27
10/30/79
2407T/S

SAND79-0893
Unlimited Release

Introduction to Computational Techniques for Boundary Layers

Frederick G. Blottner

MASTER

Prepared by Sandia Laboratories, Albuquerque, New Mexico 87185
and University of California 94680 for the United States Department
of Energy under Contract DE-AC04-76DP00789

Revised September 1979

Sandia Laboratories

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Issued by Sandia Laboratories, operated for the United States
Department of Energy by Sandia Corporation.

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the Department of Energy, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

Printed in the United States of America

**Available from
National Technical Information Service
U. S. Department of Commerce
6285 Port Royal Road
Springfield, VA 22161**

Price: Printed Copy \$6.00; Microfiche \$3.00

SAND79-0893
Unlimited Release
Printed September 1979

INTRODUCTION TO
COMPUTATIONAL TECHNIQUES FOR BOUNDARY LAYERS

Frederick G. Blottner
Fluid Mechanics and Heat Transfer Division I, 5511
Sandia Laboratories
Albuquerque, NM 87185

ABSTRACT

Finite-difference procedures to solve boundary layer flows in fluid mechanics are explained. The governing equations and the transformations utilized are described. Basic solution techniques are illustrated with the similar boundary layer equations. Nonsimilar solutions are developed for the incompressible equations. Various example problems are solved, and the numerical results in the Fortran listing of the computer codes are presented.

DISCLAIMER

This book was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

ACKNOWLEDGMENT

My thanks to Dr. Carl Zorowski, Department Head of Mechanical and Aerospace Engineering at North Carolina State University, and to Dr. Fred R. DeJarnette and Dr. James C. Williams, members of the department, for their interest and support.

CONTENTS

	<u>Page</u>
Introduction	7
Objective	7
Philosophy of Computational Fluid Dynamics	7
Equations, Properties, and Peculiarities of the Incompressible Boundary Layer	8
Worked Problem: Explicit Solution for Flat Plate Flow	10
Two-Dimensional and Axisymmetric Boundary Layers	15
Governing Equations	15
Transformation of Equations	16
Incompressible Laminar Boundary Layers	21
Similar Solutions	21
Model Problem	21
Tridiagonal Solution Method	24
Shooting Method	33
Central Difference Scheme	33
Stream Function Variables	45
Midpoint Scheme	50
Evaluation of Skin Friction	56
Variable Grid Scheme	59
Higher-Order Schemes	61
Influence of Edge Location	68
Nonsimilar Solutions	70
Worked Problem: Howarth Flow	74
References	80

ILLUSTRATIONS

<u>Figure</u>		<u>Page</u>
1	Flat Plate Geometry	11
2	Flat Plate Skin Friction	14
3	Two-Dimensional Boundary Layer	16
4	Accuracy of Schemes for Similar Solutions	38
5	Flat Plate Velocity with Suction	39
6	Velocity Across Boundary Layer for Stream Function Variable	49
7	Velocity Across Boundary Layer for Similarity Variable	49
8	Accuracy of Velocity	57
9	Importance of Method of Evaluating the Wall Velocity Gradient	58
10	Influence of Grid Spacing on Accuracy of Results for Flat Plate Turbulent Boundary Layer	61
11	Higher-Order Scheme for Flat Plate Laminar Boundary Layer	68
12	Influence of Edge Location	70
13	Accuracy of Skin Friction for Howarth Problem	77
14	Computation Time on CDC 6600	78

INTRODUCTION TO COMPUTATIONAL TECHNIQUES FOR BOUNDARY LAYERS

Introduction

This report is a compilation of notes I prepared as Engineer-in-Residence, for a special topics course on Computational Fluid Mechanics that I taught in the Department of Mechanical and Aerospace Engineering, North Carolina State University in the spring of 1978. This is Part III of the course. Part I - "Methods for Initial Value Problems," and Part II - "Semidirect Methods for Viscous Steady-State Flows" were taught by John V. Rakich and Patrick J. Roache respectively. Additions and changes have been made since the original notes were prepared.

Objective

This report describes some basic numerical procedures to solve boundary layer flows. Since the partial differential equations for two-dimensional flow are parabolic, the techniques used for solving the heat conduction equation are appropriate. However, directly applying these procedures is not completely acceptable because of difficulties described below.

Classical boundary layer theory is considered in several books,¹⁻⁴ but numerical solution procedures are not adequately covered. Certain forms of the governing equations, along with one type of procedure for solving these forms, have been described by different authors.⁵⁻¹³ Except for the scheme developed in Nash and Patel,⁷ various methods for solving flows will be described. The turbulence model found in Reference 7 results in the governing equations being hyperbolic; the solution procedures are significantly different from the parabolic approaches being considered.

Philosophy of Computational Fluid Dynamics

In all engineering problems, various assumptions are used to model physical phenomena. When computational techniques are used to solve the resulting mathematical problem, the results must be numerically accurate so the assumptions can be evaluated without being contaminated by the solution procedure. Therefore, not only must the results be accurate, but we must also develop a method to determine precision of the results themselves. To predict problems that will arise in an engineering project, however, numerical solutions need not be extremely accurate, but they should at least exhibit an accuracy that is compatible with the approximations of the model and the uses to which the results will be applied.

Equations, Properties, and Peculiarities of the Incompressible Boundary Layer

Flow in the two-dimensional boundary layer is governed by the continuity and momentum equations in dimensional form,

$$\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} = 0 \quad (1a)$$

and

$$u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} = - \frac{1}{\rho^*} \frac{dp^*}{dx^*} + \nu^* \frac{\partial^2 u^*}{\partial y^{*2}} \quad (1b)$$

These equations can be derived from several sources.^{1 2} The coordinates x^* and y^* are independent variables and represent the distances parallel to and perpendicular (normal) to the surface of the body. Velocities u^* and v^* are dependent variables and are tangent and normal to the surface. The remaining quantities are density ρ^* , pressure p^* , and kinematic viscosity ν^* , that is, the ratio of viscosity μ^* to density. To complete Eqs (1a) and (1b), the following boundary conditions must be added:

$$\begin{aligned} \text{At } y^* = 0: \quad u^* = v^* = 0 \\ \text{At } y^* = \infty: \quad u^* = U^*(x^*) \end{aligned} \quad (2)$$

These classical first-order boundary layer equations result when the method of asymptotic expansions are applied to the complete Navier-Stokes equations. Conditions at the outer edge of the boundary layer are obtained from the inviscid flow along the surface of the body. The equation for determining the momentum along the inviscid body streamline is:

$$U^* \frac{dU^*}{dx^*} = - \frac{1}{\rho^*} \frac{dp^*}{dx^*} \quad (3)$$

This equation can be used in Eq (1b) to eliminate the term that has the density and pressure gradient. The governing equations are nondimensionalized with the following quantities:

$$\begin{aligned} x &= x^* / L^* & U &= U^* / U_o^* \\ y &= \sqrt{Re} \, y^* / L^* & Re &= U_o^* L^* / \nu^* \\ u &= u^* / U_o^* \\ v &= \sqrt{Re} \, v^* / U_o^* \end{aligned}$$

where the quantity L^* is a characteristic length of the problem being considered. When these quantities are introduced into Eqs (1) and (2), the governing equations and boundary conditions become

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (4a)$$

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = U \frac{dU}{dx} + \frac{\partial^2 u}{\partial y^2} \quad (4b)$$

with

$$\text{At } y = 0: \quad u = v = 0 \quad (4c)$$

$$\text{At } y = \infty: \quad u = U(x) \quad .$$

The velocity at the edge of the boundary layer is a function of x and is obtained from the inviscid flow solution. In the governing equations above, there are two unknowns, u and v , as well as two equations. If the distribution of u and v are known across the boundary layer at x_0 , a Taylor's series expansion procedure can be used to determine the solution a small distance, Δx , downstream.

$$u(x_0 + \Delta x, y) = u(x_0, y) + \frac{\partial u}{\partial x}(x_0, y)\Delta x + \dots \quad (5)$$

A similar equation for v does not exist since there is no derivative $\partial v / \partial x$. Therefore, this is not the usual type of initial-value (Cauchy) problem. The value of v is obtained from continuity Eq (4a), which yields

$$v = - \int_0^y \frac{\partial u}{\partial x} dy \quad (6)$$

To solve the above equations, it has been assumed that u and v can be given at x_0 and the solution continued downstream by using the above solutions. The correct initial condition is the distribution of u across the boundary layer; the value of v is determined from the compatibility equation which was formulated by Prandtl and extended to other cases by Ting.¹⁴ We now combine Eqs (4a-c) to give

$$-u \frac{\partial v}{\partial y} + v \frac{\partial u}{\partial y} = U \frac{dU}{dx} + \frac{\partial^2 u}{\partial y^2} \quad ,$$

and since

$$-u \frac{\partial v}{\partial y} + v \frac{\partial u}{\partial y} = -u^2 \frac{\partial}{\partial y} \left(\frac{v}{u} \right) \quad ,$$

the compatibility equation becomes, if $v_w = 0$,

$$v = -u \int_0^y \frac{1}{u^2} \left[U \frac{dU}{dx} + \frac{\partial^2 u}{\partial y^2} \right] dy \quad (7)$$

If u is a function of y across the boundary layer and if the edge conditions are known, Eq (7) can be used to determine v ; no boundary condition is required at infinity for v .

If the velocity, u , becomes zero at any point in the flow, a singularity will exist at that point and the solution cannot be extended downstream as an initial-value problem. In fact, reverse flow will occur and the equation behaves as a "time reversal problem." With a parabolic, partial differential equation, we can obtain a unique and stable solution in the positive time direction but not in the negative direction.¹⁵ Reverse boundary layer flows are special problems that require numerical procedures different from those being considered here. In a boundary layer flow on a wall, $u = 0$ at the surface, and this results in a singularity. Since this occurs where the boundary conditions are given there is no problem in extending the solution downstream because it is already known from the wall conditions.

Another peculiarity of these equations is that the boundary condition at the outer edge is applied at infinity but a numerical solution must be performed to some finite value of y unless a transformation has been made. Since u asymptotically approaches the inviscid flow, U , the boundary condition can be applied at a reasonable value of the normal coordinate, y_e . The value for y_e is not well defined; how it should vary as the boundary layer develops along a surface must be determined as part of the solution.

Worked Problem: Explicit Solution for Flat Plate Flow

To illustrate some problems when solving the boundary layer Eqs (4a-c) in physical coordinates, we apply an explicit difference method, although some problems in stability of this scheme will be revealed. For this problem, a Cartesian coordinate system and grid is used (Figure 1). The flow is calculated within the region bounded by the dotted line and the solution starts at $x = 0$. The x and y coordinates are divided into a uniform grid of step-size Δx and Δy and there are I and J grid points along and across the computational domain, respectively. Therefore, the coordinates can be expressed as

$$\begin{aligned} x_i &= \Delta x(i - 1) \quad \text{where } i = 1, 2, 3, \dots, I \text{ and} \\ y_j &= \Delta y(j - 1) \quad \text{where } j = 1, 2, 3, \dots, J \end{aligned} \quad (8)$$

Solution will be progressively obtained until $x_I = 1$ and the value of y_J remains to be determined. Since similarity solutions of Blasius are "exact" solutions these results are used to help solve the problem and judge the accuracy of our results. The nondimensional thickness, δ of the boundary

layer, the skin friction, C_f , the velocity, v_δ , at the outer edge, and the similarity coordinate, η , are represented as follows:

$$\delta = y_\delta^* \sqrt{\text{Re}/L^*} \approx 5.0 \sqrt{x}$$

$$C_f \sqrt{\text{Re}_x^*} = 0.66412$$

(9)

$$v_\delta = 1.21678/\sqrt{2x}$$

$$\eta = y/\sqrt{2x} \quad ,$$

where

$$\text{Re}_x = U_\infty^* x^*/\nu^* \quad \text{and} \quad C_f = \tau_W \left(\frac{1}{2} \rho^* U_\infty^{*2} \right)^{-1/2} .$$

From the first statement in Eq (9), we derive the value of y_j of about 5, in order to include the boundary layer within the computational domain. At $x = 0$, the boundary layer thickness becomes zero and the velocity, u , becomes U across the computational domain except at the wall where it is zero. The velocity, v , is infinite at this point because of the singularity at the leading edge. With the assumed u velocity distribution, Eq (7) is used to obtain $v = 1/\Delta y$, and this is used for the initial condition except at the wall, where $v = 0$.

Figure 1. Flat Plate Geometry

Momentum Eq (4b) is written in finite-difference form with a forward difference in x at j and central differences for y derivatives at i . The velocity, v , is evaluated at $(i + 1/2, j)$. The resulting difference equation is

$$u_{i+1,j} = u_{i,j} - \Delta x \left[\left(\frac{1}{2} \right) v_{i+1/2,j} (u_{j+1} - u_{j-1})_i - (u_{j+1} - 2u_j + u_{j-1})_i / \Delta y \right] / (\Delta y u_{i,j}) \quad (10)$$

If $v_{i+1/2,j}$ is approximated with $v_{i-1/2,j}$, then all the quantities on the right side of this relation are known from the initial conditions or from the previous profile and the solution at the next step along the surface is readily obtained from Eq (10). Continuity Eq (4a) is used to determine velocity, v , which is evaluated at the grid point, $(i + 1/2, j - 1/2)$, with midpoint differences for the derivatives. The difference equation for the continuity equation then becomes

$$v_{i+1/2,j} = v_{i+1/2,j-1} - \Delta y \left[(u_{i+1} - u_i)_j + (u_{i+1} - u_i)_{j-1} \right] / (2\Delta x) \quad (11)$$

If the continuity equation is evaluated at the grid point, $(i + 1/2, j)$, and the y derivatives are determined with central differences, then v can be found only at grid points where $j = 3, 5, 7$, etc.

To solve the heat conduction equation with an explicit scheme, the following condition is sufficient for a stable solution:

$$\frac{\Delta t}{\Delta x^2} \leq 1/2 \quad (12)$$

When momentum Eq (4b) is applied to flat plate flow, v is equal to zero and the coefficient u is assumed a constant; the equation reduces to the heat conduction equation, and the stability condition becomes

$$S = \frac{2\Delta x}{u\Delta y^2} \leq 1 \quad (13)$$

where S is the stability parameter. In the numerical solution this quantity is monitored at the first grid point away from the wall, where it has a maximum value. The skin friction is determined from the following relations:

$$C_f \sqrt{Re}_x = 2 \sqrt{x} \frac{\partial u}{\partial y} \quad (14)$$

where

$$\frac{\partial u}{\partial y} = (2u_2 - (1/2)u_3) / \Delta y$$

A computer program, BLX, has been written to solve the foregoing equations and determine the boundary layer flow along a flat plate. The following FORTRAN notation is used:

<u>FORTRAN Name</u>	<u>Symbol</u>
IMAX, JMAX	I, J
X, Y(J), ETA(J)	x, y, η
U(J), UL(J)	$u_{i+1,j}$, $u_{i,j}$
V(J), VL(J)	$v_{i+1/2,j}$, $v_{i-1/2,j}$
DX, DY	Δx , Δy
S	Stability parameter
CF	$C_f \sqrt{Re_x}$

The program is listed below and this code is used to solve the following five cases:

<u>Case</u>	<u>I</u>	<u>J</u>	<u>Y_J</u>
1	201	41	10.0
2	201	11	2.5
3	101	41	10.0
4	26	21	10.0
5	201	21	5.0

```

C      EXPLICIT SOLUTION OF FLAT PLATE FLOW
      DIMENSION U(101), UL(101), V(101), VL(101), Y(101), ETA(101)
      IMAX = 101
      IMAX = 26
      IMAX = 201
      JMAX = 11
      JMAX = 41
      JMAX = 21
      Y(JMAX) = 2.5
      Y(JMAX) = 10.0
      Y(JMAX) = 5.0
      DX = 1.0 / ( IMAX - 1.0 )
      DY = Y(JMAX) / ( JMAX - 1.0 )
      X = 0.0
      Y(1)=0.0
      ETA(1)=0.0
      U(1)=0.0
      UL(1)=0.0
      V(1)=0.0
      VL(1)=0.0
      DO 10 J = 2,JMAX
      Y(J) = Y(J-1) * DY
      U(J)=1.0
      UL(J)=1.0
      V(J)=1.0/DY
10      VL(J)=1.0/DY
      PRINT 9500,(J,Y(J),UL(J),VL(J),J=1,JMAX)
9500  FORMAT(10X,I3,3(2X,E12.5))
      DO 100 I = 2,IMAX
      X = X + DX
      DO 50 J = 2,JMAX
      IF( J.EQ. JMAX ) GO TO 40
      U(J) = UL(J) + DX*(-0.5*VL(J)*(UL(J+1)-UL(J-1)) +
1      (UL(J+1)-2.0*UL(J)+UL(J-1))/DY)/(DY*UL(J))
40      V(J) = V(J-1) - DY*(U(J)-UL(J)+U(J-1)-UL(J-1))/(2.0*DX)
      ETA(J) = Y(J) / SQRT( 2.0*X )
50      CONTINUE
      DO 60 J =2,JMAX
      VL(J) = V(J)
60      UL(J) = U(J)
      S = 2.0*DX/(U(2)*DY**2)
      CF = 2.0 * SQRT( X ) * ( 2.0*U(2) - 0.5*U(3) ) / DY
      PRINT 9000, I, X, S, CF, V(JMAX)
9000  FORMAT(5X,I3,4(2X,E12.5))
100     CONTINUE
      PRINT 9600,(J,Y(J),ETA(J),U(J),V(J),J=1,JMAX)
9600  FORMAT(10X,I3,4(2X,E12.5))
      RETURN
      END

```

Figure 2 illustrates skin friction for the five cases and the known accurate solution which is called "exact." For all cases the solution is initially very inaccurate near the leading edge where the singularity is not properly handled. For Case 1, the scheme yields a reasonably accurate result downstream on the flat plate. At first, Case 2 is yielding the same result as Case 1, but the boundary layer thickness becomes larger than the computational domain and adversely influences the results. The problem of stability in the explicit scheme is illustrated in Case 3 where the solution becomes unstable when $S > 2.5$. Case 4 is a solution with larger step-sizes and the results are correspondingly less accurate. Case 5 shows that the same result is obtained if the outer computational boundary is large enough to contain the boundary layer flow. When the step-size, Δy , is halved with this scheme, the step-size, Δx , has to be reduced by a factor of eight. Therefore, it becomes very difficult to obtain accurate results in the y-direction without using an excessive number of steps along the wall.

