AN IMPLICIT NUMERICAL SOLUTION FOR THE LAMINAR AND TURBULENT FLOW OF AN INCOMPRESSIBLE FLUID ALONG THE AXIS OF A 90-DEGREE CORNER

by

David Tillman Klinksiek

Thesis submitted to the Graduate Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of
DOCTOR OF PHILOSOPHY
in
Mechanical Engineering

APPROVED:

H. L. Moses, Chairman

R. A. Comparin

E. F. Brown

R. K. Will

C. B. Ling

August, 1972

Blacksburg, Virginia

ACKNOWLEDGMENTS

To his wife, , the author expresses his most sincere love and appreciation for her understanding and encouragement during the time spent to satisfy the requirements for this degree.

The author is deeply indebted to Dr. J. B. Jones for employing him as an Instructor in the Mechanical Engineering Department. For without this employment, the author would not have had the opportunity to further his graduate training.

The author also extends his sincere appreciation to his advisory committee, Drs. H. L. Moses (Chairman), R. A. Comparin, E. F. Brown, R. K. Will, and C. B. Ling, for their advice, criticism, and encouragement during the author's course of study and the preparation of this thesis.

In addition, the author expresses his gratitude to

for typing the major portion of this manuscript.

Finally, to his three children, the author extends his gratitude for their love and understanding during a most difficult time.

TABLE OF CONTENTS

TABLE OF CONTENTS - continued

LIST OF FIGURES

LIST OF FIGURES - continued

LIST OF TABLES

TABLE OF NOMENCLATURE

A	van Driest's damping factor
$A_n(i), A_n(j)$	Matrix coefficient
$B_n(i), B_n(j)$	Matrix coefficient
$C_n(i), C_n(j)$	Matrix coefficient
C_f	Skin friction
$D_n(i), D_n(j)$	Matrix coefficient
D_{ij}	Rate-of-strain tensor for laminar flow
$\overline{D}_{ij}$	Time average rate-of-strain tensor for turbulent flow
F_i	Body force
H	Dummy variable
H_s	Boundary layer shape parameter, δ^*/θ
J	Defined by equation 3.16
L	Nondimensionalized mixing length for two-dimensional boundary layer
L_c	Nondimensionalized mixing length for corner region
L_p	Plate length and reference length
$\overline{P}$	Time average pressure
Q_o	Reference velocity
Q_i	Velocity vector for laminar flow with components U, V, and W
$\overline{Q}_i$	Time average velocity vector for turbulent flow with components $\overline{U}$, $\overline{V}$, and $\overline{W}$
Re_x	Reynolds number based on length from leading edge
Re_θ	Reynolds number based on momentum thickness

TABLE OF NOMENCLATURE - continued

U,V,W	Velocity components for laminar flow
$\overline{U},\overline{V},\overline{W}$	Time average velocity components
$U_\infty,V_\infty,W_\infty$	External velocity components for laminar flow
$\overline{U}_\infty,\overline{V}_\infty,\overline{W}_\infty$	Time average external velocity components for turbulent flow
X_i	Displacement vector with components X, Y, and Z
X,Y,Z	Displacement vector components
i	Index for y direction
j	Index for z direction
n	Index for x direction
k	Upper bound on i index within the corner region
m	Upper bound on j index within the corner region
ℓ	Mixing length: two-dimensional boundary layer
ℓ_c	Mixing length: corner region
r_1,r_2,r_3,r_4	Defined by equations B.2a-b and C.2a-b in Appendix A
$s_1,s_2,s_3,s_4,$ $s_5,s_6,s_7,s_8,$ $s_9,s_{10},s_{11},$ s_{12},s_{13},s_{14}	Defined by equations B.2c-d, C.2c-d, D.2a-d, E.3a-d, and E.7a-b in Appendix A
u,v,w	Nondimensionalized velocity components corresponding to $\overline{U}$, $\overline{V}$, and $\overline{W}$ and U, V, and W
$\bar{u},\bar{v},\bar{w}$	Nondimensionalized space average velocity components in finite-difference grid
u^*,v^*,w^*	Half-step nondimensionalized velocity components in finite-difference grid

TABLE OF NOMENCLATURE - continued

u_s	Shear velocity, $(\tau_\omega/\rho)^{1/2}$
u^+	$\bar{U}/u_s$
x,y,z	Nondimensionalized displacements corresponding to X, Y, and Z
$\Delta x, \Delta y_-, \Delta y_+, \Delta z_+, \Delta z_-$	Finite-difference increments. At a given node, the increment with a + subscript is always in the positive direction to the next node and greater than or equal to the negatively subscripted increment.
y^+	Yu_s/ν
δ	Boundary layer thickness defined as positive where $\bar{U}/U_\infty = .99$ in the boundary layer
δ_ℓ	Two-dimensional boundary layer thickness parameter used in mixing length model and defined as position where $\bar{U}/U_\infty = .999$ in the boundary layer
δ_{ℓ_c}	Boundary layer thickness parameter used in corner region mixing length model
δ^*	Displacement thickness defined by equation 4.1
ε	Eddy-viscosity
ε_n	Nondimensionalized eddy-viscosity
θ	Momentum thickness defined by equation 4.2
μ	Dynamic viscosity
ν	Kinematic viscosity
ρ	Density
τ_ω	Wall shear stress

I. INTRODUCTION

The growth and general characteristics of two-dimensional laminar and turbulent boundary layers have been analyzed for many years by the momentum integral method. With the advent of the digital computer, numerical techniques for the analysis of two-dimensional boundary layers have become possible.

The standard momentum integral technique used to evaluate laminar boundary layers is the Karman-Pohlhausen method. If the pressure gradient is too severe, Launder's method of dividing the boundary-layer into strips can be applied with success. For either of the methods mentioned no empirical input is necessary other than the velocity profile assumption for each method. If a numerical technique is used, the governing partial differential equations are solved directly with the application of the appropriate boundary conditions. No assumption about the velocity profile is necessary since the profile will be predicted by the solution technique.

A momentum integral method for the evaluation of turbulent boundary layers requires a shear stress correlation in addition to a velocity profile model. When a numerical technique is used to analyze a turbulent boundary layer, a model for the shear stress, or turbulence mixing, across the boundary layer is required. Thus, unlike the analysis of laminar boundary layers by a numerical method, the evaluation of turbulent boundary layers by a numerical method does require an empirical input for the shear stress across the boundary layer.

Extending this discussion to three-dimensional laminar and turbulent boundary layers, additional empirical information is required to solve the three-dimensional flow field by a momentum integral method. For either flow classification, a model for the transverse velocity profile is necessary and, in addition, for turbulent flow, a shear stress correlation is required. To solve the three-dimensional flow field by a numerical method, no empirical input is required for laminar flow. However, analysis of turbulent flow by a numerical method again requires a model for the shear stress within the flow region.

The flow of a fluid along the interior axis of a corner formed by two planes intersecting at ninety degrees is a three-dimensional flow phenomenon which results from both walls influencing the flow field near the intersection of the planes. Far removed from either of the planes, but close to the remaining plane, the flow field is two-dimensional.

The analysis of the flow field close to the corner region by a momentum integral method can be accomplished if a suitable model for the velocity profile can be established. For the laminar case, transverse velocities exist, but according to the arguments found in the literature, a closed vorticity pattern is not established or has not been experimentally observed. If the flow is turbulent, closed vorticity patterns have been observed for non-circular ducts. The fluid flows along the bisector toward the corner and then out along the walls to the mid-point of the duct and back into the free stream.

Thus a secondary flow is superimposed on the free stream.

This investigation has adopted a numerical method which was used to predict the three components of velocity in the corner region for either laminar or turbulent flow. A model for the shear stress across a two-dimensional turbulent boundary layer was modified for the corner geometry when turbulent flow was evaluated. This numerical method gave good results for laminar flow when comparisons were made with other analytical solutions which were obtained by different numerical techniques. This numerical method also produced good results when a comparison was made with the one experimental work found in the literature for turbulent flow.

The numerical method developed in this investigation is sufficiently general so that adverse pressure gradient flows and developing flows in enclosing ducts can be analyzed. However, to perform the analysis, some minor revisions must be made in the numerical method.

II. REVIEW OF LITERATURE

A. Investigations of Laminar Corner Flow

The first attempt at an analytical solution to the problem of flow in the corner region was by Carrier (1). His solution was obtained by solving only the streamwise component of the Navier-Stokes equations in conjunction with the continuity equation. Asymptotic series were used to represent the solutions for u, v, w, and p. The solution for the u profile was taken to be a combination of the Blasius profiles, which should exist on the two surfaces at points far removed from the corner, plus a function which has influence only in the corner. Results are given only for the streamwise isovels in the corner region.

Due to criticism by Kemp (2), Dowdell (3) investigated the effect that the neglect of the crosswise components of the Navier-Stokes equations would have on the streamwise velocity component in the corner. Further, Dowdell (4) also obtained a system of equations amenable to the solution of the flow field along the intersection of the planes at any angle. His analysis indicated that a 4 percent difference occurred between his analysis and that of Carrier (1) in the case of planes intersecting at ninety degrees for the mainstream velocity component in the corner region. Dowdell's (4) analysis further indicated that a flow reversal occurred deep in the corner region for planes intersecting at small angles.

Rubin (5) discussed the above references and a method of solving the flow field in the corner. He divided the flow into three regions: an irrotational flow region, a two-dimensional boundary layer on each surface far removed from the other surface, and the corner region where both surfaces influence the flow. Solutions for the potential flow and the two-dimensional boundary layer were obtained as asymptotic expansions which utilized the small perturbation parameter ($\nu/2U_\infty x$). These solutions were then matched at the intersecting boundaries of the various regions.

Pal and Rubin (6) presented the asymptotic solution to the corner region equations. Again, the small perturbation parameter was ($\nu/2U_\infty X$) and stretched similarity variables were used for the cross-flow directions. This led to a system of four partial differential equations which were then solved by the asymptotic expansion method. The expansions were matched to the asymptotic expansions of the two-dimensional boundary layer and the irrotational potential core presented by Rubin (5).

Rubin and Grossman (7) reduced the governing equations of the flow in the corner region to four Poisson-like partial differential equations which were solved numerically by the Gauss-Seidel method of successive iterations. The asymptotic expansion of Pal and Rubin (6) was used to establish the boundary values for the numerical analysis. In this series, a single arbitrary constant of unknown value was present. This constant was varied until interior mass sources were no longer generated by the numerical method.

Significant differences were obtained when a comparison was made between Rubin and Grossman (7) and Carrier (1). The results of Rubin and Grossman (7) were in some cases 25 percent greater than the results of Carrier (1) for the streamwise velocity component along the corner bisector. Also, the isovels of Rubin and Grossman (7) did not have the sharp curvature that Carrier (1) isovels indicated. The skin friction from the corner to the two-dimensional boundary layer region was found to increase monotonically from zero at the corner to its asymptotic values at the two-dimensional boundary layer.

B. Investigations of Turbulent Corner Flow

Most of the work attempted in the turbulent flow along a corner has been in the fully developed region of square and rectangular ducts. Nikuradse (8) was the first to study this type of flow experimentally and observed that the mainstream isovels were displaced toward the walls at the corner and away from the walls at the mid-point between the corners. Gessner (9) has presented detailed experimental results and analysis of the turbulence and mean flow characteristics of fully developed flow in a rectangular channel.

Bragg (10) presented experimental results and attempted to analyze the turbulent boundary layer along the corner. He used a momentum integral technique in which a streamwise simplified Navier-Stokes equation was integrated across the corner layer in conjunction with the continuity equation.

In order to integrate the momentum equation across the corner layer, the streamwise velocity profile must be specified as a function of both y and z, which are the cross-flow coordinates. In the laminar sub-layer, Bragg defines a velocity scale $U_c = \left(\nu \frac{\partial^2 U}{\partial y \partial z}\right)^{1/3}_{\substack{y=0 \\ z=0}}$ so that the resultant form of the velocity profile in this region is $U/U_c = (U_c y/\nu)(U_c z/\nu)$. This form is very similar to the $u^+ = y^+$ currently in use to describe the laminar sub-layer U-profile in two-dimensional boundary layer theory. The outer U-profile was assumed to be similar to the classic law of the wall correlation using a shear velocity based on properties of the flow along the bisector and the cross-flow coordinates.

The wall shear stress was assumed to be correlated as a function of the wall shear velocity far removed from the corner (two-dimensional flow) and the distance from the corner. This correlation seemed to hold until a position very deep in the corner region was reached. Another velocity scale determining shear stress in this region was developed and used in the correlation for wall shear stress deep in the corner region.

The comparison of experimental results with the various models for mean streamwise velocity profiles and wall shear stress indicated that fairly good agreement was obtained with no pressure gradient and poor agreement was obtained with an adverse pressure gradient. However, this comparison was only made for the corner bisector.

Toan (11) also attempted an analytical solution to the turbulent flow along a ninety-degree corner. The mainstream momentum equation, together with the continuity equation, was solved assuming that a modified Ludwieg-Tillmann shear stress equation was valid and that the mainstream velocity profile could be represented by a power law. Results were unsatisfactory since the influence of the secondary flow on the mainstream velocity was not taken into account. An improvement in the mainstream velocity profile representation was attempted by introducing a new parameter which was an indication of the intensity of the secondary flow. Also, it was assumed that the shear stress along the wall in the corner was constant from the work of Gersten (12). These assumptions resulted in better predictions of the mainstream isovels but still were not satisfactory since the curvature of the isovels in the vicinity of the bisector was not duplicated. Another method was presented using the isovels as one coordinate and their orthogonal velocity gradient lines as the other coordinate. This method was presented only for a qualitative study of the secondary flow because of the difficulty in predicting the isovels. Toan also mentioned Bragg's (10) work and stated that Bragg's hypothesis for the velocity scale required that the isovels be hyperbolas.

Veenhuizen and Meroney (13) presented measurements of the primary and secondary flows in the developing section of a square duct which was designed so that the pressure gradient was zero. Also, the turbulence distribution was measured at various points in the duct. Results indicate that the primary and secondary flow patterns and the

turbulence distribution are similar to that found in fully developed duct flow investigated by Gessner (9). No attempt was made to solve the momentum, vorticity, and continuity equations.

Launder and Ying (28) presented experimental and analytical results for fully developed turbulent flow in square ducts. Their analysis indicated closed vorticity patterns that agreed reasonably well with the experimental results. However, a coarse grid spacing (11 x 11) was used which made it necessary to fit the mainstream profile with a semi-log correlation.

III. ANALYSIS

Before the problem of turbulent flow along the intersection of two right angle planes was solved, it was advantageous to solve the laminar flow case and then add an appropriate approximation for the Reynolds stresses. The corner flow problem was solved using the Alternating Direction Implicit (ADI) finite-difference technique. Use of this method resulted in the generation of many tridiagonal matrixes which were solved for a half step in the x-direction, stepping implicitly in the y-direction and explicitly in the z-direction. Then, for the second half step in the x-direction, stepping implicitly in the z-direction and explicitly in the y-direction.

A. Governing Equations for Laminar Flow

The equations of motion and continuity for the laminar unsteady flow of an imcompressible fluid are given by Schlichting (14):

$$\rho \frac{DQ_i}{Dt} = \frac{\partial P}{\partial X_i} + \frac{\partial}{\partial X_i} (\mu D_{ji}) + F_i, \qquad i = 1,2,3 \quad j = 1,2,3 \qquad (3.1)$$

where D/Dt is the substantial derivative, and

$$\frac{\partial Q_j}{\partial X_j} = 0. \qquad j = 1,2,3 \qquad (3.2)$$

1. Corner Region Equations

For the case under investigation, the flow was steady and there were no body forces. It was further assumed that the second derivative of any velocity component with respect to the x-direction was many times less than with respect to the y and z-directions. And, since the flow was over a surface similar to a flat plate, the pressure gradient in all three directions was also negligible.

Thus, the equations of motion 3.1 were reduced as follows with the continuity equation 3.2 remaining unchanged:

$$U\frac{\partial U}{\partial X} + V\frac{\partial U}{\partial Y} + W\frac{\partial U}{\partial Z} = \nu\left(\frac{\partial^2 U}{\partial Y^2} + \frac{\partial^2 U}{\partial Z^2}\right) \quad , \tag{3.3a}$$

$$U\frac{\partial V}{\partial X} + V\frac{\partial V}{\partial Y} + W\frac{\partial V}{\partial Z} = \nu\left(\frac{\partial^2 V}{\partial Y^2} + \frac{\partial^2 V}{\partial Z^2}\right) \quad , \text{ and} \tag{3.3b}$$

$$U\frac{\partial W}{\partial X} + V\frac{\partial W}{\partial Y} + W\frac{\partial W}{\partial Z} = \nu\left(\frac{\partial^2 W}{\partial Y^2} + \frac{\partial^2 W}{\partial Z^2}\right) \quad . \tag{3.3c}$$

The equations 3.2 and 3.3a-c were then non-dimensionalized by the following substitutions:

$$U = uU_\infty \quad , \tag{3.4a}$$

$$V = v(L_p/U_\infty\nu)^{-1/2} \quad , \tag{3.4b}$$

$$W = w(L_p/U_\infty\nu)^{-1/2} \quad , \tag{3.4c}$$

$$X = xL_p \quad , \tag{3.4d}$$

$$Y = y(\nu L_p/U_\infty)^{1/2} \quad , \tag{3.4e}$$

and

$$Z = z(\nu L_p/U_\infty)^{1/2} \quad . \tag{3.4f}$$

Substitution of the non-dimensional quantities of equations 3.4a-f into equations 3.2 and 3.3a-c yielded the following results:

$$uu_x + vu_y + wu_z = u_{yy} + u_{zz} \ , \tag{3.5a}$$

$$uv_x + vv_y + wv_z = v_{yy} + v_{zz} \ , \tag{3.5b}$$

$$uw_x + vw_y + ww_z = w_{yy} + w_{zz} \ , \text{ and} \tag{3.5c}$$

$$u_x + v_y + w_z = 0 \ , \tag{3.5d}$$

where the subscripts x, y, and z denote partial derivatives of the quantities u, v, and w with respect to the subscripts, i.e, $\partial u/\partial x = u_x$.

