NUMERICAL ANALYSIS
OF TWO DIMENSIONAL INCOMPRESSIBLE
LAMINAR AND TURBULENT BOUNDARY LAYERS

by

Şahnur Agaik

B.S. in M.E., Boğaziçi University, 1981

Submitted to the Institute for Graduate Studies in

Science and Engineering in partial fulfillment of

the requirements for the degree of

Master of Science

in

Mechanical Engineering

Boğaziçi University

1984

yaşamımın her evresinde

sevgi dolu desteğini

hiç bir zaman esirgemeyen

biricik

D A Y I M ' a

NUMERICAL ANALYSIS OF TWO DIMENSIONAL INCOMPRESSIBLE LAMINAR AND TURBULENT BOUNDARY LAYERS

APPROVED BY

Doç. Dr. Muhsin MENGÜTÜRK
(Thesis Supervisor)

Doç. Dr. Amable HORTAÇSU

Prof. Dr. Akın TEZEL

DATE OF APPROVAL

ACKNOWLEDGEMENTS

I would like to express my gratitude to my Thesis Supervisor Doç. Dr. Muhsin Mengütürk for his continuous support and constructive criticism during the development of the present study.

I am deeply indebted to Dr. Doğan Güneş since his guidance and enormous help enabled me to complete the computer code.

My sincere thanks are due Miss Gülseda Dizdaroğlu whose patience and morale support are greatly appreciated.

NUMERICAL ANALYSIS OF TWO DIMENSIONAL INCOMPRESSIBLE LAMINAR AND TURBULENT BOUNDARY LAYERS

The present study deals with the numerical solution of steady, two dimensional, incompressible, laminar and turbulent boundary layers. Existing numerical methods have been reviewed and a solution method is constructed which uses Finite Difference Scheme in the streamwise direction and Finite Element Method in the normal direction. Based on this solution method three versions of a computer code have been developed. Similar, non-similar and turbulent flows are considered as applications to test the code. The results which are summarized in tabular and graphical form, are compared with exact and other available numerical solutions. It is observed that the performance of the program primarily depends on the number and size of the elements used, the type of the flow and the approach of separation. The accuracy is more than 99 per cent for most of the flows and 94 per cent for turbulent flow. It is concluded that the computer code developed is applicable. Finally, recommendations are given for which the computer method can be generalized and improved.

İKİ BOYUTLU SIKIŞTIRILAMAYAN LAMİNER VE TÜRBÜLANSLI SINIR KATMANLARININ NÜMERİK ANALİZİ

Bu çalışmada; sönümlenmiş, iki boyutlu, sıkıştırılamayan, laminer ve türbülanslı sınır katmanlarının nümerik çözümü konu edilmiştir. Var olan nümerik yöntemler araştırılmış ve yatay doğrultuda sonlu farklar, dikey doğrultuda ise sonlu elemanlar yöntemlerini kullanan bir çözüm metodu oluşturulmuştur. Bu çözüm yöntemine bağlı olarak üç ayrı tipi olan bir bilgisayar programı geliştirilmiştir. Geliştirilen bilgisayar programını sınama amacıyla, benzer, benzer olmayan ve türbülanslı akışlar uygulama olarak ele alınmıştır. Tablo ve şekiller halinde özetlenmiş olan sonuçlar, kesin çözümler ve başka nümerik yöntem çözümleriyle karşılaştırılmıştır. Program performansının öncelikle; kullanılan elemanların sayı ve büyüklüğüne, akış tipine ve ayrışım noktasına olan uzaklığa bağlı olduğu saptanmıştır. Bir çok akış için sonuçların doğruluk oranı yüzde 99'dan yüksek ve türbülanslı akışlar için yüzde 94tür. Geliştirilen bilgisayar programının uygulanabilir olduğu sonucuna varılmıştır. Son olarak, bilgisayar yönteminin genelleştirilmesi ve geliştirilmesi için tavsiyelerde bulunulmuştur.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$\underline{\underline{A}}$	Part of the element stiffness matrix
b	Parameter for Pohlhausen type velocity profile
$\underline{\underline{B}}$	Part of the element stiffness matrix
$\underline{\underline{C}}$	Part of the element stiffness matrix
$\underline{\underline{D}}$	Part of the element stiffness matrix
f	Dimensionless stream function; function
h_1	Initial grid length
H_{ij}	Second order Hermitian polynomials
i	Index of the global stiffness matrix
I	Index of the banded matrix; integral value
j	Index of the global stiffness matrix
J	Index of the banded matrix; total number of nodes
k	Geometric grid parameter
l	Length from the leading edge
L	Element length
m	Element number
$\underline{m}_e$	Element force vector
$\underline{\underline{M}}$	Global stiffness matrix
$\underline{\underline{M}}_e$	Element stiffness matrix
N_b	Half band-width of the matrix
N_e	Number of elements
N_v	Number of variables
p	Pressure
P	Dimensionless pressure
q_i	Factors of linearized momentum equation
r	Radial distance from the axis of symmetry
Re (R_x)	Reynolds number

t	Time
T	Dimensionless time
u	Velocity along the x-axis
U	Dimensionless velocity along the X-axis
U_e	Potential flow velocity (edge velocity)
U_∞	Free stream velocity
v	Velocity along the y-axis
V	Dimensionless velocity along the Y-axis
w	Weights of Gaussian integration
x	Dimensional coordinate
X	Dimensionless x-coordinate
y	Dimensional coordinate
Y	Dimensionless y-coordinate
z	Transformed η-coordinate
α	Parameter for finite difference
β	Pressure gradient parameter
δ	Boundary layer thickness
δf	Variation of the dimensionless stream function
$\Delta\eta$	Increment of the η coordinate
ε_m	Eddy viscosity
ε^+	Dimensionless eddy viscosity
ζ	Parameter for eddy viscosity
η	Transformed y-coordinate
μ	Dynamic viscosity
ν	Kinematic viscosity
ξ	Transformed x-coordinate
ρ	Density
σ	Coordinate of the Gaussian integration
τ	Shearing stress

φ	Normalized η coordinate
ψ	Stream function

Subscripts:

e	Edge; element
i	Inner layer; number of Gaussian integration points
n	Present station
n-1	Previous station
o	Outer layer
v	Variables
w	Quantity at the wall
δ	Quantity at the boundary layer thickness
∞	Quantity at the free stream

Superscripts:

(i)	Present iteration
(i+1)	Next iteration
k	Parameter of symmetricity
T	Matrix transpose
'	Fluctuating quantities; first derivative
'' , '''	Derivatives
*	Quantity of the global system
—	Time - averaged quantity

CHAPTER I

INTRODUCTION

The second half of the XIXth century witnessed great progress in the area of science and technology. The development of fluid mechanics consisted of two major components which diverged from each other. The scientific component, namely theoretical hydrodynamics, was almost complete. Its results on the other hand did not match with experiments. Consequently, technological needs gave way to the second component which was of empirical nature.

In 1904 Ludwig Prandtl's practical considerations was able to unify these two components in order to analyze viscous flows accurately. The experimentally proven theoretical considerations showed that high Reynolds number flow around a solid body could be analyzed in two parts:

a- The analysis of the thin layer in the neighbourhood of the solid body- boundary layer- and

b- The analysis of the remaining region outside this layer- potential flow region-.

The theory of the boundary layer has developed at a very fast rate. Several areas of fluid dynamics, particularly aerodynamics and gas dynamics enhanced its improvement. The calculation of skin friction drag plays the most important role in its use; e.g., flat plate at zero incidence, drag of a ship, aeroplane wing, turbine blade. Complete design of various components can only be achieved by the relevant treatment of the boundary layer theory. Stall phenomenon can only be explained by this powerful and practically very important theory.

Boundary layers may be of different types. For example,

- Two or three dimensional boundary layers
- Incompressible or compressible boundary layers
- Laminar or turbulent boundary layers
- Various combinations of the above

This theory yields a set of partial differential equations which are non-linear and parabolic the derivation of which will be reviewed in the next chapter. The solution of these equations poses several difficulties because of their non-linearity. Furthermore, turbulent boundary layers do not permit analytical solution at all. Consequently, a literature survey is presented in Chapter III to illustrate many diverse approaches to their solution. Boundary layer flow solutions may be classified as follows.

a- Analytical solutions:

This way two dimensionallaminar boundary layers can be solved by similarity transformation and various series methods.

b- Numerical solutions:

i- Integral Methods:

These methods were developed in the first quarter of the current century for the purpose of obtaining at least some approximate solutions to the boundary layer flows which are impossible to solve analytically.

ii-Differential Methods:

The theory for the differential methods of solution had already been in existance in the beginning of the XXth century. Their wide spread application, however, was only possible after the invent of the digital computers. Application of these methods by the use of the new generation high speed computers yields accurate solutions to boundary layer flows.

The principal objective of the present study is to develop a finite-element computer code to solve laminar or turbulent two dimensional incompressible boundary layer flows. The solution method used is largely based on the work by Bismarck-Nasr(34).

Chapter II of the present work outlines the basic steps leading to the classical boundary layer formulation of Prandtl.

Chapter III contains a brief literature survey on the existing numerical methods of solving boundary layer flows.

Chapter IV treats in sequence the details of the present solution method, the finite-element approach used and the computercode developed. Further details are given in the Appendices.

Some applications of the code are reported in Chapter VI and the ensuing conclusion are listed in Chapter VII.

CHAPTER II

THEORY OF THE TWO DIMENSIONAL BOUNDARY LAYERS

2.1 Derivation of Boundary Layer Equations

The Boundary Layer Theory deals with high Reynolds number flows in which the velocities are of the order of the free stream velocity, U (Fig. 1) with the exception of the immediate neighbourhood of the surface of the slender body which is also depicted in the figure mentioned. Unlike potential flow the fluid does not slide over the wall but adheres to it. The velocity reaches the value of the free stream in a very thin layer, the so called boundary layer (German, Grenzschichten or Reibungsschichten; French, couches limites; Turkish, sınır katmanı). Consequently, there are two regions to be considered in detail:

1. In the boundary layer, the velocity gradient normal to the wall is very large ($\partial u/\partial y$). Even if the dynamic viscosity, μ ,is small, the shearing stress

$$\tau = \mu \, (\partial u/\partial y) \tag{2.1}$$

   may assume very large values.
2. In the remaining region outside the boundary layer, the influence of viscosity is unimportant, hence the flow may be treated as being potential.

Prandtl's formulation of the boundary layer flow by considering an asymptotic form of the governing fluid flow equations as the Reynolds

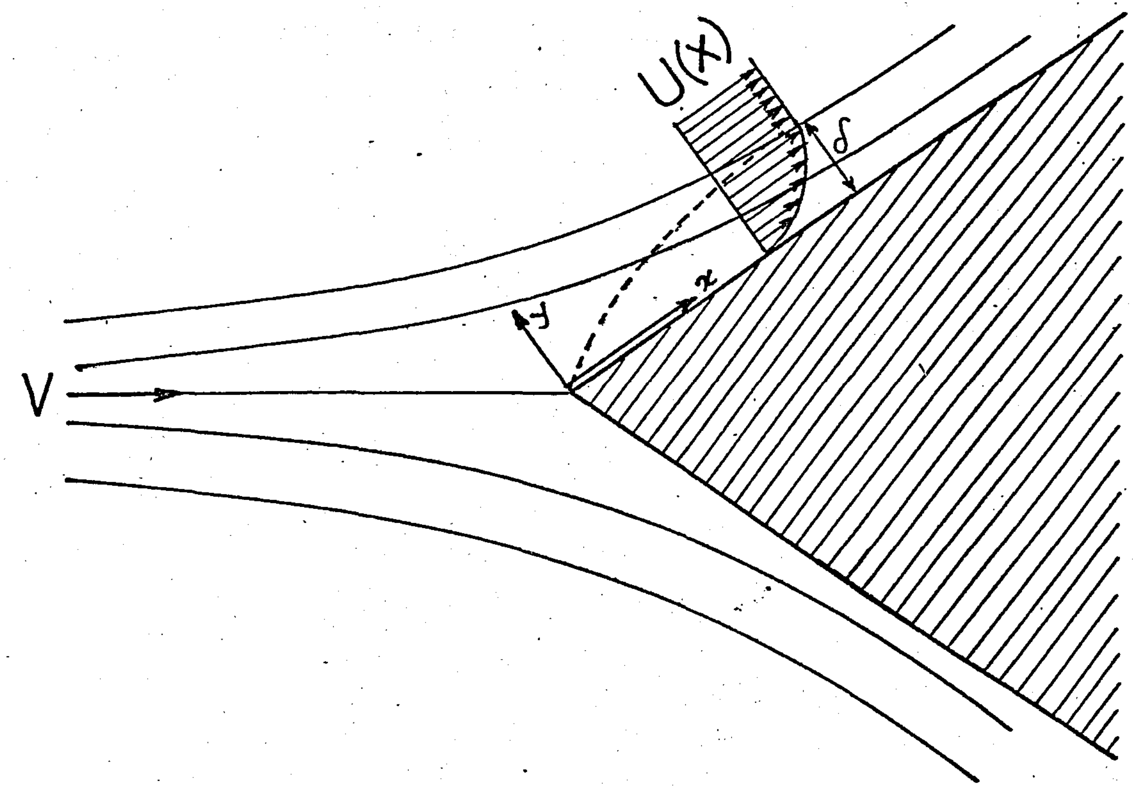


FIGURE 1.-- Development of a boundary layer along a solid body.

number approaches infinity. For this purpose the Reynolds number is defined as

$$Re \equiv \frac{l\ U_\infty}{\nu} \tag{2.2}$$

where l denotes the length from the leading edge, U the free stream velocity and ν, thekinematic viscosity.

It should be pointed out that in the actual case the Reynolds number is never infinite, although it is very large.

The full equations of motion for two-dimensional incompressible flow (Navier-Stokes Equations) are given by

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \tag{2.3.1}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) \tag{2.3.2}$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{2.4}$$

These equations can be cast into a non-dimensional form by using the following non-dimensional variables.

$$X \equiv x/l \qquad Y \equiv \sqrt{Re}\ y/l \tag{2.5.1,2}$$

$$U \equiv u/U_\infty \qquad V \equiv \sqrt{Re}\ v/U_\infty \tag{2.5.3,4}$$

$$T \equiv U_\infty\ t/l \qquad P \equiv P/\rho U_\infty^2 \tag{2.5.5,6}$$

Consequently, Eqs.(2.3.1,2) and (2.4) become

$$\frac{\partial U}{\partial T} + U\frac{\partial U}{\partial X} + V\frac{\partial U}{\partial Y} = -\frac{\partial P}{\partial X} + \frac{1}{Re}\frac{\partial^2 U}{\partial X^2} + \frac{\partial^2 U}{\partial Y^2} \tag{2.6.1}$$

$$\frac{1}{Re}\left(\frac{\partial V}{\partial T} + U\frac{\partial V}{\partial X} + V\frac{\partial V}{\partial Y}\right) = -\frac{\partial P}{\partial Y} + \frac{1}{Re}\frac{\partial^2 V}{\partial X^2} + \frac{\partial^2 V}{\partial Y^2}\frac{1}{Re} \tag{2.6.2}$$

$$\frac{\partial U}{\partial X} + \frac{\partial V}{\partial Y} = 0 \tag{2.7}$$

The derivation of Eqs.(2.6.1,2) and (2.7) is shown in Appendix A1. When the assumption of Re having an infinite value is applied, Eqs.(2.6.1,2) reduce to

$$\frac{\partial U}{\partial T} + U\frac{\partial U}{\partial X} + V\frac{\partial U}{\partial Y} = -\frac{\partial P}{\partial X} + \frac{\partial^2 U}{\partial Y^2} \tag{2.8.1}$$

$$0 = -\frac{\partial P}{\partial Y} \tag{2.8.2}$$

while Eq.(2.7) remains the same.

If Eqs.(2.8.1) and (2.7) are transformed back into the dimensional form with the additional assumption of steady flow, the system of equations for steady two dimensional, incompressible, laminar boundary layers simplifies to:

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{dp}{dx} + \nu\frac{\partial^2 u}{\partial y^2} \tag{2.9}$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{2.4}$$

with the boundary conditions

$u = 0$, $v = 0$ at $y = 0$ (zero velocity at the wall)

$u = U(x)$ at $y = \delta$ (smooth merging into the potential velocity) (2.10)

Eqs.(2.4,9,10) are commonly known as Prandtl's equations of the boundary layer flow. These equations can be generalized to two-dimensional (and axisymmetric), compressible, turbulent boundary layers as follows(21):

Continuity:

$$\frac{\partial}{\partial x}(r^k \rho u) + \frac{\partial}{\partial y}(r^k \rho v) = 0 \tag{2.11}$$

where k denotes the parameter of axisymmetricity (k=1, for axisymmetric case; k=0, for two-dimensional case); r, the distance from the axis of symmetricity. As can be seen Eq.(2.11) has the form for the compressible flow case.

Momentum:

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = - \frac{dp}{dx} + r^{-k} \frac{\partial}{\partial y}\left(r^{k}\left(\mu \frac{\partial u}{\partial y} - \rho \overline{u'v'} \right)\right) \qquad (2.12)$$

with the same boundary conditions Eq.(2.10).

It is important to note that the term $\overline{u'v'}$ in Eq.(2.12) represents the accelerated rate of momentum transfer due to the turbulent velocity fluctuations. The considerations leading to Eqs.(2.11,12) are outlilned in Appendix A2.

CHAPTER III

LITERATURE SURVEY

The solution of Eqs.(2.10,11,12) has been extensively studied since the beginning of the current century. These parabolic non-linear partial differential equations were first treated by Blasius(1) who employed a similarity transformation and succeeded in obtaining an exact solution to the laminar boundary layer flow over flat plate.

Falkner and Skan(2) obtained analytical solutions of laminar boundary layers with constant pressure gradient parameters. This way they could include either favourable ($\beta>0$) or adverse ($\beta<0$) pressure gradients. Once again similarity transformation was the major tool leading to an exact solution of every different β-case.

The next step was the treatment of the case of variable pressure gradients. Howarth(3) solved the laminar boundary layer with linearly decreasing pressure gradient parameter and determined the value of before the boundary layer separates from the surface.

Since this problem had an immediate and necessary application, approximate methods of varying accuracy which go beyond the formal processes of expansions in series were devised by Thwaites(4) and Pohlhausen(5). For this purpose the integral forms of Eqs.(2.10-12) were employed. Pohlhausen's method assumes a family of analytically defined velocity distributions. It gives poor results in regions of rising pressure.

Thewaites' method, on the other hand, combines several parameters of the boundary layer and integrates them. This method obtains results which are closer to the exact solution than the results of Pohlhausen's method.

A German scientist, Görtler(6), introduced an approximation method which was suitable for solving non-similar boundary layers on desk calculators. The advantage of the Görtler Series Method is that many of the coefficients can be worked out once and for all and looked up for the solution of a particular problem. This method uses a special kind of transformation and it can obtain solutions for complex geometires, aswell.

Finite Difference Methods (FDM) were extensively used to obtain field solution of boundary layer equations. An example for the application of the method can be found in Werle and Davis' paper(7) where the effect of adverse pressure gradient was studied past a parabola at an angle of attack. Separation point of the boundary layer was also determined. Another application of FDM was performed by Cebeci and Keller(9) to solve Falkner-Skan problem numerically.

The invent and improvement of digital computers in 1970s enabled the researchers to apply Finite Element Method (FEM) extensively. Oden and Wellford Jr.(10) solved the Blasius problem by FEM obtaining the exact solution. Tadros and Kirkhope(11) applied the same method for the same problem to study the effect of different element shapes and different approximation polynomials. The analysis showed that higher order polynomials play more important role in obtaining the exact results than the increase of number of elements.

Lynn and Alani(12) applied the Least Squares FEM for two dimensional laminar boundary layer analysis. They performed exhaustive numerical investigations in the retarded flow over a plate, flow past a circular cylinder the flow past an elliptic cylinder. The results obtained are in good agreement with exact solutions.

Another step to complication is the inclusion of turbulunce. A published lecture of Bradshaw(13) discusses several aspects of Turbulent Boundary Layers (TBL). Since there are no analytical solutions to turbulent boundary layer flows he especially stresses the importance of numerical methods based on the differential form of the boundary layer equations. Biringen and Levi(14) solved this problem for two dimensions by FDM using two-equation model of turbulence. Their results are in good agreement with previous methods and experiments.

Rastogi and Rodi(15), on the other hand, solved the three-dimensional problem using FDM and k-ε model of turbulence. Their results are also reasonable. Keller and Cebeci(16) solved the same problem for two-dimensional case using FDM and eddy viscosity model of turbulence. This study yielded accurate results effectively. Since the boundary layer has a parabolic character marching in the x-direction is possible. Using an initial velocity profile the authors obtained solutions at consequtive stations along the x-direction by an iterative method.

Both for integral and differential calculation methods Dean(17) developed a formula for the complete velocity profile in turbulent boundary layers. Comparison with experimental data shows the applicability of the proposed formulation.

Yeung and Yang(18) applied the method of integral relations (MIR) for incompressible two-dimensional TBL. Although their results are good for zero and favourable pressure gradients, they deviate drastically from experiments for adverse pressure gradients.

Turbulence modelling plays a very important role in the application of numerical methods. Rodi(19) discusses the merits and demerits of various modelling methods. Launder and Spalding(20) present in their book numerous mathematical models for the same purpose. The widely accepted and successfully applied "Eddy Viscosity Modelling" technique is explained by Cebeci and Smith(21) and Cebeci and Bradshaw(22) in detail.

Wheeler and Johnston(23) predicted three-dimensional TBL using several turbulence modelling methods. The high sensitivity of the results obtained to the free stream pressure gradient in separating flow cases is also demonstrated. The same problem is attacked by Bradshaw and Ferriss(24). Their results are in good agreement with experiment for incompressible case. The inclusion of compressibility, on the other hand, brings some problems with it.

Sharma et al.(25) investigated both experimentally and numerically the boundary layer development on turbine airfoil suction surfaces. This paper is a typical example which describes the application of both experimental and numerical methods for the design of the airfoil of a turbine blade.

CHAPTER IV

METHOD OF SOLUTION

4.1 Introduction

The aim of the present study is to solve the boundary layer Eqs.(2.11,12) for incompressible, two-dimensional, laminar, similar and non-similar and for turbulent cases. For this purpose a method of solution is developed which is largely based on the paper of Bismarck-Nasr(34).

The governing equations are reduced to the two-dimensional case. The singularity at the leading edge is removed by the application of Levy-Lees transformation. Momentum equation is linearized by Newton's method. The method of solution uses finite difference method in the streamwise direction and finite element method in the normal direction.

The computer program developed is based on the method of solution is presented in Chapter V.

4.2 Reduction of Governing Equations to Two-Dimensional Case and Transformation to Remove the Singularity at the Origin

The Reynolds shear stress in the Eq.(2.12) is related to the mean velocity field through eddy viscosity relationship:

$$-\rho\overline{u'v'} = \rho\varepsilon_m\frac{\partial u}{\partial y} \tag{4.1}$$

Substituting Eq.(4.1) into Eq.(2.12) and noting that k=0 for two-dimensional flow and ρ= constant for incompressible flow, Eqs.(2.11,12) become

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \qquad (2.4)$$

$$\rho u \frac{\partial u}{\partial x} + \rho v \frac{\partial u}{\partial y} = - \frac{dp}{dx} + \frac{\partial}{\partial y} \left(\mu \frac{\partial u}{\partial y} + \rho \varepsilon_m \frac{\partial u}{\partial y} \right) \qquad (4.2)$$

with the same boundary conditions Eq.(2.10).

These equations posess a singularity at the origin. This singularity will be removed by using the Levy-Lees transformation (for details of this transformation see Hayes and Probstein(26)). The Levy-Lees transformation causes the coordinates to be stretched according to the equations

$$d\xi = \rho\mu \, Ue \, dx \qquad (4.3.1)$$

$$d\eta = (\rho Ue / \sqrt{2\xi}\,) \, dy \qquad (4.3.2)$$

A stream function, $\psi(x,y)$, is defined according to the equations

$$\rho u = \partial\psi/\partial y \qquad (4.4.1)$$

$$\rho v = -\partial\psi/\partial x \qquad (4.4.2)$$

A dimensionless stream function, $f(\xi,\eta)$, is related to $\psi(x,y)$ as follows.

$$\psi(x,y) = \sqrt{2\xi} \, f(\xi,\eta) \qquad (4.5)$$

Hence, the partial derivative operators in the (x,y) coordinate system can be related to those in the (ξ,η) system as

$$\left(\frac{\partial}{\partial x}\right) = \rho\mu \, Ue \left(\frac{\partial}{\partial \xi} + \frac{\partial \eta}{\partial \xi}\frac{\partial}{\partial \eta}\right) \qquad (4.6.1)$$

$$\left(\frac{\partial}{\partial y}\right) = \frac{\rho Ue}{\sqrt{2\xi}} \left(\frac{\partial}{\partial \eta}\right) \qquad (4.6.2)$$

The relations are employed to yield

$$\left(\frac{\partial \psi}{\partial \xi}\right) = \sqrt{2\xi} \left(\frac{f}{2\xi} + \frac{\partial f}{\partial \xi}\right) \qquad (4.7.1)$$

$$\left(\frac{\partial \psi}{\partial \eta}\right) = \sqrt{2\xi} \, f' \qquad (4.7.2)$$

where

$$f' \equiv \partial f/\partial \eta \tag{4.8}$$

$$u = Ue\ f' \tag{4.9.1}$$

$$v = -\mu\ Ue\sqrt{2\xi}\ \left(\frac{f}{2\xi} + \frac{\partial f}{\partial \xi} + f'\ \frac{\partial \eta}{\partial \xi} \right) \tag{4.9.2}$$

If the terms of (4.2) are evaualted separately in the transformed coordinates and the necessary cancellations are performed, the following equation is obtained:

$$((1+ \varepsilon^{+})\ f'')' + ff'' + \beta(1- (f')^2) = 2\xi\ \left(f'\ \frac{\partial f'}{\partial \xi} - f''\ \frac{\partial f}{\partial \xi}\right) \tag{4.10}$$

where the pressure gradient parameter, β, and the dimensionless eddy viscosity, ε^{+}, are given by

$$\beta \equiv \frac{2\xi}{Ue}\ \frac{dUe}{d\xi} \tag{4.11}$$

and

$$\varepsilon \equiv \varepsilon_m\ /\ \nu \tag{4.12}$$

The superscript ( )' indicates the derivative with respect to η. The boundary conditions of Eq.(4.10) are obtained by transforming Eq.(2.10) into the Levy-Lees coordinate system (ξ,η).

$$f = 0 \quad , \quad f' = 0 \quad \text{at} \quad \eta = 0$$

$$f' = 1 \quad \text{at} \quad \eta = \eta_\infty \tag{4.13}$$

The details of the above development are given in Appendix B.

4.3 Modelling and Transformation of Eddy Viscosity

A two region eddy viscosity model is used un the present analysis. In this model, the boundary layer is assumed to be composed of two regions.

The eddy viscosity expression used in the inner region is based on Prandtl's mixing length theory and incorporates the modifications introduced by Van Driest(35) to account for the viscous sublayer close to the wall and by Cebeci(27) to account for flows with non-zero pressure gradient. This expression is given by

$$\varepsilon_i = (0.4y)^2 \left(1 - \exp\left(-y\left(\frac{\tau w}{\rho} + \frac{dp}{dx}\frac{y}{\rho}\right)^{1/2} / 26\nu\right)\right)^2 \left|\frac{\partial u}{\partial y}\right| \qquad (4.14)$$

In the outer region, the eddy viscosity is assumed to be constant but includes Klebanoff's intermittency factor(36) :

$$\varepsilon_o = 0.0168 \left| \int^{\infty} (Ue-u)\, dy \right| \left(1 + 5.5 \left(\frac{y}{\delta}\right)^6\right)^{-1} \qquad (4.15)$$

The inner and outer eddy-viscosity expressions, ε_i and ε_o, are matched by the requirement of continuity, i.e. if $\varepsilon_i = \varepsilon_o$ then ε_o is calculated only.

Upon application of the Levy-Lees transformation given by Eqs.(4.3.1,2), Eqs.(4.14,15) become

$$\varepsilon_i^+ = 0.16\zeta\, |f''|\, \eta^2 \left(1 - \exp\left(-(\zeta\eta/26)(|f''_w/\zeta - \beta\eta/\zeta|)^{1/2}\right)\right)^2 \qquad (4.16)$$

$$\varepsilon_o^+ = 0.0168\zeta \left| \int_0^{\infty} (1-f')\, d\eta \right| \left(1+5.5\left(\frac{\eta}{\eta_\infty}\right)^6\right)^{-1} \qquad (4.17)$$

where

$$\zeta \equiv \sqrt{2\xi}/\mu \qquad (4.18)$$

The details of this transformation can be found in Appendix C.

4.4 Marching Solution in Streamwise Direction

Since Eq.(4.10) is a parabolic partial differential equation, a solution can be obtained by marching up in the streamwise direction. For this purpose, the boundary layer is divided into a number of stations along the ξ-direction and the ξ-derivatives are approximated by a two-point finite difference formula. Thus, the right-hand-side of Eq.(4.10) is replaced by

$$\frac{\xi_n + \xi_{n-1}}{\xi_n - \xi_{n-1}}(f'_n\,(f'_n - f'_{n-1}) - f''_n\,(f_n - f_{n-1})),$$

where the subscripts n and n-1 refer to the present and preceding stations, respectively.

Hence, Eq.(4.10) can be written as a non-linear ordinary differential equation as follows

$$((1+\varepsilon^+)f'')' + ff'' + \beta\,(1-(f')^2) = \alpha(f'(f' - f'_{n-1}) - f''(f - f_{n-1})) \quad (4.19)$$

where

$$\alpha = \frac{\xi_n + \xi_{n-1}}{\xi_n - \xi_{n-1}} \quad (4.20)$$

In Eq.(4.19) and the following equations, the subscript n under f, f', etc. is omitted for convenience.

4.5 Linearization of Momentum Equation

The momentum equation (4.19) is linearized by using Newton's iterative method. The higher order iterates are replaced by the following expressions

$$f^{(i+1)} = f^{(i)} + \delta f^{(i)} \quad (4.21.1)$$

$$f'^{(i+1)} = f'^{(i)} + \delta f'^{(i)} \quad (4.21.2)$$

$$f''^{(i+1)} = f''^{(i)} + \delta f''^{(i)} \quad (4.21.3)$$

$$f'''^{(i+1)} = f'''^{(i)} + \delta f'''^{(i)} \quad (4.21.4)$$

Consequently, the following ordinary differential equation is obtained:

$$q_1\delta f''' + q_2\delta f'' + q_3\delta f' + q_4\delta f + q_5 = 0 \quad (4.22)$$

where

$$q_1 = 1 + \varepsilon^{+} \tag{4.23.1}$$

$$q_2 = (1 + \varepsilon^{+})' - \alpha f_{n-1} + (1+\alpha) f \tag{4.23.2}$$

$$q_3 = \alpha f'_{n-1} - 2(\beta+\alpha) f' \tag{4.23.3}$$

$$q_4 = (1+\alpha) f'' \tag{4.23.4}$$

$$q_5 = (1+\varepsilon^{+}) f''' + \{(1+\varepsilon^{+})' - \alpha f_{n-1}\} f'' + \alpha f'_{n-1} f'$$

$$+ (1+\alpha) f f'' - (\beta+\alpha)(f') + \beta \tag{4.23.5}$$

The derivation of Eqs.(4.22,23) is given in Appendix D.