Figure 2. Flat Plate Skin Friction

Governing Equations

The steady boundary layer equations for a compressible fluid are written with the parameter j , which is 0 for two-dimensional flow and 1 for axisymmetric flow. The nomenclature for these equations is illustrated in Figure 3. The governing equations for a perfect gas which has constant specific heat, C_p , and constant Prandtl number, Pr , in terms of dimensional quantities are:

Continuity

$$\frac{\partial}{\partial x} \left(r_b^j \rho u \right) + \frac{\partial}{\partial y} \left(r_b^j \rho v \right) = 0 \quad , \quad (15a)$$

x-Momentum

$$\rho u \frac{\partial u}{\partial x} + \rho v \frac{\partial u}{\partial y} = - \frac{\partial p}{\partial x} + \frac{\partial}{\partial y} \left(\mu \frac{\partial u}{\partial y} \right) \quad , \quad (15b)$$

Energy - Temperature Form

$$\rho c_p u \frac{\partial T}{\partial x} + \rho c_p v \frac{\partial T}{\partial y} = u \frac{\partial p}{\partial x} + \frac{\partial}{\partial y} \left(\frac{\mu c_p}{Pr} \frac{\partial T}{\partial y} \right) + \left(\mu \frac{\partial u}{\partial y} \right)^2 \quad , \text{ and} \quad (15c)$$

Energy - Total Enthalpy Form

$$\rho u \frac{\partial H}{\partial x} + \rho v \frac{\partial H}{\partial y} = \frac{\partial}{\partial y} \left[\mu u \left(1 - \frac{1}{Pr} \right) \frac{\partial u}{\partial y} + \frac{\mu}{Pr} \frac{\partial H}{\partial y} \right] \quad . \quad (15d)$$

In these equations it is assumed that the pressure is constant across the boundary layer. The governing equations are completed by the equation-of-state for a perfect gas:

$$p = \rho R T \quad \text{or} \quad h = \frac{\gamma}{\gamma - 1} p / \rho \quad , \quad (16)$$

and a viscosity law (Sutherland for air, T in K)

$$\mu \text{ (Kg/m - s)} = 1.458 \times 10^{-6} T^{3/2} / (T + 110.4) \quad . \quad (17)$$

Pressure p is specified and obtained from the inviscid flow solution for the body of interest. The dependent variables are u , v , ρ , and T or H . The boundary conditions at the wall are

$$u = v = 0$$

$$T = T_W(x) \quad \text{or} \quad k \frac{\partial T}{\partial y} = q_W(x) \quad , \quad (18)$$

while the outer edge conditions are

$$u = u_e(x) \quad \text{and} \quad T = T_e(x) \quad . \quad (19)$$

Figure 3. Two-Dimensional Boundary Layer

Transformation of Equations

As noted previously, there are several difficulties with solving the boundary equations by using physical coordinates:

- Changes in the thickness of the boundary layer along the body require the computational domain to vary to take this into account
- A singularity at the tip of an axisymmetric body or at the leading edge of a sharp two-dimensional body occurs
- Gradients in the flow tend to be large

In some of the early numerical solutions, the Crocco transformation was utilized, in which the u velocity is the independent variable across the boundary layer and the shearing stress becomes the dependent variable. The variable u/u_e varies from 0 to 1 across the boundary layer but there is a singularity at the outer edge. Also, if the velocity overshoots, the variables become double-valued as a function of u .

In the work of Patankar and Spalding,¹² the von Mises stream function form of the governing equations has been used. In this case, the stream function, ψ , is introduced as

$$r_b^j \rho u = \frac{\partial \psi}{\partial y} \quad \text{and} \quad r_b^j \rho v = - \frac{\partial \psi}{\partial x} . \quad (20)$$

and the continuity equation is satisfied. The new independent variables are $X = x$ and ψ . Derivatives in the governing equations become

$$\frac{\partial}{\partial x} = \frac{\partial}{\partial X} - r_b^j \rho v \frac{\partial}{\partial \psi} \quad (21)$$

$$\frac{\partial}{\partial y} = r_b^j \rho u \frac{\partial}{\partial \psi} .$$

The momentum and total enthalpy forms of the energy equation become

$$\begin{aligned} \frac{\partial u}{\partial X} &= - \frac{1}{\rho u} \frac{dp}{dX} + \frac{\partial}{\partial \psi} \left(r_b^{2j} \mu \rho u \frac{\partial u}{\partial \psi} \right) \\ \frac{\partial H}{\partial X} &= \frac{\partial}{\partial \psi} \left\{ \rho u \mu r_b^{2j} \left[(1 - 1/Pr) u \frac{\partial u}{\partial \psi} + \frac{1}{Pr} \frac{\partial H}{\partial \psi} \right] \right\} . \end{aligned} \quad (22)$$

The final independent variables are

$$\xi = X \quad (23)$$

$$\omega = (\psi - \psi_I) / (\psi_E - \psi_I) ,$$

where ψ_I is the stream function at the inner boundary or wall, while ψ_E is the stream function at the outer edge of the computational domain or boundary layer edge. The value of ω varies from 0 to 1 across the layer. Derivatives in the equations become

$$\frac{\partial}{\partial X} = \frac{\partial}{\partial \xi} + \frac{\partial \omega}{\partial X} \frac{\partial}{\partial \omega} \quad (24)$$

$$\frac{\partial}{\partial \psi} = \frac{1}{\psi_E - \psi_I} \frac{\partial}{\partial \omega} ,$$

and the resulting governing equations are of the form

$$\frac{\partial \phi}{\partial \xi} + (a + b\omega) \frac{\partial \phi}{\partial \omega} = \frac{\partial}{\partial \omega} \left(c \frac{\partial \phi}{\partial \omega} \right) + d , \quad (25)$$

where

$$a = -\frac{1}{\psi_E - \psi_I} \frac{d\psi_I}{dx}, \text{ and } b = -\frac{1}{\psi_E - \psi_I} \left(\frac{d\psi_E}{dx} - \frac{d\psi_I}{dx} \right).$$

The coefficients c and d depend on the conservation equation and for the momentum equation are

$$\phi = u$$

$$c = r_b^j \mu \rho u / (\psi_E - \psi_I)^2 \quad (26)$$

$$d = -\frac{1}{\rho u} \frac{dp}{dx}.$$

Equation (25) is the form of the governing equations and has been applied to various parabolic flows.^{5 12} This method has a singularity at the wall which can introduce some special problems to be discussed later.

The other prevalent procedures for solution use the governing equations in similarity coordinates but in two different forms. Both of these methods utilize the Levy-Lees transformation

$$\eta(x, y) = u_e r_b^j \sqrt{\frac{\mathcal{K}}{2\xi}} \int_0^y \rho \, dy \quad (27)$$

$$\xi(x) = \mathcal{K} \int_0^x (\rho \mu)_r u_e r_b^{2j} \, dx,$$

where $\mathcal{K}$ is a constant and the derivatives become

$$\frac{\partial}{\partial x} = \mathcal{K} (\rho \mu)_r u_e r_b^{2j} \frac{\partial}{\partial \xi} + \frac{\partial \eta}{\partial x} \frac{\partial}{\partial \eta}$$

$$\frac{\partial}{\partial y} = \rho u_e r_b^j \sqrt{\frac{\mathcal{K}}{2\xi}} \frac{\partial}{\partial \eta}.$$

The new dependent variables are

$$F = u/u_e$$

$$\theta = h/h_e = T/T_e$$

$$G = H/H_e$$

(28)

$$V = 2\xi \left(F \partial\eta/\partial x + \rho v r_b^j / \sqrt{2\xi/\mathcal{K}} \right) \left[\mathcal{K}(\rho\mu)_r u_e r_b^{2j} \right] . \quad (28)$$

(cont)

The governing equations transform to the F-V similarity form:

Continuity

$$2\xi \partial F/\partial \xi + \partial V/\partial \eta + F = 0 \quad (29a)$$

x-Momentum

$$2\xi F^2 \partial F/\partial \xi + V \partial F/\partial \eta + \beta(F^2 - \theta) - \partial(\ell \partial F/\partial \eta)/\partial \eta = 0 \quad (29b)$$

or

$$2\xi F \partial F/\partial \xi + V \partial F/\partial \eta + (\beta H_e/h_e)(F^2 - G) - \partial(\ell \partial F/\partial \eta)/\partial \eta = 0 \quad (29c)$$

Energy - Temperature Form

$$2\xi F \partial \theta/\partial \xi + V \partial \theta/\partial \eta - \alpha \ell (\partial F/\partial \eta)^2 - (1/Pr) \partial(\ell \partial \theta/\partial \eta)/\partial \eta = 0 , \quad (29d)$$

Energy - Enthalpy Form

$$2\xi F \partial G/\partial \xi + V \partial G/\partial \eta - \partial \left[\left(u_e^2/H_e \right) (1 - 1/Pr) F \partial F/\partial \eta + (\ell/Pr) \partial G/\partial \eta \right] / \partial \eta = 0 \quad (29e)$$

where the following notation is introduced

$$\ell = \rho\mu/(\rho\mu)_r$$

$$\beta = (2\xi/u_e) du_e/d\xi$$

$$\alpha = u_e^2/(c_p T_e) .$$

Equations (29a-d) can be written without the continuity equation which is automatically satisfied when the stream function, $\psi = f \sqrt{2\xi/\mathcal{K}}$, is introduced. The velocity components become

$$\rho u r_b^j = \partial \psi / \partial y = \rho u_e r_b^j \partial f / \partial \eta$$

$$\rho v r_b^j = -\partial \psi / \partial x = \sqrt{2\xi/\mathcal{K}} \left[\mathcal{K}(\rho\mu)_r u_e r_b^j \left(2\xi \frac{\partial f}{\partial \xi} + f \right) (2\xi) - (\partial \eta / \partial x) (\partial f / \partial \eta) \right] .$$

With $F = \partial f / \partial \eta$ and the definition of V , the following is obtained

$$V = -(2\xi \partial f / \partial \xi + f) \quad .$$

The governing equations are transformed to the f -similarity form

X-Momentum

$$2\xi \left(\frac{\partial f}{\partial \eta} \frac{\partial^2 f}{\partial \xi \partial \eta} - \frac{\partial f}{\partial \xi} \frac{\partial^2 f}{\partial \eta^2} \right) = \frac{\partial}{\partial \eta} \left(\ell \frac{\partial^2 f}{\partial \eta^2} \right) + f \frac{\partial^2 f}{\partial \eta^2} + \beta \left[\theta - \left(\frac{\partial f}{\partial \eta} \right)^2 \right] \quad (30a)$$

Energy - Temperature Form

$$2\xi \left(\frac{\partial f}{\partial \eta} \frac{\partial \theta}{\partial \xi} - \frac{\partial f}{\partial \xi} \frac{\partial \theta}{\partial \eta} \right) = \frac{1}{Pr} \frac{\partial}{\partial \eta} \left(\ell \frac{\partial \theta}{\partial \eta} \right) + f \frac{\partial \theta}{\partial \eta} + \alpha \ell \left(\frac{\partial^2 f}{\partial \eta^2} \right)^2 \quad (30b)$$

Equations (30a-b) have been used with a box scheme 8 but recently, a different transformation has been used for incompressible fluids which can also be applied to three-dimensional flows.^{11 13} The following new independent variables are introduced:

$$\xi = x \quad \text{and} \quad \eta = r_b^j y \sqrt{u_e(x) / \nu_o x} \quad (31)$$

and the derivatives become

$$\frac{\partial}{\partial x} = \frac{\partial}{\partial \xi} + \eta_x \frac{\partial}{\partial \eta}$$

$$\frac{\partial}{\partial y} = r_b^j \sqrt{\frac{u_e}{\nu_o x}} \frac{\partial}{\partial \eta} \quad .$$

With the new dependent variable, $F = u/u_e$, introduced, momentum Eq (15) with ρ constant becomes

$$\xi F \frac{\partial F}{\partial \xi} + \left[\xi F \eta_x + \frac{v r_b^j}{u_e} \sqrt{\frac{u_e \xi}{\nu_o}} \right] \frac{\partial F}{\partial \eta} = P(1 - F^2) + \frac{\partial}{\partial \eta} \left(\ell \frac{\partial F}{\partial \eta} \right) \quad ,$$

where

$$P = \frac{\xi}{u_e} \frac{du_e}{d\xi} \quad \text{and} \quad \ell = r_b^{2j} (\nu / \nu_o) \quad .$$

The Continuity Eq (15a) with ρ constant is satisfied by introducing the stream function which is defined as

$$r_b^j u = \frac{\partial \psi}{\partial y} \quad \text{and} \quad r_b^j v = - \frac{\partial \psi}{\partial x}$$

where

$$\psi = \sqrt{u_e \nu_o} \xi f .$$

When the foregoing relations are combined, the velocities become

$$u = u_e F \quad \text{and} \quad F = \frac{\partial f}{\partial \eta}$$

$$\sqrt{\frac{\xi}{u_e \nu_o}} r_b^j v = -\xi \frac{\partial f}{\partial \xi} - \frac{1}{2} f (1 + P) - \eta_x \xi F .$$

The governing equations for incompressible flow become the following three first-order equations:

$$\frac{\partial}{\partial \eta} (\ell S) = \xi \left(F \frac{\partial F}{\partial \xi} - S \frac{\partial f}{\partial \xi} \right) + P(F^2 - 1) - \frac{1}{2} (1 + P) f S$$

$$\frac{\partial F}{\partial \eta} = S \tag{32}$$

$$\frac{\partial f}{\partial \eta} = F$$

The boundary conditions for the foregoing equations are:

At wall (without mass transfer and specified wall temperature)

$$\eta = 0 : F = V = f = 0 \quad \text{and} \quad \theta = \theta_w \tag{33a}$$

At outer edge

$$\eta = \eta_e : F = \theta = \partial f / \partial \eta = 1 \tag{33b}$$

Incompressible Laminar Boundary Layers

Similar Solutions

Model Problem -- For boundary layers with edge conditions such that β is a constant and V and F are a function only of η , the transformed equations (29a-c) for incompressible flow become

$$\partial V / \partial \eta + F = 0 \tag{34}$$

$$\partial^2 F / \partial \eta^2 - V \partial F / \partial \eta + \beta(1 - F^2) = 0 .$$

When $u_e = \alpha x^n$ for two-dimensional flows, then $\beta = 2n/(n+1)$ which gives $\beta = 0$ for flat plate flow and $\beta = 1$ for a stagnation point. For an axisymmetric stagnation point, $\beta = 1/2$. The boundary conditions are

$$\begin{aligned} F = V = 0 \quad \text{at} \quad \eta = 0 \\ F = 1 \quad \text{at} \quad \eta \rightarrow \infty \end{aligned} \quad (35)$$

Several techniques have been used to solve the above boundary layer equations, but before we consider them, we investigate a model problem. The linear second-order differential equation with two-point boundary conditions used to represent the boundary layer problem is

$$\frac{d^2 W}{d\eta^2} + \alpha_1 \frac{dW}{d\eta} + \alpha_2 W = 0 \quad (36)$$

There are three solutions of this problem, depending on the values of α_1 and α_2 , as indicated below

When $(\alpha_1^2 - 4\alpha_2) > 0$,

$$W = C_1 e^{\tau_1 \eta} + C_2 e^{\tau_2 \eta} \quad (37a)$$

When $(\alpha_1^2 - 4\alpha_2) < 0$, $\tau_1 = a - ib$, & $\tau_2 = a + ib$,

$$W = e^{a\eta} (C_1 \cos b\eta + C_2 \sin b\eta) \quad (37b)$$

When $\alpha_1^2 = 4\alpha_2$,

$$W = (C_1 + C_2 \eta) e^{-(1/2)\alpha_1 \eta} \quad (37c)$$

where

$$\tau_1 = (1/2) \left(-\alpha_1 - \sqrt{\alpha_1^2 - 4\alpha_2} \right)$$

$$\tau_2 = (1/2) \left(-\alpha_1 + \sqrt{\alpha_1^2 - 4\alpha_2} \right)$$

The theory of two-point boundary-value problems shows that a unique solution exists for this example if $\alpha_2 \leq 0$.¹⁶ If the following boundary conditions, are used, the solution for this case becomes

$$W(0) = 1 \text{ and } W(\infty) = 0 ,$$

$$W = e^{\tau_1 \eta} . \quad (38)$$

Since $\alpha_2 < 0$, then $\tau_1 < 0$, while $\tau_2 > 0$. Therefore, C_2 must be zero to satisfy the boundary condition at $\eta = \infty$. The finite-difference solution of Eq (36) with central differences and a uniform grid, $\Delta\eta = \eta_j - \eta_{j-1}$, gives the equation

$$-AW_{j-1} + BW_j - CW_j = 0 \quad j = 2, 3, 4, \dots, J-1 , \quad (39)$$

where

$$A = 1 - (1/2) \Delta\eta \alpha_1$$

$$B = 2 - \Delta\eta^2 \alpha_2$$

$$C = 1 + (1/2) \Delta\eta \alpha_1 .$$

The exact solution of difference Eq (39) is of the form

$$W_j = \kappa \tau^j ,$$

where τ is the solution of the characteristic equation

$$-C\tau^2 + B\tau - A = 0 .$$

For the case $\alpha_2 = 0$, the finite difference solution is

$$W_j = \left(\frac{2 - \Delta\eta \alpha_1}{2 + \Delta\eta \alpha_1} \right)^{j-1} \quad j = 1, 2, 3, \dots \quad (40)$$

$$= (1 - \Delta\eta \alpha_1 + (1/2) \Delta\eta^2 \alpha_1^2 + \dots)^{j-1} .$$

The exact solution given by Eq (38) for this case becomes

$$W = e^{-\alpha_1 \eta} = \left(e^{-\alpha_1 \Delta\eta} \right)^{j-1} . \quad (41)$$

The finite-difference approximation (40) is the same as the exact result for terms up through $\Delta\eta^2$. The more interesting property of the difference equation (40) is when the following condition is not satisfied:

$$(1/2)\alpha_1\Delta\eta < 1 \quad . \quad (42)$$

For this condition, the solution will change sign according to whether j is even or odd. This oscillatory behavior contrasts with the monotonic behavior of the exact solution given by Eq (41). The condition in Eq (42) is stated by Keller¹⁶ in order to have a unique solution while Roache¹⁷ gives this condition in order to avoid wiggles. This restriction has resulted because the governing differential equation has been approximated with central differences.