After the equations 3.5a-d were cast into finite-difference form, there resulted three unknowns and four equations because of the assumptions applied to equations 3.1 and 3.2 that all the pressure gradients were zero. Thus, it was decided to use the continuity equation as a check of the solution technique.

A brief review of equations 3.5a-c revealed that these equations were in essence an initial value problem for the x-direction and a boundary value problem for the y and z-directions. This type of partial differential equation can be readily solved using the ADI numerical method.

2. Laminar Two-Dimensional Boundary Layer

For the analysis of the two-dimensional flow along the Y=0 and Z=0 walls far removed from the corner region, the standard boundary layer assumptions were applied to equations 3.5a and 3.5d. The

result of these assumptions yielded the following governing equations for the flow in the two-dimensional boundary layer along the Y=0 wall:

$$uu_x + vu_y = u_{yy} \tag{3.6a}$$

$$\text{and } u_x + v_y = 0 \; . \tag{3.6b}$$

B. Governing Equations for Turbulent Flow

The equations of motion and continuity for the turbulent flow of an incompressible fluid are very similar to the laminar case as given by equations 3.1 and 3.2 except that the Reynolds stresses due to velocity fluctuations must be included. In the following equations, time averaging has been used and the body forces neglected:

$$\rho \frac{D\bar{Q}_i}{Dt} = - \frac{\partial \bar{P}}{\partial X_i} + \frac{\partial}{\partial X_j} \left(\mu \bar{D}_{ji} - \rho \overline{Q_i Q_j} \right) , \qquad \begin{matrix} i=1,2,3 \\ j=1,2,3 \end{matrix} \tag{3.7}$$

$$\text{and} \quad \frac{\partial \bar{Q}_i}{\partial X_i} = 0 \; . \qquad i=1,2,3 \tag{3.8}$$

In equation 3.7, D/Dt is the substantial derivative.

The difference between equation 3.1 and equation 3.7 was the addition of terms representative of the Reynolds stresses. These additional terms rendered the equation 3.7 unsolvable without further information. However, this problem was circumvented by use of the hypothesis advanced by Boussinesq (19) for these stresses, which resulted in a reduction in the complexity of the equation 3.7. His hypothesis is given by the following equation:

$$- \overline{Q_i Q_j} = \varepsilon \bar{D}_{ij} \; . \tag{3.9}$$

Substitution of equation 3.9 into equation 3.7 yielded the following results:

$$\rho \frac{D\overline{Q}_i}{Dt} = - \frac{\partial \overline{P}}{\partial X_i} + \frac{\partial}{\partial X_j} [(\mu+\rho\varepsilon)D_{ji}] \qquad \begin{matrix} i=1,2,3 \\ j=1,2,3 \end{matrix} \qquad (3.10)$$

Assuming that the flow is steady, the pressure gradients are zero or negligible, and the second derivative of any variable with respect to the x axis is many times smaller than derivatives with respect to the other coordinates, the following three equations evolved from equation 3.10 and were used for the flow field in the corner region:

$$\overline{U} \frac{\partial \overline{U}}{\partial X} + \overline{V} \frac{\partial \overline{U}}{\partial Y} + \overline{W} \frac{\partial \overline{U}}{\partial Z} = \frac{\partial}{\partial Y} \left[(\nu+\varepsilon)\frac{\partial \overline{U}}{\partial Y}\right] + \frac{\partial}{\partial Z} \left[(\nu+\varepsilon)\frac{\partial \overline{U}}{\partial Z}\right] , \qquad (3.11a)$$

$$\overline{U} \frac{\partial \overline{V}}{\partial X} + \overline{V} \frac{\partial \overline{V}}{\partial Y} + \overline{W} \frac{\partial \overline{V}}{\partial Z} = \frac{\partial}{\partial Y} \left[(\nu+\varepsilon)\frac{\partial \overline{V}}{\partial Y}\right] + \frac{\partial}{\partial Z} \left[(\nu+\varepsilon)\frac{\partial \overline{V}}{\partial Z}\right] , \text{ and} \qquad (3.11b)$$

$$\overline{U} \frac{\partial \overline{W}}{\partial X} + \overline{V} \frac{\partial \overline{W}}{\partial Y} + \overline{W} \frac{\partial \overline{W}}{\partial Z} = \frac{\partial}{\partial Y} \left[(\nu+\varepsilon)\frac{\partial \overline{W}}{\partial Y}\right] + \frac{\partial}{\partial Z} \left[(\nu+\varepsilon)\frac{\partial \overline{W}}{\partial Z}\right] . \qquad (3.11c)$$

With the addition of the definition,

$$\varepsilon = \nu \varepsilon_n , \qquad (3.12)$$

to equations 3.a-f (the various velocity components are time averaged), the turbulent momentum equations were non-dimensionalized into the following form:

$$uu_x + vu_y + wu_z = [(1+\varepsilon_n)u_y]_y + [(1+\varepsilon_n)u_z]_z , \qquad (3.13a)$$

$$uv_x + vv_y + wv_z = [(1+\varepsilon_n)v_y]_y + [(1+\varepsilon_n)v_z]_z , \text{ and} \qquad (3.13b)$$

$$uw_x + vw_y + ww_z = [(1+\varepsilon_n)w_y]_y + [(1+\varepsilon_n)w_z]_z . \qquad (3.13c)$$

The non-dimensionalized continuity equation, 3.5d, remained the same and is repeated here for clarity:

$$u_x + v_y + w_z = 0 \tag{3.5d}$$

For the two-dimensional boundary layers which were allowed to develop along the walls far removed from the corner region, the y and z momentum equations (equations 3.13b and 3.13c respectively) were zero and the cross flow components, either v or w, depending on which wall the flow was developing, were zero. Also the second derivative with respect to the coordinate parallel to the wall was zero. Thus, to solve for the developing two-dimensional flow along the Y=0 wall, the following equations were used to calculate the flow field:

$$uu_x + vu_y = [(1+\varepsilon_n)u_y]_y \tag{3.14a}$$

and

$$u_x + v_y = 0 \quad . \tag{3.14b}$$

C. Eddy-Viscosity Model

A suitable eddy-viscosity, ε, model was required for the solution of the equations 3.13a-c and 3.14a-b. A literature survey revealed many eddy-viscosity models for two-dimensional flow. The one model which has gained widespread acceptance was that proposed by Prandtl (20). This model in which the eddy-viscosity is formulated as a function of a mixing length and a velocity gradient is given by

$$\varepsilon = \ell^2 \left|\frac{d\bar{U}}{dY}\right| \quad . \tag{3.15}$$

For three-dimensional flows, Prandtl (21) suggests the following replacement for the term $\left|\frac{d\bar{U}}{dY}\right|$ in equation 3.15:

$$J = [2(\bar{U}_x)^2 + 2(\bar{V}_y)^2 + 2(\bar{W}_z)^2 + (\bar{W}_y + \bar{V}_z)^2 + (\bar{U}_z + \bar{W}_x)^2 + (\bar{V}_x + \bar{U}_y)^2]^{1/2} \quad . \tag{3.16}$$

If the usual two-dimensional boundary layer approximations are applied to equation 3.16, J reduces to $\left|\frac{d\bar{U}}{dY}\right|$.

When the usual boundary layer order of magnitude approximations are applied to the various derivatives in equation 3.16, the expression for the eddy-viscosity model in the corner region becomes

$$\varepsilon = \ell_c^2[(\bar{U}_y)^2 + (\bar{U}_z)^2]^{1/2} \quad . \tag{3.17}$$

The eddy-viscosity can be written for the two-dimensional region non-dimensionally as

$$\varepsilon_n = L^2 u_y \tag{3.18a}$$

and for the corner region as

$$\varepsilon_n = L_c^2[(u_y)^2 + (u_z)^2]^{1/2} \quad , \tag{3.18b}$$

where

$$L^2 = \frac{\ell^2}{L_p^2}\left[\frac{\bar{U}_\infty L_p}{\nu}\right]^{3/2} \quad \text{and} \quad L_c = \frac{\ell_c^2}{L_p^2}\left[\frac{\bar{U}_\infty L_p}{\nu}\right]^{3/2} \tag{3.18c}$$

From the recent work of Pletcher (22), the following mixing length model was used for the two-dimensional flow.

$$\frac{\ell}{\delta_\ell} = 0.41\ [1.0 - \exp\ (-y^+/A)]\ \frac{Y}{\delta_\ell} \qquad \frac{Y}{\delta_\ell} < 0.1 \tag{3.19a}$$

$$\frac{\ell}{\delta_\ell} = 0.41\ [1.0 - \exp\ (-y^+/A)\ \frac{Y}{\delta_\ell}$$

$$- 1.53506\ (\frac{Y}{\delta_\ell} - 0.1)^2 + 2.75625\ (Y/\delta_\ell - 0.1)^3$$

$$- 1.88425\ (Y/\delta_\ell - 0.1)^4 \qquad 0.1 \leq Y/\delta_\ell \leq 0.6 \tag{3.19b}$$

$$\frac{\ell}{\delta_\ell} = 0.089 \qquad Y/\delta_\ell > 0.6 \tag{3.19c}$$

δ_ℓ is a normal distance from the wall to a value of $\overline{U}/\overline{U}_\infty$ that is within the boundary layer. However, the value of $\overline{U}/\overline{U}_\infty$ that should be used was not explicity stated by Pletcher (22). A parametric study which solved for $\overline{U}$-profiles on flat plates was made to determine the best value of $\overline{U}/U_\infty$ to use for determining the distance, δ_ℓ. This study indicated that the position in the boundary layer where $\overline{U}/\overline{U}_\infty$ = .999 gave the best distance for δ_ℓ to use in equations 3.19a-c.

The value of the constant A was taken to be equal to 25 since Klinksiek (18) has demonstrated that this value gives the best correlation with the law of the wall constants suggested by Coles (26).

For the corner region, the distance δ_ℓ and Y were modified to conform with the corner geometry. The details of this modification can be found in Appendix A.

D. Finite-Difference Formulations

1. Finite-Difference Approximations for the Two-Dimensional Laminar and Turbulent Boundary-Layer

The Crank-Nicolson (17) implicit finite-difference method was

used to analyze the flow developing in the two-dimensional flat plate region for the following reasons:

(a) Klinksiek (18) has shown that the resultant finite-difference equations are stable and convergent.

(b) Truncation errors are of $O(\Delta x)^2$, $O\Delta y_+$, and $O\Delta z_+$.

(c) Since the finite-difference approximations were implicit, the resultant matrix was tridiagonal, which allowed the use of the Thomas algorithm presented by Von Rosenburg (16) for the solution of the unknowns.

Figure 3.1 shows the grid that was used to generate the finite-difference approximations for the two-dimensional boundary layer flow. The equations were written about the point (i,m+1,n+1/2) for any variable H with a variable grid spacing in the y-direction for flow on the Y=0 wall. The second subscript, m+1, is the index notation for the z-position on the Y=0 wall where the flow is assumed to be two-dimensional. The variable grid spacing was used in the z-direction for flow on the Z=0 wall. Both of the variable grid spacings were the same. The variable grid spacing was used to ensure a high concentration of nodal positions in the region of large velocity gradients. The detailed development of the finite-difference forms of equations 3.6a-b and 3.14a-b can be found in Appendix A.

2. Corner Region Finite-Difference Approximations for Laminar and Turbulent Flow

The Alternating Direction Implicit (ADI) method was selected over the Gauss-Siedel method of solving the set of partial differential

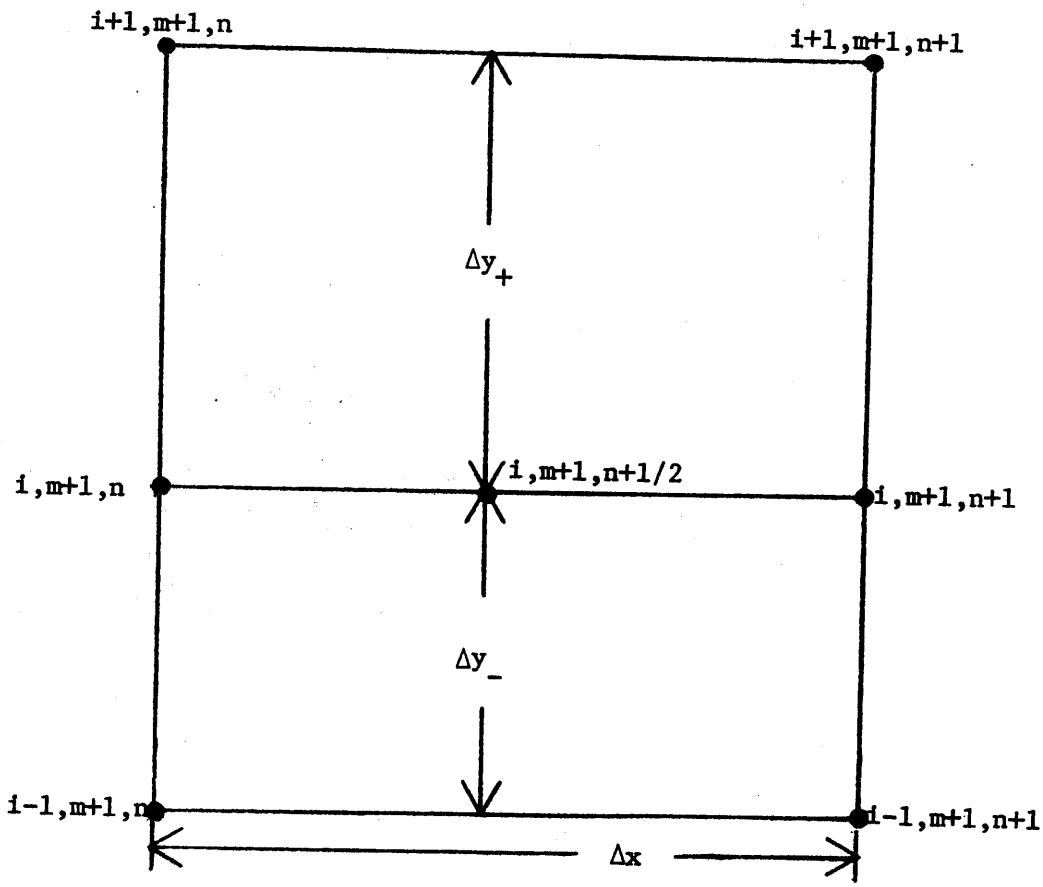

Figure 3.1. Grid for Two-Dimensional Boundary Layer on Y = 0 wall.

equations 3.5a-c and 3.13a-c. The reasons for this selection were as follows:

(i) The ADI method has been shown to be stable and convergent by Peaceman and Rachford (15). (However, stability and convergence were proven only for equal Δy and Δz spacings and linear equations).

(ii) The truncation errors are of order $(\Delta x)^2$, Δy_+, and Δz_+. This allows the use of larger Δx increments as the solution continues.

(iii) The finite-difference equations for this type of method are written such that the matrixes formed are tridiagonal, thus allowing the use of the Thomas algorithm for the solution of the unknowns.

Figure 3.2 illustrates the grid used to generate the finite-difference approximations for the corner region. Variable grid spacing was used in the y and z-directions to ensure a high concentration of nodal locations close to the wall where the largest gradients were expected and also to conserve digital computer storage space since the values of u, v, and w were saved at the initial, intermediate and the new x-position for each nodal point. The variable grid spacing was the same for both the y and z-directions.

The detailed development of the finite-difference form of equations 3.5a-c and 3.13a-c can be found in Appendix A.