An initial guess is required for this iterative solution scheme which must satisfy the boundary conditions. The boundary conditions given by Eq.(4.13) take the form

$$\delta f(\xi, 0) = 0 \tag{4.24.1}$$

$$\delta f'(\xi, 0) = 0 \tag{4.24.2}$$

$$\delta f'(\eta, \eta_\infty) = 0 \tag{4.24.3}$$

4.6 Galerkin Finite-Element Formulation

After the solution domain is divided into a number of stations in the ξ-direction, a Galerkin type finite element method is used at each station to solve Eq.(4.22) subject to the boundary conditions given by Eqs.(4.24.1-3). For this purpose, each ξ-station is further subdivided into a number of nodes. The sections between two cosequtive nodes are called elements. In these elements the unknown variables are approximated by known tral funtions and unknown nodal values.
The choice of trial functions is a crucial step for the finite-element formulation. Eq.(4.22) is a third order differential equation. Consequently, it is necessary to choose trial functions such that they satisfy the

condition of second order continuity, C_2. The details of the choice of the trial functions can be found in Zienkiewicz(28) and Huebner(29).

4.6.1 Second Order Hermitian Polynomials as Trial Functions

The reason for choosing second order Hermitian Polynomials as trial functions in the present study is that they satisfy the C_2 continuity. Detailed information about these polynomials can be found in Fort(30), Hildebrand(31) and Huebner(29).

In terms of Hermitian polynomials, H, the field variable δf is approximated by

$$\delta f = H_{01}\delta f_1 + H_{11}\delta f_1' + H_{21}\delta f_1'' + H_{02}\delta f_2 + H_{12}\delta f_2' + H_{22}\delta f_2'' \tag{4.25}$$

where the first subscript of H refers to the order of derivative and the second signifies the node number of the element. Similarly, the subscript of δf stands for the node number while ( )' denotes derivative with respect to η, as before.

As shown in Appendix E, the Hermitian polynomials are given as

$$H_{01} = 1 - 10z^3 + 15z^4 - 6z^5 \tag{4.26.1}$$

$$H_{11} = L\,(z - 6z^3 + 8z^4 - 3z^5) \tag{4.26.2}$$

$$H_{21} = (L^2/2)(z^2 - 3z^3 + 3z^4 - z^5) \tag{4.26.3}$$

$$H_{02} = 10z^3 - 15z^4 + 6z^5 \tag{4.26.4}$$

$$H_{12} = L(-4z^3 + 7z^4 - 3z^5) \tag{4.26.5}$$

$$H_{22} = (L^2/2)(z^3 - 2z^4 + z^5) \tag{4.26.6}$$

where

$$L = \eta_2 - \eta_1 \tag{4.27.1}$$

$$z = (\eta - \eta_1)/(\eta_2 - \eta_1) \tag{4.27.2}$$

In the above the subscripts 1 and 2 denote two consecutive nodes (Fig.2). The details of the development of Eqs.(4.26.1-6) are given in Appendix E. Defining the following vectors as

$$\underline{H}^T \equiv \{ H_{01}\ H_{11}\ H_{21}\ H_{02}\ H_{12}\ H_{22} \} \tag{4.28}$$

$$\underline{\delta f}^T \equiv \{ \delta f_1\ \delta f_1'\ \delta f_1''\ \delta f_2\ \delta f_2'\ \delta f_2'' \} \tag{4.29}$$

the the field variable and its derivatives can be formulated as

$$\delta f = \underline{H}^T\ \underline{\delta f} \tag{4.30.1}$$

$$\delta f' = \underline{H}^{T'}\ \underline{\delta f} \tag{4.30.2}$$

$$\delta f'' = \underline{H}^{T''}\ \underline{\delta f} \tag{4.30.3}$$

$$\delta f''' = \underline{H}^{T'''}\ \underline{\delta f} \tag{4.30.4}$$

where

$$\frac{dH}{d\eta} = \frac{dH}{dz}\frac{dz}{d\eta} = \frac{dH}{dz}\frac{1}{L} \tag{4.31}$$

4.6.2 Galerkin Finite Element Approximation of Field Problem

Eqs.(4.30.1-4) are substituted into Eq.(4.22) to yield:

$$q_1\underline{H}^{T'''}\underline{\delta f} + q_2\underline{H}^{T''}\underline{\delta f} + q_3\underline{H}^{T'}\ \underline{\delta f} + q_4\underline{H}^T\underline{\delta f} = -\ q_5 \tag{4.32}$$

According to the Galerkin method the integral of this equation weighted with the trial functions gives

$$\int(\underline{H}q_1\ \underline{H}^{T'''} + \underline{H}q_2\ \underline{H}^{T''} + \underline{H}q_3\ \underline{H}^{T'} + \underline{H}q_4\underline{H}^T)\ \underline{\delta f}\ L\ dz = -\int \underline{H}q_5\ L\ dz \tag{4.33}$$

The integrals shown in Eq.(4.33)are performed over the entire domain and can be thought of as the sum of the integrals carried out over the individual elements. In each element, one can write

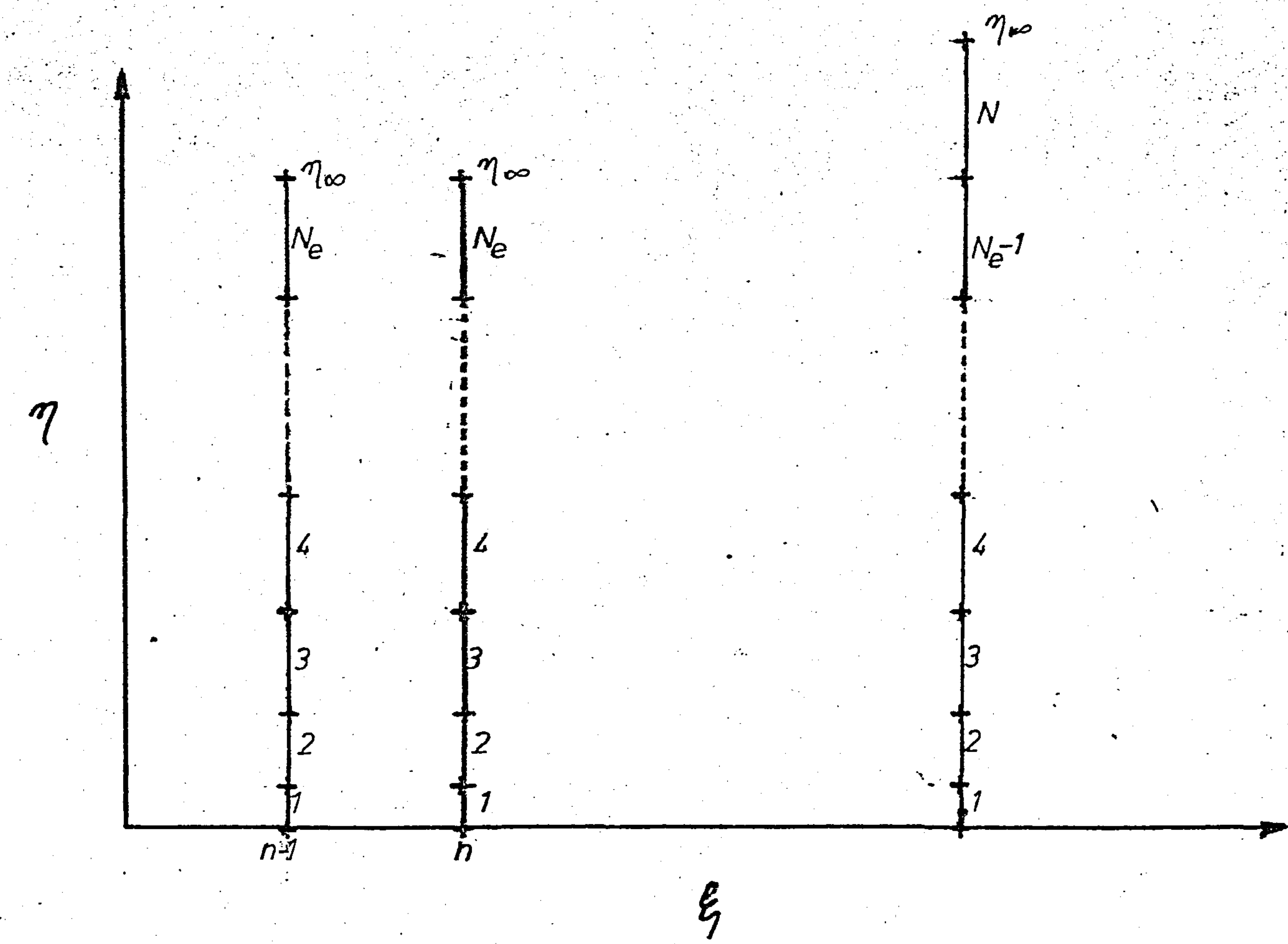


FIGURE 2.-- Domain discretization for the preparation of the solution

$$\underline{\underline{M}}_e \, \underline{\delta f}_e = \underline{m}_e \tag{4.34}$$

where $\underline{\underline{M}}_e$, $\underline{\delta f}_e$ and $\underline{m}_e$ are the element stiffnes matrix, element unknown vector and element force vector, respectively. Application of the standard assembly technique of the finite element method leads to

$$\underline{\underline{M}} \, \underline{\delta f}^* = \underline{m} \tag{4.35}$$

where $\underline{\underline{M}}$, $\underline{\delta f}^*$ and $\underline{m}$ are the global stiffness matrix, global unknown vector and global force vector, respectively. Fig.3 shows the assembly technique applied for C_2-continuity. As can be observed $\underline{\underline{M}}_e$ is a (6x6) matrix and $\underline{\underline{M}}$ is a banded matrix with a half-band-width of five.

The general expressions of the element and global stiffness matrices and element and global vectors are given in Section 4.7.3 .

4.6.3 Conversion of Global Matrix to Banded Form

In order to reduce the core storage requirement the global stiffnes matrix is converted into a banded matrix form. The dotted squares of Fig.3 represent the banded matrix.

The major advantage of this conversion can be shown as follows: If N_e is the number of elements in the η-direction, there are (N_e+1) nodes as a whole. The total number of variables will be

$$N_v = 3(N_e + 1) \tag{4.36}$$

Consequently, the global stiffness matrix will be of $(N_v x N_v)$ or with $9(N_e+1)^2$. The banded matrix form, on the other hand, has always 11 columns since the half-band-width is five (J=5+1+5).That is, the size is (N x11) or $11(N_e+1)$. It is easily concluded that the storage requirement of the banded matrix depends linearly on N_e, while that of the full global stiffness matrix increases quadratically with the number of

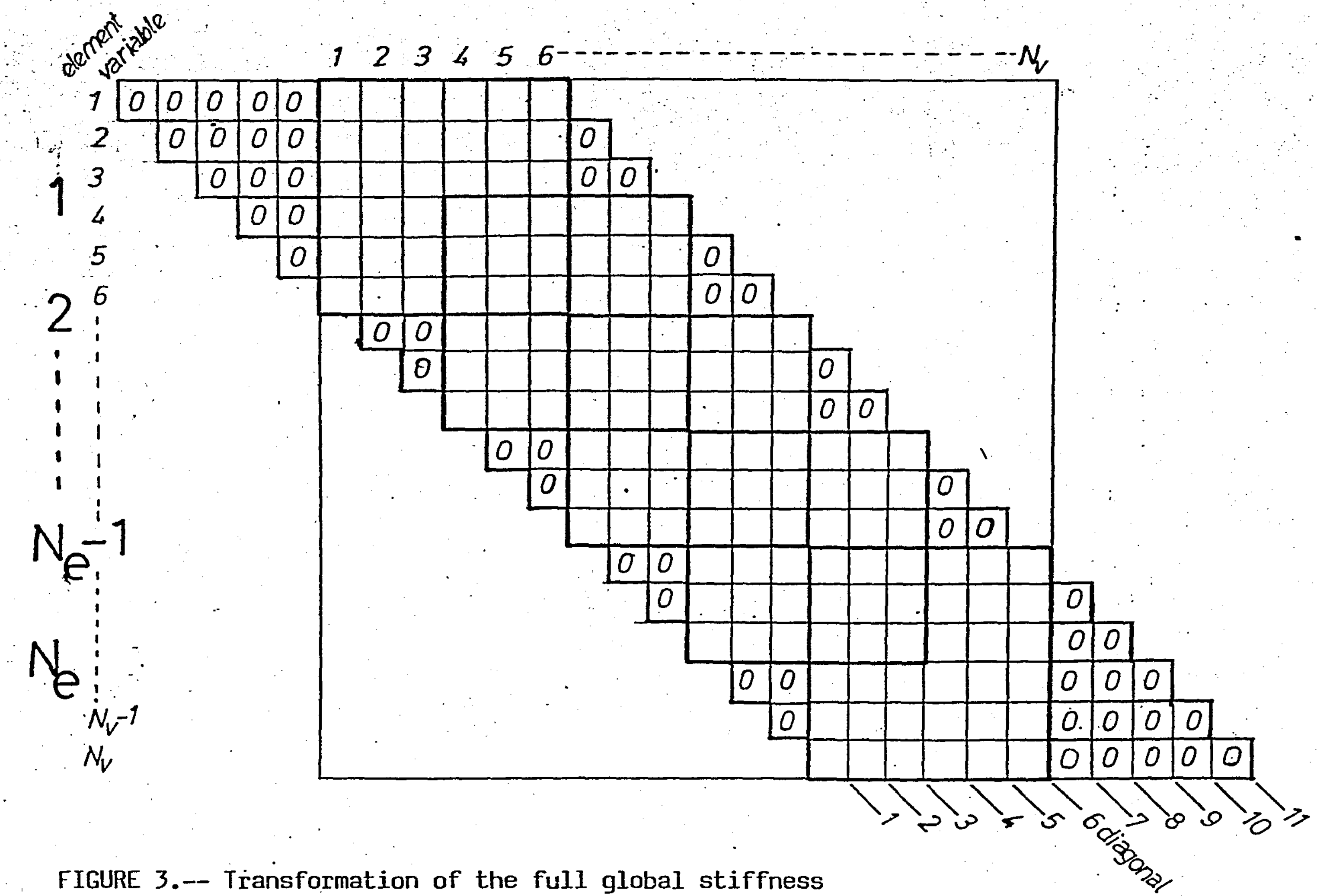


FIGURE 3.-- Transformation of the full global stiffness matrix to its banded form

elements, N_e. Fig.3 shows both the global stiffness matrix and the banded matrix. Although the row numbers are identical, the column numbers of the banded matrix must be subjected to a simple transformation. If i and j are the row and column indices of the global stiffness matrix and I and J are the row and column numbers of the banded matrix, then the following equations must be used

$$I = i \tag{4.37.1}$$

$$J = j - i + N_b + 1 \tag{4.37.2}$$

where N_b is the half-band-width (in the present analysis N_b=5).

4.7 Solution Scheme

4.7.1 Starting Velocity Profile

According to the present solution scheme, the solution obtained at the preceding station is used as the initial approximation to the solution at the present station. Therefore, a starting velocity profile is required at the first station, ξ=0. The effect of the starting profile used is established by considering three different profile types which all satisfy the boundary conditions Eq.(4.13).

i- Linear Velocity Profile:

$$f' = \varphi \tag{4.38.1}$$

where

$$\varphi = \eta/\eta_\infty$$

By differentiation and integration of the above equation,

$$f'' = 1/\eta_\infty \tag{4.38.2}$$

$$f = (\eta_\infty/2)\varphi^2 \tag{4.38.3}$$

ii-Third Order Velocity Profile:

$$f' = 1.5\varphi - 0.5\varphi^3 \tag{4.39.1}$$

similarly

$$f'' = 1.5(1-\varphi^2)/\eta_\infty \tag{4.39.2}$$

$$f = \varphi^2(6-\varphi^2)\eta_\infty/8 \tag{4.39.3}$$

iii-Pohlhausen Type Velocity Profile:

Defining

$$b \equiv \beta\eta_\infty^2/6$$

$$f' = \varphi\{2 - 2\varphi^2 + \varphi^3 + b(1-3(\varphi-\varphi^2)-\varphi^3)\} \tag{4.40.1}$$

$$f'' = \{4(1-b)\varphi^3 + 3(3b-2)\varphi^2 - 6b\varphi + (b+2)\}/\eta_\infty \tag{4.40.2}$$

$$f = \{(1-b)\varphi^3/5 + (3b-2)\varphi^2/4 - b\varphi + (b+2)/2\}\varphi^2\eta_\infty \tag{4.40.3}$$

The effect of initial velocity profiles will be discussed in Chapter VI. At the initial station the profiles f' and f'_{n-1} are the same. When the solution marches along ξ-direction they will differ from each other having their appropriate values.

4.7.2 Calculation of Stiffness Matrix and Right-Hand-Side Vector

Two alternative approaches will be suggested in the present study for the construction of the stiffness matrix and the right-hand-side vector. These are

a- the numerical integration approach that treats q_i in Eqs.(4.23.1-5) in their expanded form,

b- the analytical integration approach that treats q_i as constants.

4.7.2.a Numerical Integration Approach

Defining

$$\underline{f}^T \equiv \{f_1 \quad f'_1 \quad f''_1 \quad f_2 \quad f'_2 \quad f''_2\} \tag{4.41.1}$$

$$\underline{f}^T_{n-1} \equiv \{f_{1_{n-1}} \quad f_{1_{n-1}} \quad f_{1_{n-1}} \quad f_{2_{n-1}} \quad f_{2_{n-1}} \quad f_{2_{n-1}} \tag{4.41.2}$$

the dimensionless stream functions f and f_{n-1} and their **derivatives** can be expressed as

$$f = \underline{H}^{T} \underline{f} \qquad (4.42.1)$$

$$f' = \underline{H}^{T'} \underline{f} \qquad (4.42.2)$$

$$f'' = \underline{H}^{T''} \underline{f} \qquad (4.42.3)$$

$$f''' = \underline{H}^{T'''} \underline{f} \qquad (4.42.4)$$

$$f_{n-1} = \underline{H}^{T} \underline{f}_{n-1} \qquad (4.43.1)$$

$$f' = \underline{H}^{T'} \underline{f}_{n-1} \qquad (4.43.2)$$

The above equations allow Eqs.(4.23.2-5) to be rewritten as:

$$q_2 = (1+\varepsilon^+)' - \alpha\underline{H}^{T} \underline{f}_{n-1} + (1+\alpha) \underline{H}^{T} \underline{f} \qquad (4.44.2)$$

$$q_3 = \alpha\underline{H}^{T'} \underline{f}_{n-1} - 2(\beta+\alpha) \underline{H}^{T'} \underline{f} \qquad (4.44.3)$$

$$q_4 = (1+\alpha) \underline{H}^{T''} \underline{f} \qquad (4.44.4)$$

$$q_5 = (1+\varepsilon^+) \underline{H}^{T'''} \underline{f} + (1+\varepsilon^+)' \underline{H}^{T''} \underline{f} - \alpha\underline{H}^{T} \underline{f}_{n-1} \underline{H}^{T''} \underline{f}$$
$$+ \alpha\underline{H}^{T'} \underline{f}_{n-1}\underline{H}^{T'} \underline{f} + (1+\alpha) \underline{H}^{T} \underline{f} \underline{H}^{T''} \underline{f} - (\beta+\alpha) \underline{H}^{T'}\underline{f} + \beta \qquad (4.44.5)$$

Substitution of Eqs.(4.23.1) and (4.44.2-5) into Eq.(4.33) gives the element stiffness matrix $\underline{\underline{M}}_e$ and the right hand side vector $\underline{m}_e$ in Eq.(4.34) as follows:

$$\underline{\underline{M}}_e = \underline{\underline{A}} + \underline{\underline{B}} + \underline{\underline{C}} + \underline{\underline{D}} \qquad (4.45)$$

where

$$\underline{\underline{A}} = (1+\varepsilon^+) \int \underline{H}\, \underline{H}^{T'''} L\, dz \qquad (4.46.1)$$

$$\underline{\underline{B}} = (1+\varepsilon^+)' \int \underline{H}\, \underline{H}^{T''} L\, dz - \alpha \int \underline{H}\underline{H}^{T}\underline{f}_{n-1}\underline{H}^{T''} L\, dz$$
$$+ (1+\alpha) \int \underline{H}\underline{H}^{T}\underline{f}\, \underline{H}^{T''} L\, dz \qquad (4.46.2)$$

$$\underline{\underline{C}} = \alpha \int \underline{HH}^{T'} \underline{f}_{n-1}\ \underline{H}^{T'}\ L\ dz - 2(\beta+\alpha) \int \underline{HH}^{T'} \underline{fH}^{T'}\ L\ dz \qquad (4.46.3)$$

$$\underline{\underline{D}} = (1+\alpha) \int \underline{HH}^{T''} \underline{f}\ \underline{H}^{T}\ L\ dz \qquad (4.46.4)$$

$$\begin{aligned} \underline{m}_e = - \{ & (1+\varepsilon^+) \int \underline{HH}^{T'''} \underline{f}\ L\ dz + (1+\varepsilon^+)' \int \underline{HH}^{T''} \underline{f}\ L\ dz \\ & -\alpha \int \underline{HH}^{T} \underline{f}_{n-1}\ \underline{H}^{T''} \underline{f}\ L\ dz + \alpha \int \underline{HH}^{T'} \underline{f}_{n-1} \underline{H}^{T'} \underline{f}\ L\ dz \\ & + (1+\alpha) \int \underline{HH}^{T} \underline{fH}^{T''} \underline{f}\ L\ dz - (\beta+\alpha) \int \underline{HH}^{T'} \underline{fH}^{T'} \underline{f}\ L\ dz \\ & + \beta \int \underline{H}\ L\ dz \} \end{aligned} \qquad (4.47)$$

The above integrations will be performed by using the Gaussian Quadrature (see Zienkiewicz (28)). It should be noted that since $\underline{H}$ are polynomials of degree five. Eqs.(4.46,47) involve polynomials of degree 13. A seventh order Gaussian Quadrature formula is required to accurately integrate these polynomials. In its general form, the Gaussian Quadrature of order n for the integral of $f(\sigma)$ between normalized integration limits -1 and 1 is given by

$$I = \int_{-1}^{1} f(\sigma)\ d\sigma = \sum_{i=1}^{n} w_i\ f(\sigma_i) \qquad (4.48)$$

where w_i represent the weight assigned to each $f(\sigma_i)$. If the limits of integration are different, as in the present case, a simple transformation is required to put the given integral in the form shown above.

In the present study the points of integration σ_i are related to z_i as follows:

$$z_i = \frac{\sigma_i + 1}{2} \qquad (4.49)$$

since z_i have the limits 0 to 1.

Thus, Eq.(4.48) should be rewritten as

$$I = \sum_{i=1}^{n} w_i\ f(2z_i - 1) \qquad (4.50)$$

The points and weights used in the present Gaussian integration method are given in Table 4.1 .

TABLE 4.1

POINTS AND WEIGHTS OF GAUSSIAN NUMERICAL INTEGRATION

i	σ	z	w
1	-.949107912	.025446044	.129484966
2	-.741531186	.129234408	.279705391
3	-.405845151	.297077425	.381830051
4	.0	.5	.417959184
5	.405845151	.702922576	.381830051
6	.741531186	.870765593	.279705391
7	.949107912	.974553956	.129484966

4.7.2.b Analytical Integration Approach

If q_i (except q_5) ar etreated as constants the left-hand-side of Eq.(4.33) may be rewritten as

$$\left(q_1 \int \underline{HH}^{T'''} L\ dz + q_2 \int \underline{HH}^{T''} L\ dz + q_3 \int \underline{HH}^{T'} L\ dz + q_4 \int \underline{HH}^{T} L\ dz \right) \underline{\delta f}$$

$$= -\int \underline{H}\ q_5 L\ dz \tag{4.51}$$

The constant values of q_1, q_2, q_3 and q_4 for a given element are obtained by averaging the element nodal values.

In indicial form Eq.(4.51) can be written as:

$$\left(\int (q_1 H_i H_j''' + q_2 H_i H_j'' + q_3 H_i H_j' + q_4 H_i H_j)\ L\ dz \right) \delta f_i = -\int H_j q_5\ L\ dz \tag{4.51.a}$$

if, e.g., i=2 j=1

$$\underline{\underline{M}}_{e21} = \int (q_1 H_2 H_1''' + q_2 H_2 H_1'' + q_3 H_2 H_1' + q_4 H_2 H_1)\ L\ dz$$

after integration

$$M_{e21} = \frac{1}{L}\frac{9}{7}\,q_1 - \frac{3}{14}\,q_2 - \frac{11}{84}\,L\,q_3 + \frac{311}{4620}\,L^2\,q_4$$

The remaining elements of the matrix are obtained similarly and shown in Table 4.2 . It should be noted, that, since q_5 is not assumed to be constant the right-hand-side vector is constructed according to the procedure outlined in section 4.7.2.a .

4.7.3 Preparation of Global Stiffness Matrix and Global Force Vector

In Section 4.6.2 and 4.6.3 the main principles of the stiffness matrices are briefly explained. In this section the required algorithm for the construction of the global stiffness matrix and the global force vector will be presented.

4.7.3.1.a Preparation of Full Global Stiffness Matrix

The following variables will be employed to explain the algorithm of the construction of the global stiffness matrix:

i,j : The coordinates of the entries of the element stiffness matrix $\underline{\underline{M}}_e$

m : The element number

The algorithm can be shown by the following equations:

$i = 1, N_v$

$j = 1, N_v$

$\underline{\underline{M}}(i,j) = 0.$

$m = 1,2,\ldots\ldots,N_e$ (number of elements)

$i = 1,..,6$

$j = 1,..,6$

$$\underline{\underline{M}}\Big[i+3(m-1);j+3(m-1)\Big] = \underline{\underline{M}}\Big[i+3(m-1);j+3(m-1)\Big] + \underline{\underline{M}}_{em}(i,j) \qquad (4.52)$$

Upon application of the preceding algorithm the large square matrix is obtained which is depicted in Fig.3 .

TABLE 4.2

THE ENTRIES OF THE ELEMENT STIFFNESS MATRIX
BY THE
ANALYTICAL INTEGRATION APPROACH

i	j	q_1	q_2	q_3	q_4
1	1		-10 / 7L	- 1 / 2	181L / 462
	2	- 9 / 7L	-17 / 14	11L / 84	$311L^2$/ 4620
	3	- 8 / 7	- L / 84	L^2/ 84	$281L^3$/55440
	4		10 / 7L	1 / 2	25L / 231
	5	9 / 7L	- 3 / 14	-11L / 84	$-151L^2$/ 4620
	6	- 1 / 7	L / 84	L^2/ 84	$181L^3$/55440
2	1	9 / 7L	- 3 / 14	-11L / 84	$311L^2$/ 4620
	2	1 / 2	- 8L / 35		$52L^3$/ 3465
	3	- 9L /140	- L^2/ 60	L^3/1008	$23L^4$/18480
	4	- 9 / 7L	3 / 14	11L / 84	$151L^2$/ 4620
	5	11 / 14	L / 70	$-13L^2$/ 420	- $19L^3$/ 1980
	6	-11L /140	- L^2/210	$13L^3$/5040	$13L^4$/13860
3	1	1 / 7	- L / 84	- L^2/ 84	$281L^3$/55440
	2	9L /140	- L^2/ 60	- L^3/1008	23L /18480
	3		L^3/630		L^5/ 9240
	4	- 1 / 7	L / 84	L^2/ 84	$181L^3$/55440
	5	11L /140	L^2/210	$-13L^3$/5040	- 13L /13860
	6	- L^2/140	- L^3/1260	L^4/5040	L^5/11088
4	1		10 / 7L	- 1 / 2	25L / 231
	2	9 / 7L	3 / 14	-11L / 84	$151L^2$/ 4620
	3	1 / 7	L / 84	– L^2/ 84	$181L^3$/55440
	4		-10 / 7L	1 / 2	181L / 462
	5	- 9 / 7L	17 / 14	11L / 84	$-311L^2$/ 4620
	6	8 / 7	- L / 84	- L^2/ 84	$281L^3$/55440
5	1	- 9 / 7L	- 3 / 14	11L / 84	$151L^2$/ 4620
	2	-11 / 14	L / 70	$13L^2$/ 420	- $19L^3$/ 1980
	3	-11L /140	L^2/210	$13L^3$/5040	- $13L^4$/13860
	4	9 / 7L	3 / 14	-11L / 84	$-311L^2$/ 4620
	5	- 1 / 2	- 8L / 35		$52L^3$/ 3465
	6	- 9 /140	L / 60	L^3/1008	- $23L^4$/18480

TABLE 4.2 -- Continued

6	1	1 / 7	L / 84	- L^2/ 84	181L^3/55440
	2	11L /140	- L^2/210	-13L^3/5040	13L^4/13860
	3	L^2/140	- L^3/1260	- L^4/5040	L^5/11088
	4	- 1 / 7	- L / 84	L^2/ 84	281L^3/55440
	5	9L /140	L^2/ 60	- L^3/1008	- 23L^4/18480
	6		- L^3/630		L^5/ 9240

4.7.3.1.b Preparation of Banded Global Stiffness Matrix

Eqs.(4.37.1,2) are employed to convert the full global stiffness matrix to banded global stiffness matrix which requires less computer memory. As explained in section 4.6.3 the memory requirement increases linearly with increasing number of elements.

4.7.3.2 Preparation of Global Force Vector

Similar to the preparation of the full global stiffness matrix in subsection 4.7.3.1.a the global force vector will be loaded.

m = 1,2,..., number of elements

i = 1,..,6

$$\underline{\delta f}^*\{i+3(m-1)\} = \underline{\delta f}^*\{i+3(m-1)\} + \underline{\delta f}_{em}(i) \qquad (4.53)$$

4.7.4 Application of the Boundary Conditions

The boundary conditions (4.24.1-3) are enforced at this stage. The following entries of the global force vector are changed as follows:

$$\delta f^*(1) = 0. \qquad (4.54.1)$$

$$\delta f^*(2) = 0. \qquad (4.54.2)$$

$$\delta f^*(N_v-1) = 0. \qquad (4.54.3)$$

Furthermore the banded global stiffness matrix will be modified as follows:

$$\underline{\underline{M}}(1,\ 6) = 1. \quad \underline{\underline{M}}(2,\ 6) = 1. \quad \underline{\underline{M}}(N_v-1,\ 6) = 1. \tag{4.55.1,2,3}$$

$$\underline{\underline{M}}(N_v,5) = 0. \quad \underline{\underline{M}}(N_v-1,\ 7) = 0. \tag{4.55.4,5}$$

If the second index of any matrix entry equals six, the entry mentioned corresponds to the diagonal element of the full global stiffness matrix.