Tridiagonal Solution Method -- Difference Eq (39) is scalar tridiagonal and can be solved numerically with the Thomas algorithm. These equations, with boundary conditions at the ends of the interval, are then written as

$$\begin{aligned} B_1 W_1 - C_1 W_2 &= D_1 \\ -A_j W_{j-1} + B_j W_j - C_j W_{j+1} &= D_j \quad j = 2, 3, \dots, J-1 \\ -A_J W_{J-1} + B_J W_J &= D_J \quad . \end{aligned} \quad (43)$$

The first relation represents boundary conditions at the wall and the last, conditions at the other edge. These relations are more general than is needed for simple boundary layer flows. The above relations can be written in matrix form as follows:

$$AW = D$$

where

$$W = \begin{bmatrix} W_1 \\ W_2 \\ \cdot \\ \cdot \\ \cdot \\ W_J \end{bmatrix} \quad D = \begin{bmatrix} D_1 \\ D_2 \\ \cdot \\ \cdot \\ \cdot \\ D_J \end{bmatrix}$$

$$A = \begin{bmatrix} B_1 & -C_1 & & & & \\ -A_2 & B_2 & -C_2 & & & \\ & -A_3 & B_3 & -C_3 & & \\ & & - & - & - & \\ & & & - & - & \\ & & & & -A_{J-1} & B_{J-1} & -C_{J-1} \\ & & & & & -A_J & B_J \end{bmatrix}$$

The above matrix is factored into a lower diagonal matrix L and upper diagonal matrix U as follows:

$$A = LU = \begin{bmatrix} \alpha_1 & & & & & \\ -A_2 & \alpha_2 & & & & \\ & -A_3 & \alpha_3 & & & \\ & & - & - & & \\ & & & - & - & \\ & & & & -A_J & \alpha_J \end{bmatrix} \begin{bmatrix} 1 & -E_1 & & & & \\ & 1 & -E_2 & & & \\ & & 1 & -E_3 & & \\ & & & - & - & \\ & & & & 1 & -E_{J-1} \\ & & & & & 1 \end{bmatrix}$$

This decomposition of matrix A requires that the following relations be satisfied:

$$\begin{aligned} \alpha_1 &= B_1 \\ E_1 &= B_1^{-1} C_1 \\ \alpha_j &= B_j - A_j E_{j-1} & j = 2, 3, \dots, J \\ E_j &= \alpha_j^{-1} C_j & j = 2, 3, \dots, J-1 \end{aligned} \tag{44}$$

An intermediate solution, e , is obtained for

$$Le = D$$

where

$$e = \begin{bmatrix} e_1 \\ e_2 \\ \cdot \\ \cdot \\ \cdot \\ e_J \end{bmatrix}$$

and

$$\begin{aligned} e_1 &= \alpha_1^{-1} D_1 \\ e_j &= \alpha_j^{-1} (D_j + A_j e_{j-1}) \quad j = 2, 3, \dots, J \end{aligned} \quad (45)$$

The desired solution for W is obtained from

$$UW = e$$

which gives

$$\begin{aligned} W_J &= e_J \\ W_j &= E_j W_{j+1} + e_j \quad j = J-1, J-2, \dots, 2, 1 \end{aligned} \quad (46)$$

The procedure is to solve for E_j and e_j from the relations in Eq (44) and (45) for $j = 1, 2, 3, \dots$. Then the desired solution is obtained from (46) with decreasing values of j . This method is called the "double-sweep method" in Russia because of the technique used to solve the equation. Also, if governing Eqs (43) are of the block tridiagonal type (W and D are vectors while A , B , and C are square matrices), the foregoing procedure is valid. All steps become appropriate vector and matrix operations and requires either that the inverse of matrices to be determined or a system of linear equations be solved with LU decomposition. This is described later.

In the case of scalar tridiagonal Eqs (43), the solution is obtained with the subroutine TRIDAG:

```

      SUBROUTINE TRIDAG (A,B,C,D,JMAX,W,M,E,EL,MM)
      DIMENSION A(JMAX),B(JMAX),C(JMAX),D(JMAX)
      DIMENSION E(JMAX), EL(JMAX), W(MM,JMAX)
      JM1 = JMAX - 1
      E(1) = 0.0
      EL(1) = W(M,1)
      DO 10 J = 2, JM1
      ALPHA = - A(J) * E(J-1) * B(J)
      E(J) = C(J) / ALPHA
      EL(J) = ( D(J) * A(J) * EL(J-1) ) / ALPHA
10  CONTINUE
      DO 20 J = 2, JM1
      JK = JM1 * 2 - J
      W(M,JK) = E(JK) * W(M,JK+1) * EL(JK)
20  CONTINUE
      RETURN
      END

```

This subroutine is appropriate for the following boundary conditions:

$$\left. \begin{array}{l} W_1 = D_1 \\ W_J = e_J = D_J \end{array} \right\} \begin{array}{l} \text{These quantities are specified before the} \\ \text{subroutine is called.} \end{array}$$

The notation used in this subroutine is

<u>FORTRAN Name</u>	<u>Symbol</u>
J	j
JMAX	J
A(J), B(J), C(J), D(J)	A_j, B_j, C_j, D_j
W(M, J)	W_j (M for various dependent variables)
ALPHA	α_j
E(J)	E_j
EL(J)	e_j

When difference Eqs (43) are block-tridiagonal, the procedure for solving them becomes more complex and utilizes several subroutines. For this case, it is advantageous, for computer storage requirements, to obtain E_j and e_j after A_j, B_j, C_j , and D_j have been determined in order to avoid storing these later quantities. The subroutine EEL determines E_j and e_j and requires the use of an additional subroutine, AXB. This subroutine solves a system of linear equations, $Ax = b$, with the LU decomposition method. Equations (45) are in this form, where $A = \alpha_j$, $x = e_j$, and $b = D_j + A_j e_{j-1}$. Equations (44) are also in this form when they are written in the following manner:

$$A = \alpha_j$$

$x = n^{\text{th}}$ column of E_j

$b = n^{\text{th}}$ column of C_j .

Subroutine AXB will be described later in more detail; subroutine EEL is

```

SUBROUTINE EEL(J,MM,JMAX,E,EL,AM,BM,CM,DM,IN,AL,BE)
  DIMENSION IN(MM),E(MM,MM,JMAX),EL(MM,JMAX)
  DIMENSION AM(MM,MM),BM(MM,MM),CM(MM,MM),DM(MM)
  DIMENSION AL(MM,MM),BE(MM)
  DO 30 M = 1, MM
    TP = 0.0
    DO 20 N = 1, MM
      T1 = 0.0
      IF( J.EQ. 1 ) GO TO 10
      TP = TP + AM(M,N) * EL(N,J-1)
      DO 5 K = 1, MM
        T1 = T1 + AM(M,K) * E(K,N,J-1)
    5 CONTINUE
    AL(M,N) = BM(M,N) - T1
  20 CONTINUE
    EL(M,J) = DM(M) + TP
  30 CONTINUE
    DO 50 M = 1, MM
      DO 40 N = 1, MM
        40 E(M,N,J) = CM(M,N)
    50 CONTINUE
    CALL AXB (MM,MM,AL,E(1,1,J),BE,0,IN)
    CALL AXB (MM,1,AL,EL(1,J),BE,1,IN)
    RETURN
  END

```

The following notation is used:

FORTRAN Name	Symbol
MM	$m = \text{order of matrix}$
AM(MM, MM), BM(MM, MM), CM(MM, MM), DM(MM)	A_j, B_j, C_j, D_j
AL(MM, MM)	α_j
Stored in EL(MM, J)	$D_j + A_j E_{j-1}$
E(MM, MM, J), EL(MM, J)	E_j, e_j

After the parameters, E_j and e_j , have been determined for all values of j , the original Eqs (43) can be solved with Eqs (46). These algebraic operations are performed in the subroutine SOLU where the following notation is used:

FORTTRAN Name

Symbol

W(M, J)

$(w_1)_j, (w_2)_j, (w_3)_j, \dots$

```

SUBROUTINE SOLU(W,JMAX,MM,E,EL)
DIMENSION W(MM,JMAX), E(MM,MM,JMAX), EL(MM,JMAX)
DO 40 M = 1,MM
W(M,JMAX) = EL(M,JMAX)
40 CONTINUE
DO 50 J1 = 2,JMAX
J = JMAX + 1 - J1
DO 46 M = 1,MM
SUM = 0.0
DO 44 K = 1,MM
SUM = SUM + E(M,K,J) * W(K,J+1)
44 CONTINUE
W(M,J) = SUM * EL(M,J)
46 CONTINUE
50 CONTINUE
RETURN
END

```

The subroutine AXB can be a library program or a specialized version of the one given by Forsythe and Moler¹⁸ described below. This subroutine solves a system of linear equations of the form $Ax = b$ which is written in the following manner in order to illustrate the technique and explain the subroutines used:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad (47)$$

In this subroutine, the rows are interchanged in order to avoid dividing by zero; this process is referred to as "pivoting." The LU decomposition of the above equations will neglect pivoting in order to simplify the illustration. The first step requires the following operations:

- Multiply first row by $m_{21} = a_{21}/a_{11}$ and subtract from second row
- Multiply first row by $m_{31} = a_{31}/a_{11}$ and subtract from third row

Equations (47) then become the following:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} \\ 0 & a_{32}^{(2)} & a_{33}^{(2)} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2^{(2)} \\ b_3^{(2)} \end{bmatrix}$$

where

$$a_{22}^{(2)} = a_{22} - m_{21}a_{12} \quad b_2^{(2)} = b_2 - m_{21}b_1$$

$$a_{23}^{(2)} = a_{23} - m_{21}a_{13}$$

$$a_{32}^{(2)} = a_{32} - m_{31}a_{12} \quad b_3^{(2)} = b_3 - m_{31}b_1$$

$$a_{33}^{(2)} = a_{33} - m_{31}a_{13} .$$

The next step is to

Multiply second row by $m_{32} = a_{32}^{(2)} / a_{22}^{(2)}$ and subtract from third row.

The equations become

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} \\ 0 & 0 & a_{33}^{(3)} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3^{(3)} \end{bmatrix}$$

where

$$a_{33}^{(3)} = a_{33}^{(2)} - m_{32}a_{23}^{(2)} \quad b_3^{(3)} = b_3^{(2)} - m_{32}b_2^{(2)} .$$

The original matrix A has been decomposed into the lower and upper triangular matrices where $A = LU$

$$L = \begin{bmatrix} 1 & 0 & 0 \\ m_{21} & 1 & 0 \\ m_{31} & m_{32} & 1 \end{bmatrix} \quad U = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} \\ 0 & 0 & a_{33}^{(3)} \end{bmatrix} \quad (48)$$

These two matrices are stored as one matrix UL where the diagonal of the number 1 is neglected. The solution of original Eq (47) is obtained with back substitution as follows:

$$\begin{aligned}
 x_3 &= b_3^{(3)} / a_{33}^{(3)} \\
 x_2 &= (b_2^{(2)} - a_{23}^{(2)} x_3) / a_{22}^{(2)} \\
 x_1 &= (b_1 - a_{12} x_2 - a_{13} x_3) / a_{11}
 \end{aligned}
 \tag{49}$$

The subroutine AXB performs the above operations by calling subroutines DECOMP and SOLVE. The main subroutine AXB calls DECOMP only if INIT = 0; this allows the solutions of Eqs (47) with different values of b without repeating the decomposition of A. The notation used in these subroutines is the following:

<u>FORTRAN Name</u>	<u>Symbol</u>
N	Order of matrix A
M	Number of b's
A(N, N)	A
B(N, N)	All b's
X(N)	Temporary storage for b's
INIT	<div style="display: inline-block; vertical-align: middle;"> <div style="font-size: 2em; vertical-align: middle;">{</div> <div style="display: inline-block; vertical-align: middle;"> <div>0 Call DECOMP</div> <div>1 Use previously obtained values</div> </div> </div>
IPS	Row interchange indices
UL(N, N)	Same as A and decomposition results stored in this matrix.

The subroutine DECOMP first determines the largest coefficient in the columns of Matrix A as the decomposition is being performed, and the indices of the rows are stored in IPS(N) as the pivoting is taking place. The final part of this subroutine finds the coefficients in matrices L and U and stores these in matrix UL. The subroutine SOLVE first finds the coefficients b , $b_2^{(2)}$, $b_3^{(3)}$, etc, and stores these into X(N) and finally into B(N). The last part of the subroutine obtains the solution as given by Eqs (49) by back substitution.

AXB

```

SUBROUTINE AXB (N,M,A,B,X,INIT,IPS)
  DIMENSION A(N,N), B(N,M), IPS(N), X(N)
  IF( INIT .EQ. 0 ) CALL DECOMP(N, A, IPS )
  DO 10 I = 1,M
    CALL SOLVE (N,A,B(1,I),X,IPS)
10 CONTINUE
  RETURN
END

```

DECOMP

```

SUBROUTINE DECOMP ( N, UL, IPS )
  DIMENSION UL(N,N), IPS(N)
  DO 5 I = 1, N
    IPS(I) = I
5 CONTINUE
  NM1 = N - 1
  DO 17 K = 1, NM1
    BIG = 0.0
    DO 11 I = K, N
      IP = IPS(I)
      SIZE = ABS( UL(IP,K) )
      IF( SIZE - BIG ) 11, 11, 10
10 BIG = SIZE
      IDXPIV = I
11 CONTINUE
      IF( IDXPIV - K ) 14, 15, 14
14 J = IPS(K)
      IPS(K) = IPS( IDXPIV )
      IPS( IDXPIV ) = J
15 KP = IPS(K)
      PIVOT = UL(KP,K)
      KP1 = K + 1
      DO 16 I = KP1, N
        IP = IPS(I)
        EM = - UL(IP,K) / PIVOT
        UL(IP,K) = - EM
        DO 16 J = KP1, N
          UL(IP,J) = UL(IP,J) + EM * UL(KP,J)
16 CONTINUE
17 CONTINUE
  RETURN
END

```

SOLVE

```

SUBROUTINE SOLVE( N, UL, B, X, IPS )
  DIMENSION UL(N,N), B(N), X(N), IPS(N)
  NP1 = N + 1
  IP = IPS(1)
  X(1) = B(IP)
  DO 2 I = 2, N
    IP = IPS(I)
    IM1 = I - 1
    SUM = 0.0
    DO 1 J = 1, IM1
      SUM = SUM + UL(IP,J) * X(J)
1  X(I) = B(IP) - SUM
    IP = IPS(N)
    B(N) = X(N) / UL(IP,N)
    DO 4 IBACK = 2, N
      I = NP1 - IBACK
      IP = IPS(I)
      IP1 = I + 1
      SUM = 0.0
      DO 3 J = IP1, N
        SUM = SUM + UL(IP,J) * B(J)
3  B(I) = ( X(I) - SUM ) / UL(IP,I)
4  RETURN
END

```

Shooting Method -- Model Eq (36) can be solved as an initial-value problem in what is referred to as a "shooting method." This model illustrates the difficulty of solving the differential equation with this approach. The exact solution is given by Eq (38) when the boundary conditions are $W(0) = 1$ and $W(\infty) = 0$. The shooting method uses boundary conditions

$$W(0) = 1$$

$$W'(0) = \phi \quad (\text{assumed}) \quad .$$

When the exact solution, Eq (37a), is used with these boundary conditions, the coefficients c_1 and c_2 , become

$$c_1 = (\phi - \tau_2)/(\tau_1 - \tau_2)$$

$$c_2 = (\tau_1 - \phi)/(\tau_1 - \tau_2) \quad .$$

If the assumed value of $\phi \neq \tau_1$, then c_2 is non-zero and this term in the solution of Eq (37a) grows exponentially with η . This behavior makes solving this type of equation with a shooting method very difficult. Special codes have been developed to handle this problem, such as, the code by Scott and Watts¹⁹ which utilizes an orthonormalization.

Central Difference Scheme -- Probably the simplest method to solve the similar boundary layer equations is with the F-V form of equations with central differences. Continuity Eq (34) is evaluated at the point $(j - 1/2)$ with the terms in the equation evaluated with second-order accuracy (truncation error of the difference equation is of the order of the step-size squared) which gives

$$(\partial V / \partial \eta)_{j-1/2} = (V_j - V_{j-1}) / \Delta \eta + O(\Delta \eta^2) \quad (50)$$

$$F_{j-1/2} = (F_j + F_{j-1}) / 2 + O(\Delta \eta^2)$$

where $\Delta \eta = (\eta_j - \eta_{j-1})$. The resulting expression for the continuity equation becomes

$$\frac{1}{2} \Delta \eta (F_{j-1} + F_j) - V_{j-1} + V_j = 0 \quad j = 2, 3, \dots, J \quad (51)$$

Momentum Eq (34) is evaluated at point j with the following central differences on a uniform grid:

$$\begin{aligned}\partial^2 F / \partial \eta^2 &= (F_{j+1} - 2F_j + F_{j-1}) / \Delta \eta^2 + O(\Delta \eta^2) \\ \partial F / \partial \eta &= (F_{j+1} - F_{j-1}) / (2\Delta \eta) + O(\Delta \eta^2) .\end{aligned}\tag{52}$$

If the above differences are substituted into the momentum equation, the resulting difference equation will be nonlinear in terms of the unknown variables. Local linearization is used which is called the "Newton-Raphson method."* The linearization can be obtained from Taylor's formula for functions of several variables:

$$f(x, y) = f(a, b) + \frac{\partial f}{\partial x} (x - a) + \frac{\partial f}{\partial y} (y - b) + \dots\tag{53}$$

where

$$f = V \partial F / \partial \eta ,$$

$$x = V ,$$

$$y = \partial F / \partial \eta ,$$

$$a = \overline{V} ,$$

and

$$b = \partial \overline{F} / \partial \eta ,$$

and the resulting approximation is

$$V \partial F / \partial \eta = \overline{V} \partial F / \partial \eta + V \partial \overline{F} / \partial \eta - \overline{V} \partial \overline{F} / \partial \eta ,\tag{54}$$

when

$$f = F^2 ,$$

$$x = F ,$$

and

$$a = \overline{F} ,$$

and the linearization gives

$$F^2 = 2\overline{F} F - \overline{F}^2 ,\tag{55}$$

* Also called Newton Method, quasilinearization, linear correction

In Eqs (54) and (55) the barred quantities are evaluated from a previous iteration and the unknown variable F and V appear in a linear form. The above linearizations can be obtained with another method as follows:

$$\begin{aligned} xy &= ab + b \delta x + a \delta y + \dots \\ &= bx + ay - ab + \dots \end{aligned} \quad (56)$$

where $x = a + \delta x$ and $y = b + \delta y$. Equation (54) can be obtained from Eq (56). With $x = y = F$ and $a = b = \bar{F}$, Eq (56) also gives Eq (55).

With the linearization relations of Eqs (54) and (55) and the difference relations of Eq (52), momentum Eq (34) becomes

$$\begin{aligned} -A_j F_{j-1} + B_j F_j - C_j F_{j+1} + a_j V_j &= D_j \quad j = 2, 3, \dots, J-1 \\ A_j &= (1 + \bar{V}_j \Delta\eta/2)/2 \\ B_j &= 1 + \Delta\eta^2 \beta \bar{F}_j \\ C_j &= (1 - \bar{V}_j \Delta\eta/2)/2 \\ D_j &= (\Delta\eta^2/2) \left[\beta + \beta \bar{F}_j^2 + V_j (F_{j+1} - F_{j-1}) / (2\Delta\eta) \right] \\ a_j &= (\bar{F}_{j+1} - \bar{F}_{j-1}) \Delta\eta/4 \end{aligned} \quad (57)$$

Difference Eqs (51) and (57) are completed with boundary conditions

$$\begin{aligned} V_1 &= F_1 = 0 \\ F_J &= 1 \end{aligned} \quad (58)$$

In order to solve momentum Eq (57), iteration is necessary because the coefficients themselves depend on variables evaluated from a previous iteration. Also, the momentum equation is more complex than scalar tridiagonal Eq (43). One approach is to solve Eq (57) with $a_j V_j$ assumed from a previous iteration, making the difference equation into a scalar tridiagonal equation.

Two sample worked problems, one of the uncoupled and the other of the coupled method are printed below.