E. Boundary and Initial Conditions

The figure 3.3 illustrates the geometric shape (area 1-2-3-4-5-1) in which the finite-difference equations presented in the Appendix A were assumed to approximate the flow. Figure 3.4 is an enlargement of

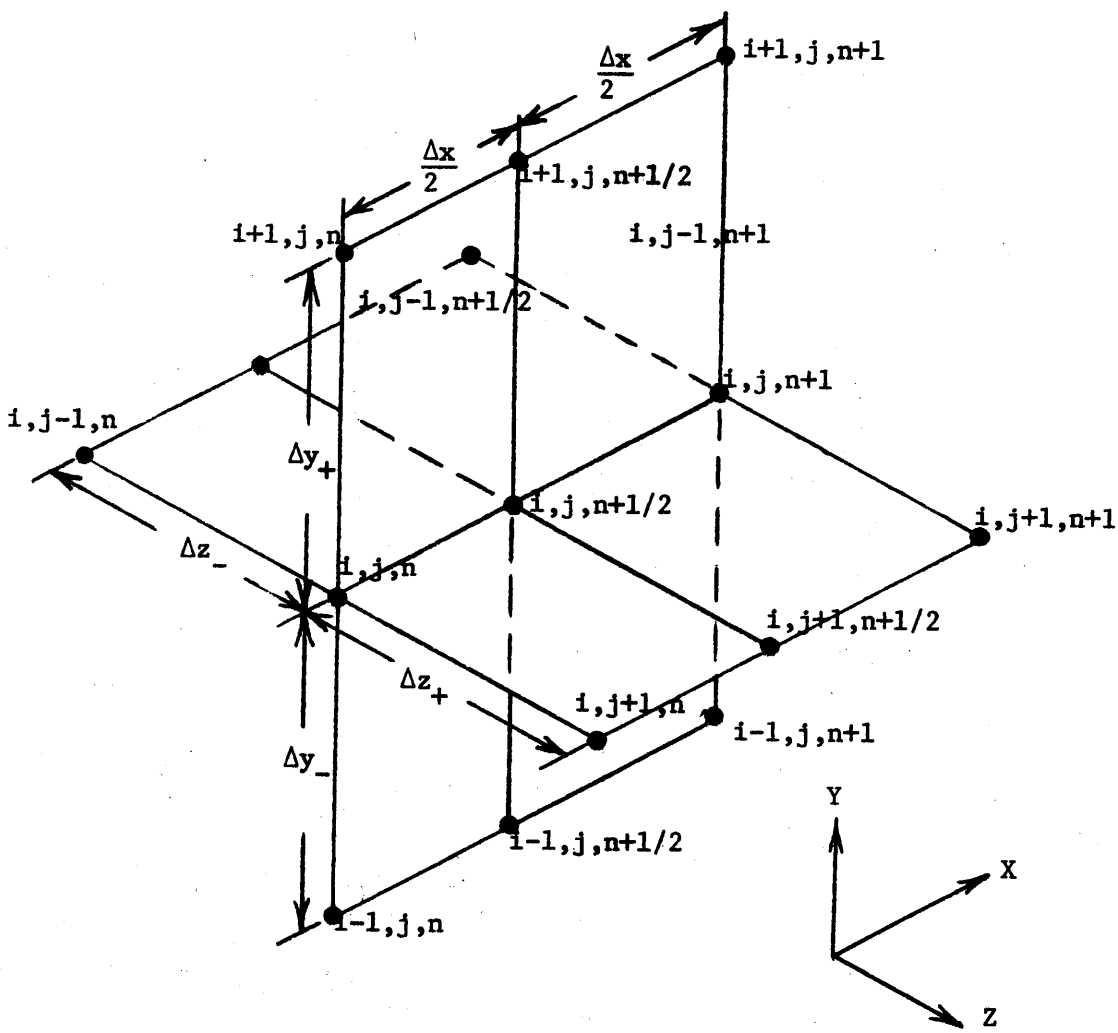

Figure 3.2. Corner Region Finite-Difference Grid.

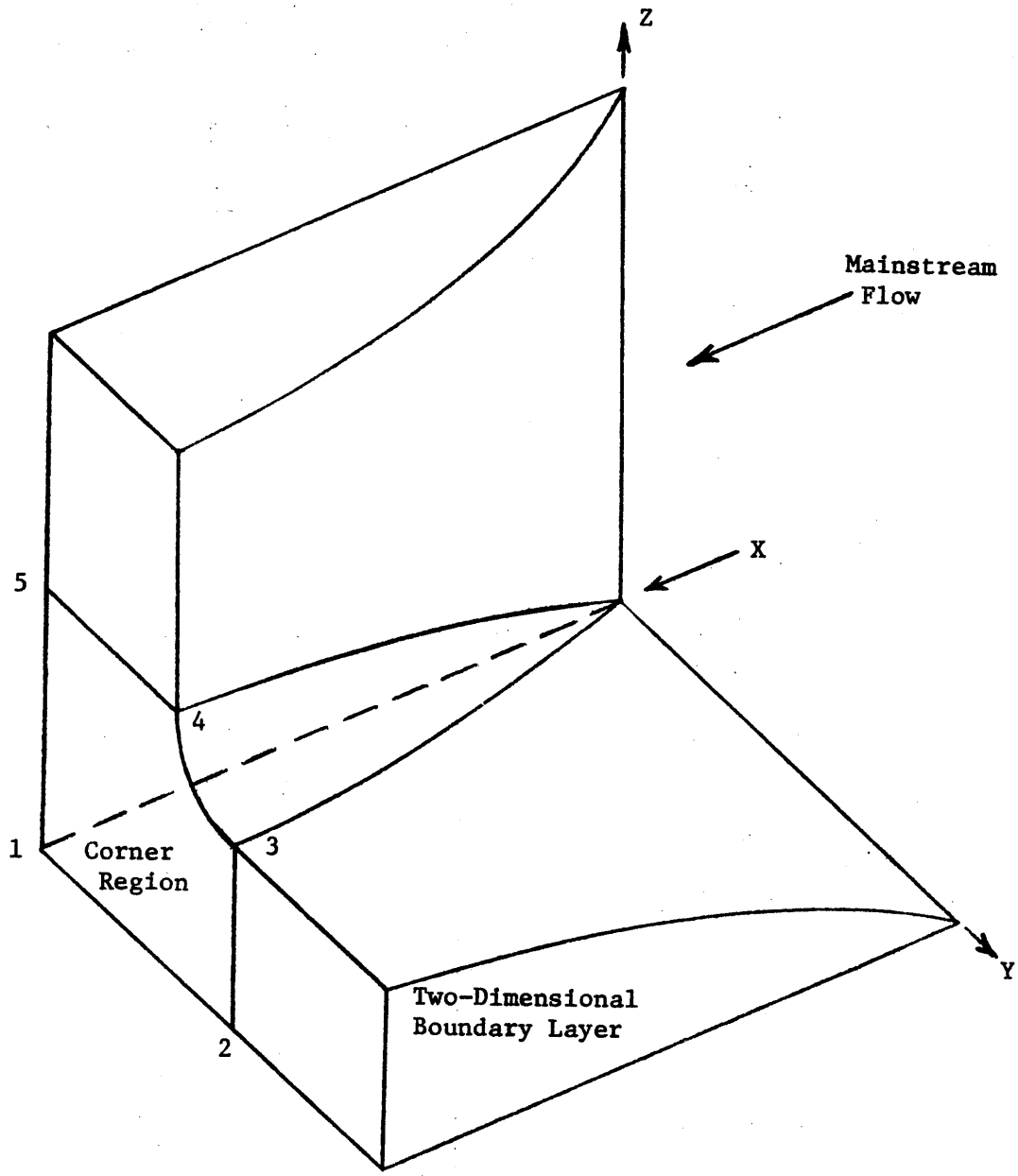

Figure 3.3. General Geometry of Corner.

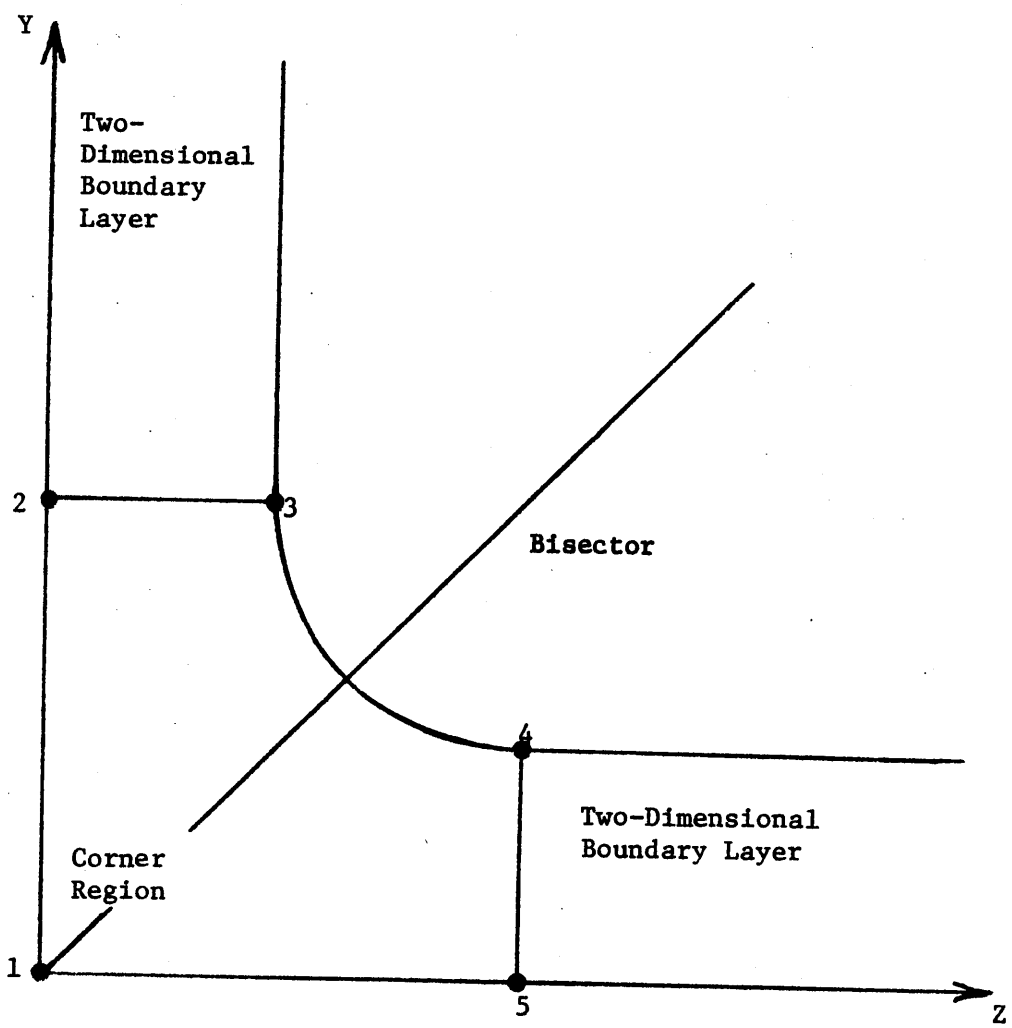

Figure 3.4. Enlargement of Plane 1-2-3-4-5-1 from Figure 3.3.

the plane 1-2-3-4-5-1 to facilitate the following discussion of the corner region and the boundary conditions associated with this region.

Along the Y=0 wall from 1 to 5, the boundary conditions imposed were that all velocity components, including the half step components, were zero, i.e., no slip along the wall and no flow into or out of the wall. A similar condition for the velocity components was imposed along the Z=0 wall from 1 to 2. The line 4-5 was assumed to be the boundary between the corner region and the two-dimensional boundary layer which was developing along the Y=0 wall. Thus, this line 4-5 was assumed to be one of the boundaries for the corner region. The velocity components along this boundary were further assumed to follow the two-dimensional flat plate case and these velocity components were calculated numerically at each x-step. Thus, the u and v velocity components were specified by the development of the boundary layer along a flat plate and the w-velocity component was assumed to be zero along this boundary, i.e., no flow from or into the corner region across this boundary. Similar boundary conditions were established along line 2-3. On this boundary, v=0, and the w component was equal to the v component along the line 4-5. The u components on line 2-3 were equal to the u components on line 4-5.

Along the lines 2-3 and 4-5, the boundary values for the half step velocity components were determined as follows. At each nodal position, the u, v, or w velocity component at the old x-position was summed with the respective u, v, or w component at the new x-position and divided by two.

Along line 3-4 it was assumed that the corner region was merging smoothly into the irrotational free stream. Boundary conditions were set along this line as $u = u_\infty$, $w = w_\infty$ above the bisector, $w = 0$ below the bisector, $v = v_\infty$ below the bisector and $v = 0$ above the bisector, where v_∞ and w_∞ were obtained from the numerical two-dimensional boundary layer calculations.

Another very important consideration was the position of the lines 2-3 and 4-5 with respect to the corner. The location of these lines had a definite influence on the results obtained for the flow field in the corner region. These lines were arbitrarily fixed at the beginning of calculations a given position from the corner. After a certain number of steps in the x-direction, the slope of the u-isovels at lines 2-3 and 4-5 at each nodal position with respect to the coordinate perpendicular to these lines was determined. If the results were greater than the slope of the u-profile in the two-dimensional boundary layer at the position where $U/U_\infty = .99$, the lines 2-3 and 4-5 were re-positioned further from the corner and the corner region flow field recomputed.

The initial conditions were applied at the leading edge of the plates which formed the ninety-degree corner. For the case under investigation, the leading edge was a singularity at which point u was one and v and w were infinite as the boundary layer approximations indicated. However, for this investigation, v and w were assigned values of zero and u was set equal to one since in reality the v and w velocity components at the leading edge should be small if not, indeed, zero.

F. Solution Technique

1. Laminar Case

Because the equations 3.5a-c and 3.6a were non-linear, it was necessary to develop a method of linearization which is presented in Appendix A. The method required that the values for the quantities u, v, and w be assumed known at the new x-position. Since the above mentioned quantities were the variables to be determined at the new x-location, an iterative procedure was used to ensure that the assumed value and the calculated value of the u, v, and w quantities at the new x-position would agree.

The following is an outline of the procedure that was used to solve equations 3.5a-c and 3.6a-b:

(a) Project all values of the variables u, v, and w from the old x-position (i,j,n) to the new x-position (i,j,n+1) for all values of i and j.

(b) Determine the values of u, v, and w in the two-dimensional boundary layers by solving the finite-difference equations presented in Appendix A for equations 3.6a-b.

(c) Determine the values of u, v, and w in the corner region by solving simultaneously the finite-difference equations presented in Appendix A for equations 3.5a-c. The solution was performed three times, each time using the newly calculated values to linearize the equations.

(d) After a certain number of steps were taken in the x-direction, the slope of the u-isovels at each nodal position along

lines 2-3 and 4-5 were moved one nodal position further from the corner and steps (c) and (d) were then repeated.

(e) Repeat steps (a) through (d) for each new position in the x-direction.

Due to the numerical methods used, the resultant form of the finite-difference representations of the equations 3.5a-c and 3.6a always produced matrixes of unknowns in a tridiagonal arrangement. The solution for the unknowns was accomplished by the Thomas algorithm presented by Rosenburg (16). A description of this technique can be found in Appendix A.

2. Turbulent Case

The method used to linearize the turbulent flow equations and the reason for iterating these equations are the same as given in section III. F.1.

In addition, the eddy-viscosity had to be determined at the new x-position. The eddy-viscosity is dependent on the u-profile at the new x-position since the position of $\overline{U}/\overline{U}_\infty = 0.999$ in the boundary layer determines the value of δ_ℓ. Therefore, a check was introduced to ensure that the assumed value of δ_ℓ at the new x-position was within $\pm$ 1 percent of the resultant δ_ℓ when the iteration of the two-dimensional flow equations was completed.

The following is an outline of the procedure that was used to solve equations 3.14a-b and 3.13a-c:

(a) Project all values of the variables u, v, and w from the old x-position (i,j,n) to the new x-position (i,j,n+1)

for all values of i and j.

(b) Determine the values of u, v, and w in the two-dimensional boundary layer by solving the finite-difference equations presented in Appendix A for equations 3.14a-b. A check was performed to ensure that the assumed value of δ_ℓ at the new x-position was within $\pm$ 1 percent of the calculated δ_ℓ value when the iteration of equations 3.14a-b was completed.

(c) Same as step (c) of the laminar case.

(d) Same as step (d) of the laminar case.

(e) Same as step (e) of the laminar case.

The resultant form of the finite-difference representation of equations 3.13a-c and 3.14a always produced matrixes of unknowns in a tridiagonal pattern. Again the Thomas algorithm was used to solve for the unknowns.

3. Step Size and Grid Spacing

For both turbulent and laminar flow, the calculations were started at the leading edge and $\overline{U}_\infty/\nu$ was approximately 6.25×10^{-7} sec/ft. The x-increment size was assigned the physical dimension of 0.001 inches and was held at this value for the first twenty steps. After the twentieth step was calculated, the x-increment was allowed to increase at a rate of 10 percent of the previous x-increment size. However, the x-increment was never allowed to exceed the boundary layer thickness.

The same spacing for the grid increments was used for the y and z-directions. The first five increments from the wall were assigned

equal values of 0.001 inches. The size of each of the next 15 increments was increased in succession by 5 percent of the previous increment size. For the next 20 increments, the percent increase was 10 percent. All the remaining increments were increased by 15 percent.

IV. RESULTS

A. Laminar Flow

The two-dimensional u and v-profiles predicted by this method are compared in Figs. 4.1 and 4.2 with the Blasius solution. As these figures indicate, the method was accurate in predicting the velocity profiles but the prediction of the v-profile was not accurate until Reynolds numbers of over 10,000 were reached. This discrepancy was attributed to the truncation error of the finite-difference equation that was used to represent the continuity equation 3.6b.

Figure 4.3 illustrates the boundary layer thickness prediction capability of this method for the two-dimensional case. Inaccuracies occurred until the Reynolds number was greater than 1000. The inaccuracies of the magnitude depicted by Fig. 4.3 near the leading edge were expected because of the initial conditions applied.