The consecutive three algorithms enforce the required entries of the banded global stiffness matrix to be equal zero.

$$\begin{aligned} j &= 2,..,6 \\ \underline{\underline{M}}(j,7-j) &= 0. \\ \underline{\underline{M}}(1,5+j) &= 0. \end{aligned} \tag{4.56}$$

$$\begin{aligned} j &= 3,..,6 \\ \underline{\underline{M}}(j,8-j) &= 0. \\ \underline{\underline{M}}(2,4+j) &= 0. \end{aligned} \tag{4.57}$$

$$\begin{aligned} j &= N_v-5\ ,\ N_v-2 \\ \underline{\underline{M}}(j,N_v+5-j) &= 0. \\ \underline{\underline{M}}(N_v-1,7+j-N_v\ ) &= 0. \end{aligned} \tag{4.58}$$

The system of linear equations are thus ready to be solved.

4.7.5 Solution of Banded Matrix

During the solution of the banded matrix, global stiffness matrix is decomposed by rowwise permutation of itself and the system of equations are solved. The algorithm employed uses equilibration and partial pivoting for the solution. The details of the algorithm can be found in the paper of Martin and Wilkinson(32).

4.7.6 Check for Convergence

Upon application of the algorithm, which is introduced in the previous section, the solution of the linear system of equations(4.35) is obtained. At this step Newton's iterative method is employed and higher order iterates are calculated by the use of Eqs.(4.21.1-4). Any algorithm which incorporates an iterative method of solution has a certain criterion for convergence. The present analysis uses the following convergence criterion:

$$|\delta f''_w| \leqq 0.00005 \qquad (4.59)$$

i.e. the change of the second derivative of the transformed stream function at the wall.

This choice has both numerical and physical reasons:

(i) Numerically, $\delta f''_w$ displays the most rapid change between iterations

(ii) Physically, it corresponds to the variation of the shear stress at the wall

As soon as inequality(4.59) is satisfied f, f' and f" at the nodes of the solution domain are calculated. Depending on the type of the problem the computer code either stops (similar boundary layers) or proceeds to the next ξ-station (non-similar boundary layers).

4.7.7 Test Cases to Be Solved

The solution method developed in this study will handle the following test cases:

a- Laminar boundary layers with constant and variable pressure gradient parameters

i- Howarth's flow

ii- Flow past a Circular Cylinder

b- Turbulent boundary layers

The details of the test cases to be solved can be found in Appendix G.

4.7.8 Augmentation of Solution Domain for Turbulent Boundary Layer Flows

Having obtained β and ξ and consequently α, there is a need to check $|f''|$ at the prescribed value of η_∞. For almost all laminar boundary layers, the upper limit of the solution domain η_∞, may be taken constant. However, in the case of turbulent boundary layers η_∞ must be increased downstream due to the rapid growth of the boundary layer. If at a station n-1 the $|f''_{\eta\infty}|$ exceeds 0.0001, i.e.

$$|f''_{\eta\infty}| > 0.0001 \tag{4.60}$$

then the solution domain is augmented at the next station by adding extra elements and nodes. The additional nodes at stations n-1 and n are assigned the following initial values.

$$f_{j+1} = f_j + \Delta\eta_j \tag{4.61.1}$$

$$f_{\substack{j+1\\n-1}} = f_{\substack{j\\n-1}} + \Delta\eta_j \tag{4.61.2}$$

$$f'_{j+1} = 1.0 \tag{4.61.3}$$

$$f'_{\substack{j+1\\n-1}} = 1.0 \tag{4.61.4}$$

$$f''_{j+1} = 0. \tag{4.61.5}$$

Consequently, the boundary layer at the next station is calculated with increased number of nodes.

4.7.9 Eddy Viscosity Model Used

As depicted in Fig.2, the solution of every ξ-station is obtained separately. If the point of transition from laminar to turbulent flow is passed, the terms of eddy viscosity also have to be employed in order to describe the characteristics of the turbulence. For this purpose Eqs.(4.16-18) are used.

Finally, the program is terminated on two conditions. First, the whole domain has been covered, i.e. the solution process is complete. Secondly, at a -station the maximum number of iterations has been exceeded, i.e. convergence can not be obtained within the specified iteration limits.

CHAPTER V

DESCRIPTION OF THE COMPUTER CODE

The present chapter deals with the considerations leading to the computerization of the algorithm presented in Chapter IV.

The computer code developed in this study has three versions. The first version RBLGAU developed constructs the element matrices by analytical integration and attempts to solve the resulting global equations in the full matrix form by the Gaussian Elimination Method.

The second version, RBLELE, obtains the element matrices similarly and solves the resulting global equations in banded form by the Banded Matrix Solver.

The third version RBLAYER developed constructs the element matrices by numerical integration and obtains the solution of the global system of equations like the version RBLELE. These versions are discussed in the following pages in detail.

5.1 Version RBLGAU

5.1.1 Input Description

The following variables have to be input:

1- η_∞, specified upper limit of the solution domain which is greater than calculated boundary layer thickness, i.e. $\eta_\infty > \eta_\delta$

2- k, geometric grid parameter (see section 5.1.2)

3- Print option parameter (see section 5.1.3)

4- Index of the transition station where the flow becomes turbulent

5- Total number of x-stations excluding the initial station

6- Initial velocity profile flag, which gives the type of the initial velocity profile (Eq.(4.38.1), (4.39.1) or (4.40.1)) to be used

7- Pressure gradient flag, which indicates the type of flow which is presented in section 4.7.7 to be treated

8- ρ, fluid density in kg/m^3

9- μ, dynamic viscosity of the fluid in kg/ms

10- U_∞, free stream velocity of the fluid in m/s

11- h_1, initial grid length (see section 5.1.2)

12- Type of grid arrangement (constant or variable mesh)

13- β,pressure gradient distribution in the streamwise direction is input from an inviscid flow solution

14- The coordinates of the x-stations except for similar flows and flows for which the potential flow solution is to be specified

15- The coordinates of the x-stations with their respective potential velocity values

16- Maximum number of allowed iterations for the Finite Element solution at the x-stations.

5.1.2 Grid Arrangement of the Solution Domain

In order to obtain the solution of a particular problem the solution domain has to be discretized. As presented in Cebeci and Smith(21) the laminar boundary layers pose no difficulty of discretization, since their upper limit of solution domain can be taken as constant. Furthermore, since the velocity gradient is not too steep in laminar flows, mesh size is also constant, i.e. consecutive elements in the y (or η) direction are of the same thickness.

The grid length, i.e. the difference between two consecutive x (or ξ) stations, may also be taken constant or, if necessary, variable. Especially, if a region is suspected of containing the separation point, the grid length has to be taken decreasing towards the suspected separation point.

The skin friction encountered in turbulent flows is larger than in laminar flows and velocity gradient $\partial u/\partial y$ at the wall is greater in the former, i.e. the thickness of the gridshave to decrease towards the wall. If, k is the ratio of the thickness of the successive grids, and h_1, the thickness of the first $\Delta\eta$ step (first grid), the total number of nodes, J, throughout the boundary layer can be calculated as follows:

$$J = \ln\{1+(k-1)(\eta_\infty/h_1)\}/\ln(k) \tag{5.1}$$

The thickness of the j-th element in η-direction is

$$\eta_j = h_1(k^j - 1)/(k - 1) \quad , k>1 \ , j<J \tag{5.2}$$

5.1.3 Output Description

There are various output options of the computer code. Item 3 in sub-section 5.1.1 indicates the desired output option. Table 5.1 describes the option and the corresponding output data printed.

5.1.4 Flow of the Program

The program executes the solution scheme presented in Chapter IV. Functional relationship between the subroutines and the main program is depicted in Fig.4 .

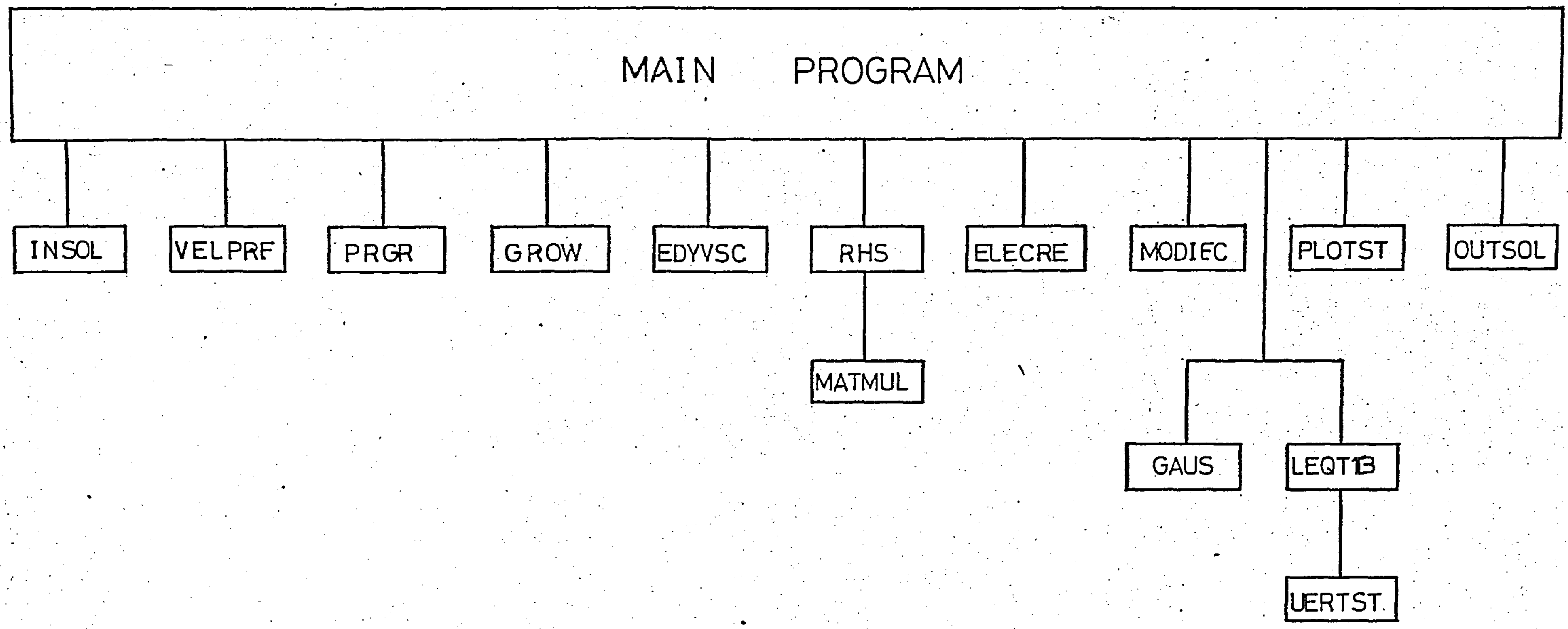


FIGURE 4.-- Functional relationship of the main program with the subroutines

TABLE 5.1

DESCRIPTION OF OUTPUT

Desired Output	Option
Initial velocity profile (with f, f")...................	0 1 2 3
All element stiffness matrices...........................	1
Global stiffness matrix..................................	1
Modified global stiffness matrix.........................	1
Global stiffness matrix after Gaussian Elimination......	1 2
Solution vector..	1
Third entry of solution vector (f" at the wall)........	0 1 2 3
Intermediate iterative solutions.........................	1 2 3
Final solution of a -station............................	0 1 2 3

Subroutine INSOL inputs the variables listed in Section 5.1.1. This subroutine calculates the η and ξ coordinates of the boundary layer and prepares the standard matrix of Gaussian Integration for future reference.

Subroutine VELPRF prepares the initial velocity profiles (item 6). Section 4.7.1 contains the equations employed by this subroutine.

The main loop of the program prepares the element matrices, assembles the global stiffness matrix, and the global force vector.

Subroutine ELECRE calculates the entries of the element stiffness matrix presented in Table 4.2 by the analytical integration approach.

Subroutine RHS calculates the right-hand-side vector (the element force vector) by Gaussian Numerical Integration. The element stiffness matrices and the element force vectors are then assembled into the global stiffness matrix and global force vector, respectively, by the combination of Eqs.(4.52,53).

Subroutine MODIFC modifies the global stiffness matrix and global force vector by the application of the boundary conditions (see Eqs.(4.24.1-3)). For this purpose, the algorithms presented by Eqs.(4.55-58) are employed.

Subroutine GAUS solves the system of equations in the matrix form by the method of Gaussian Elimination and back substitution. Depending on the convergence criterion, Eq.(4.62), the main loop is repeated. Each iteration updates the vectors $\underline{f}$, $\underline{f}'$ and $\underline{f}''$ using Eqs.(4.21.1-4). If the convergence criterion for a -station is satisfied then the vectors $\underline{f}$ and f' are redefined as $\underline{f}_{n-1}$ and f'_{n-1}, and the program proceeds to the next station.

Subroutine PLOTST plots the results obtained, if required.

The iteration procedure is repeated at the next ξ-station, if any. Subroutine PRGR calculates the ξ and β in the streamwise direction using equations presented in Appendix G.

For the case of turbulent flow ξ-station of transition from laminar to turbulent flow has to be input (5.1.1). Subroutine EDYVSC calculates the non-dimensional eddy viscosity term ε^+ and itsderivative $\varepsilon^{+'}$. This subroutine uses Eqs.(4.16-18) to calculate the inner and outer eddy viscosity coefficients of the turbulent boundary layer. The inner and outer regions are matched by imposing the requirement of continuity. The inner eddy viscosity expression, Eq.(4.16), is used up to the node where $\varepsilon_i^+ > \varepsilon_o^+$ after which the outer eddy viscosity expression takes over. The derivatives of the eddy viscosity terms are obtained by the use of the three point numerical differentiation. (see Appendix H)

If inequality (4.63) is satisfied, subroutine GROW employs Eqs.(4.64.1-5) to augment the upper limit of the solution domain.

Subroutine OUTSOL prints either the intermediate iterative solutions or the final solution of a ξ-station, depending on the output option. (see Table 5.1)

Detailed description of the input of the computer code can be found in Appendix

5.2 Program RBLELE

The version RBLELE has only slight vaiations from the version RBLGAU. In this section, only the deviations will be explained. Instead of using the full global stiffness matrix, RBLELE employs the banded matrix formulation presented in section 4.6.3 . The solution of the banded matrix force vector system is obtained by subroutine LEQT1B which uses the algorithm briefly explained in section 4.7.5 and employs a further subroutine UERTST to check the matrix. The rest of the program is identical with RBLGAU.

5.3 Program RBLAYER

This program is the final version of the computer code. It is completely identical with RBLELE except for the calculation of the element stiffness matrices. RBLAYER uses the numerical integration approach (see section 4.7.2.a). TThis version formulates the element stiffness matrix by Eqs.(4.41-46) and integrates them by the Gaussian Numerical Integration. The listing of the program (version RBLELE and RBLAYER) can be found in Appendix I.

CHAPTER VI

RESULTS AND DISCUSSIONS

6.1 Introduction

The final test of a numerical method is the comparison of calculated results with exact solutions and with experiments. For laminar flows, there are many analytically obtained solutions, as well as solutions obtained by well tested and well established numerical methods. On the other hand, there are no exact solutions for turbulent flows, and all one can do is to compare the calculated results with experiments.

In this chapter the numerical accuracy of the present study is tested by applying the present method to a number of laminar and turbulent boundary layer flows and comparing the results with solutions obtained either analytically or by other numerical methods. All calculations reported here have been performed on CDC CYBER 815 in itssingle precision version.

6.2 Mathematical Test of the Computer Code

In order to check the basic elements of the computer code a special case of Eq.(4.22) with q_i=1 is considered.

$$y''' + y'' + y' + y + 1 = 0 \tag{6.1}$$

It can easily be seen that the exact solution is given by

$$y = \sin x + \cos x + 1 \tag{6.2}$$

subject to the boundary conditions

$$y(0) = 0 \qquad\qquad y'(0) = 1 \qquad\qquad (6.3.1,2)$$

$$y'(\eta_\infty) = \cos(\eta_\infty) - \sin(\eta_\infty) \qquad\qquad (6.3.3)$$

Comparison of the computer results with the exact solution is shown in Table 6.1 for seven different mesh structures. The first six cases involveuniform mesh with constant element length h_1. The seventh case makes use of a variable mesh constructed in accordance with Eq.(5.1).

TABLE 6.1

ACCURACY STUDY FOR EQ.(6.1)

Case	h_1	η_∞	N_e	$y''(0)$	per cent error
1	0.5	3.0	6	-0.999899	0.0101
2	0.2	3.0	15	-0.999999	0.0001
3	2.0	8.0	4	-0.972463	2.7537
4	1.0	8.0	8	-0.999151	0.0849
5	0.5	8.0	16	-0.999956	0.0044
6	0.25	8.0	32	-0.999997	0.0003
7	var. mesh	7.365	19	-0.999805	0.0195

a- Effect of increasing the number of elements for a given η_∞:

In cases 1 and 2, when the number of elements is increased 2.5 times, the error is reduced 100 times.

Each time the element number is doubled, from case 3 to 4, 5 and 6, the error is reduced by 32.4 times, 19.3 times, and 14.7 times, respectively.

It is concluded that for a given η_∞, an increase in the number of elements drastically reduces the error.

b- Effect of having different η_∞ for a given element number:

With reference to cases 2 and 5, although the latter case contains one more element than the former, its error is 44 times greater, since the corresponding η_∞ is 2.667 times greater.

It is concluded that the larger the domain the more the probability of error despite the increase in the number of elements.

c- Effect of having a variable mesh:

With reference to cases 5 and 7, although the latter uses 1.19 times more elements and 1.09 times less η_∞, the corresponding error is 4.43 times greater than the former case.

One tends to conclude that the use of variable mesh reduces the accuracy of the solution despite the increase in the number of elements and the decrease in η_∞. On the other hand, it will be necessary for the analysis of the turbulent boundary layers.

6.3 Application to Similar Boundary Layer Flows

For all similar flows Eq.(4.19) is independent of ξ, furthermore, $\varepsilon^+=0$. Therefore Eq.(4.19) is reduced to

$$f''' + ff'' + \beta\,\{1- (f')^2\} = 0 \tag{6.4}$$

and the coefficients of Eq.(4.22) become

$$q_1 = 1 \qquad\qquad q_2 = f \tag{6.5.1,2}$$

$$q_3 = -\,2\beta f' \qquad\qquad q_4 = f'' \tag{6.5.3,4}$$

$$q_5 = f''' + ff'' - \beta(f')^2 + \beta \tag{6.5.5}$$

The analytical solutions to Eq.(6.4) have been extensively studied and are known as the Falkner-Skan family. It can be shown that solutions $\beta>0$, $\beta=0$ and $\beta<0$ correspond, respectively, to accelerating, constant velocity and decelerating flows. It can also be shown that solutions exist for all values of β in the range $-0.19884 \leq \beta < 2$. Outside this range, separation occurs.

6.3.1 Effect of Initial Velocity Profile:

To see the effect of the initial guess, the linear velocity profile, Eq.(4.38.1), and the Pohlhausen type velocity profile, Eq.(4.40.1), have been considered in connection with the boundary layer with zero pressure gradient.

Program RBLAYER has obtained the exact solution of the problem in five iterations using the linear velocity profile, $\Delta\eta = 0.2$, $\eta_\infty = 6.0$, i.e. 30 elements. The convergence has been less than quadratic, i.e. the rate of convergence between iterations is less than two (1.78-1.86). (See Fig.5)

Program RBLAYER has obtained the exact solution of the problem in four iterations using the Pohlhausen type velocity profile, $\Delta\eta = 0.25$, $\eta_\infty = 6.0$, i.e. 24 elements. The convergence has been almost quadratic (1.87-2.00).(See Fig.6)

This application shows that the use of the Pohlhausen type velocity profile is more advantageous because it reduces the number of iterations.

On the basis of this conclusion the remaining results reported in this thesis have been calculated by using the Pohlhausen type velocity profile.

6.3.2 Accuracy of the Method for Different β-Values:

The test of the computer code under different β-values are outlined

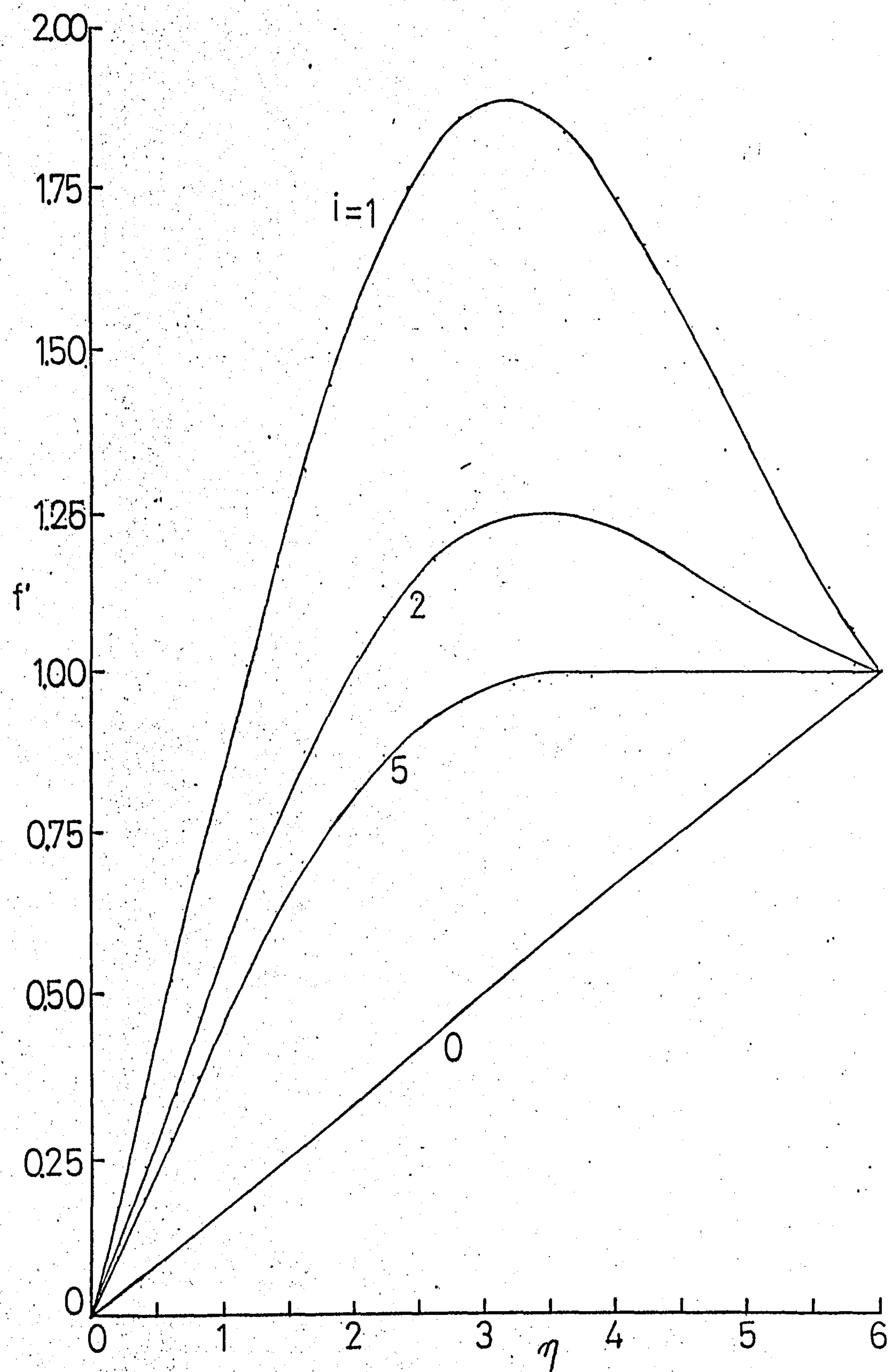
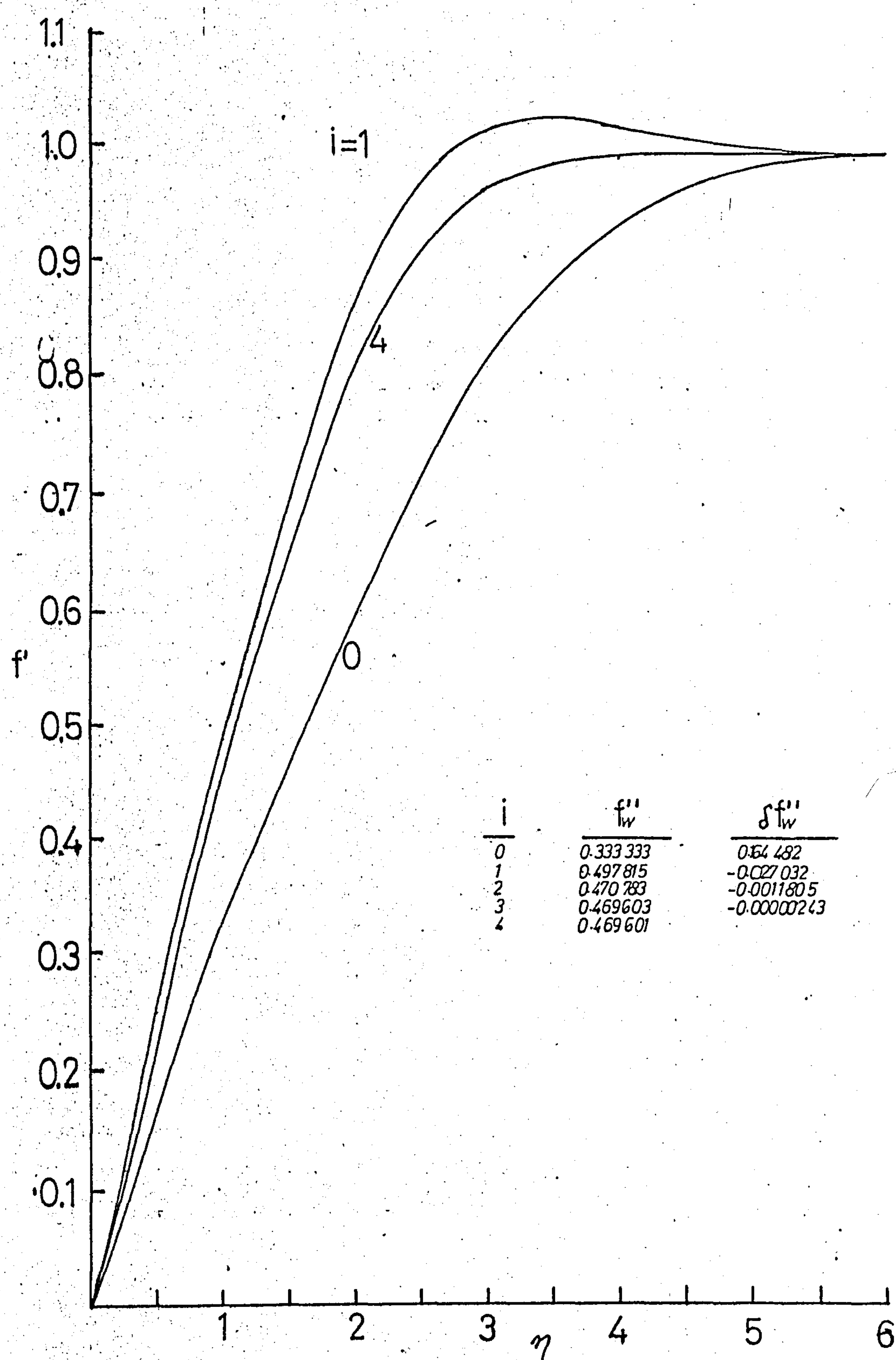


FIGURE 5

Effect of Linear Velocity Profile

Program: RBLAYER

$\Delta\eta = 0.2$ $\eta_\infty = 6.0$ $N_e = 30$ $\beta = 0.$



i	f''_w	$\delta f''_w$
0	0.333 333	0.164 482
1	0.497 815	-0.027 032
2	0.470 783	-0.0011805
3	0.469603	-0.00000243
4	0.469 601	

FIGURE 6

Effect of Pohlhausen Type Velocity Profile

Program: RBLAYER

$\Delta\eta = 0.25 \quad \eta_\infty = 6.0 \quad N_e = 24 \quad \beta = 0.$

in Table 6.2 which shows f''_w calculated by the exact solution, the Cebeci and Smith method and the present method (RBLGAU version).

TABLE 6.2

ACCURACY STUDY FOR FALKNER-SKAN FLOWS

Values of f''_w

β	Exact	Present analysis	Cebeci Smith	per cent error of pr.anal.	$\Delta\eta$	η_∞
1.0	1.232588	1.234961	1.326940	0.1925	1.0	8.0
		1.232654	1.257044	0.0054	0.5	
0.0	0.469600	0.470736	0.506103	0.2419	1.0	
		0.469596	0.478914	0.0009	0.5	
-0.1	0.319270	0.320272	0.355730	0.3138	1.0	
		0.319281	0.328762	0.0034	0.5	
-0.19884	0.0	0.013431	-0.113413	-	1.0	
		0.081656	-0.058364	-	0.5	

As concluded earlier more elements lead to more accurate results. The results are very satisfactory for β = 1.0, 0., -0.1 except at the point of separation it was impossible to predict a correct velocity profile.

6.3.2.1 Accelerated Flows ($\beta > 0$)

These flows have a positive pressure gradient parameter. Recalling Eq.(4.11) one can conclude that they are accelerated flows. In order to show their effect the case $\beta = 1$ is analyzed. It is also known as stagnation point flow. Exact solution of this flow has been obtained and presented in Rosenhead et al.(33), p.232. Table 6.3 compares the results of the programs RBLELE and RBLAYER with the exact solution. It can be seen that RBLAYER has obtained the exact solution. On the other hand, RBLELE has at least five-digit accuracy.

TABLE 6.3

ACCURACY STUDY FOR ACCELERATED FLOWS

$\beta = 1.0$, $N_e = 24$, $\eta_\infty = 6.0$

Values of f'

η	Exact	RBLAYER	RBLELE
.0	.0	.0	.0
.5	.494649	.494649	.494650
1.0	.777865	.777865	.777866
1.5	.916168	.916168	.916169
2.0	.973217	.973217	.973217
3.0	.998424	.998424	.998424
4.0	.999958	.999958	.999958
5.0	.9999995	1.000000	.999999

As mentioned in section 6.3, the extreme case for an accelerated similar laminar boundary layer is β=2. Table 6.4 compares the results in Rosenhead et al.(33) with the results of programs RBLAYER and RBLELE.

6.3.2.2 Flat Plate Flow (β=0.)

This case is also known as Blasius problem(1). It is solved analytically by series expansions. Its practical importance has been the reason for extensive study. Table 6.5 compares the present study with the exact result of Blasius(1908), Töpfer(1912),Goldstein(1930) and Howarth(1938).

6.3.2.3 Decelerated Flows (β<0)

These flows are important since they approach the region of separation. Analytically, the value of separation of β equals -0.19884 . In the present study two cases of decelerated flows are considered

(i) β = -0.1

(ii) β = -0.19884

Table 6.6 compares the results of the program RBLAYER with the exact solution for case (ii). On the other hand, RBLELE has turned out to

TABLE 6.4

ACCURACY STUDY FOR ACCELERATED FLOWS

$\beta = 2.0$, $N_e = 24$, $\eta_\infty = 6.0$

Values of f'

η	Exact	RBLAYER	RBLELE
.0	.0	.0	.0
.5	.610	.610	.610
1.0	.872	.872	.872
2.0	.991	.991	.991
4.0	1.000	1.000	1.000

TABLE 6.5

ACCURACY STUDY FOR FLAT PLATE FLOW

Values of f' β=0. , η_∞=6.0

η	Exact	RBLAYER	RBLELE
.0	.0	.0	.0
.5	.234227	.234228	.234225
1.0	.460632	.460633	.460629
1.5	.661473	.661475	.661470
2.0	.816694	.816695	.816693
3.0	.969054	.969055	.969056
4.0	.997770	.997771	.997771
5.0	.999936	.999937	.999937
6.0	.999999	1.000000	1.000000
		N_e=24	N_e=12

be unstable and has not converged.