1. Worked Problem: Uncoupled Method -- The solution of this problem is obtained with the following program. The notation used is the following:

<u>FORTTRAN Name</u>	<u>Symbol</u>
A(J), B(J), C(J), D(J), AS(J)	A_j, B_j, C_j, D_j, a_j
W(1, J), W(2, J)	F_j, V_j
ETA(J)	η_j
JMAX	J
DETA	$\Delta\eta$
ETA_E	η_e
CF	$c_f \sqrt{\text{Re}_x}$

```

C      SIMILAR SOLUTION      UNCOUPLED SCHEME      CENTRAL DIFFERENCES
      DIMENSION A(161), B(161), C(161), D(161), E(161), EL(161)
      DIMENSION W(2,161), ETA(161), AS(161)
      JMAX = 21
      JMAX = 161
      JMAX = 41
      JMAX = 81
      BETA = 0.0
      ETA_E = 6.0
      JM1 = JMAX - 1
      DETA = ETA_E / ( JMAX - 1.0 )
      ETA(1) = 0.0
      W(1,1) = 0.0
      W(2,1) = 0.0
      DO 10 J = 2, JMAX
      ETA(J) = ETA(J-1) * DETA
      W(1,J) = 1.0
10    W(2,J) = W(2,J-1) - 0.5 * DETA * ( W(1,J) + W(1,J-1) )
      PRINT 9500, (J, ETA(J), W(1,J), W(2,J), J=1, JMAX)
9500  FORMAT(10X, I3, 2X, E12.5, 2X, E12.5, 2X, E12.5)
      DO 40 ITER = 1, 100
      DO 20 J = 2, JM1
      A(J) = 0.5 * ( 1.0 + 0.5 * DETA * W(2,J) )
      B(J) = 1.0 + ( DETA**2 ) * BETA * W(1,J)
      C(J) = 0.5 * ( 1.0 - 0.5 * DETA * W(2,J) )
      D(J) = 0.5 * ( DETA**2 ) * ( BETA * ( 1.0 * W(1,J) **2 ) * W(2,J) * 0.5 *
1    ( W(1,J+1) - W(1,J-1) ) / DETA )
      AS(J) = 0.25 * ( W(1,J+1) - W(1,J-1) ) * DETA
      D(J) = D(J) - AS(J) * W(2,J)
20    CONTINUE
      CALL TRIDAG (A,B,C,D,JMAX,W,1,E,EL,2)
      DO 30 J = 2, JMAX
      W(2,J) = W(2,J-1) - 0.5 * DETA * ( W(1,J) + W(1,J-1) )
      CF = SQRT(2.0) * ( 2.0 * W(1,2) - 0.5 * W(1,3) ) / DETA
      PRINT 9600, ITER, CF
9600  FORMAT( 20X, I3, 5X, E16.9)
40    CONTINUE
      PRINT 9500, (J, ETA(J), W(1,J), W(2,J), J=1, JMAX)
      RETURN
      END

```

This program first assumes the dependent variable $F_j = 1$, except at the wall, and determines V_j from Eq (51). Then the coefficients in Eq (57) are determined and the solution obtained with sub-routine TRIDAG. The solution for V_j is obtained from Eq (51) and the solution is iterated 20 times. The skin friction at the wall is determined from

$$C_f \sqrt{\text{Re}_x} = \phi \frac{\partial F}{\partial \eta} \quad (59)$$

where

$$C_f = \tau_w / \left(\frac{1}{2} \rho_e u_e^2 \right) \quad \varphi = \begin{cases} \sqrt{4/(2-\beta)} & 2-D \text{ Flow } (u_e = Kx^n \text{ and } n = \beta/(2-\beta)) \\ 2\sqrt{1+j} & \text{Stagnation Point Flow} \end{cases}$$

$$Re_x = \rho_e u_e x / \mu_e$$

$$\frac{\partial F}{\partial \eta} = \left(2F_2 - \frac{1}{2} F_3 \right) \Delta \eta \quad .$$

The solution is obtained with $\eta_e = 6$, $\beta = 0$ and for various number of grid points across the boundary layer. The results for the skin friction is

<u>JMAX</u>	<u>$C_f \sqrt{Re_x}$</u>	<u>% Error</u>
21	0.667125	4.53×10^{-1}
41	0.664604	7.36×10^{-2}
81	0.664205	1.35×10^{-2}
161	0.664134	2.80×10^{-3}
∞	{ 0.664110 (extrapolated) 0.66411538 (more accurate value used)	

The "exact" value of the skin friction can be obtained with Richardson extrapolation which, for a second-order method with the step-size halved, is the following:

$$W_{\Delta\eta \rightarrow 0} = W_c \left(\frac{1}{2} \Delta\eta \right) + \frac{1}{3} \left[W_c \left(\frac{1}{2} \Delta\eta \right) - W_c (\Delta\eta) \right] \quad . \quad (60)$$

In this relation, $W_c(k\Delta\eta)$ is the computed solution with step-size, $k\Delta\eta$, and W represents the skin friction parameter. These results are also presented in Figure 4 along with results from other methods which will be described later. Another item is the rate of convergence of the iteration procedure with the presented method. The error at each iteration for the skin friction is given in the table below where JMAX = 161 and other conditions are the same:

<u>Iteration No.</u>	<u>% Error of $C_f \sqrt{Re_x}$</u>	<u>$E^{(i+1)}/E^{(i)}$</u>
1	67.5	
2	13.5	0.20
3	3.70	0.27
4	1.06	0.29
5	3.08×10^{-1}	0.30
6	9.01×10^{-2}	0.29
7	2.63×10^{-2}	0.29
8	7.71×10^{-3}	0.29

Figure 4. Accuracy of Schemes for Similar Solutions

The convergence rate can be expressed in the form

$$E^{(i+1)} = K(E^{(i)})^n \quad (61)$$

which (i) is the iteration number. This method results in linear convergence with $n = 1$, and the Newton-Raphson procedure gives quadratic convergence with $n = 2$. This loss of quadratic convergence has resulted from the uncoupling of the continuity and momentum equations. This behavior will be demonstrated later.

The final step is to determine if oscillatory solutions occur as were predicted in the model problem. The flat plate boundary layer case is investigated when there is suction at the surface such that $V(0) = -10$. The velocity, F , is shown in Figure 5 for two values of JMAX; oscillations occur when JMAX = 21. The condition represented in Eq (42) for two cases are given below:

JMAX	$\frac{1}{2} \Delta \eta V_{MAX}$
21	2.39
41	1.19

Figure 5. Flat Plate Velocity with Suction

In the first case, oscillations occur but in the second, there are no wiggles. Therefore, the condition of Eq (42) gives a reasonable estimate of when oscillations will appear. It should be noticed that there are very few grid points within the boundary layer when there are oscillations.

2. Worked Problem: Coupled Method -- We will now solve the previous problem, treating the continuity and momentum equations in a coupled manner. For this case Difference Eqs (51) and (57) are written in block tridiagonal form as given by Eq (43). When the momentum equation is treated first equation, the matrix coefficients become

$$W_j = \begin{bmatrix} F_j \\ V_j \end{bmatrix} \quad B_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad C_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad D_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (62)$$

$$A_j = \begin{bmatrix} A_j & 0 \\ -\frac{1}{2} \Delta \eta & 1 \end{bmatrix} \quad B_j = \begin{bmatrix} B_j & a_j \\ \frac{1}{2} \Delta \eta & 1 \end{bmatrix} \quad C_j = \begin{bmatrix} C_j & 0 \\ 0 & 0 \end{bmatrix} \quad D_j = \begin{bmatrix} D_j \\ 0 \end{bmatrix}$$

$$A_J = \begin{bmatrix} 0 & 0 \\ -\frac{1}{2} \Delta\eta & 1 \end{bmatrix} \quad B_J = \begin{bmatrix} 1 & 0 \\ \frac{1}{2} \Delta\eta & 1 \end{bmatrix} \quad D_J = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

In the matrices, the scalar coefficients A_j , B_j , C_j , D_j , and a_j are defined with Eq. (57).
The program and subroutine used to solve the coupled similar equations is the following:

```

PROGRAM BLC (INPUT,OUTPUT)
C
C   SIMILAR SOLUTION - COUPLED EQUATIONS - F-V VARIABLES
C
COMMON /1/ DETA, BETA
DIMENSION E(2,2,161), FL(2,161)
DIMENSION AL(2,2), BE(2)
DIMENSION AM(2,2), RM(2,2), CM(2,2), DM(2)
DIMENSION W(2,161), ETA(161), IN(2)
JMAX = 161
JMAX = 21
BETA = 0.0
ETA0 = 6.0
MM = 2
DETA = ETA0 / ( JMAX - 1.0 )
ETA(1) = 0.0
W(1,1) = W(2,1) = 0.0
DO 10 J = 2, JMAX
  ETA(J) = ETA(J-1) + DETA
  W(1,J) = 1.0
10 W(2,J) = W(2,J-1) - 0.5 * DETA * ( W(1,J) + W(1,J-1) )
PRINT 9500, (J,ETA(J),W(1,J),W(2,J),J=1,JMAX)
9500 FORMAT(10X,I3,3(2X,E12.5))
DO 40 ITER = 1,20
  CALL COEF(MM,JMAX,E,EL,AM,RM,CM,DM,IN,W,AL,BE)
  CALL SOLU(W,JMAX,MM,E,EL)
  CF = SQRT(2.0) * ( 2.0*W(1,2) - 0.5*W(1,3) ) / DETA
  PRINT 9600, ITER, CF
9600 FORMAT( 20X,I3,5X,E16.9)
40 CONTINUE
PRINT 9500, (J,ETA(J),W(1,J),W(2,J),J=1,JMAX)
RETURN
END

```

```

SUBROUTINE COEF(MM,JMAX,E,EL,AM,BM,CM,DM,IN,W,AL,BE)
COMMON /1/ DETA, BETA
DIMENSION IN(MM),E(MM,MM,JMAX),EL(MM,JMAX),W(MM,JMAX)
DIMENSION AM(MM,MM),BM(MM,MM),CM(MM,MM),DM(MM)
DIMENSION AL(MM,MM),BF(MM)
DO 15 M = 1, MM
DO 14 N = 1, MM
14 AM(M,N) = BM(M,N) = CM(M,N) = 0.0
15 DM(M) = 0.0
BM(1,1) = BM(2,2) = 1.0
CALL EEL(1,MM,JMAX,E,EL,AM,BM,CM,DM,IN,AL,BE)
DO 20 J = 2, JMAX
AM(2,1) = - 0.5*DETA
AM(2,2) = 1.0
BM(2,1) = 0.5*DETA
BM(2,2) = 1.0
IF( J.EQ. JMAX ) GO TO 18
AM(1,1) = 0.5 * ( 1.0 + 0.5 * DETA * W(2,J) )
BM(1,1) = 1.0 + ( DETA**2 ) * BETA * W(1,J)
BM(1,2) = 0.25 * ( W(1,J+1) - W(1,J-1) ) * DETA
CM(1,1) = 0.5 * ( 1.0 - 0.5 * DETA * W(2,J) )
DM(1) = 0.5*(DETA**2)*(BETA*(1.0+W(1,J)**2)+W(2,J)*0.5*
1 (W(1,J+1)-W(1,J-1))/DETA)
GO TO 20
18 CONTINUE
AM(1,1) = 0.0
BM(1,1) = 1.0
BM(1,2) = 0.0
DM(1) = 1.0
20 CALL EEL(J,MM,JMAX,E,EL,AM,BM,CM,DM,IN,AL,BE)
RETURN
END

```

This program is similar to the uncoupled code except subroutines COEF, EEL, and SOLU are used. The coefficients in Eq (62) are evaluated in subroutine COEF and subroutine EEL is called to obtain quantities E_j and e_j used in solving the block tridiagonal equations (see Eqs (43) to (45)). Then the solution is obtained in the main program by calling SOLU at the point where Eq (46) is solved. The new FORTRAN notation introduced for this problem is the following:

<u>FORTRAN Name</u>	<u>Symbol</u>
AM(M, M), BM(M, M)	{ Matrix Coefficients
CM(M, M), DM(M)	
E(M, M, J), EL(M, J)	E_j, e_j

The solutions obtained with this code are the same as those obtained with the uncoupled code, but the convergence of the iteration process is different. The error of the skin friction as a function of the number of iterations follows:

Iteration No.	% Error of $C_f \sqrt{Re}_x$	$E^{(i+1)} / (E^{(i)})^2$
1	64.6	
2	4.33	1.04×10^{-3}
3	8.76×10^{-2}	4.67×10^{-3}
4	5.21×10^{-5}	6.79×10^{-3}

In the coupled scheme, results converge much faster than when the equations are handled in an uncoupled manner. The listing above shows that quadratic convergence is approximately obtained as expected.

The foregoing uses the general method to solve block tridiagonal equations. For the present case, the procedure can be greatly simplified with the Davis modified tridiagonal algorithm. To solve the following finite-difference equations:

$$\begin{aligned}
 -A_j F_{j-1} + B_j F_j - C_j F_{j+1} + a_j V_j + b_j V_{j-1} &= D_j \\
 V_j &= V_{j-1} - c_j(F_j + F_{j-1}) + d_j
 \end{aligned} \tag{63}$$

These equations are a particular form of the block-tridiagonal equations and a special form of the procedure given by Eqs (44) to (46). First, the following parameters are determined:

$$\left. \begin{aligned} E_J &= 0 \\ G_J &= 0 \\ e_J &= 1 \end{aligned} \right\} \text{obtained with boundary conditions at outer edge} \tag{64}$$

$$\left. \begin{aligned} E_j &= [A_j - c_j(C_j G_{j+1} - a_j)] / \mathcal{D} \\ G_j &= (C_j G_{j+1} - a_j - b_j) / \mathcal{D} \\ e_j &= [D_j + (C_j G_{j+1} - a_j) d_j + C_j e_{j+1}] / \mathcal{D} \end{aligned} \right\} j = (J-1), (J-2), \dots$$

where

$$\mathcal{D} = B_j - C_j E_{j+1} + c_j(C_j G_{j+1} - a_j)$$

Then the solution is obtained from

$$\left. \begin{array}{l} F_1 = 0 \\ V_1 = 0 \end{array} \right\} \text{boundary conditions at wall} \quad (65)$$

$$\left. \begin{array}{l} F_j = E_j F_{j-1} + G_j V_{j-1} + e_j \\ V_j = V_{j-1} - c_j (F_j + F_{j-1}) + d_j \end{array} \right\} \quad j = 2, 3, 4, \dots, J$$

Equations (64) and (65) are solved in subroutine TRIDM where the following notation is used:

FORTRAN Name	Symbol
E(J), EL(J), G(J)	E_j, e_j, G_j
A(J), B(J), C(J), D(J)	A_j, B_j, C_j, D_j
AS(J), BS(J), CS(J), DS(J)	a_j, b_j, c_j, d_j
W(1,J), W(2,J)	F_j, V_j

```

SUBROUTINE TRIDM(JMAX,A,B,C,D,AS,BS,CS,DS,W,E,EL,G)
DIMENSION A(JMAX),B(JMAX),C(JMAX),D(JMAX)
DIMENSION AS(JMAX),BS(JMAX),CS(JMAX),DS(JMAX)
DIMENSION W(2,JMAX), E(JMAX), EL(JMAX), G(JMAX)
JM2 = JMAX - 2
E(JMAX) = 0.0
G(JMAX) = 0.0
EL(JMAX) = 1.0
DO 20 JK=1,JM2
  J = JMAX - JK
  DEN = B(J) - C(J)*E(J+1) + CS(J)*( C(J)*G(J+1) - AS(J) )
  E(J) = ( A(J) - CS(J)*( C(J)*G(J+1) - AS(J) ) ) / DEN
  G(J) = ( C(J)*G(J+1) - AS(J) - BS(J) ) / DEN
  EL(J) = ( D(J) + ( C(J)*G(J+1) - AS(J) ) * DS(J) + C(J)*EL(J+1) ) / DEN
20 CONTINUE
DO 10 J=2,JMAX
  W(1,J) = E(J)*W(1,J-1) + G(J)*W(2,J-1) + EL(J)
  W(2,J) = W(2,J-1) - CS(J)*( W(1,J) + W(1,J-1) ) + DS(J)
10 CONTINUE
RETURN
END

```

This subroutine replaces the subroutines EEL, SOLU, AXB, DECOMP, and SOLVE. The following program and subroutine COEF complete the necessary coding for the coupled similar solutions.

The coefficients in Eqs (63) have been evaluated with the relations after Eq (57) except $b_j = 0$. The remaining terms are obtained from Eq (51) which gives $c_j = \Delta\eta/2$ and $d_j = 0$. The coefficients are evaluated in subroutine COEF. The main program is nearly the same as the previous one used for this problem except that subroutine TRIDM is called to obtain the solution of the difference equations.

```

PROGRAM BIS (INPUT,OUTPUT,TAPE60=INPUT,TAPE61=OUTPUT)
C
C      SIMILAR BOUNDARY LAYER      F.V. VARIABLES
C      CENTRAL DIFFERENCES        COUPLED SOLUTION
C      MODIFIED TRIDIAGONAL SOLUTION
C
COMMON /1/ BETA, DELTA
DIMENSION A(161), R(161), C(161), D(161), E(161), EL(161), G(161)
DIMENSION W(2,161), ETA(161), AS(161), BS(161), CS(161), DS(161)
1 CONTINUE
READ 9000, ETAE, BETA, JMAX
9000 FORMAT (5X, 2E10.0, I10)
IF (EOF,60) 9, 35
35 CONTINUE
PRINT 8000, ETAE, BETA, JMAX
8000 FORMAT (1H1/// 15X*FAIKNER SKAN SOLUTION*//5X*ETAE = *E11.4,4X
1 *BETA = *E11.4,5X*JMAX = *I3)
DELTA = ETAE / (JMAX - 1.0)
ETA(1) = 0.0
W(1,1) = W(2,1) = 0.0
DO 10 J = 2, JMAX
ETA(J) = ETA(J-1) + DELTA
W(1,J) = 1.0
10 W(2,J) = W(2,J-1) - 0.5 * DELTA * ( W(1,J) + W(1,J-1) )
PRINT 9500, (J,ETA(J),W(1,J),W(2,J),J=1,JMAX)
9500 FORMAT(10X,I3,3(2X,E12.5))
DO 40 ITER = 1,20
CALL COFF(JMAX,A,B,C,D,AS,BS,CS,DS,W)
CALL TRIDM(JMAX,A,B,C,D,AS,BS,CS,DS,W,E,EL,G)
CF = SORT(2.0) * ( 2.0*W(1,2) - 0.5*W(1,3) ) / DELTA
PRINT 9600, ITER, CF
9600 FORMAT( 20X,I2,5X,E16.9)
40 CONTINUE
PRINT 9500, (J,ETA(J),W(1,J),W(2,J),J=1,JMAX)
GO TO 1
9 CONTINUE
RETURN
END

```

```

SUBROUTINE COFF(JMAX,A,B,C,D,AS,BS,CS,DS,W)
COMMON /1/ BETA, DELTA
DIMENSION A(JMAX),B(JMAX),C(JMAX),D(JMAX),W(2,JMAX)
DIMENSION AS(JMAX),BS(JMAX),CS(JMAX),DS(JMAX)
JM1 = JMAX - 1
DO 20 J = 2, JM1
A(J) = 0.5 * ( 1.0 + 0.5 * DELTA * W(2,J) )
B(J) = 1.0 + ( DELTA**2 ) * BETA * W(1,J)
C(J) = 0.5 * ( 1.0 - 0.5 * DELTA * W(2,J) )
D(J) = 0.5*(DELTA**2)*(BETA*(1.0+W(1,J)**2)+W(2,J)*0.5*
1 (W(1,J+1)-W(1,J-1))/DELTA)
AS(J) = 0.25*(W(1,J+1)-W(1,J-1))*DELTA
BS(J) = 0.0
CS(J) = 0.5 * DELTA
DS(J) = 0.0
20 CONTINUE
CS(JMAX) = 0.5 * DELTA
DS(JMAX) = 0.0
RETURN
END

```

Stream Function Variables -- The similar solutions are not easily obtained from this form of governing equations except for flat plate flow. For the complete Falkner-Skan similar solutions, governing Eq (25) requires further transformation with a new dependent variable. For the case of the flat plate, the stream functions at the wall and edge are

$$\psi_I = 0 \quad \text{and} \quad \psi_E = \alpha \sqrt{x} \quad .$$

The governing Eq (25) becomes

$$\omega \frac{\partial F}{\partial \omega} + \frac{\partial}{\partial \omega} \left(KF \frac{\partial F}{\partial \omega} \right) = 0 \quad (66)$$

where

$$F = u/u_e$$

$$K = 2\rho\mu u_e / \alpha^2 \quad .$$

The normal coordinate y is obtained from Eq (20) and becomes, with Eq (23)

$$y = \int_0^\omega \frac{\psi_E}{\rho u} d\omega \quad . \quad (67)$$

The similar coordinate η is obtained from Eq (27) and, with Eq (67) becomes

$$\eta = \frac{1}{\sqrt{K}} \int_0^\omega \frac{d\omega}{F} \quad . \quad (68)$$

The value of α or K determines where the outer edge of the boundary layer is located. The value of the stream function is obtained from Eq (20) with the use of Eq (27) which becomes

$$\psi_E = \sqrt{2\rho\mu u_e x} \int_0^{\eta_e} F d\eta \quad .$$

The parameter α has the value

$$\alpha = \sqrt{2\rho\mu u_e} \int_0^{\eta_e} F d\eta$$

and the constant K becomes

$$K = \left(\int_0^{\eta_e} F \, d\eta \right)^{-2} . \quad (69)$$

For large value of η , the integral has the value $\eta - 1.21678$ which gives $K = 0.0437078$ when $\eta_e = 6$.