Figure 4.3 also demonstrates the accuracy with which the method can predict the displacement and momentum thicknesses. These thicknesses were obtained by integration of the u-velocity profile by the trapezoidal rule in accordance with equations 4.1 and 4.2, which define the displacement and momentum thicknesses respectively.

$$\delta^* = \int_0^{\delta} (1 - U/U_\infty)\, dy \tag{4.1}$$

$$\theta = \int_0^{\delta} \frac{U}{U} (1 - U/U_\infty) dy \tag{4.2}$$

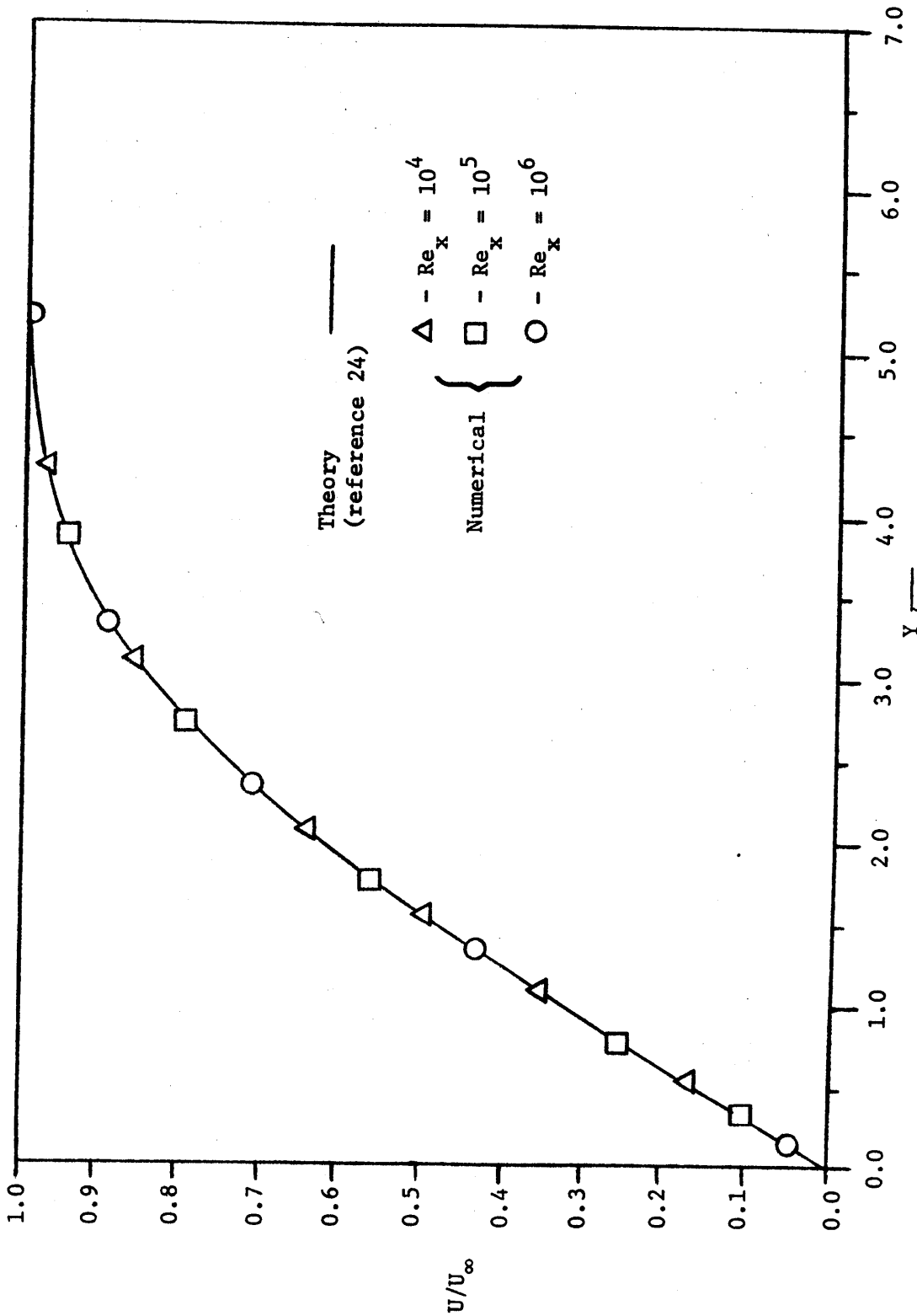

Figure 4.1. U-Velocity Profile Compared with Blasius Solution.

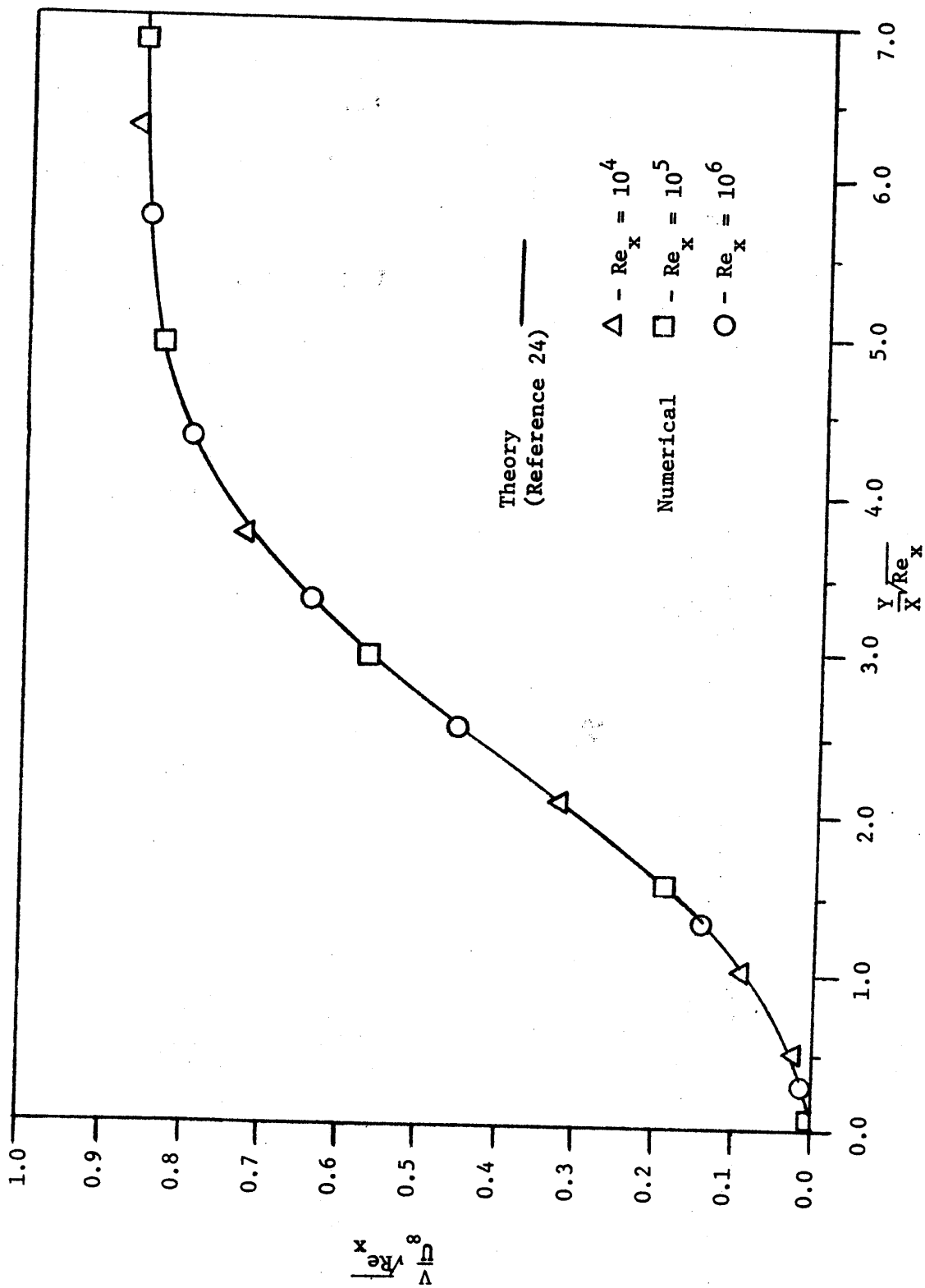

Figure 4.2. V-Velocity Profile Compared with Blasius Solution.

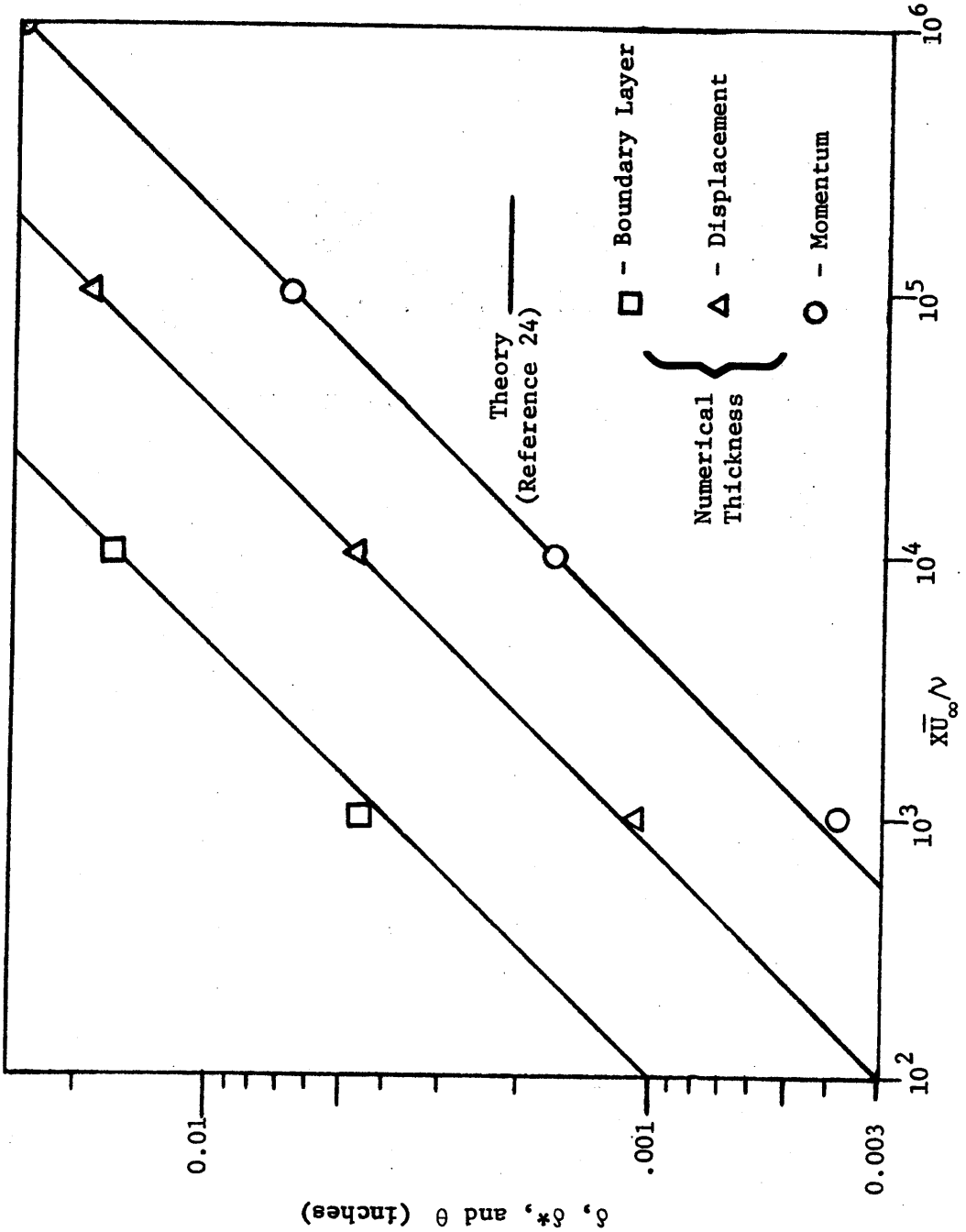

Figure 4.3. δ, δ*, and θ Compared with Blasius Solution.

As is evident in Fig. 4.3, the method accurately predicted the displacement and momentum thicknesses after the Reynolds number reached a value of 1000.

Figure 4.4 compares the predicted skin friction coefficient with theoretical, and again excellent agreement was achieved when the Reynolds number was greater than 1000. The skin friction was calculated by equation 4.3.

$$C_f = 2\tau_\omega / \rho U_\infty^2 \tag{4.3}$$

The shear stress at the wall, τ_ω, was obtained by using the value of u at the first station away from the wall, dividing this value by the normal distance from the wall to this station, and multiplying by the dynamic viscosity. This calculation is given by equation 4.4.

$$\tau_\omega = \mu U(2,m+1,n+1)\Delta y_{wall} \tag{4.4}$$

Figure 4.5 illustrates the results of this numerical technique for the corner region when compared to the techniques advanced by Carrier (1), Dowdell (4), and Rubin and Grossman (7). Only the numerical results at a position sufficiently removed from the leading edge are used for comparison. This method compares favorably with the results of the other investigators of the laminar corner flow problem. Deep in the corner, all of the methods seem to agree with discrepancies developing as the free stream is approached. This method predicts a larger increase in the u-velocity as the free stream is approached than either of the previous methods.

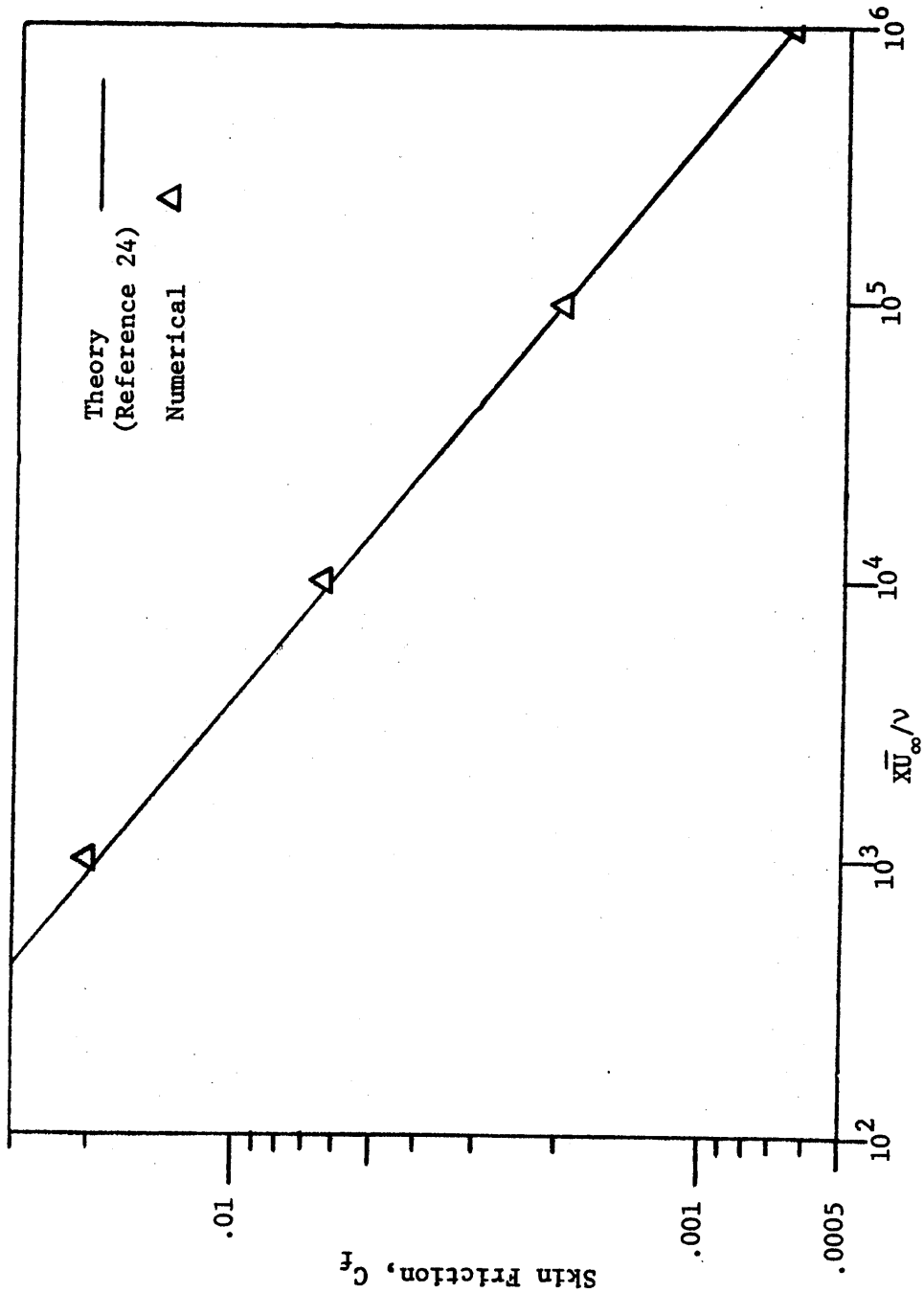

Figure 4.4. Skin Friction Compared with Blasius Solution.

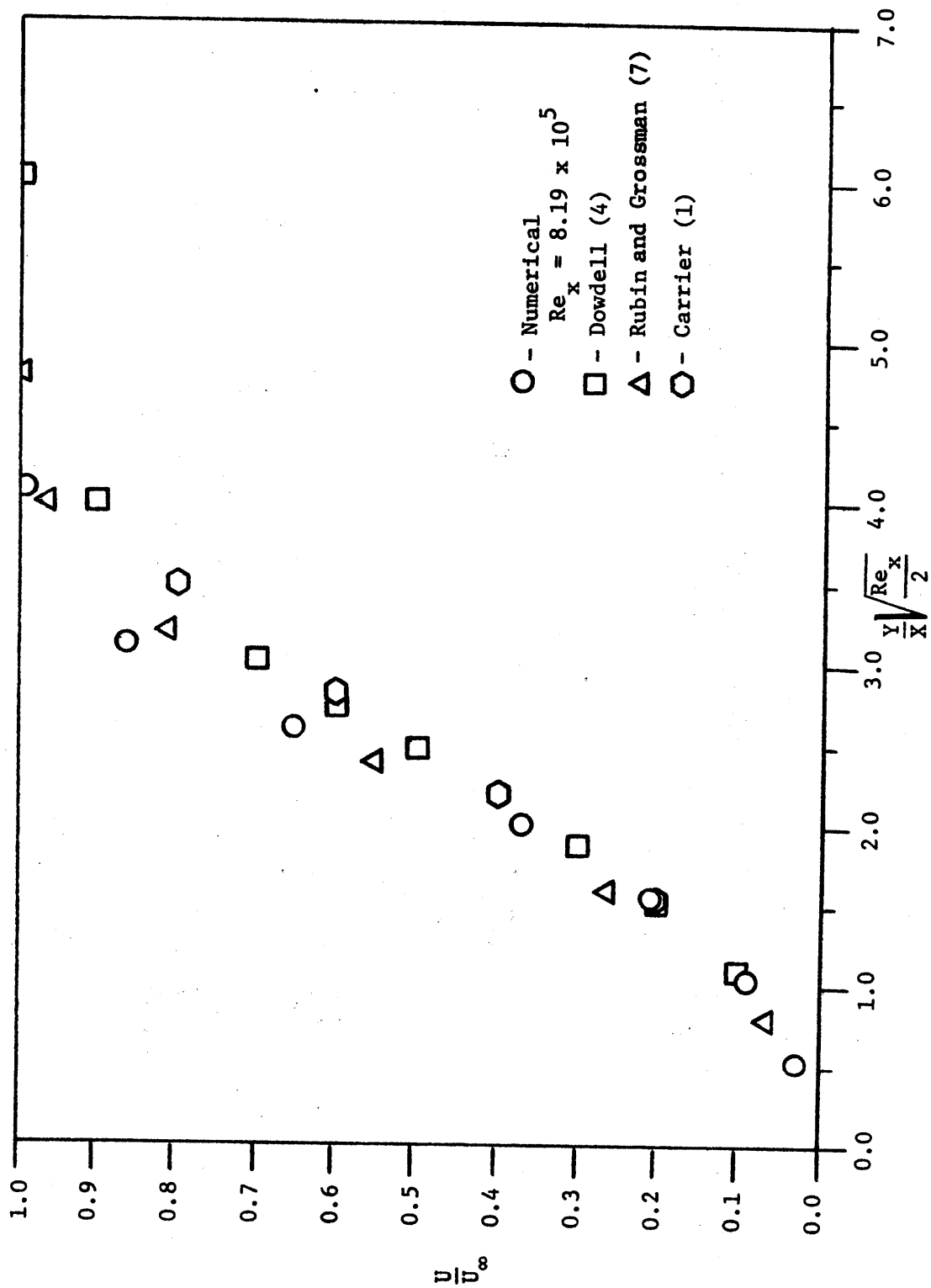

Figure 4.5. U-Velocity Profile on Bisector Compared with Dowdell, Carrier and Rubin and Grossman Solutions.