It is peculiar to note that the six-element solution is better than the more element solutions. It is because of the convergence criterion ε which is 0.00005 . A more tight criterion can yield better results. Comparing the results of the case N_e=12 and case N_e=24 shows that there is no appreciable difference between them. Consequently, the

TABLE 6.6

ACCURACY STUDY FOR DECELERATED FLOWS

Values of f' β=-0.19884 , η_∞=6.0

		RBLAYER		
η	Exact	Ne=6	Ne=12	Ne=24
.0	.0	.0	.0	.0
.5	.025	-	.029509	.029510
1.0	.099	.107154	.108374	.108375
2.0	.380	.394379	.396463	.396465
4.0	.940	.944595	.945236	.945237
6.0	1.000	1.000000	1.000000	1.000000

program runs at the separation point have to be performed with less number of elements and more tight convergence criterion.

6.3.2.4 Performance of the Present Study Regarding Similar Flows:

In order to analyze the performance of the present study, the following criteria will be considered:

(i) Accuracy

(ii) Order of Convergence

(iii) CPU-Time

(iv) Number of Iterations

Table 6.7 compares the results of the present study with Cebeci and Smith(21) under similar conditions. This table contains information about the last two versions of the computer code. (Table 6.2 compares the similar conditions with program RBLGAU).

(i) As can be noted, both programs RBLAYER and RBLELE yield more accurate results than Cebeci and Smith. The latter's results can be improved by Richardson extrapolation. On the other hand, both of the versions of the present study can produce more improved results with N_e=24 (elements). Consequently,

TABLE 6.7

ACCURACY STUDY FOR FALKNER-SKAN FLOWS

WITH DIFFERENT $\Delta\eta$ SPACINGS

Values of f''_w , η_∞=6.0

β	Exact	$\Delta\eta$	Cebeci Smith	RBLAYER	RBLELE
1.0	1.232588	1.0	1.326940	1.234962	1.234963
		0.5	1.257044	1.232736	1.232732
.0	.469600	1.0	.506065	.470746	.470778
		0.5	.478914	.469603	.469597
-0.1	.319270	1.0	.355731	.320292	.320256
		0.5	.328762	.319283	.319288
-0.19884	.0	1.0	.123289	.007171	-
		0.5	.060844	.009325	-

the accuracy of the present study is satisfactory.

(ii) Fig.7 depicts the convergence of the program RBLAYER. Except for the case of separation the convergence is quadratic. Furthermore, the higher the β-value, the better is the convergence. Fig.8 depicts the convergence of the program RBLELE. Its convergence is quadratic for all cases but for the case of separation. In this particular case the stability can not be observed. Consequently, the order of convergence of the present study is quadratic in general (in most of the similar cases).

(iii-iv) Program RBLAYER has a constant ratio of CPU time/element≈3 sec. except for the case of separation. For this particular case the ratio is higher because of the higher number of iterations. Table 6.8 compares the CPU-times of various performances of RBLAYER for the case of separation. The CPU-time/element≈4.67 sec. for six iterations. On the other hand, program RBLELE has the ratio CPU-time/element≈1 sec. For 24 elements the ratio decreases to 0.8 because of the less number of iterations.

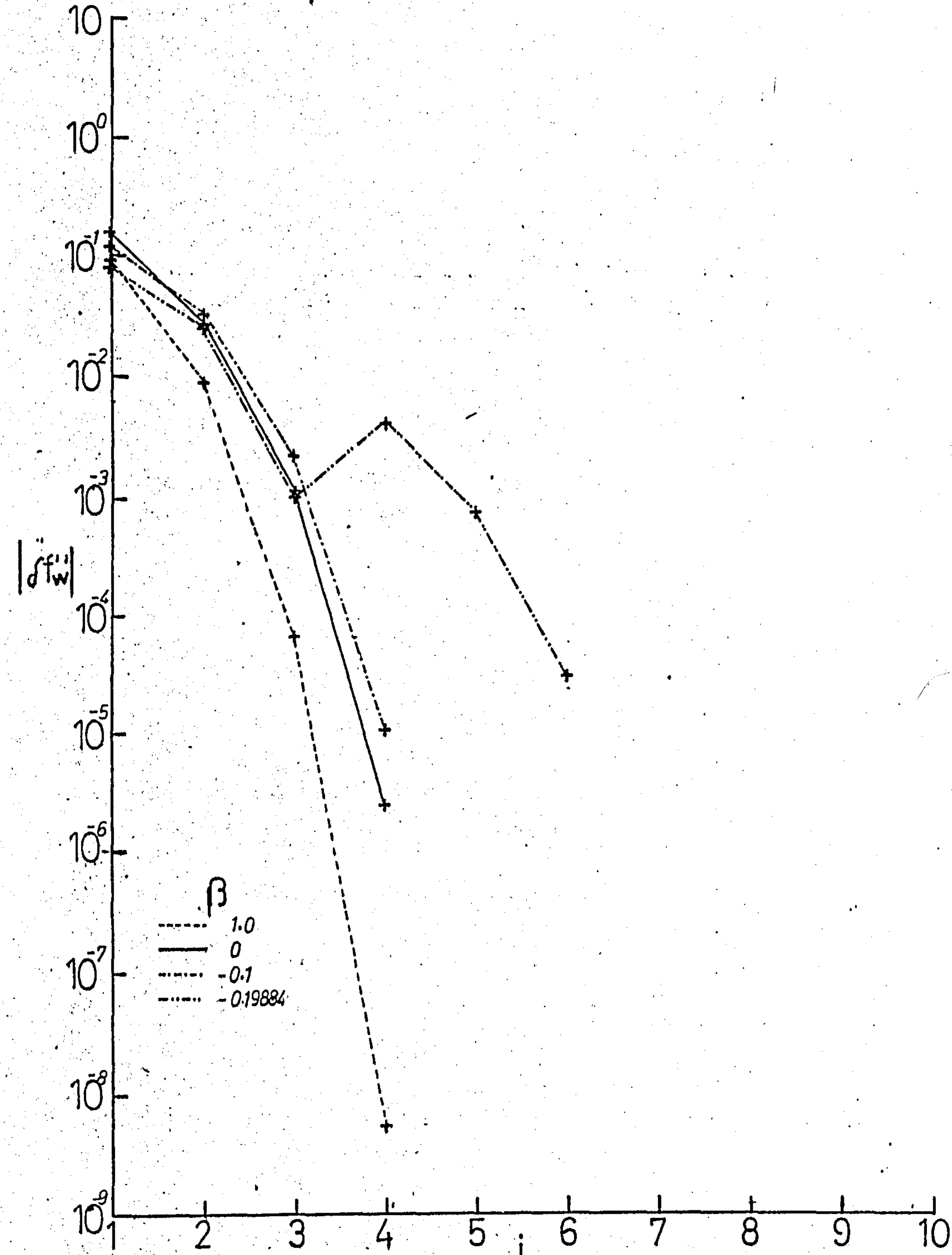


FIGURE 7

Rate of Convergence of f''_w for Similar Flows

Program: RBLAYER

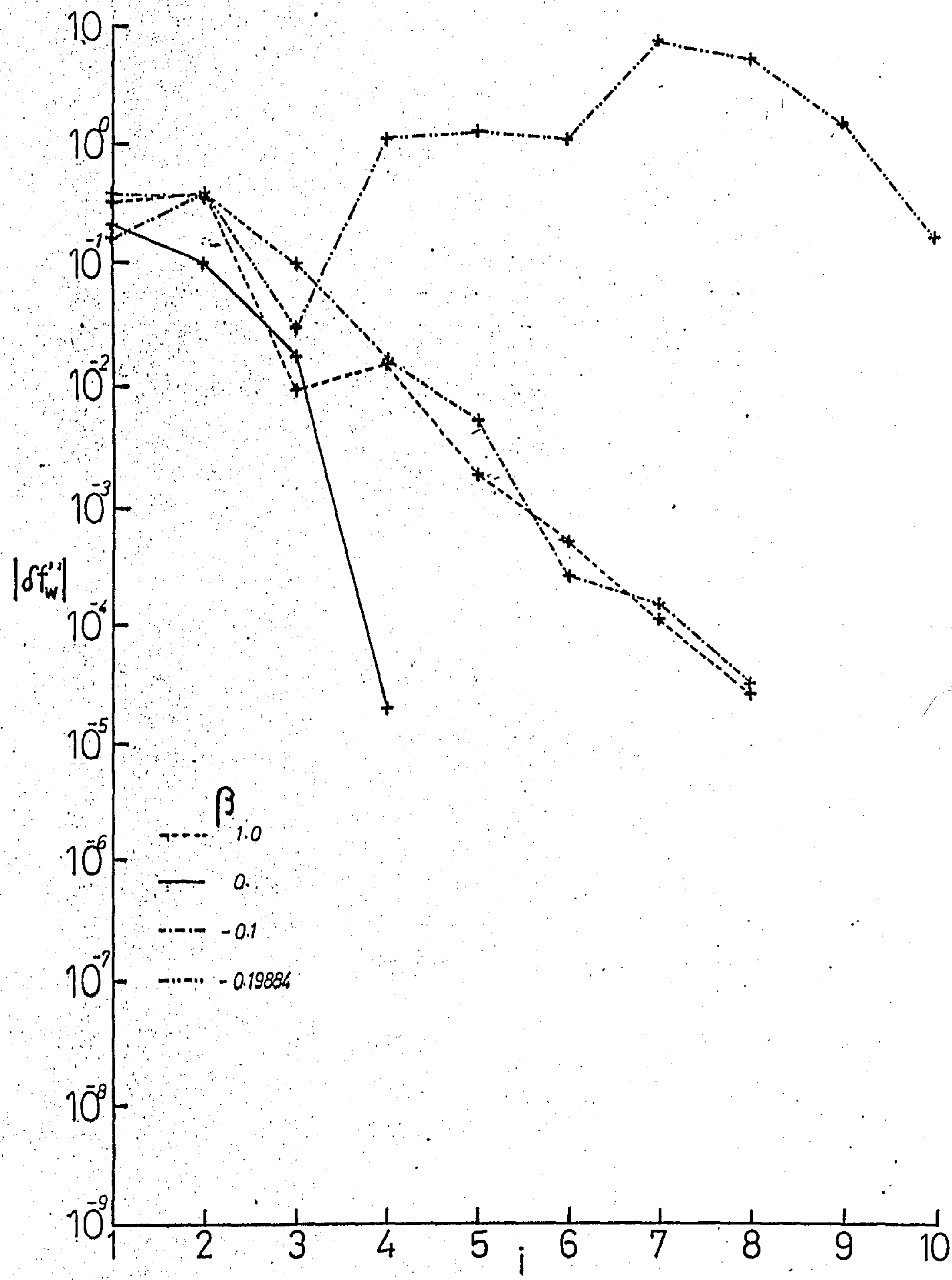


FIGURE 8

Rate of Convergence of f''_w for Similar Flows

Program: RBLELE

TABLE 6.8

PERFORMANCE OF PROGRAM RBLAYER

β =-0.19884 , η_∞=6.0

N_e	Iteration	CPU-time(sec)
6	7	32.845
12	6	55.916
24	6	112.178

6.4 Application to Non-Similar Boundary Layer Flows

From a practical point of view, the non-similar flows are more important than the similar flows, because similar flows are rare in technical applications. In non-similar flows the dimensionless stream function f depends on both ξ and η.

These flows can be solved by various methods, such as:

(i) Series methods

(ii) Momentum integral methods(MIM)

(iii) Local similarity method

(iv) Numerical-differential methods

For example, the MIM are easy to use, quick to solve, but their accuracy can hardly approach the level of the differential-numerical methods. In the following two sub-sections two different applications to the non-similar boundary layer flows will be presented.

6.4.1 Howarth's Flow:

As explained in Appendix G1 the non-dimensional velocity field is given by

$$\bar{u} = 1 - a\bar{x} \qquad (G.1)$$

The pressure gradient parameter, β, decreases along the $\overline{x}$-axis. Consequently, the flow separates downstream. It is, therefore, important to carefully choose the $\overline{x}$-spacing around the region of the possible separation. In the region mentioned one has to have more $\overline{x}$-stations in order to avoid the loss of accuracy. Program RBLAYER has been run for this flow. The most important variable to be used for comparison is the dimensionless shear parameter (DSP). (See Eq.(G.11)) Fig.9 shows the variation of the DSP with respect to the $\overline{x}$-coordinate. The solid curve represents the exact solution of Howarth(3). The dotted curve depicts the deviation of the present analysis. Table 6.9 compares the numerical values of the exact solution with the results of the present study. It can be observed that the error of the present method increases towards the point of separation.

6.4.2 Flow past a Circular Cylinder

As given in Appendix G2 the velocity field is given by:

$$U_e = 2\, U_\infty \sin\overline{x} \qquad \text{(G.12)}$$

The pressure gradient parameter, β, changes sinusoidally. Behind the top of the cylinder separation occurs. Program RBLAYER has been run for this flow. Fig.10 shows the variation of the DSP with respect to the angular coordinate($\overline{x}$). The solid curve represents the exact solution of Tifford(38). The dotted curve shows the deviation of the present method. Table 6.10 compares the numerical values of the exact solution with the results of the present study. It is again noted that the present method looses its accuracy drastically as the point of separation is approached.

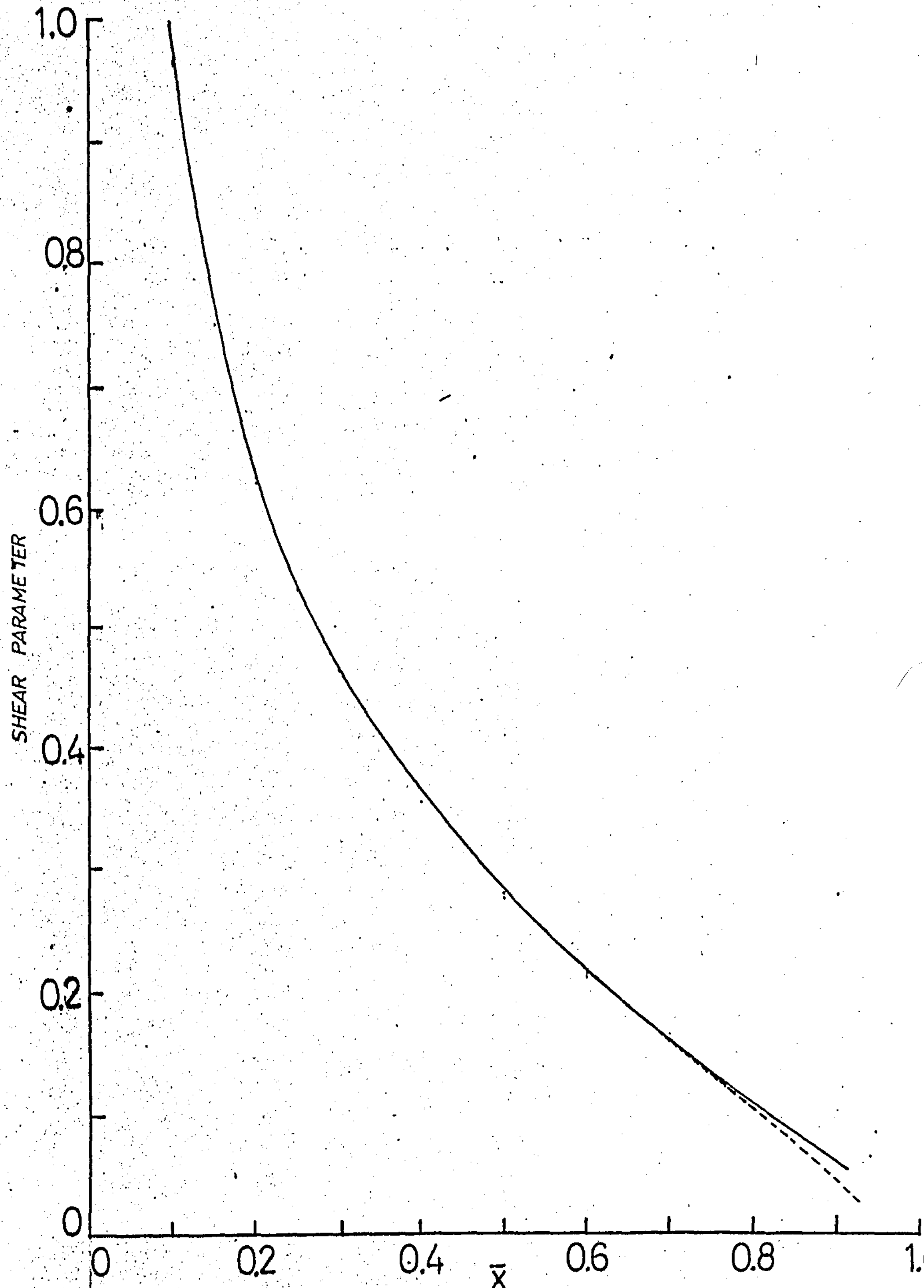


FIGURE 9

Variation of the shear parameter for Howarth's Flow

—— Exact Solution of Howarth(3)

--- Deviation of the Present Method

TABLE 6.9

ACCURACY OF THE PRESENT METHOD FOR HOWARTH'S FLOW

Dimensionless Shear Parameter

$\bar{x}$	Howarth(3)	Pres.Meth.	per cent error
0.1	0.968382	0.965046	0.344
0.2	0.626496	0.623301	0.510
0.3	0.462801	0.459611	0.689
0.4	0.357442	0.354017	0.958
0.5	0.279307	0.275634	1.315
0.6	0.216728	0.212000	2.182
0.7	0.162281	0.156363	3.647
0.8	0.111369	0.103098	7.427
0.9	0.057629	0.044684	22.463
$\bar{x}$-sep.	0.96	0.94	

TABLE 6.10

ACCURACY OF THE PRESENT METHOD FOR THE FLOW PAST A CIRCULAR CYLINDER

Dimensionless Shear Parameter

(x/r)deg.	Tifford(38)	Pres.Meth.	per cent error
15	0.888432	0.888332	0.011
30	1.635088	1.634387	0.043
45	2.118444	2.116264	0.103
60	2.253531	2.250248	0.146
75	1.999958	1.992247	0.386
90	1.355562	1.326423	2.150
100	0.712656	0.593627	16.702
x-sep	108.8°	104°	

6.5 Application to Turbulent Boundary Layer Flows with Zero Pressure Gradient

Practically, turbulent flows are more important than laminar flows because the Reynolds number in technical applications are normally

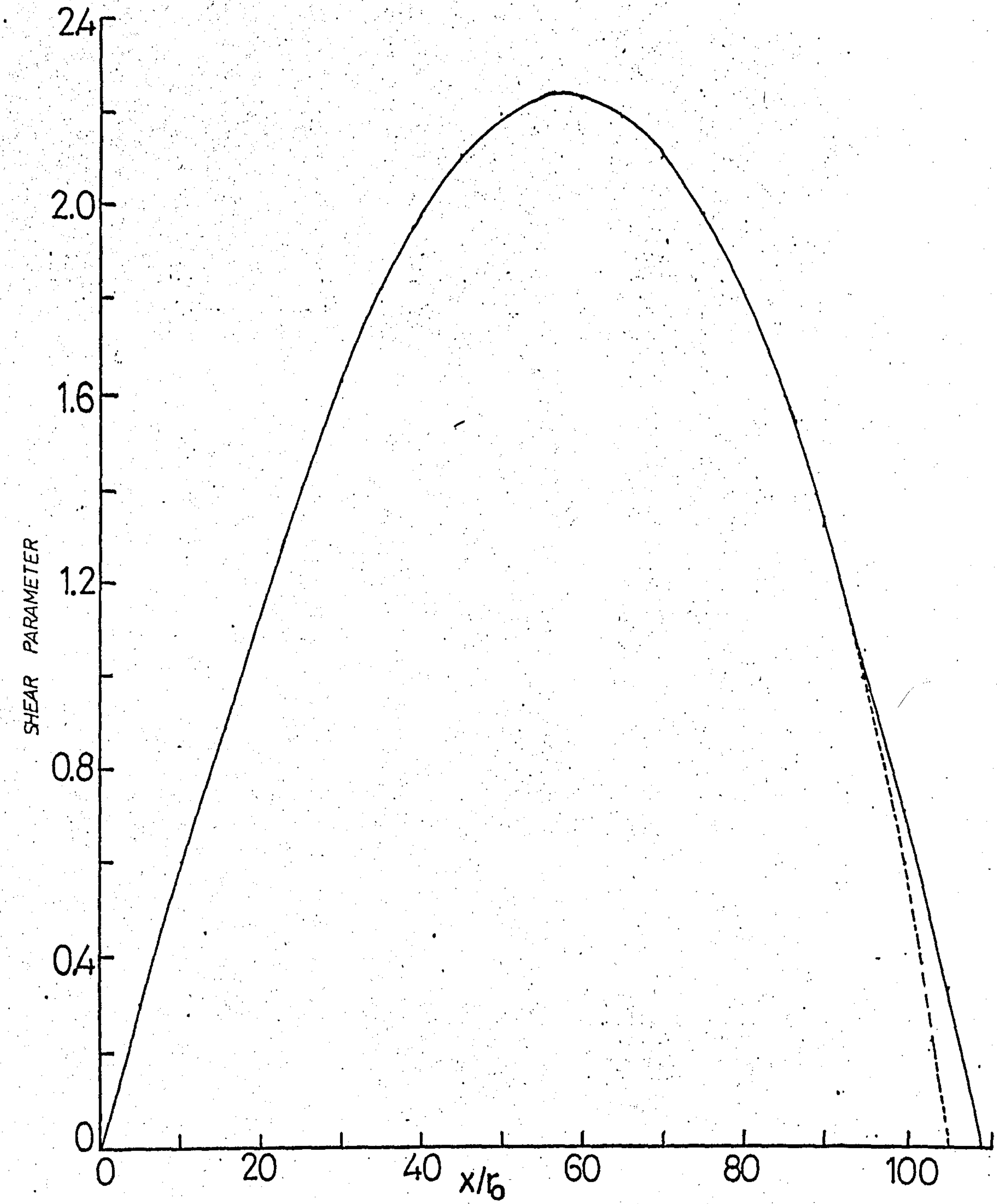


FIGURE 10

Variation of the Shear Parameter for the Flow past a Circular Cylinder

—— Exact Solution (Tifford Series)
--- Deviation of the Present Method

very large and the eddying motion of the real fluids is more dominant than their laminar behaviour.

The last application of the present analysis is the calculation of the turbulent boundary flow over a flat plate, i.e. pressure gradient is zero.

It is assumed that the flow is laminar at the leading edge and then becomes turbulent at the next station. To compare the present results with those of other investigators, the calculations have been performed for a variable grid length along the ξ-direction. The results obtained in the present study are compared in Table 6.11 with the results of Cebeci and Smith(37) and Bismarck-Nasr(34).

The development of the local skin friction coefficient can be found in Appendix G4. Fig.11 shows the variation of this coefficient with respect to Reynolds number.

It is concluded that the results of the present analysis are about six per cent less than the results of Cebeci and Smith(37). The difference may be attributed to two basic reasons. First, the study of Cebeci and Smith used 230 nodes in the boundary layer whereas only 40 nodes were employed in the present study due to the time limitations of the existing computer system. Secondly, the mesh structure of the solution domain of Cebeci and Smith was optimized whereas the present analysis does not posess this capability.

Fig.12 depicts the variation of the boundary layer thickness normalized by the distance with respect to Reynolds number. The solid curve represents the equation deduced from the 1/7 th power velocity distribution law for the smooth flat plate(39):

$$\frac{\delta(x)}{x} = 0.37\ Re^{(-1/5)} \tag{6.6}$$

TABLE 6.11

LOCAL SKIN FRICTION COEFFIENT FOR TURBULENT FLOW OVER A FLAT PLATE

Values of $C_f \times 10^3$

$h_1 = 0.01$, $k=1.15$, $\eta_\infty = 15.462$, $N_e = 39$

Re-number	Cebeci-Smith	Bismarck-Nasr	Pres.Meth.
860			24.7588
4 300			13.0596
8 600			10.3038
43 000			6.3999
100 000			5.2197
170 000			4.6600
260 000			4.2722
340 000			4.0408
510 000	3.9954	3.894	3.7012
680 000	3.8002	3.674	3.4921
860 000	3.6614	3.491	3.4749
1 030 000	3.5536	3.362	3.3896
1 280 000	3.4244	3.202	3.2140
1 500 000	3.3429	3.086	3.1112
1 710 000	3.2741	2.988	3.0739
1 880 000	3.2272	2.917	3.0312

The dotted curve represents the present analysis which uses the Levy-Lees transformation for the evaluation of the same quantity. Integrating the Eqs.(4.3.1,2) one obtains

$$\eta = \frac{\rho\ Ue}{\sqrt{2\rho\mu Ue\ x}}\, y \tag{6.7.1}$$

Inserting the relevant quantities:

$$\eta_\delta = \frac{\rho\ Ue}{\sqrt{2\rho\mu Ue\ x}}\ \delta \tag{6.7.2}$$

After simplification

$$\frac{\delta}{x} = \sqrt{2}\eta_\delta \quad Re^{(-1/2)} \tag{6.8}$$

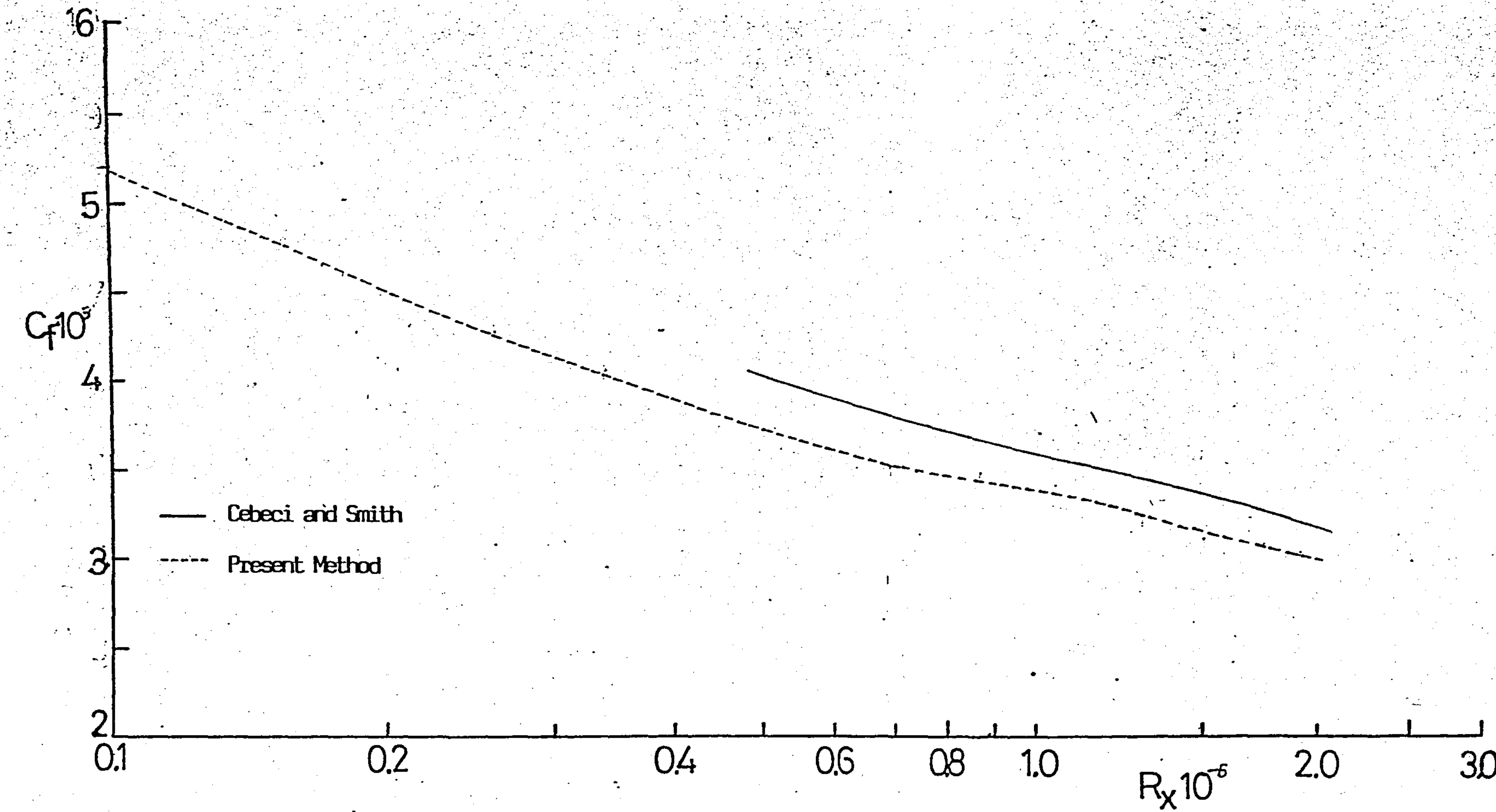


FIGURE 11

Variation of the Skin Friction Coefffficient of the Turbulent Flow over a Flat Plate

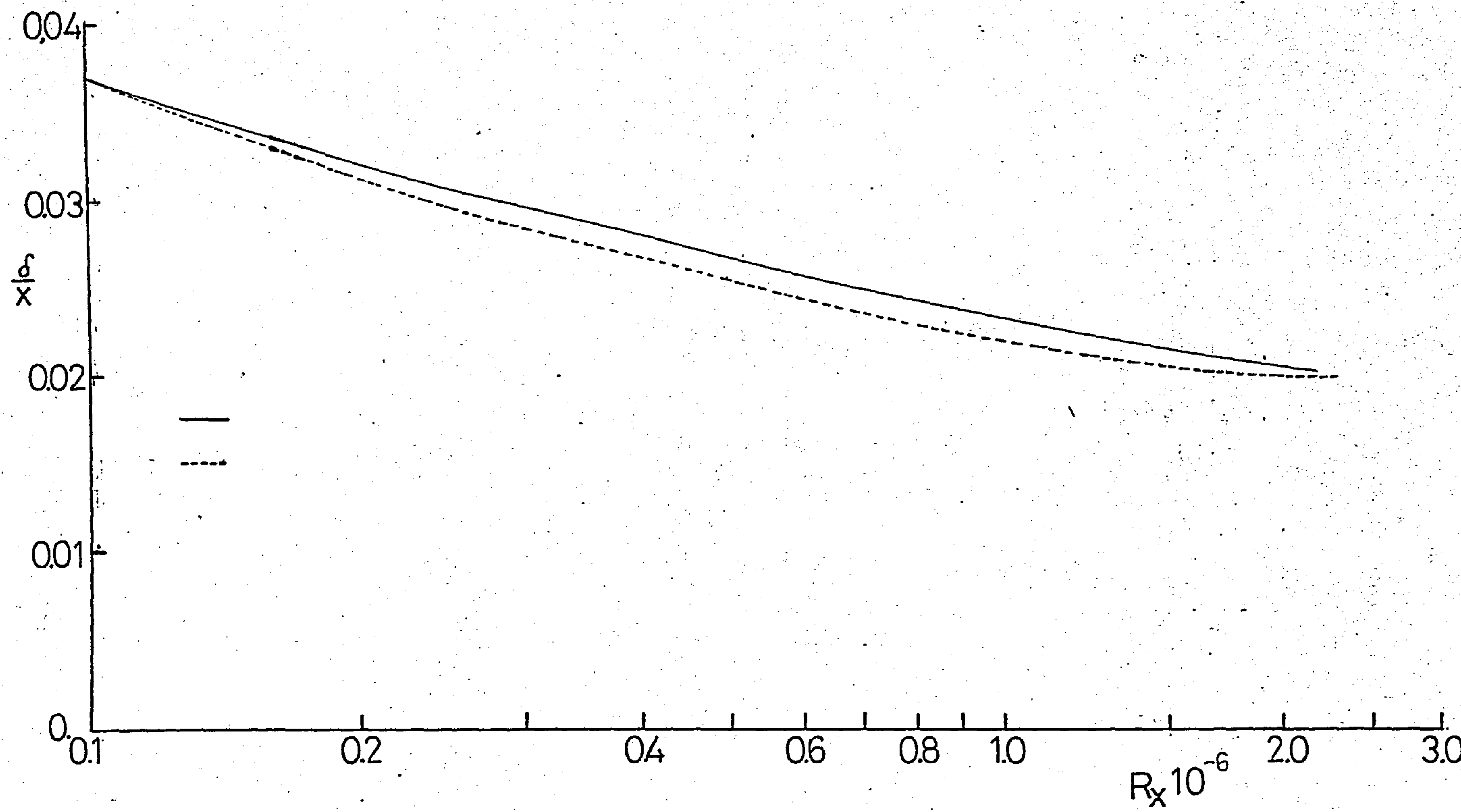


FIGURE 12

Variation of the Boundary Layer Thickness of the Turbulent Flow over a Flat Plate

It is again concluded that the results of the present analysis about 6.3 per cent less than the results of Eq.(6.6). This difference may be explained by the same reasons mentioned.

