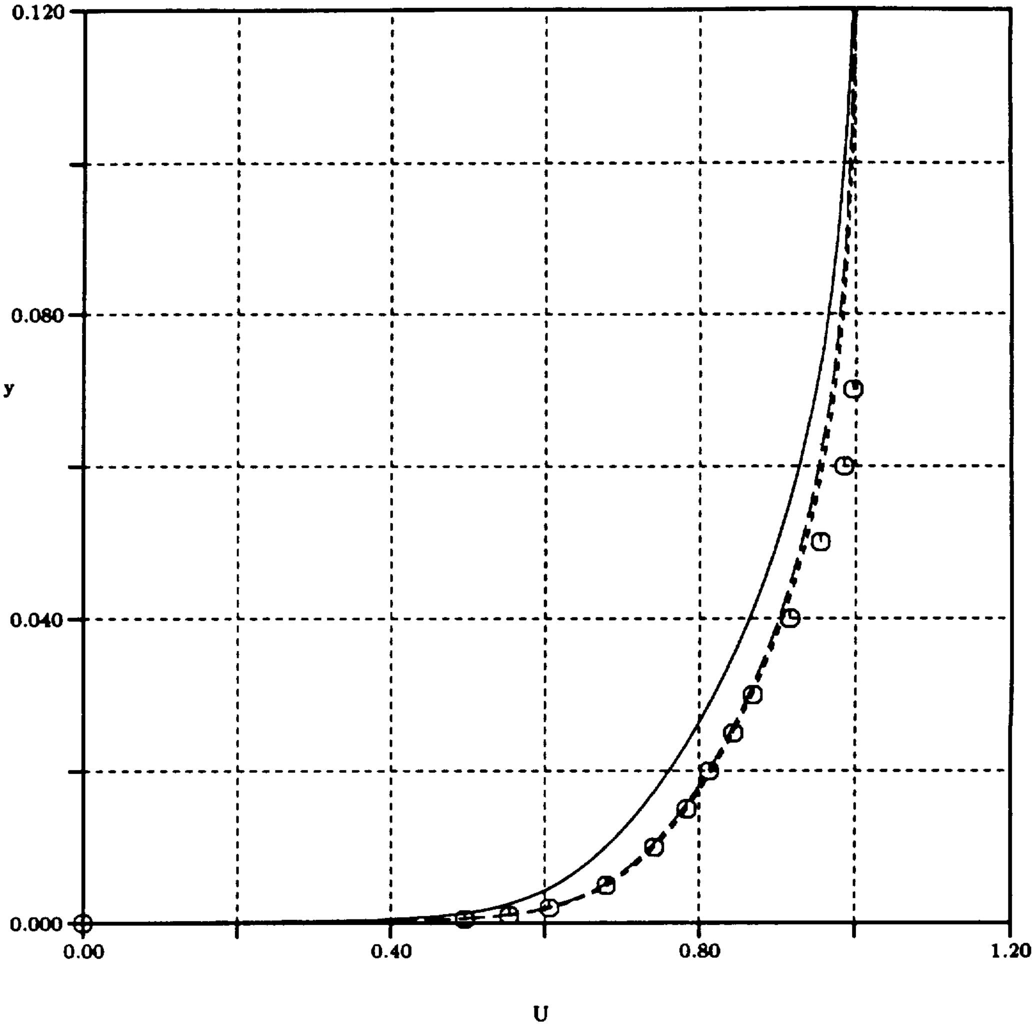


Figure 7.5 Flat plate nondimensional velocity profile at x=4.987 m obtained with Chien's low Reynolds number $\kappa - \epsilon$ turbulence model for different artificial dissipations: solid line - eigenvalue scaling, dashed line - local velocity scaling, short dashed line - zero artificial dissipaton in the boundary layer and symbol - experimental data.

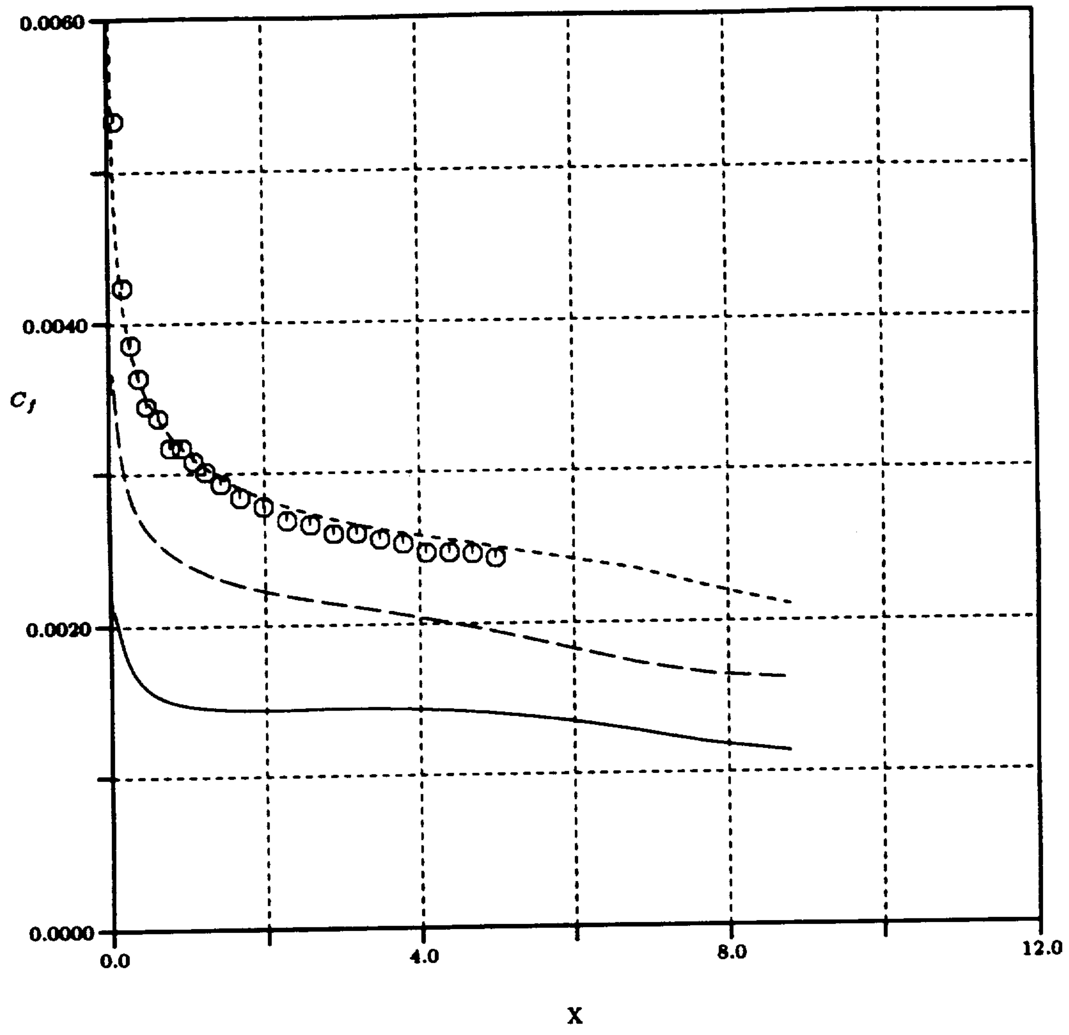


Figure 7.6 Flat plate surface friction coefficient obtained with Baldwin-Barth one-equation turbulence model for different artificial dissipations: solid line - eigenvalue scaling, dashed line - local velocity scaling, short dashed line - zero artificial dissipaton in the boundary layer and symbol - experimental data.

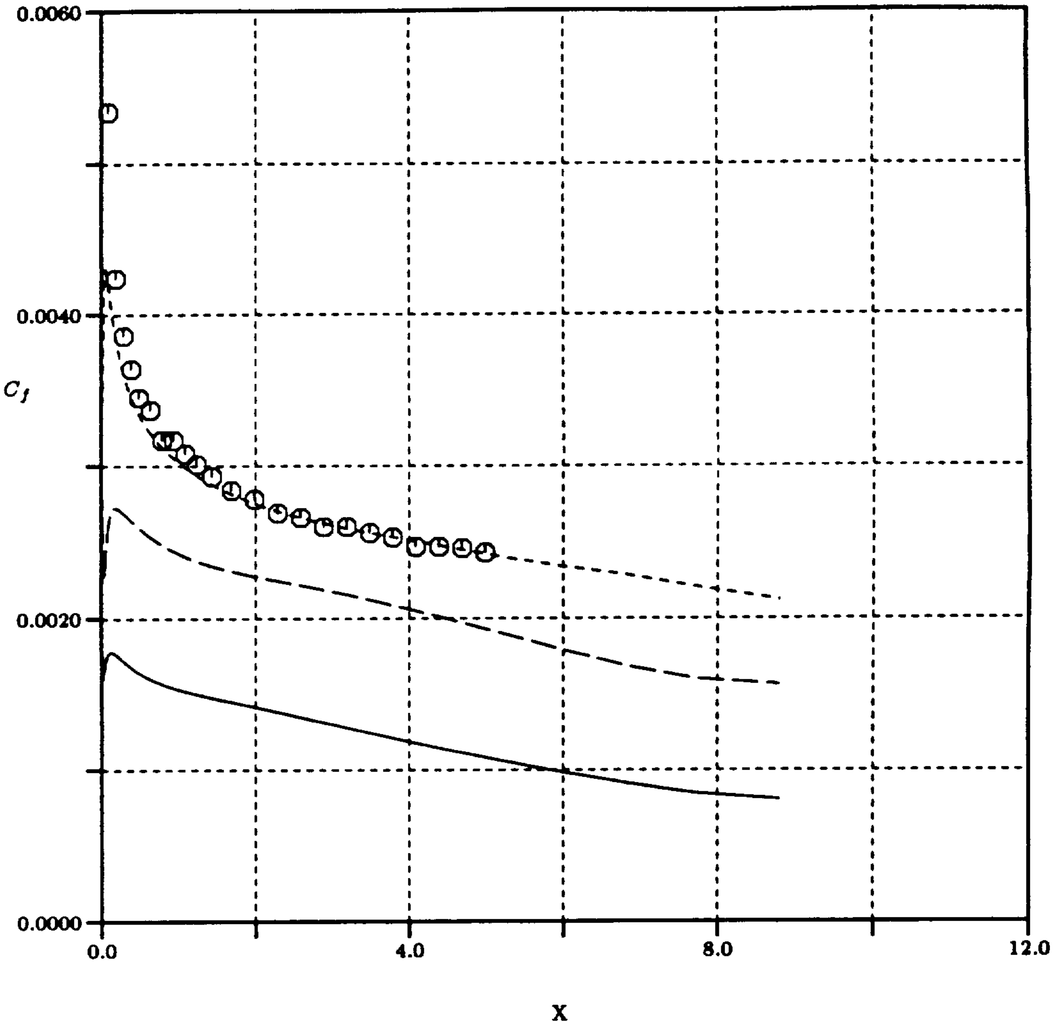


Figure 7.7 Flat plate surface friction coefficient obtained with Chien's low Reynolds number $\kappa - \epsilon$ turbulence model for different artificial dissipations: solid line - eigenvalue scaling, dashed line - local velocity scaling, short dashed line - zero artificial dissipaton in the boundary layer and symbol - experimental data.

7.2 RAE2822 Airfoil

The first test case presented for the solution adaptive scheme is the 1987 VTA Workshop test case B1. This case is a RAE2822 airfoil operating at $M_\infty = 0.725$, $\alpha = 2.92°$, and $Re_L = 6.5\text{x}10^6$. This flow involves a strong shock wave at 55 % chord on the upper surface. The lift coefficient in this case depends strongly on the predicted shock wave location. This makes a good resolution of the shock wave very important. Experimental data is given by Cook, McDonald, and Firmin [20]. Note that a transition trip was used near the leading edge suction surface in the experiment. The flow conditions used for the present calculation are listed in Table 7.2. For this test case, each participant of the 1987 VTA Workshop used a different corrected angle of attack in his numerical calculations to obtain a better agreement with experimental data. The range of corrected angles varied from 2.3 to 2.92 in the workshop [32]. Since the Baldwin-Barth turbulence model is used in the present work, the corrected angle of attack, $\alpha_{\text{comp}} = 2.31°$, suggested by Barth [7] is used. The far field and near field meshes are presented in Figures 7.8 and 7.9, respectively. The far field boundary is set at a radius of 5 chords from the airfoil. At this boundary the vortex far field boundary condition [83] is applied. A no-slip adiabatic wall condition and a zero normal pressure gradient are imposed at wall.

The initial local structured mesh is constructed through the specification of a normal wall mesh scale ($\delta_{y^+} = \Delta y \; y^+_{\text{pr}} = 8.6\text{x}10^{-6}$ chord) and a streamwise wall mesh scale ($N_s = 25$ and $\delta_w = A_w \; \delta_{y^+} = 0.0215$ chord). In the far field the unstructured mesh is uniform with a length scale of $\delta_{\max} = 0.5$ chord. The refinement ratio equals 39 for the first adaptation and 2 for every additional refinement. The reason of using a large ratio in the first adaptation is to decrease the minimum length scale to the streamwise wall mesh scale so that the number of adaptation cycles can be reduced. Figures 7.9 to 7.13 present the resulting solution adapted meshes. Mesh statistics and force coefficient information for each mesh are listed in Table 7.3. Note this case takes about 2.5 hours of Cray-YMP CPU time.

Comparing the initial mesh solution with the final mesh solution as shown in Figures 7.14 and 7.15 shows improved shock wave resolution for the adapted solution. Figures 7.16 to 7.21 show detailed mesh and flow structures in the shock wave and trailing edge regions. The present adaptive remeshing scheme does a good job in resolving the shock wave, the viscous wake and the boundary layer growth resulting from the strong shock wave. Figures 7.22 and 7.23 show the comparisons of surface pressure coefficient with the experimental data for the initial and final meshes, respectively. These plots show a slight difference in the leading edge suction surface pressure distribution between the computed and the experimental results. This may be due to the fact that the transition trip was not simulated in the numerical solution. Further, the present solution does not accurately predict the shock wave location. This leads to the 3 % difference between the computed and measure lift coefficient (see Table 7.3). Comparing the initial mesh surface friction coefficient shown in Figure 7.24 with the final mesh surface friction coefficient in Figure 7.25 shows a change of upper and lower surface friction coefficient as the mesh is refined. The change of upper surface friction coefficient is due to the resolution of the shock wave. A possible explanation to the change of lower surface friction coefficient is that the turbulence effect was not fully developed in the initial mesh solution yet.

Since the flow conditions and turbulence model used here are the same as those used by Barth [7], the present solution shown in Figure 7.23 is compared to Barth's solution shown in Figure 7.27. The present solution does not predict the same the shock wave location as Barth. In order to study the viscous effect on the shock

Table 7.2 Flow conditions for the RAE2822 airfoil.

M_∞	α_{meas}	α_{comp}	T_∞ (° K)	Re_L	L_{ref} (m)
0.725	2.92°	2.31°	293	$6.5\mathrm{x}10^6$	0.61
Note: Experimental data is available in Reference [2					

wave location, the inviscid solution for the same flow conditions, $M_\infty = 0.725$ and $\alpha = 2.31$, is also computed. The inviscid surface pressure coefficient is presented as dashed line in Figure 7.26. The inviscid shock wave location is downstream of the experimental location as expected. Note that the artificial dissipation plays an important role in the accuracy of viscous flow solution as shown in the flat plate case. Since the difference between the shock wave locations of the experiment and viscous solution is 4 % of chord upstream, This shift may be due to too much level of artificial dissipation in the viscous flow region. However, comparing the present viscous results with the results of the 1987 Viscous Transonic Airfoil Workshop [32] as shown in Figures 7.23, 7.25, 7.28, and 7.29, the present results are well in the range of the workshop results. This indicates that this is very difficult test case to accurately predict.

Figures 7.30 to 7.33 show comparisons of nondimensional velocity profiles at two stations on the airfoil upper surface. Since it is difficult to interpolate data within unstructured mesh regions, only the nondimensional velocity profiles within the local viscous mesh are shown. The nondimensional velocity is defined as q/U_e, the ratio of the local velocity to the edge velocity. Because it is difficult to define an edge velocity on the unstructured mesh, the edge velocity is estimated using the procedure described in Reference [20], where the velocity ratio U_e/U_∞ is given as:

$$\frac{U_e}{U_\infty} = \frac{M_e}{M_\infty}\left(\frac{1+0.2M_\infty^2}{1+0.2M_e^2}\right)^{1/2} \tag{7.2}$$

with the edge Mach number M_e defined as:

$$M_e = \sqrt{5}\left(\frac{1+0.2M_\infty^2}{(1+0.7M_\infty^2(C_p)_e)^{2/7}} - 1\right)^{1/2} \tag{7.3}$$

where M_∞ is the freestream Mach number and $(C_p)_e$ is the edge pressure coefficient. With the assumption of constant static pressure through the boundary layer, $(C_p)_e$ can be replaced with the surface pressure coefficient. The velocity ratio U_e/U_∞ is then evaluated using the above expressions. The plots show little difference between the initial and final mesh solutions. This is due to the fact that the normal wall mesh scale is nearly constant during adaptation. Therefore, the resolution of viscous region is nearly constant during remeshing.

Convergence histories for the pressure portion of the lift and drag coefficients are shown in Figures 7.34 and 7.35. Figures 7.36 and 7.37 present the convergence histories for the average residuals, $|\delta(\rho u)/\delta t|$ and $|\delta R_t/\delta t|$, respectively. The turbulence quantities take about 1,500 iterations to develop and the complete procedure converged in 8,000 iterations. These plots also show that all adapted mesh solutions do not satisfy the convergence criteria based on the average residual $|\delta(\rho u)/\delta t|$.

Table 7.3 RAE2822 airfoil: $M_\infty = 0.725$, $\alpha = 2.31^\circ$, $Re_L = 6.5\text{x}10^6$, $C_{l_{meas}} = 0.743$, and $C_{d_{meas}} = 0.0127$.

MESH	0	1	2	3	4
nodes	6,568	7,168	8,401	10,420	14,348
elements	12,934	14,133	16,597	20,626	28,462
C_{l_p}	0.7350	0.7209	0.7195	0.7179	0.7177
C_{d_p}	0.0068	0.0061	0.0059	0.0057	0.0056
C_{l_v}	0.0040	0.0037	0.0035	0.0035	0.0034
C_{d_v}	0.00034	0.00032	0.00030	0.00030	0.00030
δ_{max}/chord	0.5	0.5	0.5	0.5	0.5
$\delta_{\text{max}}/\delta_{\text{min}}$	1	39	78	156	312
$\Delta y_{\text{wall}}(\text{x}10^6)$	4.28	3.32	3.69	3.92	4.02
y^+_{pr}	2	2.5	2.5	2.5	2.5
A_w	2,500	2,500	2,500	2,500	2,500
Cray-YMP CPU (sec)	3,012	1,932	1,289	874	1,793

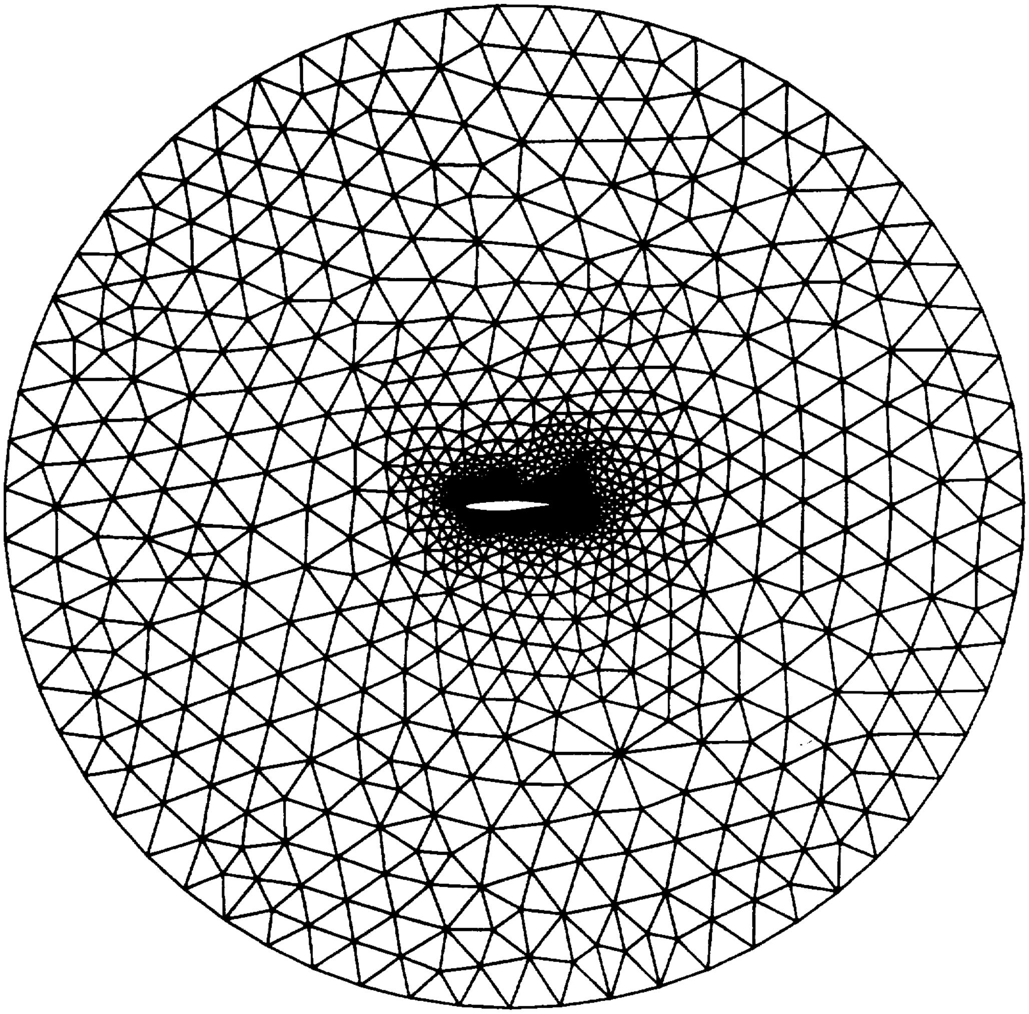


Figure 7.8 RAE2822 airfoil initial mesh: far field view.

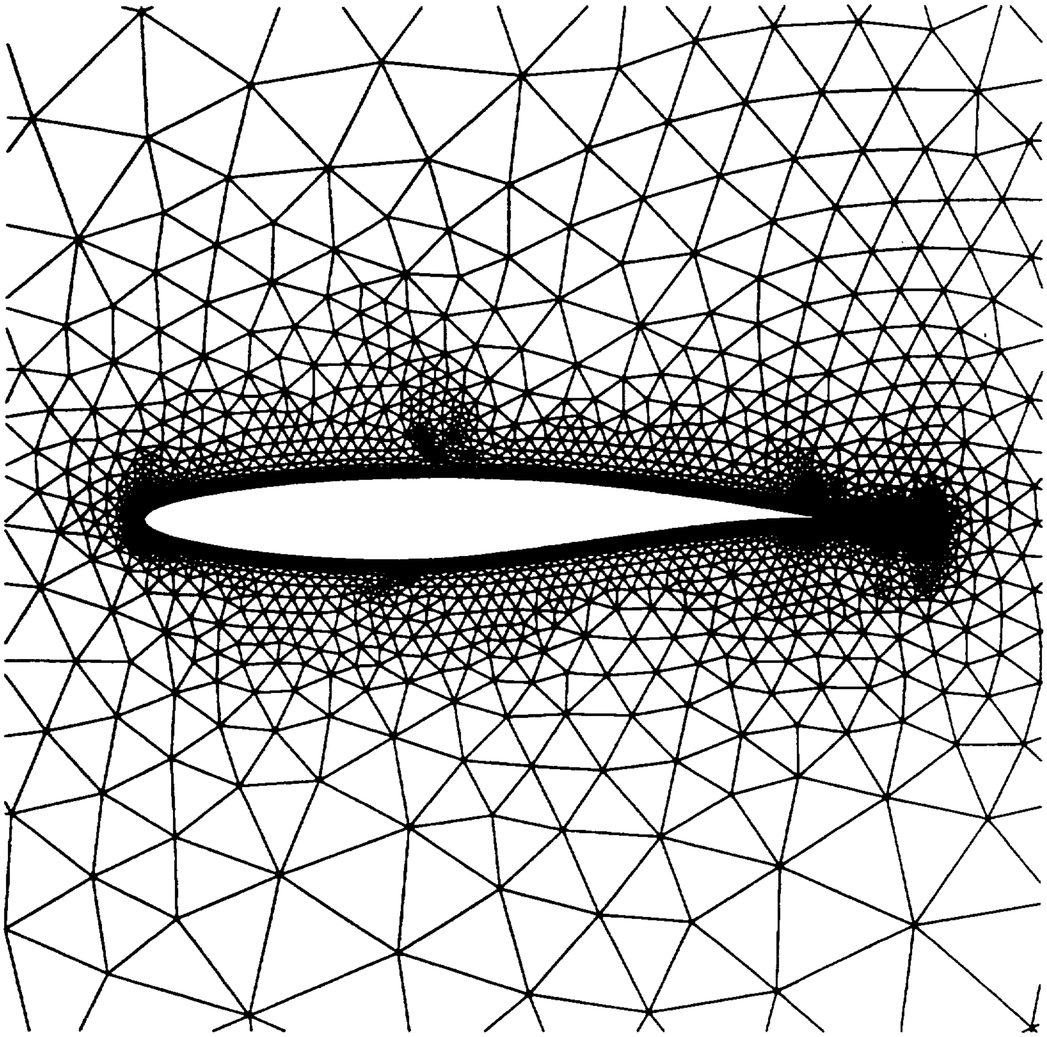


Figure 7.9 RAE2822 airfoil initial mesh: 6,568 nodes and 12,934 elements.

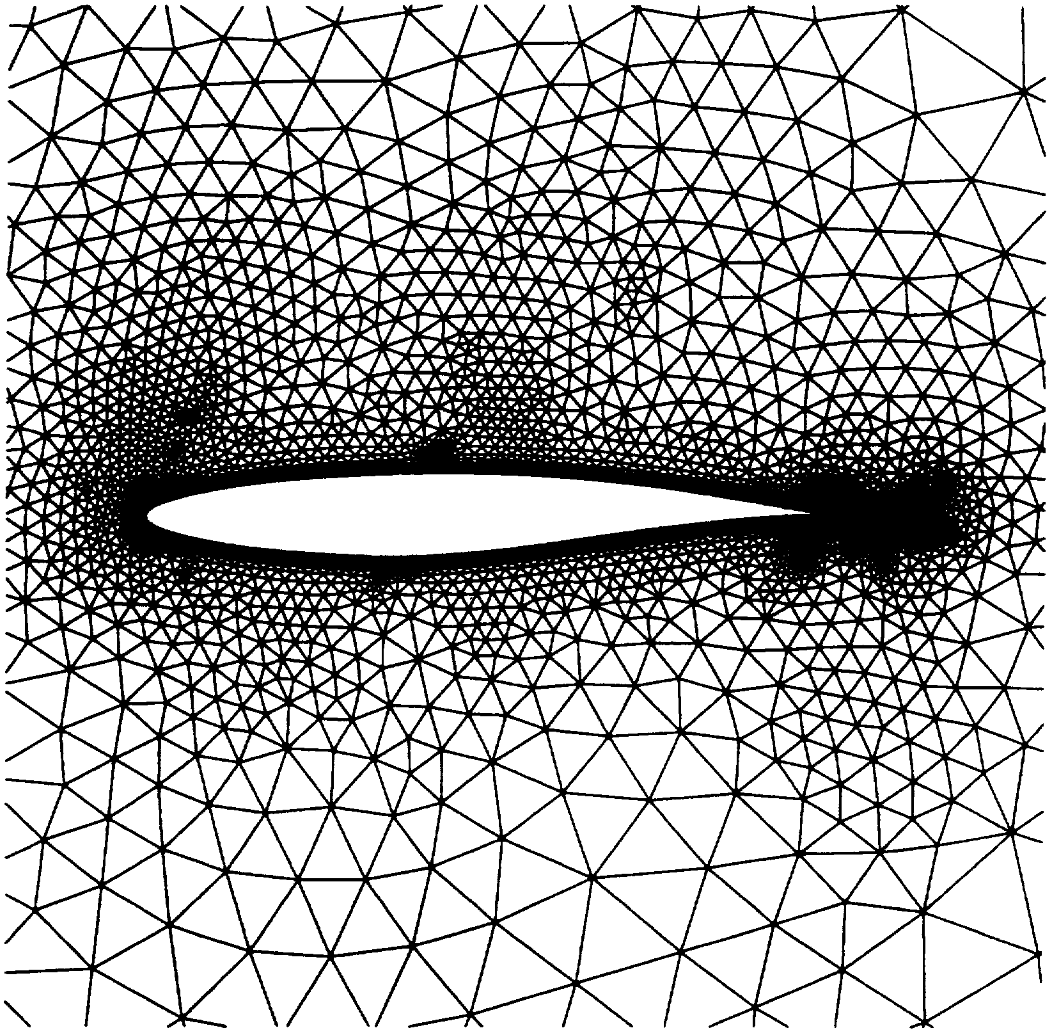


Figure 7.10 RAE2822 airfoil 1st adapted mesh: 7,168 nodes and 14,133 elements.

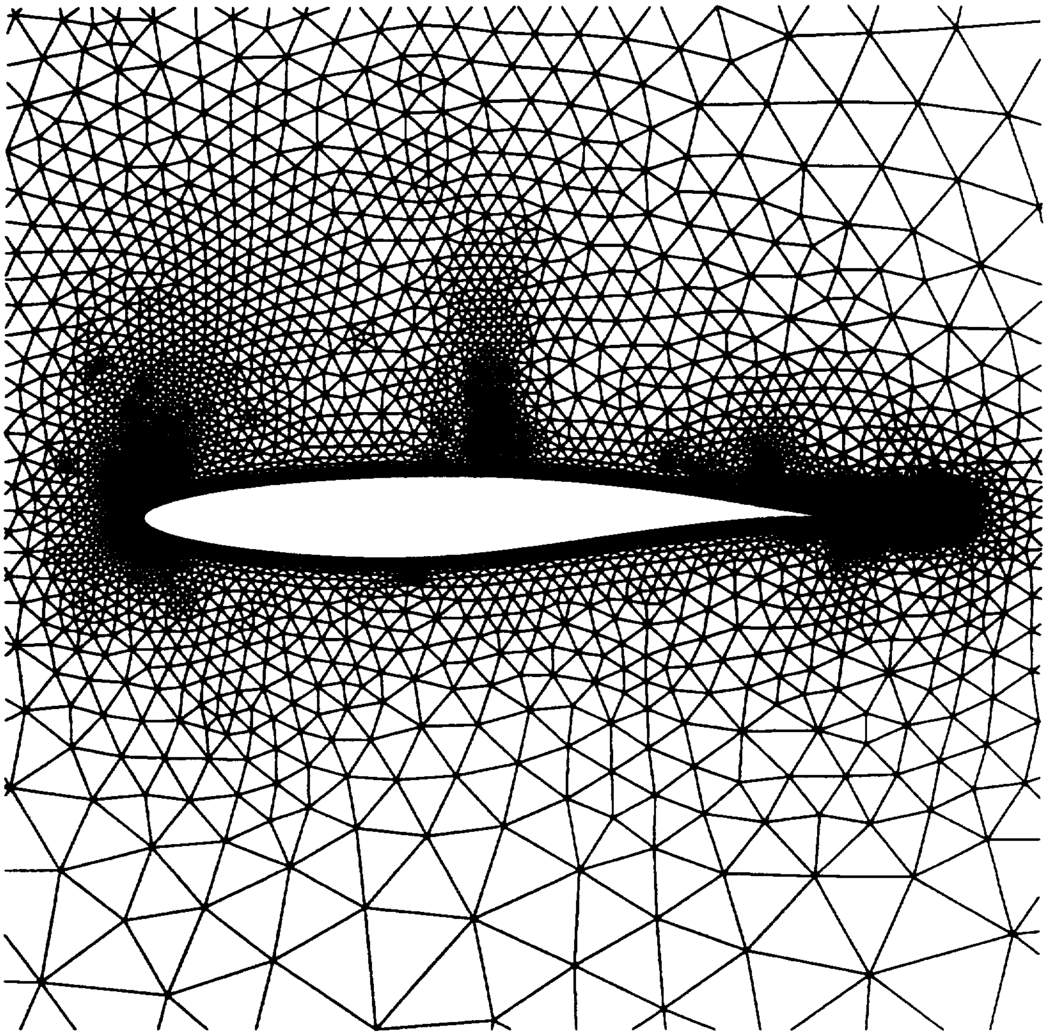


Figure 7.11 RAE2822 airfoil 2nd adapted mesh: 8,401 nodes and 16,597 elements.

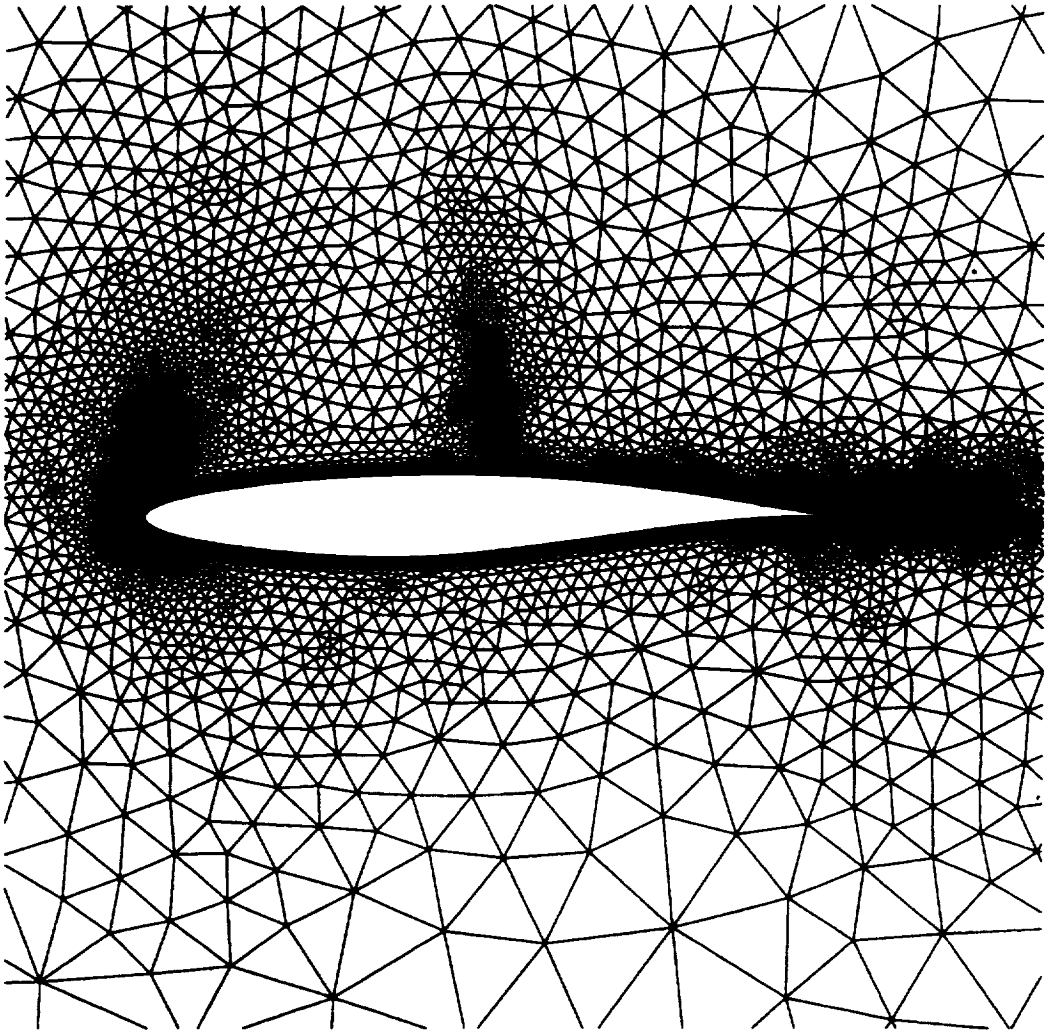


Figure 7.12 RAE2822 airfoil 3rd adapted mesh: 10,420 nodes and 20,626 elements.