Figure 4.6 demonstrates the change with Reynolds number of the u-velocity profile along the bisector. This variation was most probably caused by the initial conditions applied at the leading edge for the velocity components. The numerical method predicted a single curve for the u-velocity profile after the Reynolds number was greater than 100,000. A similar trend was observed for the two-dimensional case but the u-velocity profile reached its limiting value at a smaller Reynolds number. None of the previous investigators reported whether or not any change in the u-velocity profile on the bisector occurred with Reynolds number.

Because of the assumption that the pressure gradient was zero and the final form of the finite-difference equations, four equations involving only three unknowns resulted. Therefore, the continuity equation was used to independently verify that the velocity components predicted by the solution of the three momentum equations were reasonable. Figure 4.7 illustrates that the continuity equation residual approaches zero as the calculations advanced in the x-direction. The high value for the continuity equation residual at the leading edge was expected.

Figure 4.8 is provided to illustrate the value of the skin friction from the corner to the two-dimensional boundary layer. A monotonic increase in skin friction was predicted.

Table 4.1 presents typical results from the corner region solution to illustrate that the numerical method predicted a symmetric flow field. The values of u at the same positions on

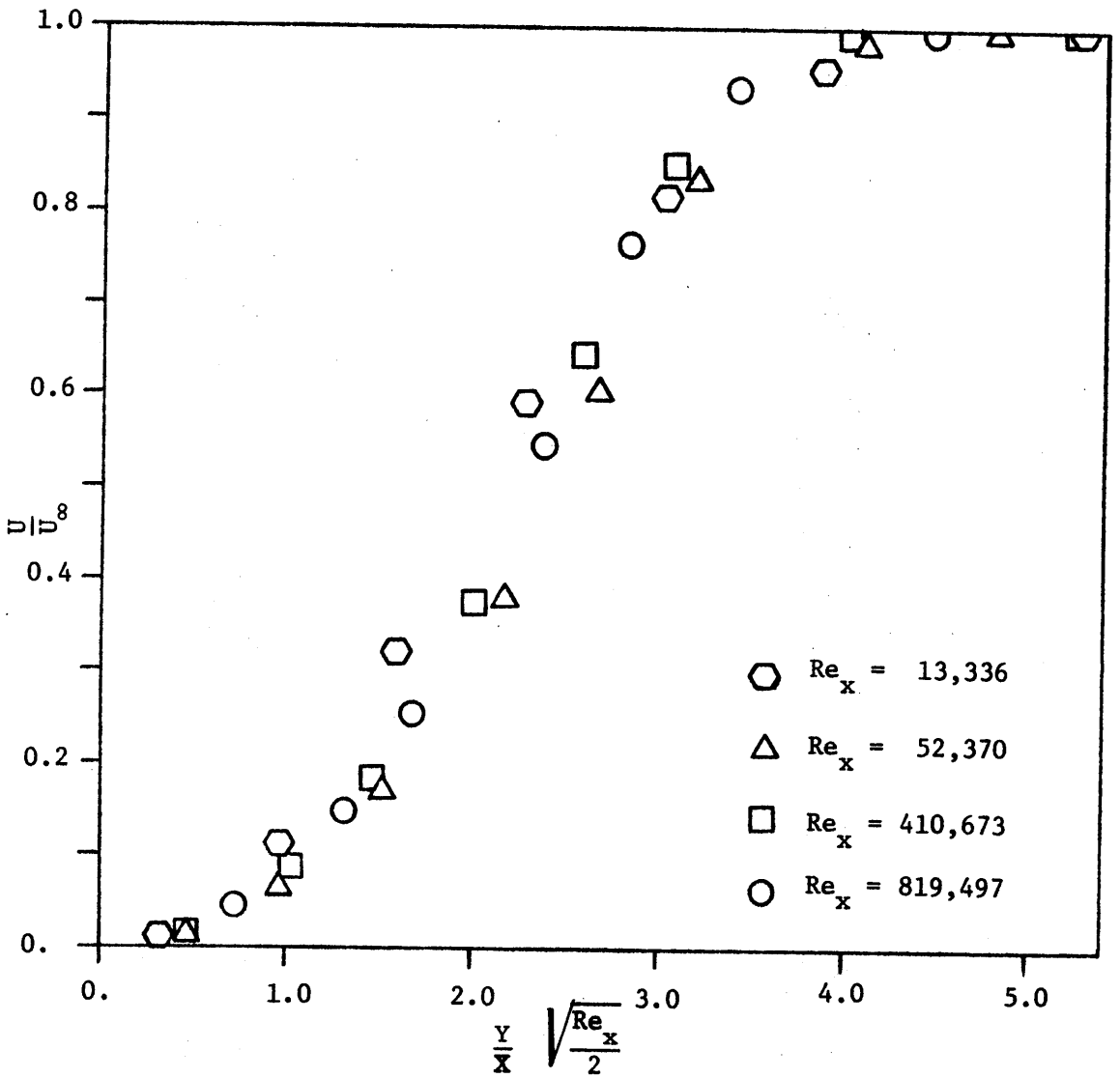

Figure 4.6. U-Velocity Profile on Bisector for Various Values of Reynolds Number.

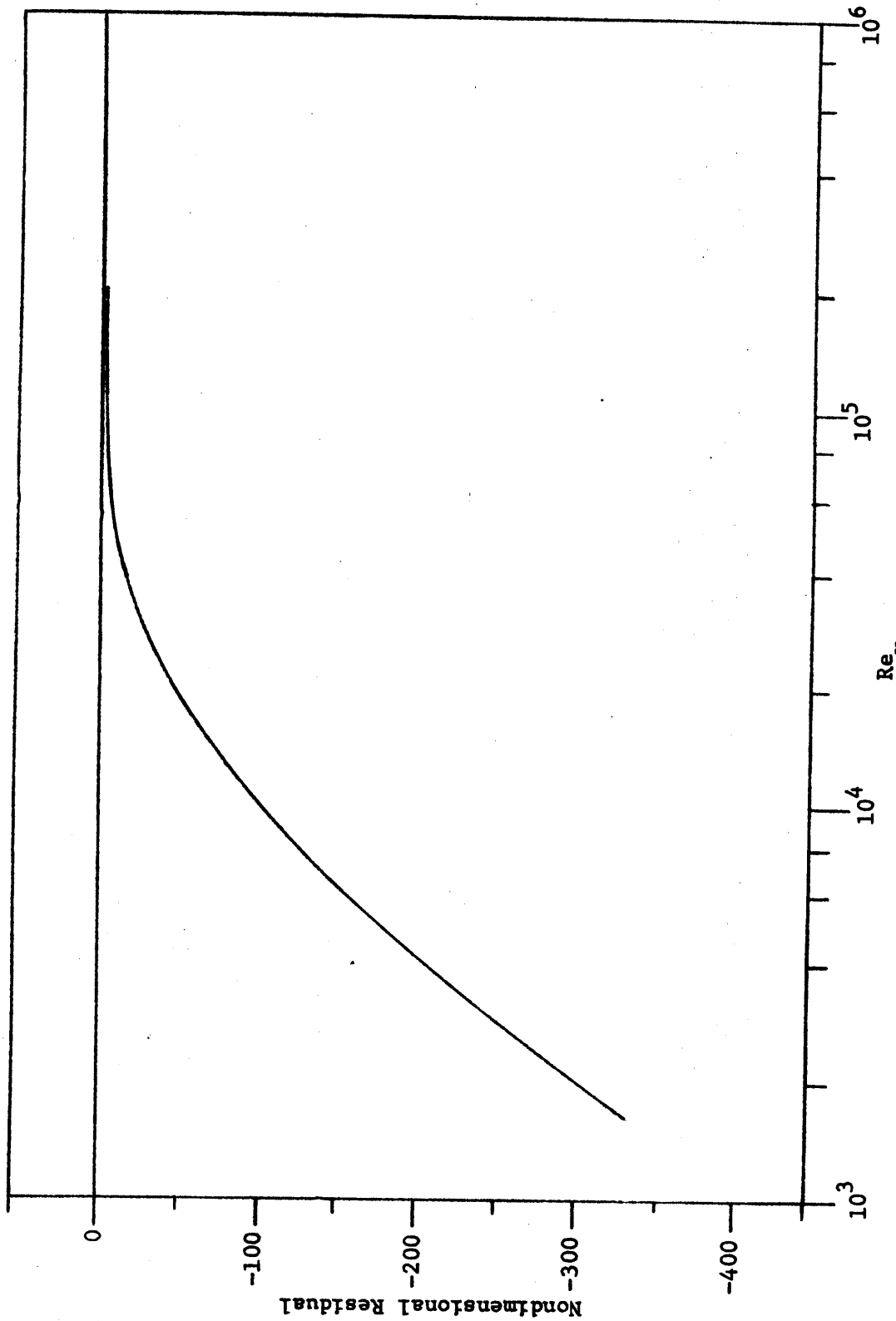

Figure 4.7. Average Continuity Equation Residual for Various Values of Reynolds Number, Laminar Flow.

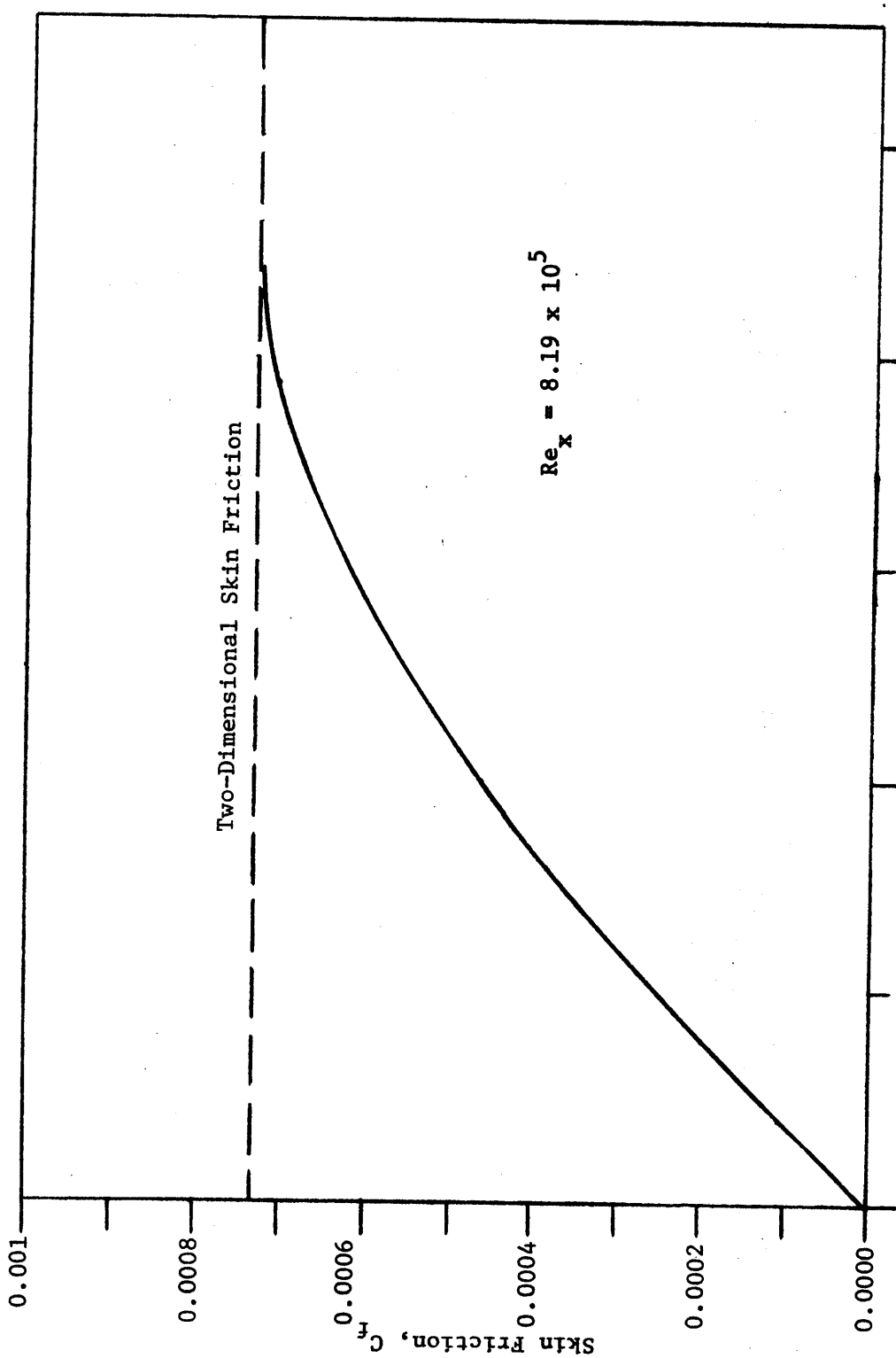

Figure 4.8. Skin Friction Along Y = 0 Wall in Corner Region.

Table 4.1. u, v, and w on and about the Bisector, Laminar Flow.

$\frac{Y}{X}\sqrt{Re_x/2}$	Re_x = 819000					
.6744	.02390 .08765 .09104	.02689 .09898 .1021	.03003 .1110 .1135	.03334 .1239 .1254	.03682 .1377 .1377	u v w
.6113	.02165 .07988 .08201	.02435 .09019 .09191	.02720 .1012 .1022	.03019 .1129 .1129	.03334 .1254 .1239	u v w
.5512	.01951 .07236 .07354	.02194 .08169 .08241	.02450 .09162 .09163	.02720 .1020 .1012	.03003 .1135 .1111	u v w
.494	.01748 .06509 .06559	.01965 .07348 .07349	.02194 .08240 .08170	.02435 .09190 .09021	.02689 .1020 .09899	u v w
.4395	.01554 .05810 .05811	.01748 .06557 .06511	.01951 .07352 .07237	.02165 .08199 .07990	.02390 .09103 .08766	u v w
	.4395	.494	.5512	.6113	.6744	$\frac{Z}{X}$ $\sqrt{Re_x/2}$

opposite sides of the bisector were within 0.1 percent of each other. The value of v at a given position on one side of the bisector was within 0.1 percent of the value of w at the symmetrical position on the other side of the bisector. The same result was observed for similar comparisons between w and v. On the bisector, v and w were within 0.1 percent of the same magnitude.

B. Turbulent Flow

Because turbulent flows cannot be described completely by mathematical statements which can be solved analytically, solutions for even the simplest flow geometries do not exist. Therefore, the reliability of the numerical method to predict two-dimensional turbulent boundary-layer development was ascertained by comparisons with empirical correlations, experimental data, and a simple momentum integral solution.

The prediction by the numerical method of the u-velocity profile development in the two-dimensional region was compared to Coles' (26) wall-wake model. This model was considered the best of the empirical correlations for turbulent flows. Figure 4.9 illustrates that excellent agreement was obtained since the u-velocity profile developed as expected. The development of the wake function was also predicted but was not as great as that given by Coles (26).