Equation (66) is written in finite-difference form with central differences (not the same approach as Spalding)¹², which gives

$$\omega \frac{\partial F}{\partial \omega} = \omega_j (F_{j+1} - F_{j-1}) / (2\Delta\omega) \quad (70a)$$

$$\frac{\partial}{\partial \omega} \left(K F \frac{\partial F}{\partial \omega} \right) = \frac{K}{\Delta\omega^2} \left[F_{j+1/2} (F_{j+1} - F_j) - F_{j-1/2} (F_j - F_{j-1}) \right]$$

where

$$F_{j+1/2} = \frac{1}{2} (F_{j+1} + F_j) \quad (70b)$$

The resulting difference equation is locally linearized with Eq (55) and becomes

$$-A_j F_{j-1} + B_j F_j - C_j F_{j+1} = D_j \quad (70c)$$

where

$$A_j = 2\overline{F}_{j-1} - \Delta\omega \omega_j / K$$

$$B_j = 4\overline{F}_j \quad (70d)$$

$$C_j = 2\overline{F}_{j+1} + \Delta\omega \omega_j / K$$

$$D_j = 2\overline{F}_j^2 - \overline{F}_{j+1}^2 - \overline{F}_{j-1}^2 .$$

It should be emphasized that in the Spalding work, a control volume approach is utilized to obtain the finite-difference form of the governing equations in the case of nonsimilar flows.

As in previous work calculating the skin friction is one of the results desired in the numerical solution. The skin friction becomes

$$c_f \sqrt{Re_x} = 2 \sqrt{K/2} \left(F \frac{\partial F}{\partial \omega} \right)_w \quad (71)$$

$$= \sqrt{K/2} \left(\frac{\partial F^2}{\partial \omega} \right)_w .$$

The second form of this equation shows that near the wall

$$F = F_0 \sqrt{\omega} + \dots \quad (72)$$

The first form of Eq (71) is indeterminant because $\partial F / \partial \omega$ is infinite and $F = 0$. This singularity results with the stream function variables. The similar coordinate requires special attention at the wall because the singularity appears in Eq (68) also. With Eq (72) the relation (68) near the wall is transformed into:

$$\eta = 2\omega / (\sqrt{KF}) \quad (73)$$

which permits η to be evaluated at the first grid point away from the wall. The remaining values of η can be obtained from Eq (68) with the trapezoidal integration formula. Below is a worked problem with stream function variables.

Worked Problem: Stream Function Variables

The program below solves scalar tridiagonal Eqn (70) with subroutine TRIDAG. The skin friction is determined from Eq (71) and the similarity coordinate η is calculated with Eqs (73) and (68). The notation used is

<u>FORTRAN Name</u>	<u>Symbol</u>
FK	K
Y(J)	ω_j
DY	$\Delta\eta$

Solutions with various JMAX are

JMAX	$c_f \sqrt{Re_x}$	% Error	η_e
21	0.735378	10.7	5.96837
41	0.691324	4.10	5.98933
81	0.674223	1.52	5.99663
161	0.667810	5.56×10^{-1}	5.99898
∞	0.66411538		

The error for the skin friction is given above and shown in Figure 3. Accuracy with the stream function variables is significantly less than with the similarity variables. The velocity profile, in terms of stream function variable, is given in Figure 6 and, in terms of the similarity variable in Figure 7. The symbols in these figures are the grid points when JMAX = 21. These figures show that there are few grid points near the wall and this results in the inaccuracy of this method. A variable grid system would alleviate this problem; a new coordinate $\sqrt{\omega}$ may be desirable. In the table above, the value of η at the boundary is also given. The value of K was chosen to have $\eta_e = 6$; as the step-size $\Delta\omega$ is decreased, the numerical result appears to be approaching this value.

```

PROGRAM BLS (INPUT,OUTPUT)
C
C SIMILAR SOLUTION   STREAM FUNCTION VARIABLE
C
  DIMENSION A(161), B(161), C(161), D(161), E(161), FL(161)
  DIMENSION W(2,161), ETA(161), Y(161)
  FK = 0.0437078
  JMAX = 41
  YE = 1.0
  JMI = JMAX - 1
  DY = YE / ( JMAX - 1.0 )
  Y(1) = 0.0
  ETA(1) = 0.0
  W(1,1) = 0.0
  DO 10 J = 2, JMAX
    Y(J) = Y(J-1) + DY
    ETA(J) = ETA(J-1)
  10 W(1,J) = 1.0
  PRINT 9500, (J,Y(J),W(1,J),J=1,JMAX)
9500 FORMAT(10Y,13,2(2X,E12.5))
  DO 40 ITER = 1,20
    DO 20 J = 2, JMI
      A(J) = 2.0 * W(1,J-1) - DY * Y(J) / FK
      B(J) = 4.0 * W(1,J)
      C(J) = 2.0 * W(1,J+1) + DY * Y(J) / FK
      D(J) = 2.0 * W(1,J)**2 - W(1,J+1)**2 - W(1,J-1)**2
    20 CONTINUE
    CALL TRIDAG (A,B,C,D,JMAX,W,1,E,EL,2)
    CF = SQRT(FK/2.0) * ( 2.0*W(1,2)**2 - 0.5*W(1,3)**2 ) / DY
    PRINT 9600, ITER, CF
9600 FORMAT( 20X,13,5X,E16.0)
  40 CONTINUE
  ETA(2) = 2.0 * Y(2) / ( SQRT(FK) * W(1,2) )
  DO 60 J = 2, JMI
    ETA(J+1) = ETA(J) + 2.0*DY/(SQRT(FK)*(W(1,J+1)+W(1,J)))
  60 PRINT 9700, (J,Y(J),W(1,J),ETA(J),J = 1,JMAX)
9700 FORMAT(10Y,13,3(2X,E12.5))
  RETURN
  END

```

Figure 6. Velocity Across Boundary Layer for Stream Function Variable

Figure 7. Velocity Across Boundary Layer for Similarity Variable

Midpoint Scheme -- For the nonsimilar boundary layer equations, this method becomes the box scheme introduced by Keller.¹³ For the similarity form of boundary layer Eq (32), the following difference equations are obtained at $j = 2, 3, 4, \dots, J$:

$$(f_j - f_{j-1})/h_j = F_{j-1/2} = \frac{1}{2} (F_j + F_{j-1}) \quad (74a)$$

$$\left[(\ell S)_j - (\ell S)_{j-1} \right] / h_j + P \left(1 - F_{j-1/2}^2 \right) + \frac{1}{2} (1 + P) (fS)_{j-1/2} = 0 \quad (74b)$$

$$(F_j - F_{j-1})/h_j = S_{j-1/2} = \frac{1}{2} (S_j + S_{j-1}) \quad (74c)$$

where $h_j = (\eta_j - \eta_{j-1})$. Two terms in the second statement above must be locally linearized. The method utilized by Keller¹³ uses the following quantities:

$$\begin{aligned} f_j^{(i+1)} &= f_j^{(i)} + \delta f_j^{(i)} \\ F_j^{(i+1)} &= F_j^{(i)} + \delta F_j^{(i)} \quad i = 0, 1, 2, \dots, \\ S_j^{(i+1)} &= S_j^{(i)} + \delta S_j^{(i)} \end{aligned} \quad (75)$$

Linearization is accomplished by dropping the quadratic and higher-order terms. However, there is no unique method of evaluating the nonlinear terms in Eq (74b) and the following form is chosen for these terms (notation is simplified by using $\delta F_j \equiv \delta F_j^{(i)}$, etc.

$$\begin{aligned} (F_{j-1/2}^2)^{(i+1)} &= \frac{1}{2} (F_j^2 + F_{j-1}^2)^{(i+1)} \\ &= (F_{j-1/2}^2)^{(i)} + F_j^{(i)} \delta F_j + F_{j-1}^{(i)} \delta F_{j-1} + \dots \end{aligned} \quad (76)$$

and

$$\begin{aligned} (fS)_{j-1/2}^{(i+1)} &= \frac{1}{2} [(fS)_j + (fS)_{j-1}]^{(i+1)} \\ &= (fS)_{j-1/2}^{(i)} + \frac{1}{2} (S \delta f + f \delta S)_j^{(i)} + \frac{1}{2} (S \delta f + f \delta S)_{j-1}^{(i)} + \dots \end{aligned}$$

When these linearized terms are used in conjunction with Eqs (75) in the difference relations of Eq (74), the following correction equations are obtained for $j = 2, 3, \dots, J$:

$$\delta f_j - \delta f_{j-1} - \frac{1}{2} h_j (\delta F_j + \delta F_{j-1}) = (r_1)_j$$

$$(s_1)_j \delta S_j + (s_2)_j \delta S_{j-1} + (s_3)_j \delta f_j + (s_4)_j \delta f_{j-1} + (s_5)_j \delta F_j$$

$$+ (s_6)_j \delta F_{j-1} = (r_2)_j$$

$$\delta F_j - \delta F_{j-1} - \frac{1}{2} h_j (\delta S_j + \delta S_{j-1}) = (r_3)_{j-1}$$

where

(77)

$$(r_1)_j = -(f_j - f_{j-1})^{(i)} + \frac{1}{2} h_j (F_j + F_{j-1})^{(i)}$$

$$(r_2)_j = -(\ell S)_j / h_j + (\ell S)_{j-1} / h_j - P + P(F_{j-1/2}^2)^{(i)} - \frac{1}{2} (1 + P)(fS)_{j-1/2}^{(i)}$$

$$(r_3)_{j-1} = -(F_j - F_{j-1})^{(i)} + \frac{1}{2} h_j (S_j + S_{j-1})^{(i)}$$

$$(s_1)_j = \ell_j^{(i)} / h_j + \frac{1}{4} (1 + P) f_j^{(i)}$$

$$(s_2)_j = -\ell_{j-1}^{(i)} h_j + \frac{1}{4} (1 + P) f_{j-1}^{(i)}$$

$$(s_3)_j = \frac{1}{4} (1 + P) S_j^{(i)}$$

$$(s_4)_j = \frac{1}{4} (1 + P) S_{j-1}^{(i)}$$

$$(s_5)_j = -PF_j^{(i)}$$

$$(s_6)_j = -PF_{j-1}^{(i)} .$$

In the foregoing, the term ℓ has not been linearized with the Newton-Raphson method. If the correct linearization is applied to this term for a turbulent boundary layer with the usual eddy viscosity formulation, the resulting difference equations will no longer be block-tridiagonal. Partial linearization has been used by Carter.²⁰

Difference Eqs (77) and the boundary conditions are now written in the following form where L_{1j} , L_{2j} and L_{3j+1} represent the left side of Eqs (77):

$$\delta f_1 = 0$$

$$\delta F_1 = 0$$

$$(L_3)_1 = (r_3)_1$$

$$\left. \begin{aligned} L_{1j} &= (r_1)_j \\ L_{2j} &= (r_2)_j \\ L_{3j} &= (r_3)_j \end{aligned} \right\} \quad j = 2, 3, \dots, J-1 \quad (78)$$

$$L_{1J} = (r_1)_J$$

$$L_{2J} = (r_2)_J$$

$$\delta F_J = 0$$

These equations can now be readily written in block-tridiagonal form as follows:

$$B_1 W_1 - C_1 W_2 = D_1$$

$$-A_j W_{j-1} + B_j W_j - C_j W_{j+1} = D_j \quad j = 2, 3, \dots, J-1 \quad (79)$$

$$-A_J W_{J-1} + B_J W_J = D_J$$

where $W_j = \begin{bmatrix} \delta f_j \\ \delta F_j \\ \delta S_j \end{bmatrix}^T$. The coefficients in these equations are the following:

$$D_1 = \begin{bmatrix} 0 \\ 0 \\ (r_3)_1 \end{bmatrix} \quad D_j = \begin{bmatrix} (r_1)_j \\ (r_2)_j \\ (r_3)_j \end{bmatrix} \quad j = 2, 3, \dots, J-1 \quad D_J = \begin{bmatrix} (r_1)_J \\ (r_2)_J \\ 0 \end{bmatrix} \quad (80)$$

$$B_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & -\frac{1}{2}h_2 \end{bmatrix} \quad C_j = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & -\frac{1}{2}h_{j+1} \end{bmatrix} \quad j = 1, 2, \dots, J-1$$

$$B_j = \begin{bmatrix} 1 & -\frac{1}{2}h_j & 0 \\ (s_3)_j & (s_5)_j & (s_1)_j \\ 0 & -1 & -\frac{1}{2}h_{j+1} \end{bmatrix} \quad j = 2, 3, \dots, J-1$$

$$A_j = \begin{bmatrix} 1 & -\frac{1}{2}h_j & 0 \\ -(s_4)_j & -(s_6)_j & -(s_2)_j \\ 0 & 0 & 0 \end{bmatrix} \quad j = 2, 3, \dots, J$$

$$B_J = \begin{bmatrix} 1 & -\frac{1}{2}h_J & 0 \\ (s_3)_J & (s_5)_J & (s_1)_J \\ 0 & 1 & 0 \end{bmatrix} .$$

Block-tridiagonal Eqs (79) with the midpoint scheme can be readily solved with the general procedure described previously. An even more efficient procedure has been developed by Keller for this system by introducing a special zero pattern in the C_j and A_j matrices. The details of this method and a listing of the required subroutine have been presented elsewhere.¹¹

Below a worked problem of midpoint difference scheme is presented.

Worked Problem: Midpoint Difference Scheme

The flat plate similar problem ($P = 0$ and $r_b^j = 1.0$) has been solved with the code below where the coefficients given by Eq (80) are evaluated in COEF. For this problem the η is defined by Eq (31) and is different than that given by Eq (27). Therefore, for this coordinate system, the computation is performed to $\eta_e = 9$ and the skin friction parameter is

$$c_f \sqrt{Re_x} = 2 \sqrt{\ell_w / \ell_e} \frac{\partial F}{\partial \eta} \quad (81)$$

where the subscripts indicate that ℓ is evaluated at the wall (w), and outer edge, (e), of the boundary layer. The new notation introduced is the following

FORTRAN Name	Symbol
P	P
DETAC	Constant $\Delta\eta$
SL(J)	ℓ_j
DW(1, J), DW(2, J), DW(3, J)	$\delta f_j, \delta F_j, \delta S_j$
DETA(J)	h_j

Subroutine COEF has been written to include a variable value of ℓ_j and a variable grid, h_j . The solutions have been obtained with $\ell_j = 1.0$ and $h_j = \Delta\eta = \text{constant}$. The accuracy of the midpoint scheme has been evaluated and is

```

C      SIMILAR SOLUTION% BOX SCHEME
C      ORDER OF VARIABLES FL,F,S
COMMON /A1/ DETA(161), SL(161), P
DIMENSION E(3,3,161), EL(3,161)
DIMENSION AL(3,3), BE(3)
DIMENSION AM(3,3), BM(3,3), CM(3,3), DM(3)
DIMENSION W(3,161), DW(3,161), ETA(161), IN(3)
JMAX = 41
JMAX = 81
JMAX = 161
JMAX = 21
P = 0.0
ETA1 = 9.0
MM = 3
DETAC = ETA1 / ( JMAX - 1.0 )
SL(1) = 1.0
ETA(1) = 0.0
W(1,1) = 0.0
W(2,1) = 0.0
DW(1,1) = 0.0
DW(2,1) = 0.0
DW(3,1) = 0.0
DO 10 J = 2, JMAX
ETA(J) = ETA(J-1) * DETAC
DETA(J) = DETAC
SL(J) = 1.0
DW(1,J) = 0.0
DW(2,J) = 0.0
DW(3,J) = 0.0
W(2,J) = 1.0
W(1,J) = W(1,J-1) + 0.5*( W(2,J) + W(2,J-1) ) * DETA(J)
W(3,J) = 0.0
10 CONTINUE
W(3,1) = 1.0 / DETA(2)
PRINT 9500, (J,ETA(J),W(1,J),W(2,J),W(3,J),J=1,JMAX)

```



```

9500 FORMAT(10X,I3,2X,E12.5,2X,E12.5,2X,E12.5,2X,E12.5)
DO 40 ITER = 1,20
CALL COEF(MM,JMAX,E,EL,AM,BM,CM,DM,IN,W,AL,BE)
CALL SOLU(DW,JMAX,MM,E,EL)
DO 30 J = 1, JMAX
DO 20 M = 1, MM
20 W(M,J) = W(M,J) + DW(M,J)
30 CONTINUE
CF = 2.0 * SQRT( SL(1)**2/SL(JMAX) ) * W(3,1)
ERROR = ( CF/0.6641154 - 1.0 ) * 100.0
PRINT 9600, ITER, CF,ERROR
9600 FORMAT( 20X,I3,5X,E16.9,5X,E16.9)
40 CONTINUE
PRINT 9500,(J,ETA(J),W(1,J),W(2,J),W(3,J),J=1,JMAX)
RETURN
END