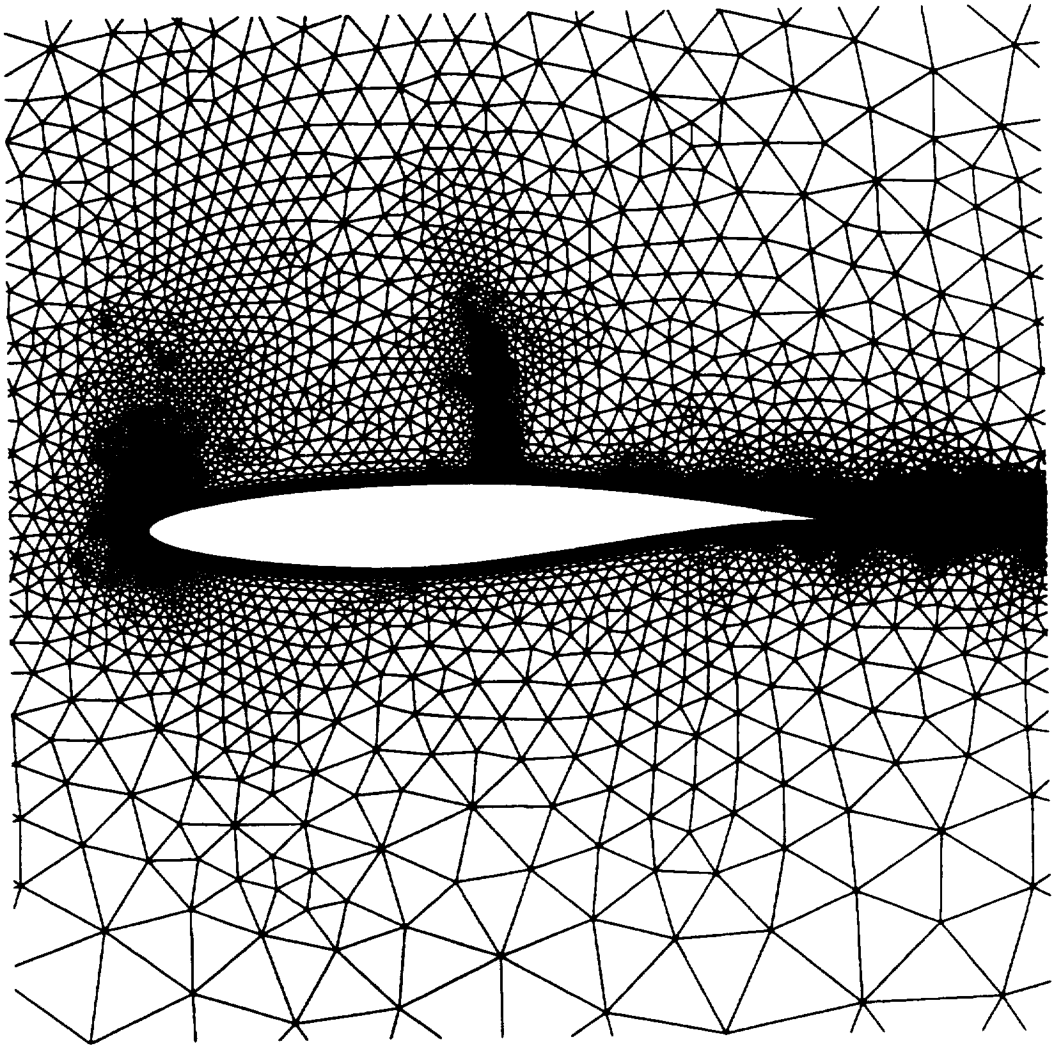


Figure 7.13 RAE2822 airfoil 4th (final) adapted mesh: 14,348 nodes and 28,462 elements.

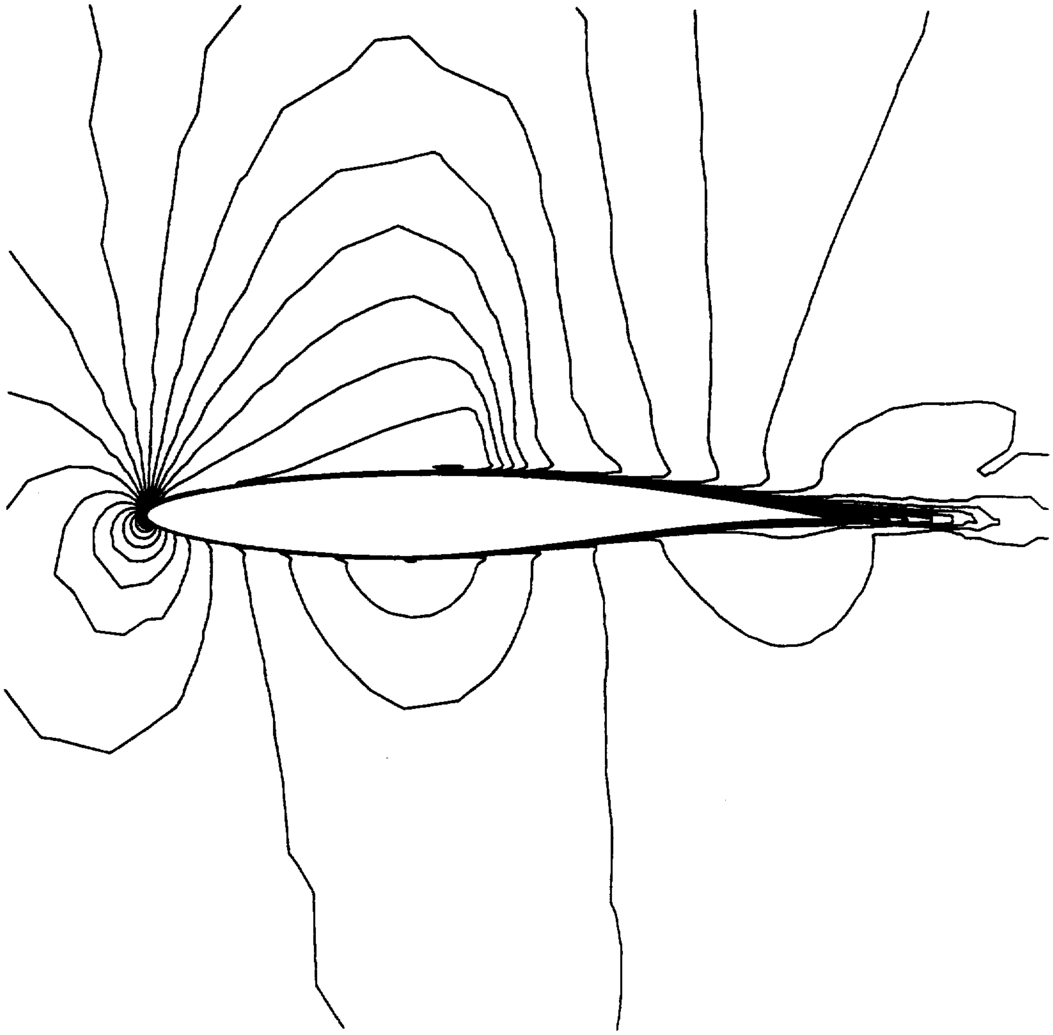


Figure 7.14 RAE2822 airfoil Mach number contours on the initial mesh: cmin = 0.0, cmax = 1.20, and inc = 0.05.

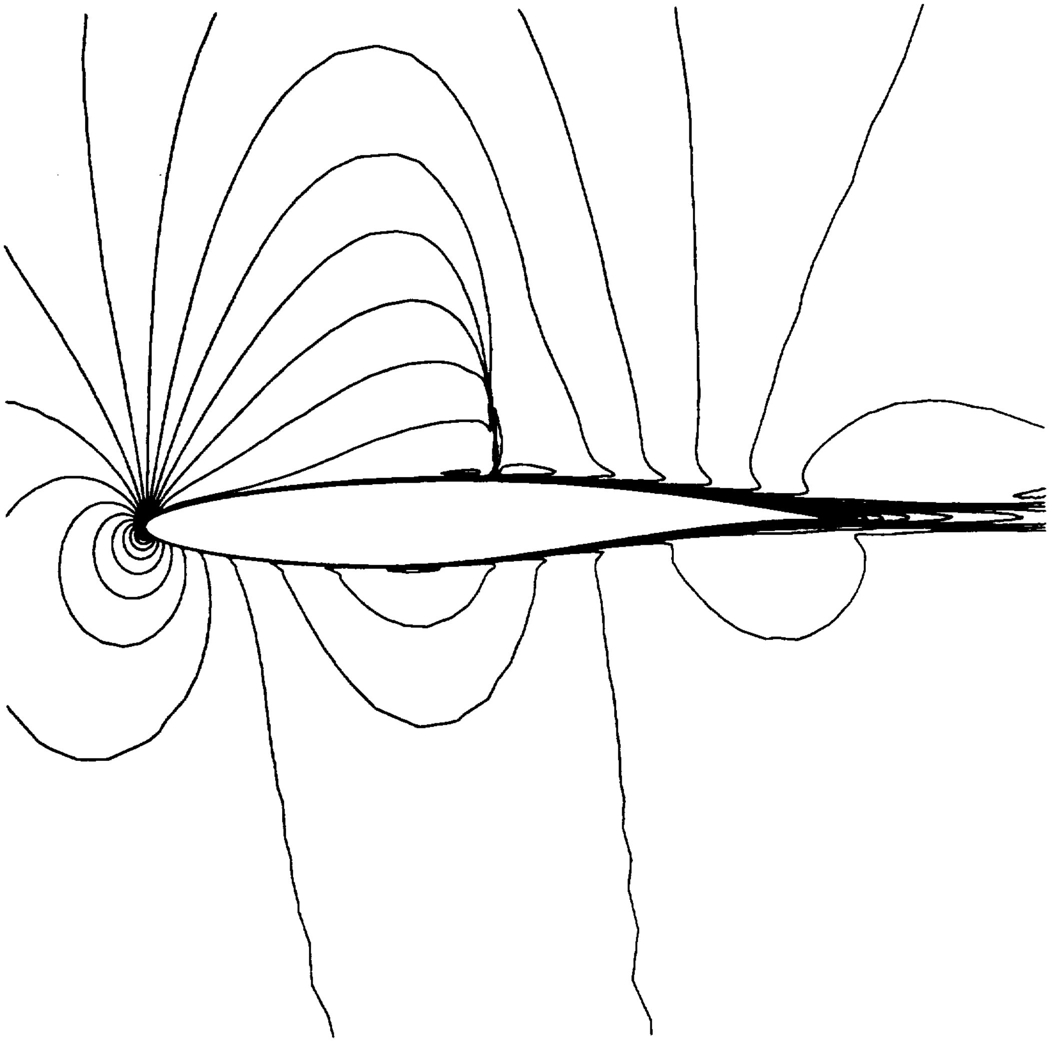


Figure 7.15 RAE2822 airfoil Mach number contours on the final mesh: cmin = 0.0, cmax = 1.20, and inc = 0.05.

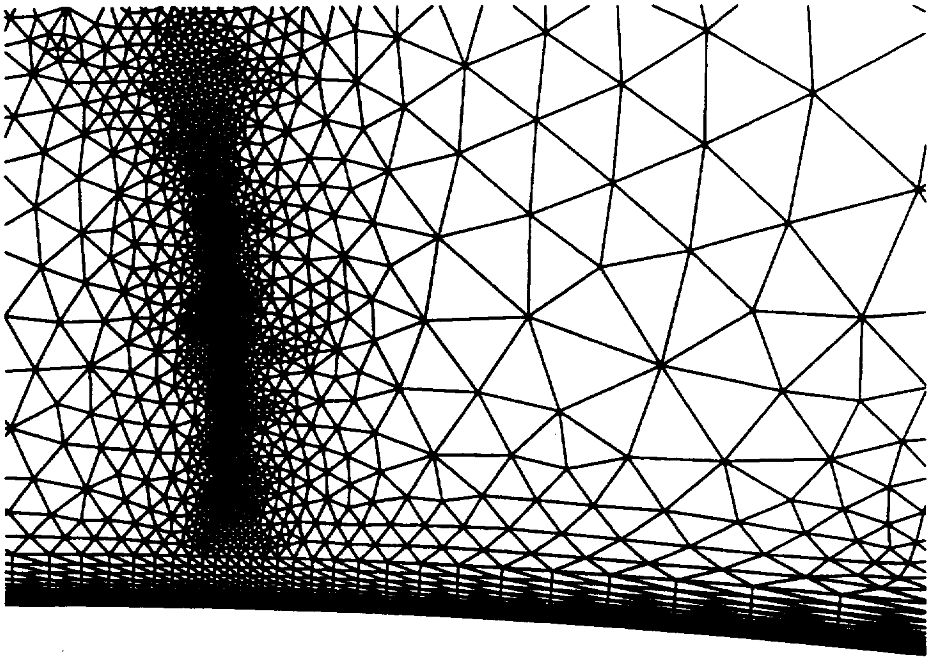


Figure 7.16 RAE2822 airfoil: shock region blowup of the final mesh.

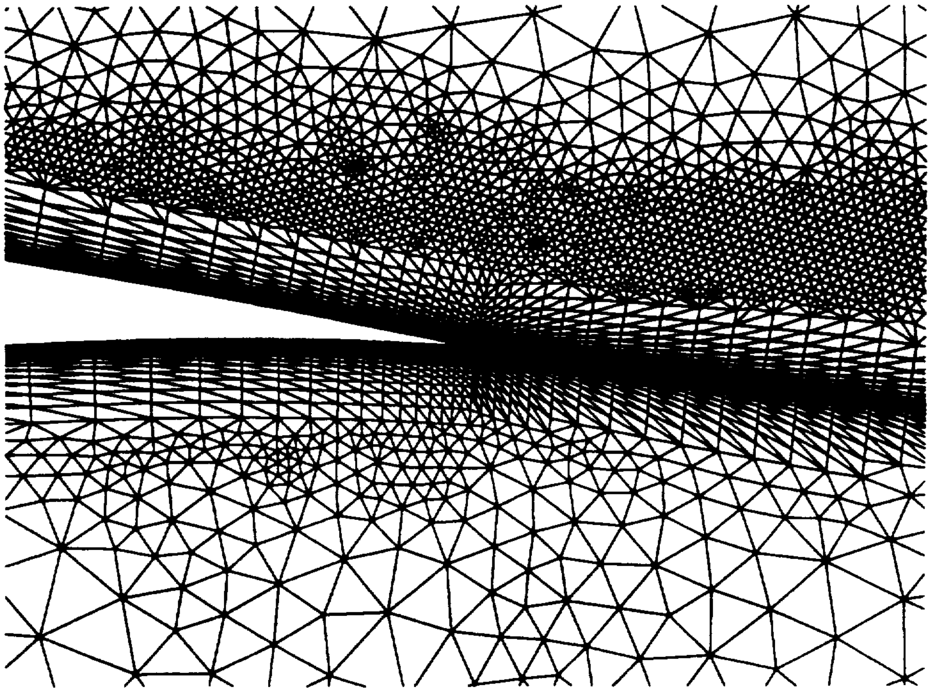


Figure 7.17 RAE2822 airfoil: trailing edge blowup of the final mesh.

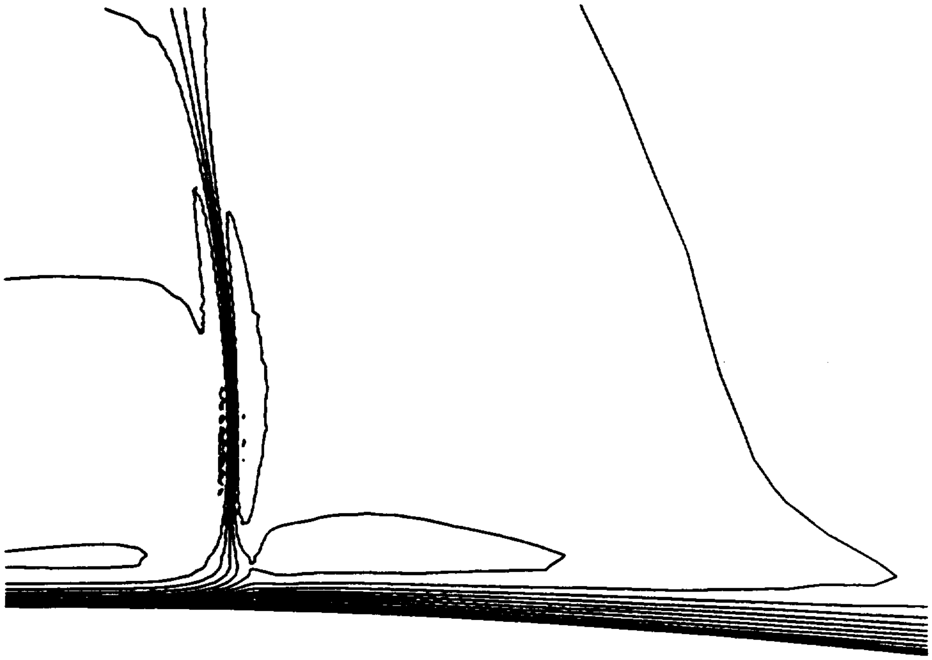


Figure 7.18 RAE2822 airfoil Mach number contours (cmin = 0.0, cmax = 1.20, and inc = 0.05): final mesh shock region.

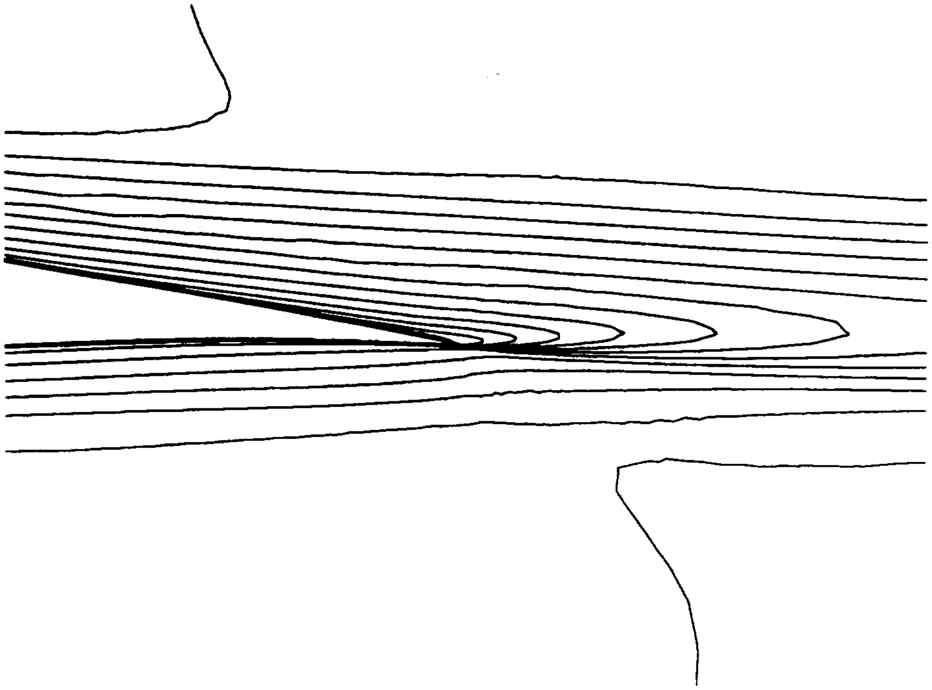


Figure 7.19 RAE2822 airfoil Mach number contours (cmin = 0.0, cmax = 1.20, and inc = 0.05): final mesh trailing edge region.

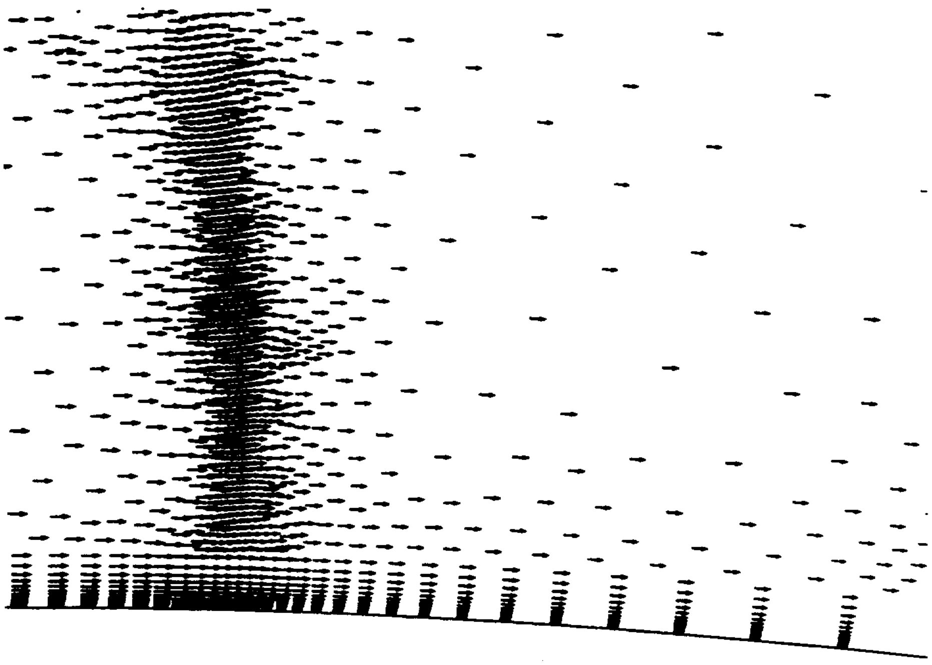


Figure 7.20 RAE2822 airfoil velocity vector: final mesh shock region.

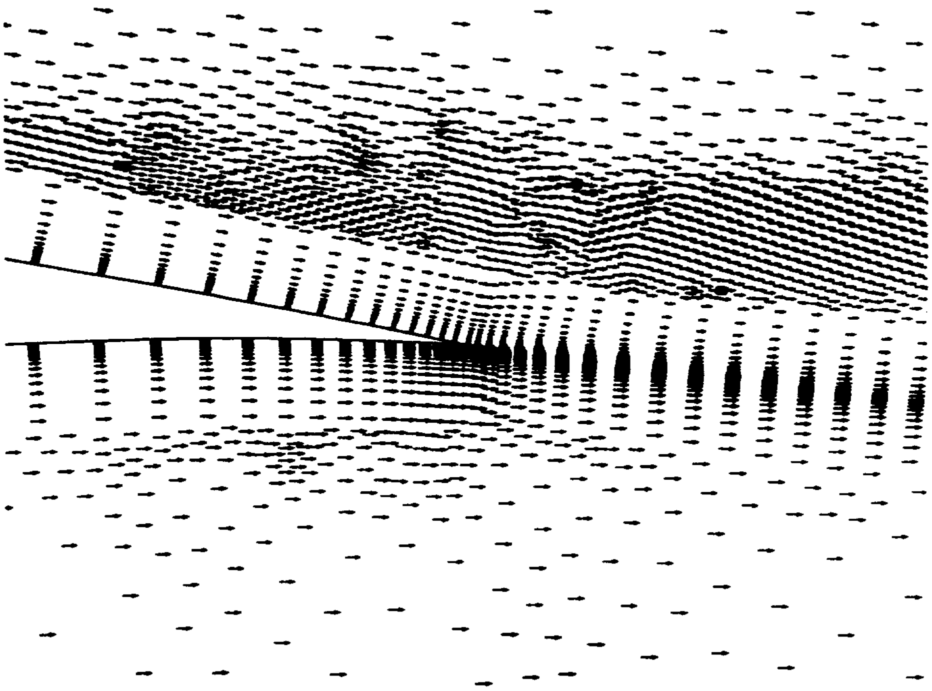


Figure 7.21 RAE2822 airfoil velocity vector: final mesh trailing edge region.

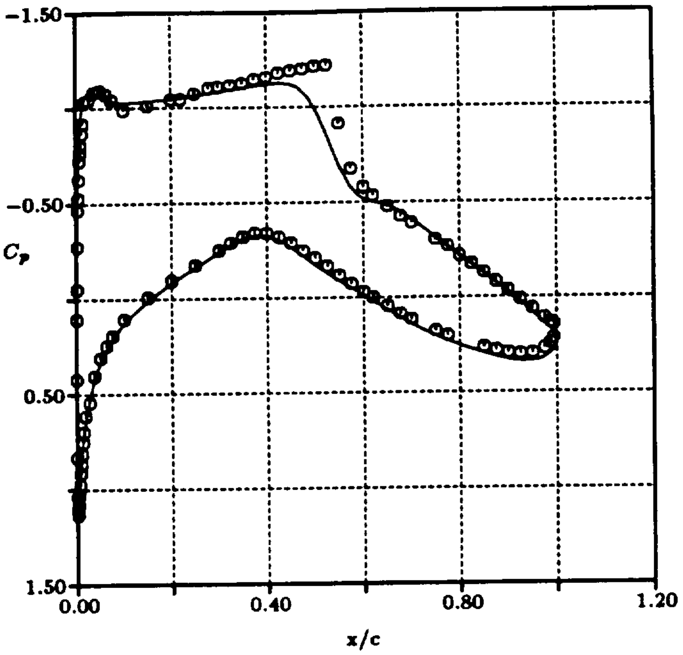


Figure 7.22 RAE2822 airfoil surface pressure coefficient for the initial mesh: solid line - numerical solution and symbol - experimental data.

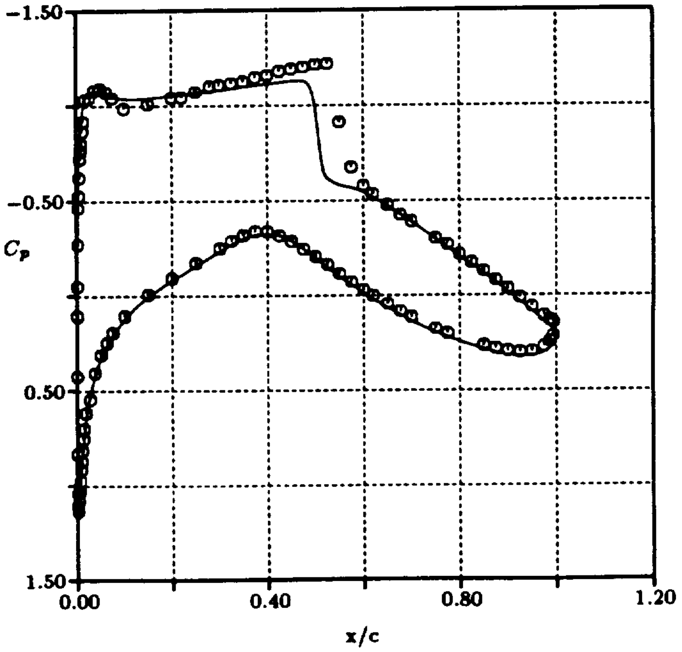


Figure 7.23 RAE2822 airfoil surface pressure coefficient for the final mesh: solid line - numerical solution and symbol - experimental data.

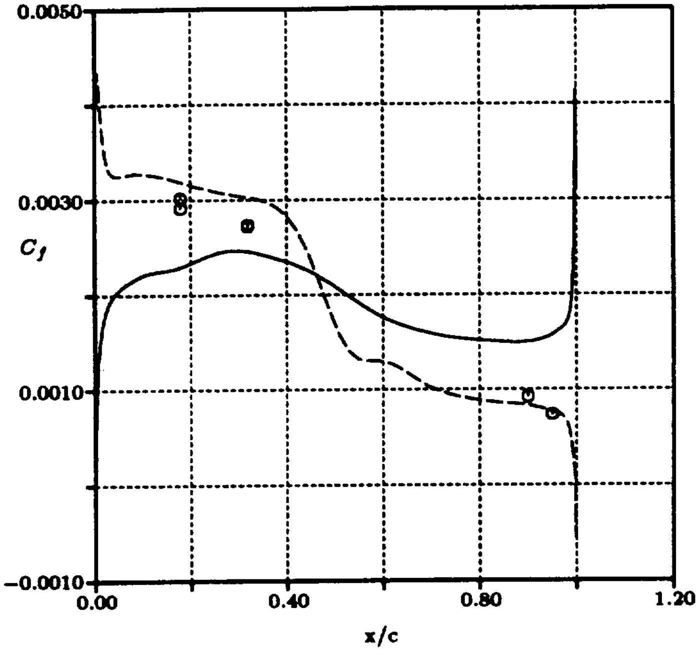


Figure 7.24 RAE2822 airfoil surface friction coefficient for the initial mesh: solid line - numerical solution at lower surface, dashed line - numerical solution at upper surface and symbol - experimental data at upper surface.

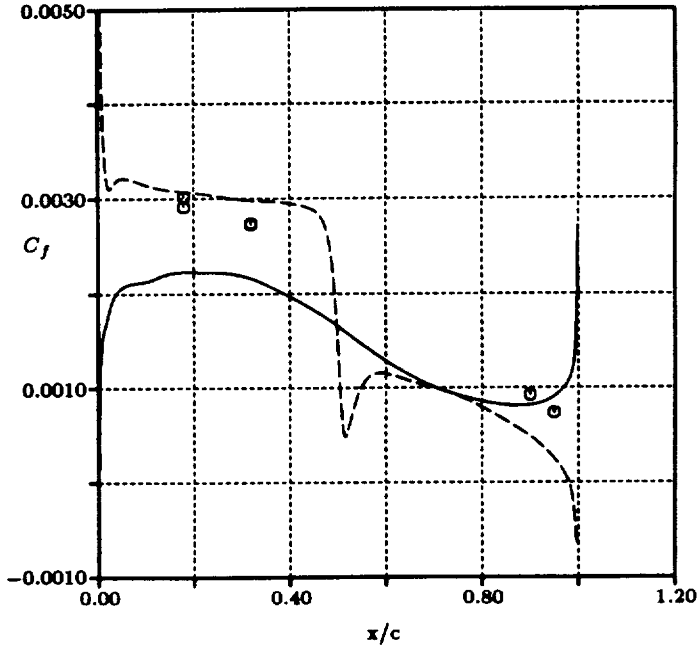


Figure 7.25 RAE2822 airfoil surface friction coefficient for the final mesh: solid line - numerical solution at lower surface, dashed line - numerical solution at upper surface and symbol - experimental data at upper surface.

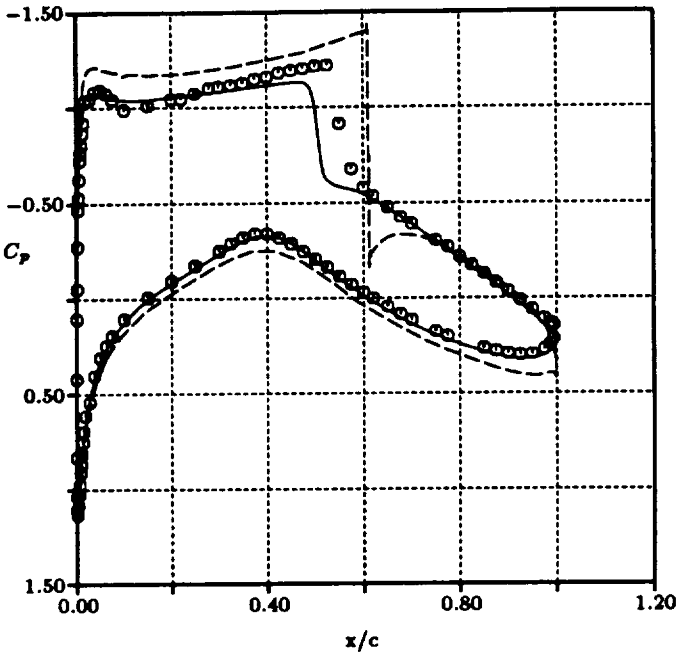


Figure 7.26 RAE2822 airfoil surface pressure coefficient for the final mesh: solid line - numerical turbulent solution, dashed line - numerical inviscid solution, and symbol - experimental data.

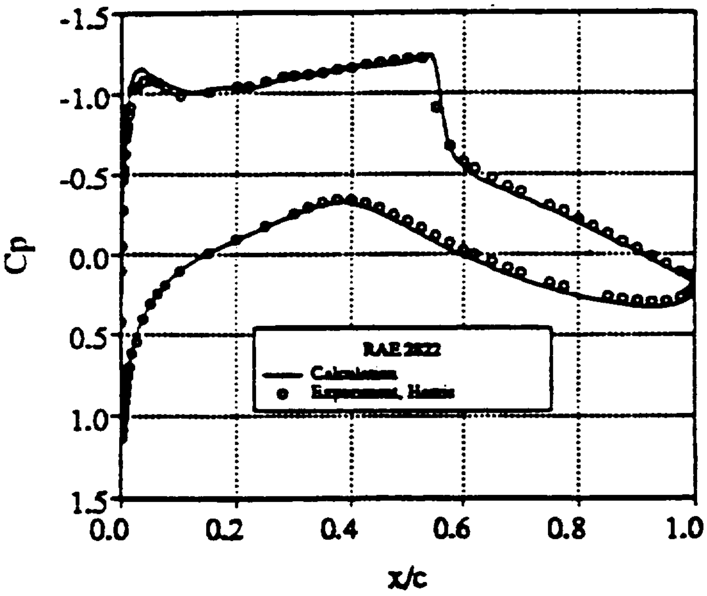


Figure 7.27 RAE2822 airfoil surface pressure coefficient: Barth's numerical results [7].

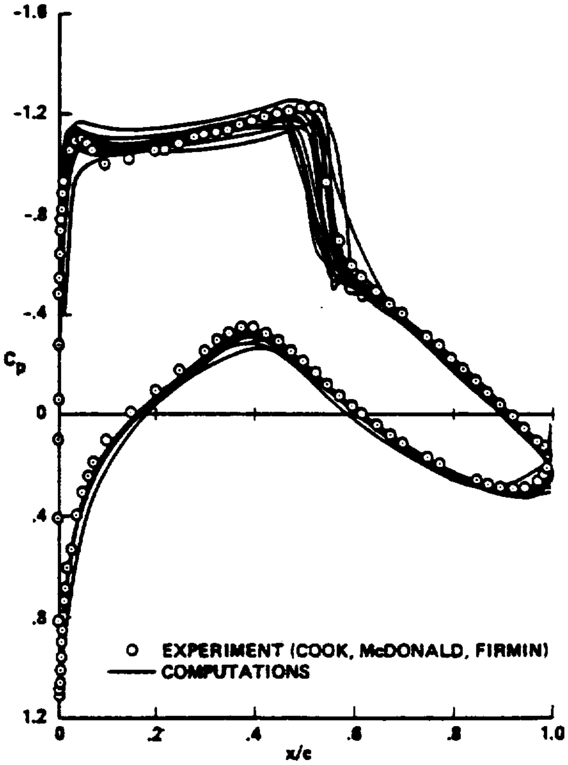


Figure 7.28 RAE2822 airfoil surface pressure coefficient: the 1987 viscous transonic airfoil workshop compendium of results [32].

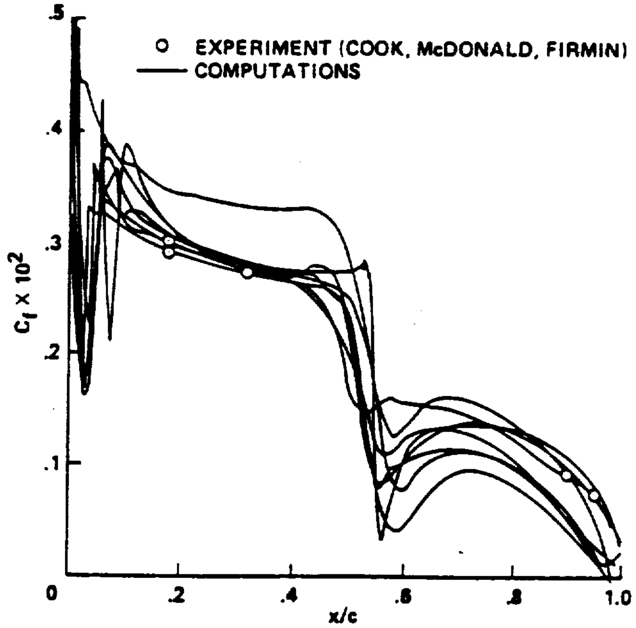


Figure 7.29 RAE2822 airfoil surface friction coefficient: the 1987 viscous transonic airfoil workshop compendium of results [32].

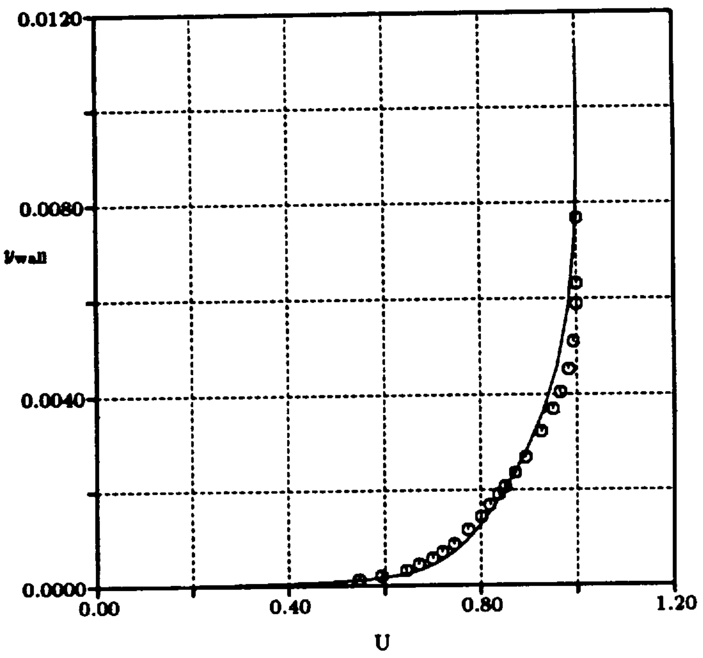


Figure 7.30 RAE2822 airfoil nondimensional velocity profile at x=0.319 for the initial mesh: solid line - computed results and symbol - experimental data.