Fig. 13 shows the variation of the dimensionless velocity, f', with respect to transformed coordinate, η, normalized by the transformed boundary layer thickness, η_δ, with different Reynolds numbers. As can be noted there are slight variations between the curves shown. They can be explained by two reasons. First, the restriction of the computer system which may also cause some inaccuracy. Secondly, the calculation of η_δ which is performed by linear interpolation.

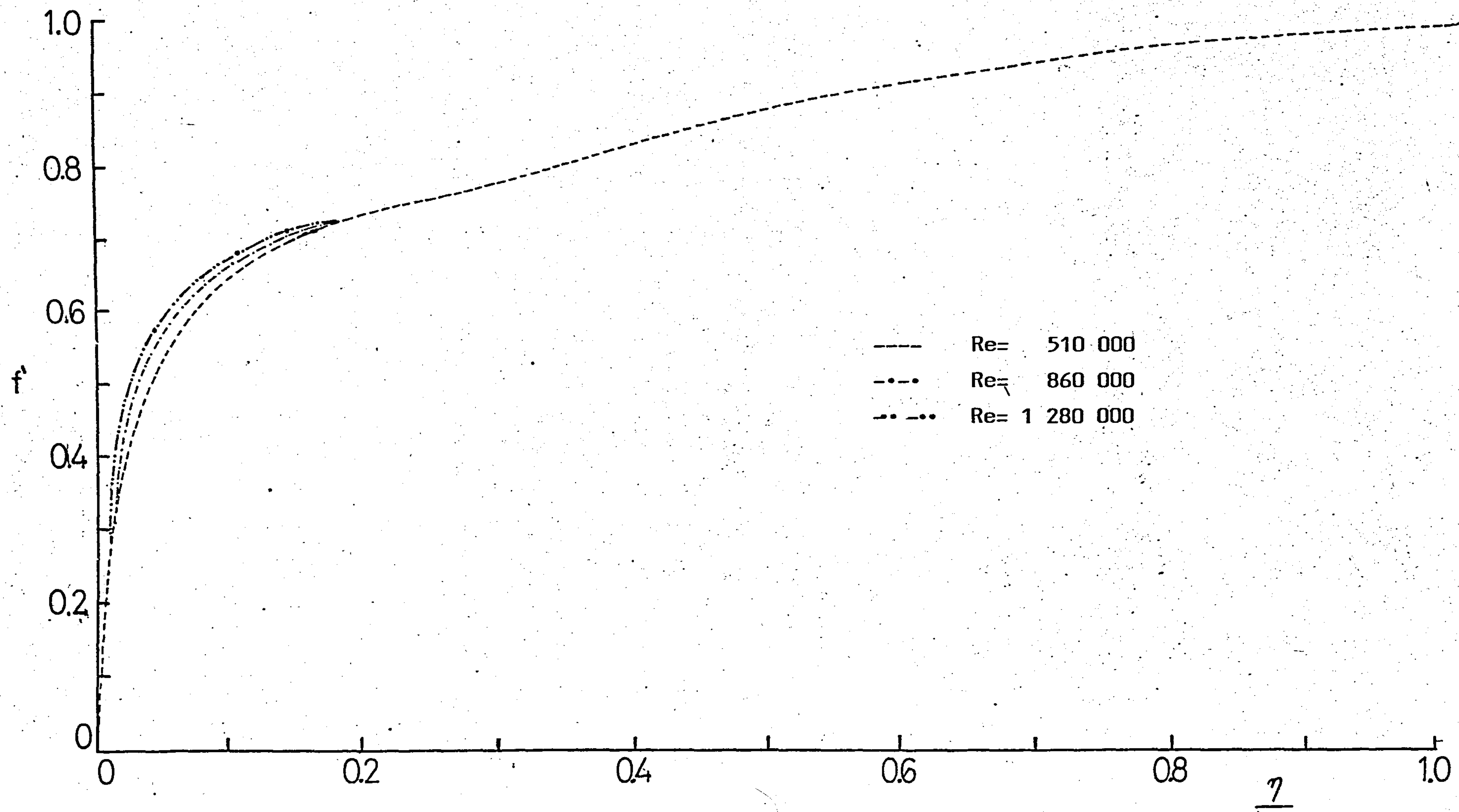


FIGURE 13

Velocity Profiles of the Turbulent Flow over a Flat Plate for Different Re-numbers

CHAPTER VII

CONCLUSIONS AND RECOMMENDATIONS

1. Three different versions of a computer code have been developed to solve two-dimensional laminar or turbulent incompressible boundary layer flows. The code is largely based on the solution method developed by Bismarck-Nasr(34).

2. The validity of the solution method proposed and of the computer code developed have been tested by comparing the results obtained with exact solutions and other available numerical solutions. Based on these applications the following conclusions have been obtained:

   a- In a prescribed solution domain the accuracy can be improved by increasing the number of elements.

   b- Smaller elements should be used as the velocity gradient increases.

   c- The accuracy of the present analysis decreases as the point of separation is approached.

   d- In case of similar flows the results obtained by the present analysis are in perfect agreement with exact solutions, e.g. in the solution of representative accelerated, flat plate and decelerated flows the error terms for f''_w are 0.0007 per cent, 0.0002 per cent and 0.0013 per cent, respectively.

   e- In case of non-similar flows, although the accuracy of the present study decreases gradually as the point of separation is approched, the prediction of the separation point is acceptable, e.g. for Howarth's Flow and Flow Past a Circular Cylinder the error terms are 2.08 per cent and 4.41 per cent, respectively.

f- In case of turbulent flows, although the present analysis uses six times fewer elements than other widely accepted numerical methods, the results obtained are acceptable, e.g. the difference in the calculation of skin friction coefficient is not more than six per cent.

3. The number of nodes used in the present applications should be increased to improve the accuracy.

4. A wider range of applications should be considered to obtain more confidence in the code, e.g. incompressible boundary layer development in turbines, pumps, fans and airplane wings.

5. The computer code developed can be improved by the following measures:

   a- Other turbulence models should be tested to obtain an improved mathematical description of the turbulent boundary layers.

   b- The code should be modified such that an optimum value for the upper limit of solution domain is selected automatically. Similarly, mesh generation routines should be developed to discretize the solution domain automatically.

   c- The present code that relies upon a two-step marching technique in the streamwise direction can be improved by incorporating a three step marching algorithm, instead.

6. The range of applicability of the code should be extended to axisymmetric terms existing in the formulation given in Chapter IV. This will allow the calculation of boundary layers in missiles, rockets, etc.

7. Extension of the code to subsonic compressible flows is also possible by the inclusion of the energy equation. This would allow the application of the code to compressible boundary layer flows encountered in turbines, compressors and high speed vehicles.

8. The boundary layer development on rotating surfaces can be calculated by including the Coriolis forces. This would enable the treatment of rotating blades.

APPENDIX A1

TRANSFORMATION OF TWO DIMENSIONAL NAVIER-STOKES EQUATIONS AND THE EQUATION OF CONTINUITY

As mentioned in Chapter II for the transformation of Eq.(2.3.1,2) and Eq.(2.4) the non-dimensional quantities which are defined by Eqs.(2.5.1,..,6) are employed. Consequently

$$x = l\,X \qquad \partial x = l\,\partial X \qquad (A.1.1,2)$$

$$y = l\,Y/\sqrt{Re} \qquad \partial y = (l/\sqrt{Re})\,\partial Y \qquad (A.1.3,4)$$

$$t = (l/U_\infty)\,T \qquad \partial t = (l/U_\infty)\,\partial T \qquad (A.1.5,6)$$

$$u = U_\infty\,U \qquad v = (U_\infty/\sqrt{Re})\,V \qquad (A.1.7,8)$$

$$p = \rho\,U_\infty^2\,P \qquad (A.1.9)$$

The terms employed in Eqs.(2.3.1,2) and Eq.(2.4) are transformed into

$$\frac{\partial u}{\partial t} = \frac{U_\infty}{l}\frac{\partial}{\partial T}(U_\infty U) = \frac{U_\infty^2}{l}\frac{\partial U}{\partial T} \qquad (A.2.1)$$

$$u\frac{\partial u}{\partial x} = U_\infty U\,\frac{1}{l}\frac{\partial}{\partial X}(U_\infty U) = \frac{U_\infty^2}{l}\,U\frac{\partial U}{\partial X} \qquad (A.2.2)$$

$$v\frac{\partial u}{\partial y} = \frac{U_\infty}{\sqrt{Re}}V\,\frac{\sqrt{Re}}{l}\frac{\partial}{\partial Y}(U_\infty U) = \frac{U_\infty^2}{l}\,V\frac{\partial U}{\partial Y} \qquad (A.2.3)$$

$$-\frac{1}{\rho}\frac{\partial p}{\partial x} = -\frac{1}{\rho}\frac{1}{l}\frac{\partial}{\partial X}(\rho\,U_\infty^2\,P) = -\frac{U_\infty^2}{l}\frac{\partial P}{\partial X} \qquad (A.2.4)$$

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{l}\frac{\partial}{\partial X}\left(\frac{U_\infty}{l}\frac{\partial U}{\partial X}\right) = \frac{U_\infty}{l^2}\frac{\partial^2 U}{\partial X^2} \qquad (A.2.5)$$

$$\frac{\partial^2 u}{\partial y^2} = \frac{\sqrt{Re}}{l}\frac{\partial}{\partial Y}\left(\frac{\sqrt{Re}}{l}U_\infty\frac{\partial U}{\partial Y}\right) = \frac{Re\,U_\infty}{l^2}\frac{\partial^2 U}{\partial Y^2} \qquad (A.2.6)$$

$$\frac{\partial v}{\partial t} = \frac{U}{l}\frac{\partial}{\partial T}\left(\frac{U_\infty}{\sqrt{Re}} V\right) \qquad = \frac{U_\infty^2}{l\sqrt{Re}}\frac{\partial V}{\partial T} \tag{A.2.7}$$

$$u\frac{\partial v}{\partial x} = (U_\infty\, U)\ \frac{1}{l}\frac{\partial}{\partial X}\left(\frac{U_\infty}{\sqrt{Re}} V\right) = \frac{U_\infty^2}{l\sqrt{Re}} U \frac{\partial V}{\partial X} \tag{A.2.8}$$

$$v\frac{\partial v}{\partial y} = \left(\frac{U_\infty}{\sqrt{Re}} V\right)\frac{\sqrt{Re}}{l}\frac{\partial}{\partial Y}\left(\frac{U_\infty}{\sqrt{Re}} V\right) = \frac{U_\infty^2}{l\sqrt{Re}} V \frac{\partial V}{\partial Y} \tag{A.2.9}$$

$$-\frac{1}{\rho}\frac{\partial p}{\partial y} = -\ \frac{1}{\rho}\ \frac{\sqrt{Re}}{l}\frac{\partial}{\partial Y}(\rho\ U_\infty^2\ P) = -\frac{U_\infty^2}{l}\sqrt{Re}\frac{\partial P}{\partial Y} \tag{A.2.10}$$

$$\frac{\partial^2 v}{\partial x^2} = \frac{1}{l}\frac{\partial}{\partial X}\left(\frac{U_\infty}{l\sqrt{Re}}\ \frac{\partial V}{\partial X}\right) \qquad = \frac{U_\infty}{l^2\sqrt{Re}}\frac{\partial^2 V}{\partial X^2} \tag{A.2.11}$$

$$\frac{\partial^2 v}{\partial y^2} = \frac{\sqrt{Re}}{l}\frac{\partial}{\partial Y}\left(\frac{U_\infty}{l}\ \frac{\partial V}{\partial Y}\right) \qquad = \frac{U_\infty}{l^2}\sqrt{Re}\ \frac{\partial^2 V}{\partial Y^2} \tag{A.2.12}$$

Eq.(2.4) becomes

$$\frac{U_\infty}{l}\frac{\partial U}{\partial X} + \frac{U_\infty}{l}\frac{\partial V}{\partial Y} = 0 \tag{A.2.13}$$

$$\frac{\partial U}{\partial X} + \frac{\partial V}{\partial Y} \qquad = 0 \tag{2.7}$$

Eq.(2.3.1) is transformed as follows:

$$\frac{U_\infty^2}{l}\frac{\partial U}{\partial T} + \frac{U_\infty^2}{l} U \frac{\partial U}{\partial X} + \frac{U_\infty^2}{l} V \frac{\partial U}{\partial Y} = -\frac{U_\infty^2}{l}\frac{\partial P}{\partial X} + \frac{\nu U_\infty}{l^2}\frac{\partial^2 U}{\partial X^2} + \frac{\nu\ Re\ U_\infty}{l^2}\frac{\partial^2 U}{\partial Y^2}$$

multiplying by $(l\ /\ U_\infty^2)$

$$\frac{\partial U}{\partial T} + U\frac{\partial U}{\partial X} + V\frac{\partial U}{\partial Y} = -\frac{\partial P}{\partial X} + \frac{\nu}{U_\infty l}\frac{\partial^2 U}{\partial X^2} + \frac{\nu\ Re}{l\ U_\infty}\frac{\partial^2 U}{\partial Y^2}$$

recalling Eq.(2.2), definition of Re

$$\frac{\partial U}{\partial T} + U\frac{\partial U}{\partial X} + V\frac{\partial U}{\partial Y} = -\frac{\partial P}{\partial X} + \frac{1}{Re}\frac{\partial^2 U}{\partial X^2} + \frac{\partial^2 U}{\partial Y^2} \tag{2.6.1}$$

Eq.(2.3.2) is transformed as follows:

$$\frac{U_\infty^2}{l\sqrt{Re}}\frac{\partial V}{\partial T} + \frac{U^2}{l\sqrt{Re}} U \frac{\partial V}{\partial X} + \frac{U_\infty^2}{l\sqrt{Re}} V \frac{\partial V}{\partial Y} = -\frac{U_\infty^2}{l}\sqrt{Re}\frac{\partial P}{\partial Y} + \nu\frac{U_\infty}{l^2\sqrt{Re}}\frac{\partial^2 V}{\partial X^2} + \nu\frac{U_\infty}{l^2}\sqrt{Re}\frac{\partial^2 V}{\partial Y^2}$$

multiplying by $(l\sqrt{Re}\ /\ U_\infty^2)$

$$\frac{\partial V}{\partial T} + U\frac{\partial V}{\partial X} + V\frac{\partial V}{\partial Y} = -\ Re\frac{\partial P}{\partial Y} + \frac{\nu}{lU_\infty}\frac{\partial^2 U}{\partial X^2} + \frac{\nu}{lU_\infty}\frac{\partial^2 V}{\partial Y^2}$$

recalling Eq.(2.1) and dividing by Re

$$\frac{1}{Re}\left(\frac{\partial V}{\partial T} + U\frac{\partial V}{\partial X} + V\frac{\partial V}{\partial Y}\right) = -\frac{\partial P}{\partial Y} + \frac{1}{Re^2}\frac{\partial^2 U}{\partial X^2} + \frac{1}{Re}\frac{\partial^2 V}{\partial Y^2} \qquad (2.6.2)$$

APPENDIX A2

INCLUSION OF THE AXISYMMETRICITY AND TURBULENCE

1. Inclusion of the Axisymmetric Flows (21) :

The governing boundary layer equations for axisymmetric flows do not differ much from those of two-dimensional flows. The external potential velocity is a function of one dimension and the velocity within the boundary layer has two components.

The arbitrary radial distance can be obtained from the following equation:

$$r(x,y) = r_0(x) + y \cos\alpha \tag{A.3}$$

where r_0, the normal radius of the solid body dependent on the distance; y, the normal physical coordinate and α, the angle between the x-axis and the axis of the symmetricity.

The term r^k is included in Eqs.(2.11,12) to generalize the equations for axisymmetirc flows. For axisymmetric flows, k equals one, wheras it is zero for two-dimensional flows.

2. Mean Motion and Fluctuations of Turbulent Flows:

In describing a turbulent flow in mathematical terms it is convenient to separate it into a mean motion and into a fluctuation, or eddying motion. Consequently, the velocity components and pressure can be written as follows:

$$u = \bar{u} + u' \qquad v = \bar{v} + v' \qquad , p = \bar{p} + p' \tag{A.4.1,2,3}$$

where $\bar{u}$, $\bar{v}$ and $\bar{p}$ are the time-average of u-velocity, v-velocity and pressure, respectively. The terms u', v' and p' denote the fluctuations of u-velocity, v-velocity and pressure, respectively.

By definition the time averages of all quantities describing the fluctuations equal to zero.

$$\bar{u}' = 0 \qquad \bar{v}' = 0 \qquad \bar{p}' = 0 \tag{A.5.1,2,3}$$

The main physical importance of turbulent motion consists of the apparent increase of the viscosity of the mean stream. Therefore, it is obligatory to find a mathematical expression, which describes the effect of the "apparent" viscosity.

Since the flux of momentum per unit time through an area is always equivalent to an equal and opposite force exerted on the area by the surroundings, it is concluded that the area under consideration, which is normal to the x-axis, is acted upon by the stresses

$$- \rho \left(\bar{u}^2 + \overline{u'^2} \right) \tag{A.6.1}$$

in the x-direction, and

$$- \rho \left(\bar{u}\,\bar{v} + \overline{u'v'} \right) \tag{A.6.2}$$

in the y-direction

Consequently, it is seen that the superposition of fluctuation on the mean motion gives rise to two additional stresses

$$\sigma'_x = - \rho\, \overline{u'^2} \qquad \tau'_{yx} = -\rho\, \overline{u'v'} \tag{A.7.1,2}$$

They are termed "apparent" or Reynolds stresses. Eq.(A.7.2) is the second component of the shear stress which will be superposed to the normal viscous shear stress.

3. Derivation of Eqs.(2.11,12)

Assuming steady, compressible, two-dimensional flow and including the facts of the first two sections of the present appendix give rise to the following set of boundary-layer equations:

Continuity:

$$\frac{\partial}{\partial x}\left(r^k \ \rho \ u \right) + \frac{\partial}{\partial y}\left(r^k \ \rho \ v\right) = 0 \tag{2.11}$$

Momentum:

$$\rho u \frac{\partial u}{\partial x} + \rho v \frac{\partial u}{\partial y} = - \frac{dp}{dx} + \frac{1}{r^k} \frac{\partial}{\partial y}\left(r^k\left(\mu \frac{\partial u}{\partial y} - \rho \ \overline{u'v'}\right)\right) \tag{2.12}$$

APPENDIX B

APPLICATION OF LEVY-LEES TRANSFORMATION

For the transformation Eqs.(4.3.1,2), (4.4.1,2), (4.5), (4.6.1,2) and (4.8) will be used.

$$\frac{\partial\psi}{\partial\xi} = \frac{\partial}{\partial\xi} (\sqrt{2\xi}\ f(\xi,\eta))$$

$$= \frac{\partial}{\partial\xi} \sqrt{2\xi}\ f + \sqrt{2\xi}\ \frac{\partial f}{\partial\xi}$$

$$= \frac{f}{\sqrt{2\xi}} + \sqrt{2\xi}\ \frac{\partial f}{\partial\xi}$$

$$\frac{\partial\psi}{\partial\xi} = \sqrt{2\xi}\ (\frac{f}{2\xi} + \frac{\partial f}{\partial\xi}) \tag{4.7.1}$$

$$\frac{\partial\psi}{\partial\eta} = \frac{\partial}{\partial\eta} (\sqrt{2\xi}\ f(\xi,\eta))$$

$$= \sqrt{2\xi}\ \frac{\partial f}{\partial\eta}$$

$$\frac{\partial\psi}{\partial\eta} = \sqrt{2\xi}\ f' \tag{4.7.2}$$

Eq.(4.4.1,2) and (4.6.1,2) will be used to evaluate

$$\rho u = \frac{\partial\psi}{\partial y} \tag{4.4.1}$$

$$\rho u = \frac{\rho Ue}{\sqrt{2\xi}} \frac{\partial\psi}{\partial\eta}$$

using Eq.(5.7.2)

$$\rho u = \frac{\rho Ue}{\sqrt{2\xi}} \sqrt{2\xi}\ f'$$

$$u = U_e\ f' \tag{4.9.1}$$

Similarly

$$\rho v = - \frac{\partial\psi}{\partial x} \tag{4.4.2}$$

$$\rho v = - \rho\mu Ue \left(\frac{\partial\psi}{\partial\xi} + \frac{\partial\eta}{\partial\xi}\frac{\partial\psi}{\partial\eta} \right)$$

$$v = - \mu Ue \left(\sqrt{2\xi} \left(\frac{f}{2\xi} + \frac{\partial f}{\partial\xi} \right) + \frac{\partial\eta}{\partial\xi}\sqrt{2\xi}\, f' \right)$$

$$v = - \mu Ue \sqrt{2\xi} \left(\frac{f}{2\xi} + \frac{\partial f}{\partial\xi} + f' \frac{\partial\eta}{\partial\xi} \right) \tag{4.9.2}$$

At this stage each term of Eq.(4.2) will be evaluated for transformation and future reference. Using Eqs.(4.6.1,2) and Eq.(4.9.1,2)

$$\frac{\partial u}{\partial x} = \rho\mu Ue \left(\frac{\partial u}{\partial\xi} + \frac{\partial\eta}{\partial\xi}\frac{\partial u}{\partial\eta} \right)$$

where

$$\frac{\partial u}{\partial\xi} = \frac{\partial}{\partial\xi}(Ue\ f')$$

$$\frac{\partial u}{\partial\xi} = \frac{dUe}{d\xi} f' + Ue \frac{\partial f'}{\partial\xi} \tag{B.1.1}$$

Since $Ue = Ue(x) \longrightarrow Ue(\xi)$, and

$$\frac{\partial u}{\partial\eta} = \frac{\partial}{\partial\eta}(Ue\ f')$$

$$\frac{\partial u}{\partial\eta} = Ue\ f'' \tag{B.1.2}$$

Hence

$$\frac{\partial u}{\partial x} = \rho\mu Ue \left(f' \frac{dUe}{d\xi} + Ue \frac{\partial f'}{\partial\xi} + Ue\ f'' \frac{\partial\eta}{\partial\xi} \right) \tag{B.2}$$

Similarly

$$\frac{\partial u}{\partial y} = \frac{\rho Ue}{\sqrt{2\xi}} \frac{\partial u}{\partial\eta}$$

recalling (B.1.2)

$$\frac{\partial u}{\partial y} = \frac{\rho\ Ue}{\sqrt{2\xi}} Ue\ f''$$

$$\frac{\partial u}{\partial y} = \frac{\rho}{\sqrt{2\xi}} U_e^2\ f'' \tag{B.3}$$

$$\frac{\partial^2 u}{\partial y^2} = \frac{\rho Ue}{\sqrt{2\xi}} \frac{\partial}{\partial \eta} \left(\frac{\rho}{\sqrt{2\xi}} U_e^2\ f'' \right)$$

$$= \frac{\rho^2 U_e^3}{2\xi} f''' \tag{B.4}$$

$$\frac{dp}{dx} = - \rho\ Ue \frac{dUe}{dx}$$

$$= - \rho Ue\ \rho\mu Ue \left(\frac{\partial Ue}{\partial \xi} + \frac{\partial \eta}{\partial \xi} \frac{\partial Ue}{\partial \eta} \right)$$

$$\frac{dp}{dx} = -\mu\rho^2\ U_e^2 \frac{dUe}{d\xi} \tag{B.5}$$

Eq.(4.2) is expanded:

$$\rho u \frac{\partial u}{\partial x} + \rho v \frac{\partial u}{\partial y} = - \frac{dp}{dx} + \mu \frac{\partial^2 u}{\partial y^2} + \rho \frac{\partial \varepsilon}{\partial y} \frac{\partial u}{\partial y} + \rho\varepsilon \frac{\partial^2 u}{\partial y^2}$$

Dividing by ρ

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = - \frac{1}{\rho} \frac{dp}{dx} + \nu \frac{\partial^2 u}{\partial y^2} + \frac{\partial \varepsilon m}{\partial y} \frac{\partial u}{\partial y} + \varepsilon_m \frac{\partial^2 u}{\partial y^2}$$

using Eq.(5.9.1) and (B.2)

$$u \frac{\partial u}{\partial x} = U_e f'\ \rho\mu U_e \left(f' \frac{dUe}{d\xi} + U_e \frac{\partial f'}{\partial \xi} + U_e f'' \frac{\partial \eta}{\partial \xi}\right)$$

$$u \frac{\partial u}{\partial x} = \rho\mu U_e^3 f' \left(f' \frac{1}{Ue} \frac{dUe}{d\xi} + \frac{\partial f'}{\partial \xi} + f'' \frac{\partial \eta}{\partial \xi} \right) \tag{B.7}$$

using Eq.(4.9.2) and (B.3)

$$v \frac{\partial u}{\partial y} = - \mu U_e \sqrt{2\xi} \left(\frac{f}{2\xi} + \frac{\partial f}{\partial \xi} + f' \frac{\partial \eta}{\partial \xi}\right) \frac{\rho U_e^2}{\sqrt{2\xi}} f''$$

$$v \frac{\partial u}{\partial y} = - \rho\mu U_e^3\ f'' \left(\frac{f}{2\xi} + \frac{\partial f}{\partial \xi} + f' \frac{\partial \eta}{\partial \xi}\right) \tag{B.8}$$

Furthermore, eddy viscosity expression is only function of η.

$$\varepsilon = \varepsilon\ (\eta) \tag{B.9}$$

$$\frac{\partial \varepsilon m}{\partial y} = \frac{\rho Ue}{\sqrt{2\xi}} \frac{\partial \varepsilon}{\partial \eta}$$

$$\frac{\partial \varepsilon m}{\partial y} = \frac{\rho Ue}{\sqrt{2\xi}}\,\varepsilon' \tag{B.10}$$

$$\frac{\partial \varepsilon}{\partial y}\,\frac{\partial u}{\partial y} = \frac{\rho Ue}{\sqrt{2\xi}}\,\varepsilon'\frac{\rho Ue^2}{\sqrt{2\xi}}\,f''$$

$$\frac{\partial \varepsilon}{\partial y}\,\frac{\partial u}{\partial y} = \frac{\rho^2 Ue^3}{2\xi}\,f''\,\varepsilon' \tag{B.11}$$

Eqs.(B.4,5,7,8,11) will be substituted into (B.6) to obtain

$$\mu Ue^3\rho f'\left(\frac{f'}{Ue}\frac{dUe}{d\xi} + \frac{\partial f'}{\partial \xi} + \frac{\partial \eta}{\partial \xi} f''\right) - \rho\mu U_e^3 f''\left(\frac{f}{2\xi} + \frac{\partial f}{\partial \xi} + \frac{\partial \eta}{\partial \xi} f'\right)$$

$$= \mu\rho U_e^2\,\frac{dUe}{d\xi} + (\varepsilon+\nu)\,\frac{\rho^2 Ue^3}{2\xi}\,f''' + \rho^2 U_e^3 f''\,\varepsilon' \tag{B.12.a}$$

Multiplying both sides of Eq.(B.12.a) by $\frac{2\xi}{Ue^3\rho\mu}$

and defining $\varepsilon^+ \equiv \frac{\varepsilon}{\nu}$ (4.12)

$$2\xi f'\left(\frac{f'}{Ue}\frac{dUe}{d\xi} + \frac{\partial f'}{\partial \xi} + \frac{\partial \eta}{\partial \xi}f''\right) - 2\xi f''\left(\frac{f}{2\xi} + \frac{\partial f}{\partial \xi} + f'\frac{\partial \eta}{\partial \xi}\right) = \frac{2\xi}{Ue}\frac{dUe}{d\xi} + (\varepsilon^+ + 1)f''' + \varepsilon^{+}{}' f'' \tag{B.12.b}$$

Recalling definition(4.11) and after expansion of Eq.(B.12.b) and cancellation of the unnecessary terms one obtains

$$\beta f'^2 + 2\xi f'\frac{\partial f'}{\partial \xi} - ff'' - 2\xi f''\frac{\partial f}{\partial \xi} = \beta + \varepsilon^+ f''' + f''' + \varepsilon^{+}{}' f'' \tag{B.12.c}$$

since $\varepsilon^{+}{}' f'' + f''' + \varepsilon^+ f''' = \{(1+\varepsilon^+)f''\}'$, the final form becomes

$$2\xi\left(f'\frac{\partial f'}{\partial \xi} - f''\frac{\partial f}{\partial \xi}\right) = \beta(1-f'^2) + ff'' + \{(1+\varepsilon^+)f''\}' \tag{4.10}$$

To evaluate the boundary conditions one uses (4.9.1,2)

if $y=0 \rightarrow \eta=0$

$y=\infty \rightarrow \eta=\infty$

Hence $0 = U_e\,f'(\xi,0)$

$$f'(\xi,0) = 0 \tag{4.13.a}$$

$$U_e = U_e\,f'(\xi,\eta_\infty)$$

$$f'(\xi,\eta_\infty) = 1 \tag{4.13.b}$$

$$0 = -\mu U_e \sqrt{2\xi} \left(\frac{f}{2\xi} + \frac{\partial f}{\partial \xi} + \frac{\partial \eta}{\partial \xi} f'\right)$$

$$0 = \frac{f}{2\xi}$$

$$f(\xi,0) = 0 \qquad (4.13.c)$$

APPENDIX C

APPLICATION OF LEVY-LEES TRANSFORMATION TO EDDY VISCOSITY EXPRESSIONS

Eqs.(4.14) and (4.15) will be transformed by Levy-Lees transformation. For this purpose several quantities which have previously been evaluated will be used.