```

```

SUBROUTINE COEF(MM,JMAX,E,EL,AM,BM,CM,DM,IN,W,AL,BE)
COMMON /A1/ DETA(161), SL(161), P
DIMENSION IN(MM),E(MM,MM,JMAX),EL(MM,JMAX),W(MM,JMAX)
DIMENSION AM(MM,MM),BM(MM,MM),CM(MM,MM),DM(MM)
DIMENSION AL(MM,MM),BE(MM)
DO 15 M = 1, MM
DO 14 N = 1, MM
14 AM(M,N)=0.0
BM(M,N)=0.0
CM(M,N)=0.0
15 DM(M) = 0.0
BM(1,1) = 1.0
BM(2,2) = 1.0
BM(3,2) = - 1.0
BM(3,3) = - 0.5 * DETA(2)
CM(3,2) = - 1.0
CM(3,3) = 0.5 * DETA(2)
DM(3) = - W(2,2) + 0.5*DETA(2)*(W(3,2) + W(3,1) )
CALL EEL(1,MM,JMAX,E,EL,AM,BM,CM,DM,IN,AL,BE)
DO 20 J = 2, JMAX
F2 = 0.5 * ( W(2,J)**2 + W(2,J-1)**2 )
FS = 0.5 * ( W(1,J)*W(3,J) + W(1,J-1)*W(3,J-1) )
AM(1,1) = 1.0
AM(1,2) = 0.5 * DETA(J)
BM(1,2) = - 0.5 * DETA(J)
DM(1) = -W(1,J)+W(1,J-1)+0.5*DETA(J)*(W(2,J)+W(2,J-1))
AM(2,1) = - 0.25*(1.0+P)*W(3,J-1)
AM(2,2) = P*W(2,J-1)
AM(2,3) = SL(J-1)/DETA(J) - 0.25*(1.0+P)*W(1,J-1)
BM(2,1) = 0.25*(1.0+P)*W(3,J)
BM(2,2) = - P*W(2,J)
BM(2,3) = SL(J)/DETA(J) + 0.25*(1.0+P)*W(1,J)
DM(2) = - (SL(J)*W(3,J)-SL(J-1)*W(3,J-1))/DETA(J) - P + P*F2
1 - 0.5*(1.0+P)*FS
IF( J.EQ. JMAX ) GO TO 18
BM(3,3) = - 0.5 * DETA(J+1)
CM(3,3) = 0.5 * DETA(J+1)
DM(3) = - W(2,J+1) + W(2,J) + 0.5*DETA(J+1)*(W(3,J+1)+W(3,J))
GO TO 20
18 CONTINUE
BM(3,2) = 1.0
BM(3,3) = 0.0
DM(3) = 0.0
20 CALL EEL(J,MM,JMAX,E,EL,AM,BM,CM,DM,IN,AL,BE)
RETURN
END

```

JMAX	$c_f \sqrt{Re}_x$	% Error
21	0.663883	1.10×10^{-1}
41	0.663935	2.72×10^{-2}
81	0.664070	6.87×10^{-3}
161	0.664104	1.79×10^{-3}
	0.6641154	

These results are plotted in Figure 4 and show second-order behavior for all of the step-sizes employed. Also, it should be noted that the formulation of the midpoint scheme used here gives significantly more accurate results for this problem than another formulation.⁹

Evaluation of Skin Friction -- In studying the accuracy of boundary layer schemes, the wall skin friction has been used to evaluate the various methods. With the central difference scheme, the shear stress is not obtained directly as part of the solution. It is readily obtained between the grid points but requires a special relation at the wall. When the midpoint scheme is used, the wall shear stress is obtained as part of the solution. The results in Figure 4 show that the midpoint scheme is second-order accurate for all solutions, while the central difference scheme is approaching this behavior with a large number of intervals. The velocity F , which is always determined as part of the solution, is second-order accurate with both schemes. The velocity at $\eta = 0.6$ is given below for the central difference (F_{CD}) and midpoint (F_{MP}) schemes for the case illustrated in Figure 4.

JMAX	F_{CD}	% Error	F_{MP}	% Error
11	0.283328	9.81×10^{-1}	0.286103	1.97
21	0.281252	2.44×10^{-1}	0.281986	5.02×10^{-1}
41	0.280744	5.99×10^{-2}	0.280930	1.26×10^{-1}
81	0.280618	1.50×10^{-2}	0.280664	3.14×10^{-2}
161	0.280586	3.56×10^{-3}	0.280598	7.84×10^{-3}
Extrapolated	0.280576		0.280576	

A fourth-order method with JMAX = 161 was used to obtain the solution, and the "exact" value of F at $\eta = 0.6$ is 0.280576. The error in the velocity for the two schemes is given in Figure 8 and both techniques show second-order behavior. These results suggest that the use of a polynomial to approximate the velocity profile as used in Eq (59) may result in poor skin friction results.

Figure 8. Accuracy of Velocity

One approach that gives second-order accuracy for the skin friction for all intervals considered is obtained by first writing Eq (34) as

$$\frac{\partial \tau}{\partial \eta} = L = V \frac{\partial F}{\partial \eta} + \beta (F^2 - 1) \quad (82)$$

$$\frac{\partial F}{\partial \eta} = \tau \quad .$$

When these relations are written with midpoint differences at the first interval away from the wall and τ_2 is eliminated, the wall velocity gradient becomes

$$\left(\frac{\partial F}{\partial \eta} \right)_w = F_2 / \Delta \eta - \frac{1}{2} \Delta \eta L_{1.5} \quad (83)$$

The value of L at the midpoint is determined from governing Eq (82) which gives

$$L_{1.5} = \frac{1}{2} (V_1 + V_2) F_2 / \Delta \eta + \beta \left(\frac{1}{4} F_2^2 - 1 \right) \quad (84)$$

The skin friction for similar flows becomes, with the use of Eqs (59), (83), and (84)

$$C_f \sqrt{Re_x} = \varphi \left[F_2 / \Delta\eta - \frac{1}{4} (V_1 + V_2) F_2 - \frac{1}{2} \beta \Delta\eta \left(\frac{1}{4} F_2^2 - 1 \right) \right] \quad (85)$$

where φ is defined after Eq (59).

Two-dimensional, stagnation point flow ($\beta = 1$) is used to illustrate the accuracy of the skin friction relations Eqs (59) and (85). The "exact" value of the skin friction is 2.46517536 which was obtained with a fourth-order order method with JMAX = 161. The results with the central difference scheme are given in Figure 9. The skin friction evaluated with Eq (85) is second-order accurate with a slight deviation at the largest mesh size. Although Eq (59) is a second-order accurate expression, the skin friction error shows erratic behavior in Figure 9. For a small number of intervals, the calculated skin friction is smaller than the exact value, while for a large number of intervals the calculated skin friction becomes larger than the exact value. The truncation error of Eq (59) is of the form

$$\text{truncation error} = A \Delta\eta^2 - B \Delta\eta^3 + \dots \quad (86)$$

where for large $\Delta\eta$, the second-term dominates and the method appears to be third order. For very small $\Delta\eta$, the first term dominates and the method becomes second order as expected.

Figure 9. Importance of Method of Evaluating the Wall Velocity Gradient

Although the dependent variables which are solved for directly in the central difference scheme are second-order accurate, many skin friction relations can give erroneous predictions of the accuracy of the scheme. The foregoing discussion shows that arbitrary second-order relations are not adequate for evaluating the skin friction if a small number of mesh intervals are used.

Variable Grid Scheme -- Above, in the midpoint scheme, there was no restriction placed on the step-size variation across the boundary layer. The midpoint scheme has second-order accuracy with a variable grid; for example, if all the intervals are halved, the numerical error is reduced by a factor of four. This behavior allows Richardson extrapolation to be employed for more accurate results and provides a procedure to judge the accuracy of the solutions. When central differences are applied with a variable grid, a special procedure is required in order to retain second-order accuracy. The truncation error for the usual central difference approximation to the second derivative is $O(\Delta\eta)$. When a transformation is introduced to determine the grid spacing as the grid is refined, second-order behavior with central differences is obtained.

A new coordinate N is introduced where a uniform grid spacing ΔN is taken. This transformation is expressed as

$$\eta_j = \eta(N_j) \quad j = 1, 2, 3, \dots, J \quad (87)$$

and

$$\frac{d\eta}{dN} = \eta' \quad .$$

The first derivative in terms of the coordinate becomes

$$\begin{aligned} \left(\frac{\partial W}{\partial \eta} \right)_j &= \left(\frac{\partial W}{\partial N} / \eta' \right)_j \\ &= (W_{j+1} - W_{j-1}) / (2\Delta N \eta'_j) + O(\Delta N^2) \quad . \end{aligned} \quad (88)$$

The coordinate derivative is evaluated as

$$\eta'_j = (\eta_{j+1} - \eta_{j-1}) / (2\Delta N) + O(\Delta N^2) \quad . \quad (89)$$

The combination of the above gives the following for the first derivative:

$$\left(\frac{\partial W}{\partial \eta} \right)_j = (W_{j+1} - W_{j-1}) / (\eta_{j+1} - \eta_{j-1}) + O(\Delta N^2) \quad (90)$$

where the expression is second-order accurate in terms of ΔN . The second derivative is developed in a similar manner:

$$\begin{aligned} \frac{\partial}{\partial \eta} \left(\ell \frac{\partial W}{\partial N} \right) &= \frac{\partial W}{\partial N} \left(\ell \frac{\partial}{\partial \eta} / \eta' \right)_j / \eta'_j \\ &= \frac{1}{\eta'_j} \left[(\ell / \eta')_{j+1/2} (W_{j+1} - W_j) / \Delta N - (\ell / \eta')_{j-1/2} (W_j - W_{j-1}) / \Delta N \right] \\ &\quad + O(\Delta N^2) . \end{aligned} \tag{91}$$

The coordinate derivatives are introduced with the final result becoming

$$\frac{\partial}{\partial \eta} \left(\ell \frac{\partial W}{\partial \eta} \right)_j = \frac{2}{(\eta_{j+1} - \eta_{j-1})} \left[\ell_{j+1/2} \left(\frac{W_{j+1} - W_j}{\eta_{j+1} - \eta_j} \right) - \ell_{j-1/2} \left(\frac{W_j - W_{j-1}}{\eta_j - \eta_{j-1}} \right) \right] + O(\Delta N^2) \tag{92}$$

The choice of grid spacing depends upon the problem being solved and must be specified. At present, self-adaptive grid techniques are just beginning to be developed for finite-difference methods. For turbulent boundary layers, where a variable grid is necessary, the following grid spacing is often employed:

$$\Delta \eta_{j+1/2} = K \Delta \eta_{j-1/2} ;$$

the grid spacing becomes

$$\eta_j = \eta_J \left(K^{N_j / \Delta N_0} - 1 \right) / \left(K^{1 / \Delta N_0} - 1 \right) \tag{93}$$

where

$$N_j = (j - 1) \Delta N \quad J = 1, 2, \dots, J$$

$$N_J = 1 .$$

The above grid spacing relation involves two parameters, ΔN_0 and K , which determine how the grid varies across the boundary layer. Typical values of these parameters are $K = 1.5$ and $\Delta N_0 = 0.1$. Reference 9 presents more details of this method and the application to the nonsimilar boundary layer equations.

The importance of using a variable grid to obtain accurate results without an excessive number of grid points is illustrated in Figure 10. The turbulent flow along a flat plate is solved with a uniform grid and a variable grid determined by Eq (93) where $K = 1.82$ and $\Delta N_0 = 0.1$. The skin friction error at $Re_x = 1.88 \times 10^6$ is given for both grid spacings in Figure 10. For the same accuracy in this case, the variable grid scheme reduces the number of intervals by a factor of approximately 30 and gives a corresponding reduction in computer time.

Higher Order Schemes -- Although various higher-order techniques²¹⁻²⁸ have been developed for solving the boundary layer equations, considerably more work is required to establish conclusively the superiority of one approach over the others. Higher-order accuracy can be obtained by using difference relations with higher-order truncation errors and by using Richardson extrapolation or deferred corrections. Some of the properties of the higher-order schemes are illustrated with the technique developed by Wornom²⁵ and this method is a midpoint scheme with fourth-order truncation errors.

Figure 10. Influence of Grid Spacing on Accuracy of Results for Flat Plate Turbulent Boundary Layer

Equation (30a) for similar flow and $\ell = \theta = 1$ is written as a system of first-order equations

$$\frac{\partial f}{\partial \eta} = u \quad (94a)$$

$$\frac{\partial}{\partial \eta} (v + uf) = u^2(1 + \beta) - \beta \quad (94b)$$

$$\frac{\partial u}{\partial \eta} = v \quad (94c)$$

These equations are completed with the boundary conditions

$$\begin{aligned} u(0) &= 0 \\ f(0) &= f_w \\ u(\infty) &= 1.0 \end{aligned} \quad (95)$$

The governing Eqs (94) are expressed as

$$\frac{dg}{d\eta} = \varphi(\eta) \quad (96)$$

Euler-Maclurin sum formula gives

$$g_j - g_{j-1} - \frac{1}{2} h_j (g'_j + g'_{j-1}) + \frac{1}{12} h_j^2 (g''_j - g''_{j-1}) + O(h_j^4) = 0 \quad (97)$$

where fourth-order terms have been neglected. The variable g has the following values for the governing Eqs (94):

$$g = \begin{bmatrix} f \\ v + uf \\ u \end{bmatrix} \quad g' = \begin{bmatrix} u \\ (1 + \beta)u^2 - \beta \\ v \end{bmatrix} \quad g'' = \begin{bmatrix} v \\ 2(1 + \beta)uv \\ \beta(u^2 - 1) - fv \end{bmatrix} \quad (98)$$

The variable, g , and its derivatives are substituted into Eq (97) which results in a system of nonlinear equations in terms of the dependent variable $W_j = [f_j, u_j, v_j]^T$. The nonlinear terms are linearized with Eq (54) which gives, for example, the relation

$$uf = -u\bar{f} + \bar{u}f + u\bar{f} \quad (99)$$

When the linearization is completed, the resulting difference equations are of the form given by Eqs (79) where Eq (94c) with $j = 1$ is used along with wall boundary conditions while Eqs (94a, b) with $j = J$ are used with the outer edge boundary condition. The coefficients in these equations are the same as those given by Eqs (80) except for the following:

Coefficients for D_1

$$(r_1)_1 = f_w$$

$$(r_2)_1 = 0 \quad (100)$$

$$(r_3)_1 = \text{defined below with } j = 1 \quad .$$

Coefficients for D_j

$$(r_1)_j = 0$$

$$(r_2)_j = -h_j \beta + (\bar{u} \bar{f})_j - (\bar{u} \bar{f})_{j-1} - \frac{1}{2} h_j (1 + \beta) (\bar{u}_j^2 + \bar{u}_{j-1}^2) + T_j [(\bar{u} \bar{v})_j - (\bar{u} \bar{v})_{j-1}]$$

$$(r_3)_j = S_{j+1} \left[\beta (\bar{u}_{j+1}^2 - \bar{u}_j^2) + (\bar{f} \bar{v})_j - (\bar{f} \bar{v})_{j+1} \right] \quad .$$

Coefficients for D_J

$$(r_1)_J = 0$$

$$(r_2)_J = \text{defined above with } j = J$$

$$(r_3)_J = 1 \quad .$$

Coefficients for B_1

$$B_{31} = S_2 \bar{v}_1$$

$$B_{32} = -1 - R_2 \bar{u}_1$$

$$B_{33} = -\frac{1}{2} h_2 + S_2 \bar{f}_1 \quad .$$

Coefficients for C_j

$$C_{31} = (S\bar{v})_{j+1}$$

$$C_{32} = -1 - (R\bar{u})_{j+1}$$

$$C_{33} = \frac{1}{2} h_{j+1} + (S\bar{f})_{j+1} \quad .$$

Coefficients for B_j

$$B_{13} = S_j$$

$$B_{21} = \bar{u}_j$$

$$B_{22} = \bar{f}_j - h_j(1 + \beta)\bar{u}_j + (T\bar{v})_j$$

$$B_{23} = 1 + (T\bar{u})_j$$

$$B_{31} = S_{j+1} \bar{v}_j$$

$$B_{32} = -1 - R_{j+1} \bar{u}_j$$

$$B_{33} = -\frac{1}{2} h_{j+1} + S_{j+1} \bar{f}_j \quad .$$

Coefficients for A_j

$$A_{13} = S_j$$

$$A_{21} = \bar{u}_{j-1}$$

$$A_{22} = \bar{f}_{j-1} + h_j(1 + \beta)\bar{u}_{j-1} + T_j \bar{v}_{j-1}$$

$$A_{23} = 1 + T_j \bar{u}_{j-1} \quad .$$

Coefficients for B_j

$$B_{13} = S_j$$

$$\left. \begin{array}{l} B_{21} = \\ B_{22} = \\ B_{23} = \end{array} \right\} \quad \text{Defined above with } j = J \quad .$$

The following notation is used in the above coefficients:

$$h_j = \eta_j - \eta_{j-1}$$

$$R_j = \frac{1}{6} h_j^2 \beta$$

$$S_j = \frac{1}{12} h_j^2$$

$$T_j = \frac{1}{6} h_j^2 (1 + \beta) \quad .$$

(101)

Terms involving h^2 in the above coefficients occur when the midpoint scheme is made fourth-order accurate. Therefore, if these terms are set to zero ($R_j = S_j = T_j = 0$), the resulting method will be equivalent to the previously described midpoint scheme with second-order accuracy.