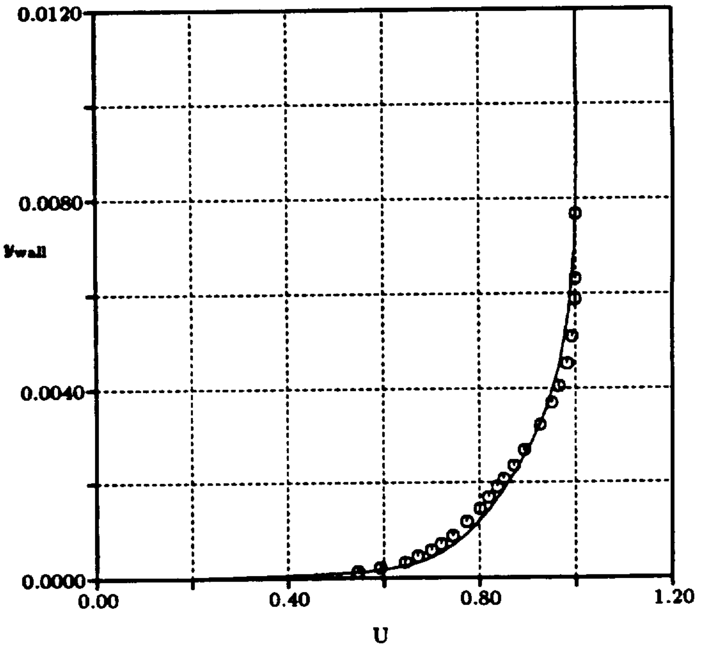


Figure 7.31 RAE2822 airfoil nondimensional velocity profile at x=0.319 for the final mesh: solid line - computed results and symbol - experimental data.

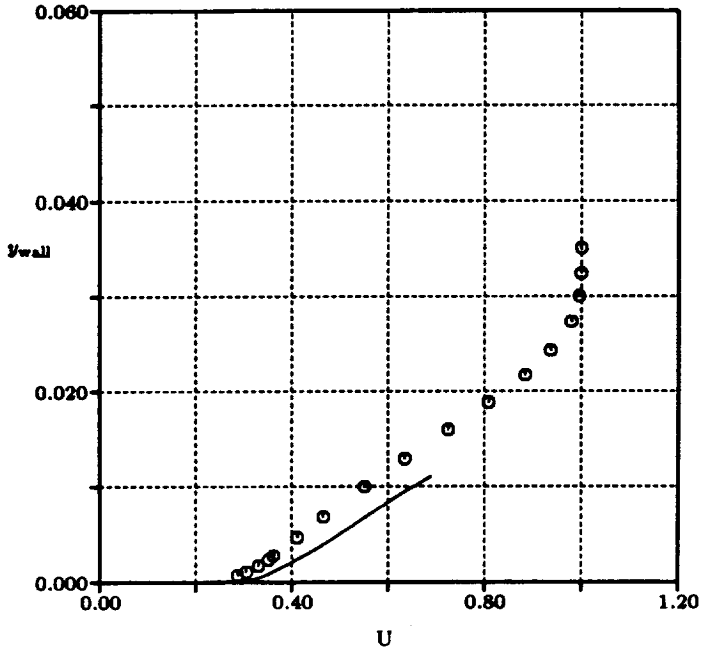


Figure 7.32 RAE2822 airfoil nondimensional velocity profile at x=0.95 for the initial mesh: solid line - computed results and symbol - experimental data.

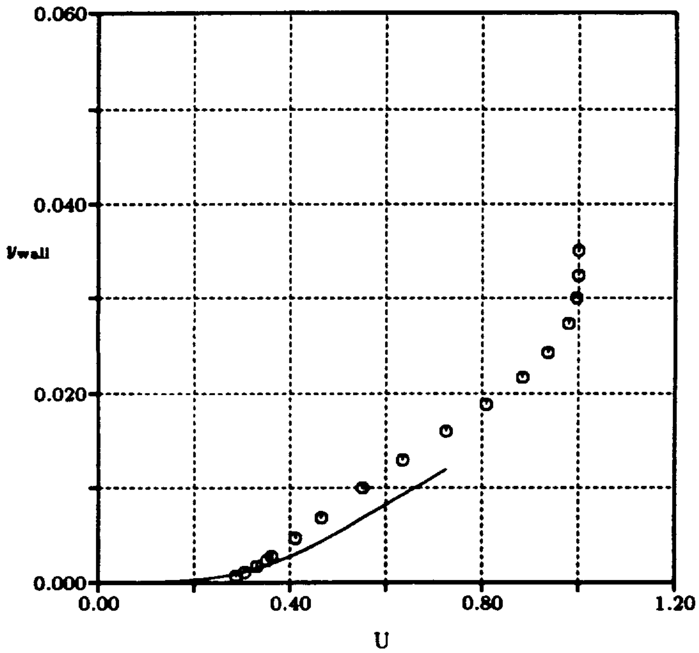


Figure 7.33 RAE2822 airfoil nondimensional velocity profile at x=0.95 for the final mesh: solid line - computed results and symbol - experimental data.

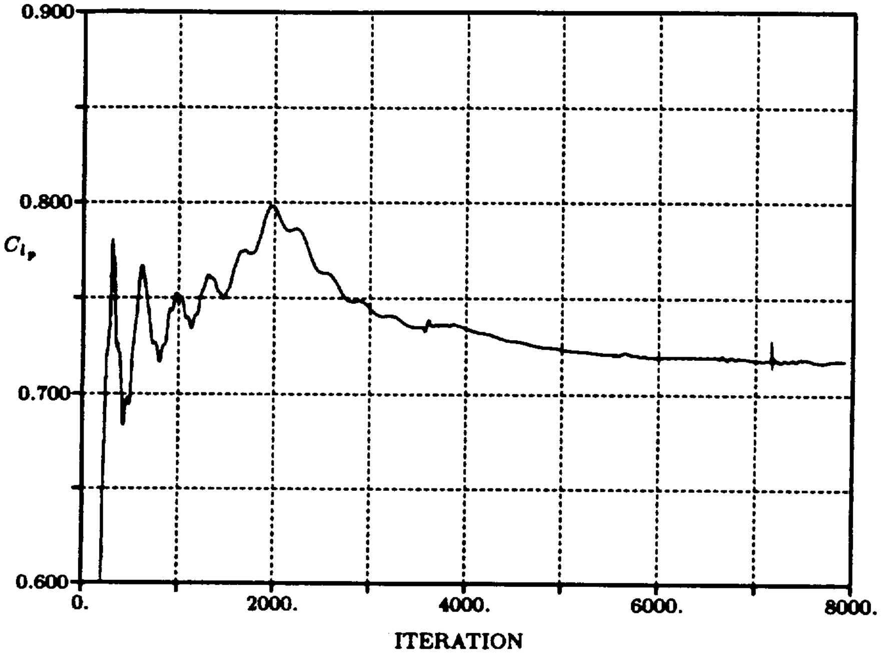


Figure 7.34 RAE2822 airfoil: C_{l_p} verses iteration.

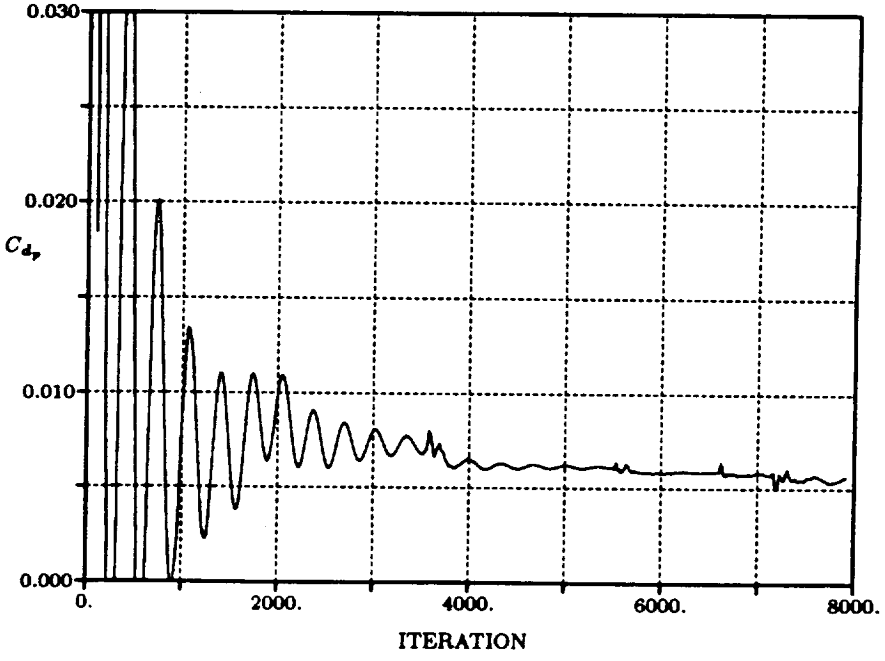


Figure 7.35 RAE2822 airfoil: C_{d_p} verses iteration.

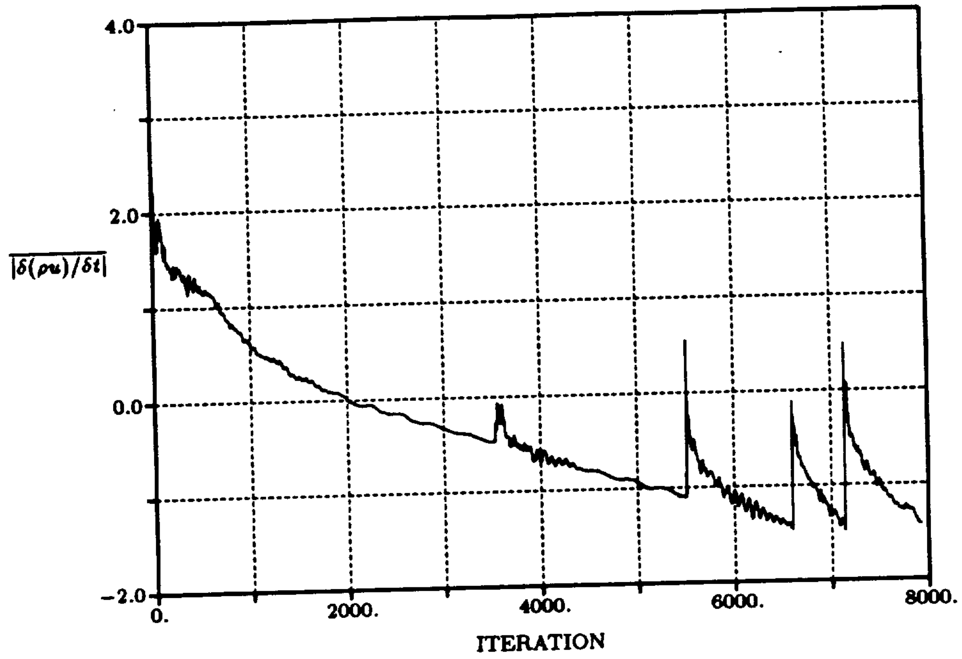


Figure 7.36 RAE2822 airfoil: average $|\delta(\rho u)/\delta t|$ verses iteration.

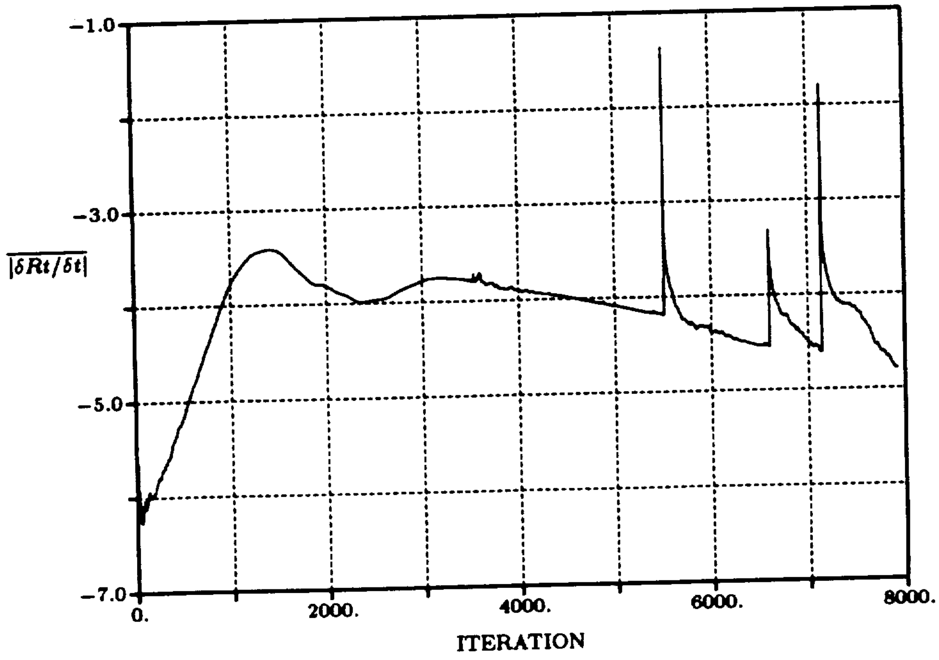


Figure 7.37 RAE2822 airfoil: average $|\delta Rt/\delta t|$ verses iteration.

7.3 NLR Two-Element Airfoil

The NLR two-element airfoil case illustrates the ease with which viscous flow problems on arbitrary multiply-connected regions can be solved. This two-element airfoil is composed of a NLR7301 main airfoil with a trailing edge flap. Experiments have been done for various angles of attack with a single flap deflection angle of 20° by van den Berg [84]. For the case presented the flow conditions are summarized in Table 7.4. This flow has a rapid expansion around the leading edge. There is a small laminar separation bubble on the upper surface starting from 2.6 to 4 % chord according to flow visualizations [84]. For such a small separation bubble, it is very important to have correct level of turbulent viscosity to resolve it.

The far field and near field meshes are shown in Figures 7.38 and 7.39, respectively. The far field boundary is placed at a radius of 5 chords from the airfoil. The vortex far field boundary condition [83] is specified at this boundary. Along wall surfaces, no-slip condition with adiabatic wall and zero normal pressure gradient boundary conditions are applied. The initial C-type local structured mesh is generated around the airfoil through the specification of a normal wall mesh scale, $\delta_{y^+} = 2\text{x}10^{-5}$ chord, with a streamwise wall mesh scale ($N_s = 40$ and $\delta_w = 0.015$ chord). In the far field the unstructured mesh is uniform with a mesh scale of $\delta_{\max} = 0.5$ chord. The sequence of adapted meshes are shown in Figures 7.39 to 7.41. Mesh properties and force coefficients are listed in Table 7.5. Comparing the final mesh Mach number contours with the initial mesh solution as shown in Figures 7.42 and 7.43 shows improved resolution of the boundary layer edges. The mesh and flow at the leading edge and in the airfoil-flap gap region are shown in

Table 7.4 Flow conditions for the NLR two-element airfoil.

M_∞	α	δ_{flap}	Flap gap % c	T_∞ (° K)	Re_L	L_{ref} (m)
0.185	6.0°	20.0°	2.6	293	$2.51\text{x}10^6$	0.57
Note: Experimental data is available in Reference [84].						

Figures 7.44 to 7.47. Although these plots show an accurate prediction of stagnation point location, the leading edge separation bubble is not resolved. This is also confirmed by the positive value of the friction coefficient distribution along the surface as shown in Figure 7.53. Since the refinement parameter based on surface curvature does a good job in resolving the leading edge region, plots of the initial and final mesh surface pressure distribution are almost exactly the same (see Figures 7.50 and 7.51). The surface friction coefficient distribution does improved as the mesh is refined as shown in Figures 7.52 and 7.53. Similar improvements are shown in the velocity plots (see Figures 7.54 to 7.57). This improvement is due to a better resolution of near-wall viscous regions since a smaller wall distance Δy_{wall} is obtained as the solution develops (see Table 7.5).

Figures 7.58 and 7.59 present the convergence histories for the pressure portion of the lift and drag coefficients. Convergence histories for the average residual of $|\delta(\rho u)/\delta t|$ and $|\delta R_t/\delta t|$ are given in Figures 7.60 and 7.61, respectively. This case takes about 5.7 hours of Cray-YMP CPU time. Although the difference in the force coefficient between last two meshes does not fall within the convergence criteria (.002), the adaptation cycle has been stopped due to the huge computation time required for continued adaptations. Further refinement of the mesh will not improve the solution since the wall distance has converged to a constant value on the last two adaptations (see Table 7.5). The computed total lifting coefficient is about 1.25 % larger than the measured value.

Table 7.5 NLR two-element airfoil: $M_\infty = 0.185$, $\alpha = 6.0°$, $\delta_{\text{flap}} = 20°$, $Re_L = 2.51\text{x}10^6$, $C_{l_{\text{meas}}} = 2.416$, and $C_{d_{\text{meas}}} = 0.0229$.

MESH	0	1	2
nodes	9,510	11,579	13,666
elements	18,708	22,793	26,931
C_{l_p}	2.3983	2.4345	2.4415
C_{d_p}	0.0470	0.0411	0.0341
C_{l_v}	0.00358	0.00415	0.00491
C_{d_v}	0.00073	0.00084	0.00094
δ_{max}/chord	0.5	0.5	0.5
$\delta_{\text{max}}/\delta_{\text{min}}$	1	39	78
$\Delta y_{\text{wall}}(\text{x}10^6)$	10.0	5.56	4.81
y^+_{pr}	2	2.5	2.5
A_w	750	750	750
Cray-YMP CPU (sec)	9,770	3,362	7,411

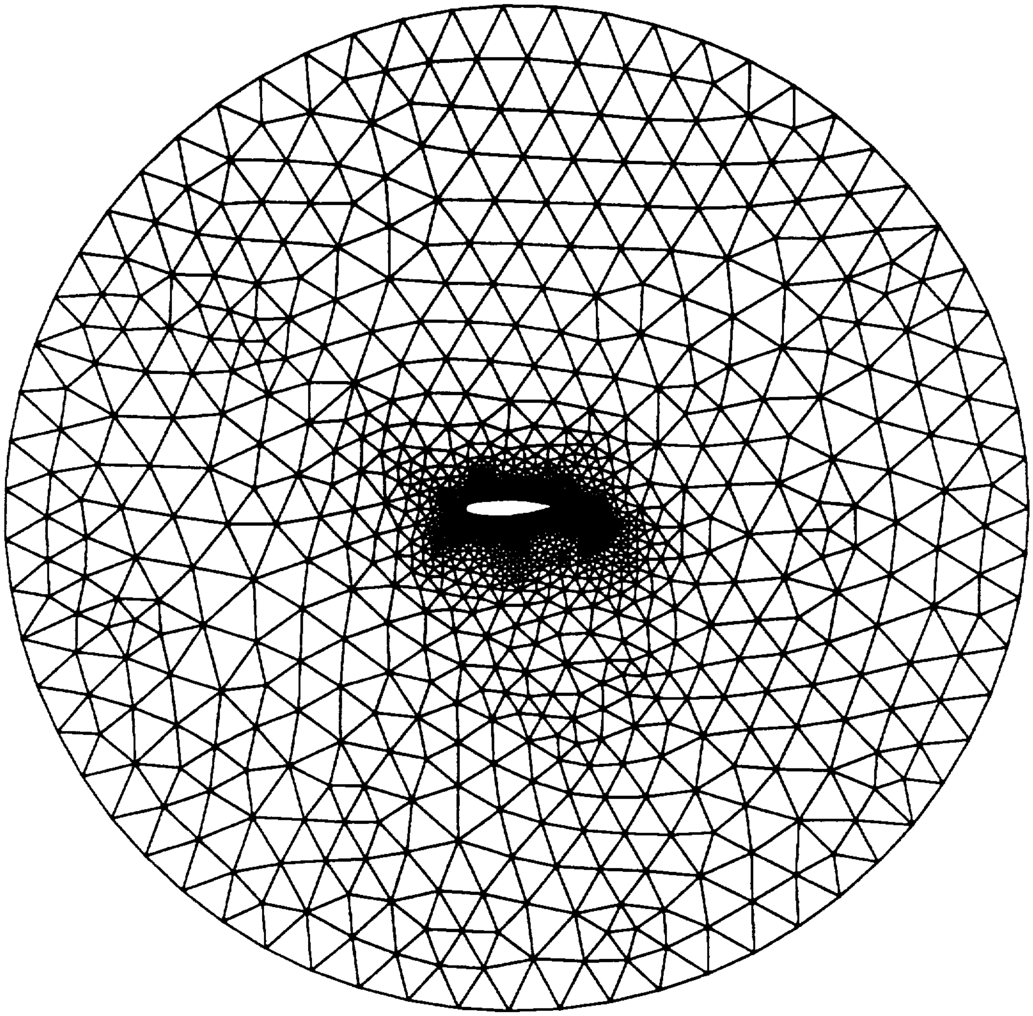


Figure 7.38 NLR two-element airfoil initial mesh: far field view.

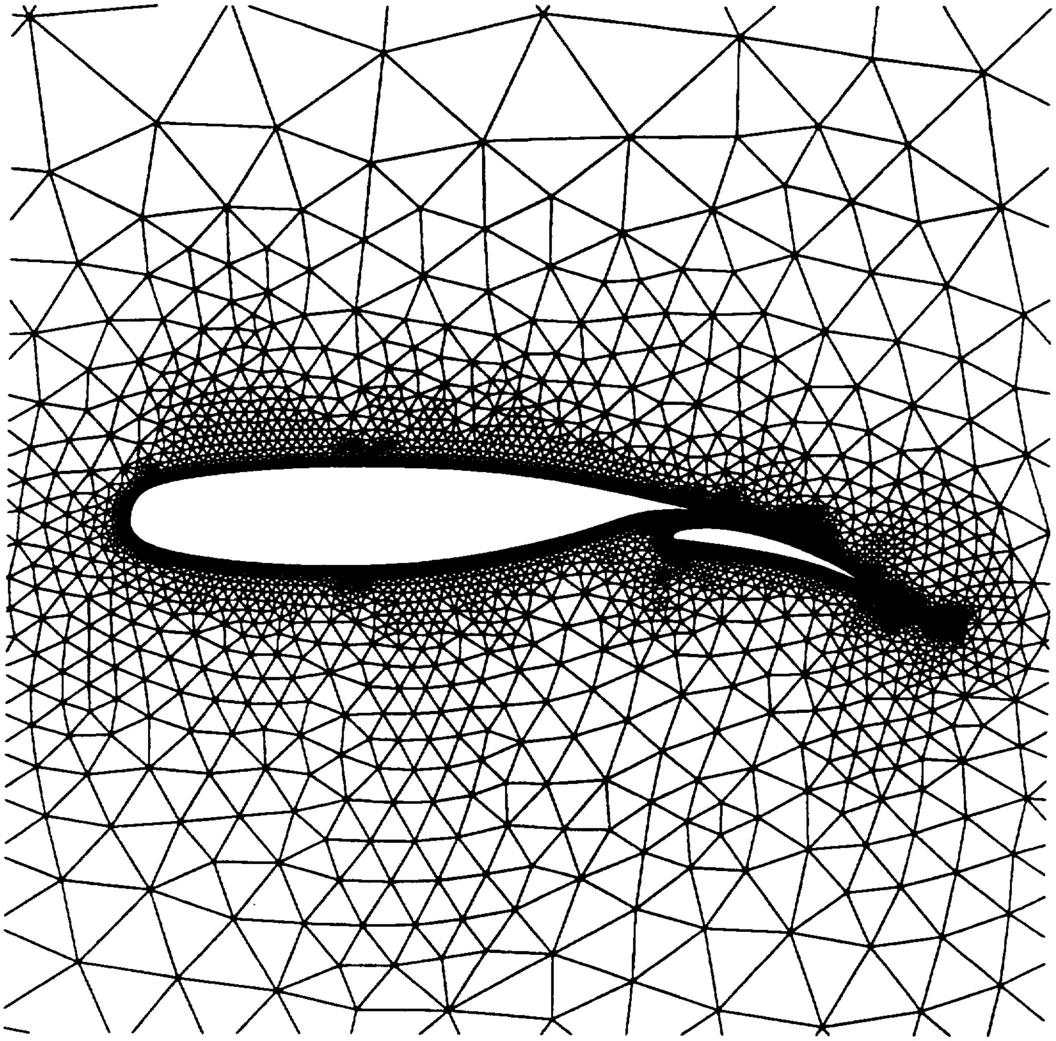


Figure 7.39 NLR two-element airfoil initial mesh: 9,510 nodes and 18,708 elements.

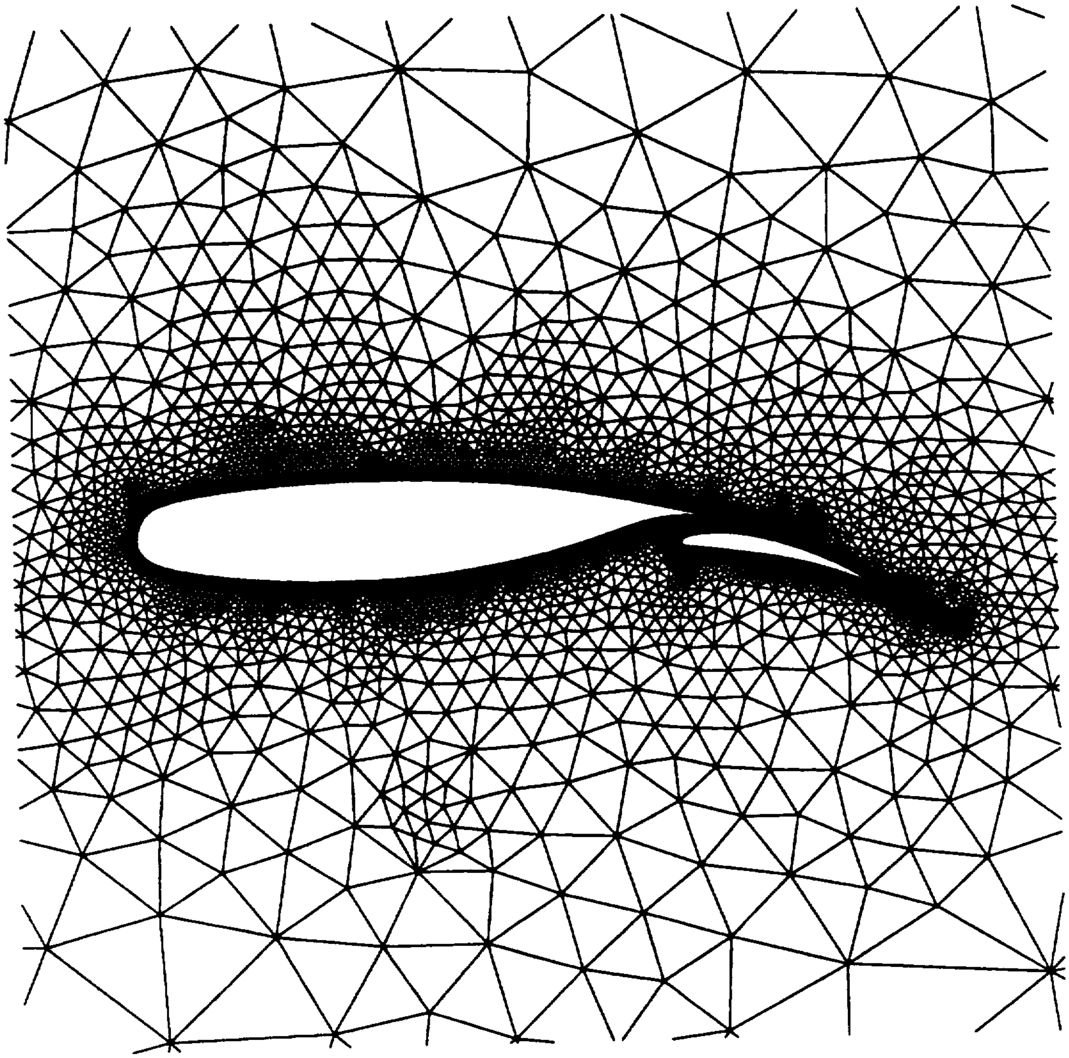


Figure 7.40 NLR two-element airfoil 1st adapted mesh: 11,579 nodes and 22,793 elements.

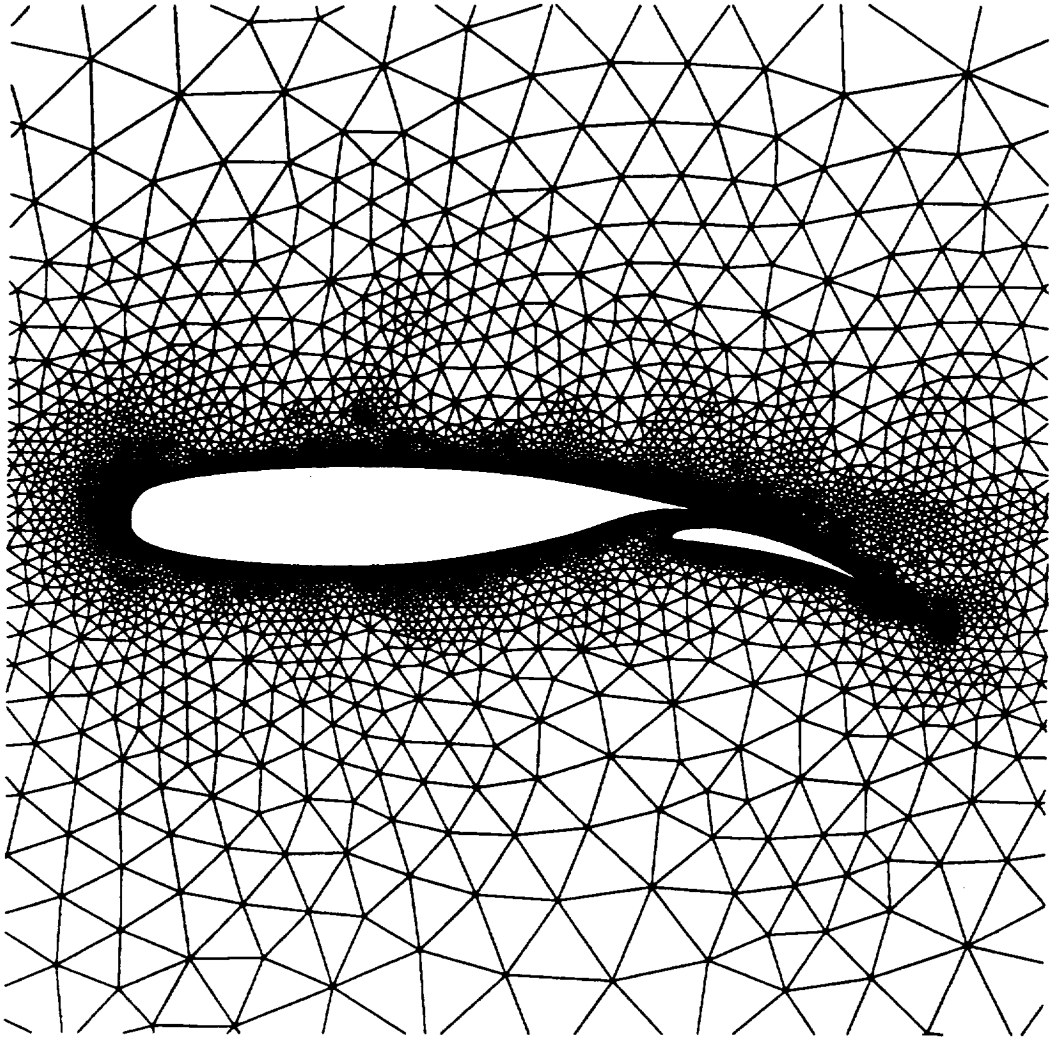


Figure 7.41 NLR two-element airfoil 2nd (final) adapted mesh: 13,666 nodes and 26,931 elements.

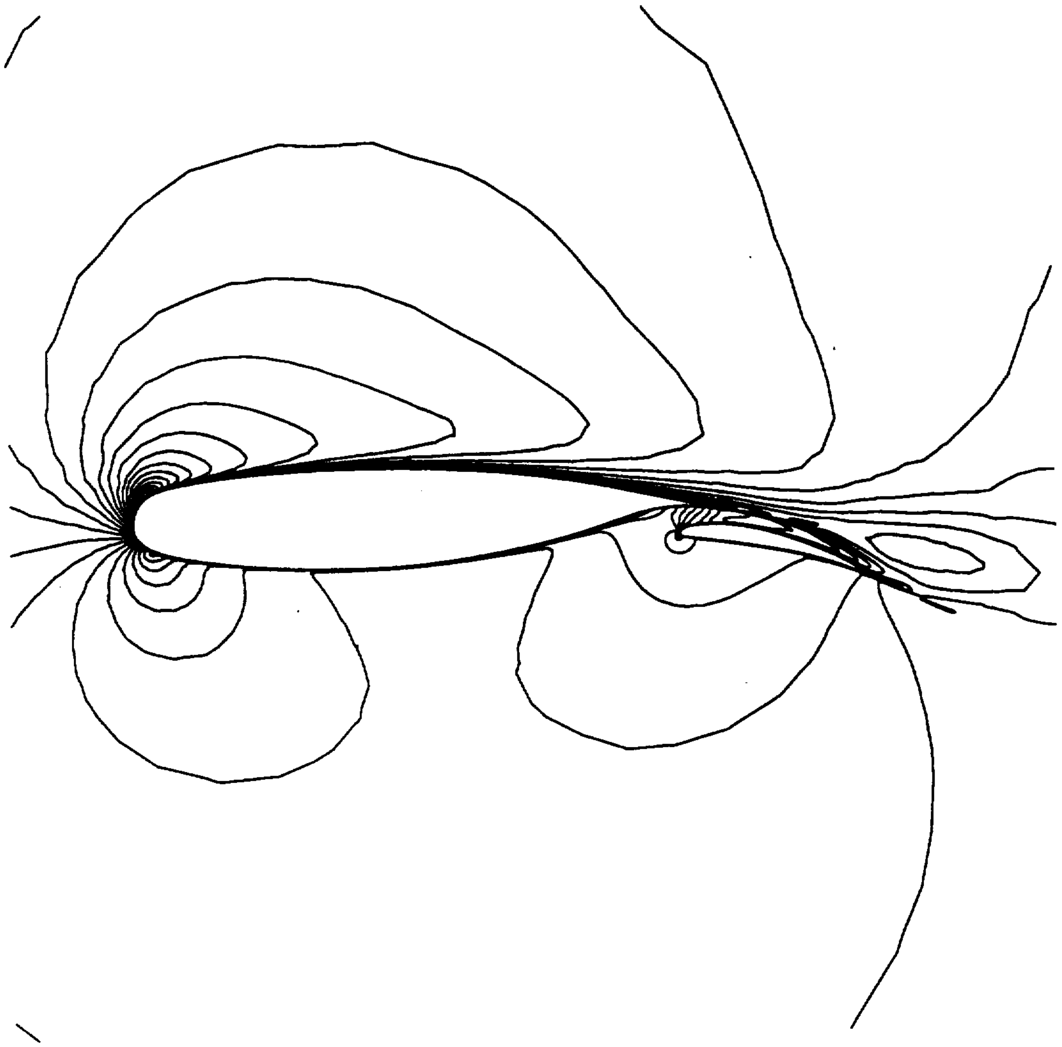


Figure 7.42 NLR two-element airfoil Mach number contours on the initial mesh: cmin = 0.0, cmax = 0.50, and inc = 0.02.

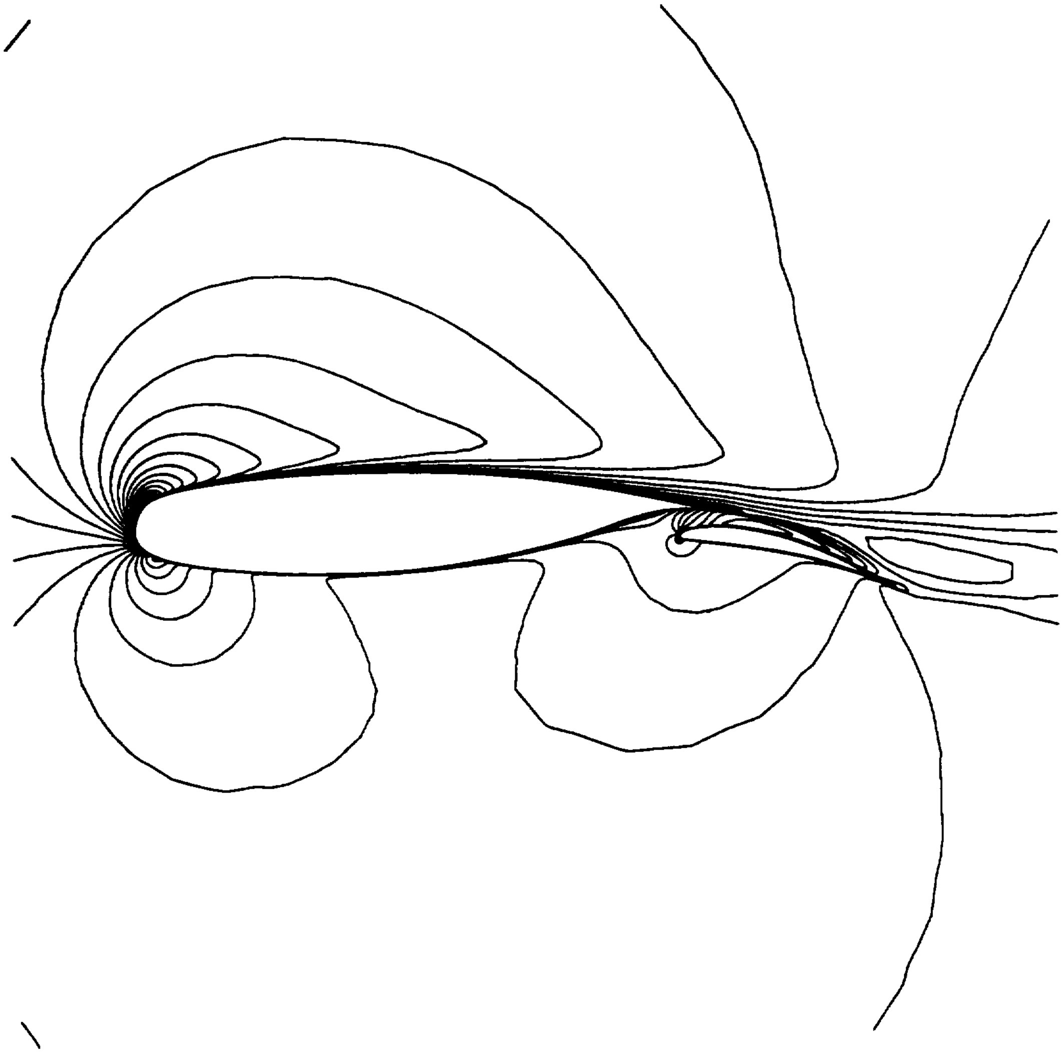


Figure 7.43 NLR two-element airfoil Mach number contours on the final mesh: cmin = 0.0, cmax = 0.50, and inc = 0.02.