The method was further checked by comparing the predicted δ^* and θ development against a simple momentum integral technique presented by Schlichting (14) which assumed a one-seventh power law velocity

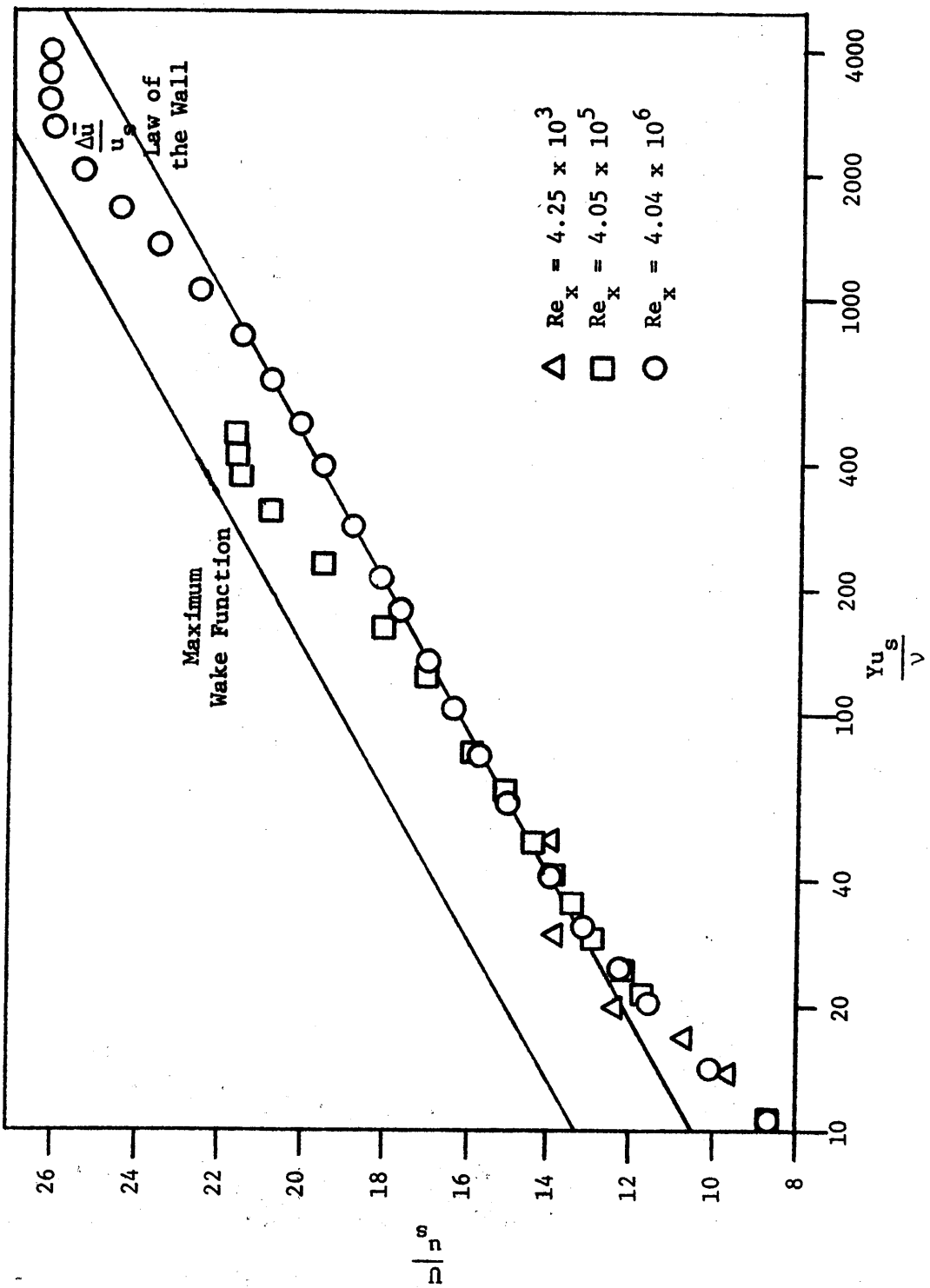

Figure 4.9. Streamwise Turbulent Velocity Profile Development Compared with the Law of the Wall.

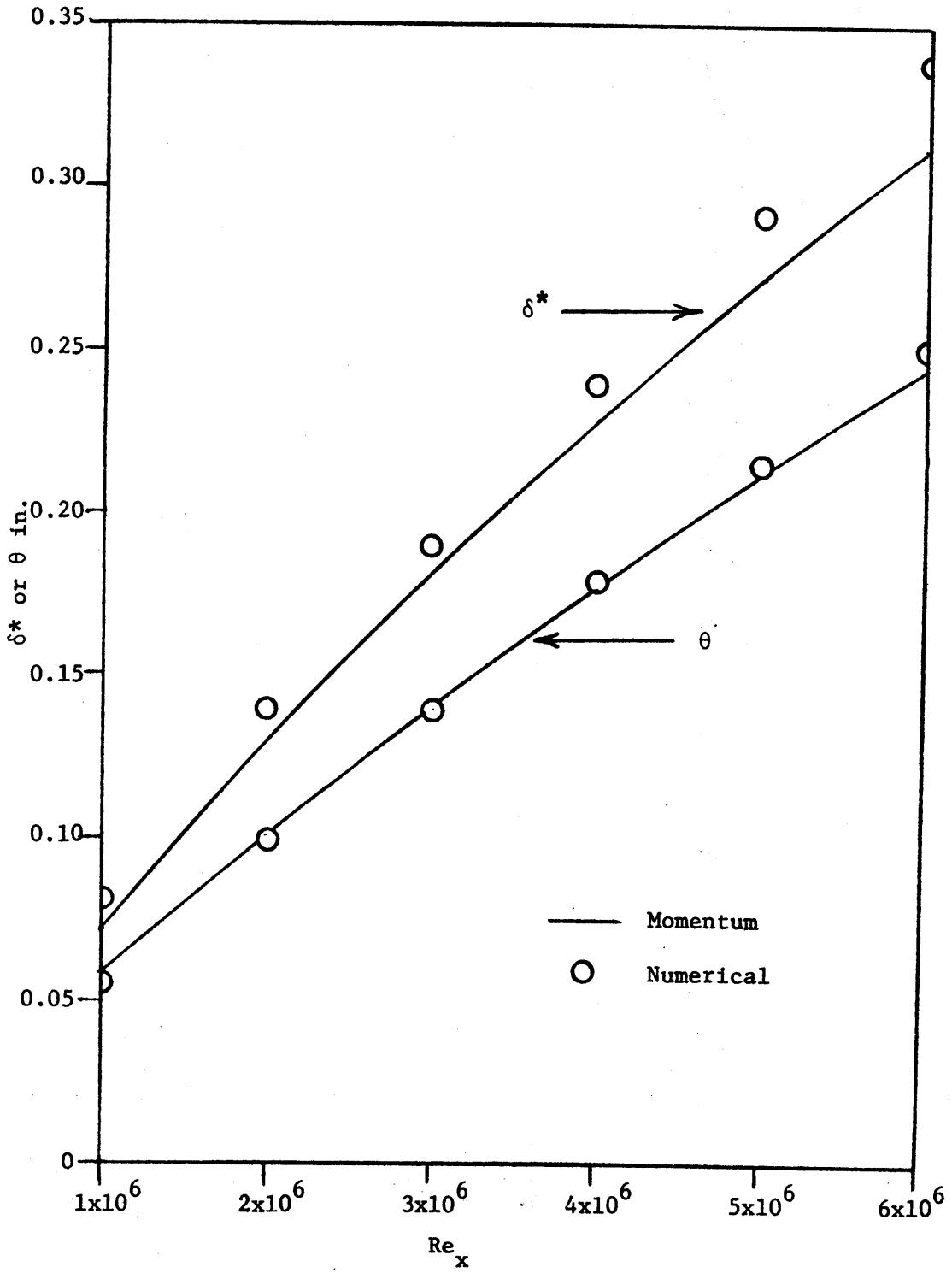

Figure 4.10. δ* and θ Compared with a Momentum Integral Analysis.

profile in conjunction with the Blasius law of friction. Figure 4.10 illustrates the method's capability of predicting δ^* and θ compared with the momentum integral technique. δ^* and θ were obtained by integrating the predicted u-velocity profile by the trapezoidal rule in accordance with equations 4.1 and 4.2.

The wall shear stress estimated by equation 4.4 is only valid for the turbulent flow conditions if the first nodal position in the normal direction to the wall is located within the laminar sublayer. The resultant skin friction obtained from equation 4.3 is compared in Fig. 4.11 with the Ludwieg and Tillmann (27) empirical shear stress correlation. The local value of θ and H_s necessary in the correlation were obtained from the momentum integral method. Although the agreement is poor, the general slope and shape of the skin friction curve is predicted by the present method.

Figure 4.12 demonstrates the ability of the numerical method to predict the two-dimensional turbulent boundary layer u-velocity profile as reported by Bragg (10). Very favorable results are indicated by the comparison.

Figures 4.13, 4.14, 4.15 and 4.16 demonstrate that the method was capable of predicting the u-velocity profile for turbulent flow along the bisector of a corner. The method tends to underpredict the experimental results as the corner is approached, which is most probably due to the eddy-viscosity model used with this method.

The method predicted symmetry about the bisector reasonably well but the symmetry was not as good as in the laminar case. Table 4.2

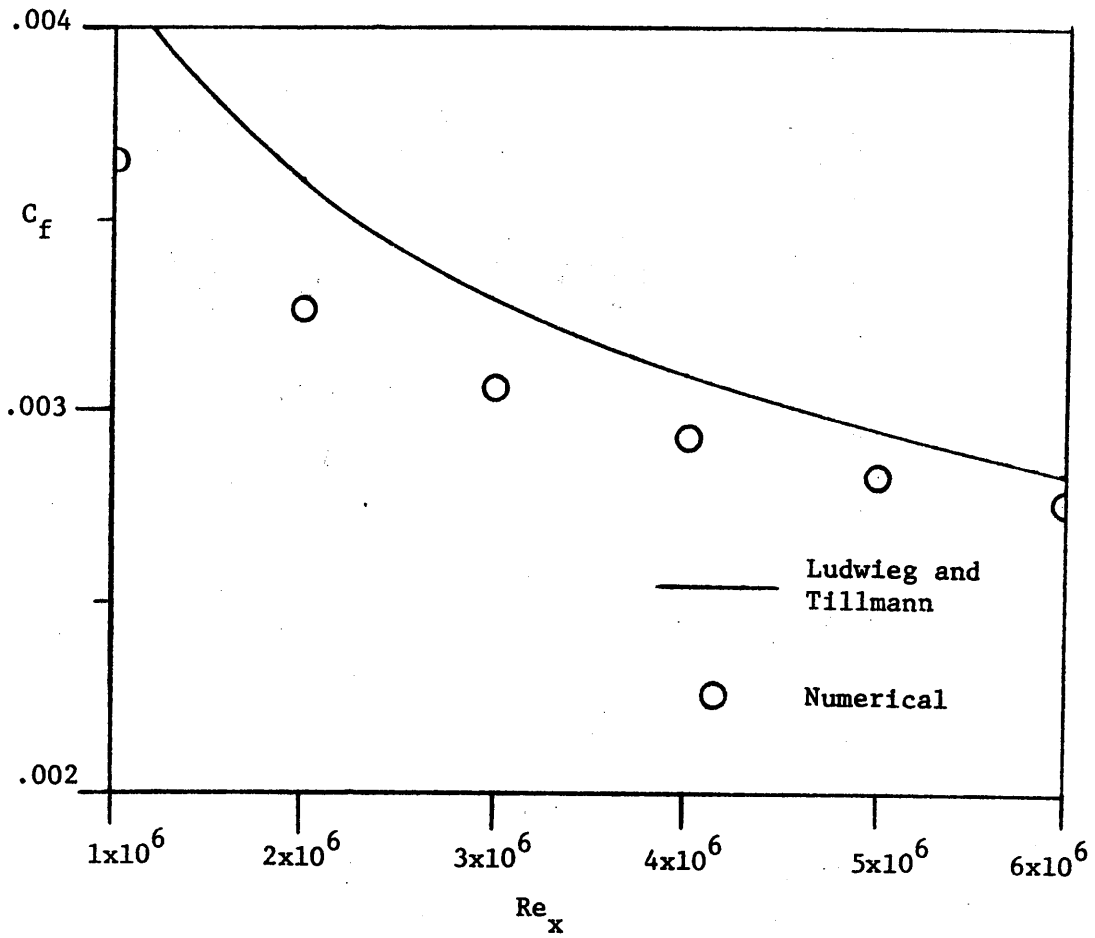

Figure 4.11. Skin Friction Compared with Ludwieg and Tillmann Correlation.

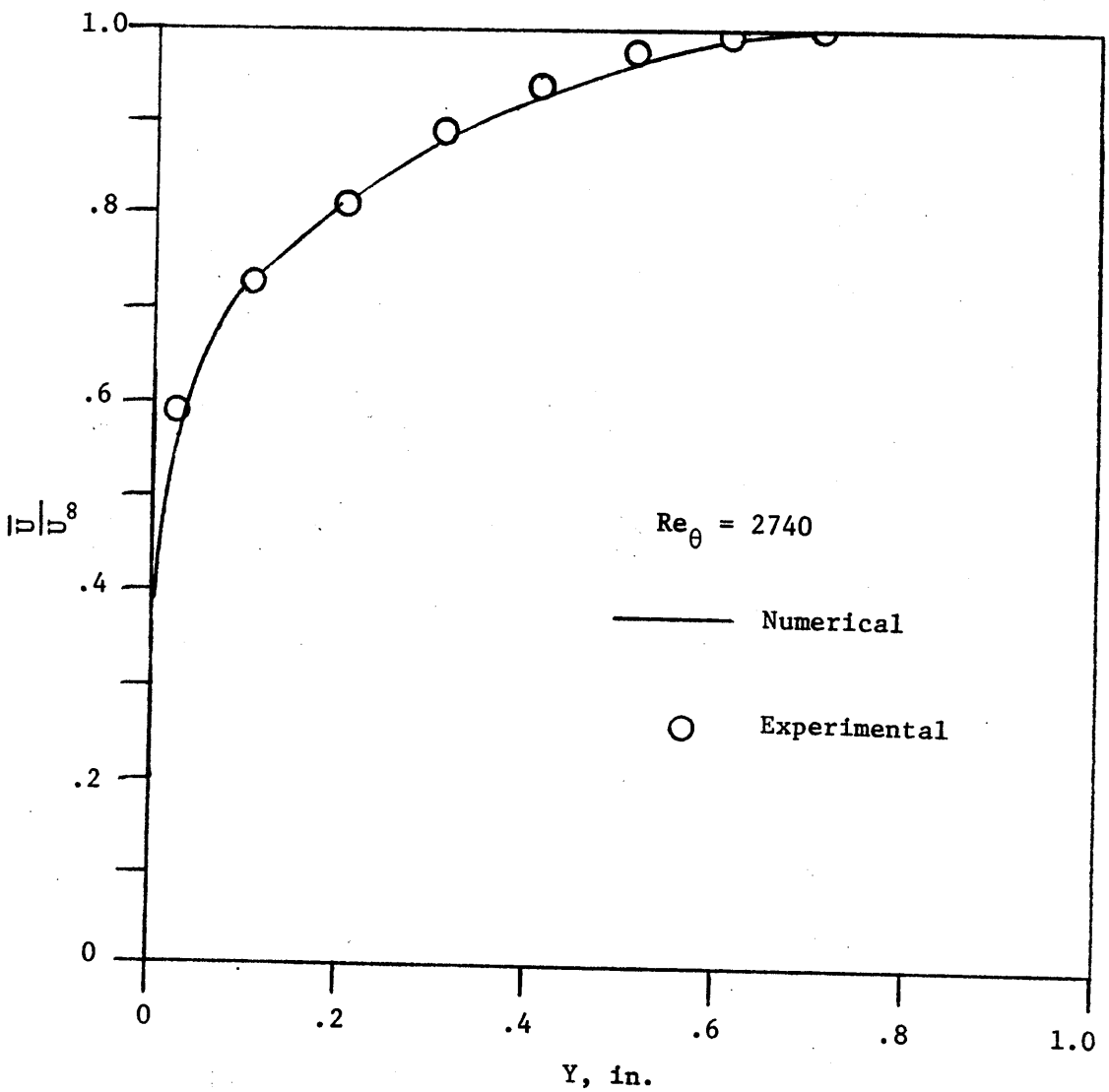

Figure 4.12. U-Velocity Profile Development Compared with Bragg's Data in Two-Dimensional Region.

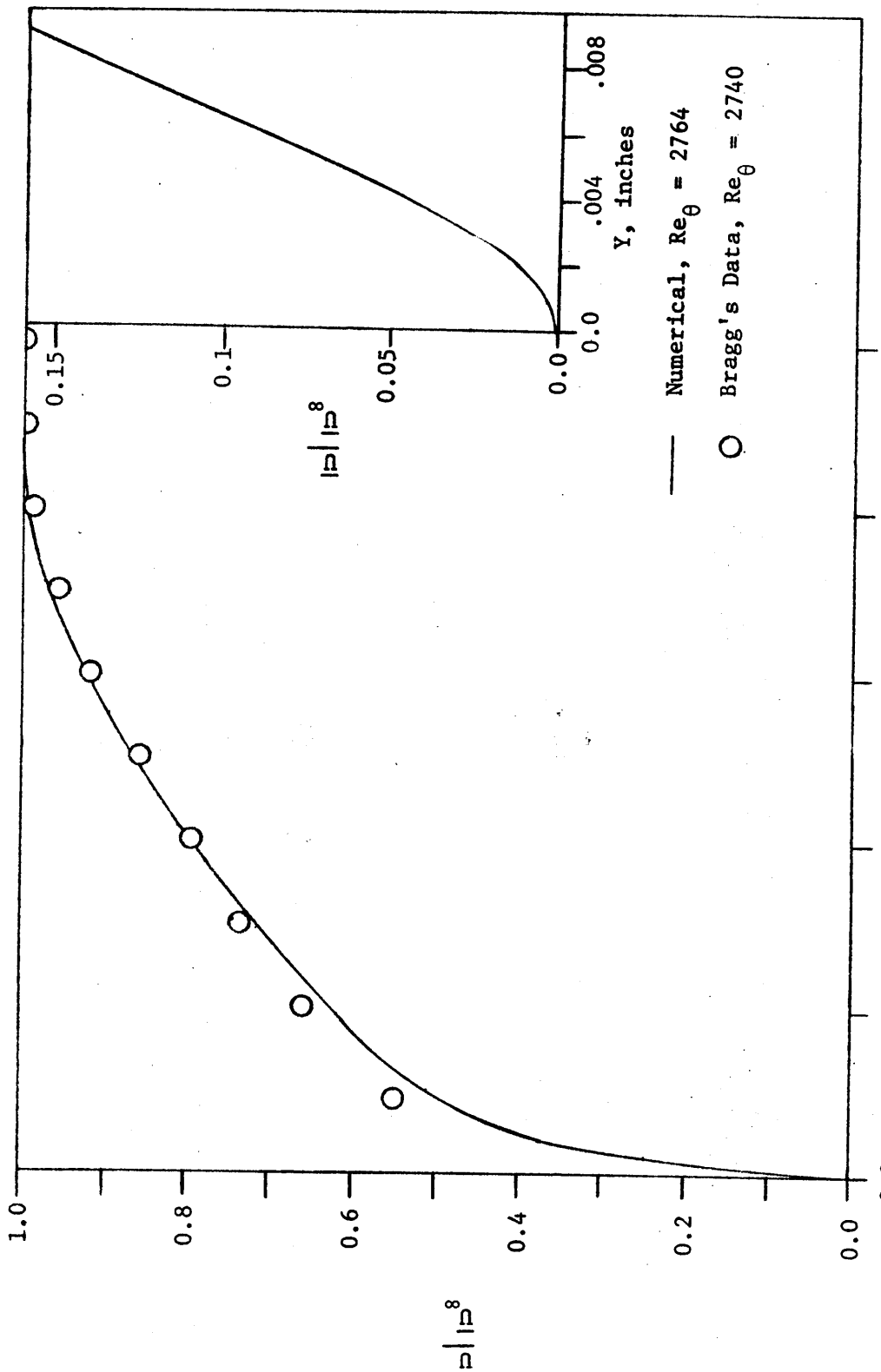

Figure 4.13. $\bar{U}$-Velocity Profile on Bisector Compared with Bragg's Data.