The main program and the subroutine COEF for calculating the matrix coefficients are given on the next pages. A uniform grid is used in the program and results for a flat plate flow are given in Figure 11. The error in the wall shear stress is given for the foregoing scheme with the fourth-order terms included and neglected. These results show that a significant reduction in the number of intervals across the boundary layer can be obtained with the fourth-order scheme. The reduction becomes even more important as the desired accuracy is increased. The total saving in computer time depends on the decrease in number of intervals and the increase in solution time with the fourth-order scheme. If highly accurate solutions are desired, the fourth-order method is appropriate. For crude results, both methods are competitive.

```

      PROGRAM BIC (INPUT,OUTPUT)
C
C  SIMILAR SOLUTION - WORNOM 4TH ORDER SCHEME
C  ORDER OF VARIABLES F,U,V
C
      COMMON /1/ DELTA, BETA
      DIMENSION E(3,3,161), EL(3,161)
      DIMENSION AL(3,3), BE(3)
      DIMENSION AM(3,3), BM(3,3), CM(3,3), DM(3)
      DIMENSION W(3,161), ETA(161), IN(3)
      JMAX = 81
      JMAX = 161
      BETA = 1.0
      ETAE = 4.0
      ETAE = 7.0
      ETAE = 6.0
      MM = 3
      DELTA = ETAE / ( JMAX - 1.0 )
      ETA(1) = 0.0
      W(1,1) = - 0.5
      W(2,1) = 0.0
      W(3,1) = 1.0/DELTA
      DO 10 J = 2, JMAX
      ETA(J) = ETA(J-1) + DELTA
      W(2,J) = 1.0
      W(1,J) = W(1,J-1) + 0.5*DELTA*( W(2,J) + W(2,J-1) )
      W(3,J) = 0.0
10  CONTINUE
      PRINT 9500, (J,ETA(J),W(1,J),W(2,J),W(3,J),J=1,JMAX)
9500 FORMAT(10X,I3,4(2X,E12.5))
      DO 40 ITER = 1,20
      CALL COEF(MM,JMAX,E,EL,AM,BM,CM,DM,IN,W,AL,BE)
      CALL SOLU(W,JMAX,MM,E,EL)
30  CONTINUE
      CF = SQRT( 2.0 ) * W(3,1)
      PRINT 9600, ITER, CF
9600 FORMAT( 20X,I3,5X,E16.4)
40  CONTINUE
      PRINT 9500, (J,ETA(J),W(1,J),W(2,J),W(3,J),J=1,JMAX)
      RETURN
      END

```

```

SUBROUTINE COEF(MM,JMAX,E,EL,AM,BM,CM,DM,IN,W,AL,BE)
COMMON /1/ DELTA, BETA
DIMENSION IN(MM),E(MM,MM,JMAX),EL(MM,JMAX),W(MM,JMAX)
DIMENSION AM(MM,MM),BM(MM,MM),CM(MM,MM),DM(MM)
DIMENSION AL(MM,MM),BE(MM)
DO 15 M = 1, MM
DO 14 N = 1, MM
14 AM(M,N) = BM(M,N) = CM(M,N) = 0.0
15 DM(M) = 0.0
DH = 0.5 * DELTA
C IF TS IS ZERO , THEN SECOND ORDER SCHEME
TS = ( DELTA**2 ) / 12.0
T1 = 2.0 * TS * ( 1.0 + BETA )
DO 20 J = 1, JMAX
IF( J.EQ. JMAX ) GO TO 16
BM(3,1) = TS * W(3,J)
BM(3,2) = - 1.0 - 2.0 * TS * BETA * W(2,J)
BM(3,3) = - DH + TS * W(1,J)
CM(3,1) = TS * W(3,J+1)
CM(3,2) = - 1.0 - 2.0 * TS * BETA * W(2,J+1)
CM(3,3) = DH + TS * W(1,J+1)
DM(3) = TS*(BETA*(W(2,J+1)**2-W(2,J)**2)+W(1,J)*W(3,J)
1 -W(1,J+1)*W(3,J+1))
GO TO 17
16 CONTINUE
BM(3,1) = 0.0
BM(3,2) = 1.0
BM(3,3) = 0.0
DM(3) = 1.0
17 IF( J.GT. 1 ) GO TO 18
BM(1,1) = 1.0
BM(2,2) = 1.0
DM(1) = W(1,1)
GO TO 20
18 CONTINUE
AM(1,1) = 1.0
AM(1,2) = DH
BM(1,2) = - DH
DM(1) = 0.0
AM(2,1) = W(2,J-1)
AM(2,2) = W(1,J-1) + DELTA*( 1.0 + BETA ) * W(2,J-1)
AM(2,3) = 1.0
BM(2,1) = W(2,J)
BM(2,2) = W(1,J) - DELTA * ( 1.0 + BETA ) * W(2,J)
BM(2,3) = 1.0
DM(2) = - DELTA*BETA + W(2,J)*W(1,J) - W(2,J-1)*W(1,J-1)
1 - 0.5*DELTA*( 1.0 + BETA )*( W(2,J)**2 + W(2,J-1)**2 )
AM(1,3) = TS
BM(1,3) = TS
AM(2,2) = AM(2,2) + T1 * W(3,J-1)
AM(2,3) = AM(2,3) + T1 * W(2,J-1)
BM(2,2) = BM(2,2) + T1 * W(3,J)
BM(2,3) = BM(2,3) + T1 * W(2,J)
DM(2) = DM(2) + T1*(W(2,J)*W(3,J) - W(2,J-1)*W(3,J-1))
20 CALL EEL(J,MM,JMAX,E,EL,AM,BM,CM,DM,IN,AL,BE)
RETURN
END

```

Figure 11. Higher-Order Scheme for Flat Plate Laminar Boundary Layer

Influence of Edge Location -- In the previous solutions, the boundary conditions at the outer edge have been applied at $\eta_e = 6$ rather than infinity and the effect on accuracy of the results has been neglected. As more accuracy is desired in the numerical solutions, the value of η_e must be increased to avoid the boundary conditions influencing the results. Solutions have been obtained for the similar boundary layer equations with $\beta = 0$ and with the fourth-order finite-difference scheme. The results for the skin friction parameter are given below where various values of η_e and JMAX are used

JMAX	$C_f \sqrt{Re_x}$				
	$\eta_e = 4$	$\eta_e = 5$	$\eta_e = 6$	$\eta_e = 7$	$\eta_e = 8$
21			0.6641141539		
41			0.6641153079		
81	0.6663120857	0.6641785235	0.6641153796	0.6641146665	0.6641146572
161	0.6663120865	0.6641785254	0.6641153840	0.6641146749	0.6641146715
Extrapolated	0.6663120866	0.6641785255	0.6641153843	0.6641146755	0.6641146725
% Error	3.31×10^{-1}	9.61×10^{-3}	1.07×10^{-4}	4.5×10^{-7}	-

Also presented above is the value of the skin friction parameter when the grid size goes to zero. This result is obtained with Richardson extrapolation procedure for a fourth-order method which gives

$$W_{\Delta\eta \rightarrow 0} = W_c\left(\frac{1}{2}\Delta\eta\right) + \frac{1}{15} \left[W_c\left(\frac{1}{2}\Delta\eta\right) - W_c(\Delta\eta) \right] . \quad (102)$$

With the extrapolated value of the skin friction parameter for $\eta_e = 8$ taken as the exact value, the present error of the extrapolated values of the skin friction for the various η_e are also given in the table.

The influence of η_e on the solution with a second-order Crank-Nicolson method is now considered. The skin friction parameter and its error for various η_e are given below for the case of $\beta = 0$.

JMAX	$\eta_e = 4$		$\eta_e = 5$		$\eta_e = 6$	
	$C_f \sqrt{Re_x}$	% Error	$C_f \sqrt{Re_x}$	% Error	$C_f \sqrt{Re_x}$	% Error
11	0.6671407673	4.56×10^{-1}	0.6666643798	3.84×10^{-1}	0.6678111919	5.57×10^{-1}
21	0.6665088138	3.60×10^{-1}	0.6647844684	1.01×10^{-1}	0.6650215416	1.36×10^{-1}
41	0.6663606420	3.38×10^{-1}	0.6643290333	3.23×10^{-2}	0.6643407712	3.40×10^{-2}
81	0.6663241866	3.33×10^{-1}	0.6642160913	1.53×10^{-2}	0.6641716584	8.58×10^{-3}
161	0.6663151091	3.31×10^{-1}	0.6641879131	1.10×10^{-2}	0.6641294483	2.22×10^{-3}

In the determination of the percent error, the exact value of the skin friction parameter is taken as 0.6641146725 which is the extrapolated value obtained with the fourth-order method for $\eta_e = 8$. These results are also presented in Figure 12 where it can be seen that the solutions with $\eta_e = 6$ and $JMAX \leq 161$ are not influenced by the edge condition being applied at too small a value of η . If 10 or 20 grid points are being used, then an $\eta_e = 5$ is adequate. With $\eta_e = 4$, all of the results except $JMAX = 11$ are strongly influenced with the boundary condition being applied at too small a value of η .

For other values of β , the appropriate value of η_e will change. For accelerated flows ($\beta > 0$) the layer is thinner and smaller η_e would be allowed while for retarded flows ($\beta < 0$) larger values of η_e are required.

Figure 12. Influence of Edge Location

Nonsimilar Solutions

The various techniques used to solve the similar boundary layer equations can be applied to the complete boundary layer equations. The governing equations for incompressible flow in terms of the similarity variables were given in Eq (29) and are

$$2\xi \frac{\partial F}{\partial \xi} + \frac{\partial V}{\partial \eta} + F = 0 \quad (103)$$

$$2\xi F \frac{\partial F}{\partial \xi} + V \frac{\partial F}{\partial \eta} + \beta (F^2 - 1) - \frac{\partial^2 F}{\partial \eta^2} = 0 .$$

The nonsimilar solution procedure will be illustrated with the Crank-Nicholson solution scheme. The continuity equation is evaluated at the point $(i + 1/2, j - 1/2)$ with the following relations:

$$2\xi \frac{\partial F}{\partial \xi} = \xi_{i+1/2} \left[\left(F_{i+1} - F_i \right)_j + \left(F_{i+1} - F_i \right)_{j-1} \right] / \Delta \xi$$

$$\frac{\partial V}{\partial \eta} = \left(V_j - V_{j-1/2} \right)_{i+1/2} / \Delta \eta \quad (104)$$

$$F = \frac{1}{4} \left[\left(F_{i+1} + F_i \right)_j + \left(F_{i+1} + F_i \right)_{j-1} \right] .$$

The continuity equation becomes

$$-A_{21} F_{i+1, j-1} + B_{21} F_{i+1, j} - A_{22} V_{i+1/2, j-1} + B_{22} V_{i+1/2, j} = D_2 \quad (105)$$

where

$$A_{21} = -\Delta\eta \left(\frac{1}{4} + \xi_{i+1/2} / \Delta\xi \right)$$

$$A_{22} = 1$$

$$B_{21} = \Delta\eta \left(\frac{1}{4} + \xi_{i+1/2} / \Delta\xi \right)$$

$$B_{22} = 1$$

$$D_2 = \Delta\eta \left(-\frac{1}{4} + \xi_{i+1/2} / \Delta\xi \right) (F_j + F_{j-1})_i$$

The various terms in momentum Eq (103) are written in finite-difference form as given below.

This is evaluated at location $(i + 1/2, j)$ with second-order accurate difference relations.

$$2\xi F \frac{\partial F}{\partial \xi} = 2 \xi_{i+1/2} \left(\frac{F_{i+1} + F_i}{2} \right)_j \left(\frac{F_{i+1} - F_i}{\Delta\xi} \right)_j \quad (106a)$$

$$= \frac{\xi_{i+1/2}}{\Delta\xi} \left(2 \bar{F}_{i+1} F_{i+1} - F_{i+1}^2 - F_i^2 \right)_j$$

$$V \frac{\partial F}{\partial \eta} = \frac{1}{2} V_{i+1/2, j} \left[\left(\frac{\partial F}{\partial \eta} \right)_{i+1} + \left(\frac{\partial F}{\partial \eta} \right)_i \right]_j$$

$$= \frac{1}{2} V_{i+1/2, j} \left[\left(\frac{\partial F}{\partial \eta} \right)_i + \left(\frac{\partial \bar{F}}{\partial \eta} \right)_{i+1, j} \right] + \frac{1}{2} \bar{V}_{i+1/2, j} \left(\frac{F_{j+1} - F_{j-1}}{2\Delta\eta} \right)_{i+1} \quad (106b)$$

$$- \frac{1}{2} \bar{V}_{i+1/2, j} \left(\frac{\partial \bar{F}}{\partial \eta} \right)_{i+1, j}$$

$$\beta(F^2 - 1) = \beta_{i+1/2} \left[\frac{1}{2} (F_{i+1}^2 + F_i^2)_j - 1 \right] \quad (106c)$$

$$= \beta_{i+1/2} \left[\bar{F}_{i+1} F_{i+1} + \frac{1}{2} (F_i^2 - \bar{F}_{i+1}^2) - 1 \right]_j \quad (106c)$$

cont

$$-\frac{\partial^2 F}{\partial \eta^2} = -\frac{1}{2} (F_{j+1} - 2F_j + F_{j-1})_{i+1} / \Delta \eta^2 - \frac{1}{2} (F_{j+1} - 2F_j + F_{j-1})_i / \Delta \eta^2 \quad (106d)$$

When the relations in Eqs (106) are used in momentum Eq (103), the difference equation becomes

$$-A_{11} F_{i+1,j-1} + B_{11} F_{i+1,j} + B_{12} V_{i+1/2,j} - C_{11} F_{i+1,j+1} = D_1 \quad (107)$$

where

$$A_{11} = \frac{1}{2} \left(1 + \frac{1}{2} \Delta \eta \bar{V}_{i+1/2,j} \right)$$

$$B_{11} = 1 + \Delta \eta^2 \bar{F}_{i+1,j} \left(\beta_{i+1/2} + \underline{2\xi_{i+1/2}/\Delta \xi} \right)$$

$$B_{12} = \frac{1}{4} \Delta \eta \left[\underline{(F_{j+1} - F_{j-1})_i} + (\bar{F}_{j+1} - \bar{F}_{j-1})_{i+1} \right]$$

$$C_{11} = \frac{1}{2} \left(1 - \frac{1}{2} \Delta \eta \bar{V}_{i+1/2,j} \right)$$

$$D_1 = \frac{1}{2} \underline{(F_{j+1} - 2F_j + F_{j-1})_i} + \frac{1}{2} \Delta \eta^2 \beta_{i+1/2} \left[(1 + \bar{F}_{i+1}^2)_j + \underline{(1 - F_i^2)_j} \right] \\ + \frac{1}{2} \Delta \eta^2 \bar{V}_{i+1/2,j} \left(\frac{\bar{F}_{j+1} - \bar{F}_{j-1}}{2\Delta \eta} \right)_{i+1} + \underline{\Delta \eta^2 \xi_{i+1/2} (\bar{F}_{i+1}^2 + F_i^2)_j / \Delta \xi} \quad .$$

The quantities underlined result when the nonsimilar terms in the boundary layer equations are included.

The resulting difference Eqs (105) and (107), along with the boundary conditions, are block-tridiagonal:

$$B_1 W_1 - C_1 W_2 = 0$$

$$-A_j W_{j-1} + B_j W_j - C_j W_{j+1} = D_j \quad j = 2, 3, 4, \dots, J-1 \quad (108)$$

$$-A_J W_{J-1} + B_J W_J = D_J$$

where

$$\begin{aligned}
 W_j &= \begin{bmatrix} F_j \\ V_j \end{bmatrix} & B_1 &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & C_1 &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & D_1 &= \begin{bmatrix} 0 \\ V_w \end{bmatrix} \\
 A_j &= \begin{bmatrix} A_{11} & 0 \\ A_{21} & 1 \end{bmatrix}_j & B_j &= \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & 1 \end{bmatrix}_j & C_j &= \begin{bmatrix} C_{11} & 0 \\ 0 & 0 \end{bmatrix} & D_j &= \begin{bmatrix} D_1 \\ D_2 \end{bmatrix}_j \\
 A_J &= \begin{bmatrix} 0 & 0 \\ A_{21} & 1 \end{bmatrix} & B_J &= \begin{bmatrix} 1 & 0 \\ B_{21} & 1 \end{bmatrix}_J & D_J &= \begin{bmatrix} 1 \\ D_2 \end{bmatrix}_J
 \end{aligned}$$

The system of Eq (108) is readily solved with the block-tridiagonal procedure previously described. The coefficients in the matrices in Eq (108) are given with Eqs (105) and (107) and require iterative evaluation at each of the marching steps in the ξ - direction.

In order to begin solving the foregoing equations, initial conditions are required for dependent variables F and V across the boundary layer. At $\xi = 0$, governing Eq (103) reduce to the similarity form and are a function of η . These relations have been given in Eqs (62) and (57); with the present notation the coefficients in Eq (108) become

$$\begin{aligned}
 A_{11} &= \frac{1}{2} \left(1 + \frac{1}{2} \Delta\eta \bar{V}_j \right) \\
 A_{21} &= -\frac{1}{2} \Delta\eta \\
 B_{11} &= 1 + \Delta\eta^2 \beta \bar{F}_j \\
 B_{12} &= \frac{1}{4} \Delta\eta (\bar{F}_{j+1} - \bar{F}_{j-1}) \\
 B_{21} &= \frac{1}{2} \Delta\eta \\
 C_{11} &= \frac{1}{2} \left(1 - \frac{1}{2} \Delta\eta \bar{V}_j \right)
 \end{aligned} \tag{109}$$

$$D_1 = \frac{1}{2} \Delta \eta^2 \left[\beta \left(1 + \bar{F}_j^2 \right) + \bar{V}_j \left(\bar{F}_{j+1} - \bar{F}_{j-1} \right) / (2 \Delta \eta) \right]$$

$$D_2 = 0 \quad .$$

For the momentum equation the coefficients for the nonsimilar case are the same as the similar case but with additional terms added.