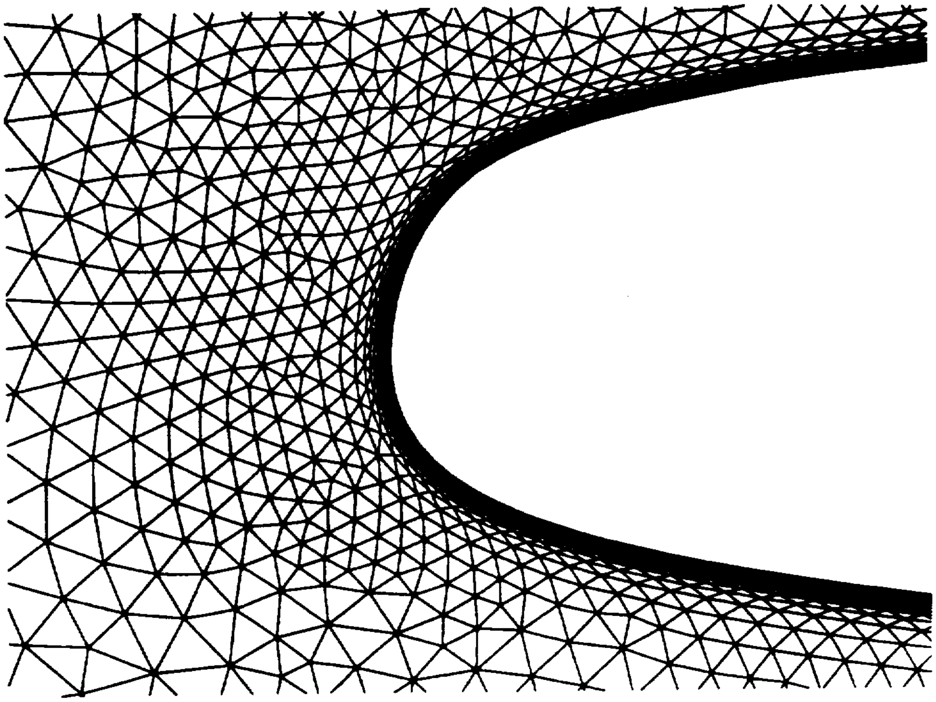


Figure 7.44 NLR two-element airfoil: leading edge blowup of the final mesh.

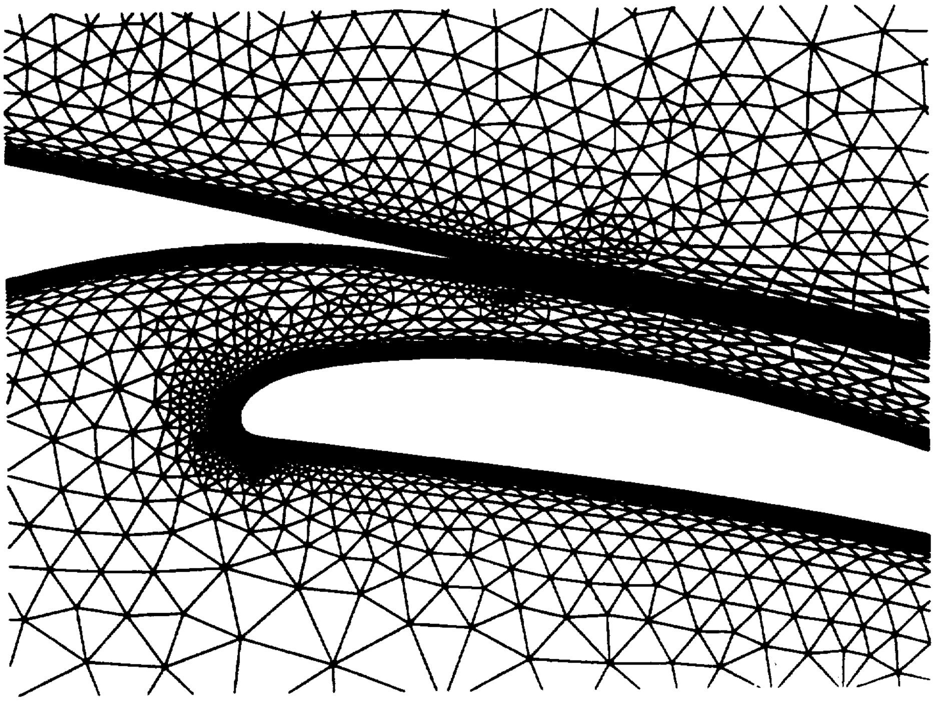


Figure 7.45 NLR two-element airfoil: airfoil-flap gap blowup of the final mesh.

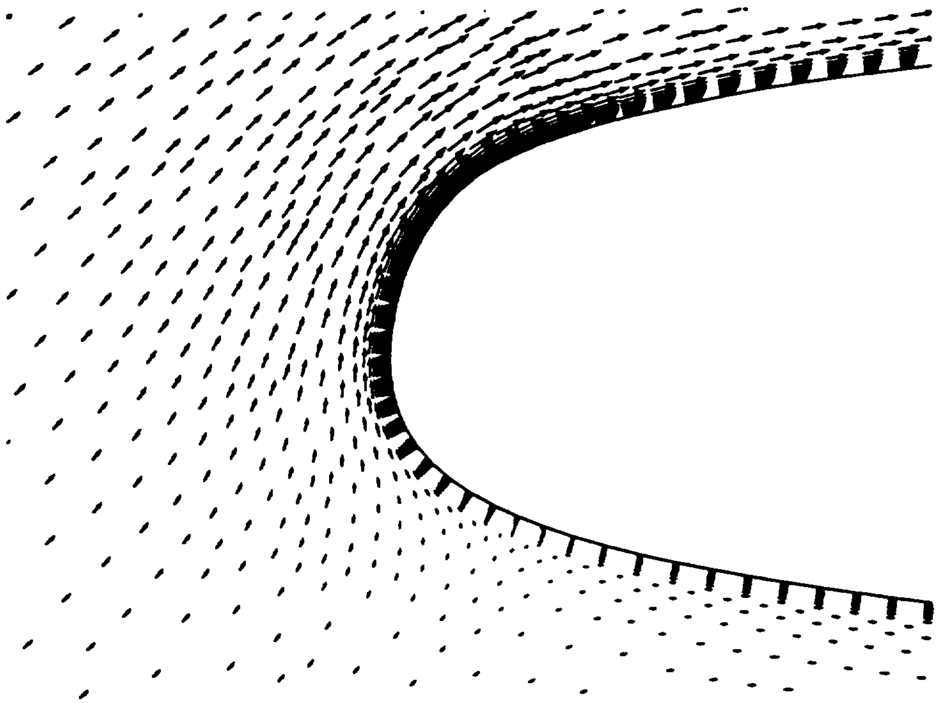


Figure 7.46 NLR two-element airfoil velocity vector: final mesh leading edge region.

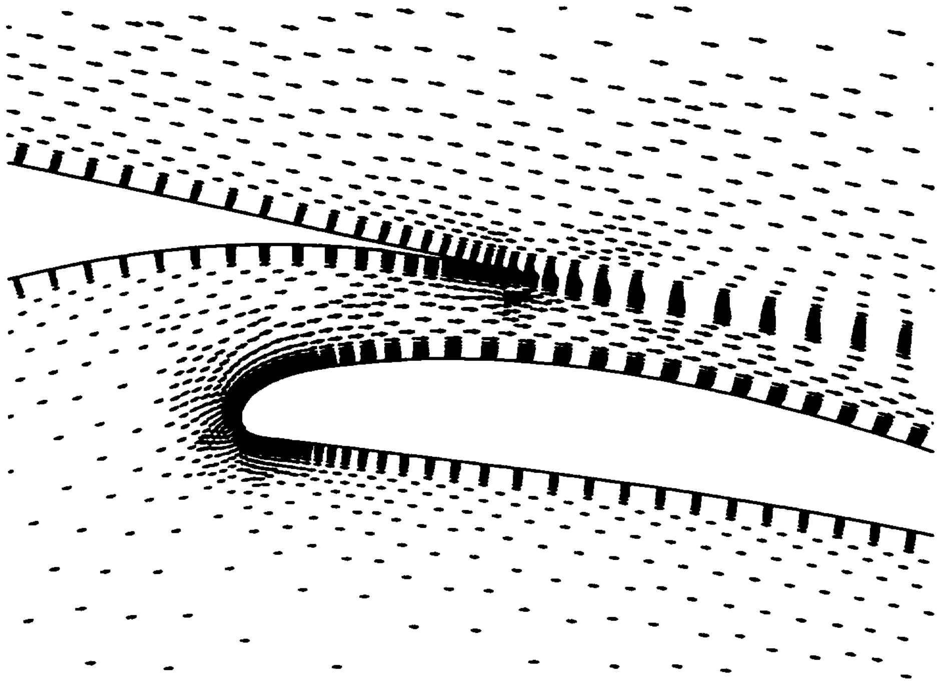


Figure 7.47 NLR two-element airfoil velocity vector: final mesh airfoil-flap region.

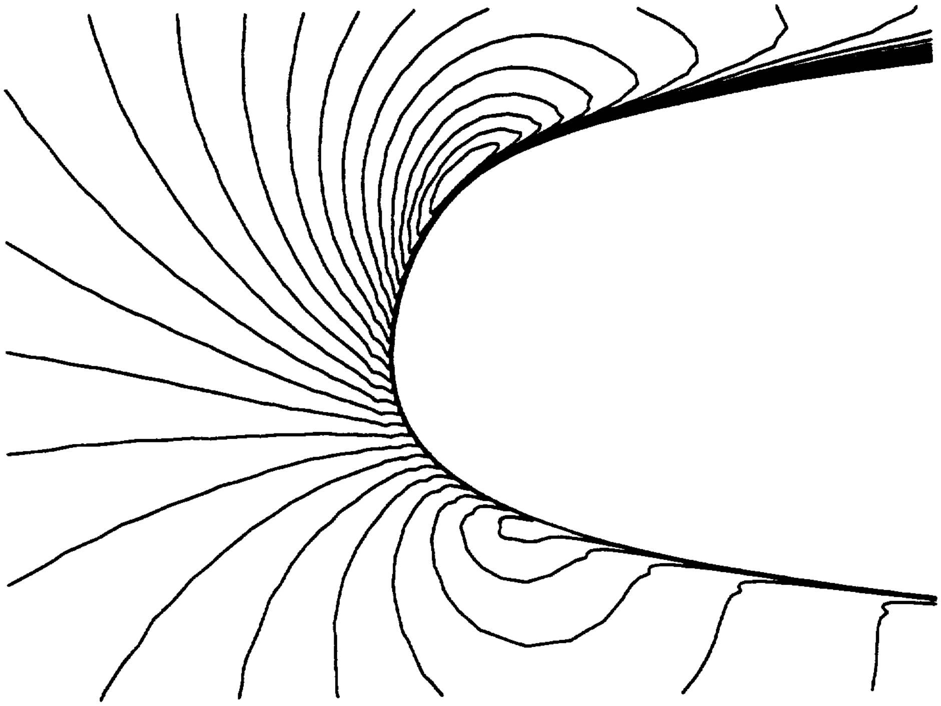


Figure 7.48 NLR two-element airfoil Mach number contours (cmin = 0.0, cmax = 1.20, and inc = 0.05): final mesh leading edge region.

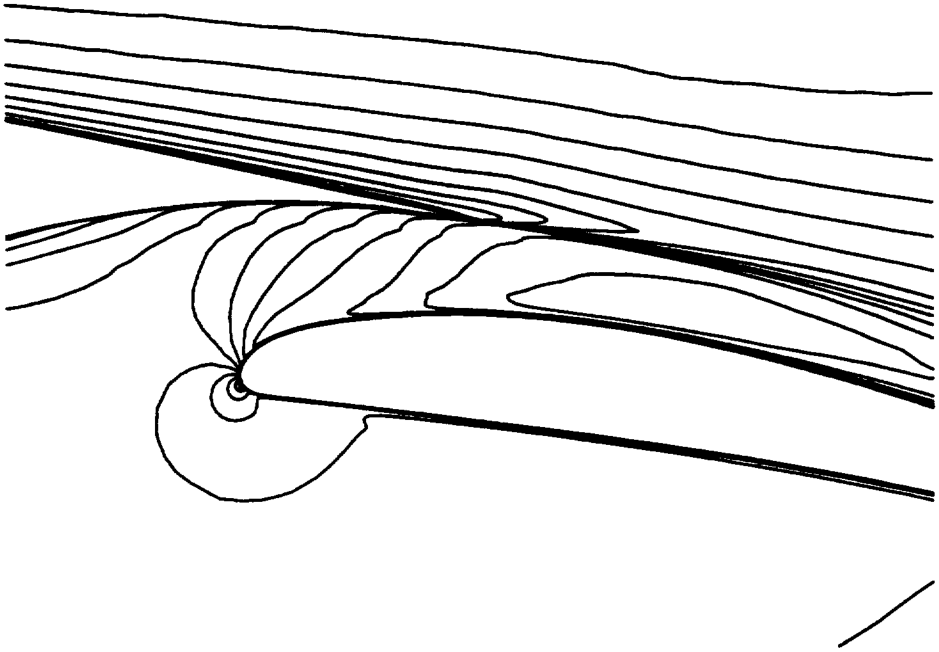


Figure 7.49 RAE2822 airfoil Mach number contours (cmin = 0.0, cmax = 1.20, and inc = 0.05): final mesh airfoil-flap region.

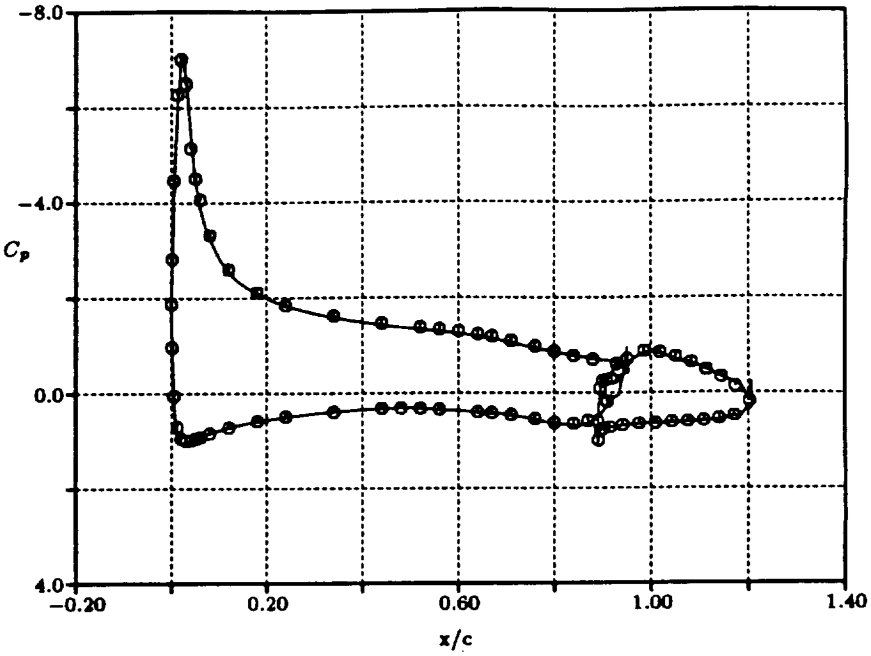


Figure 7.50 NLR two-element airfoil surface pressure coefficient for the initial mesh: solid line - numerical solution and symbol - experimental data.

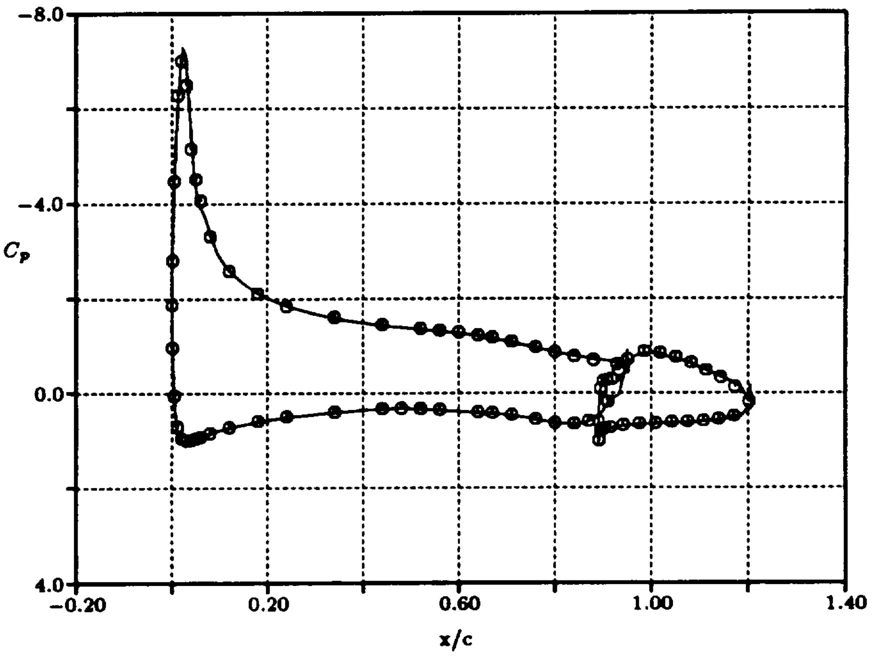


Figure 7.51 NLR two-element airfoil surface pressure coefficient for the final mesh: solid line - numerical solution and symbol - experimental data.

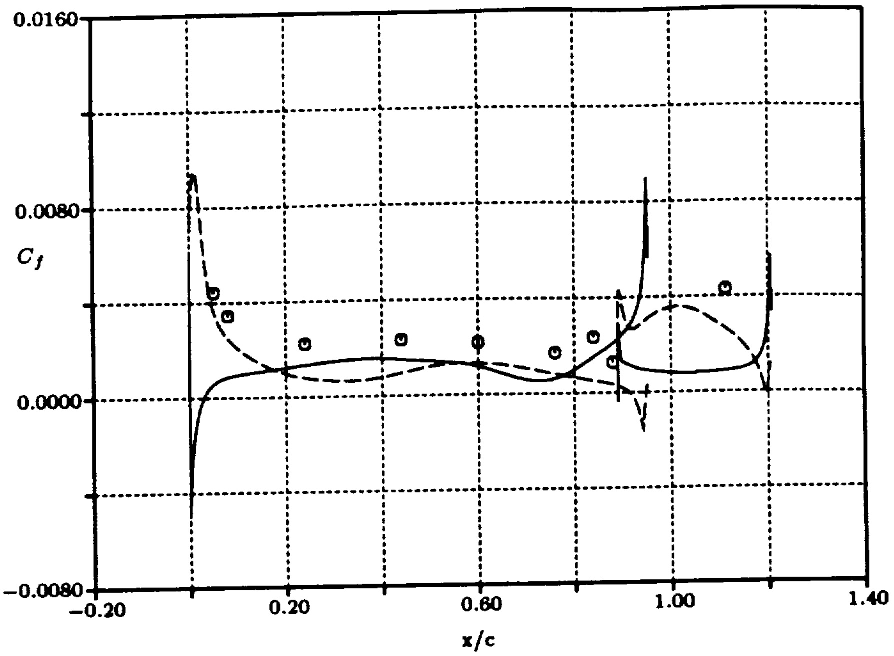


Figure 7.52 NLR two-element airfoil surface friction coefficient for the initial mesh: solid line - numerical solution at lower surface, dashed line - numerical solution at upper surface and symbol - experimental data at upper surface.

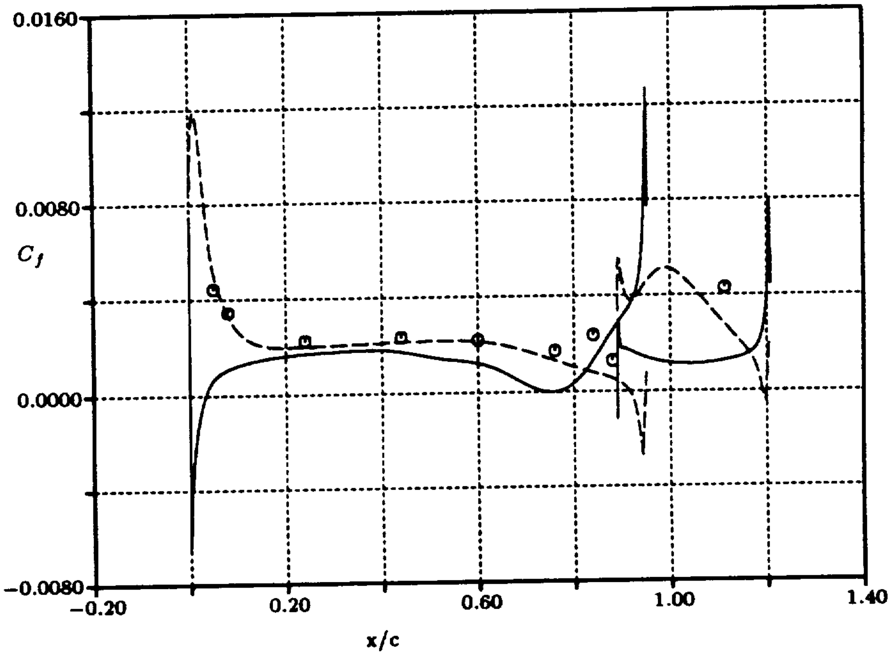


Figure 7.53 NLR two-element airfoil surface friction coefficient for the final mesh: solid line - numerical solution at lower surface, dashed line - numerical solution at upper surface and symbol - experimental data at upper surface.

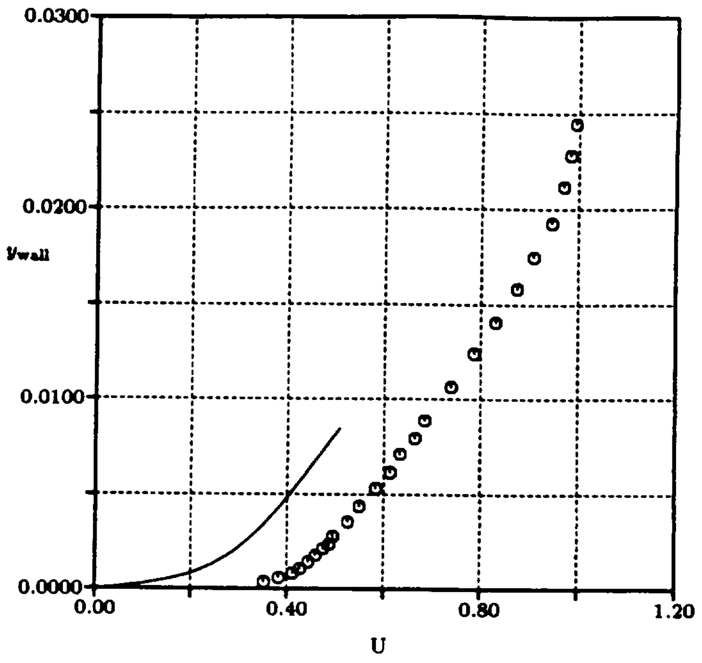


Figure 7.54 NLR two-element airfoil nondimensional velocity profile at x=0.88 for the initial mesh: solid line - computed results and symbol - experimental data.

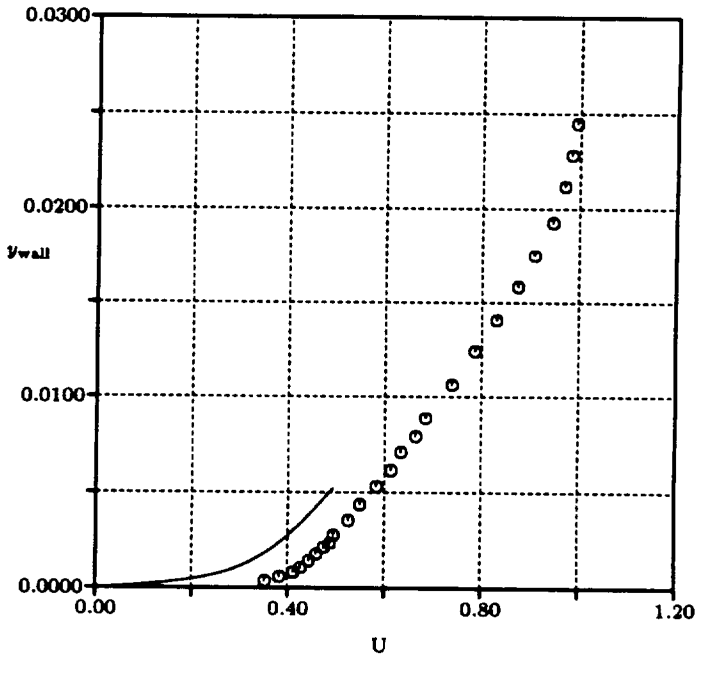


Figure 7.55 NLR two-element airfoil nondimensional velocity profile at x=0.88 for the final mesh: solid line - computed results and symbol - experimental data.

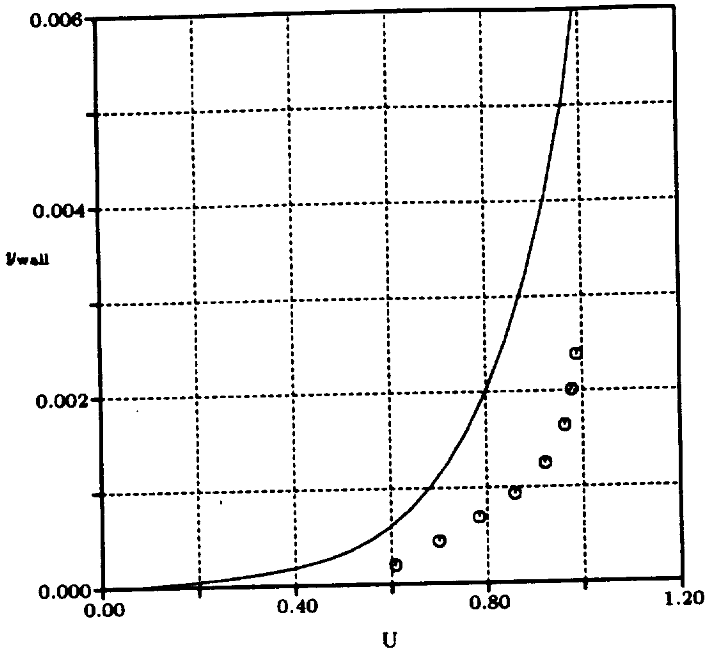


Figure 7.56 NLR two-element airfoil nondimensional velocity profile at x=1.11 for the initial mesh: solid line - computed results and symbol - experimental data.

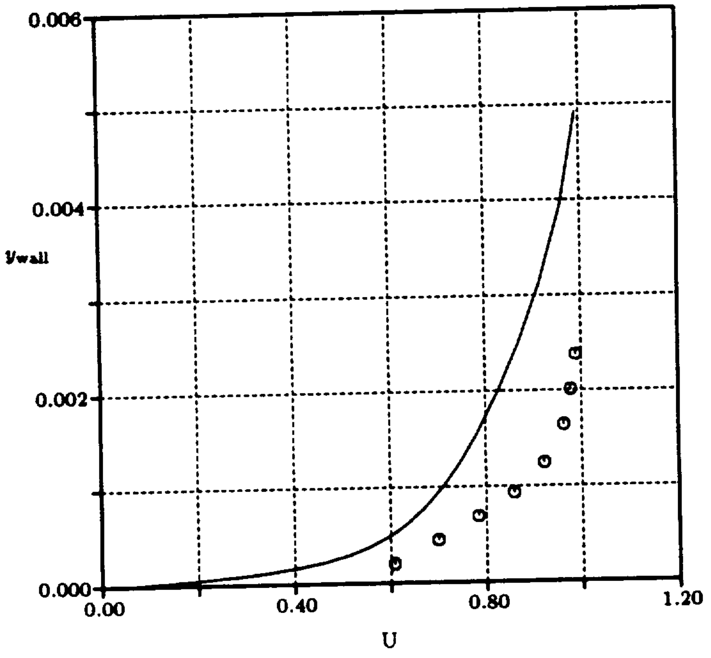


Figure 7.57 NLR two-element airfoil nondimensional velocity profile at x=1.11 for the final mesh: solid line - computed results and symbol - experimental data.

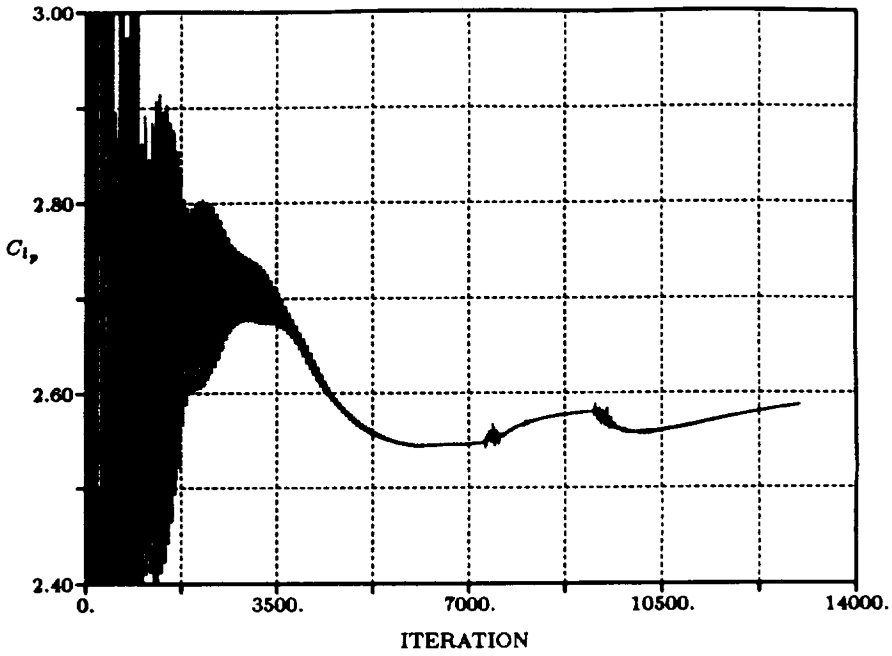


Figure 7.58 NLR two-element airfoil: C_{l_p} verses iteration.

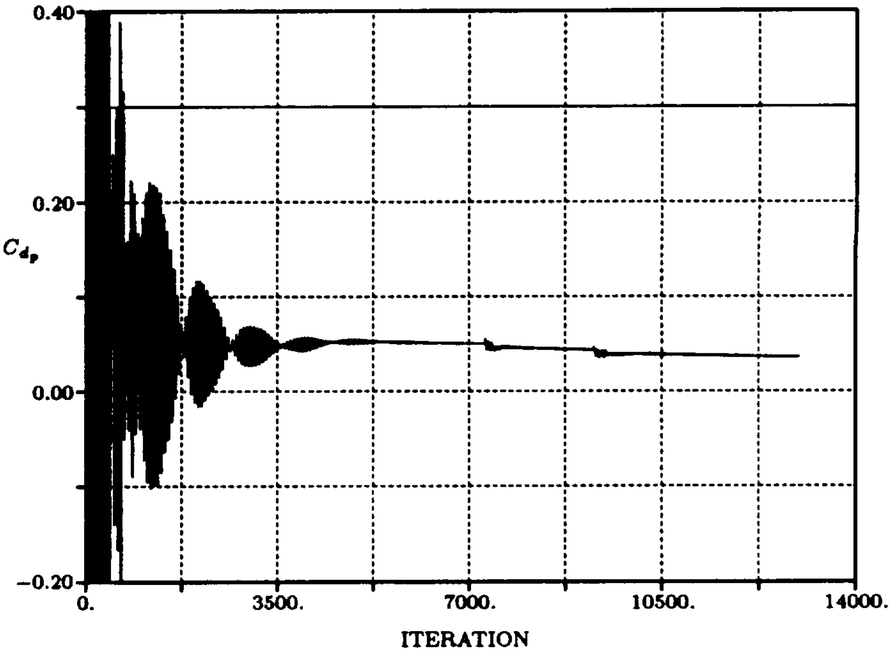


Figure 7.59 NLR two-element airfoil: C_{d_p} verses iteration.

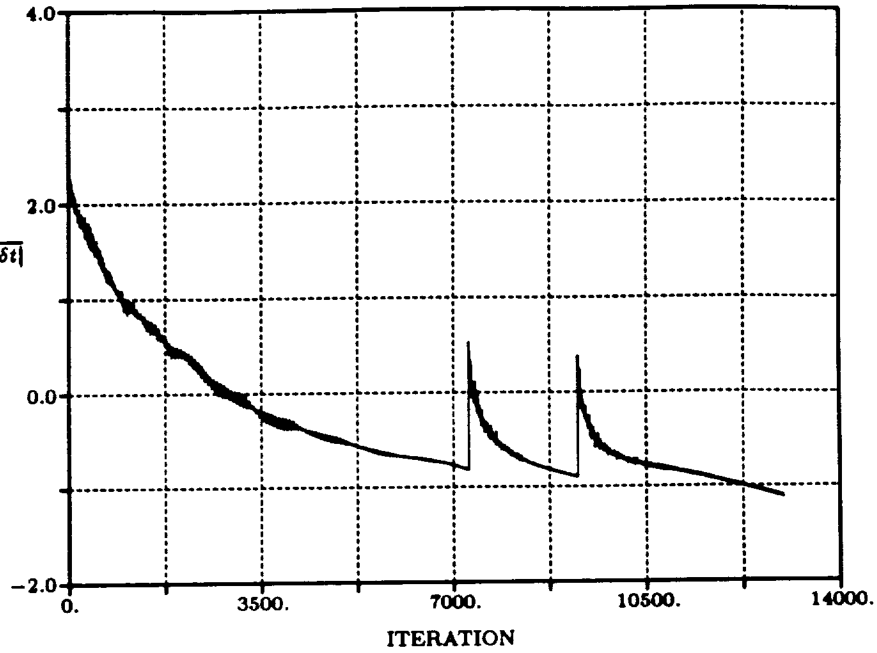


Figure 7.60 NLR two-element airfoil: average $|\delta(\rho u)/\delta t|$ verses iteration.

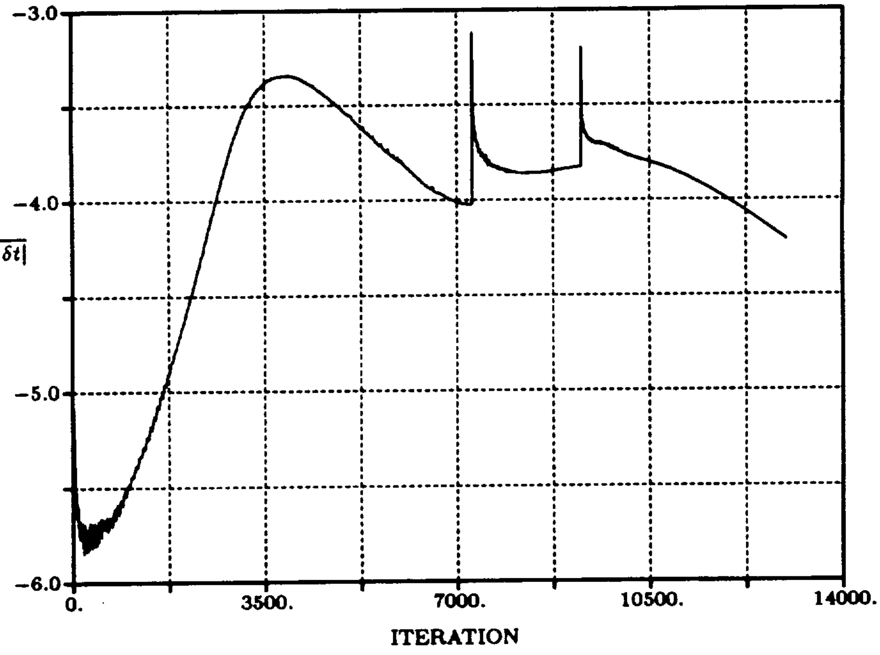


Figure 7.61 NLR two-element airfoil: average $|\delta Rt/\delta t|$ verses iteration.