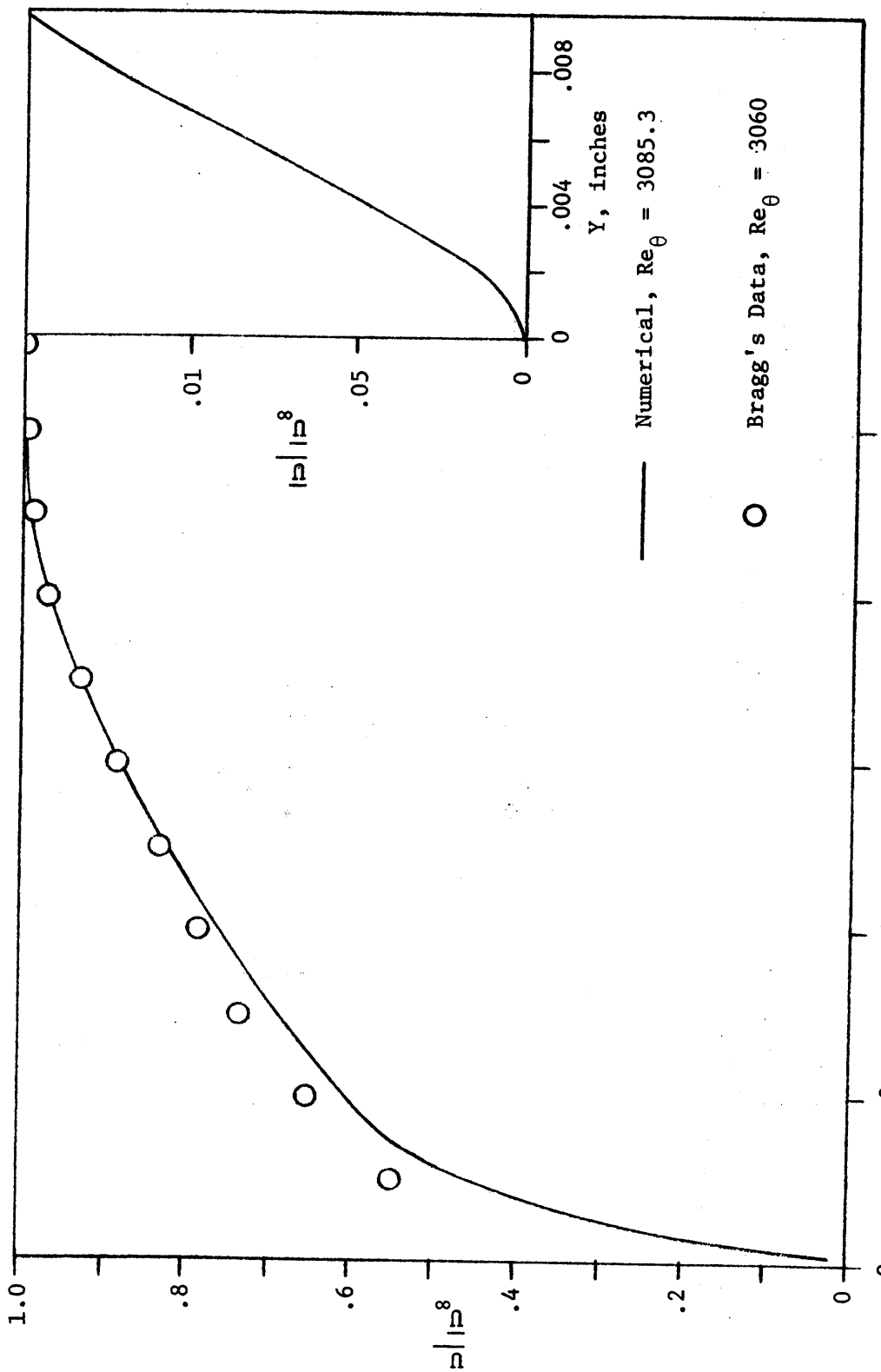

Figure 4.14. $\bar{U}$-Velocity Profile on Bisector Compared with Bragg's Data.

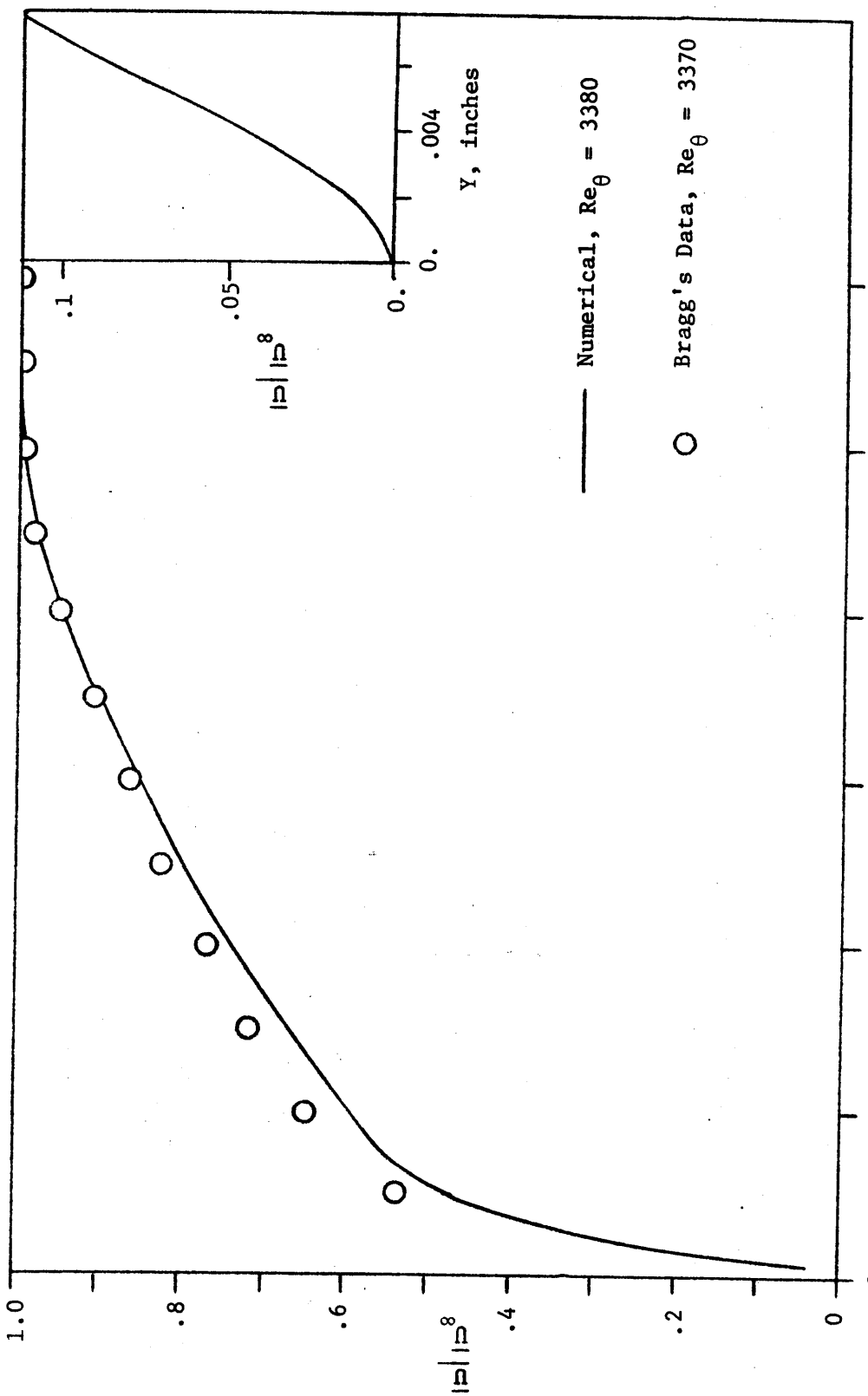

Figure 4.15. $\bar{U}$-Velocity Profile on Bisector Compared with Bragg's Data.

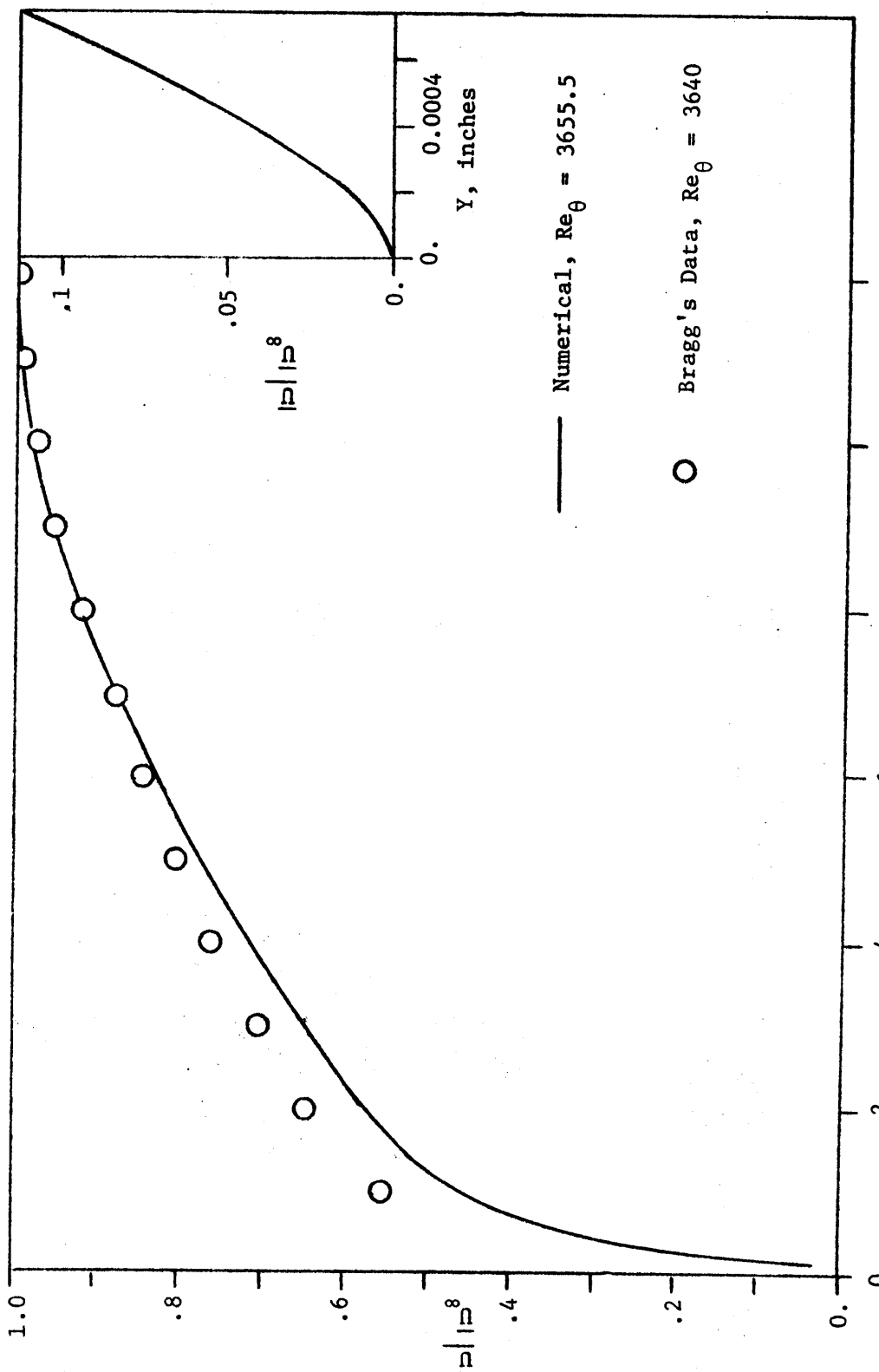

Figure 4.16. $\bar{U}$-Velocity profile on Bisector Compared with Bragg's Data.

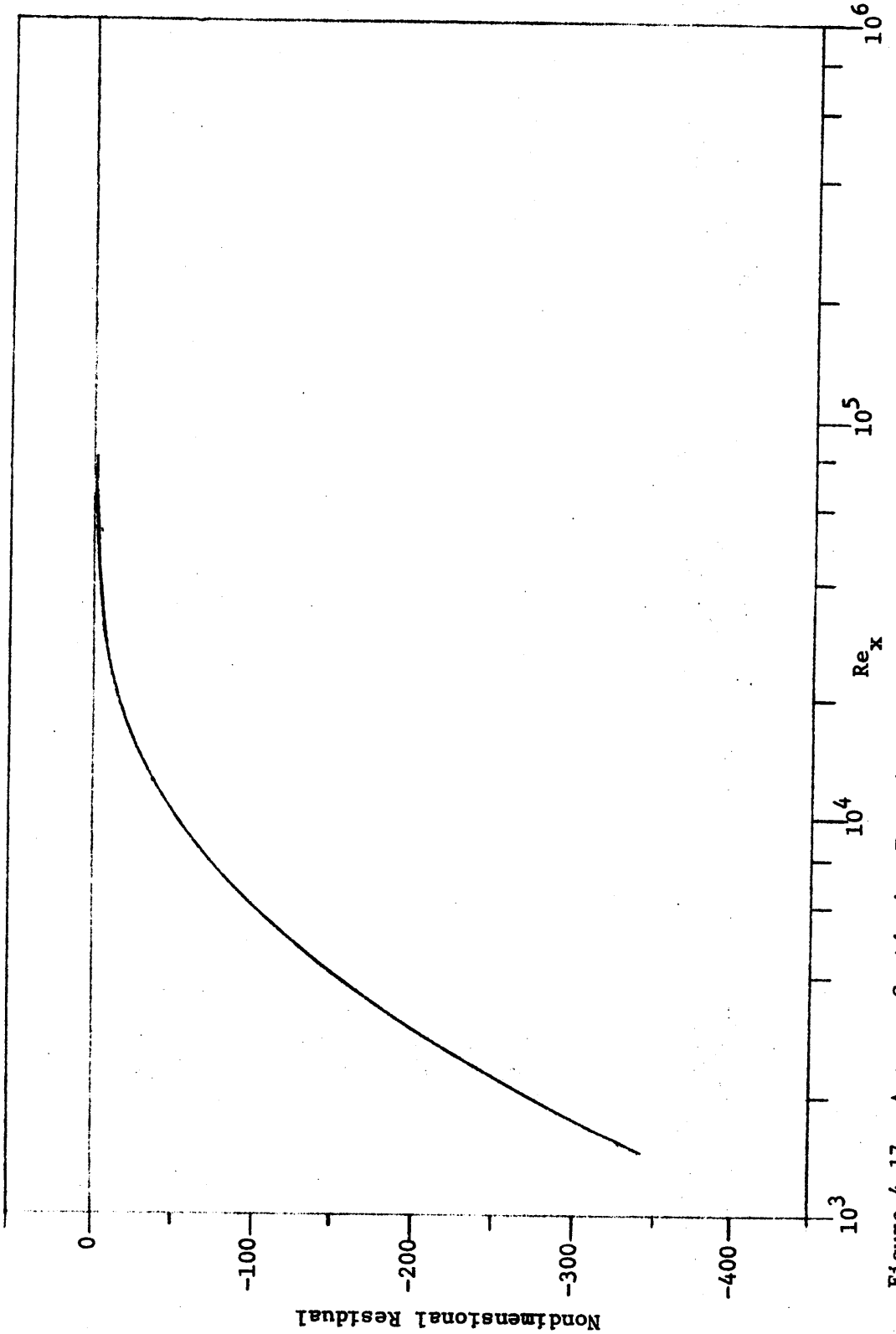

Figure 4.17. Average Continuity Equation Residual for various Values of Reynolds Number, Turbulent Flow.

Table 4.2. u, v, and w velocity component on and near the bisector, turbulent flow.