Integrating Eq.(4.3.2) and solving for y yields

$$y = \frac{\sqrt{2\xi}}{\rho Ue}\,\eta \tag{C.1}$$

Using Eq.(2.16) with Eq.(B.3)

$$\tau_w = \mu\,\frac{\rho Ue}{\sqrt{2\xi}}\,f''_w \tag{C.2}$$

Eqs.(C.1,2) and (B.5) will be substituted into Eq.(4.14)

$$\varepsilon_i = \{0.4\frac{\sqrt{2\xi}}{\rho Ue}\eta\}^2\{1-\exp(-\frac{\sqrt{2\xi}}{Ue}\eta(\mu\rho\frac{Ue\,fW}{\sqrt{2\xi}\,\rho} - \mu\rho^2\,\frac{Ue^2}{\rho}\,\frac{dUe}{d\xi}\,\frac{\sqrt{2\xi}}{\rho Ue}\eta)^{1/2} / 26\nu)\}^2\left|\frac{\rho Ue^2}{\sqrt{2\xi}}f''\right|$$

$$\varepsilon_i = 0.16\frac{\sqrt{2\xi}}{\rho}\eta^2|f''|\{1-\exp(-\frac{\sqrt{2\xi}}{\rho}\,\frac{\eta\rho}{26\mu}(\frac{fW}{\zeta} - \frac{\beta\eta}{\zeta})^{1/2})\}^2$$

where the definitions(4.11) and (4.18) are employed. Dividing by ν

$$\varepsilon_i^+ = 0.16\zeta\eta^2|f''|\{1-\exp(-\frac{\zeta\eta}{26}(\left|\frac{fW}{\zeta} - \frac{\beta\eta}{\zeta}\right|)^{1/2})\}^2 \tag{4.16}$$

Similarly, the expression of the outer section will be treated. Using Eq.(C.1) and substituting

$$\delta = \frac{\sqrt{2\xi}}{\rho Ue}\,\eta_\infty \tag{C.3}$$

$$\frac{y}{\delta} = \frac{\eta}{\eta_\infty} \tag{C.4}$$

Furthermore the integral expression can be written as follows:

$$\int_0^\infty (U_e - u)\, dy = U_e \int_0^\infty \left(1 - \frac{u}{U_e}\right) dy$$

$$= U_e \int_0^\infty (1 - f')\, \frac{\sqrt{2\xi}}{\rho U_e}\, d\eta$$

$$= \frac{\sqrt{2\xi}}{\rho} \int_0^\infty (1 - f')\, d\eta \tag{C.5}$$

Hence

$$\varepsilon_o = 0.0168\, \frac{\sqrt{2\xi}}{\rho} \left| \int_0^\infty (1 - f')\, d\eta \right| \left\{1 + 5.5\left(\frac{\eta}{\eta_\infty}\right)^6\right\}^{-1}$$

Dividing by ν

$$\varepsilon_o^+ = 0.0168\zeta \left| \int_0^\infty (1 - f')\, d\eta \right| \left\{1 + 5.5\left(\frac{\eta}{\eta_\infty}\right)^6\right\}^{-1} \tag{4.17}$$

APPENDIX D

LINEARIZATION OF THE MOMENTUM EQUATION

Combining Eqs.(4.21.1,4) and (4.20)

$$(1+\varepsilon^{+})(f'''+\delta f''')+\varepsilon^{+'}(f''+\delta f'')+(f+\delta f)(f''+\delta f'')+\beta\{1-(f'+\delta f')^{2}\}$$

$$= \alpha\{(f'+\delta f')(f'+\delta f'-f'_{n-1})-(f''+\delta f'')(f+\delta f-f_{n-1})\} \qquad \text{(D.1)}$$

Neglecting the second order variation terms one obtains

$$(1+\varepsilon^{+})\delta f'''+(\varepsilon^{+'}+f-\alpha f_{n-1}+\alpha f)\delta f''+(-2\beta f'-2\alpha f'+\alpha f'_{n-1})\delta f'+(f''+\alpha f'')\delta f$$

$$+\ (1+\varepsilon^{+})f'''+\varepsilon^{+'}f''+ff''+\beta-\beta f'^{2}-\alpha f'^{2}+\alpha f'f'_{n-1}+\alpha ff''-\alpha f''f_{n-1} = 0$$

Collecting and arranging the terms one obtains Eq.(4.23.1,4)

APPENDIX E

DERIVATION OF SECOND ORDER HERMITIAN POLYNOMIALS

Any function is approximated as

$$y(x) = h_i(x)\ f(x_i) + \bar{h}_i\ f'(x_i) + \bar{\bar{h}}_i\ f''(x_i) \qquad (E.1.1)$$

$$y'(x) = h_i'(x)\ f(x_i) + \bar{h}_i'\ f'(x_i) + \bar{\bar{h}}_i'\ f''(x_i) \qquad (E.1.2)$$

$$y''(x) = h_i''(x)\ f(x_i) + \bar{h}_i''\ f'(x_i) + \bar{\bar{h}}_i''\ f''(x_i) \qquad (E.1.3)$$

$$h_i(x) = r_i(x)\ \{l_i(x)\}^3 \qquad (E.2.1)$$

$$\bar{h}_i(x) = s_i(x)\ \{l_i(x)\}^3 \qquad (E.2.2)$$

$$\bar{\bar{h}}_i(x) = t_i(x)\ \{l_i(x)\}^3 \qquad (E.2.3)$$

where

$l_i(x)$: Lagrange Interpolation Function of which two point (first order) version will be used

Since n-th order Hermitian polynomial is of power (2n+1); r_i, s_i and t_i must be second order to add up the power of the functions $h_i(x)$, $\bar{h}_i(x)$ and $\bar{\bar{h}}_i(x)$ to five.

For the rest of the derivation the variable will be used instead of x, since it is encountered in the present study.

The following definition will be used

$$L \equiv \eta_2 - \eta_1 \qquad (E.3)$$

The following transformation will also be employed

$$z = \frac{\eta - \eta 1}{L} \tag{E.4}$$

Similarly

$$\eta - \eta_1 = \frac{\eta - \eta 1}{L} L = z\,L \tag{E.5.1}$$

$$\eta - \eta_2 = (z - 1)\,L \tag{E.5.2}$$

The Lagrange Intepolation Functions become

$$l_1(\eta) = \frac{\eta - \eta 2}{\eta 1 - \eta 2} \qquad l_2(\eta) = \frac{\eta - \eta 1}{\eta 2 - \eta 1}$$

Using Eq.(E.3)

$$l_1(\eta) = \frac{\eta - \eta 1}{-L} + 1 \quad \text{(E.6.1)} \qquad l_2(\eta) = \frac{\eta - \eta 1}{L} \quad \text{(E.7.1)}$$

$$l_1(z) = -z + 1 \quad \text{(E.6.2)} \qquad l_2(z) = z \quad \text{(E.7.2)}$$

Eqs.(E.2.1,2,3) are rewritten as the functions of η. Their derivatives become

$$h_i'(\eta) = r_i'(\eta)\,\{l_i(\eta)\}^3 + 3r_i(\eta)\,\{l_i(\eta)\}^2\,l_i'(\eta) \tag{E.8.1}$$

$$\bar{h}_i'(\eta) = s_i'(\eta)\,\{l_i(\eta)\}^3 + 3s_i(\eta)\,\{l_i(\eta)\}^2\,l_i'(\eta) \tag{E.9.1}$$

$$\bar{\bar{h}}_i'(\eta) = t_i'(\eta)\,\{l_i(\eta)\}^3 + 3t_i(\eta)\,\{l_i(\eta)\}^2\,l_i'(\eta) \tag{E.10.1}$$

$$h_i''(\eta) = r_i''(\eta)\,\{l_i(\eta)\}^3 + 6r_i'(\eta)\,\{l_i(\eta)\}^2\,l_i'(\eta) + 6r_i(\eta)\,l_i(\eta)\,\{l_i'(\eta)\}^2 + 3r_i(\eta)\,\{l_i(\eta)\}^2\,l_i''(\eta) \tag{E.8.2}$$

$$\bar{h}_i''(\eta) = s_i''(\eta)\,\{l_i(\eta)\}^3 + 6s_i'(\eta)\,\{l_i(\eta)\}^2\,l_i'(\eta) + 6s_i(\eta)\,l_i(\eta)\,\{l_i'(\eta)\}^2 + 3s_i(\eta)\,\{l_i(\eta)\}^2\,l_i''(\eta) \tag{E.9.2}$$

$$\bar{\bar{h}}_i''(\eta) = t_i''(\eta)\,\{l_i(\eta)\}^3 + 6t_i'(\eta)\,\{l_i(\eta)\}^2\,l_i'(\eta) + 6t_i(\eta)\,l_i(\eta)\,\{l_i'(\eta)\}^2 + 3t_i(\eta)\,\{l_i(\eta)\}^2\,l_i''(\eta) \tag{E.10.2}$$

Since at the points of interpolation the exact and interpolated values have to be equal to each other, i.e.

$$y(\eta_i) = f(\eta_i) \qquad y'(\eta_i) = f'(\eta_i) \qquad y''(\eta_i) = f''(\eta_i) \tag{E.11.1,2,3}$$

the following equations can be obtained:

$$h_i(\eta_i) = 1 \qquad \bar{h}_i(\eta_i) = 0 \qquad \bar{\bar{h}}_i(\eta_i) = 0 \tag{E.12.1,2,3}$$

$$h_i'(\eta_i) = 0 \qquad \bar{h}_i'(\eta_i) = 1 \qquad \bar{\bar{h}}_i'(\eta_i) = 0 \tag{E.13.1,2,3}$$

$$h_i''(\eta_i) = 0 \qquad \bar{h}_i''(\eta_i) = 0 \qquad \bar{\bar{h}}_i''(\eta_i) = 1 \tag{E.14.1,2,3}$$

Based on Eqs.(E.12.1,2,3) Eqs.(E.8,9,10) yield

$$r_i(\eta_i) = 1 \qquad s_i(\eta_i) = 0 \qquad t_i(\eta_i) = 0 \tag{E.15.1,2,3}$$

Since $\quad l_i(\eta_i) = 1 \qquad$ (E.16)

Based on Eqs.(E.13.1,2,3) Eqs.(E.8.1,9.1,10.1) yield

$$h_i'(\eta_i) = 0 \qquad \bar{h}_i'(\eta_i) = 1 \qquad \bar{\bar{h}}_i'(\eta_i) = 0 \tag{E.17.1,2,3}$$

Using Eqs.(E.17.1), (E.16) and (E.15.1), Eq.(E.8.1) becomes

$$r_i'(\eta_i) + 3l_i'(\eta_i) = 0$$

which is rearranged to yield

$$r_i'(\eta_i) = -3l_i'(\eta_i) \tag{E.18}$$

Using Eqs.(E.17.2) and (E.16), Eq.(E.9.1) becomes

$$s_i'(\eta_i) = 1 \tag{E.19}$$

Eq.(E.10.1) yields

$$t_i'(\eta_i) = 0 \tag{E.20}$$

Based on Eqs.(E.14.1,2,3), Eqs.(E.8.2, 9.2, 10.2) yield

$$h_i''(\eta_i) = 0 \qquad \bar{h}_i''(\eta_i) = 0 \qquad \bar{\bar{h}}_i''(\eta_i) = 1 \tag{E.21.1,2,3}$$

Using Eqs.(E.21.1), (E.16), (E.15.1) and (E.18), Eq.(E.8.2) becomes after rearrangement

$$r_i''(\eta_i) = 12\{l_i'(\eta_i)\}^2 - 3l_i''(\eta_i) \tag{E.22}$$

Using Eq.(E.21.2), (E.16) and (E.19), Eq.(E.9.2) becomes after rearrangement

$$s_i''(\eta_i) = -6l_i'(\eta_i) \tag{E.23}$$

Using Eq.(E.21.3), (E.15.3), (E.20) and (E.16), Eq.(E.10.2) becomes

$$t_i''(\eta_i) = 1 \tag{E.24}$$

This equation is integrated to yield

$$t_i'(\eta) = \quad - \quad {}_i \tag{E.24.1}$$

$$t_i(\eta) = \frac{1}{2}(\eta - \eta_i)^2 \tag{E.25}$$

Consequently

$$t_1(\eta) = \frac{1}{2}(\eta - \eta_1)^2 \qquad t_2(\eta) = \frac{1}{2}(\eta - \eta_2)^2 \tag{E.26.1,2}$$

Using Eqs.(E.5.1,2) one obtains

$$t_1(z) = \frac{1}{2}L^2 z^2 \qquad t_2(z) = \frac{1}{2}L^2(z - 1)^2 \tag{E.27.1,2}$$

Differentiating Eqs.(E.6.1) and (E.7.1)

$$l_1'(\eta) = -\frac{1}{L} \qquad l_2'(\eta) = \frac{1}{L} \qquad l_i''(\eta_i) = 0 \tag{E.28.1,2,3}$$

Using Eqs.(E.23), (E.19) and (E.15.2) it is obtained

$$s_i''(\eta) = -6\, l_i'(\eta_i) \tag{E.29.1}$$

$$s_i'(\eta) = -6l_i'(\eta_i)\,(\eta - \eta_i) + 1 \tag{E.29.2}$$

$$s_i(\eta) = -3l_i'(\eta_i)\,(\eta - \eta_i)^2 + (\eta - \eta_i) \tag{E.29.3}$$

after rearrangement

$$s_i(\eta) = (\eta - \eta_i)\,\{1 - 3l_i'(\eta_i)(\eta - \eta_i)\} \tag{E.30}$$

Substituting the subscripts and using Eqs.(E.28.1,2)

$$s_1(\eta) = (\eta - \eta_1)\,\{1 + \frac{3}{L}(\eta - \eta_1)\} \tag{E.31.1}$$

$$s_2(\eta) = (\eta - \eta_2)\,\{1 - \frac{3}{L}(\eta - \eta_2)\} \tag{E.31.2}$$

Using Eqs.(E.5.1,2) it is obtained

$$s_1(z) = Lz(1 + 3z) \qquad \text{(E.32.1)}$$

$$s_2(z) = L\,(z - 1)(4 - 3z) \qquad \text{(E.32.2)}$$

Finally, using Eqs.(E.22), (E.18) and (E.15.1) it is determined that

$$r_i''(\eta) = 12\,\{l_i'(\eta_i)\}^2 \qquad \text{(E.33.1)}$$

$$r_i'(\eta) = 12\,\{l_i'(\eta_i)\}^2(\eta - \eta_i) - 3l_i'(\eta_i) \qquad \text{(E.33.2)}$$

$$r_i(\eta) = 6\,\{l_i'(\eta_i)\}^2(\eta - \eta_i)^2 - 3l_i'(\eta_i)(\eta - \eta_i) + 1 \qquad \text{(E.33.3)}$$

after rearrangement

$$r_i(\eta) = 3(\eta - \eta_i)\{2(l_i'(\eta_i))^2(\eta - \eta_i) - l_i'(\eta_i)\} \qquad \text{(E.34)}$$

Substituting the subscripts and using Eqs.(E.28.1,2)

$$r_1(\eta) = 3(\eta-\eta_1)\{\frac{2}{L^2}(\eta-\eta_1) + \frac{1}{L}\} + 1 \qquad \text{(E.35.1)}$$

$$r_2(\eta) = 3(\eta-\eta_2)\{\frac{2}{L^2}(\eta-\eta_2) + \frac{1}{L}\} + 1 \qquad \text{(E.35.2)}$$

Using Eqs.(E.5.1,2) it can be obtained

$$r_1(z) = 3z\,(2z + 1) + 1 \qquad \text{(E.36.1)}$$

$$r_2(z) = 3(z-1)\,(2z-3) + 1 \qquad \text{(E.36.2)}$$

Recalling Eqs.(E.6.2) and (E.7.2), Eqs.(E.2.1,2,3) are substituted by the preceding Eqs.(E.36.1,2)

$$h_1(z) = r_1(z)\{l_1(z)\}^3$$

rearranging

$$h_1(z) = -6z^5 + 15z^4 - 10z^3 + 1 \qquad \text{(E.37.1)}$$

similarly substituting Eq.(E.36.2) and (E.7.2)

$$h_2(z) = 10z^3 - 15z^4 + 6z^5 \qquad \text{(E.37.2)}$$

Substituting Eqs.(E.32.1,2) and (E.6.2) into (E.2.2) one obtains

$$\bar{h}_1(z) = L(-3z^5 + 8z^4 - 6z^3 + z) \qquad (E.38.1)$$

Similarly

$$\bar{h}_2(z) = L(-3z^5 + 7z^4 - 4z^3) \qquad (E.38.2)$$

Substituting Eqs.(E.33.1,2) and (E.2.3) one obtains

$$\bar{\bar{h}}_1(z) = \frac{L^2}{2}(-z^5 + 3z^4 - 3z^3 + z^2) \qquad (E.39.1)$$

$$\bar{\bar{h}}_2(z) = \frac{L^2}{2}(z^5 - 2z^4 + z^3) \qquad (E.39.2)$$

Eqs.(E.37.1,2) correspond to Eqs.(4.26.1) and (4.26.4), respectively.

Eqs.(E.38.1,2) correspond to Eqs.(4.26.2) and (4.26.5), respectively.

Eqs.(E.39.1,2) correspond to Eqs.(4.26.3) and (4.26.6), respectively.

APPENDIX F

DETAILED DESCRIPTION OF THE INPUT OF THE COMPUTER CODE

This appendix explains the computer code's input in detail. All of the input variables are read by the subroutine INSOL.

Subroutine INSOL contains a dictionary of all computer code variables. The first "card" (or line) requires the following information:

EI = η_∞, specified upper limit of the solution domain.(Real)
8. is sufficient for all laminar flow applications. (After the transition to turbulent flow the code automatically augments the value of this variable, if needed(see Eq.(4.63))

PA = K, geometric grid parameter (Real)
1. for laminar flows
>1. for turbulent flow applications

PM = Print option parameter (Integer)
(see Table 5.1 for details)

NTURB= Index of the transition station where the flow becomes turbulent (Integer)
>NST for laminar flows (see next item)

NST = Total number of x-stations excluding the initial station (Integer)
0 for similar flows
n being the number of stations for other flows

IVPL = Initial velocity flag (Integer)

1 Linear velocity profile

2 Cubic velocity profile

3 Pohlhausen type velocity profile

IPR = Pressure gradient flag, which indicates the type of flow (Integer)

0 Similar flows

1 Howarth's flow (See Appendix G1)

2 Flow past a circular cylinder (See Appendix G2)

6 Case of Numerical Calculation (See Appendix G3)

The second "card" (or line) requires the following information:

RHO = ρ, fluid density in kg/m^3 (Real)

CMU = μ, dynamic viscosity of the fluid in kg/m s (Real)

UINF = U_∞, free stream velocity of the fluid in m/s (Real)

The third "card" (or line) conveys the following information:

TYP = Type of grid arrangement (Character)

'L' for laminar flows

'T' for turbulent flows

DETA(1)= h_1, initial η-grid length (See section 5.1.2) (real)

The fourth "card" (or line) contains only

BETA = β, pressure gradient parameter (Real)

The value for similar flows; $-0.19884 \leq \beta \leq 2$.

also for non-similar flows the initial value has to be input:

0 Howarth's flow

1. Flow past a circular cylinder

The "fifth" card set is omitted if NST = 0 .

If NST > 0 and IPR = 1 or 2 NST-number of cards have to be filled with the coordinates of x-stations. (Real)

If NST > 0 and IPR=6 NST-number of cards have to be filled with the coordinates of x-stations and potential flow solution velocities, respectively. (Pairwise)

The sixth (or final) "card" has

ITER = Maximum number of allowed iterations (Integer)

APPENDIX G

TEST CASES CONSIDERED FOR THE COMPUTER PACKAGE

G1 HOWARTH'S FLOW

The non-dimensional velocity field is given by

$$\bar{u} = 1 - a\bar{x} \tag{G.1}$$

where $\bar{u} = U_e/U_\infty$, $\bar{x} = x/L_r$ (G.2.1,2)

U_e : Potential flow, edge velocity

U_∞ : Free stream velocity

L_r : An arbitrary reference length

a : constant

If (G.2.1,2) are substituted into Eq.(4.3.1) one obtains

$$d\xi = \rho\mu U_\infty L_r (1 - a\bar{x})\, d\bar{x} \tag{G.3}$$

$$\xi = \rho\mu U_\infty L_r \bar{x} (1 - a\bar{x}/2) \tag{G.4}$$

If a non-dimensional ξ-coordinate is defined as

$$\bar{\xi} = \frac{\xi}{\rho\mu Lr\, U\infty} \tag{G.5}$$

Eq.(G.4) becomes

$$\bar{\xi} = \bar{x} (1 - \frac{a}{2} \bar{x}) \tag{G.6}$$

Recalling Eq.(4.11) and substituting the necessary values

$$\beta = \frac{2\xi}{Ue}\frac{dUe}{dx}\frac{d\bar{x}}{d\xi} \tag{G.7}$$

From Eq.(G.1) and (G.2.1)

$$U_e = U_\infty(1 - a\bar{x}) \tag{G.8}$$

Using Eqs.(G.3.4) and (G.8) and rearranging

$$\beta = -\frac{1}{4}\frac{x(1 - a\bar{x}/2)}{(1 - ax)^2} \tag{G.9}$$

Finally the dimensionless shear parameter will be evaluated:

From Eq.(2.2) and (B.3)

$$\tau_w \equiv \tau(0) = \frac{\mu\rho U_e^2}{\sqrt{2\xi}} f''(0) \tag{G.10}$$

Inserting Eq.(G.8) and (G.4) into Eq.(G.10)

$$\tau_w = \frac{\mu\rho U_\infty^2 (1 - a\bar{x})^2}{\sqrt{2\rho\mu U_\infty L_r\, x(1 - ax/2)}} f''(0)$$

Dimensionless shear parameter

$$\left[\frac{\tau_w}{\rho U_\infty^2}\right]\left[\frac{U_\infty L_r}{\nu}\right]^{1/2} = \frac{(1 - a\bar{x})^2}{\sqrt{2x(1 - ax/2)}} f''(0) \tag{G.11}$$

G2 FLOW PAST A CIRCULAR CYLINDER

Treatment of this flow is very similar to Howarth's flow. In this case the velocity filed is given as

$$U_e = 2U_\infty \sin \bar{x} \tag{G.12}$$

where, $x = \bar{x}/r_0$, r_0 being the radius of the cylinder (G.13)

$$d\xi = 2\rho\mu r_0 U_\infty \sin \bar{x}\, d\bar{x} \tag{G.14}$$

$$\xi = 2\rho\mu r_0 U_\infty (1 - \cos \bar{x}) \tag{G.15}$$

similarly after simplification of several terms pressure gradient becomes

$$\beta = \frac{2 \cos \bar{x}}{1 + \cos \bar{x}} \tag{G.16}$$

Analog to Eq.(G.10)

$$\tau_w = \frac{4\mu\rho U_\infty^2 \sin^2 \bar{x}}{\sqrt{4\mu\rho U_\infty r_0 (1 - \cos x)}} f''(0) \tag{G.17}$$

After simplification and rearrangement the dimensionless shear parameter for this flow becomes

$$\left[\frac{\tau_w}{\rho U_\infty^2}\right]\left[\frac{U_\infty r_0}{\nu}\right]^{1/2} = \frac{2 \sin \bar{x}}{\sqrt{1 - \cos x}} f''(0) \tag{G.18}$$

G3 NUMERICAL CALCULATION OF PARAMETERS

This case requires both the streamwise physical coordinates x_i and their respective potential flow velocities U_{ei} as input. By three-point-numerical-differentiation dU_{ei}/dx values are calculated. The following equations calculate the necessary parameters for the computation of the boundary layer field.

$$m = \frac{\bar{x}}{Ue} \frac{dUe}{dx} \tag{G.19}$$

and finally

$$\beta = \frac{2m}{m + 1} \tag{G.20}$$

Recalling Eq.(G.5) and (G.6)

$$d\xi = \bar{u}\, d\bar{x} \tag{G.21}$$

G4 DERIVATION OF THE SKIN-FRICTION-COEFFICIENT C_f OVER THE FLAT PLATE

For the case of the flat plate

$$U_e/U_\infty = 1 \qquad \text{(G.22)}$$

which yields

$$U_e = U_\infty \qquad \text{(G.22.1)}$$

Recalling Eq.(4.3.1) and substituting Eq.(G.22.1) and integrating

$$\xi = \rho\mu U_\infty x \qquad \text{(G.23)}$$

since $$C_f = \frac{\tau w}{\rho U_\infty^2/2} \qquad \text{(G.24)}$$

where $$\tau_w = \mu\rho\frac{U_e^2}{\sqrt{2\xi}} f''_w \qquad \text{(G.25)}$$

Substituting Eqs.(G.22) and (G.23) into (G.25)

$$\tau_w = \mu\rho\,\frac{U_\infty^2}{\sqrt{2\rho\mu U_\infty x}}\, f''_w \qquad \text{(G.25.1)}$$

Substituting Eq.(G.25.1) into (G.24)

$$C_f = \sqrt{2\,\frac{\nu}{xU_\infty}}\; f''_w$$

$$C_f = (Re_x/\ 2)^{-1/2}\; f''_w \qquad \text{(G.26)}$$

APPENDIX H

DERIVATION OF THREE-POINT-NUMERICAL-DIFFERENTIATION EQUATIONS WITH UNEQUALLY SPACED BASE POINTS

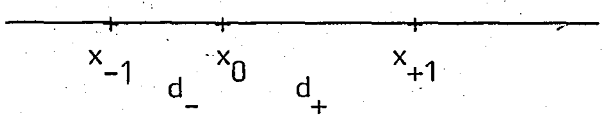


FIGURE H1

Mathematical relationship of the variables shown in Fig.H1 is as follows:

$$x_1 - x_0 = d_+ \qquad x_0 - x_{-1} = d_+ \tag{H.1.1,2}$$

Any function can be approximated by Lagrange Interpolation by the follwing equation:

$$y(x) = l_{-1}(x)\ f_{-1} + l_0(x)\ f_0 + l_1(x)\ f_1 \tag{H.2}$$

where l_{-1}, l_0 and l_1 are Lagrange Interpolation Polynomials given by

$$l_{-1}(x) = \frac{(x - x_0)(x - x_1)}{(x_{-1} - x_0)(x_{-1} - x_1)} \tag{H.3.1}$$

$$l_0(x) = \frac{(x - x_{-1})(x - x_1)}{(x_0 - x_{-1})(x_0 - x_1)} \tag{H.3.2}$$

$$l_1(x) = \frac{(x - x_{-1})(x - x_0)}{(x_1 - x_{-1})(x_1 - x_0)} \tag{H.3.3}$$

f_{-1}, f_0 and f_1 are the values of the function to be approximated at the given nodes. Using relations Eq.(H.1.1,2) Eqs.(H.3.1,2,3) can be modified as:

$$l_{-1}(x) = \frac{(x - x_0)(x - x_1)}{d_- (d_+ + d_-)} \tag{H.4.1}$$

$$l_0(x) = \frac{(x - x_{-1})(x - x_1)}{- d_- d_+} \tag{H.4.2}$$

$$l_1(x) = \frac{(x - x_{-1})(x - x_0)}{d_+ (d_+ + d_-)} \tag{H.4.3}$$

Differentiating Eq.(H.2) one obtains

$$y'(x) = l'_{-1}(x)\ f_{-1} + l'_0(x)\ f_0 + l'_1(x)\ f_1 \tag{H.5}$$

where $$l'_{-1}(x) = \frac{2x - x_0 - x_1}{d_-(d_+ + d_+)} \tag{H.6.1}$$

$$l'_0(x) = \frac{2x - x_{-1} - x_1}{- d_- d_+} \tag{H.6.2}$$

$$l'_1(x) = \frac{2x - x_{-1} - x_0}{d_+(d_+ + d_-)} \tag{H.6.3}$$

The numerical values of the derivatives can be given as:

$$f'_{-1} = l'_{-1}(x_{-1})f_{-1} + l'_0(x_{-1})f_0 + l'_1(x_{-1})f_1 \tag{H.7.1}$$

$$f'_0 = l'_{-1}(x_0)f_{-1} + l'_0(x_0)f_0 + l'_1(x_0)f_1 \tag{H.7.2}$$

$$f'_1 = l'_{-1}(x_1)f_{-1} + l'_0(x_1)f_0 + l'_1(x_1)f_1 \tag{H.7.3}$$

If $l'(x_i)$ are evaluated the equations are obtained as:

$$l'_{-1}(x_{-1}) = \frac{-(2d_- + d_+)}{d_-(d_- + d_+)} \qquad l'_{-1}(x_0) = \frac{-d_+}{d_-(d_- + d_+)} \tag{H.8.1,2}$$

$$l'_{-1}(x_1) = \frac{d_+}{d_-(d_- + d_+)} \tag{H.8.3}$$

$$l_0'(x_{-1}) = \frac{d_+ + d_-}{d_+ d_-} \qquad l_0'(x_0) = \frac{d_+ + d_-}{d_+ d_-} \tag{H.9.1,2}$$

$$l_0'(x_1) = -\frac{d_+ + d_-}{d_+ d_-} \tag{H.9.3}$$

$$l_1'(x_{-1}) = -\frac{d_-}{d_+(d_+ + d_-)} \qquad l_1'(x_0) = \frac{d_-}{d_+(d_+ + d_-)} \tag{H.10.1,2}$$

$$l_1'(x_1) = \frac{2d_+ + d_-}{d_+(d_+ + d_-)} \tag{H.10.3}$$

Substituting Eqs.(H.8.1, 9.1, 10.1) into Eq.(H.7.1)

$$f'_{-1} = \{-d_+(2d_-+d_+)f_{-1}+(d_++d_-)^2 f_0 - d_-^2 f\}/\{d_+d_-(d_++d_-)\} \tag{H.11.1}$$

$$f'_0 = \{-d_+^2 f_{-1}+(d_+^2-d_-^2)f_0+d_-^2 f_1\}/\{d_+d_-(d_++d_-)\} \tag{H.11.2}$$

$$f'_1 = \{d_-^2 f_{-1}-(d_++d_-)^2 f_0+d_-(2d_++d_-)f_1\}/\{d_+d_-(d_++d_-)\} \tag{H.11.3}$$