7.4 VKI LS82-1 Turbine Cascade

This turbine cascade case is referred to as case 1 in the 1982 VKI Workshop [77]. Experimental results for the surface isentropic Mach number distribution and Schlieren photographs for different exit pressures are given by Sieverding, and Van Hove *et al.* [77]. The isentropic Mach number is defined as

$$M_{is} = \left[\frac{2}{\gamma - 1} \left[\frac{1}{(p/p_{t_{in}})^{\frac{\gamma-1}{\gamma}}} - 1 \right] \right]^{1/2} \tag{7.4}$$

The blade characteristics and the flow conditions for the results presented are summarized in Table 7.6. The exit pressure is measured in an area averaged fashion and is given in terms of the isentropic exit Mach number. The ratio of the area averaged exit static pressure to the inlet total pressure is 0.3012 for the present value of $(M_{is})_{exit}$. This flow condition is of particular interest because there is a strong oblique shock wave propagating to downstream boundary as shown in Schlieren photograph (Figure 7.68a). The total exit velocity is supersonic with a subsonic axial component. Therefore, one boundary condition is still needed at the exit boundary. In addition, the flow in this case is choked, making the boundary layer resolution very important. Note that experimental results are measured using a finite number of blades with tailboard placed on the upper boundary as shown in the Schlieren photograph. The use of tailboards results in reflected shock waves which propagate to the blade row. These reflected shock waves are not presented in an infinite cascade problem.

Table 7.6 Flow conditions for the VKI LS82-1 turbine cascade.

M_{in}	$(M_{is})_{exit}$	β_{in}	β_{exit}	T_∞ (° K)	Re_L	Pitch/chord
0.085	1.43	0.0°	−78.0°	278.0	2.51x10⁶	0.78
Note:	• Experimental data is available in Reference [77]. • Nonuniform pressure present at the exit.					

The initial mesh is given in Figure 7.62. The upstream and downstream mesh boundaries are placed one half an axial chord from the blade leading and trailing edges, respectively. At the upstream boundary total pressure, total temperature, and absolute flow angle are specified as boundary conditions. At the downstream boundary the area averaged exit static pressure is imposed as described in Chapter 3. No-slip condition, adiabatic wall, and zero normal pressure gradient are enforced on the blade and periodicity is imposed at the upper and lower boundaries. Specifying a normal wall mesh scale ($\delta_{y^+} = 6\text{x}10^{-5}$ chord) and a streamwise wall mesh scale ($N_s = 8$ and $\delta_w = 0.018$ chord) produces the initial O-type local structured mesh around the blade. In the far field the unstructured mesh is uniform with a length scale of $\delta_{\mathbf{max}} = 0.0385$ (pitch).

The sequence of solution adapted meshes are shown in Figures 7.62 to 7.65. The characteristics of each mesh are tabulated in Table 7.7. Figures 7.66 and 7.67 present the initial and final mesh solutions, respectively. The latter plot shows a good resolution of the oblique shock wave and viscous wake. Since the Schlieren photograph is based on the reflection of light passing through the flow, the density contours of the computed results are compared to the Schlieren photograph. Comparing the final mesh density contours shown in Figure 7.68b with the Schlieren photograph in Figure 7.68a shows an accurate prediction of the strong oblique shock wave location and orientation, as well as the weak passage shock wave. Detailed mesh and Mach contours at the blade trailing edge are presented in Figures 7.69 and 7.70. These plots demonstrate how the solution adaptive scheme resolves complex flow structures. Comparing the initial mesh surface Mach number distribution shown in Figure 7.71 with the final mesh solution in Figure 7.72 shows an improved solution as the mesh is refined. Figure 7.72 shows a slight difference between the numerical solution and experimental data on the uncovered portion of the blade suction surface. This is due to the interference of the reflected shock waves from the tailboard with the suction surface of blade 6 in the experimental data (see Figure 7.68a).

Although the solution converges after three mesh adaptations, further refinement is forced to improve the resolution of the weak shock wave. The 4^{th} adapted mesh is fine enough to resolve the vortex shedding at the blade trailing edge as shown in Figure 7.77. Since vortices are shed from the blunt trailing edge, this flow is unsteady. In order to obtain a realistic unsteady solution, the solution should be marched time accurately. This is too costly to be practical. In the present work a "pseudo" time averaged quantity is used to estimate the difference in loading coefficients between steady state and unsteady solutions. The term "pseudo" is used to distinguish it from the actual time average. Due to the use of local time stepping and implicit residual smoothing, the present calculation does not produce a time-accurate solution. The mesh in the unsteady wake flow region as shown in Figure 7.75 is quite uniform, therefore, the solution in this region may be nearly time-accurate. A "pseudo" time averaged loading coefficient can be estimated from the convergence histories of the solution on the 4^{th} adapted mesh. It is believed that the trailing edge shedding has a small effect on the loading coefficient. Since the loading coefficient is a major concern in practical applications it might be reasonable to accept the steady state solution. This is confirmed by the convergence histories for the axial and normal force coefficients shown in Figures 7.81 and 7.82, respectively. The fluctuating part of the loading coefficient is less than 1 % of the steady state or "pseudo" time averaged value. Comparing the 4^{th} adapted mesh solutions as shown in Figures 7.79 and 7.78 with the steady state solution in Figures 7.72 and 7.78 also shows little difference in the two solutions. The total CPU time for three mesh adaptations is about 2.2 hours of Cray-YMP CPU time.

Table 7.7 VKI LS82-1 turbine cascade: $(M_{\text{is}})_{\text{exit}} = 1.43$, $\beta_{\text{in}} = 0.0°$, and $Re_L = 2.0\text{x}10^5$.

MESH	0	1	2	3	4
nodes	4,610	5,212	7,783	11,588	13,291
elements	8,780	9,918	14,943	22,487	25,932
C_{fn_p}	20.3200	20.3499	20.4902	20.6563	
C_{ft_p}	91.2638	91.2557	91.3174	91.3652	
C_{fn_v}	0.0688	0.0706	0.0693	0.0672	
C_{ft_v}	-0.15789	-0.16353	-0.15597	-0.14954	
δ_{max}/pitch	0.0385	0.0385	0.0385	0.0385	0.0385
$\delta_{\text{max}}/\delta_{\text{min}}$	1	3.1	6.2	12.5	25
$\Delta y_{\text{wall}}(\text{x}10^5)$	3.0	2.55	2.43	2.33	2.34
y^+_{pr}	2	2.5	2.5	2.5	2.5
A_w	300	300	300	300	300
Cray-YMP CPU (sec)	2,432	1,255	1,277	2,838	3,194

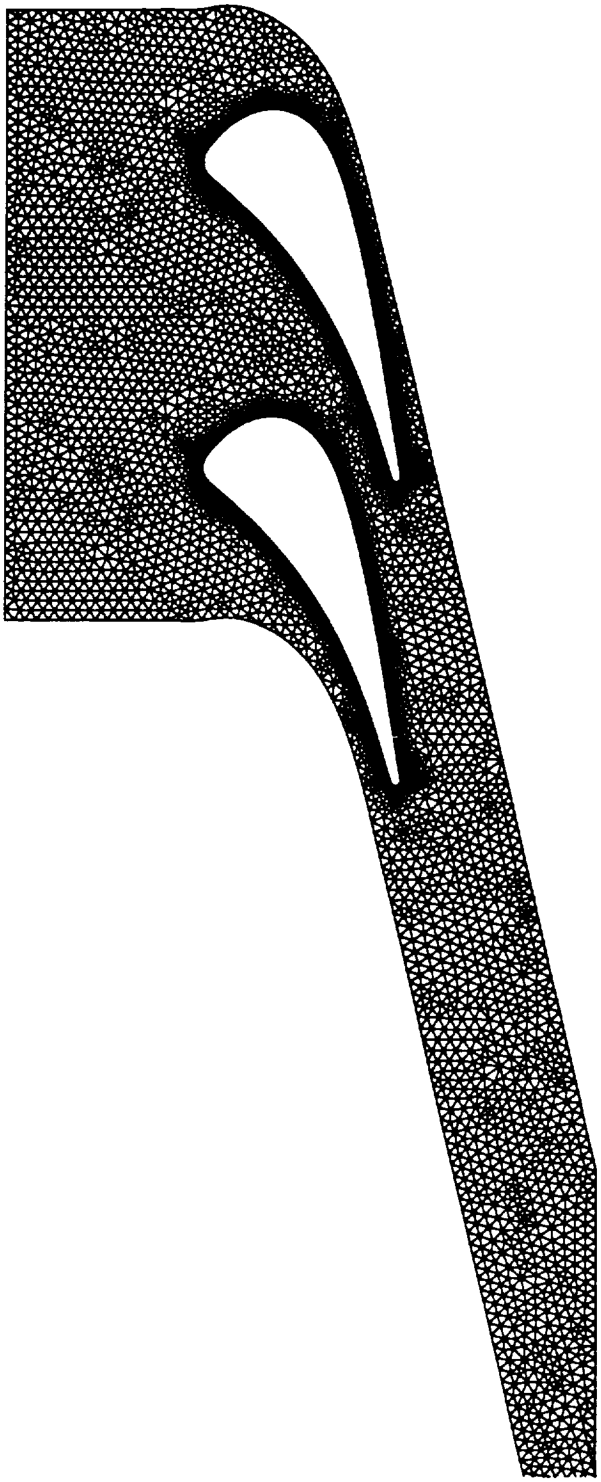


Figure 7.62 VKI LS82-1 turbine cascade initial mesh: 4,610 nodes and 8,780 elements.

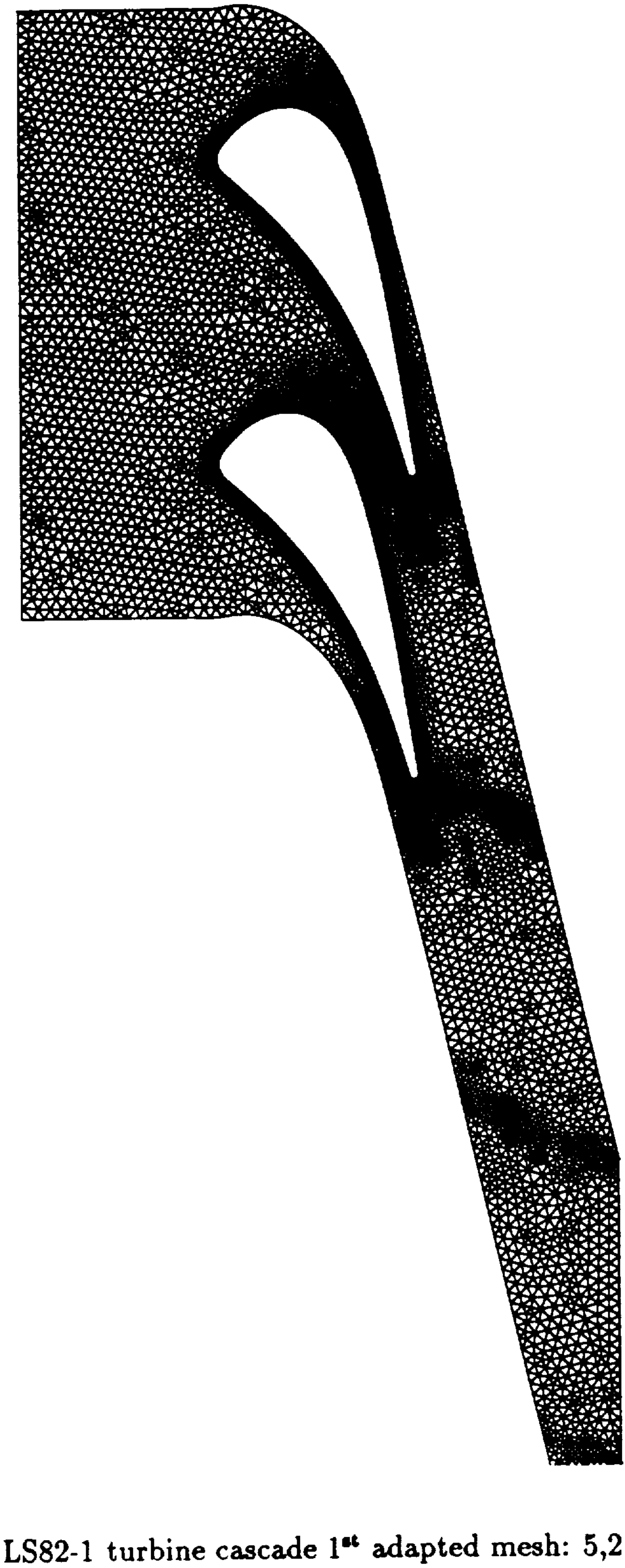


Figure 7.63 VKI LS82-1 turbine cascade 1st adapted mesh: 5,212 nodes and 9,918 elements.

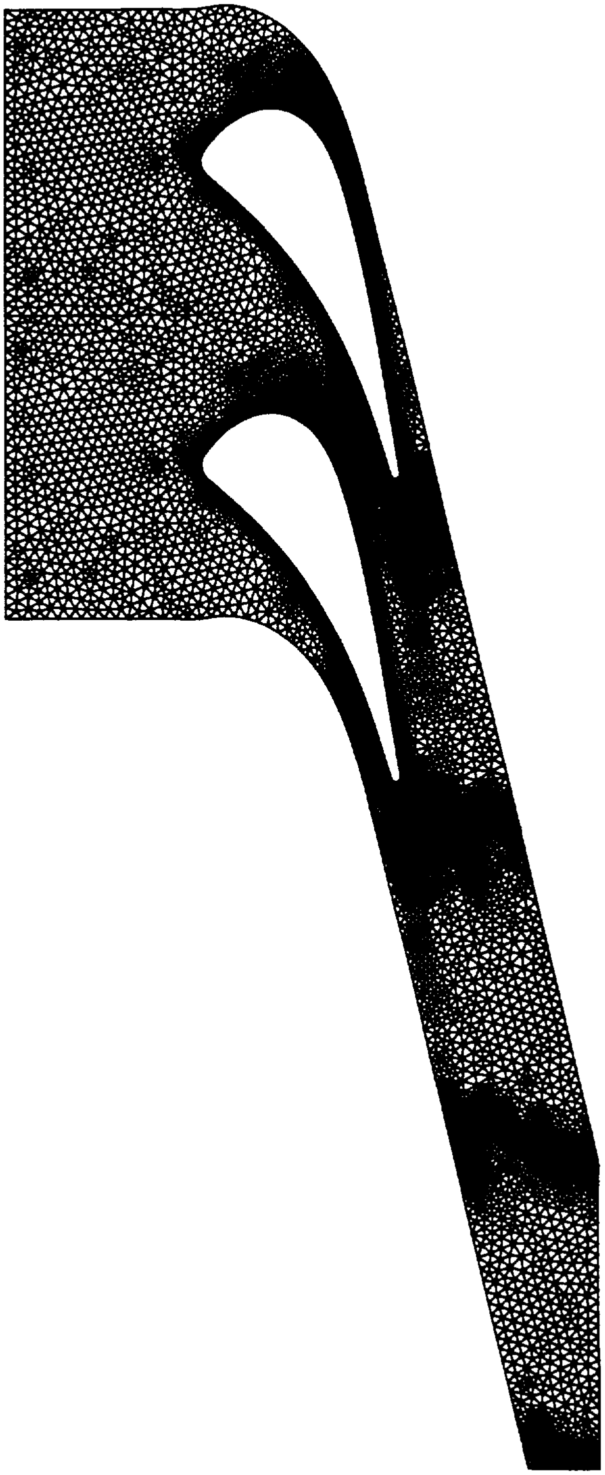


Figure 7.64 VKI LS82-1 turbine cascade 2nd adapted mesh: 7,783 nodes and 14,943 elements.

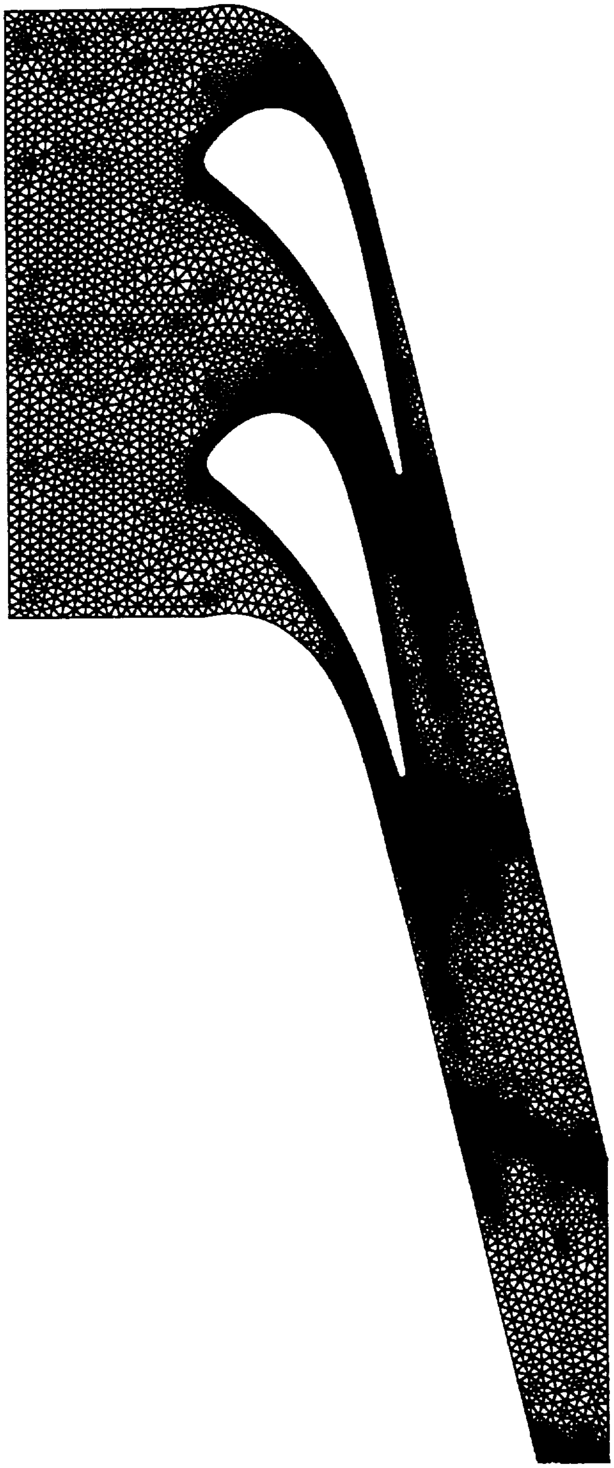


Figure 7.65 VKI LS82-1 turbine cascade 3^{rd} adapted mesh: 11,588 nodes and 22,487 elements.

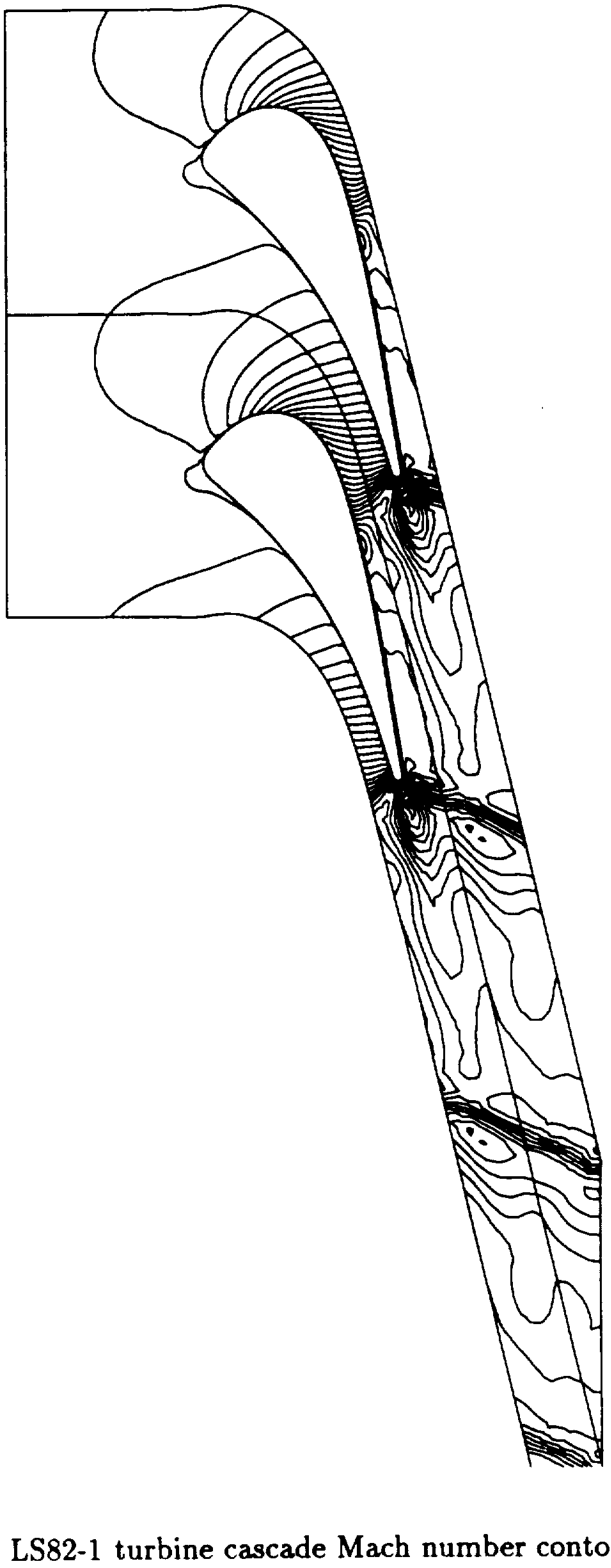


Figure 7.66 VKI LS82-1 turbine cascade Mach number contours on the initial mesh: cmin = 0.0, cmax = 1.8, and inc = 0.05.

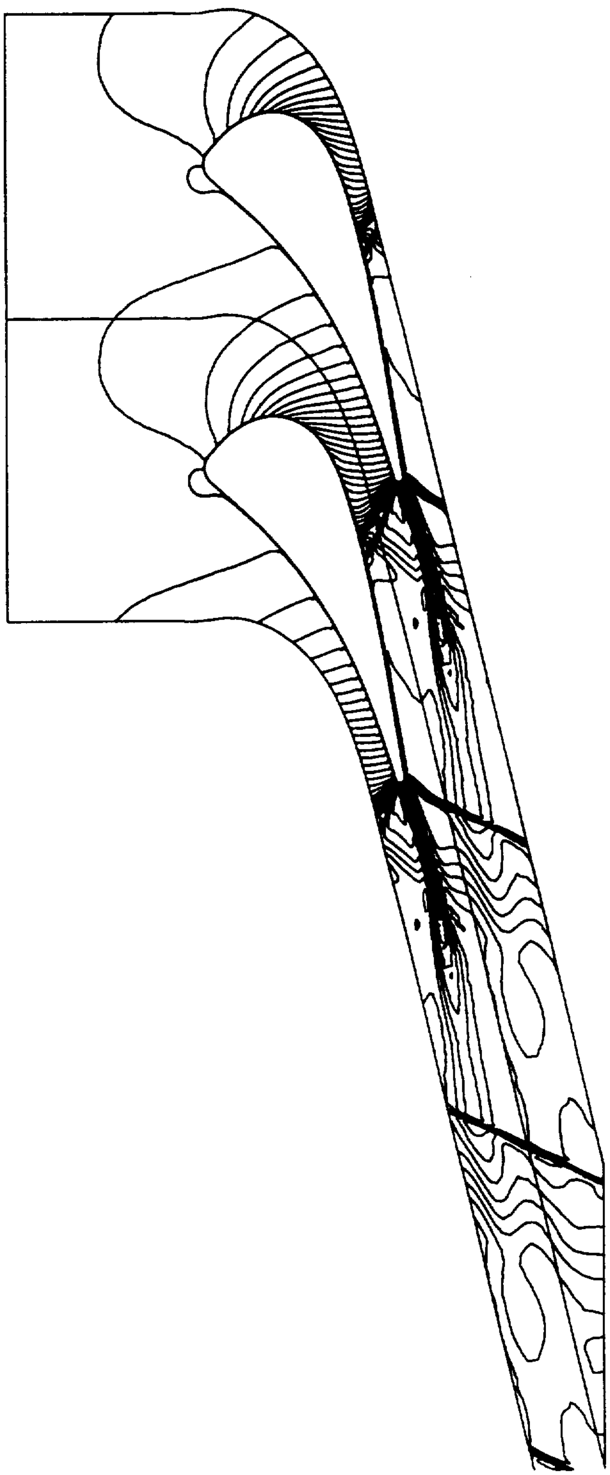


Figure 7.67 VKI LS82-1 turbine cascade Mach number contours on the 3rd adapted mesh: cmin = 0.0, cmax = 1.8, and inc = 0.05.

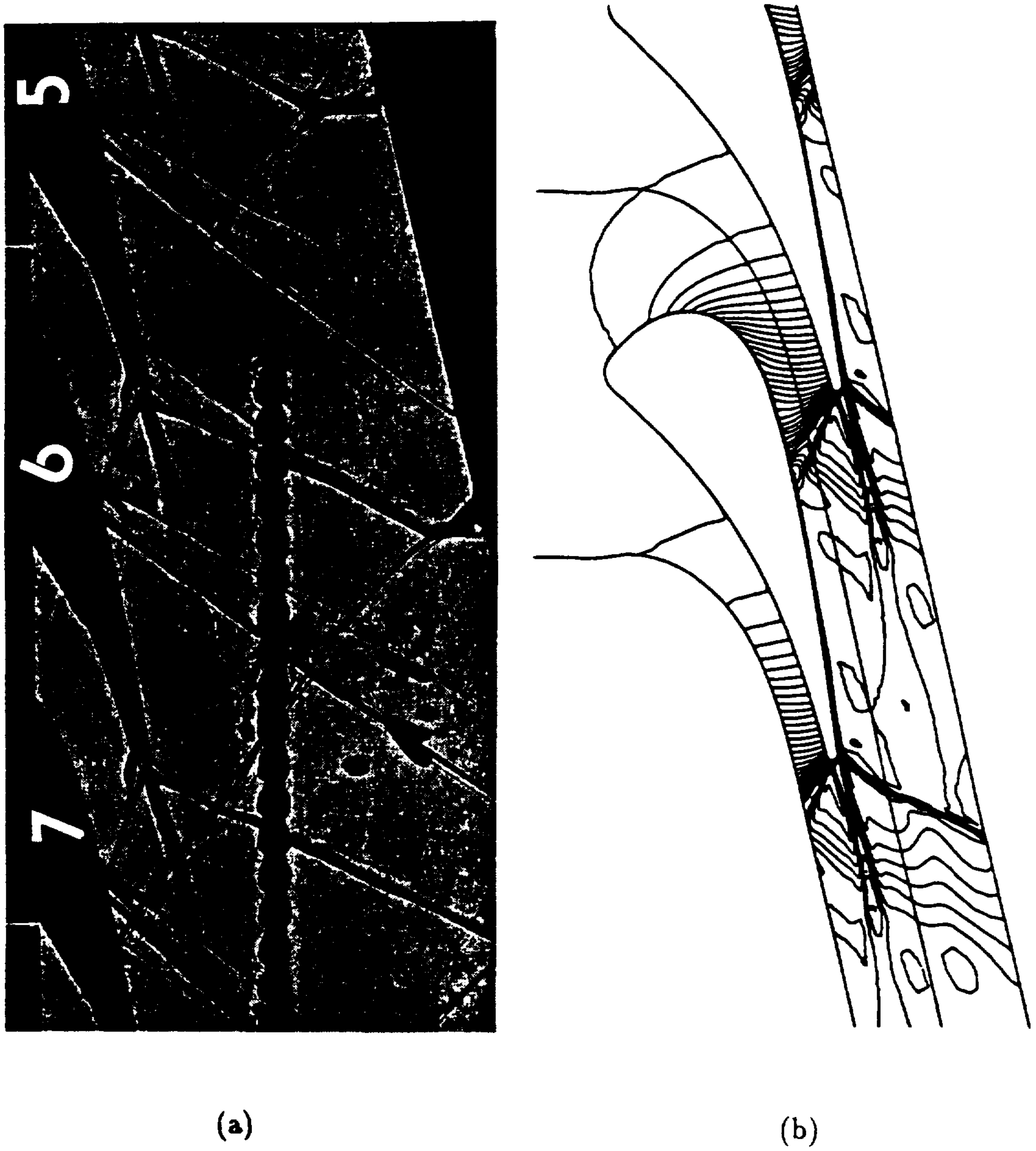


Figure 7.68 VKI LS82-1 turbine cascade: (a) Schlieren photographs (Sieverding [77]). (b) Density contours on the 3rd adapted mesh, cmin = 0.17, cmax = 0.99, and inc = 0.02.

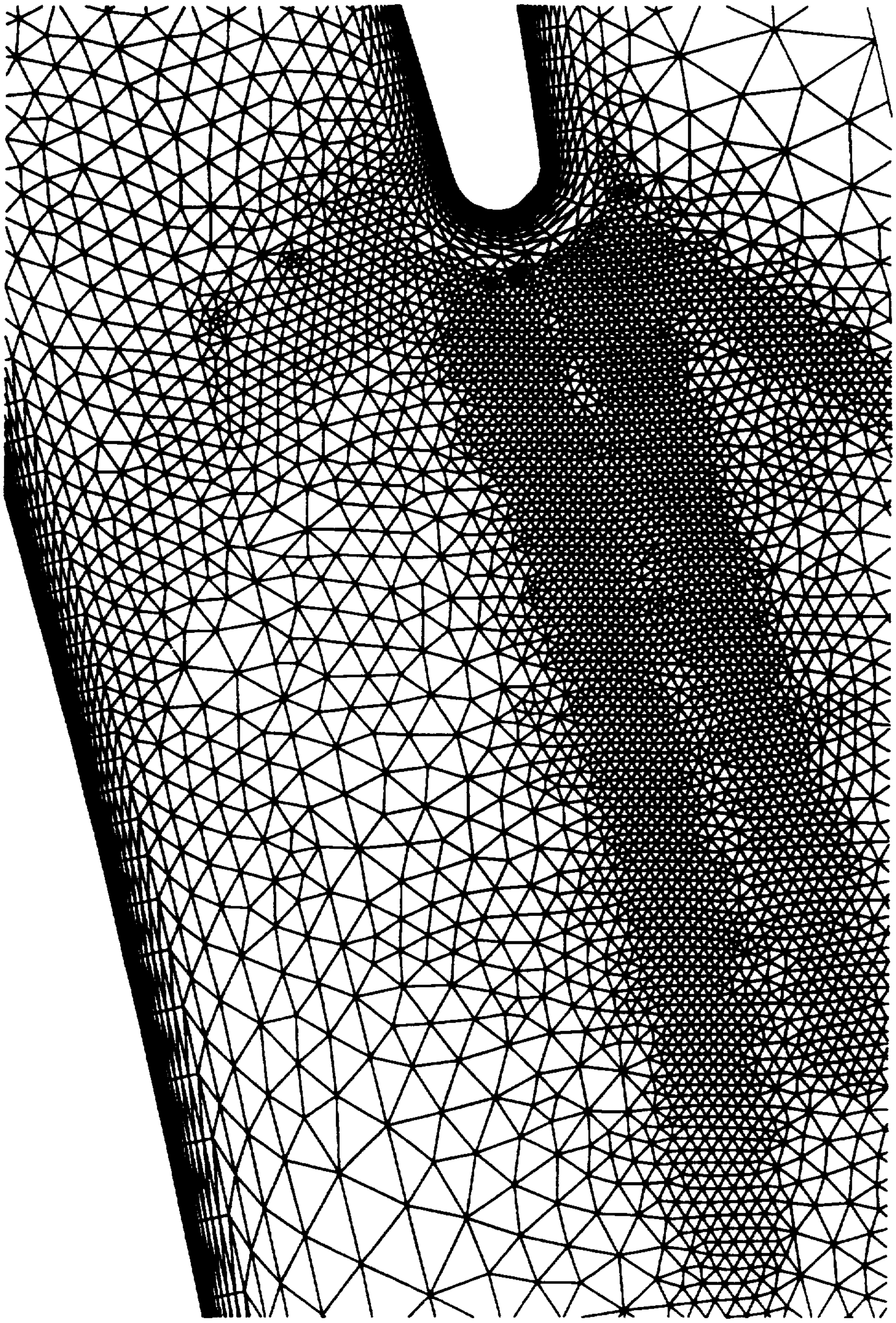


Figure 7.69 VKI LS82-1 turbine cascade: trailing edge blowup of the 3rd adapted mesh.

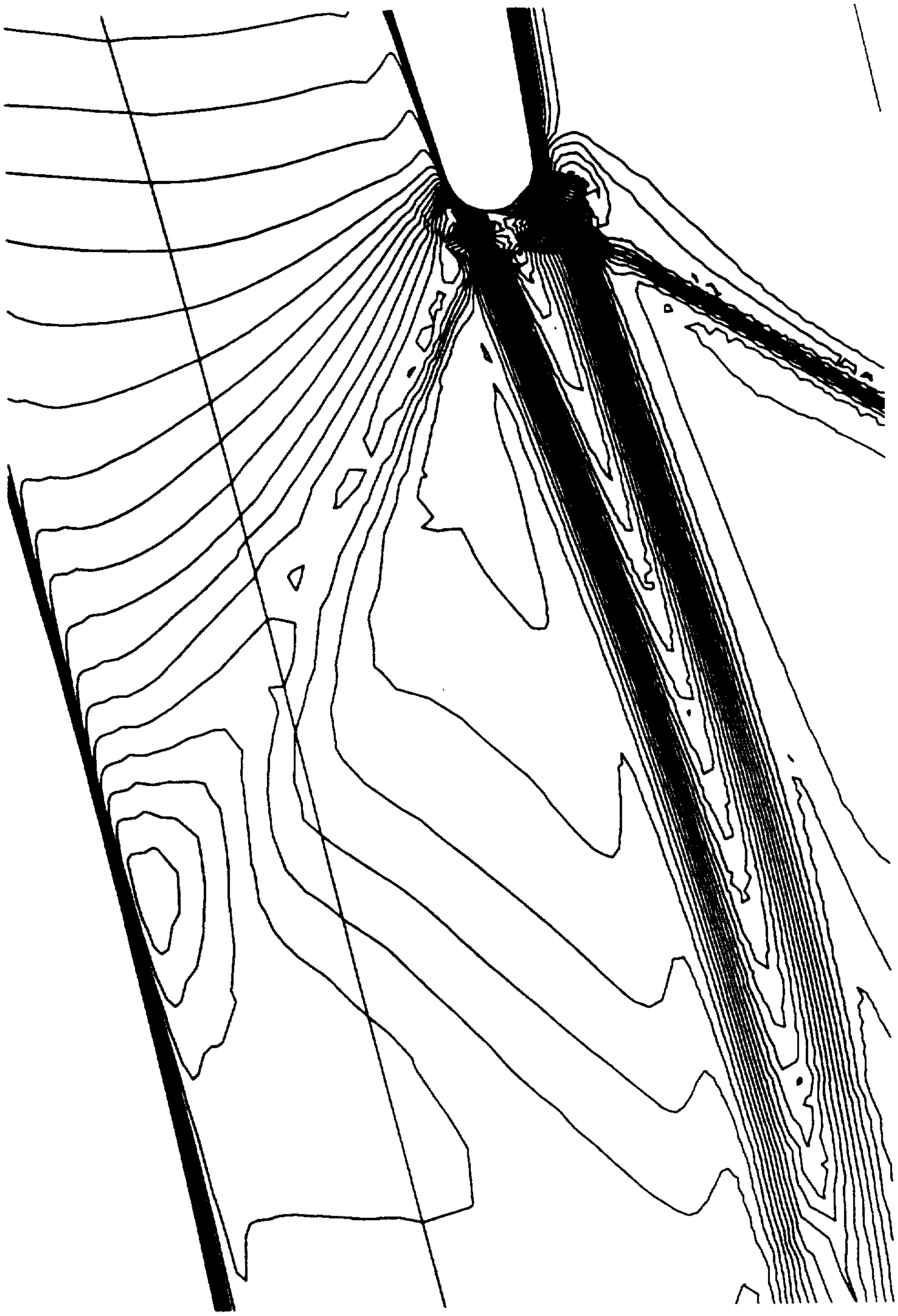


Figure 7.70 VKI LS82-1 turbine cascade Mach number contours (cmin = 0.0, cmax = 1.80, and inc = 0.05): 3rd adapted mesh trailing edge region.

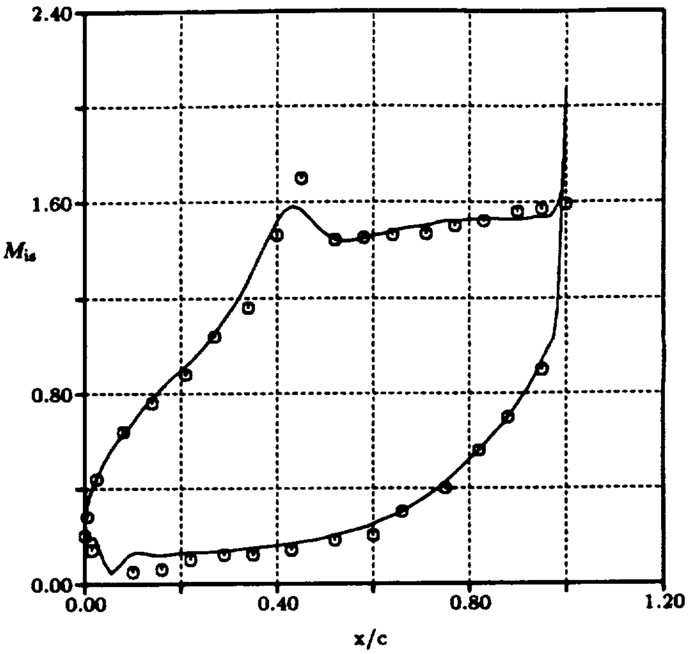


Figure 7.71 VKI LS82-1 turbine cascade surface isentropic Mach number for the initial mesh: solid line - numerical solution and symbol - experimental data.