Worked Problem: Howarth Flow

In this case, there is incompressible flow along a flat plate where the edge velocity is decreasing linearly and is expressed as

$$u_e / U_\infty = (1 - x/L) \quad (110)$$

for the present problem, $L = 8$. The transformed ξ coordinate is obtained from Eq (27) with $j = 0$ and

$$(\rho\mu)_r = (\rho\mu)_\infty \quad (111)$$

$$\mathcal{K} = 1/(\rho\mu U)_\infty \quad .$$

The transformed coordinate becomes

$$\xi = x \left(1 - \frac{1}{2} x/L \right) \quad (112)$$

or

$$x/L = 1 - \sqrt{1 - 2\xi/L} \quad .$$

The pressure gradient parameter was defined with Eq (29) and for this case becomes

$$\beta = (2\xi/L) \left[(2\xi/L) - 1 \right] \quad . \quad (113)$$

The desired result is the skin friction along the flat plate and it is obtained from

$$\begin{aligned} c_f \sqrt{\text{Re}_x} &= 2(u_e / U_\infty)^2 \sqrt{x/2\xi} \frac{\partial F}{\partial \eta} \\ &= \left[\left(2 - \frac{1}{2} \xi \right) \sqrt{1 + \sqrt{1 - (1/4) \xi}} \right] \frac{\partial F}{\partial \eta} \quad . \end{aligned} \quad (114)$$

This problem has been solved with the programs given below. The notation is the following:

<u>FORTRAN Name</u>	<u>Symbol</u>
WL(1,J), WL(2,J)	$F_{i,j}, V_{i-1/2,j}$ (not used)
W(1,J), W(2,J)	$F_{i+1,j}, V_{i+1/2,j}$
BETA	β
I	$i = 1, 2, \dots, \text{IMAX}$
XI	$\xi_{i+1/2}$
DXI	$\Delta\xi$

The solutions were arrived at with JMAX = 31, one iteration for each marching step, and various IMAX were used to obtain the boundary layer from the leading edge ($\xi = 0$) to $\xi = 0.8$. The results for the skin friction parameter at $\xi = 0.8$ is given below:

<u>IMAX</u>	<u>$c_f \sqrt{\text{Re}_x}$</u>	<u>% Error</u>	<u>NT</u>	<u>Time (s)</u>
5	0.172931	4.04	28	2.6
9	0.168117	1.14	36	2.9
17	0.166708	2.97×10^{-1}	52	3.5
33	0.166337	7.38×10^{-2}	84	4.6
65	0.166245	1.85×10^{-2}	148	7.3
∞	0.166214			

```

C
C
C      BOUNDARY LAYER ALONG BODY      COUPLED EQUATIONS      F,V  VARIABLES
COMMON /A1/ DETA, BETA, I, XIDXI
DIMENSION E(2,2,161), EL(2,161)
DIMENSION AM(2,2), BM(2,2), CM(2,2), DM(2)
DIMENSION W(2,161), WL(2,161), ETA(161), IN(2)
DIMENSION AL(2,2), BE(2)
JMAX = 31
IMAX = 33
BETA = 0.0
ETA = 6.0
XI = 0.0
XIMAX = 0.8
I = 1
MM = 2
DETA = ETA / ( JMAX - 1.0 )
DXI = XIMAX / ( IMAX - 1.0 )
ETA(1) = 0.0
W(1,1) = 0.0
W(2,1) = 0.0
DO 10 J = 2, JMAX
  ETA(J) = ETA(J-1) + DETA
  W(1,J) = 1.0
10  W(2,J) = W(2,J-1) - 0.5 * DETA * ( W(1,J) + W(1,J-1) )
DO 40 ITER = 1, 20
  CALL COEF(MM, JMAX, E, EL, AM, BM, CM, DM, IN, W, WL, AL, BE)
  CALL SOLU(W, JMAX, MM, E, EL)
  CF = SQRT(2.0) * ( 2.0 * W(1,2) - 0.5 * W(1,3) ) / DETA
  PRINT 9600, ITER, CF
9600 FORMAT( 20X, I3, 5X, E16.9)
40  CONTINUE
PRINT 9500, (J, ETA(J), W(1,J), W(2,J), J=1, JMAX)
9500 FORMAT(10X, I3, 2X, E12.5, 2X, E12.5, 2X, E12.5)
DO 100 I = 2, IMAX
  XI = XI + DXI
  XIH = XI - 0.5 * DXI
  XIDXI = XIH / DXI
  BETA = 0.25 * XIH / ( 0.25 * XIH - 1.0 )
DO 60 J = 1, JMAX
DO 50 M = 1, MM

```

```

50 WL(M,J) = W(M,J)
60 CONTINUE
DO 70 ITER = 1,2
CALL COEF(MM,JMAX,E,EL,AM,BM,CM,DM,IN,W,WL,AL,BE)
CALL SOLU(W,JMAX,MM,E,EL)
70 CONTINUE
FACTOR = (2.0-0.5*XI)/SQRT(1.0+SQRT(1.0-0.25*XI))
CF = FACTOR * (2.0*W(1,2) - 0.5*W(1,3)) / DETA
PRINT 9700, I, XI, CF, BETA
9700 FORMAT(10X,I3,3(5X,E12.5))
100 CONTINUE
PRINT 9500, (J,ETA(J),W(1,J),W(2,J),J=1,JMAX)
RETURN
END

```

```

SUBROUTINE COEF(MM,JMAX,E,EL,AM,BM,CM,DM,IN,W,WL,AL,BE)
COMMON /A1/ DETA, BETA, I, XIDXI
DIMENSION IN(MM),E(MM,MM,JMAX),EL(MM,JMAX),W(MM,JMAX),WL(MM,JMAX)
DIMENSION AM(MM,MM),BM(MM,MM),CM(MM,MM),DM(MM)
DIMENSION AL(MM,MM),BE(MM)
DO 15 M = 1, MM
DO 14 N = 1, MM
AM(M,N)=0.0
BM(M,N)=0.0
CM(M,N)=0.0
14 DM(M) = 0.0
BM(1,1)=1.0
BM(2,2)=1.0
CALL EEL(1,MM,JMAX,E,EL,AM,BM,CM,DM,IN,AL,BE)
DO 20 J = 2, JMAX
AM(2,1) = - 0.5*DETA
AM(2,2) = 1.0
BM(2,1) = 0.5*DETA
BM(2,2) = 1.0
IF ( I.EQ. 1 ) GO TO 16
AM(2,1) = - ( 0.25 + XIDXI ) * DETA
BM(2,1) = - AM(2,1)
DM(2) = DETA * ( -0.25 + XIDXI ) * ( WL(1,J) + WL(1,J-1) )
16 IF ( J.EQ. JMAX ) GO TO 18
AM(1,1) = 0.5 * ( 1.0 + 0.5 * DETA * W(2,J) )
BM(1,1) = 1.0 + ( DETA**2 ) * BETA * W(1,J)
BM(1,2) = 0.25 * ( W(1,J+1) - W(1,J-1) ) * DETA
CM(1,1) = 0.5 * ( 1.0 - 0.5 * DETA * W(2,J) )
DM(1) = 0.5*(DETA**2)*(BETA*(1.0+W(1,J)**2)+W(2,J)*0.5*
1 (W(1,J+1)-W(1,J-1))/DETA)
IF ( I.EQ. 1 ) GO TO 20
BM(1,1) = BM(1,1) + (DETA**2)*2.0*XIDXI*W(1,J)
BM(1,2) = BM(1,2) + 0.25*( WL(1,J+1) - WL(1,J-1) ) *DETA
DM(1) = DM(1) + 0.5*(WL(1,J+1)-2.0*WL(1,J)+WL(1,J-1)) + (DETA**2)*
1 (0.5*BETA*(1.0-WL(1,J)**2)+XIDXI*(W(1,J)**2+WL(1,J)**2))
GO TO 20
18 CONTINUE
AM(1,1) = 0.0
BM(1,1) = 1.0
BM(1,2) = 0.0
DM(1) = 1.0
20 CALL EEL(J,MM,JMAX,E,EL,AM,BM,CM,DM,IN,AL,BE)
RETURN
END

```

These results are given in Figure 13 showing that the method is second-order accurate in the ξ -direction. The "exact" result above is obtained with Richardson extrapolation. The distribution of the skin friction parameter along the flat plate is given below with the calculation performed with IMAX = 65:

ξ	$c_f \sqrt{Re_x}$
0	0.665140
0.2	0.560344
0.4	0.447755
0.6	0.322124
0.8	0.166245

Figure 13. Accuracy of Skin Friction for Howarth Problem

The influence of the number of iterations at each marching steps on the accuracy of the skin friction is determined. When the coefficients in Eq (108) are evaluated without iteration (barred quantities are evaluated at "i"), the results for skin friction at $\xi = 0.8$ are

<u>IMAX</u>	<u>$C_f \sqrt{Re_x}$</u>	<u>% Error</u>	<u>NI</u>	<u>Time (sec)</u>
9	0.184414	10.9	28	2.7
17	0.176358	6.10	36	3.0
33	0.171674	3.28	52	3.7
65	0.169070	1.72	84	4.8
257	0.166958	0.448	276	12.2
613	0.166529	0.190	632	26.1
∞	0.166214			

Percent error in the skin friction is given in Figure 13, along with the previous result for one iteration at each marching step. Further iteration at each step will result in a slight improvement in the accuracy and is usually not considered worthwhile. The results with one iteration are second-order accurate in the ξ -direction while the results without iteration are first-order with a sufficiently small step-size $\Delta\xi$. The results with iteration require significantly fewer intervals

in the marching direction and one would expect the computation time to be substantial less when more accurate results are desired.

The computation time for the solutions is given above and a large part of this time results from the solution of the block-tridiagonal equation. Let NT represent the number of times these equations are solved and is obtained from

$$NT = \begin{cases} 20 + (IMAX - 1) & \text{No iterations} \\ 20 + 2(IMAX - 1) & \text{One iteration} \end{cases} \quad (115)$$

where 20 solutions are used to obtain the initial profiles at $\xi = 0$. The computation time is expected to be proportional to NT and the number of intervals across the boundary layer. This type of correlation is used in Figure 14 and the CDC 6600 computer time (sec) is represented with the equation

$$T = 1.6 + 0.030 (NT)(JMAX - 1)/30 \quad (116)$$

Figure 14. Computation Time on CDC 6600

With the above information, the computer time required for various accuracies of the skin friction can be determined. The number of intervals, IMAX, is determined from Figure 13 and Eqs (115) and (116) are used to determine the computer time required. The results with JMAX = 31 are

Desired Accuracy of $C_f \sqrt{Re_x}$ (%)	IMAX		Computer Time	
	No Iterations	One iteration	No Iterations	One Iteration
10	9	3	2.7	2.5
1	110	9	6.6	3.0
0.1	1200	28	49.1	4.5

When accurate solutions are needed, the method with one iteration saves a significant amount of computer time. The formulation of the first-order method (no iterations) used in this study could be replaced with a scheme which requires the solution of scalar tridiagonal matrices and would be much faster. For low accuracy results, a first-order scheme without iterations probably requires less computer time than a second-order method with iteration at each marching step.

References

1. Herman Schlichting, Boundary-Layer Theory (New York: McGraw-Hill Book Company, 1968).
2. L. Rosenhead, ed, Laminar Boundary Layers (London: Oxford University Press, 1963).
3. F. K. Moore, ed Theory of Laminar Flows, Vol 14, (Princeton, NJ: Princeton University Press, 1964).
4. K. Stewartson, The Theory of Laminar Boundary Layers in Compressible Fluids (London: Oxford University Press, 1964).
5. S. V. Patankar and D. B. Spalding, Heat and Mass Transfer in Boundary Layers (2nd ed; London: Intertext Books, 1970).
6. N. A. Jaffe and A. M. O. Smith, "Calculation of Laminar Boundary Layers by Means of a Differential-Difference Method," Progress in Aerospace Sciences, Vol 12, ed. D. Küchemann (New York: Pergamon Press, 1972) pp 49-212.
7. J. F. Nash and V. C. Patel, Three-Dimensional Turbulent Boundary Layers (Atlanta: Scientific and Business Consultants, Inc, 1972).
8. T. Cebeci and A. M. O. Smith, Analysis of Turbulent Boundary Layers (New York: Academic Press, 1974).
9. F. G. Blottner, "Computational Techniques for Boundary Layers," (AGARD Lecture Series, No 73, 1975).
10. Maurice Holt, Numerical Methods in Fluid Dynamics (Berlin: Springer-Verlag, 1977).
11. Tuncer Cebeci and Peter Bradshaw, Momentum Transfer in Boundary Layers (Washington: Hemisphere Publishing Corp, 1977).
12. D. B. Spalding, GENMIX - A General Computer Program for Two-Dimensional Parabolic Phenomena (New York: Pergamon Press Inc, 1977).
13. Herbert B. Keller, "Numerical Methods in Boundary-Layer Theory," Annual Review of Fluid Mechanics, ed Milton Van Dyke (Palo Alto, California: Annual Reviews Inc, 1978).
14. Lu Ting, "On the Initial Conditions for Boundary Layer Equations, J Math and Phys, 44(4):353-67, 1965.
15. Philip M. Morse and Herman Feshbach, Methods of Theoretical Physics, (New York: McGraw-Hill Book Company, Inc, 1953), p 691.
16. H. B. Keller, Numerical Method for Two-Point Boundary-Value Problems (Waltham, Mass: Blaisdell Publishing Company, 1968).
17. P. J. Roache, Computational Fluid Dynamics, (Albuquerque: Hermosa Press, 1972).
18. G. E. Forsythe and C. B. Moler, Computer Solutions of Linear Algebraic Equations (New York: Prentice-Hall, 1967).
19. M. R. Scott and H. A. Watts, "Computational Solution of Linear Two-Point Boundary-Value Problems via Orthonormalization," Siam J. Numer Anal 14: March 1977.
20. James E. Carter, Inverse Boundary-Layer Theory and Comparison with Experiments, US, NASA, Technical Paper 1208, (Washington: Government Printing Office, Sept 1978).

21. Norbert Peters, "Solutions of Boundary Layer Equations for Chemically Reacting Gases with a Multi-Point Method" (PhD dissertation, Technical University Berlin, Germany, 1971).
22. E. Krause, E. H. Hirschel, and W. Kordulla, "Fourth-Order "Mehrstellen" - Integration for Three-Dimensional Turbulent Boundary Layers," AIAA Computational Fluid Dynamics Conference, July 1973.
23. R. S. Hirsh, "Higher Order Accurate Difference Solutions of Fluid Mechanics Problems by a Compact Differencing Technique," J Computational Phys, 19(1):90-109, 1975.
24. S. G. Rubin and P. K. Khosla, "Polynomial Interpolation Methods for Viscous Flow Calculations," J Computational Phys, 24:217-244, 1975.
25. Stephen F. Wornom, "A Critical Study of Higher-Order Numerical Methods for Solving the Boundary-Layer Equations," AIAA Computational Fluid Dynamics Conference, June 1977.
26. Yves Adam, "Highly Accurate Compact Implicit Methods and Boundary Conditions," J Computational Phys, 24:10-22, 1977.
27. M. Ciment, S. H. Leventhal, and B. C. Weinberg, "The Operator Compact Implicit Method for Parabolic Equations," J Computational Phys, 28:135-166, 1978.
28. F. Thiele, "Accurate Numerical Solutions of Boundary Layer Flows by the Finite-Difference Method of Hermitian Type," J Computational Phys, 27:138-159, 1978.

DISTRIBUTION:

AIAA Journal
1290 Ave. of the Americas
New York, NY 10019
Attn: R. Brynes

Aerodynamics Projects Branch/VKF
Arnold Air Force Station, TN 37389
Attn: J. C. Adams, Jr

University of Southern California
Aerospace Sciences Lab
University Park
Los Angeles, CA 90007
Attn: H. K. Cheng

Princeton University
Dept of Aerospace & Mechanical
Science
Princeton, NJ 08540
Attn: S. I. Cheng

US Naval Weapons Lab
Dahlgren, VA 22418
Attn: B. Clark, Code GBJ

Dept of the Navy
Office of Naval Research
Washington, DC 20360
Attn: M. Cooper, Code 438

Fluid Mechanics
P.O. Box 369
Oak Ridge, TN 37830

University of Cincinnati
Dept of Aerospace Engr & Applied
Mechanics
Cincinnati, OH 45221
Attn: R. T. Davis

University of California
Mechanical Engineering Dept
Davis, CA 95616
Attn: H. A. Dwyer

Los Alamos Scientific Lab
Los Alamos, NM 87544
Attn: T. Hirt, T-3

NASA Langley Research Center
Hampton, VA 23365
Attn: J. E. Harris

University of California
Aeronautical Sciences
Berkeley, CA 94707
Attn: M. Holt

California Institute of Technology
Pasadena, CA 91109
Attn: H. Keller

Purdue University
School of Aeronautics, Astronautics
& Engr Sciences
Lafayette, IN 47909
Attn: C. P. Kentzer

NASA/JSC
Houston, TX 77058
Attn: D. Curry, ES32

Virginia Polytechnic Institute and
State University (2)
College of Engineering
Blacksburg, VA 24061
Attn: C. Lewis
Dept of Aerospace Engr
A. H. Nayfeh
Dept of Engr Sciences
& Mechanics

Lockheed Electronics Co
Mail Code A-11
16811 El Camino Real
Houston, TX 77058
Attn: Chien-peng Li

NASA Ames Research Center
Moffett Field, CA 94035
Attn: R. W. MacCormack, MS 223-1

Polytechnic Institute of Brooklyn (2)
Graduate Center
Route 110
Farmingdale, NY 11735
Attn: G. Moretti
S. C. Rubin

North Carolina State University (3)
Dept of Mechanical and Aerospace
Engineering
Raleigh, NC 27607
Attn: J. C. Williams (2)
F. DeJarnette

Aerotherm Corp
485 Clyde Ave.
Mountain View, CA 94040
Attn: R. Kendall

University of Tennessee
Dept of Engr Science & Mechanics
Knoxville, TN 37916
Attn: A. J. Baker

P. J. Roache
4925 Kathryn Circle SE
Albuquerque, NM 87108

DISTRIBUTION (cont)

US Army Research Office
Box CM, Duke Station
Durham, NC 27706
Attn: R. E. Singleton
Chief, Fluid Mechanics Branch

ALR (LH)
Bldg 450
Wright-Patterson AFB, OH 45433
Attn: W. Hankey

Douglas Aircraft Co
McDonnell Douglas Corp
3855 Lakewood Blvd
Long Beach, CA 90801
Attn: T. Cebeci

University of Cincinnati (2)
Dept of Aerospace Engr
Cincinnati, OH 45221
Attn: U. Ghia
K. Ghia

General Electric Co
Space Sciences Lab, Rm L-9515
P. O. Box 8555
Philadelphia, PA 19101
Attn: T. C. Dellinger

Flow Research, Inc
1819 Central Ave. South
Kent, WA 98031
Attn: E. Murman

Gruman Aerospace Corp
Research Dept
Bethpage, NY 11714
Attn: R. E. Melnik

Notre Dame University
Dept of Aerospace & Mech Engr
Notre Dame, IN 46556
Attn: T. J. Mueller

National Science Foundation
Div of Mathematics, Physical and
Engr Sciences
Washington, DC 20550

NASA Ames Research Center
Moffett Field, CA 94035
Attn: J. Rakich

Exterior Ballistics Research Lab,
USARDC
Aberdeen Proving Ground, MD 21105
Attn: R. Sedney

Aerospace Corp
2350 E. El Segundo Blvd
El Segundo, CA 90245
Attn: T. D. Taylor

Naval Ship Research & Development Center
Bethesda, MD 20034
Attn: C. VonKerczek

Howard University
Mechanical Engr Dept
Washington, DC 20001
Attn: C. B. Watkins

United Technology Research Center
Gas Dynamic Section
East Hartford, CT 06108
Attn: M. Werle

Stanford University (3)
Stanford, CA 94305
Attn: M. Van Dyke
J. Ferziger
Dept of Mechanical Engr
W. M. Kays
School of Engineering

University of Washington
Dept of Mechanical Engr
Seattle, WA 98105
Attn: F. B. Gessner

Naval Academy
Aerospace Engr Dept
Annapolis, MD 21402
Attn: D. Rogers

Martin-Marietta Corp
Research Institute for Advanced Studies
1450 South Rolling Rd
Baltimore, MD 21227
Attn: K. C. Wang

University of Maryland
Dept of Aerospace Engr
College Park, MD 20742
Attn: J. Anderson

Air Force Rocket Propulsion Lab
Edwards Air Force Base, CA 93523
Attn: J. N. Levine

Universite' Pierre et Marie Curie
Place Jussieu
75005 Paris, France
Attn: R. Peyret

DISTRIBUTION (cont)

Aerodynamisches Institut
RWTH Aachen
Aachen, West Germany
Attn: E. Krause

2623 M. R. Scott
5500 O. E. Jones
5510 D. B. Hayes
5511 D. F. McVey
5511 F. G. Blottner (25)
5512 H. C. Hardee
5512 M. R. Baer
5520 T. B. Lane
5530 W. Herrmann
5630 R. C. Maydew
8331 R. J. Kee
8354 B. R. Sanders
8266 E. A. Aas
3141 T. L. Werner (5)
3151 W. L. Garner (3)
for DOE/TIC (Unlimited Release)
DOE/TIC (25)
(R. P. Campbell, 3154-3)