APPENDIX I

LISTING OF THE COMPUTER CODE

R-B L A Y E R

AND

R B L E L E

```
      PROGRAM ASLI(INPUT,OUTPUT)
C
C
C   MASTER THESIS  OF SAHNUR  AGAIK      1 9 8 1  -  1 9 8 4
C
C
C        A  LIST  VARIABLES  AND  THEIR  BRIEF  EXPLANATION
C              CAN  BE FOUND  IN SUBROUTINE  INSOL
C
C
      PARAMETER (NLC=5,NUC=5)
      COMMON/RIGHTS/F(6),FB(6),FC(6)
      COMMON/GAUI/G(6,7),GP(6,7),GPP(6,7),GPPP(6,7),W(7),Z(7)
      COMMON/GR2/F0(51),F1(51),F2(51),B0(51),B1(51),B2(51)
     1,F3(51)
      COMMON/ZETC/IT,ITER,NST,IZSC,ZETA(101),XBAR(101),
     1UE(101),SPRM
      COMMON/GR3/CR(51),DETA(153)
      COMMON/GR4/N,ET,PA,PM,NTURB,NE,NV,IVPL,IPK
      COMMON/GR6/Q(5)
      COMMON/GR5/ALPHA,BETA

      COMMON/PHYS/CMU,RHO,UINF
      COMMON/EDI/EPS(51),EPSP(51)
      COMMON/GR1/GSM(154,NLC+NUC+1),SM(6,6),SOL(154)
      EXTERNAL MATMUL
      INTEGER PM,P,P1,P0
      CHARACTER*1 IPLOT
      CHARACTER*1 TYP
      DIMENSION XL(154*(NLC+1))
      COMMON/TYPE/TYP,IPLOT
C
C
C     SUBROUTINE INPUT
C
C
      CALL INSOL
C
C
C
C     ALL F AND RESP. DERIV  VECTORS WILL BE CREATED
C
      CALL VELPRF
C
C
C     MAIN EXECUTION ROUTINE
C
C
C     ALL ELEMENTS WILL BE CREATED
C
      ZETA(1)=0.
      IZSC=1
      ALPHA=0.
      GO TO 45
   47 CALL PRGR
      IF((IT .EQ. 0 ).AND.( ABS(F2(NE+1)).GT..00010).AND.(TYP.EQ.
     1'T'))CALL GROW
  101 ALPHA=(ZETA(IZSC)+ZETA(IZSC-1))/(ZETA(IZSC)-ZETA(IZSC-1))
      IF(IZSC.LT.NTURB)GO TO 45
   49 CALL EDYVSC
   45 DO 4 J=1,NE
C
C     ALL ELEMENTS WILL BE CREATED
C
      F(1)=F0(J)
      F(2)=F1(J)
      F(3)=F2(J)
      F(4)=F0(J+1)
      F(5)=F1(J+1)
      F(6)=F2(J+1)
      FB(1)=B0(J)
      FB(2)=B1(J)
      FB(3)=B2(J)
      FB(4)=B0(J+1)
      FB(5)=B1(J+1)
      FB(6)=B2(J+1)
      ET=DETA(J)
      E2=ET**2
      E3=ET**3
      CALL RHS(ET,J)
```

```
      C
      C     VERSION RBLELE CALLS ELECRE
      C
      C     CALL ELECRE(J)
            IF(PM.NE.1)GOTO 690
            PRINT 9
      9     FORMAT(//,5X,'STIFFNESS MATRIX  &  FORCE VECTOR      ')
            PRINT 10,((SM(I,JX),JX=1,6),FC(I),I=1,6)
      10    FORMAT(//,5X,6G15.7,10X,G15.7)
      690   KS=(J-1)*3+1
            KE=KS+5
            LS=KS
            LE=KE
      C
      C     GLOBAL MATRIX WILL BE LOADED
      C
            DO 11 K=KS,KE
            K1=K-KS+1
            DO 12 L=LS,LE
            L1=L-LS+1
            LB=(L-K)+(NLC)+1
            GSM(K,LB)=GSM(K,LB)+SM(K1,L1)
      12    CONTINUE
      C
      C     GLOBAL-FORCE-VECTOR-WILL-BE-LOADEDD
      C
            SOL(K)=SOL(K)+FC(K1)
      11    CONTINUE
      C
      C     END OF MAIN EXECUTION ROUTINE
      C
      4     CONTINUE
            IF(PM.NE.1)GOTO 799
            PRINT 13
      13    FORMAT(//,5X,'GLOBAL STIFF. MATR. &  GLOBAL  FORCE VECTOR')
            DO 37 I=1,NV
            PRINT 38,I
      38    FORMAT(/,10X,' R O W   #  = ',I3)
            PRINT 14,(GSM (I,J),J=1,NLC+NUC+1),SOL(I)
      14    FORMAT(2X,10G13.7)
      37    CONTINUE
      C
      C     MODIFICATION PHASE
      C     TO ENFORCE THE BOUNDARY CONDITIONS
      C
       799  CALL MODIFC
            IF(PM.NE.1)GOTO 840
            PRINT 18
      18    FORMAT(//,5X,'  MODIFIED GLOBAL STIFFNESS MATRIX   ')
            DO 39 I=1,NV
            PRINT 38,I
            PRINT 14,(GSM (I,J),J=1,NLC+NUC+1),SOL(I)
      39    CONTINUE
      C
      C
      C     SOLUTION BY LEQT1B
      C
      C
        840 CONTINUE
            CALL LEQT1B(GSM,NV,NLC,NUC,154,SOL,1,154,0,XL,IER)
            CFL=SOL(3)
            DO 23 K=NV,1,-1
            IF((PM.NE.1).AND.(K.NE.3))GO TO 23
            PRINT 22,K,SOL(K)
         22 FORMAT(/,5X,I4,2X,'TH VARIABLE SOLUTION',5X,G15.7)
      23    CONTINUE
            DO 35 JS=1,NV-2,3
            JT=(JS/3)+1
            F0(JT)=F0(JT)+   SOL(JS)
            F1(JT)=F1(JT)+   SOL(JS+1)
            F2(JT)=F2(JT)+   SOL(JS+2)
      35    CONTINUE
            IT=IT+1
            IF(PM.LT.3)GO TO 171
            CALL OUTSOL
        171 IF(IPLOT.EQ.'Y') THEN
            PRINT *,'  PLOT ROUTINE IS CALLED'
            CALL PLOTST(CR,F0,NE+1)
            END IF
            DO 153 I=1,NV
```

```
      DO 80 J=1,NLC+NUC+1

 80   GSM(I,J)=0.
 153  SOL(I)=0.
      IF( ABS(CFL    ).LE.0.001000)GO TO 163
      IF(IT.GT.ITER)GOTO167
      PRINT*, '    I Z S C  =========  ',IZSC
      IF(IZSC.GE.NTURB)GO TO 47
      GO TO 45
 163  SPRM=SPRM*F2(1)
      CALL OUTSOL
      IZSC=IZSC+1
      DO 165 II=1,NE+1
      B0(II)=F0(II)
      B1(II)=F1(II)
      B2(II)=F2(II)
 165  CONTINUE
      IT=0
      IF(IZSC.LE.(NST+1))GO TO 47
      GO TO 169
 167  PRINT*,'       NO    CONVERGENCE  IS ACHIEVED      '
      GO TO 169
 169  STOP
      END
C
C     SUBROUTINE TO CREATE THE INITIAL VELOCITY PROFILES
C
      SUBROUTINE VELPRF
      COMMON/GR3/CR(51),DETA(153)
      COMMON/GR2/F0(51),F1(51),F2(51),B0(51),B1(51),B2(51)
     1 ,F3(51)
      COMMON/GR4/N,EI,PA,PM,NTURB,NE,NV,IVPL,IPK
      COMMON/GR5/ALPHA,BETA
      INTEGER PM
      IF(IVPL.EQ.2)GOTO900
      IF(IVPL.EQ.3)GOTO920
C
C     LINEAR VELOCITY PROFILE
C
      DO 5 IE=1,NE+1
      W=CR(IE)/EI
      F1(IE)=W
      F2(IE)=1./EI
      F0(IE)=W**2*EI/2.
    5 CONTINUE
      GOTO910
C
C     THIRD ORDER VELOCITY PROFILE
C
 900  DO 2 JE=1,NE+1
      W=CR(JE)/EI
      F1(JE)=(W/2.)*(3.-W**2)
      F0(JE)=(EI*W**2/8.)*(6.-W**2)
      F2(JE)=(1.5/EI)*(1.-W**2)
      F3(JE)=-3*W/(EI**2)
 2    CONTINUE
      GO TO 910
C
C     POHLHAUSEN TYPE VELOCITY PROFILE
C
C     CEBECI  A SMITH (1974) P.300 (8.2.5.B)
C
 920  B=BETA*EI**2/6.
      DO 8 KE=1,NE+1
      W=CR(KE)/EI
      F0(KE)=(((0.2*(1.-B)*W+0.25*(3.*B-2.))*W-B)*W+0.5*(B+2.))
     1*W*W*EI
      F1(KE)=((((1.-B)*W+(3.*B-2.))*W-3.*B)*W+(B+2.))*W
      F2(KE)=(((4.*(1.-B)*W+3.*(3.*B-2.))*W-6.*B)*W+B+2.)/EI
    8 CONTINUE
 910  DO 9 IB=1,NE+1
      B0(IB)=F0(IB)
      B1(IB)=F1(IB)
    9 B2(IB)=F2(IB)
      PRINT 33
      DO 889 JX=1,NE+1
      PRINT 34,JX,F0(JX),F1(JX),F2(JX)
 889  CONTINUE
  33  FORMAT(//,10X,'INITIAL GUESS OF VELPRF',//,5X,'NODE ',3X,
     D'STF',13X,'DSTF',12X,'DDSTF',/,1X,50('_'),/)
```

```
   34 FORMAT(/,5X,I2,7X,G12.6,5X,G12.6,3X,G12.6)
      RETURN
      END
C
C     SUBROUTINE TO CALCULATE THE RIGHT HAND SIDE VECTOR
C
      SUBROUTINE RHS(ET,JEL)
      PARAMETER (NLC=5,NUC=5)
C
C     THIS SUBROUTINE CALCULATES Q4 AND COMPLETE RHS
C
      COMMON/GR1/GSM(154,NLC+NUC+1),SM(6,6),SOL(154)
      COMMON/GAUI/G(6,7),GP(6,7),GPP(6,7),GPPP(6,7),W(7),Z(7)
      COMMON/RIGHTS/F(6),FB(6),FC(6)
      COMMON/GR6/Q(5)
      COMMON/GR5/ALPHA,BETA
      COMMON/EDI/EPS(51),EPSP(51)
      DIMENSION COEFF(7)
      DIMENSION GAM1(6,6),GAM2(6,6),GAM3(6,6),GAM4(6,6),GAM5(6,6),
     1GAM6(6,6),GAM7(6,6),GAMT1(6,6),GAMT2(6,6),GAMT3(6,6),GAMT4
     2(6,6),GAMT5(6,6),GAMT6(6,6),GAMT7(6,6)
      DIMENSION H(6,7),HP(6,7),HPP(6,7),HPPP(6,7),Q4(6)
      E2=ET**2
      E3=ET**3
      DO 1 I=1,7
      H(1,I)=G(1,I)
      HP(1,I)=GP(1,I)
      HPP(1,I)=GPP(1,I)
      HPPP(1,I)=GPPP(1,I)
      H(2,I)=ET*G(2,I)
      HP(2,I)=ET*GP(2,I)
      HPP(2,I)=ET*GPP(2,I)
      HPPP(2,I)=ET*GPPP(2,I)
      H(3,I)=E2*G(3,I)
      HP(3,I)=E2*GP(3,I)
      HPP(3,I)=E2*GPP(3,I)
      HPPP(3,I)=E2*GPPP(3,I)
      H(4,I)=G(4,I)
      HP(4,I)=GP(4,I)
      HPP(4,I)=GPP(4,I)
      HPPP(4,I)=GPPP(4,I)
      H(5,I)=ET*G(5,I)
      HP(5,I)=ET*GP(5,I)
      HPP(5,I)=ET*GPP(5,I)
      HPPP(5,I)=ET*GPPP(5,I)
      H(6,I)=E2*G(6,I)
      HP(6,I)=E2*GP(6,I)
      HPP(6,I)=E2*GPP(6,I)
      HPPP(6,I)=E2*GPPP(6,I)
 1    CONTINUE
      EPL=(EPS(JEL)+EPS(JEL+1))/2.
      EPLP=(EPSP(JEL)+EPSP(JEL+1))/2.
      DO 195 IC=1,6
C     GO TO 20
      DO 195 JC=1,6
  195 SM(IC,JC)=0.
C
C     K   FOR POINTS  OF GAUSSIAN INTEGRATION
C
      DO 200 K=1,7
      DO 205 I=1,6
      DO 210 J=1,6
      GAM1(I,J)=H(I,K)*HPPP(J,K)/E3
      GAM2(I,J)=H(I,K)*HPP(J,K)/E2
      GAMT1(I,J)=H(I,K)*H(J,K)
      GAMT2(I,J)=FB(I)*HPP(J,K)/E2
      GAMT3(I,J)=F(I)*HPP(J,K)/E2
      GAMT4(I,J)=H(I,K)*HP(J,K)/ET
      GAMT5(I,J)=FB(I)*HP(J,K)/ET
      GAMT6(I,J)=F(I)*HP(J,K)/ET
      GAMT7(I,J)=F(I)*H(J,K)
  210 CONTINUE
  205 CONTINUE
      CALL MATMUL(GAMT1,GAMT2,GAM3)
      CALL MATMUL(GAMT1,GAMT3,GAM4)
      CALL MATMUL(GAMT4,GAMT5,GAM5)
      CALL MATMUL(GAMT4,GAMT6,GAM6)
      CALL MATMUL(GAM2,GAMT7,GAM7)
      DO 215 IL=1,6
      DO 220 JL=1,6
```

```
      COF=((1.+EPL)*GAM1(IL,JL)+EPLP*GAM2(IL,JL)-
     1ALPHA*GAM3(IL,JL)+(1.+ALPHA)*GAM4(IL,JL)+ALPHA*GAM5(IL,JL)-
     22.*(BETA+ALPHA)*GAM6(IL,JL)+(1.+ALPHA)*GAM7(IL,JL))*W(K)*
     3ET/2.
      SM(IL,JL)=SM(IL,JL)+COF
  220 CONTINUE
  215 CONTINUE
C
C-----END OF GAUSSIAN INTGRAT-----------------------------------
C
  200 CONTINUE
   20 DO 2 K=1,6
      Q4(K)=0.
      DO 3 I=1,7
      CO=0.
      CO1=0.0
      CO2=0.
      CO3=0.
      CO4=0.
      CO5=0.
      CO6=0.
      CO7=0.
      DO 4 J=1,6
      CO=CO+HPPP(J,I)*F(J)*(1.+EPL)/(ET**3)
      CO1=CO1+H(J,I)*FB(J)*ALPHA
      CO2=CO2+HPP(J,I)*F(J)/(ET**2)
      CO3=CO3+H(J,I)*F(J)
      CO5=CO5+HP(J,I)*F(J)/ET
      CO6=CO6+HP(J,I)*FB(J)/ET
4     CONTINUE
      COEFF(I)=(CO+(EPLP-CO1)*CO2+(1.+ALPHA)*CO3*CO2
     &-(ALPHA+BETA)*CO5**2+ALPHA*CO6*CO5+BETA)*W(I)
      Q4(K)=Q4(K)-COEFF(I)*H(K,I)
3     CONTINUE
      Q4(K)=Q4(K)*ET/2.
2     CONTINUE
      Q(1)=1.
      Q(2)=(1.+ALPHA)*CO3-CO1
      Q(4)=(1.+ALPHA)*CO2
      Q(3)=ALPHA*CO6-2.*(ALPHA+BETA)*CO5
C
C     PRINT *,'Q  VALUES IN RHS ARE',Q
C
      DO 5 JQ=1,6
5     FC(JQ)=Q4(JQ)
      RETURN
      END
C
C     SUBROUTINE TO CALCULATE THE EDDY VISCOSITY COEFFICIENTS
C
      SUBROUTINE EDYVSC
      COMMON/PHYS/CMU,RHO,UINF
      COMMON/EDI/EPS(51),EPSP(51)
      COMMON/GR4/N,EI,PA,PM,NTURB,NE,NV,IVPL,IPR
      COMMON/ZETC/IT,ITER,NST,IZSC,ZETA(101),XBAR(101),
     1UE(101),SPRM
      COMMON/GR5/ALPHA,BETA
      COMMON/GR3/CR(51),DETA(153)
      COMMON/GR2/F0(51),F1(51),F2(51),B0(51),B1(51),B2(51)
     1,F3(51)
      DIMENSION EPO(51),EPI(51)
      SINT=0.
      DO 299 IELEM=1,NE
      HG=DETA(IELEM)
      VER=(2.-F1(IELEM+1)-F1(IELEM))/2.
  299 SINT=SINT+HG*VER
      SINT=0.0168*ABS(SINT)
      XSI=SQRT(2.*ZETA(IZSC))/CMU
      DO 301 IPO=1,NE+1
      DEN=1.+5.5*(CR(IPO)/EI)**6
  301 EPO(IPO)=SINT*XSI/DEN
      DO 319 IPO=1,NE+1
      APL=XSI*CR(IPO)/26.
      PRX=SQRT(ABS((F2(1)-BETA*CR(IPO))/XSI))
      APX=-APL*PRX
      APW=(1.-EXP(APX))**2
      EPI(IPO)=0.16*XSI*ABS(F2(IPO))*APW*CR(IPO)**2
      IF(EPI(IPO).GT.EPO(IPO))GO TO 320
      EPS(IPO)=EPI(IPO)
  319 CONTINUE
```

```
  320 DO 325 ICON=IPO,NE+1
  325 EPS(ICON)=EPO(ICON)
      DO 339 IJO=2,NE
      HM=DETA(IJO-1)
      HP=DETA(IJO)
      FM1=EPS(IJO-1)
      FZ=EPS(IJO)
      FP1=EPS(IJO+1)
      DENM=HM*HP*(HM+HP)
      AM1=-HP**2
      AZ=(HP+HM)*(HP-HM)
      AP1=HM**2
  339 EPSP(IJO)=(AM1*FM1+AZ*FZ+AP1*FP1)/DENM
      HM=DETA(1)
      HP=DETA(2)
      FM1=EPS(1)
      FZ=EPS(2)
      FP1=EPS(3)
      DENM=HM*HP*(HM+HP)
      AM1=-HP*(2.*HM+HP)
      AZ=(HP+HM)**2
      AP1=-HM**2
      EPSP(1)=(AM1*FM1+AZ*FZ+AP1*FP1)/DENM
      HM=DETA(NE-1)
      HP=DETA(NE)
      FM1=EPS(NE-1)
      FZ=EPS(NE)
      FP1=EPS(NE+1)
      DENM=HM*HP*(HM+HP)
      AM1=HP**2
      AZ=-(HP+HM)**2
      AP1=HM*(2.*HP+HM)
      EPSP(NE+1)=(AM1*FM1+AZ*FZ+AP1*FP1)/DENM
      RETURN
      END
C
C     SUBROUTINE TO ENFORCE THE BOUNDARY CONDITIONS
C
      SUBROUTINE MODIFC
      PARAMETER (NLC=5,NUC=5)
      COMMON/GR4/N,EI,PA,PM,NTURB,NE,NV,IVPL,IPR
      COMMON/GR1/GSM(154,NLC+NUC+1),SM(6,6),SOL(154)
      SOL(1)=0.
C
C     FOLLOWING BOUNDARY CONDITIONS WITH 'C' ARE FOR THE MATHEMATICAL TE
C
C     SOL(1)=1.
      SOL(2)=0.
      SOL(NV-1)=0.
C     SOL(NV-1)=COS(EI)-SIN(EI)
C     DO 18 JX=2,NV-2
C     SOL(JX)=SOL(JX)-GSM(JX,0)*SOL(0)-GSM(JX,1)*SOL(1)-GSM(JX,NV-1)
C    &*SOL(NV-1)
C 18  CONTINUE
C     SOL(NV)=SOL(NV)-GSM(NV,0)*SOL(0)-GSM(NV,1)*SOL(1)-GSM(NV,NV-1)
C    &*SOL(NV-1)
      GSM(1,NLC+1)=1.
      GSM(2,NLC+1)=1.
      GSM(NV-1,NLC+1)=1.
      GSM(NV,NLC)=0.
      GSM(NV-1,NLC+2)=0.
      DO 15 JX=2,NLC+1
      GSM(JX,-JX+NLC+2)=0.
      GSM(1,JX+NLC)=0.
 15   CONTINUE
      DO 16 JX=3,NLC+1
      GSM(JX,(2-JX)+(NLC+1))=0.
      GSM(2,(JX-2)+(NLC+1))=0.
 16   CONTINUE
      DO 17 JX=NV-5,NV-2
      GSM(JX,(NV-1-JX)+(NLC+1))=0.
      GSM(NV-1,JX-NV+NLC+2)=0.
 17   CONTINUE
      RETURN
      END
C
C     SUBROUTINE TO PRINT THE OUTPUT OF THE PROGRAM
C
      SUBROUTINE OUTSOL
      COMMON/GR5/ALPHA,BETA
```

```
      COMMON/GR2/STF(51),DSTF(51),DDSTF(51),B0(51),B1(51)
     1,B2(51),F3(51)
      COMMON/GR3/CR(51),DETA(153)
      COMMON/PHYS/CMU,RHO,UINF
      COMMON/GR4/N,EI,PA,PM,NTURB,NE,NV,IVPL,IPR
      COMMON/ZETC/IT,ITER,NST,IZSC,ZETA(101),XBAR(101),
     1UE(101),SPRM
      INTEGER PM,P,P1,P0
      CHARACTER*1 TYP
      CHARACTER*1 IPLOT
      COMMON/TYPE/TYP,IPLOT
      REX=UINF*RHO*XBAR(IZSC)/CMU
      PRINT 95,XBAR(IZSC),IT,BETA,UE(IZSC),SPRM,REX
      DO 1 I=1,NE+1
      PRINT 110,    I,CR(I),STF(I),DSTF(I),DDSTF(I)
    1 CONTINUE
      IF((TYP.EQ.'T').AND.(IZSC.GE.2))CF=SQRT(2./REX)*
     1DDSTF(1)
      PRINT*,' WALL SHEAR   CF = ',CF
      RETURN
   95 FORMAT(1H1,/,5X,' STATION  XBAR =  ',F6.3,6X,'IT=',I4,3X,'BET=',
     1F10.8,2X,'UE =',F8.5,2X,' SHEAR PARAM= ',F11.8,3X,' REY  = ',
     0F15.4,////,2X,'NODE',6X,'ETA',8X,'STF',7X,'DSTF',6X,'DDSTF',
     1 /,1X,50('-'),/)
  110 FORMAT(2X,I4,4F11.6)
      END
C
C     SUBROUTINE TO PLOT THE OUTPUT OF THE PROGRAM
C
      SUBROUTINE PLOTST(XARY,YARY,NPLOT)
      DIMENSION XARY(51),YARY(51)
      CHARACTER*1 IUNI
      PRINT *,'DO YOU WANT A HARD COPY FILE,NPFILE'
      READ *,IUNI
      CALL INITIG(.TRUE.,.TRUE.,3HSTF)
      CALL SPLIM(0.,-3.,10.,3.)
      CALL SPPORT(0.,-3.,10.,3.)
      IF(IUNI.EQ.'Y') THEN
      CALL UNION
      END IF
      CALL SMSYM(2)
      CALL MOVEA(0.,-3.)
      CALL DRAWA(0.,3.)
      CALL MOVEA(0.,-3.)
      CALL DRAWA(10.,-3.)
      CALL MOVEA(0.,0.)
      CALL DRAWA(10.,0.)
      CALL MOVEA(0.,3.)
      CALL DRAWA(10.,3.)
      CALL DRAWA(10.,-3)
      CALL MOVEA(0.,-3.)
      CALL PLOTA(NPLOT,XARY,YARY,.TRUE.)
      IF(IUNI.EQ.'Y') THEN
      CALL UNIOFF
      END IF
      CALL AWTKEY(1,ITRIG,1,NCHAR,ICHAR)
      CALL CLRPT
      CALL QUITIG(.TRUE.)
      RETURN
      END
C
C     SUBROUTINE TO INPUT THE DATA
C
      SUBROUTINE INSOL
C
C
C     MASTER THESIS  OF SAHNUR  AGAIK
C
C
C  NE = NUMBER OF ELEMENTS
C  EI = ETA INFINITY
C  PA = GRID INCREASE PARAMETER (FOR TURBULENT CASE ONLY)
C  PM = PRINTING OPTION PARAMETER
C     = 1   PRINT ALL STATEMENTS
C     = 2          ONLY AFTER ELIMINATION
C     = 0          NO INFORMATION
C     = 3        ITERATIVE COMPLETE STATION INFORMATION
C IVPL= INITIAL VELOCITY PROFILE
C        2   NORMAL 3RD ORDER POHL
C        3   COMPLEX POHL 4TH ORDER
```

```
C       OTHERWISE  LINEAR
C IPR = PRESSURE GRADIENT FLAG
C         0   BETA=CONST. FLOW FOR TURB APPLICATION
C         1   HOWARTH'S  FLOW  UBAR=1.-XBAR/8.
C         2   CIRC. CYLINDER   UBAR=2.*SIN(XBAR)
C         6   FREE NUMERICAL
C  NV = NUMBER OF FINAL VARIABLES
C  GSM(NV,NLC+NUC+1)=GLOBAL STIFFNESS MATRIX
C SM(5,5)=  STIFFNESS MATRIX
C    Q(5)= Q(I) VALUES TO BE MULTIPLIED
C   FC(6)=FC    FORCE VECTOR
C  SOL(6)=GLOBAL FORCE VECTOR
C  SOL(6)=SOLUTION  FORCE VECTOR
C F0(NE+1)= F VECTOR
C F1(NE+1)= F'    VECTOR
C F2(NE+1)= F''   VECTOR
C F3(NE+1)= F'''  VECTOR
C B0(NE+1)= F   N-1   VECTOR
C B1(NE+1)= F'  N-1   VECTOR
C CR(NE+1)= COORDINATE VECTOR OF ETA     0 TO INFINITY
C ET      = DELTA ETA
C BETA    = BETA PARAMETER
C ZE      = ZETA PARAMETER
C NST     = N  OF STATIONS             0 FOR  SIMILAR FLOW
C IZSC    = ZETA STATION COUNTER
C ALPHA   = ALPHA  PARAMETER
C X0(NE)  = F      VECTOR AVG FOR ELEMENT
C X1(NE)  = F'     VECTOR AVG FOR ELEMENT
C X2(NE)  = F''    VECTOR AVG FOR ELEMENT
C X3(NE)  = F'''   VECTOR AVG FOR ELEMENT
C Y0(NE)  = F   N-1      VECTOR AVG FOR ELEMENT
C Y1(NE)  = F'  N-1      VECTOR AVG FOR ELEMENT
C DETA(NE)= INCREMENTS OF ELEMENTS
C DETA(1) = INITIAL GRID WHICH MUST BE GIVEN
C TYP     = GRID MESH ACCORDING TO TURB 'T' OR 'L' LAMINAR
      COMMON/ZETC/IT,ITER,NST,IZSC,ZETA(101),XBAR(101),
     1UE(101),SPRM
      COMMON/GAUI/G(6,7),GP(6,7),GPP(6,7),GPPP(6,7),W(7),Z(7)
      COMMON/GR3/CR(51),DETA(153)
      COMMON/GR4/N,EI,PA,PM,NTURB,NE,NV,IVPL,IPR
      COMMON/PHYS/CMU,RHO,UINF
      COMMON/GR5/ALPHA,BETA
      INTEGER PM,P,P1,P0
      CHARACTER*1 IPLOT
      CHARACTER*1 TYP
      COMMON/TYPE/TYP,IPLOT
      DATA Z/.025446044,.129234408,.297077425,.5,.702922576,.
     1.870765593,.974553956/
      DATA W/.129484966,.279705391,.381830051,.417959184,
     1.381830051,.279705391,.129484966/
      DATA RAD/0.0174532925/
      PRINT *,' DO YOU WANT A PLOT OF F VERSUS ETA'
      READ *,IPLOT
      PRINT*,' ENTER     EI,PA,PM,NTURB,NST,IVPL,IPR '
      READ*,EI,PA,PM,NTURB,NST,IVPL,IPR
      PRINT*,'  ENTER  RHO  AND  M  AND UINF'
      READ*,RHO,CMU,UINF
      CR(1)=0.
C
C     NODE COORDINATES FOR TURBULENT OR LAMINAR CASE
C
27    PRINT*,' ENTER  TYP T ,  L OR M; DETA(1)  '
      READ*,TYP,DETA(1)
      CR(2)=DETA(1)
      IF((TYP.EQ.'T').OR.(TYP.EQ.'M'))GO TO 26
      IF(TYP.NE.'L')GOTO 27
C
C     LAMINAR  CASE ASSUMED
C
      NE=INT(EI/DETA(1)+0.5)
      DO 29 JD=2,NE
      DETA(JD)=DETA(1)
      CR(JD+1)=CR(JD)+DETA(1)
   36 FORMAT(5X,I2,10X,F10.3)
   29 CONTINUE
      GOTO 30
C
C     TURBULENT CASE ASSUMED
C
26    NE=ALOG((EI/DETA(1))*(PA-1.0)+1.0)/ALOG(PA)
```

```
      IF(TYP.EQ.'M')GO TO 40
      DO 31 JD=2,NE
      DETA(JD)=DETA(JD-1)*PA
      CR(JD+1)=CR(JD)+DETA(JD)
   31 CONTINUE
      EI=CR(NE+1)
      GO TO 30
   40 NE=NE*3
      PAR=DETA(1)/3.
      CR(2)=CR(1)+PAR
      CR(3)=CR(2)+PAR
      CR(4)=CR(3)+PAR
      DETA(2)=DETA(1)
      DETA(3)=DETA(1)
      DO 45 IM=4,NE-2,3
      PAR=PAR*PA
      DO 47 IN=1,3
      CR(IM+IN)=CR(IM+IN-1)+PAR
      DETA(IM+IN-1)=PAR
   47 PRINT*, IM+IN, 'COOR ',CR(IM+IN)
   45 CONTINUE
      EI=CR(NE+1)
   30 NV=NE*3+3
      PRINT*,'  ENTER       BETA       '
      READ*,   BETA
      IF(NST.EQ.0)GO TO 48
      IF(IPR.GE.6)GO TO 60
      ZETA(1)=0.
      XBAR(1)=0.
      DO 49 JX=2,NST+1
      READ*,XBAR(JX)
   49 IF(IPR.EQ.2)XBAR(JX)=XBAR(JX)*RAD
      GO TO 48
   60 UE(1)=1.
      DO 70 JS=2,NST+1
   70 READ*,XBAR(JS),UE(JS)
   48 IT=0
      PRINT *,'   ENTER  NO.OF ITERATIONS , ITER'
      READ *, ITER
      PRINT 32,NE,NV,EI,BETA,TYP,NST
32    FORMAT(/,5X,'  IMPORTANT PARAMETERS ARE:    ',/,5X,'  NE=',I6,
     &'  NV=',I6,'  EI=',F8.3,'  BETA=',F8.5,'  TYP=',A3,3X,'NST=',I3)
C
C     VALUES OF GAUSSIAN INTEGRATION
C
      DO 1 I=1,7
      G(1,I)=1.-10.*Z(I)**3+15.*Z(I)**4-6.*Z(I)**5
      GP(1,I)=30.*(-Z(I)**2+2.*Z(I)**3-Z(I)**4)
      GPP(1,I)=60.*(-Z(I)+3.*Z(I)**2-2.*Z(I)**3)
      GPPP(1,I)=60.*(-1.+6.*Z(I)-6.*Z(I)**2)
      G(2,I)=Z(I)-6.*Z(I)**3+8.*Z(I)**4-3.*Z(I)**5
      GP(2,I)=1.-18.*Z(I)**2+32.*Z(I)**3-15.*Z(I)**4
      GPP(2,I)=12.*(-3.*Z(I)+8.*Z(I)**2-5.*Z(I)**3)
      GPPP(2,I)=12.*(-3.+16.*Z(I)-15.*Z(I)**2)
      G(3,I)=0.5*(Z(I)**2-3.*Z(I)**3+3.*Z(I)**4-Z(I)**5)
      GP(3,I)=0.5*(2.*Z(I)-9.*Z(I)**2+12.*Z(I)**3-5.*Z(I)**4)
      GPP(3,I)=1.-9.*Z(I)+18.*Z(I)**2-10.*Z(I)**3
      GPPP(3,I)=3.*(-3.+12.*Z(I)-10.*Z(I)**2)
      G(4,I)=10.*Z(I)**3-15.*Z(I)**4+6.*Z(I)**5
      GP(4,I)=30.*(Z(I)**2-2.*Z(I)**3+Z(I)**4)
      GPP(4,I)=60.*(Z(I)-3.*Z(I)**2+2.*Z(I)**3)
      GPPP(4,I)=60.*(1.-6.*Z(I)+6.*Z(I)**2)
      G(5,I)=-4.*Z(I)**3+7.*Z(I)**4-3.*Z(I)**5
      GP(5,I)=-12.*Z(I)**2+28.*Z(I)**3-15.*Z(I)**4
      GPP(5,I)=12.*(-2.*Z(I)+7.*Z(I)**2-5.*Z(I)**3)
      GPPP(5,I)=12.*(-2.+14.*Z(I)-15.*Z(I)**2)
      G(6,I)=0.5*(Z(I)**3-2.*Z(I)**4+Z(I)**5)
      GP(6,I)=0.5*(3.*Z(I)**2-8.*Z(I)**3+5.*Z(I)**4)
      GPP(6,I)=3.*Z(I)-12.*Z(I)**2+10.*Z(I)**3
      GPPP(6,I)=3.*(1.-8.*Z(I)+10.*Z(I)**2)
    1 CONTINUE
      RETURN
      END
C
C     SUBROUTINE TO CALCULATE THE PRESSURE GRADIENT PARAMETERS
C
      SUBROUTINE PRGR
C
C     WILL CALCULATE  ZETA  AND  PRES. GRAD. PARAM.
C     FOR NON-SIMILAR  FLOWS
```

```
C
      COMMON/GR4/N,EI,PA,PM,NTURB,NE,NV,IVPL,IPR
      COMMON/GR5/ALPHA,BETA
      COMMON/ZETC/IT,ITER,NST,IZSC,ZETA(101),XBAR(101),
     1UE(101),SPRM
      COMMON/PHYS/CMU,RHO,UINF
C
C     HOWARTH'S  FLOW      UE=1 - 0.125*XBAR
C
      IF(IPR.EQ.0)GO TO 25
      IF(IZSC.EQ.NST)GO TO 40
      IF(IPR.EQ.1)GO TO 20
      IF(IPR.EQ.2)GO TO 30
      DM=XBAR(IZSC)-XBAR(IZSC-1)
      DP=XBAR(IZSC+1)-XBAR(IZSC)
      DUDX=(-DP**2*UE(IZSC-1)+(DP**2-DM**2)*UE(IZSC)+DM**2*
     1UE(IZSC+1))/(DM*DP*(DM+DP))
      GO TO 60
   40 DUDX=DP**2*UE(NST-2)-(DP+DM)**2*UE(NST-1)+DM*(2.*DP+DM)
     1*UE(NST)/(DM*DP*(DP+DM))
   60 CM=XBAR(IZSC)*DUDX/UE(IZSC)
      BETA=2.*CM/(CM+1.)
      ZETA(IZSC)=ZETA(IZSC-1)+(UE(IZSC)+UE(IZSC-1))*DM/2.
      GO TO 50
   20 UE(IZSC)=1.-XBAR(IZSC)/8.
      BETA=-0.25*XBAR(IZSC)*(1.-XBAR(IZSC)/16.)/((1.-XBAR(IZSC)
     1/8.)**2)
      ZETA(IZSC)=XBAR(IZSC)*(1.-XBAR(IZSC)/16.)
      SPRM=UE(IZSC)**2/SQRT(2.*ZETA(IZSC))
      GO TO 50
   30 SN=SIN(XBAR(IZSC))
      CS=COS(XBAR(IZSC))
      BETA=2.*CS/(1.+CS)
      UE(IZSC)=2.*SN
      SPRM=2.*SN**2/SQRT(1.-CS)
      ZETA(IZSC)=2.*(1.-CS)
      GO TO 50
   25 XM=BETA/(2.-BETA)
      ZETA(IZSC)=(XBAR(IZSC))**(XM+1.)/(XM+1.)*CMU*RHO*UINF
   50 RETURN
      END
C
C     SUBROUTINE FOR MATRIX MULTIPLICATION
C
      SUBROUTINE MATMUL(A,B,C)
      DIMENSION A(6,6),B(6,6),C(6,6)
      DO 6 I=1,6
      DO 6 K=1,6
      C(I,K)=0.
      DO 8 J=1,6
    8 C(I,K)=C(I,K)+A(I,J)*B(J,K)
    6 CONTINUE
      RETURN
      END
C
C
CELEMENT=TFFT1.LEQT1B/
C     SUBROUTINE LEQT1B (A,N,NLC,NUC,IA,B,M,IB,IJOB,XL,IER)
C
C-LEQT1B---------S--------LIBRARY 2----------------------------------------
C
C   FUNCTION            - MATRIX DECOMPOSITION, LINEAR EQUATION
C                           SOLUTION - SPACE ECONOMIZER SOLUTION -
C                           BAND STORAGE MODE
C   USAGE               - CALL LEQT1B (A,N,NLC,NUC,IA,B,M,IB,IJOB,XL,
C                           IER)
C   PARAMETERS   A      - INPUT/OUTPUT MATRIX OF DIMENSION N BY
C                           (NUC+NLC+1). SEE PARAMETER IJOB.
C                N      - ORDER OF MATRIX A AND THE NUMBER OF ROWS IN
C                           B. (INPUT)
C                NLC    - NUMBER OF LOWER CODIAGONALS IN MATRIX A.
C                           (INPUT)
C                NUC    - NUMBER OF UPPER CODIAGONALS IN MATRIX A.
C                           (INPUT)
C                IA     - ROW DIMENSION OF A AS SPECIFIED IN THE
C                           CALLING PROGRAM. (INPUT)
C                B      - INPUT/OUTPUT MATRIX OF DIMENSION N BY M.
C                           ON INPUT, B CONTAINS THE M RIGHT-HAND SIDES
C                           OF THE EQUATION AX = B. ON OUTPUT, THE
C                           SOLUTION MATRIX X REPLACES B. IF IJOB = 1,
```

```
C                          B IS NOT USED.
C                M       - NUMBER OF RIGHT HAND SIDES (COLUMNS IN B).
C                            (INPUT)
C                IB      - ROW DIMENSION OF B AS SPECIFIED IN THE
C                            CALLING PROGRAM. (INPUT)
C                IJOB    - INPUT OPTION PARAMETER. IJOB = I IMPLIES WHEN
C                          I = 0, FACTOR THE MATRIX A AND SOLVE THE
C                            EQUATION AX = B. ON INPUT, A CONTAINS THE
C                            COEFFICIENT MATRIX OF THE EQUATION AX = B,
C                            WHERE A IS ASSUMED TO BE AN N BY N BAND
C                            MATRIX. A IS STORED IN BAND STORAGE MODE
C                            AND THEREFORE HAS DIMENSION N BY
C                            (NLC+NUC+1). ON OUTPUT, A IS REPLACED
C                            BY THE U MATRIX OF THE L-U DECOMPOSITION
C                            OF A ROWWISE PERMUTATION OF MATRIX A. U IS
C                            STORED IN BAND STORAGE MODE.
C                          I = 1, FACTOR THE MATRIX A. A CONTAINS THE
C                            SAME INPUT/OUTPUT INFORMATION AS IF
C                            IJOB = 0.
C                          I = 2, SOLVE THE EQUATION AX = B. THIS
C                            OPTION IMPLIES THAT LEQT1B HAS ALREADY
C                            BEEN CALLED USING IJOB = 0 OR 1 SO THAT
C                            THE MATRIX A HAS ALREADY BEEN FACTORED.
C                            IN THIS CASE, OUTPUT MATRICES A AND XL
C                            MUST HAVE BEEN SAVED FOR REUSE IN THE
C                            CALL TO LEQT1B.
C                XL      - WORK AREA OF DIMENSION N*(NLC+1). THE FIRST
C                            NLC*N LOCATIONS OF XL CONTAIN COMPONENTS OF
C                            THE L MATRIX OF THE L-U DECOMPOSITION OF A
C                            ROWWISE PERMUTATION OF A. THE LAST N
C                            LOCATIONS CONTAIN THE PIVOT INDICES.
C                IER     - ERROR PARAMETER.
C                        TERMINAL ERROR = 128+N.
C                          N = 1 INDICATES THAT MATRIX A IS
C                            ALGORITHMICALLY SINGULAR. (SEE THE
C                            CHAPTER L PRELUDE).
C   PRECISION            - SINGLE
C   REQ'D IMSL ROUTINES  - UERTST
C   LANGUAGE             - FORTRAN
C-----------------------------------------------------------------------
C   LATEST REVISION      - NOVEMBER 27,1973
C
      SUBROUTINE LEQT1B(A,N,NLC,NUC,IA,B,M,IB,IJOB,XL,IER)
C
      DIMENSION          A(IA,1),XL(N,1),B(IB,1)
      DATA               ZERO/0./,ONE/1.0/
      IER = 0
      JBEG = NLC+1
      NLC1 = JBEG
      IF (IJOB .EQ. 2) GO TO 80
      RN = N
C                                  RESTRUCTURE THE MATRIX
C                                  FIND RECIPROCAL OF THE LARGEST
C                                  ABSOLUTE VALUE IN ROW I
      I = 1
      NC = JBEG+NUC
      NN = NC
      JEND = NC
      IF (N .EQ. 1 .OR. NLC .EQ. 0) GO TO 25
    5 K = 1
      P = ZERO
      DO 10 J = JBEG,JEND
         A(I,K) = A(I,J)
         Q =  ABS(A(I,K))
         IF (Q .GT. P) P = Q
         K = K+1
   10 CONTINUE
      IF (P .EQ. ZERO) GO TO 135
      XL(I,NLC1) = ONE/P
      IF (K .GT. NC) GO TO 20
      DO 15 J = K,NC
         A(I,J) = ZERO
   15 CONTINUE
   20 I = I+1
      JBEG = JBEG-1
      IF (JEND-JBEG .EQ. N) JEND = JEND-1
      IF (I .LE. NLC) GO TO 5
      JBEG = I
      NN = JEND
   25 JEND = N-NUC
```

```
      DO 40 I = JBEG,N
         P = ZERO
         DO 30 J = 1,NN
            Q =  ABS(A(I,J))
            IF (Q .GT. P) P = Q
   30    CONTINUE
         IF (P .EQ. ZERO) GO TO 135
         XL(I,NLC1) = ONE/P
         IF (I .EQ. JEND) GO TO 37
         IF (I .LT. JEND) GO TO 40
         K = NN+1
         DO 35 J = K,NC
            A(I,J) = ZERO
   35    CONTINUE
   37    NN = NN-1
   40 CONTINUE
      L = NLC
C                                             L-U DECOMPOSITION
      DO 75 K = 1,N
         P =  ABS(A(K,1))*XL(K,NLC1)
         I = K
         IF (L .LT. N) L = L+1
         K1 = K+1
         IF (K1 .GT. L) GO TO 50
         DO 45 J = K1,L
            Q = ABS(A(J,1))*XL(J,NLC1)
            IF (Q .LE. P) GO TO 45
            P = Q
            I = J
   45    CONTINUE
   50    XL(I,NLC1) = XL(K,NLC1)
         XL(K,NLC1) = I
C                                             SINGULARITY FOUND
         IF (RN+P .EQ. RN) GO TO 135
C                                             INTERCHANGE ROWS I AND K
         IF (K .EQ. I) GO TO 60
         DO 55 J = 1,NC
            P = A(K,J)
            A(K,J) = A(I,J)
            A(I,J) = P
   55    CONTINUE
   60    IF (K1 .GT. L) GO TO 75
         DO 70 I = K1,L
            P = A(I,1)/A(K,1)
            IK = I-K
            XL(K1,IK) = P
            DO 65 J = 2,NC
               A(I,J-1) = A(I,J)-P*A(K,J)
   65    CONTINUE
         A(I,NC) = ZERO
   70    CONTINUE
   75 CONTINUE
      IF (IJOB .EQ. 1) GO TO 9005
C                                             FORWARD SUBSTITUTION
   80 L = NLC
      DO 105 K = 1,N
         I = XL(K,NLC1)
         IF (I .EQ. K) GO TO 90
         DO 85 J = 1,M
            P = B(K,J)
            B(K,J) = B(I,J)
            B(I,J) = P
   85    CONTINUE
   90    IF (L .LT. N) L = L+1
         K1 = K+1
         IF (K1 .GT. L) GO TO 105
         DO 100 I = K1,L
            IK = I-K
            P = XL(K1,IK)
            DO 95 J = 1,M
               B(I,J) = B(I,J)-P*B(K,J)
   95       CONTINUE
  100    CONTINUE
  105 CONTINUE
C                                             BACKWARD SUBSTITUTION
      JBEG = NUC+NLC
      DO 125 J = 1,M
         L = 1
         K1 = N+1
         DO 120 I = 1,N
```

```
                  K = K1-I
                  P = B(K,J)
                  IF (L .EQ. 1) GO TO 115
                  DO 110 KK = 2,L
                     IK = KK+K
                     P = P-A(K,KK)*B(IK-1,J)
  110             CONTINUE
  115             B(K,J) = P/A(K,1)
                  IF (L .LE. JBEG) L = L+1
  120          CONTINUE
  125 CONTINUE
      GO TO 9005
  135 IER = 129
 9000 CONTINUE
      CALL UERTST(IER,6HLEQT1B)
 9005 RETURN
      END
C
C
CELEMENT=TFFT1.UERTST/
C     SUBROUTINE UERTST (IER,NAME)
C
C-UERTST----------------LIBRARY 2------------------------------------
C
C   FUNCTION            - ERROR MESSAGE GENERATION
C   USAGE               - CALL UERTST(IER,NAME)
C   PARAMETERS   IER    - ERROR PARAMETER. TYPE + N  WHERE
C                            TYPE= 128 IMPLIES TERMINAL ERROR
C                                   64 IMPLIES WARNING WITH FIX
C                                   32 IMPLIES WARNING
C                            N    = ERROR CODE RELEVANT TO CALLING ROUTINE
C                NAME   - INPUT SCALAR (DOUBLE PRECISION ON DEC)
C                            CONTAINING THE NAME OF THE CALLING ROUTINE
C                            AS A 6-CHARACTER LITERAL STRING.
C   LANGUAGE            - FORTRAN
C--------------------------------------------------------------------
C   LATEST REVISION     - OCTOBER 1,1975
C
      SUBROUTINE UERTST(IER,NAME)
C
      DIMENSION                      IBIT(4)
      INTEGER                  WARN,WARF,TERM,PRINTR
      EQUIVALENCE              (IBIT(1),WARN),(IBIT(2),WARF),(IBIT(3),TERM)
      CHARACTER*6 ITYP(3,4)
      DATA                     ITYP/'WARNIN','G     ','      ',
     *                          'WARNIN','G(WITH',' FIX) ',
     *                          'TERMIN','AL    ','      ',
     *                          'NON-DE','FINED ','      '/,
     *                         IBIT/ 32,64,128,0/
      DATA                     PRINTR/ 6/
      IER2=IER
      IF (IER2 .GE. WARN) GO TO 5
C                                  NON-DEFINED
      IER1=4
      GO TO 20
    5 IF (IER2 .LT. TERM) GO TO 10
C                                  TERMINAL
      IER1=3
      GO TO 20
   10 IF (IER2 .LT. WARF) GO TO 15
C                                  WARNING(WITH FIX)
      IER1=2
      GO TO 20
C                                  WARNING
   15 IER1=1
C                                  EXTRACT 'N'
   20 IER2=IER2-IBIT(IER1)
C                                  PRINT ERROR MESSAGE
      WRITE (PRINTR,25) (ITYP(I,IER1),I=1,3),NAME,IER2,IER
   25 FORMAT(' *** I M S L(UERTST) ***  ',3A6,2X,A6,2X,I2,
     1   ' (IER = ',I3,')')
      RETURN
      END
C
C     SUBROUTINE TO CALCULATE THE STIFFNESS MATRIX IN PRECALCULATED FORM
C
      SUBROUTINE ELECRE(J)
      PARAMETER (NLC=5,NUC=5)
      COMMON/GR3/CR(51),DETA(153)
      COMMON/GR2/F0(51),F1(51),F2(51),B0(51),B1(51),B2(51)
```

```
     1,F3(51)
      COMMON/GR4/N,EI,PA,PM,NTURB,NE,NV,IVPL,IPR
      COMMON/GR5/ALPHA,BETA
      COMMON/GR6/Q(5)
      COMMON/GR1/GSM(154,NLC+NUC+1),SM(6,6),SOL(154)
      ET=DETA(J)
      E2=ET**2
      E3=ET**3
      E4=ET**4
      E5=ET**5
      SM(1,1)=-(10./7.)*Q(2)/ET-Q(3)/2.+(181./462.)*Q(4)*ET
      SM(4,4)=-(10./7.)*Q(2)/ET+Q(3)/2.+(181./462.)*Q(4)*ET
      SM(3,3)=-(1./630.)*Q(2)*E3+(1./9240.)*Q(4)*E5
      SM(6,6)=SM(3,3)
      SM(2,1)=(9./7.)*Q(1)/ET-(3./14.)*Q(2)-(11./84.)*Q(3)*ET+
     &(311./4620.)*Q(4)*E2
      SM(1,2)=-(9./7.)*Q(1)/ET-(17./14.)*Q(2)+(11./84.)*Q(2)*ET+
     &(311./4620.)*Q(4)*E2
      SM(2,2)=Q(1)/2.-(8./35.)*Q(2)*ET+(52./3465.)*Q(4)*E3
      SM(5,5)=-Q(1)/2.-(8./35.)*Q(2)*ET+(52./3465.)*Q(4)*E3
      SM(3,1)=Q(1)/7.-Q(2)*ET/84.-Q(3)*E2/84.+(281./55440.)*
     &Q(4)*E3
      SM(1,3)=-(8./7.)*Q(1)-Q(2)*ET/84.+Q(3)*E2/84.+(281./55440.)
     &*Q(4)*E3
      SM(4,1)=(10./7.)*Q(2)/ET-Q(3)/2.+(25./231.)*Q(4)*ET
      SM(1,4)=(10./7.)*Q(2)/ET+Q(3)/2.+(25./231.)*Q(4)*ET
      SM(5,1)=-(9./7.)*Q(1)/ET-(3./14.)*Q(2)+(11./84.)*Q(3)*ET-
     &(151./4620.)*Q(4)*E2
      SM(1,5)= (9./7.)*Q(1)/ET-(3./14.)*Q(2)-(11./84.)*Q(3)*ET-
     &(151./4620.)*Q(4)*E2
      SM(6,1)=Q(1)/7.+Q(2)*ET/84.-Q(3)*E2/84.+(181./55440.)*Q(4)*E3
      SM(1,6)=-Q(1)/7.+(Q(2)*ET+Q(3)*E2)/84.+(181./55440.)*Q(4)*E3
      SM(3,2)=(9./140.)*Q(1)*ET-Q(2)*E2/60.-Q(3)*E3/1008.+(23./18
     &480.)*Q(4)*E4
      SM(2,3)=-(9./140.)*Q(1)*ET-Q(2)*E2/60.+Q(3)*E3/1008.+(23./
     &18480.)*Q(4)*E4
      SM(4,2)=(9./7.)*Q(1)/ET+(3./14.)*Q(2)-(11./84.)*Q(3)*ET+
     &(151./4620.)*Q(4)*E2
      SM(2,4)=-(9./7.)*Q(1)/ET+(3./14.)*Q(2)+(11./84.)*Q(3)*ET+
     &(151./4620.)*Q(4)*E2
      SM(5,2)=-(11./14.)*Q(1)+Q(2)*ET/70.+(13./420.)*Q(3)*E2
     &-(19./1980.)*Q(4)*E3
      SM(2,5)=(11./14.)*Q(1)+Q(2)*ET/70.-(13./420.)*Q(3)*E2
     &-(19./1980.)*Q(4)*E3
      SM(6,2)=(11./140.)*Q(1)*ET-Q(2)*E2/210.-(13./5040.)*
     &Q(3)*E3+(13./13860.)*Q(4)*E4
      SM(2,6)=-(11./140.)*Q(1)*ET-Q(2)*E2/210.-(13./5040.)*
     &Q(3)*E3+(13./13860.)*Q(4)*E4
      SM(4,3)=Q(1)/7.+(Q(2)*ET-Q(3)*E2)/84.+(181./55440.)*Q(3)*E3
      SM(3,4)=-Q(1)/7.+(Q(2)*ET-Q(3)*E2)/84.+(181./55440.)*Q(3)*E3
      SM(5,3)=-(11./140.)*Q(1)*ET+Q(2)*E2/210.+(13./5040.)*Q(3)*E3
     &-(13./13860.)*Q(4)*E4
      SM(3,5)=(11./140.)*Q(1)*ET+Q(2)*E2/210.-(13./5040.)*Q(3)*E3
     &-(13/13860.)*Q(4)*E4
      SM(6,3)=Q(1)*E2/140.-Q(2)*E3/1260.-Q(3)*E4/5040.+Q(4)*E5/11088.
      SM(3,6)=-Q(1)*E2/140.-Q(2)*E3/1260.+Q(3)*E4/5040.+Q(4)*E5/11088.
      SM(5,4)=(9./7.)*Q(1)/ET+(3./14.)*Q(2)-(11./84.)*Q(3)*ET-
     &(311./4620.)*Q(4)*E2
      SM(4,5)=-(9./7.)*Q(1)/ET+(17./14.)*Q(2)+(11./84.)*Q(3)*ET-
     &(311./4620.)*Q(4)*E2
      SM(6,4)=-Q(1)/7.-(Q(2)*ET-Q(3)*E2)/84.+(281./55440.)*Q(4)*E3
      SM(4,6)=8.*Q(1)/7.-(Q(2)*ET+Q(3)*E2)/84.+(281./55440.)*Q(4)*E3
      SM(6,5)=(9./140.)*Q(1)*ET+Q(2)*E2/60.-Q(3)*E3/1008.
     &-(23./18480.)*Q(4)*E4
      SM(5,6)=-(9./140.)*Q(1)*ET+Q(2)*E2/60.+Q(3)*E3/1008.
     &-(23./18480.)*Q(4)*E4
      RETURN
      END
C
C     SUBROUTINE TO AUGMENT THE ESTIMATED BOUN-LAYER THICKNESS
C
      SUBROUTINE GROW
      COMMON/GR2/ F0(51),F1(51),F2(51),B0(51),B1(51),B2(51)
     1,F3(51)
      COMMON/GR3/CR(51),DETA(153)
      COMMON/GR4/N,EI,PA,PM,NTURB,NE,NV,IVPL,IPR
      NE=NE+1
      IF(NE.GT.50)GO TO 100
      NV=NV+3
      DETA(NE)=2.
```

```
      CR(NE+1)=CR(NE)+DETA(NE)
      EI=CR(NE+1)
      FO(NE+1)=FO(NE)+DETA(NE)
      BO(NE+1)=BO(NE)+DETA(NE)
      F1(NE+1)=1.0
      B1(NE+1)=1.0
      F2(NE+1)=0.
      B2(NE+1)=0.
  100 RETURN
      END
```