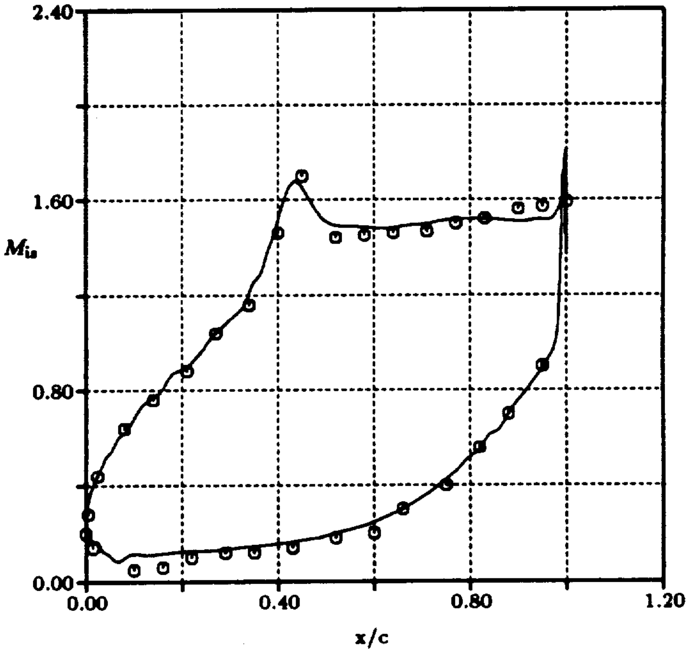


Figure 7.72 VKI LS82-1 turbine cascade surface isentropic Mach number for the 3rd adapted mesh: solid line - numerical solution and symbol - experimental data.

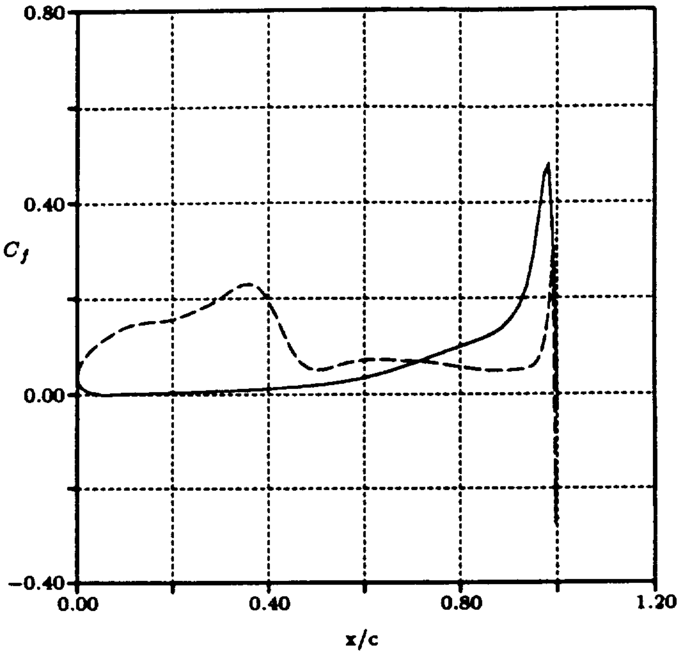


Figure 7.73 VKI LS82-1 turbine cascade surface friction coefficient for the initial mesh.

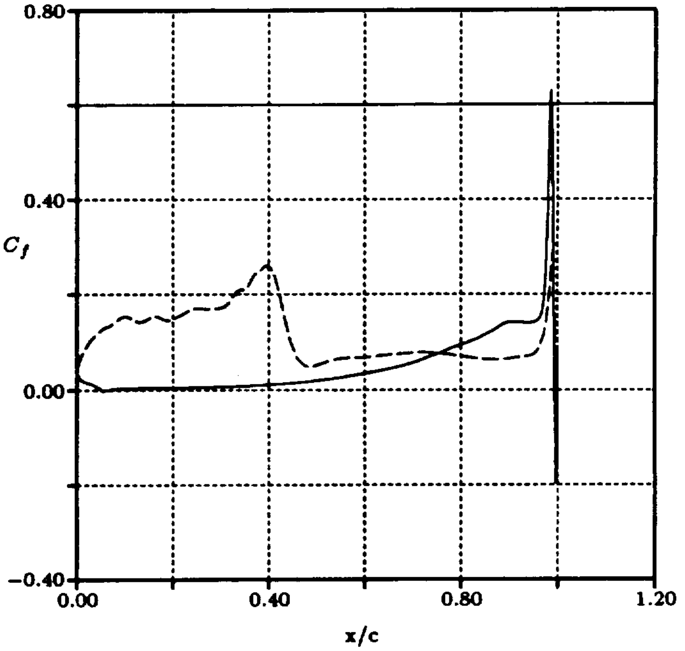


Figure 7.74 VKI LS82-1 turbine cascade surface friction coefficient for the 3rd adapted mesh.

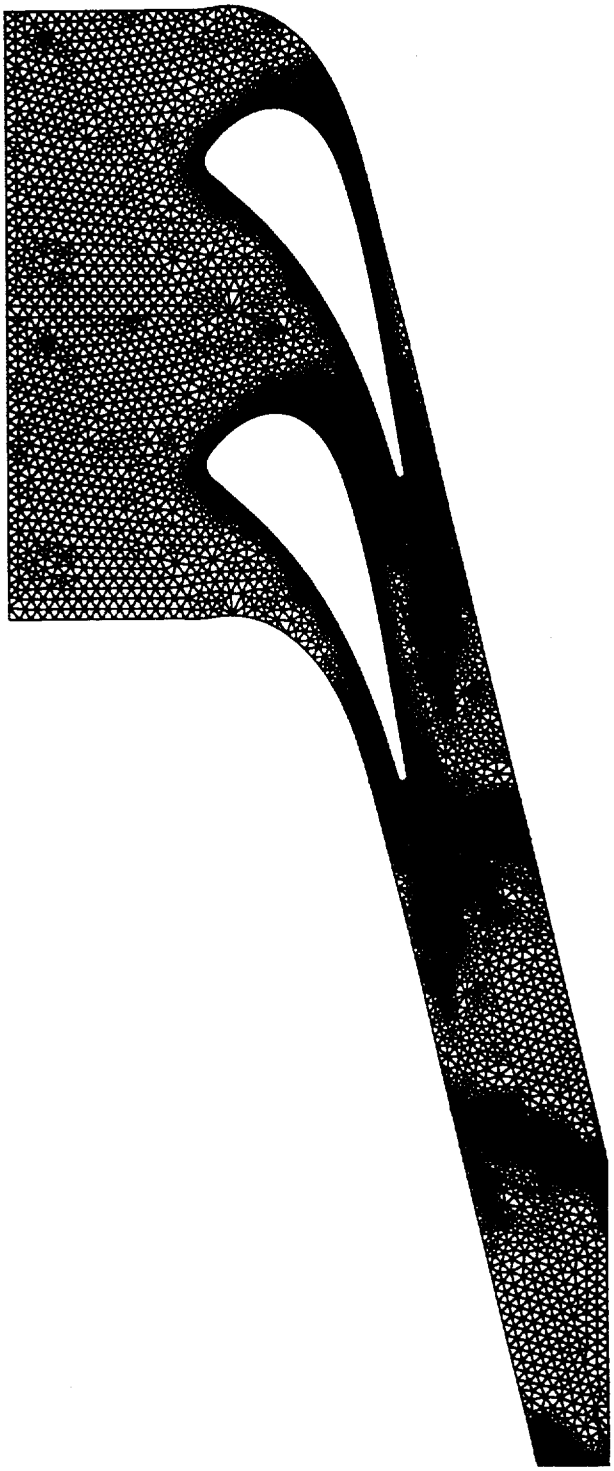


Figure 7.75 VKI LS82-1 turbine cascade 4^{th} adapted mesh: 13,291 nodes and 25,932 elements.

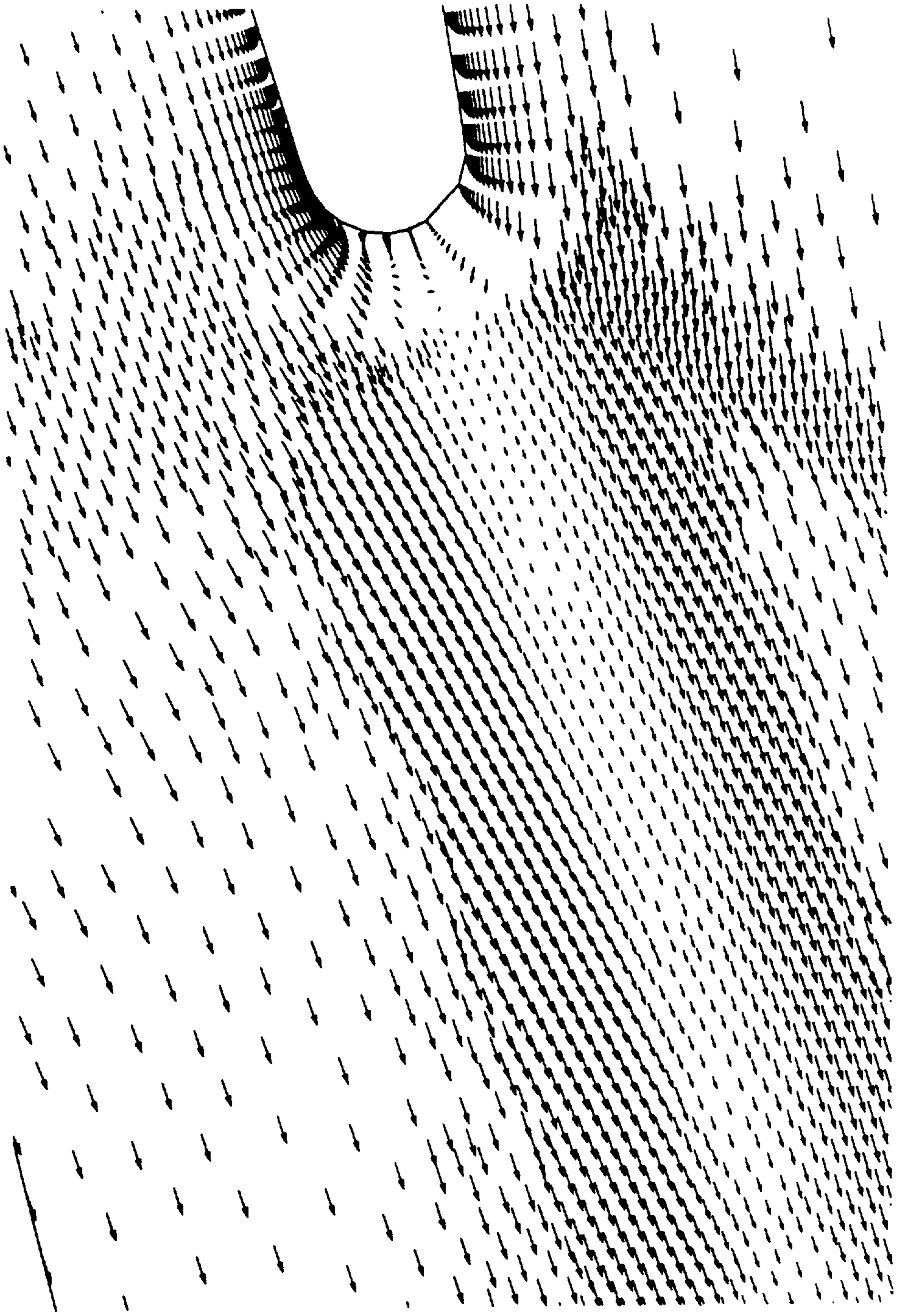


Figure 7.76 VKI LS82-1 turbine cascade velocity vector: 3rd adapted mesh trailing edge region.

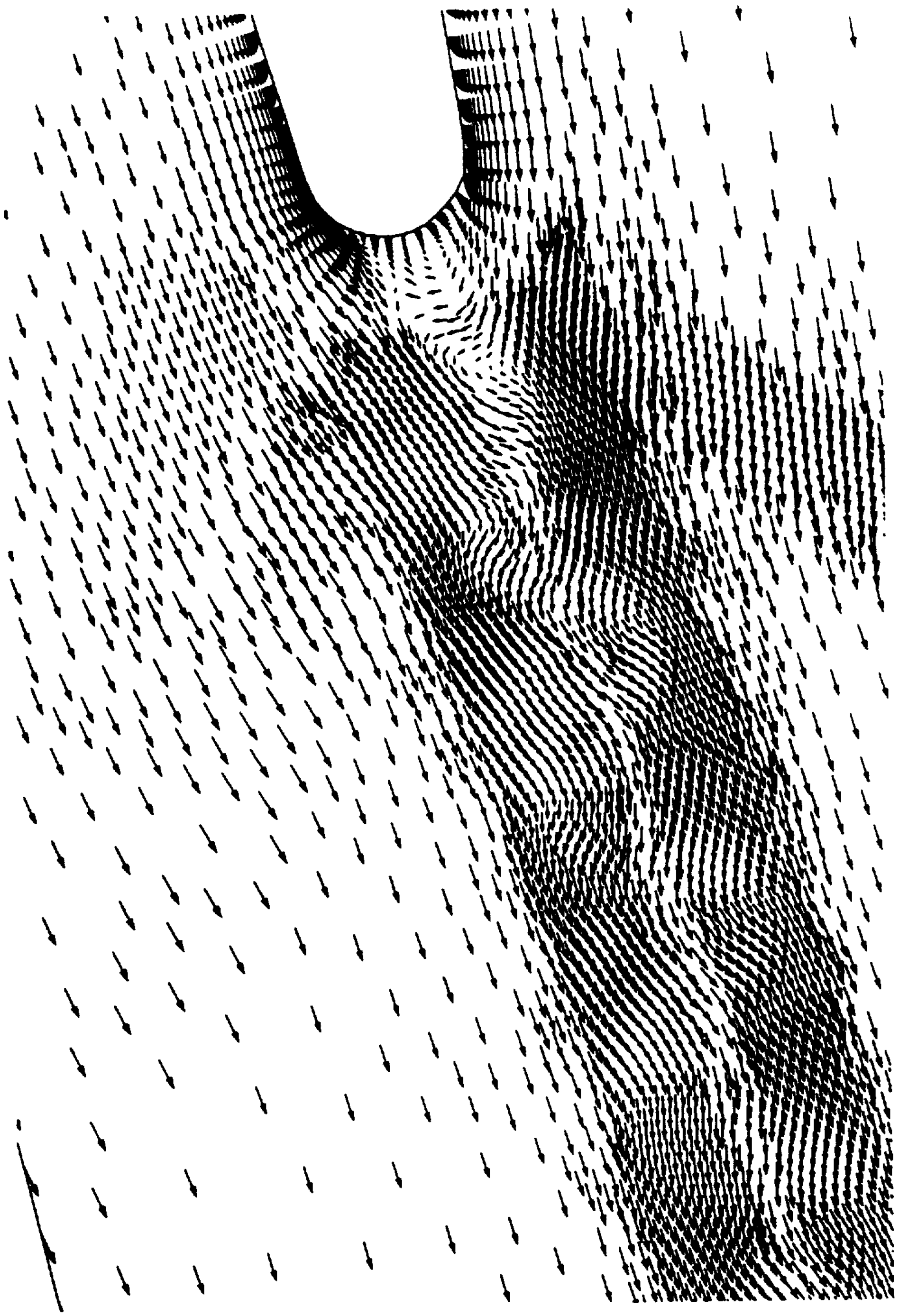


Figure 7.77 VKI LS82-1 turbine cascade velocity vector: 4th adapted mesh trailing edge region.

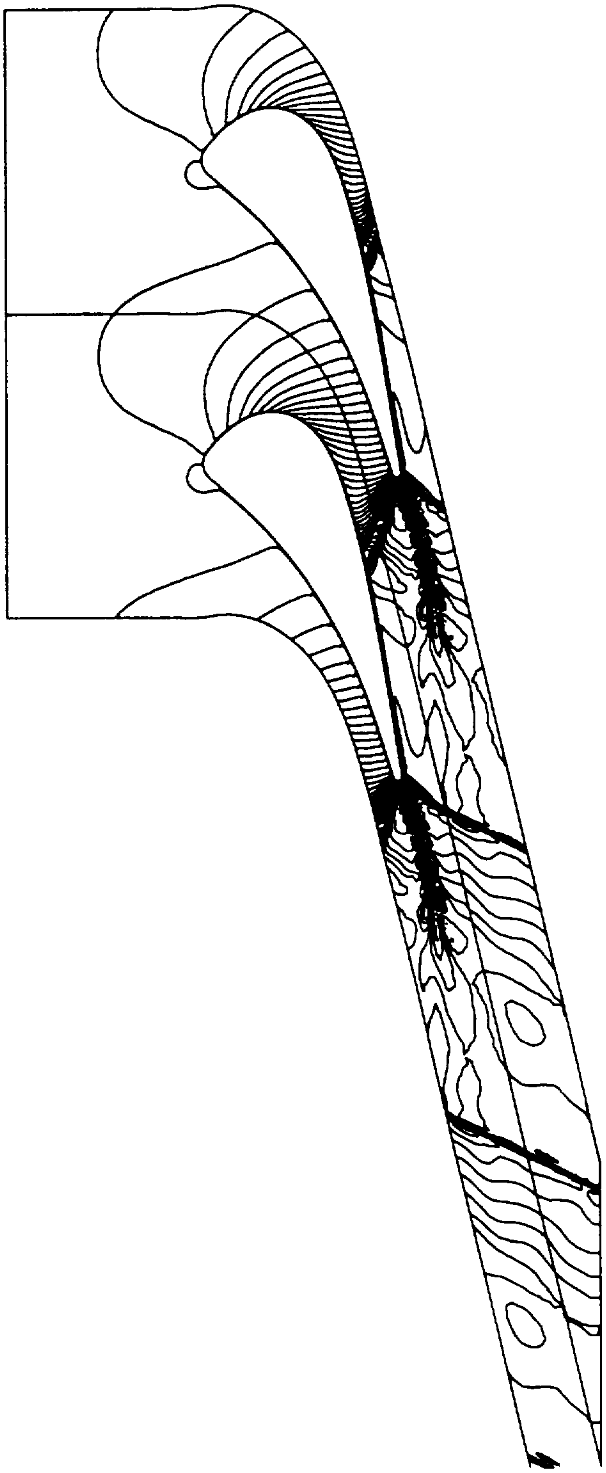


Figure 7.78 VKI LS82-1 turbine cascade Mach number contours on the 4th adapted mesh: cmin = 0.0, cmax = 1.80, and inc = 0.05.

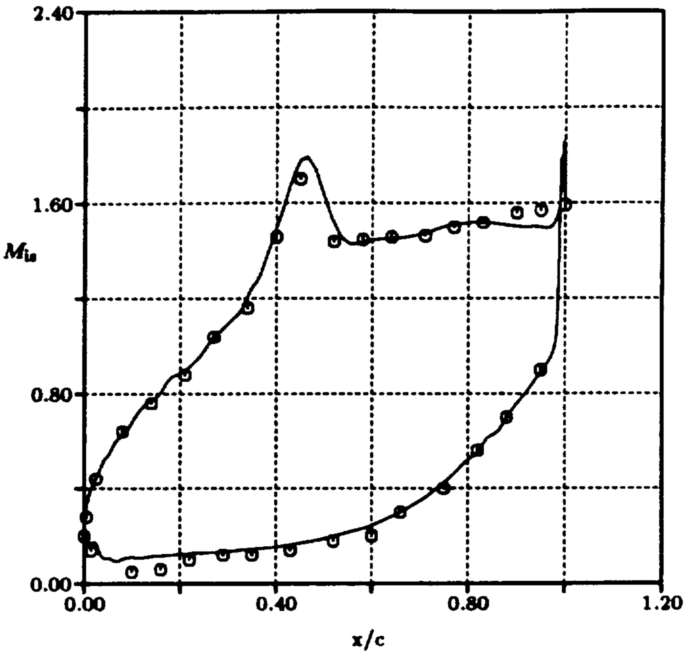


Figure 7.79 VKI LS82-1 turbine cascade surface isentropic Mach number for the 4th adapted mesh: solid line - numerical solution and symbol - experimental data.

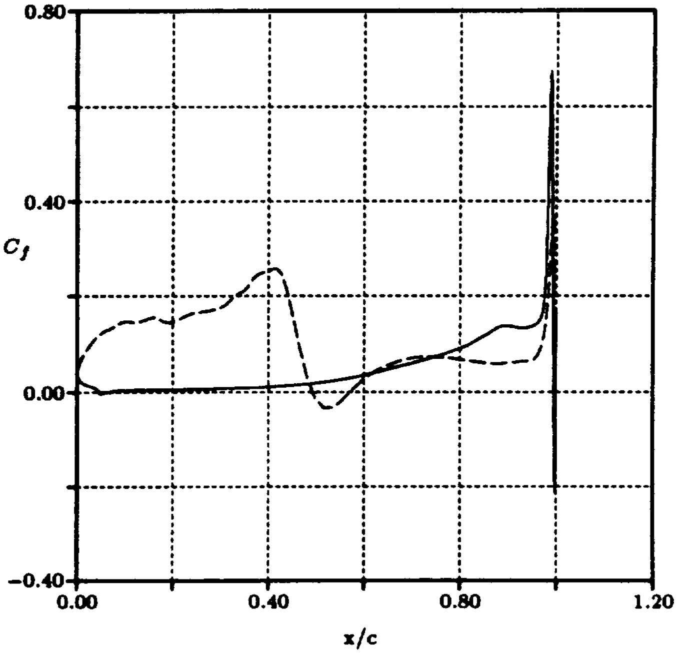


Figure 7.80 VKI LS82-1 turbine cascade surface friction coefficient for the 4th adapted mesh.

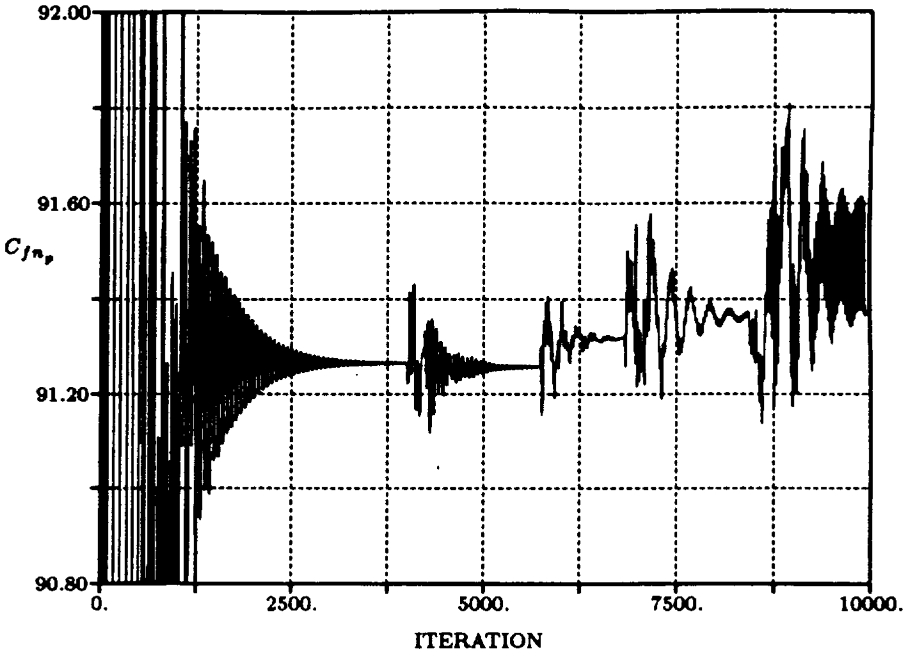


Figure 7.81 VKI LS82-1 turbine cascade: C_{ft_p} verses iteration.

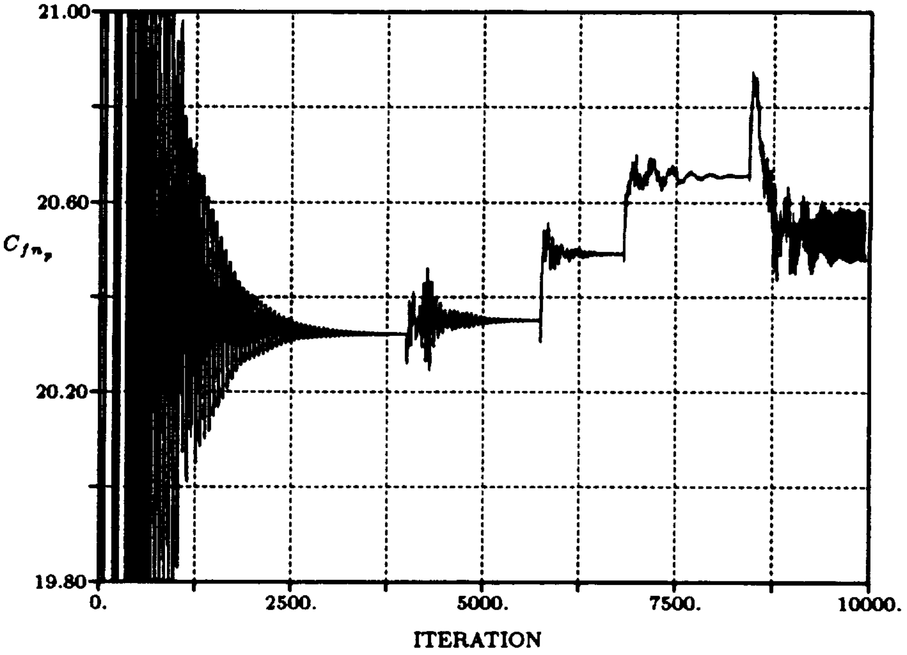


Figure 7.82 VKI LS82-1 turbine cascade: C_{fn_p} verses iteration.

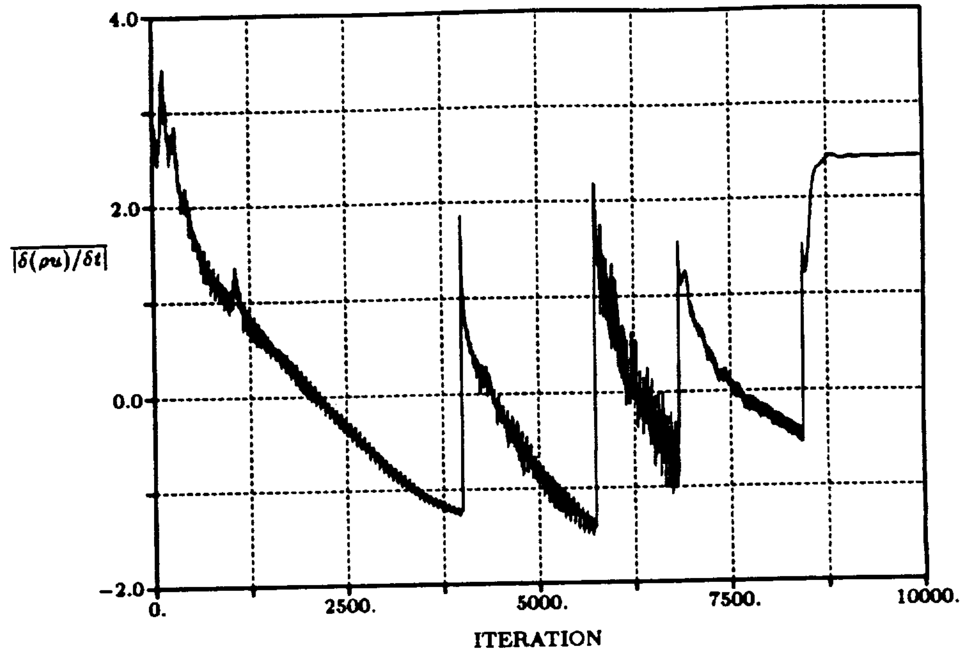


Figure 7.83 VKI LS82-1 turbine cascade: average $|\delta(\rho u)/\delta t|$ verses iteration.

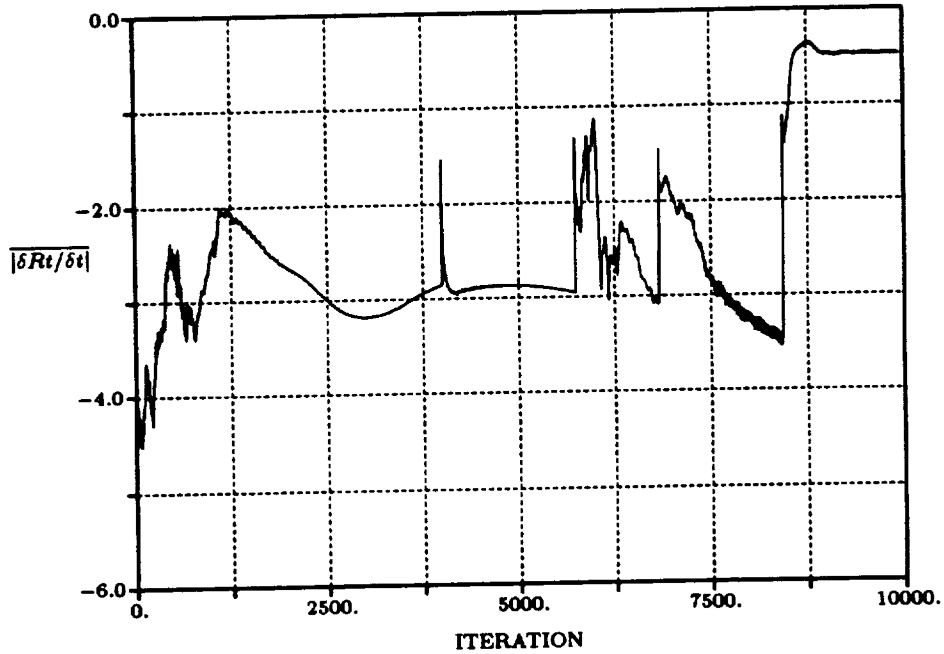


Figure 7.84 VKI LS82-1 turbine cascade: average $|\delta Rt/\delta t|$ verses iteration.

7.5 Allison Tandem Blade Cascade

This test case demonstrates the present solution adaptive scheme for a complex blade design. There is no experimental data available. Since the flow is three-dimensional, through-flow information is required as input to the present quasi-three-dimensional analysis. The streamsurface location and thickness data used in the present calculations were obtained from Reference [60]. The adapted mesh solutions presented here are for 70 % span station measured from the shroud. This is the same case for which the inviscid solution of section 6.6 was discussed. The inflow and outflow conditions are determined from the through-flow analysis. The exit static pressure is adjusted until a correct incident is obtained. The flow conditions for the 70 % span station are summarized in Table 6.12.

The initial mesh is given in Figure 7.85. The upstream and downstream mesh boundaries are located one axial chord from the blade edges. At the upstream boundary total pressure, total temperature, and absolute flow angle are imposed. At the downstream boundary the exit static pressure is specified. At blade surfaces a no-slip condition with adiabatic wall and zero normal pressure gradient is imposed. The initial O-type local structured mesh is constructed using a normal wall mesh scale δ_{y^+} = 9.48x10^{-5} chord and a streamwise wall mesh scale (N_s = 20 and δ_w = 0.055 chord). In the far field the unstructured mesh is uniform with a length scale of $\delta_{\max}$ = 0.046 chord.

The initial mesh solution is massively separated on the suction surface starting from 80 % of chord of the first blade to all the way downstream as shown in Figure 7.87. Similar separated flow solutions have been obtained by another investigator [24]. Comparing the turbulent flow results shown in Figure 7.87 with the

Table 7.8 Allison tandem blade cascade far field flow conditions.

Case	$P_{t_{in}}$(psi)	$T_{t_{in}}$ (R)	$r\,v_\theta$	$p_{exit}/P_{t_{in}}$	β_{in}	R_{eL}
70 % span	74.5	872.7°	0	0.721	0.0°	5.3 x 10^5

inviscid flow results in Figure 6.61 shows the importance of modeling the viscous effects.

It is meaningless to perform a further mesh adaptation on this kind of flow structure. The mesh adaptation is performed one time to illustrate how the solution adaptive scheme resolves such separated flow structures. Figure 7.94 shows mesh refinment on the separated and wake flow regions. The present mesh refinement parameters do a good job in resolving viscous flow structures. The mesh statistics and force coefficient information is summarized in Table 7.9. This case demonstrates that the unstructured flow solver can easily produce solutions for flow problems which are difficult to predict using structured mesh approach.

Table 7.9 Allison tandem blade cascade 70 % span station: $M_{\text{exit}} = 0.71$, $\beta_{\text{in}} = 0.0°$, and $Re_L = 5.3\text{x}10^5$.

MESH	0	1
nodes	5,417	9,367
elements	10,481	18,186
C_{fn_p}	0.7479	0.7629
C_{ft_p}	0.3917	0.3634
C_{fn_v}	0.0114	0.0075
C_{ft_v}	-0.0142	-0.0203
$\delta_{\text{max}}/\text{chord}$	0.046	0.046
$\delta_{\text{max}}/\delta_{\text{min}}$	1	4
$\Delta y_{\text{wall}}(\text{x}10^6)$	3.1	7.3
y^+_{pr}	2.0	1.0
A_w	600	600

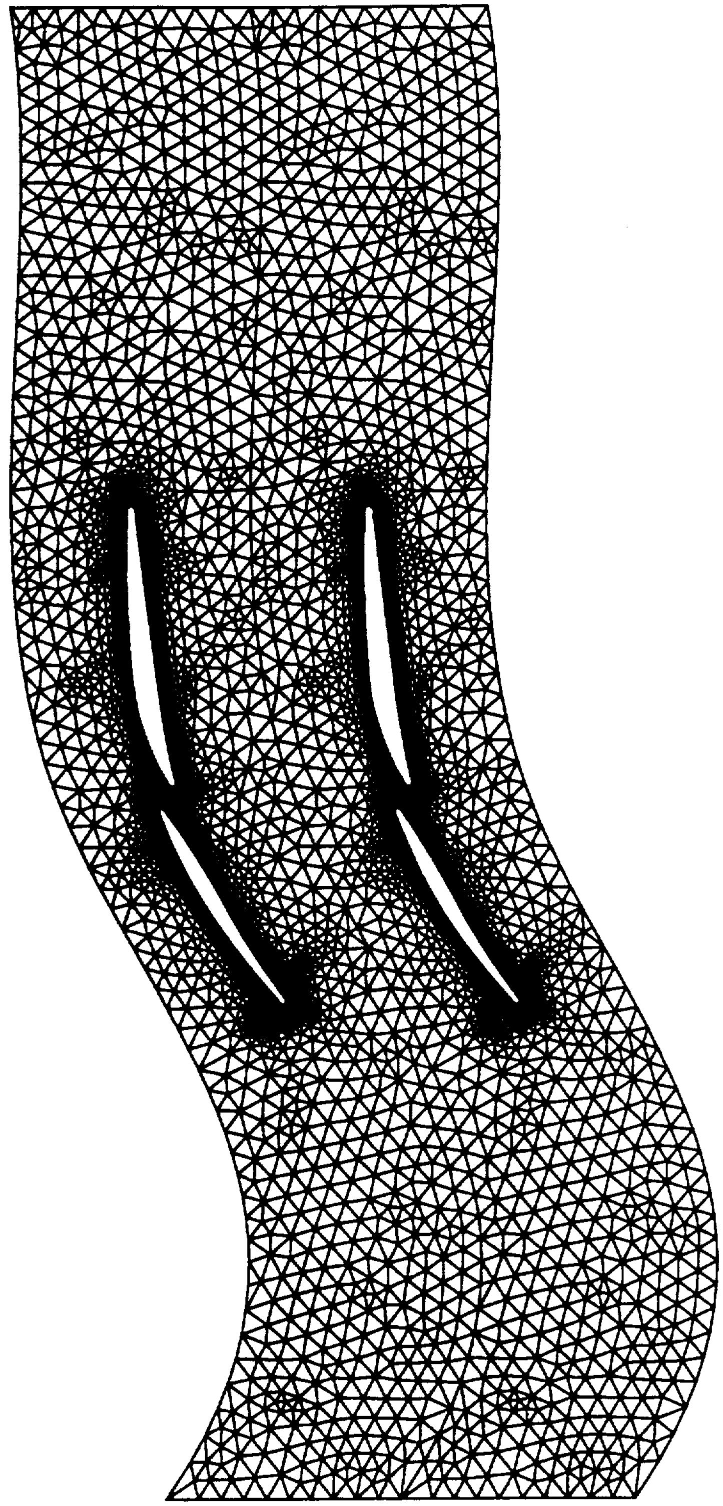


Figure 7.85 Allison tandem blade cascade 70 % span from shroud initial mesh: far field view.

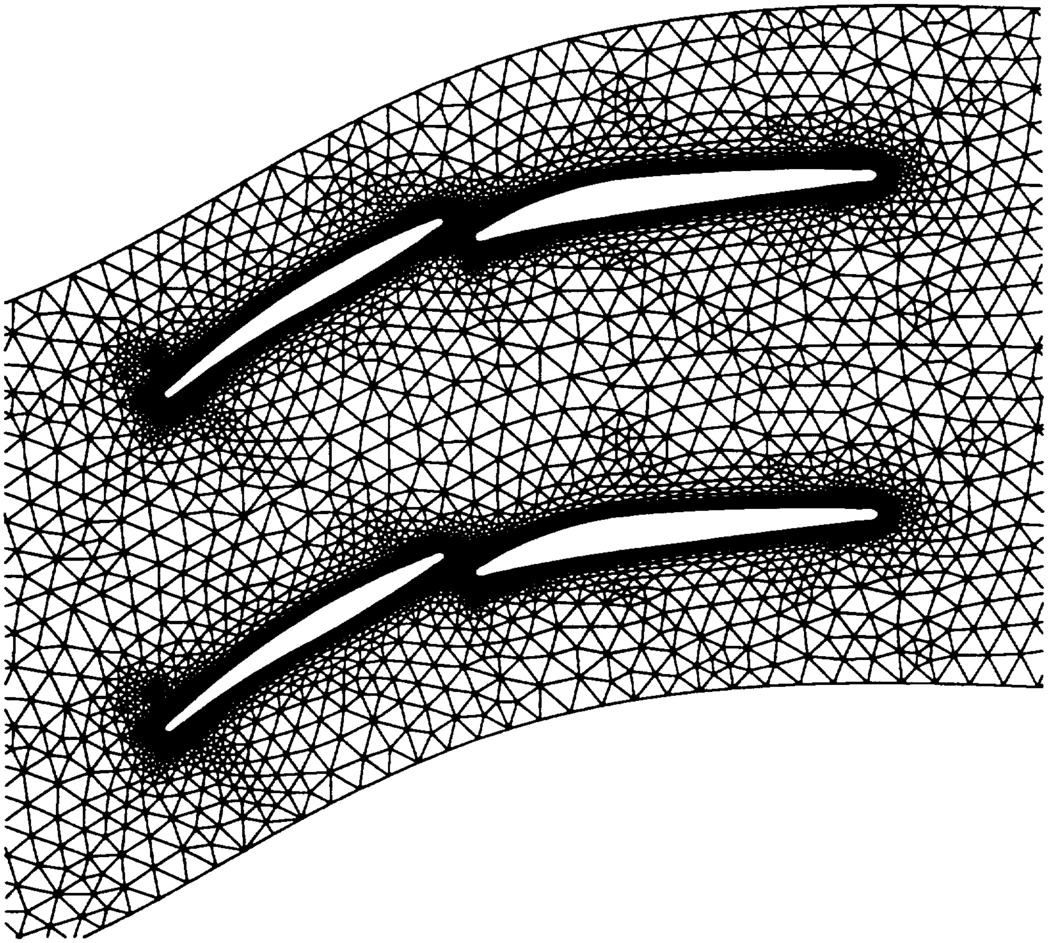


Figure 7.86 Allison tandem blade cascade 70 % span from shroud initial mesh: 5,417 nodes and 10,481 elements.

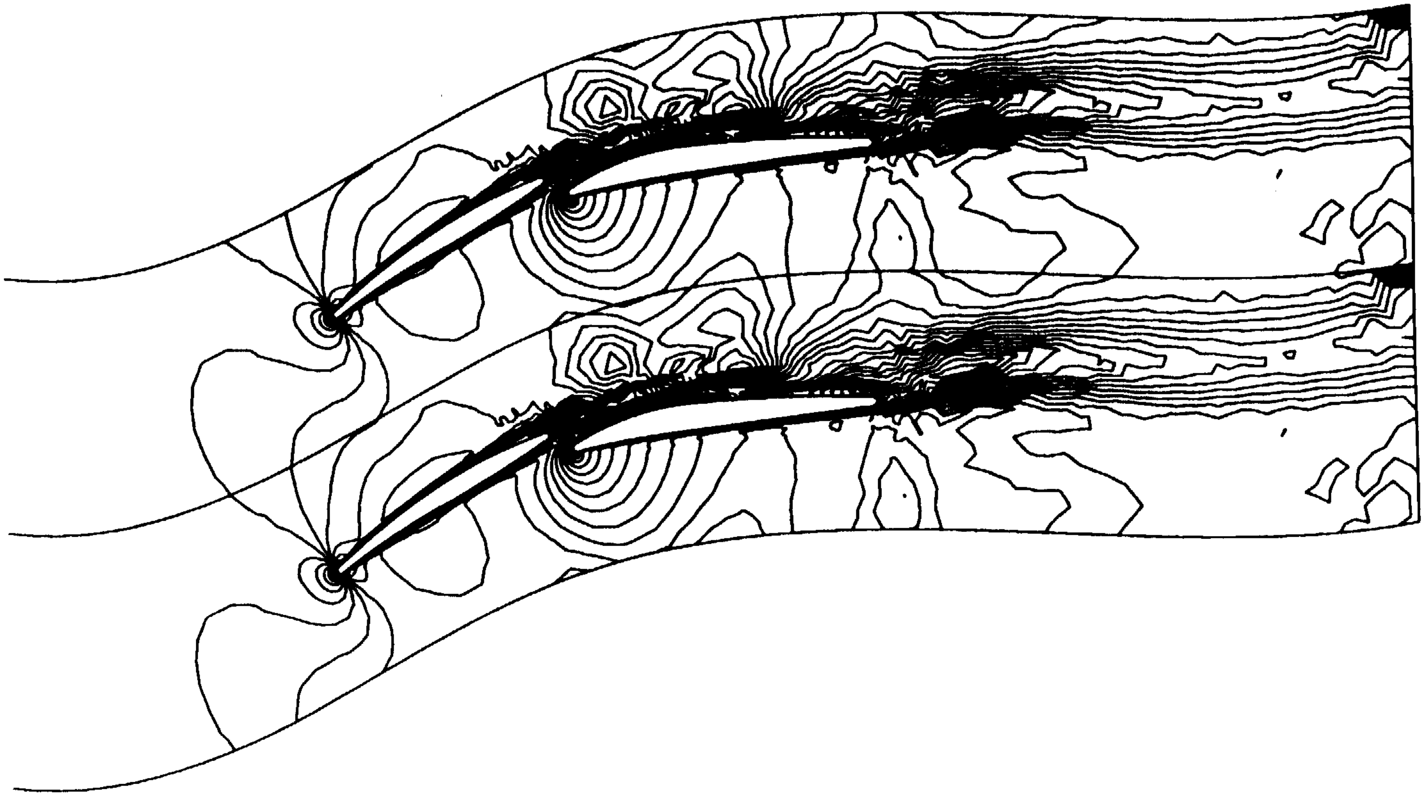


Figure 7.87 Allison tandem blade cascade 70 % span from shroud Mach number contours on the initial mesh: cmin =0.0, cmax= 1.45, and inc = 0.05.

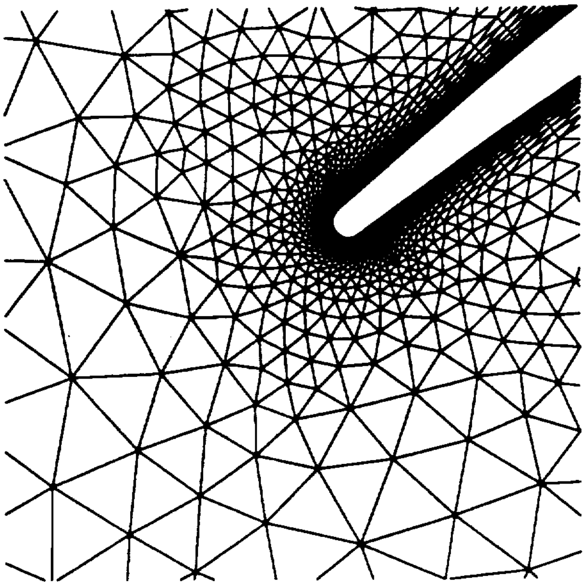


Figure 7.88 Allison tandem blade cascade 70 % span from shroud: leading edge blowup of the initial mesh.

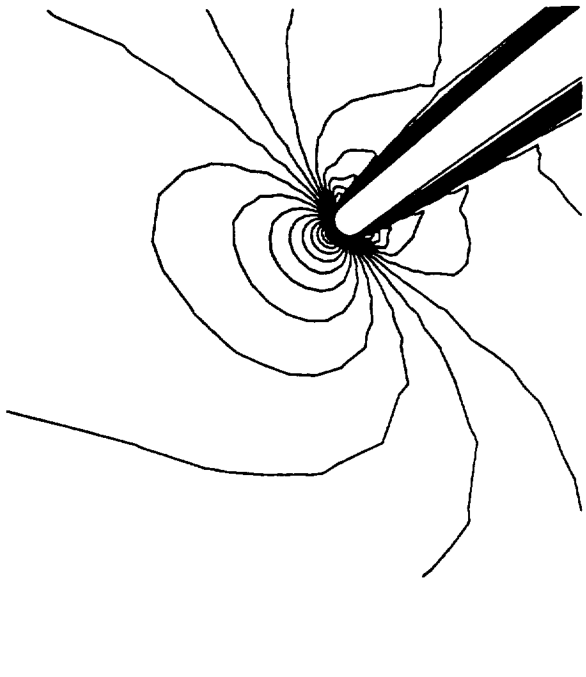


Figure 7.89 Allison tandem blade cascade 70 % span from shroud Mach number contours (cmin = 0.0, cmax = 1.45, inc = 0.05): initial mesh leading edge region.

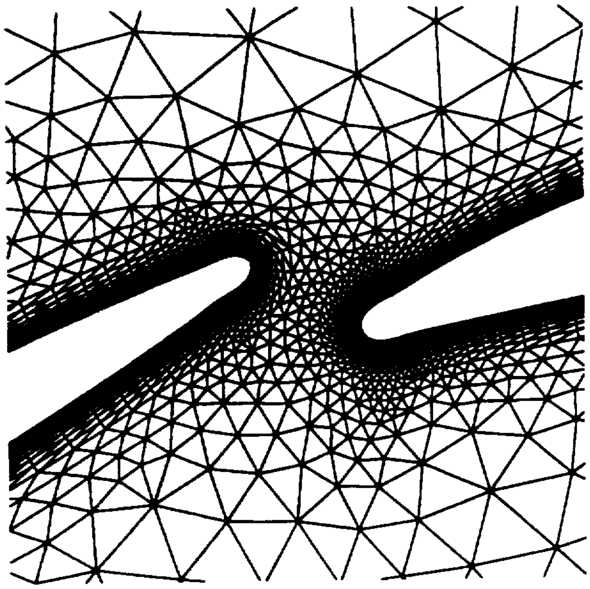


Figure 7.90 Allison tandem blade cascade 70 % span from shroud: tandem blade gap blowup of the initial mesh.

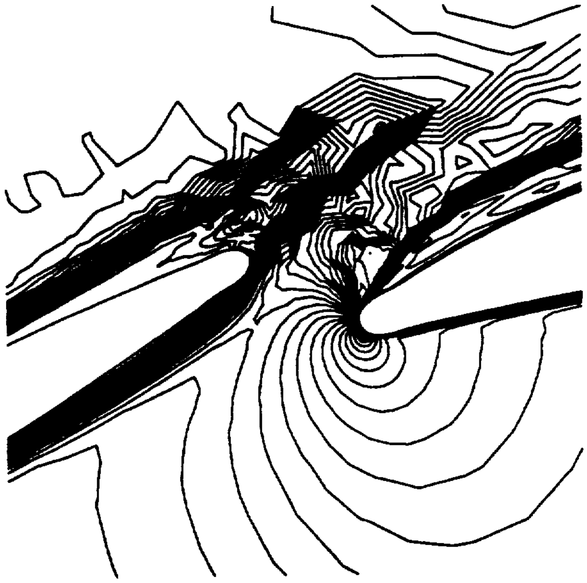


Figure 7.91 Allison tandem blade cascade 70 % span from shroud Mach number contours (cmin = 0.0, cmax = 1.45, inc = 0.05): initial mesh tandem blade gap region.

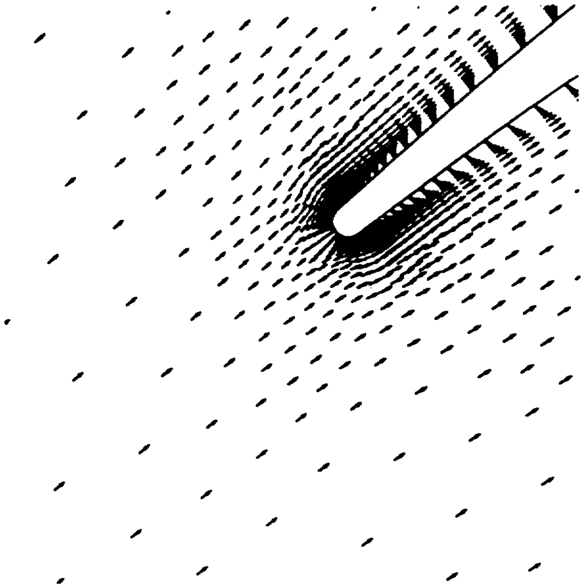


Figure 7.92 Allison tandem blade cascade 70 % span from shroud velocity vector: initial mesh leading edge region.

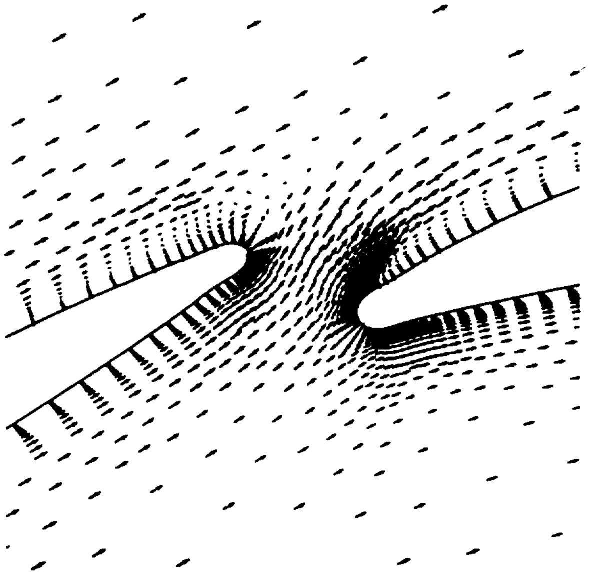


Figure 7.93 Allison tandem blade cascade 70 % span from shroud velocity vector: initial mesh tandem blade gap region.

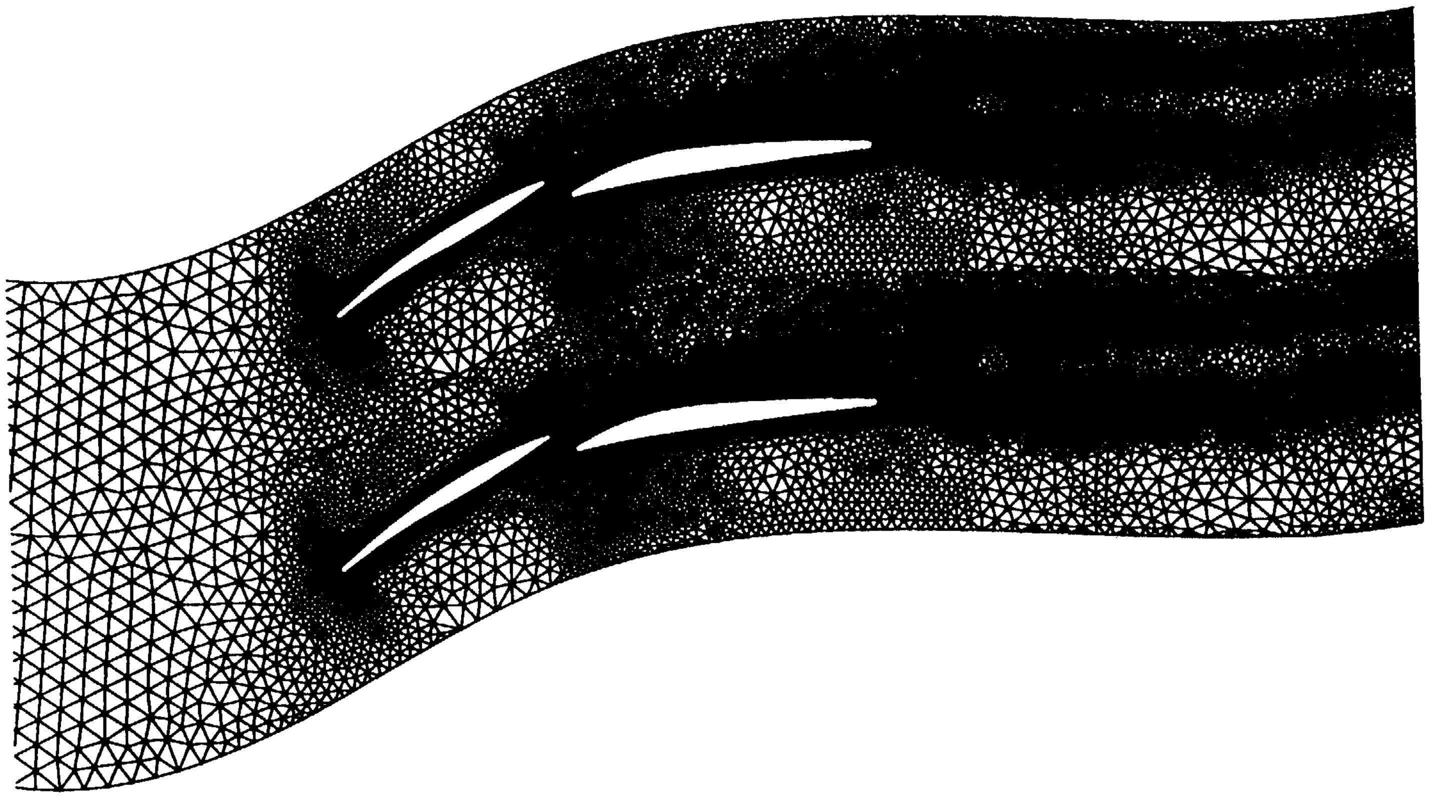


Figure 7.94 Allison tandem blade cascade 70 % span from shroud 1st adapted mesh: 9,367 nodes and 18,186 elements.

8. SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

A general solution adaptive unstructured remesh scheme has been implemented and verified for a wide range of both inviscid and viscous two-dimensional and quasi-three-dimensional turbomachinery flows. The solution adaptive scheme used here incorporates an explicit finite-volume Runge-Kutta time-marching flow solver and an advancing front mesh generation scheme. The mesh is adapted by periodic remeshing as the solution evolves. In the present approach, an edge-based local coordinate system has been introduced for the formulation of the stability criteria and eigenvalue scaling of the artificial dissipation. A general stability analysis of an unstructured scheme is not possible due to the unstructured mesh connectivity. By employing the edge-based coordinate system, an analogy to the structured formulations is obtained. To enhance the convergence rate to steady state, an improved iterative method for solving the implicit residual averaging operator is formulated which allows larger local time steps to be used than those obtained from standard Jacobi iterative method.

In the solution adaptive scheme, new mesh refinement parameters have been derived based on a combination of the local surface curvature and the gradient of flow speed. The coupling of geometric and flow field information results in an accurate and efficient adaptation criterion for problems with complex flow structures and complex geometries. The present work uses a local structured triangular mesh within viscous flow regions. The solution adaptive remesh scheme is general and is not restricted to this local structured mesh.

The major contributions of the present research are summarized below.

- A solution adaptive unstructured remesh algorithm has been developed for turbomachinery applications.
- New mesh refinement parameters have been formulated for an accurate and efficient resolution of complex flow problems.
- Improvements to the basic cell-vertex finite-volume Runge-Kutta scheme have been made in the following areas:
  - Introduction of an edge-based local coordinate system for the formulation of the stability criteria and eigenvalue scaling artificial dissipation.
  - Development of an improved iterative method for the implicit residual averaging equation.
- Demonstration of the excellent performance of the solution adaptive unstructured remesh scheme.

8.1 Summary of Results

The solution adaptive remesh scheme has been demonstrated with the solution of various two-dimensional and quasi-three-dimensional inviscid and turbulent flows. Each of these cases has different flow features which need to be resolved for an accurate solution. In the cases presented, the adaptive remesh scheme was used to automatically resolve the flow features in the region of shock waves, expansion waves, stagnation regions, viscous layers and viscous wakes. These cases illustrates the automatic detection of the location, the size and the orientation of the important flow features through the use of the solution adaptive remesh algorithm. The minimum mesh scale of the sequence of solution adapted meshes is user-specified and can be defined in an efficient fashion to enhance the convergence of mesh adaptation cycles. This leads to a great advantage over the mesh refinement approach where mesh cells are subdivided in a constant ratio of 2.

Flow problems of arbitrary multiply-connected regions are no more difficult to solve than flow over an isolated body as shown by multi-element airfoil and tandem blade cascade cases. Small finite radius leading and trailing edge regions are also resolved as shown in the NASA Rotor 67 30%-span station case (Figure 6.51). These cases clearly illustrate the great flexibility of the solution adaptive unstructured remesh scheme for geometrically complex flow problems.

Even when complex flow structures develop within the solution domain, the solution adaptive remesh algorithm accurately resolves them as shown in the Denton supersonic wedge cascade and the VKI turbine cascade cases. These complex flow structures are very difficult to be accurately resolved using structured mesh approaches. Comparing the final solutions with the corresponding analytical or experimental data has shown that the important flow structures are accurately resolved without any prior knowledge. There is also a great saving in the computational effort and storage requirements since mesh points are not wasted in regions where they are not required. Furthermore, any level of the accuracy of solutions can be achieved as the mesh continues to be refined.

The effect of artificial dissipation on the solution of turbulent flows has also been studied. While the adaptive artificial dissipation works well in the solution of inviscid flows, it causes unacceptable errors in the solution of turbulent flows as shown in the flat plate case. Shutting off artificial dissipation within boundary layer regions produces an accurate prediction of the turbulent flow, but it is impossible to turn off the artificial dissipation completely for general flow problems. With proper scaling of the artificial dissipation, it has been shown that these errors can be greatly reduced.

In a preliminary study of directional mesh adaptations, it has been found that the advancing front method produces highly distorted meshes when mesh cell aspect ratios are greater than 20. The aspect ratio limitation in the advancing front method limits its use in viscous flow regions since high aspect ratio meshes are required for efficient viscous flow calculations. In the present work, a local structured mesh is used in viscous regions. Adopting a local structured mesh in viscous flow regions yields several advantages over a simple unstructured mesh approach. First, high aspect ratio mesh cells are easily achieved. This in turn yields an efficient mesh point distribution for the solution of turbulent flow problems. Second, viscous mesh scale is easily controlled, which reduces the possible stability problem caused by insufficient resolution of the near-wall region for turbulent flows.

For turbomachinery cascade calculations, repeated mesh refinement resolves vortex shedding at blunt trailing edges as shown in the VKI turbine cascade case. Since vortex shedding is an unsteady phenomena, a time-accurate solution scheme is required to obtain the correct solution. However, the present scheme can be marched time-accurately, this requires an enormous computational effort. Since the fluctuation in force coefficients is less than 1% of the steady state or pseudo time averaged force coefficients, the steady state solutions computed for this case are considered a good engineering approximation to the real flow.

The present solution adaptive scheme has been shown to accurately predict complex turbulent flows, although the explicit time-marching scheme requires a great amount of computational time for solving turbulent flow problems due to the requirement of fine meshes in the near-wall regions. Nevertheless, the results demonstrate the capability of the present scheme in analyzing the complex turbomachinery problems which can not be solved using classic structured mesh approaches (e.g., plate splitter, tandem blade, etc.).

8.2 Future Work

This thesis has successfully demonstrated a flexible solution adaptive remesh scheme for the accurate solution of complex flows. Further research in the following areas is suggested to enhance the efficiency of the present approach:

1. Multi-grid acceleration:

   Multi-grid acceleration should be incorporated into the present solver. Multigrid methods have been demonstrated for unstructured triangular mesh Euler and Navier-Stokes solutions by Mavriplis [54]. While multi-grid has been found to be more computationally intensive for triangular mesh formulations due to the the complexity of interpolation and projection operations, there is still a significant reduction in the overall computational work.

2. Alternate flow solvers:
   For viscous flow calculations the computation time becomes extremely large due to the stability criteria of the present Runge-Kutta scheme. This may be overcome through the adoption of an implicit formulation. There has been some promising work in this area by Batina [10], Barth [7], and others, but much work remains to be done.

3. Directional adaptation:
   Since many flow structures, such as viscous layers and shock waves, involve multiple length scales where the change in flow properties along one direction is much larger than the change in other directions, it is inefficient to use equilateral triangles over the domain. A directional adaptation method, which allows high aspect ratio triangles to be aligned with the flow features, will further reduce the computation expense and storage requirements encountered in these regions. The present mesh generation scheme is capable of generating directionally adapted mesh with cell aspect ratios as high as 20. With further development of the refinement parameters this could be incorporated in inviscid flow regions. The high aspect ratio meshes required for efficient viscous flow calculations require a new unstructured mesh generation scheme.

REFERENCES

PRECEDING PAGE BLANK NOT FILMED

REFERENCES

[1] J. D. Anderson, Jr., *Fundamentals of Aerodynamics*, pages 546–547, 2nd Edition, McGraw-Hill, New York, N. J., 1991.

[2] W. K. Anderson, J. L. Thomas, and D. L. Whitfield, "Multigrid Acceleration of the Flux Split Euler Equations," AIAA Paper No. 86-0274, January 1986.

[3] F. Angrand, A. Dervieux, V. Boulard, J. Périaux, and G. Vijayasundaram, "Transonic Euler Simulations by Means of Finite Element Explicit Schemes," In *AIAA 6th Computational Fluid Dynamics Conference Proceedings*, pages 267–273, July 13-15 1983.

[4] R. Aris, *Vectors, Tensors, and the Basic Equations of Fluid Mechanics*, pages 106, 151, Prentice-Hall, Englewood Cliffs, N. J., 1962.

[5] E. H. Atta and J. Vadyak, "A Grid Overlapping Scheme for Flowfield Computations about Multicomponent Configurations," *AIAA Journal*, 21(9), pp. 1271–1277, September 1983.

[6] B. S. Baldwin and T. J. Barth, "A One-Equation Turbulence Transport Model for High Reynolds number Wall-Bounded Flows," NASA TM 102874, August 1990.

[7] T. J. Barth, "Numerical Aspects of Computing Viscous High Reynolds Number Flows on Unstructured Meshes," AIAA Paper No. 91-0721, January 1991.

[8] T. J. Barth and D. C. Jespersen, "The Design and Application of Upwind Schemes on Unstructured Meshes," AIAA Paper No. 89-0366, January 1989.

[9] J. T. Batina, "Unsteady Euler Algorithm with Unstructured Dynamic Mesh for Complex-Aircraft Aeroelastic Analysis," AIAA Paper No. 89-1189, April 1989.

[10] J. T. Batina, "Implict Upwind-Euler Solution Algorithm for Unstructured-Grid Applications," In *The First European Computational Fluid Dynamics Conference*, September 7-11 1992.

[11] C. W. Boppe, "Computational Transonic Flow About Realistic Aircraft Configurations," AIAA Paper No. 78-0104, January 1978.

[12] C. W. Boppe, "Computational Transonic Flow About Realistic Aircraft Configurations," NASA CR 3243, May 1980.

[13] P. Bradshaw, D. H. Ferris, and N. P. Altwell, "Calculation of Boundary Layer Development Using the Turbulent Energy Equation," *J. Fluid Mechanics*, 28, pp. 593–616, 1967.

[14] D. A. Caughey, "A Diagonal Implicit Multigrid Algorithm for the Euler Equations,"

AIAA Paper No. 87-0599, January 1987.

[15] D. A. Caughey and E. Turkel, "Effects of Numerical Dissipation on Finite Volume Solutions of Compressible Flow Problems," AIAA Paper No. 88-0621, January 1988.

[16] T. Cebeci and A. M. O. Smith, *Analysis of Turbulent Boundary Layer*, pages 212–213, Applied Mathematics and Mechanics, Vol.15. Academic Press, New York, 1974.

[17] K. Y. Chien, "Predictions of Channel and Boundary-Layer Flows with a Low-Reynolds-Number Turbulence Model," *AIAA Journal*, 20, pp. 33–38, 1982.

[18] R. V. Chima, "Explicit Multigrid Algorithm for Quasi-Three-Dimensional Viscous Flows in Turbomachinery," *Journal of Propulsion and Power*, 3(5), pp. 397–405, Sept.-Oct. 1987.

[19] R. V. Chima and G. M. Johnson, "Efficient Solution of the Euler and Navier Stokes Equations with a Vectorized Multiple-Grid Algorithm," In *AIAA 6th Computational Fluid Dynamics Conference Proceedings*, pages 72–89, July 13-15 1983.

[20] T. J. Cook, M. A. McDonald, and M. C. P. Firmin, "Airfoil RAE 2822 Pressure Distributions and Boundary Layer and Wake Measurements," In *AGARD Advisory Report No. 138*, May 1979.

[21] J. F. Dannenhoffer, III., *Grid Adaptation for Complex Two-Dimensional Transonic Flows*, PhD thesis, Massachusetts Institute of Technology, August 1987.

[22] J. F. Dannenhoffer, III., "A Comparison of Two Adaptive Grid Techniques," In *Numerical Grid Generation in Computational Fluid Mechanics '88*. Pineridge Press Ltd, 1988.

[23] R. L. Davis, "The Prediction of Compressible, Laminar Viscous Flows Using A Time-Marching Control Volume and Multigrid Technique," In *AIAA 6th Computational Fluid Dynamics Conference Proceedings*, pages 90–109, July 13-15 1983.

[24] R. Delaney, Private communication., 1993.

[25] J. D. Denton, Ch. Hirsch, and C. Meauzé, "Analytical Test Cases for Cascade," In *AGARD Advisory Report No. 275*, 1990.

[26] D. I. Driscoll, *Calculation of Nozzle/Aftbody Flowfield Interactions*, PhD thesis, Purdue University, December 1991.

[27] A. Favre, "Équations des gaz Turbulents Compressibles I," *Journal des Mécanique*, 4(3), pp. 361–390, 1965.

[28] A. Favre, "Équations des gaz Turbulents Compressibles II," *Journal des Mécanique*, 4(4), pp. 391–421, 1965.

[29] P. A. Gnoffo, "A Finite-Volume, Adaptive Grid Algorithm Applied to Planetary Entry Flowfields," *AIAA Journal*, 21(9), pp. 1249–1254, September 1983.

[30] O. Hassan, K. Morgan, E. Peraire, E. J. Probert, and R. R. Thareja, "Adaptive Unstructured Mesh Methods for Steady Viscous Flow," In *AIAA 10th Computational Fluid Dynamics Conference Proceedings*, pages 125–133, June 24-27 1991.

[31] D. Holmes, S. H. Lamson, and S. D. Connell, "Quasi-3D Solutions for Transonic, Inviscid Flows by Adapative Triangulation," ASME Paper No. 88-GT-83, ASME Gas Turbine and Aeroengine Congress, June 1988.

[32] T. L. Holst, "Viscous Transonic Airfoil Workshop Compendium of Results," AIAA Paper No. 87-1460, June 1987.

[33] A. Jameson, "Solution of the Euler Equation for two Dimensional Transonic Flow by a Multigrid Method," MAE Report No. 1613, Princeton University, June 1983.

[34] A. Jameson, "Numerical Solution of the Euler Equations for Compressible Inviscid Equations," In F. Angrand *et al.*, editor, *Numerical Methods for the Euler Equations of Fluid Dynamics1*. SIAM, Philadelphia, 1985.

[35] A. Jameson, T. J. Baker, and N. P. Weatherill, "Calculation of Inviscid Transonic Flow over a Complete Aircraft," , AIAA Paper No. 86-0103, January 1986.

[36] A. Jameson and D. J. Mavriplis, "Finite Volume Solution of the Two-Dimensional Euler Equations on a Regular Triangular Mesh," *AIAA Jounal*, 24(4), pp. 611–618, April 1986.

[37] A. Jameson, W. Schmidt, and E. Turkel, "Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time -Stepping Schemes," AIAA Paper No. 81-1259, June 1981.

[38] A. Jameson and S. Yoon, "Multigrid Solution of the Euler Equations Using Implicit Schemes," , AIAA Paper No. 85-0293, January 1985.

[39] W. P. Jones and B. E. Launder, "The Prediction of Laminarization with a Two-Equation Model of Turbulence," *International Journal of Heat and Mass Transfer*, 15, pp. 301–314, 1972.

[40] J. Kallinderis, *Adaptation Methods for Viscous Flows*, PhD thesis, Massachusetts Institute of Technology, May 1989.

[41] T. Katsanis and W. D. McNally, "Revised FORTRAN Program for Calculating Velocities and Streamlines on the Hub-Shroud Midchannel Stream Surface of an Axial-, Radial-, or Mixed-Flow Turbomachine or Annular Duct, I – User's Manual," NASA TN-D-8430, 1977.

[42] R. Kunz, *Explicit Navier-Stokes Computation of Turbomachinery Flowfields*, PhD thesis, Pennsylvania State University, August 1991.

[43] R. Kunz and B. Lakshminarayana, "Stability of Explicit 3D Navier-Stokes/$\kappa-\epsilon$ Procedures in Complex Internal Flow Calculations," AIAA Paper No. 91-0018, January 1991.

[44] B. Lakshminarayana, "Turbulence Modeling for Complex Shear Flows," *AIAA Journal*, 24(12), pp. 1900–1918, December 1986.

[45] L. D. Landau and E. M. Lifshitz, *Fluid Mechanics*, pages 120–123, Course of Theoretical Physics, Vol.6. Pergamon Press, New York, 1959.

[46] B. E. Launder and D. B. Spalding, *Mathematical Models of Turbulence*, pages 102–103, Academic Press, New York, 1972.

[47] R. Lõhner, "The Efficient Simulation of Strongly Unsteady Flows by the Finite Element Method," AIAA Paper No. 87-0555, January 1987.

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

[49] R. Lõhner, "Adaptive H-Refinement on 3-D Unstructured Grids for Transient Problems," AIAA Paper No. 89-0365, January 1989.

[50] R. Lõhner, K. Morgan, Peraire J., and O. C. Zienkiewicz, "Adaptive Grid Refinement for the Compressible Euler Equations," In I. Babv̌ska *et al.*, editor, *Accuracy Estimates and Adaptive Refinement in Finite Element Computations*. Wiley, New York, 1986.

[51] R. Lõhner, K. Morgan, and O. C. Zienkiewicz, "The Solution of Non-linear Hyperbolic Equation System by the Finite Element Method," *Int. J. Num. Meth. Fluids*, 4, pp. 1043–1063, 1984.

[52] L. Martinelli, *Calculations of Viscous Flows with a Multigrid Method*, PhD thesis, Princeton University, October 1987.

[53] L. Martinelli, A. Jameson, and F. Grasso, "A Multigrid Method for the Navier Stokes Equations," AIAA Paper No. 86-0208, January 1986.

[54] A. Mavriplis, D. J. Jameson and L. Martinelli, "Multigrid Solution of the Navier-Stokes Equations on Triangular Meshes," AIAA Paper No. 89–120, AIAA 27th Aerospace Sciences Meeting, January 1989.

[55] D. J. Mavriplis, "Multigrid Solution of the Two-Dimensional Euler Equations on Unstructured Triangular Meshes," *AIAA Journal*, 26(7), pp. 824–831, July 1988.

[56] D. J. Mavriplis, "Algebraic Turbulence Modeling for Unstructured and Adaptive Meshes," ICASE Report 90-30, May 1990.

[57] D. J. Mavriplis and A. Jameson, "Multigrid Solution of the Euler Equations on Unstructured and Adaptive Meshes," ICASE Report 87-53, July 1987.

[58] D. J. Mavriplis and A. Jameson, "Multigrid Solution of the Two-dimensional Euler Equations on Unstructured Triangular Meshes," AIAA Paper No. 87-0353, January 1987.

[59] D. J. Mavriplis and L. Martinelli, "Multigrid Solution of Compressible Turbulent Flow on Unstruc tured Meshes Using a Two-Equation Model.," AIAA Paper No. 91–0237, January 1991.

[60] D. P. Miller.

[61] L. M. Milne-Thomson, *Theoretical Aerodynamics*, page 415, Dover Publications, New York, 1973.

[62] R. A. Mitcheltree, M. D. Salas, and H. A. Hassan, "One-Equation Turbulence Model for Transonic Airfoil Flows," *AIAA Journal*, 28(9), pp. 1625–1632, September 1990.

[63] K. Morgan and J. Peraire, "Finite Element Methods for Compressible Flows," Lecture Series 1987-04, Von Karman Institute for Fluid Dynamics, March 1987.

[64] K. Morgan, J. Peraire, R. R. Thareja, and J. R. Stewart, "An Adaptive Finite Element Scheme for the Euler and Navier Stokes Equations," In *AIAA 8th Computational Fluid Dynamics Conference Proceedings*, pages 749–757, June 9-11 1987.

[65] R. H. Ni, "A Multiple Grid Scheme for Solving Euler Equations," *AIAA Journal*, 20(11), pp. 1565–1571, November 1982.

[66] P. Parikh, R. Löhner, C. Gumber, and S. Pirzadeh, "Numerical Solutions on a PATHFINDER and Other Configurations Using Unstructured Grids and a Finite Element Solver," AIAA Paper No. 89-0362, January 1989.

[67] J. Peraire, M. Vahdati, K. Morgan, and O. C. Zienkiewicz, "Adaptive Remeshing for Compressible Flow Computation," *Journal of Computational Physics*, 72, pp. 449–466, 1987.