BIBLIOGRAPHY

1. Blasius, H., "Grenzschichten in Flüssigkeiten mit kleiner Reibung," Z. Math. und Phys., Vol. 56, pp. 1-37, 1908.

2. Falkner, V.M., and Skan, S.W., "Solutions of the Boundary-Layer Equations," Phil. Mag., Vol. 12, pp. 865-896, 1931.

3. Howarth, L., "On the Solution of the Laminar Boundary Layer Equations," Proc. Roy. Soc. A, Vol. 164, pp. 547-579, 1938.

4. Thwaites, B., "Approximate Calculation of the Laminar Boundary Layer," The Aeronautical Journal, Vol. 1, pp. 245-281, 1949.

5. Pohlhausen, K., "Zur näherungsweisen Integration der Differentialgleichung der Grenzschicht," Zeitschrift f. angew. Math. u. Mech., Vol. 1, pp. 252-268, 1921.

6. Görtler, H., "A New Series for the Calculation of Steady Laminar Boundary Layer Flows," Journal of Math. and Mechanics, Vol. 6, pp. 1-66, 1957.

7. Werle, M. J., and Davis, R. T., "Incompressible Laminar Boundary Layers on a Parabola at Angle of Attack: A Study of the Separation Point," Journal of Applied Mechanics, pp. 7-12, 1972.

8. Abbott, D. E., and Kline, S. J., "Experimental Investigation of Subsonic Turbulent Flow Over Single and Double Backward Facing Steps," Trans. ASME Journ. of Basic Eng'g., pp. 317-325, 1962.

9. Cebeci, T., and Keller, H. B., "Shooting and Parallel Shooting Methods for Solving the Falkner Skan Boundary-Layer Equation," Jour. of Comp. Physics, Vol. 7, no. 2, April 1971.

10. Oden, J. T., and Wellford Jr., L. C., "Analysis of Flow of Viscous Fluids by the Finite Element Method," AIAA Journal, Vol. 10, pp. 1590-1599, 1972.

11. Tadros, R. N., and Kirkhope, J., "Galerkin Finite Element Formulation in Viscous Flow," Computers and Fluids, Vol. 6, pp. 293-298, 1978.

12. Lynn, P. P., and Alani, K., "Efficient Least Squares Finite Elements for Two Dimensional Laminar Boundary Layer Analysis," Int.'l Journ. for Numer. Methods in Eng'g., Vol. 10, pp. 809-825, 1976.

13. Bradshaw, P., "Turbulent Boundary Layers," Aeronautical Journal, Vol. 72, pp. 451-459, May 1968.

14. Biringen, S., and Levi, J., "Calculation of Two Dimensional Turbulent Boundary Layers," AIAA Journal, Vol. 16, pp. 1016-1019, 1978.

15. Rastogi, A. K., and Rodi, W., "Calculation of General Three Dimensional Turbulent Boundary Layers," AIAA Journal, Vol. 16, pp. 151-159, 1978.

16. Keller, H. B., and Cebeci, T., "Accurate Numerical Methods for Boundary Layer Flows. II: Two Dimensional Turbulent Flows," AIAA Journal, Vol. 10, pp. 1193-1199, 1972.

17. Dean, R. B., "A Single Formula for the Complete Velocity Profile in a Turbulent Boundary Layer," Trans. of ASME, Journ. of Fluids Eng'g., pp. 723-726, 1976.

18. Yeung, W.-S., and Yang, R.-J., "Application of the Method of Integral Relations to the Calculation of Two Dimensional Turbulent Boundary Layers," Trans. ASME, Journ. of Applied Mechanics, Vol. 48, pp. 701-706, 1981.

19. Rodi, W., "Examples of Turbulence Models for Incompressible Flows," AIAA Journ., Vol. 20, pp. 872-879, 1982.

20. Launder, B. E., and Spalding, D. B., Mathematicl Models of Turbulence, London, Academic Press, 1972.

21. Cebeci, T., and Smith, A. M. O., Analysis of Turbulent Boundary Layers, New York, Academic Press, 1974.

22. Cebeci, T., and Bradshaw, P., Momentum Transfer in Boundary Layers, Washington, Hemisphere Publishing Corp., Mc Graw Hill, 1977.

23. Wheeler, A. J., and Johnston, J. P., "An Assesment of Three Dimensional Turbulent Boundary Layer Prediction Methods," Trans. ASME, Journ. of Fluids Eng'g., pp. 415-420, 1973.

24. Bradshaw, P., Ferriss, D. H., "Calculation of Boundary Layer Development Using the Turbulent Energy Equation: Compressible Flow on Adiabatic Walls," J. of Fluid Mechanics, Vol. 46, pp. 83-110, 1971.

25. Sharma, O. P.; Wells, R. A.; Schlinker, R. H.; Bailey, D. A., "Boundary Layer Development on Turbine Airfoil Suction Surfaces," Trans. ASME, Journ. of Eng'g. for Power, Vol. 104, pp. 698-706, 1982.

26. Hayes, W. D., and Probstein, R. F., Hypersonic Flow Theory, New York, Academic Press, 1959.

27. Cebeci, T., "The Behaviour of Turbulent Flow Near a Porous Wall with Pressure Gradient," AIAA Journ., Vol. 8, pp. 2152-2156, 1970.

28. Zienkiewicz, O. C., The Finite Element Method in Engineering Science, London, Mc Graw Hill, 1971.

29. Huebner, K. H., The Finite Element Method for Engineers, New York, Wiley, 1975.

30. Fort, T., Finite Differences and Difference Equations in the Real Domain, New York, Oxford University Press, 1948.

31. Hildebrand, F. B., Introduction to Numerical Analysis, New York, Mc Graw Hill, 1956.

32. Martin, R. S., and Wilkinson, J. H., "Solution of Symmetric and Unsymmetric Band Equations and the Calculation of Eigenvectors of Band Matrices," Numerische Mathematik, Vol. 9(4), pp. 279-301, 1967.

33. Rosenhead, L., Laminar Boundary Layers, Oxford, Clarendon Press, 1966.

34. Bismarck-Nasr, M. N., "A Finite-Difference / Galerkin Finite-Element Solution of a Turbulent Boundary Layer," AIAA Journal, Vol. 15, pp. 1813-1816, 1977.

35. Van Driest, E. R., "On Turbulent Flow near a Wall," J. Aeronaut. Sci., Vol. 23, pp. 1007, 1956.

36. Klebanoff, P. S., "Characteristics of Turbulence in a Boundary Layer with Zero Pressure Gradient," NACA Tech. Note 3178, 1954.

37. Cebeci, T., and Smith, A. M. O., "A Finite-Difference Method for Calculating Compressible Laminar and Turbulent Boundary Layers," Journ. of Basic Engineering, Vol. 92, pp. 523-535, 1970.

38. Tifford, A. N., "Heat Transfer and Frictional Effects in Laminar Boundary Layers. Part 4: Universal Series Solutions," WADC Technical Report, pp. 53-288, Part 4, 1954.

39. Schlichting, H., Boundary Layer Theory, 6th edition, Mc Graw Hill, New York, 1968.