[68] M. M. Rai, "Navier Stokes Simulations of Rotor-Stator Interaction using Patched and overlaid Grids," In *AIAA 7th Computational Fluid Dynamics Conference Proceedings*, pages 282–298, July 15-17 1985.

[69] W. Rodi, "Turbulence Models for Practical Applications," Lecture Series 1987-06, Von Karman Institute for Fluid Dynamics, May 1987.

[70] P. Roe, "Error Estimates for Cell-Vertex Solutions of the Compressible Euler Equations.," ICASE Report 87-6, January 1987.

[71] P. Rostand, "Algebraic Turbulence Models for the Computation of Two-Dimen sional High Speed Flows Using Unstructured Grids," ICASE Report 88-63, November 1988.

[72] P. E. Rubber and K. D. Lee, "Patched Coordinate Systems," In J. F. Thompson, editor, *Numerical Grid Generation*. North-Holland, New York, 1982.

[73] M. W. Rubesin, "A One-Equation Model of Turbulence for Use with the Compressible Navier-Stokes Equations," NASA TM X-73128, April 1976.

[74] M. W. Rubesin and W. C. Rose, "The Turbulent Mean-Flow, Reynolds-Stress, and Heat-Flux Equations in Mass-Averaged Dependent Variables," NASA TM X-62248, March 1973.

[75] J. M. Sanz, "Lewis Inverse Design Code (LINDES) - Users Manual," NASA TP 2676, 1987.

[76] H. Schlichting, *Boundary-Layer Theory*, pages 312–313, 598–600, 638–639, 6th Edition. McGraw-Hill, New York, 1968.

[77] C. H. Sieverding, W. Van Hove, E. Boletis, H. Gotthardt, and U. Stark, "Description of Test Cases and Presentation of Experimental Results," Lecture Series 1982-05 Vol.3, Von Karman Institute for Fluid Dynamics, April 1982.

[78] B. Stoufflet, J. Periaux, F. Fezoui, and A. Dervieux, "Numerical Simulation of 3-D Hypersonic Euler Flows Around Space Vehicles Using Adaptive Finite Element," AIAA Paper No. 87-0560, January 1987.

[79] A. J. Strazisar, J. R. Wood, M. D. Hathaway, and K. L. Suder, "Laser Anemometer Measurements in a Transonic Axial-Flow Fan Rotor," NASA TP-2879, November 1989.

[80] A. Suddhoo and I. M. Hall, "Test Cases for the Plane Potential Flow Past Multi-Element Aerofoils," *Aeronautical Journal*, pages 403–414, December 1985.

[81] H. Tennekes and J. L. Lumley, *A First Course in Turbulence*, pages 42–44, The MIT Press, Cambridge, 1972.

[82] J. F. Thompson, "Review on the Start of Art of Adaptive Grids," AIAA Paper No. 84-1606, June 1984.

[83] W. J. Usab, Jr., *Embedded Mesh Solutions of the Euler Equations Using a Multiple-Grid Method*, PhD thesis, Massachusetts Institute of Technology, December 1983.

[84] B. van den Berg, "Boundary Layer Measurements on a Two-Dimensional Wing with Flap," , NLR TR 79009 U, National Aerospace Laboratory NLR, the Netherlands, 1979.

[85] H. Vandromme and H. Haminh, "Turbulence Modeling for Compressible Flows," Lecture Series 1987-06, Von Karman Institute for Fluid Dynamics, May 1987.

[86] Z. Wang, "A Method for Aerodynamic Design of Blades in Quasi-Three-Dimensional Calculation of Turbomachines," *Journal of Turbomachinery*, 110, pp. 181–186, April 1988.

[87] N. P. Weatherill, "A Method for Generating Irregular Computational Grids in Multiply Connected Planar Domains," *Int. J. Num. Meth. Fluids*, 8, pp. 181–197, 1988.

[88] N. P. Weathrill, "The Generation of Unstructured Grids Using Dirichlet Tessellations," MAE Report No. 1715, Princeton University, July 1985.

[89] K. Weighardt, "Flat-Plate Flow," In D. Coles and E. A. Hirst, editors, *Proceedings, Computation of Turbulent Boundary Layer–1968 A FOSR-IFP-Stanford Conference Vol. 2*, pages 98–123. Stanford University, 1969.

[90] D. L. Whitaker, B. Grossman, and R. Löhner, "Two Dimensional Euler Computations on a Triangular Mesh Using an Upwind, Finite-Volume Scheme," AIAA Paper No. 89-0470, January 1989.

[91] C. H. Wu, "A General Theory of Three-Dimesional Flow in Subsonic and Supersonic Turbomachines of Axial-, Radial-, and Mixed-Flow Types," NACA TN-21604, 1952.

[92] S. Yoon, Y. K. Choo, and A. Jameson, "An LU-SSOR Scheme for the Euler and Navier Stokes Equations," AIAA Paper No. 87-0453, January 1987.

APPENDICES

Appendix A: Derivation of the Quasi-Three-Dimensional Navier-Stokes Equations

The vector form of the three-dimensional Navier-Stokes equations may be expressed as follows.

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \vec{V} = 0 \tag{A.1}$$

$$\frac{D\vec{V}}{Dt} + \frac{\nabla p}{\rho} = \frac{1}{\rho} \nabla \cdot \bar{\bar{\sigma}} \tag{A.2}$$

and

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot \rho \vec{V} H = \nabla \cdot (\kappa_T \nabla T + \bar{\bar{\sigma}} \cdot \vec{V}) \tag{A.3}$$

The above equations represent the physical conservation law of mass, momentum and energy for a fluid flow without body force and external heat transfer. For analysis of general flow problem it is useful to consider flow on the generalized curvilinear coordinate system because they can be transformed to any coordinate system.

Let (x_1, x_2, x_3) be a set of generalized curvilinear coordinates and $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ be the set of corresponding unit vectors. The expressions for the vector operators appearing in the Navier-Stokes equations are described below.

The gradient of a scalar ϕ is described by

$$\nabla \phi = \frac{1}{h_1}\frac{\partial \phi}{\partial x_1}\mathbf{e}_1 + \frac{1}{h_2}\frac{\partial \phi}{\partial x_2}\mathbf{e}_2 + \frac{1}{h_3}\frac{\partial \phi}{\partial x_3}\mathbf{e}_3 \tag{A.4}$$

The gradient of a vector $\vec{A}$ is given by

$$\nabla \vec{A} = [A_{ij}] \tag{A.5}$$

where A_{ij} is the ij^{th} element of a tensor matrix $\bar{\bar{A}}$ and is described as

$$A_{ij} = \frac{1}{h_j}\frac{\partial A_i}{\partial x_j} + \delta_{ij}\sum_{k=1} \frac{A_k}{h_i h_k}\frac{\partial h_i}{\partial x_k} - \frac{A_j}{h_i h_j}\frac{\partial h_j}{\partial x_i} \tag{A.6}$$

where δ_{ij} is the Kronecker delta function.

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

Applying Eq.(A.5), the directional derivative of a vector $\vec{B}$ can be obtained as

$$\begin{aligned}(\vec{A}\cdot\nabla)\vec{B} \;=\;& \frac{\mathbf{e_1}}{h_1}\left[(A_1\frac{\partial B_1}{\partial x_1}+A_2\frac{\partial B_2}{\partial x_1}+A_3\frac{\partial B_3}{\partial x_1})-\frac{A_2}{h_2}(\frac{\partial h_2B_2}{\partial x_1}-\frac{\partial h_1B_1}{\partial x_2})\right.\\ &\left.+\frac{A_3}{h_3}(\frac{\partial h_1B_1}{\partial x_3}-\frac{\partial h_3B_3}{\partial x_1})\right]+\frac{\mathbf{e_2}}{h_2}\left[(A_1\frac{\partial B_1}{\partial x_2}+A_2\frac{\partial B_2}{\partial x_2}+A_3\frac{\partial B_3}{\partial x_2})\right.\\ &\left.-\frac{A_3}{h_3}(\frac{\partial h_3B_3}{\partial x_2}-\frac{\partial h_2B_2}{\partial x_3})+\frac{A_1}{h_1}(\frac{\partial h_2B_2}{\partial x_1}-\frac{\partial h_1B_1}{\partial x_2})\right]\\ &+\frac{\mathbf{e_3}}{h_3}\left[(A_1\frac{\partial B_1}{\partial x_3}+A_2\frac{\partial B_2}{\partial x_3}+A_3\frac{\partial B_3}{\partial x_3})-\frac{A_1}{h_1}(\frac{\partial h_1B_1}{\partial x_3}-\frac{\partial h_3B_3}{\partial x_1})\right.\\ &\left.+\frac{A_2}{h_2}(\frac{\partial h_3B_3}{\partial x_2}-\frac{\partial h_2B_2}{\partial x_3})\right]\end{aligned} \qquad \text{(A.7)}$$

The divergence of a vector $\vec{A}$ is given by

$$\nabla\cdot\vec{A}=\frac{1}{h_1h_2h_3}\left[\frac{\partial}{\partial x_1}(h_2h_3A_1)+\frac{\partial}{\partial x_2}(h_3h_1A_2)+\frac{\partial}{\partial x_3}(h_1h_2A_3)\right] \qquad \text{(A.8)}$$

The divergence of the stress tensor $\bar{\bar{\sigma}}$ is expressed as

$$\begin{aligned}\nabla\cdot\bar{\bar{\sigma}} \;=\;& \mathbf{e}_1\left\{\frac{1}{h_1h_2h_3}\left[\frac{\partial}{\partial x_1}(h_2h_3\sigma_{11})+\frac{\partial}{\partial x_2}(h_1h_3\sigma_{12})+\frac{\partial}{\partial x_3}(h_1h_2\sigma_{13})\right]\right.\\ &\left.+\sigma_{12}\frac{1}{h_1h_2}\frac{\partial h_1}{\partial x_2}+\sigma_{13}\frac{1}{h_1h_3}\frac{h_1}{x_3}-\sigma_{22}\frac{1}{h_1h_2}\frac{\partial h_2}{\partial x_1}-\sigma_{33}\frac{1}{h_1h_3}\frac{\partial h_3}{\partial x_1}\right\}\\ &+\mathbf{e}_2\left\{\frac{1}{h_1h_2h_3}\left[\frac{\partial}{\partial x_1}(h_2h_3\sigma_{12})+\frac{\partial}{\partial x_2}(h_1h_3\sigma_{22})+\frac{\partial}{\partial x_3}(h_1h_2\sigma_{23})\right]\right.\\ &\left.+\sigma_{23}\frac{1}{h_2h_3}\frac{\partial h_2}{\partial x_3}+\sigma_{12}\frac{1}{h_1h_2}\frac{\partial h_2}{\partial x_1}-\sigma_{33}\frac{1}{h_2h_3}\frac{\partial h_3}{\partial x_2}-\sigma_{11}\frac{1}{h_1h_2}\frac{\partial h_1}{\partial x_2}\right\}\\ &+\mathbf{e}_3\left\{\frac{1}{h_1h_2h_3}\left[\frac{\partial}{\partial x_1}(h_2h_3\sigma_{13})+\frac{\partial}{\partial x_2}(h_1h_3\sigma_{23})+\frac{\partial}{\partial x_3}(h_1h_2\sigma_{33})\right]\right.\\ &\left.+\sigma_{13}\frac{1}{h_1h_3}\frac{\partial h_3}{\partial x_1}+\sigma_{23}\frac{1}{h_2h_3}\frac{\partial h_3}{\partial x_2}-\sigma_{11}\frac{1}{h_1h_3}\frac{\partial h_1}{\partial x_3}-\sigma_{22}\frac{1}{h_2h_3}\frac{\partial h_2}{\partial x_3}\right\}\end{aligned} \qquad \text{(A.9)}$$

The stress tensor can be described as

$$\bar{\bar{\sigma}}=\mu\left(\bar{\bar{S}}+\bar{\bar{S}}^T\right)+\delta_{ij}\lambda\,\nabla\cdot\vec{V} \qquad \text{(A.10)}$$

where $\bar{\bar{S}}$ denotes the velocity gradient tensor matrix.

$$\bar{\bar{S}}=\nabla\vec{V}$$

The metric coefficients h_1, h_2, and h_3 are defined as

$$\begin{aligned} h_1^2 &= \left(\frac{\partial x}{\partial x_1}\right)^2 + \left(\frac{\partial y}{\partial x_1}\right)^2 + \left(\frac{\partial z}{\partial x_1}\right)^2 \\ h_2^2 &= \left(\frac{\partial x}{\partial x_2}\right)^2 + \left(\frac{\partial y}{\partial x_2}\right)^2 + \left(\frac{\partial z}{\partial x_2}\right)^2 \\ h_3^2 &= \left(\frac{\partial x}{\partial x_3}\right)^2 + \left(\frac{\partial y}{\partial x_3}\right)^2 + \left(\frac{\partial z}{\partial x_3}\right)^2 \end{aligned} \tag{A.11}$$

In order to obtain the Navier-Stokes equations for the quasi-3D blade to blade flow, we assume the coordinate system (m, θ) is on the streamsurface and the third component n is perpendicular to the plane (m, θ). With this third coordinate, we can define a three-dimensional curvilinear coordinate system using (m, θ, n). In this coordinate system we have $\vec{V} \cdot \mathbf{e}_n = 0$ or $V_n = 0$, so the velocity may be described as

$$\vec{V} = (V_m, V_\theta, 0) \tag{A.12}$$

If we let the streamtube height h be defined in the direction of n and measured in the units of the variable n, then the metric coefficients may be obtained as

$$h_1 = |\frac{\partial \vec{r}}{\partial m}| = 1, \qquad h_2 = |\frac{\partial \vec{r}}{\partial \theta}| = r, \qquad h_3 = |\frac{\partial \vec{r}}{\partial n}| = h \tag{A.13}$$

Substituting the vector operators, metric coefficients, and velocity into the vector form of the Navier-Stokes equations gives the quasi-3D Navier-Stokes equations as follows.

Continuity equation:

$$\begin{aligned} & \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{V}) = 0 \\ \rightarrow \quad & \frac{\partial \rho}{\partial t} + \frac{1}{rh}\left[\frac{\partial}{\partial m}(rh\rho V_m) + \frac{\partial}{\partial \theta}(h\rho V_\theta)\right] = 0 \\ \rightarrow \quad & \frac{\partial}{\partial t}(rh\rho) + \frac{\partial}{\partial m}(rh\rho V_m) + \frac{\partial}{\partial \theta}(h\rho V_\theta) = 0 \end{aligned} \tag{A.14}$$

m- momentum equation:

$$
\begin{aligned}
& \frac{\partial V_m}{\partial t} + (\vec{V}\cdot\nabla)V_m + \frac{1}{\rho}\frac{\partial p}{\partial m} = \frac{1}{\rho}\nabla\cdot\bar{\bar{\sigma}}\cdot\mathbf{e}_m \\
\rightarrow\quad & \frac{\partial V_m}{\partial t} + \left[V_m\frac{\partial V_m}{\partial m} + V_\theta\frac{\partial V_\theta}{\partial m} - \frac{V_\theta}{r}(\frac{\partial rV_\theta}{\partial m} - \frac{\partial V_m}{\partial \theta})\right] + \frac{1}{\rho}\frac{\partial p}{\partial m} \\
& \quad = \frac{1}{\rho}\left\{\frac{1}{rh}\left[\frac{\partial}{\partial m}(rh\sigma_{11}) + \frac{\partial}{\partial\theta}(h\sigma_{12})\right] - \sigma_{22}\frac{1}{r}\frac{dr}{dm} - \sigma_{33}\frac{1}{h}\frac{dh}{dm}\right\} \\
\rightarrow\quad & \frac{\partial V_m}{\partial t} + V_m\frac{\partial V_m}{\partial m} + \frac{V_\theta}{r}\frac{\partial V_m}{\partial\theta} + \frac{1}{\rho}\frac{\partial p}{\partial m} = \frac{V_\theta^2}{r}\frac{dr}{dm} \\
& \quad + \frac{1}{\rho}\left\{\frac{1}{rh}\left[\frac{\partial}{\partial m}(rh\sigma_{11}) + \frac{\partial}{\partial\theta}(h\sigma_{12})\right] - \sigma_{22}\frac{1}{r}\frac{dr}{dm} - \sigma_{33}\frac{1}{h}\frac{dh}{dm}\right\}
\end{aligned}
\tag{A.15}
$$

θ- momentum equation:

$$
\begin{aligned}
& \frac{\partial V_\theta}{\partial t} + (\vec{V}\cdot\nabla)V_\theta + \frac{1}{r\rho}\frac{\partial p}{\partial\theta} = \frac{1}{\rho}\nabla\cdot\bar{\bar{\sigma}}\cdot\mathbf{e}_\theta \\
\rightarrow\quad & \frac{\partial V_\theta}{\partial t} + \frac{1}{r}\left[V_m\frac{\partial V_m}{\partial\theta} + V_\theta\frac{\partial V_\theta}{\partial\theta} + V_m(\frac{\partial rV_\theta}{\partial m} - \frac{\partial V_m}{\partial\theta})\right] + \frac{1}{r\rho}\frac{\partial p}{\partial\theta} \\
& \quad = \frac{1}{\rho}\left\{\frac{1}{rh}\left[\frac{\partial}{\partial m}(rh\sigma_{12}) + \frac{\partial}{\partial\theta}(h\sigma_{22})\right] + \sigma_{12}\frac{1}{r}\frac{dr}{dm}\right\} \\
\rightarrow\quad & \frac{\partial V_\theta}{\partial t} + V_m\frac{\partial V_\theta}{\partial m} + \frac{V_\theta}{r}\frac{\partial V_\theta}{\partial\theta} + \frac{1}{r\rho}\frac{\partial p}{\partial\theta} = -\frac{V_mV_\theta}{r}\frac{dr}{dm} \\
& \quad + \frac{1}{\rho}\left\{\frac{1}{rh}\left[\frac{\partial}{\partial m}(rh\sigma_{12}) + \frac{\partial}{\partial\theta}(h\sigma_{22})\right] + \sigma_{12}\frac{1}{r}\frac{dr}{dm}\right\}
\end{aligned}
\tag{A.16}
$$

energy equation:

$$
\begin{aligned}
& \frac{\partial\rho E}{\partial t} + \nabla\cdot(\rho\vec{V}H) = \nabla\cdot(\kappa_\tau\nabla T + \bar{\bar{\sigma}}\cdot\vec{V}) \\
\rightarrow\quad & \frac{\partial\rho E}{\partial t} + \frac{1}{rh}\left[\frac{\partial}{\partial m}(rh\rho V_mH) + \frac{\partial}{\partial\theta}(h\rho V_\theta H)\right] \\
& = \frac{1}{rh}\left\{\frac{\partial}{\partial m}\left[rh\left(\kappa_\tau\frac{\partial T}{\partial m} + V_m\sigma_{11} + V_\theta\sigma_{12}\right)\right] + \frac{\partial}{\partial\theta}\left[h\left(\kappa_\tau\frac{\partial T}{\partial\theta} + V_m\sigma_{12} + V_\theta\sigma_{22}\right)\right]\right\} \\
\rightarrow\quad & \frac{\partial}{\partial t}(rh\rho E) + \frac{\partial}{\partial m}(rh\rho V_mH) + \frac{\partial}{\partial\theta}(h\rho V_\theta H) \\
& = \frac{\partial}{\partial m}\left[rh\left(\kappa_\tau\frac{\partial T}{\partial m} + V_m\sigma_{11} + V_\theta\sigma_{12}\right)\right] + \frac{\partial}{\partial\theta}\left[h\left(\kappa_\tau\frac{\partial T}{\partial\theta} + V_m\sigma_{12} + V_\theta\sigma_{22}\right)\right]
\end{aligned}
\tag{A.17}
$$

Equations (A.15) and (A.16) can be cast in the conservation form using the continuity equation(A.14).

Performing a linear combination of Eq.(A.15)$\cdot rh\rho$ + Eq.(A.14)$\cdot V_m$, the m-momentum equation is obtained as

$$\begin{aligned}&\frac{\partial}{\partial t}(rh\rho V_m) + \frac{\partial}{\partial m}[rh(\rho V_m^2 + p)] + \frac{\partial}{\partial \theta}(h\rho V_\theta V_m)\\ &= \frac{\partial}{\partial m}(rh\sigma_{11}) + \frac{\partial}{\partial \theta}(h\sigma_{12}) + rh\left[(\rho V_\theta^2 + p - \sigma_{22})\frac{1}{r}\frac{dr}{dm} + (p - \sigma_{33})\frac{1}{h}\frac{dh}{dm}\right]\end{aligned} \tag{A.18}$$

Similarly, a linear combination of $r\cdot$ (Eq.(A.16)$\cdot rh\rho$ + Eq.(A.14)$\cdot V_\theta$) results in the conservation form of the θ- momentum equation.

$$\begin{aligned}&r\left\{\frac{\partial}{\partial t}(rh\rho V_\theta) + \frac{\partial}{\partial m}(rh\rho V_\theta V_m) + \frac{\partial}{\partial \theta}[h(\rho V_\theta^2 + p)] + h\rho V_m V_\theta \frac{dr}{dm}\right\}\\ &= r\frac{\partial}{\partial m}(rh\sigma_{12}) + (rh\sigma_{12})\frac{dr}{dm} + r\frac{\partial}{\partial \theta}(h\sigma_{22})\\ \rightarrow\quad &\frac{\partial}{\partial t}(rh\rho V_\theta r) + \frac{\partial}{\partial m}(rh\rho V_\theta V_m r) + \frac{\partial}{\partial \theta}\left[h(\rho V_\theta^2 + p)r\right]\\ &= \frac{\partial}{\partial m}(rh\sigma_{12} r) + \frac{\partial}{\partial \theta}(h\sigma_{22} r)\end{aligned} \tag{A.19}$$

Therefore, the quasi-3D Navier-Stokes equation may be expressed as

$$\frac{\partial(rh\vec{U})}{\partial t} + \frac{\partial(rh\vec{F})}{\partial m} + \frac{\partial(h\vec{G})}{\partial \theta} = \frac{\partial(rh\vec{R})}{\partial m} + \frac{\partial(h\vec{S})}{\partial \theta} + rh\vec{K} \tag{A.20}$$

where

$$\vec{U} = \begin{pmatrix} \rho \\ \rho V_m \\ \rho V_\theta r \\ \rho E \end{pmatrix}, \quad \vec{F} = \begin{pmatrix} \rho V_m \\ \rho V_m^2 + p \\ \rho V_m V_\theta r \\ V_m(\rho E + p) \end{pmatrix}, \quad \vec{G} = \begin{pmatrix} \rho V_\theta \\ \rho V_m V_\theta \\ (\rho V_\theta^2 + p)r \\ V_\theta(\rho E + p) \end{pmatrix}$$

$$\vec{R} = \begin{pmatrix} 0 \\ \sigma_{11} \\ \sigma_{12} r \\ R_4 \end{pmatrix}, \quad \vec{S} = \begin{pmatrix} 0 \\ \sigma_{12} \\ \sigma_{22} r \\ S_4 \end{pmatrix}, \quad \vec{K} = \begin{pmatrix} 0 \\ K_2 \\ 0 \\ 0 \end{pmatrix}$$

where

$$K_2 = (\rho V_\theta^2 + p - \sigma_{22})(\frac{1}{r}\frac{dr}{dm}) + (p - \sigma_{33})(\frac{1}{h}\frac{dh}{dm})$$

$$\begin{aligned} R_4 &= (\kappa_T \partial_m T + V_m \sigma_{11} + V_\theta \sigma_{12}) \\ S_4 &= (\frac{\kappa_T}{r}\partial_\theta T + V_m \sigma_{12} + V_\theta \sigma_{22}) \end{aligned} \qquad \text{(A.21)}$$

Employing Eq.(A.10) the stress terms are

$$\begin{aligned} \sigma_{11} &= 2\mu \partial_m V_m + \lambda \nabla \cdot \vec{V} \\ \sigma_{22} &= 2\mu(\partial_\theta V_\theta + V_m \frac{dr}{dm})/r + \lambda \nabla \cdot \vec{V} \\ \sigma_{33} &= 2\mu V_m (\frac{1}{h}\frac{dh}{dm}) + \lambda \nabla \cdot \vec{V} \\ \sigma_{12} &= \mu(\partial_m V_\theta - V_\theta(\frac{1}{r}\frac{dr}{dm}) + \frac{1}{r}\partial_\theta V_m) \end{aligned} \qquad \text{(A.22)}$$

We may also write the equations in the relative frame which the coordinate system rotates with blade rows. Let Ω be the rotating speed of blade rows, $\bar{\theta}$ be the relative angle with respect to the blade rows, and W_θ denote the relative tangential velocity, then we have

$$\bar{\theta} = \theta - \Omega t, \quad W_\theta = V_\theta - r\Omega$$

and

$$\frac{\partial}{\partial t} = \frac{\partial}{\partial t} + \frac{\partial}{\partial \bar{\theta}}\frac{\partial \bar{\theta}}{\partial t} = \frac{\partial}{\partial t} - \Omega \frac{\partial}{\partial \bar{\theta}} \qquad \text{(A.23)}$$

Substituting the above relation into equation (A.20), we have

$$\frac{\partial(rh\vec{U})}{\partial t} + \frac{\partial(rh\vec{F})}{\partial m} + \frac{\partial(h\bar{\vec{G}})}{\partial \bar{\theta}} = \frac{\partial(rh\vec{R})}{\partial m} + \frac{\partial(h\vec{S})}{\partial \bar{\theta}} + rh\vec{K} \qquad \text{(A.24)}$$

$$\bar{\vec{G}} = \vec{G} - \Omega\vec{U} = \begin{pmatrix} \rho W_\theta \\ \rho V_m W_\theta \\ (\rho V_\theta W_\theta + p)r \\ W_\theta(\rho E + p) + r\Omega p \end{pmatrix}$$

Appendix B: Quasi-Three-Dimensional Favre-Averaged Navier-Stokes Equations

In compressible flows, the density-weighted average suggested by Favre [27, 28] is helpful to simplify the formulation of the mean flow equations. Favre averaging is a hybrid average method which uses density-weighted averaging on all fluid properties except for pressure and density on which an ensemble (or time) averaging is used. An ensemble-averaged quantity, represented by angular brackets <>, is defined as

$$\langle f \rangle = \frac{1}{N} \sum_{i=1}^{N} f_i \tag{B.1}$$

and a density-weighted averaged quantity, represented by a tilde, is defined as

$$\tilde{f} = \frac{\langle \rho f \rangle}{\langle \rho \rangle} \tag{B.2}$$

where f is some indenpendent variable. Follow the Reynolds averaging procedure, the quantity is decomposed into the averaged and fluctuating parts.

$$f = \langle f \rangle + f', \quad \text{and} \quad \langle f' \rangle = 0 \tag{B.3}$$

and

$$f = \tilde{f} + f'', \quad \text{and} \quad \langle \rho f'' \rangle = 0 \quad \text{but} \quad \widetilde{f''} \neq 0 \tag{B.4}$$

Employing Favre averaging the vector form of the mean flow equations in the generalized curvilinear coordinate system are [74]

$$\frac{\partial \langle \rho \rangle}{\partial t} + \nabla \cdot \left(\langle \rho \rangle \tilde{\vec{V}} \right) = 0 \tag{B.5}$$

$$\frac{D\tilde{\vec{V}}}{Dt} + \frac{\nabla \langle p \rangle}{\langle \rho \rangle} = \frac{1}{\langle \rho \rangle} \nabla \cdot \left(\tilde{\bar{\bar{\sigma}}} - \langle \vec{V}'' \vec{V}'' \rangle \right) \tag{B.6}$$

and

$$\frac{\partial \langle \rho \rangle \tilde{E}}{\partial t} + \nabla \cdot \left(\tilde{\vec{V}} \langle \rho \rangle \widetilde{H} \right) = \nabla \cdot \left(\kappa_\tau \nabla \tilde{T} + \langle \rho \vec{V}'' H'' \rangle + \tilde{\vec{V}} \cdot \left(\tilde{\bar{\bar{\sigma}}} - \langle \rho \vec{V}'' \vec{V}'' \rangle \right) \right) \tag{B.7}$$

The mean flow equations are in a similar form of the full Navier-Stokes equations Eqs.(A.1)– (A.3) except for the appearance of the unknown terms in the mean momentum and energy equations. Applying the same procedures given in Appendix A, the Favre-averaged N-S equations in the generalized curvilinear coordinate system can be converted to the desired quasi-3D relative coordinate system.

In the particular (m, θ, n) coordinate system, the mean-flow velocity $\vec{V}$ and the fluctuating velocity $\vec{V''}$ are given by

$$\begin{aligned} \tilde{\vec{V}} &= \tilde{V}_m \mathbf{e}_m + \tilde{V}_\theta \mathbf{e}_\theta \\ \vec{V''} &= \vec{V} - \tilde{\vec{V}} = V_m{''}\mathbf{e}_m + V_\theta{''}\mathbf{e}_\theta + V_n{''}\mathbf{e}_n \end{aligned} \tag{B.8}$$

In the above expressions, θ now represents the relative angle with respect to the blade rows. Using the coordinate transformation procedure given in the Appendix A, the mean-flow equations Eqs.(B.5)–(B.7) are converted to

Mean-flow continuity equation:

$$\frac{\partial}{\partial t}(rh\langle\rho\rangle) + \frac{\partial}{\partial m}\left(rh\langle\rho\rangle\tilde{V}_m\right) + \frac{\partial}{\partial\theta}\left(h\langle\rho\rangle\widetilde{W}_\theta\right) = 0 \tag{B.9}$$

Mean flow m- momentum equation:

$$\begin{aligned} &\frac{\partial}{\partial t}\left(rh\langle\rho\rangle\tilde{V}_m\right) + \frac{\partial}{\partial m}\left[rh\left(\langle\rho\rangle\tilde{V}_m^2 + \langle p\rangle\right)\right] + \frac{\partial}{\partial\theta}\left(h\langle\rho\rangle\tilde{V}_m\tilde{V}_\theta\right) \\ &= \frac{\partial}{\partial m}\left(rh\left(\tilde{\sigma}_{11} - \langle\rho V_m''^2\rangle\right)\right) + \frac{\partial}{\partial\theta}\left(h\left(\tilde{\sigma}_{12} - \langle\rho V_m''V_\theta''\rangle\right)\right) \\ &\quad + rh\left[\left(\langle\rho\rangle\tilde{V}_\theta^2 + \langle p\rangle - \tilde{\sigma}_{22} + \langle\rho V_\theta''^2\rangle\right)\frac{1}{r}\frac{dr}{dm}\right. \\ &\quad \left. + \left(\langle p\rangle - \tilde{\sigma}_{33} + \langle\rho V_n''^2\rangle\right)\frac{1}{h}\frac{dh}{dm}\right] \end{aligned} \tag{B.10}$$

Mean-flow θ- momentum equation:

$$\begin{aligned} &\frac{\partial}{\partial t}\left(rh\langle\rho\rangle\tilde{V}_\theta r\right) + \frac{\partial}{\partial m}\left(rh\langle\rho\rangle\tilde{V}_m\tilde{V}_\theta r\right) + \frac{\partial}{\partial\theta}\left[h\left(\langle\rho\rangle\tilde{V}_\theta\widetilde{W}_\theta + \langle p\rangle\right)r\right] \\ &= \frac{\partial}{\partial m}\left(rh\left(\tilde{\sigma}_{12} - \langle\rho V_m''V_\theta''\rangle\right)r\right) + \frac{\partial}{\partial\theta}\left(h\left(\tilde{\sigma}_{22} - \langle\rho V_\theta''^2\rangle\right)r\right) \end{aligned} \tag{B.11}$$

Mean-flow energy equation:

$$\begin{aligned} &\frac{\partial}{\partial t}\left(rh\langle\rho\rangle\tilde{E}\right) + \frac{\partial}{\partial m}\left[rh\tilde{V}_m\left(\langle\rho\rangle\tilde{E} + \langle p\rangle\right)\right] + \frac{\partial}{\partial\theta}\left[h\left(\widetilde{W}_\theta\left(\langle\rho\rangle\tilde{E} + \langle p\rangle\right) + r\Omega\langle p\rangle\right)\right] \\ &= \frac{\partial}{\partial m}\left[rh\left(\kappa_\tau\frac{\partial\tilde{T}}{\partial m} + \langle\rho V_m''H''\rangle + \tilde{V}_m\left(\tilde{\sigma}_{11} - \langle\rho V_m''^2\rangle\right) + \tilde{V}_\theta\left(\tilde{\sigma}_{12} - \langle\rho V_m''V_\theta''\rangle\right)\right)\right] \\ &\quad + \frac{\partial}{\partial\theta}\left[h\left(\frac{\kappa_\tau}{r}\frac{\partial\tilde{T}}{\partial\theta} + \langle\rho V_\theta''H''\rangle + \tilde{V}_m\left(\tilde{\sigma}_{12} - \langle\rho V_m''V_\theta''\rangle\right) + \tilde{V}_\theta\left(\tilde{\sigma}_{22} - \langle\rho V_\theta''^2\rangle\right)\right)\right] \end{aligned} \tag{B.12}$$

These mean-flow equations may be rewritten in a condensed vector form as

$$\frac{\partial(rh\langle\vec{U}\rangle)}{\partial t}+\frac{\partial(rh\langle\vec{F}\rangle)}{\partial m}+\frac{\partial(h\langle\vec{G}\rangle)}{\partial\theta}-\left(\frac{\partial(rh\langle\vec{R}\rangle)}{\partial m}+\frac{\partial(h\langle\vec{S}\rangle)}{\partial\theta}\right)=rh\langle\vec{K}\rangle \qquad \text{(B.13)}$$

where

$$\langle\vec{U}\rangle=\begin{pmatrix}\langle\rho\rangle\\ \langle\rho\rangle\widetilde{V}_m\\ \langle\rho\rangle\widetilde{W}_\theta\\ \langle\rho\rangle\widetilde{E}\end{pmatrix},\quad \langle\vec{F}\rangle=\begin{pmatrix}\langle\rho\rangle\widetilde{V}_m\\ \langle\rho\rangle\widetilde{V^2}_m+\langle p\rangle\\ \langle\rho\rangle\widetilde{V}_m\widetilde{V}_\theta r\\ \widetilde{V_m}(\langle\rho\rangle\widetilde{E}+\langle p\rangle)\end{pmatrix}$$

$$\langle\vec{G}\rangle=\begin{pmatrix}\langle\rho\rangle\widetilde{W}_\theta\\ \langle\rho\rangle\widetilde{V}_m\widetilde{W}_\theta\\ (\langle\rho\rangle\widetilde{V}_\theta\widetilde{W}_\theta+\langle p\rangle)r\\ \widetilde{W}_\theta(\langle\rho\rangle\widetilde{E}+\langle p\rangle)+r\Omega\langle p\rangle\end{pmatrix}$$

$$\langle\vec{R}\rangle=\begin{pmatrix}0\\ \widetilde{\sigma}_{11}-\langle\rho V_m''^2\rangle\\ (\widetilde{\sigma}_{12}-\langle\rho V_m''V_\theta''\rangle)r\\ \widetilde{V}_m(\widetilde{\sigma}_{11}-\langle\rho V_m''^2\rangle)+\widetilde{V}_\theta(\widetilde{\sigma}_{12}-\langle\rho V_m''V_\theta''\rangle)+\kappa_\tau\frac{\partial\widetilde{T}}{\partial m}-\langle\rho V_m''H''\rangle\end{pmatrix}$$

$$\langle\vec{S}\rangle=\begin{pmatrix}0\\ \widetilde{\sigma}_{12}-\langle\rho V_m''V_\theta''\rangle\\ (\widetilde{\sigma}_{22}-\langle\rho V_\theta''^2\rangle)r\\ \widetilde{V}_m(\widetilde{\sigma}_{12}-\langle\rho V_m''V_\theta''\rangle)+\widetilde{V}_\theta(\widetilde{\sigma}_{22}-\langle\rho V_\theta''^2\rangle)+\frac{\kappa_T}{r}\frac{\partial\widetilde{T}}{\partial\theta}-\langle\rho V_\theta''H''\rangle\end{pmatrix}$$

$$\langle\vec{K}\rangle=\begin{pmatrix}0\\ \left[\langle\rho\rangle\widetilde{V^2}_\theta+\langle p\rangle-(\widetilde{\sigma}_{22}-\langle\rho V_\theta''^2\rangle)\right]\left(\frac{1}{r}\frac{dr}{dm}\right)+(\langle p\rangle-(\widetilde{\sigma}_{33}-\langle\rho V_n''^2\rangle))\left(\frac{1}{h}\frac{dh}{dm}\right)\\ 0\\ 0\end{pmatrix}$$

VITA

VITA

Yi-Tsann Jiang was born on [redacted]. He entered the Department of Aeronautical Engineering, National Cheng Kung University, in September of 1979 and graduated with a degree of Bachelor of Science in June of 1983. After completing two years of mandatory military service in the Republic of China Air Force, he joined the Institute of Aeronautics and Astronautics, National Cheng Kung University, as a teaching assistant in August of 1985. He was awarded the Taiwan Aerospace Scholarship in 1986. In August of 1986 he attended Purdue University to pursue an advanced degree in the School of Aeronautical and Astronautical Engineering. In May of 1987 he returned to his home country and married Rong-Jang Teresa Wu. He then came back to Purdue University to continue his study in July of 1987. Upon graduation from Purdue University with a degree of Master of Science in December of 1987, he continued to pursue a degree of Doctor of Philosophy at Purdue University.