Y (inches)	$Re_x = 1.83 \times 10^6$					
.01992	.2237 .5130 .5145	.2354 .5399 .5415	.2451 .5618 .5635	.2516 .5768 .5785	.2562 .5874 .5891	u v w
.01821	.2097 .4808 .4822	.2195 .5031 .5046	.2289 .5245 .5260	.2375 .5447 .5463	.2447 .5613 .5630	u v w
.01658	.1938 .4441 .4454	.2035 .4663 .4676	.2160 .4953 .4967	.2267 .5203 .5219	.2354 .5401 .5417	u v w
.01503	.1899 .4353 .4366	.1968 .4514 .4528	.2063 .4736 .4750	.2162 .4962 .4976	.2261 .5186 .5202	u v w
.01355	.1908 .4378 .4391	.1965 .4512 .4526	.2032 .4663 .4677	.2116 .4854 .4868	.2212 .5073 .5088	u w
	.01355	.01503	.01658	.01821	.01992	Z(inches)

is included to give a representative result for the u, v, and w components on and near the bisector.

Figure 4.17 presents the continuity equation residual as a function of the Reynolds number based on distance from the leading edge. As before in the laminar case, the residual tended to zero as the solution proceeded in the x-direction. This again was taken as an indication that the predictions of the velocity components in the corner region were reasonable.

The method did not predict the negative secondary velocities on the bisector as shown by published experimental work. However, a small amount of u-isovel distortion was predicted in the region of the bisector, but the magnitude was not as pronounced as that indicated by experiments. Neither was a closed vorticity pattern predicted.

The failure of the numerical method to predict the secondary flow pattern is probably due to the simple eddy-viscosity model used for the corner region.

V. CONCLUSIONS AND RECOMMENDATIONS

A technique is presented to solve the equations of motion for both the laminar and turbulent flow of an incompressible fluid along the axis of a ninety-degree corner. The Alternating Direction Implicit (ADI) finite-difference method was used to cast the governing partial differential equations into a finite-difference form.

An eddy-viscosity model expressed in terms of a mixing length was used to represent the turbulent shear stresses in the corner region. The mixing length model given by Pletcher (22) was adapted for use in the corner region.

The final form of the finite-difference equations was such that the corresponding coefficient matrixes were always tridiagonal. Therefore, the Thomas algorithm (16) applied.

This method was used to solve the laminar and turbulent flow along the axis of a ninety-degree corner from which the following observations and conclusions were obtained:

1. The ADI finite-difference scheme resulted in a stable and convergent method for the corner region.
2. For the laminar corner flow case, good agreement with the solutions of Carrier (1), Dowdell (4), and Rubin and Grossman (7) was obtained.
3. For the turbulent corner flow case, good agreement with the results of Bragg's (10) experiments was obtained.

4. The computational time required to solve the Bragg (10) geometry was 69 minutes on an IBM 370, model 165.
5. This method should be extended so that adverse pressure gradient flows can be studied. The procedure is sufficiently general that these studies can be conducted without a major alteration.
6. Additional experiments should be conducted to ascertain the true nature of the turbulent shear stresses in the corner region.

VI. BIBLIOGRAPHY

1. CARRIER, G. F., "The Boundary Layer in a Corner," Quarterly of Applied Mathematics, Vol. 4, No. 4, pp. 367-370, 1947.

2. KEMP, N. H., "The Laminar Three-Dimensional Boundary Layer and a Study of Flow Past a Side Edge," Master's Thesis, Cornell University, June, 1951.

3. DOWDELL, R. B., "Corner Boundary Layer," Sc.M. Thesis, Brown University, May, 1952.

4. DOWDELL, R. B., "Incompressible Boundary Layer Flow Along Interior Corners," Proceedings of 10th Midwestern Mechanics Conference, Developments in Mechanics, Vol. 4, pp. 1355-1378, August, 1967.

5. RUBIN, S. G., "Incompressible Flow Along a Corner," Journal of Fluid Mechanics, Vol. 26, No. 1, pp. 97-110, 1966.

6. PAL, A. and S. G. RUBIN, "Asymptotic Features of Viscous Flow Along a Corner," Quarterly of Applied Mathematics, Vol. 24, No. 1, pp. 91-108, April, 1971.

7. RUBIN, S. G. and B. GROSSMAN, "Viscous Flow Along a Corner: Numerical Solution of the Corner Layer Equations," Quarterly of Applied Mathematics, Vol. 24, No. 2, pp. 169-186, July, 1971.

8. NIKURADSE, J., "Untersuchungen uber die Geschwindigkeitsverteilung in Turbulenten Stromungen," VDI-Forschungsheft 281, Berlin, 1926.

9. GESSNER, F. B., "Turbulence and Mean Flow Characteristics of Fully-Developed Flow in Rectangular Channels," Ph.D. Dissertation, Purdue University, January, 1964.

10. BRAGG, G. M., "The Turbulent Boundary Layer in a Corner," Journal of Fluid Mechanics, Vol. 36, Part 3, pp. 485-503, 1969.

11. TOAN, N. K., "Applications of Integral Methods to the Turbulent Corner Flow Problems," ASME Paper No. 71-WA/FE-36, Presented at Winter Annual Meeting, November 28-December 2, 1971.

12. GERSTEN, K., "Corner Interference Effects," AGARD Report No. 299, March, 1959.

13. VEENHUIZEN, S. D. and R. N. MERONEY, "Secondary Flow in a Boundary Layer," Project THEMIS, T.R. No. 3, ONR Contract No. N00014-68-A-0493-0001, June, 1968.

14. SCHLICHTING, H., Boundary-Layer Theory, Sixth Edition, McGraw-Hill Book Company, p. 62, 1968.

15. PEACEMAN, D. W. and H. H. RACHFORD, JR.,"The Numerical Solution of Parabolic and Elliptic Differential Equations," Journal of the Society of Industrial Applied Mathematics, Vol. 3, No. 1, pp. 28-41, March, 1955.

16. VON ROSENBURG, D. U., Methods for the Numerical Solution of Partial Differential Equations, American Elsevier Publishing Company, Inc., New York, p. 113, 1969.

17. CRANK, J. and P. NICHOLSON, "A Practical Method for Numerical Evaluation of Solutions of Partial Differential Equations of the Heat Conduction Type," Proceedings of Cambridge Philosophical Society, Vol. 43, pp. 50-67, 1947.

18. KLINKSIEK, W. F., "An Implicit Numerical Solution of the Turbulent Three-Dimensional Incompressible Boundary-Layer Equations," Ph.D. Dissertation, Virginia Polytechnic Institute and State University, June, 1971.

19. HINZE, J. O., Turbulence, an Introduction to Its Mechanism and Theory, McGraw-Hill Book Company, Chapter 1, p. 20, 1959.

20. SCHLICHTING, H., Boundary-Layer Theory, Sixth Edition, Mc-Graw Hill Book Company, p. 548, 1968.

21. GOLDSTEIN, S., Modern Developments in Fluid Dynamics, Vol. 1, Dover Publications, Inc., pp. 206-208, 1965.

22. PLETCHER, R. H., "On a Finite-Difference Solution for the Constant-Property Turbulent Boundary-Layer," AIAA Journal, Vol. 7, No. 2, pp. 305-334, February, 1969.

23. EAST, J. L., "An Exact Numerical Solution of the Three Dimensional Incompressible Turbulent Boundary-Layer Equations," Ph.D. Dissertation, Virginia Polytechnic Institute and State University, 1970.

24. SCHLICHTING, H., Boundary-Layer Theory, Sixth Edition, McGraw-Hill Book Company, p. 129, 1968.

25. KELLY, L. G., Handbook of Numerical Methods and Applications, First Edition, Addison-Wesley Publishing Company, Inc., pp. 224-225, 1967.

26. COLES, D., "The Law of the Wake in the Turbulent Boundary-Layer," Journal of Fluid Mechanics, Vol. 1, Part 2, pp. 191-226, 1956.

27. LUDWIEG, H. and W. TILLMANN, "Investigations of the Wall Shearing Stress in Turbulent Boundary Layers," NACA TM 1285, 1950.

28. LAUNDER, B. E. and W. M. YING, "Fully-Developed Turbulent Flow in Ducts of Square Cross Section," Report No. TM/TN/A/11., Imperial College of Science and Technology, Department of Mechanical Engineering, July, 1971.

VII. APPENDIX A

Appendix A contains the development of the finite-difference equations which were used to approximate the equations 3.5a-d, 3.6a-b, 3.13a-c, and 3.14a-b. Also, the Thomas algorithm and the model used for the corner region mixing length are presented.

A. General Discussion of Finite-Differences

It is possible to represent the value of a function removed a distance from a position where the value of the function is known by a Taylor series.

$$g(x+\Delta x) = g(x) + \Delta x g'(x) + \frac{(\Delta x)^2}{2!} g''(x) + \frac{(\Delta x)^3}{3!} g'''(x) + \text{---} \quad \text{(A.1)}$$

Using the figure A.1 as a reference, the central difference representation of the second derivative at x can be approximated to a truncation error of $0\left[\frac{(\Delta x)^2}{12}\right]$ by writing $g(x+\Delta x)$ and $g(x-\Delta x)$ in two Taylor series and then adding these two series together. This equation is as follows:

$$g''(x) = \frac{g(x+\Delta x) - 2g(x) + g(x+\Delta x)}{(\Delta x)^2} + 0\left[\frac{(\Delta x)^2}{12}\right] + \text{---} \quad \text{(A.2)}$$

The equation A.2 assumes that the function is continuous throughout the region of applicability, is many times differentiable and the neglected terms are many times less than the first term retained in the resultant series.

Again using two Taylor series for $g(x+\Delta x)$ and $g(x-\Delta x)$ and substracting these series from one another, the central difference representation for the first derivative can be obtained and will have a truncation error of $0\left[\frac{(\Delta x)^2}{6}\right]$. The result is as follows:

$$g'(x) = \frac{g(x+\Delta x) - g(x-\Delta x)}{\Delta x} + 0\left[\frac{(\Delta x)^2}{6}\right] + \text{---} \quad \text{(A.3)}$$

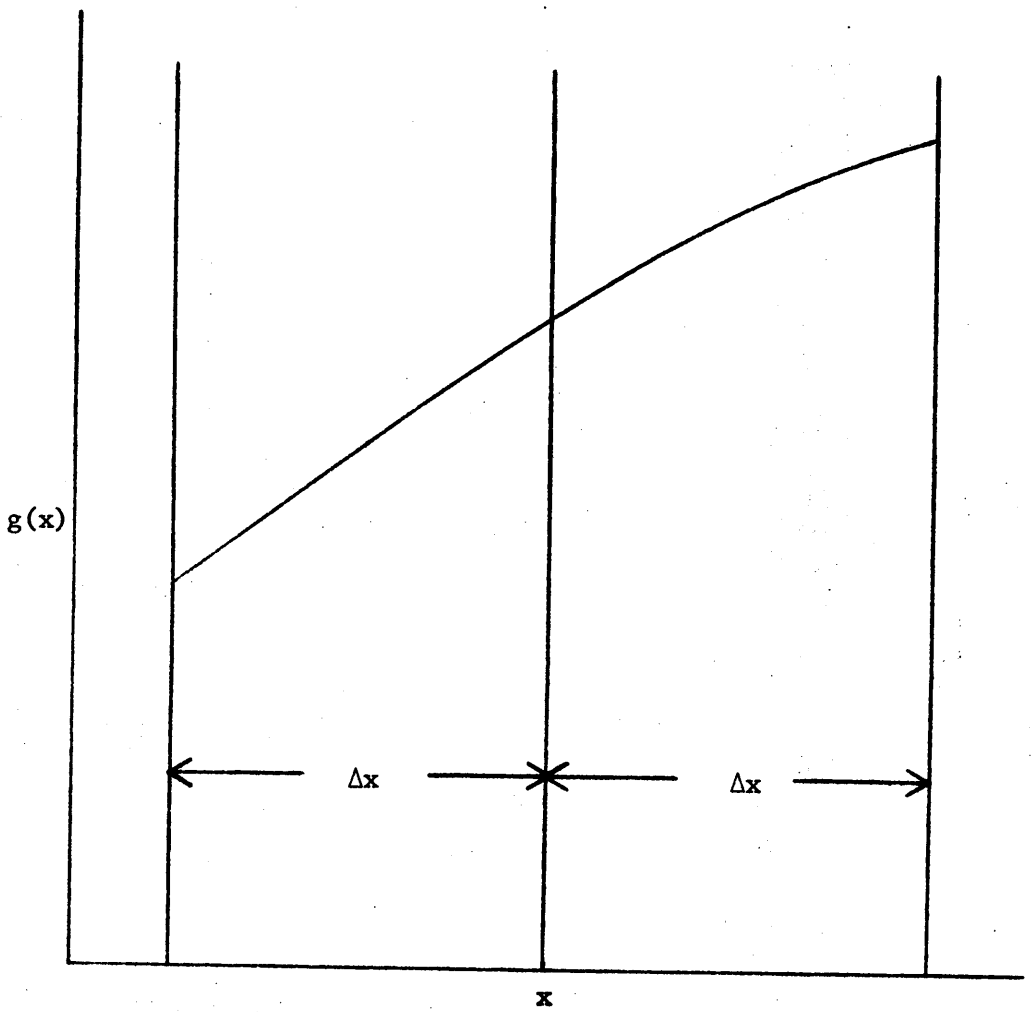

Fig. A.1. Grid Used for Representation of g'(x) and g"(x).

The same assumptions applied to equation A.2 must also be applied to equation A.3 in order to ensure a valid approximation of the first derivative.

If variable grid spacing between nodes is desirable, the following equations are representative of u_y, u_{yy}, u_z, and u_{zz} when applied to figure A.2:

$$u_y(e) = \frac{\frac{\Delta y_-}{\Delta y_+}(u(d) - u(e)) + \frac{\Delta y_+}{\Delta y_-}(u(e) - u(b))}{(\Delta y_+ + \Delta y_-)} + \frac{(\Delta y_+)^2}{6} u(e)_{yyy} \tag{A.4a}$$

$$u_{yy}(e) = \frac{\frac{1}{\Delta y_+}(u(d) - u(e)) - \frac{1}{\Delta y_-}(u(e) - u(b))}{(\Delta y_+ + \Delta y_-)/2} + \frac{\Delta y_+}{3} u(e)_{yyy} + \frac{(\Delta y_+)^2}{12} u(e)_{yyyy} \tag{A.4b}$$

$$u_z(e) = \frac{\frac{\Delta z_-}{\Delta z_+}(u(c) - u(e)) + \frac{\Delta z_+}{\Delta z_-}(u(e) - u(a))}{(\Delta z_+ + \Delta z_-)} + \frac{(\Delta z_+)^2}{6} u(e)_{zzz} \tag{A.4c}$$

$$u_{zz}(e) = \frac{\frac{1}{\Delta z_+}(u(c) - u(e)) + \frac{1}{\Delta z_-}(u(e) - u(a))}{(\Delta z_+ + \Delta z_-)/2} + \frac{\Delta z_+}{3} u(e)_{zzz} + \left(\frac{\Delta z_+}{12}\right)^2 u(e)_{zzzz} \tag{A.4d}$$

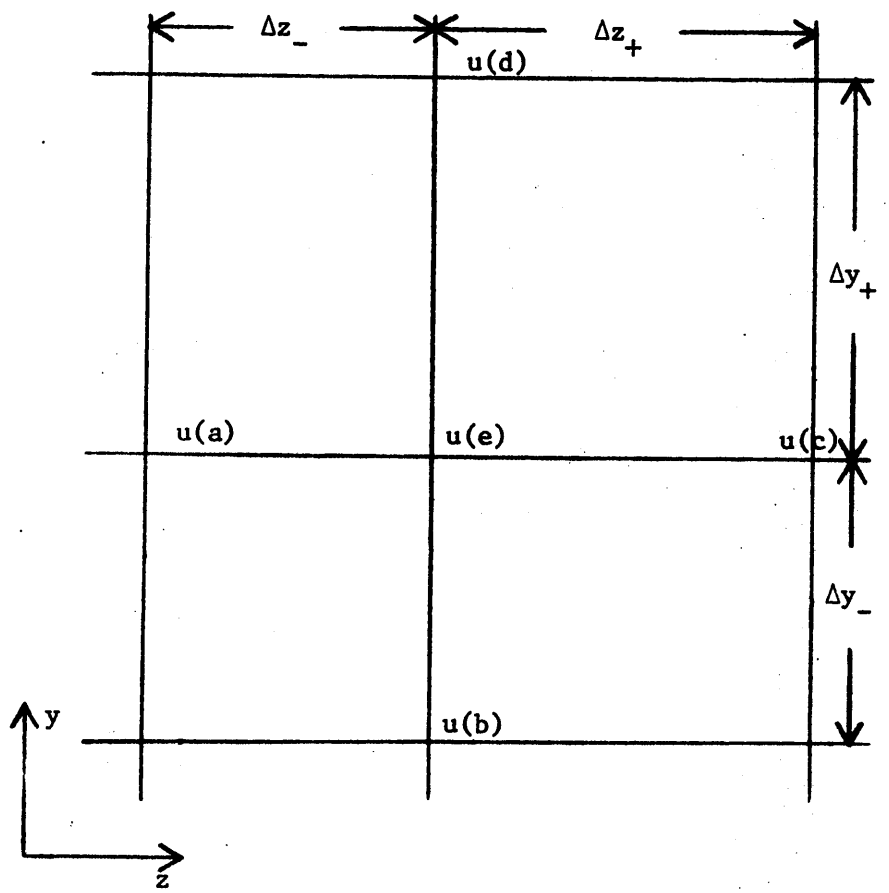

Fig. A.2. Variable Grid Spacing used for u_y, u_{yy}, u_z, and u_{zz}.

Thus, it is obvious that in order to keep the truncation error small, equal grid spacing should be used since both of the finite-difference equations A.2 and A.3 are of approximately the same order of truncation error. However, in order to conserve memory storage and to keep computer calculation times to a minimum, the variable grid size was used in this investigation.

Examination of equations A.4a-d indicates that the first and second partial derivatives of a function can be determined if the function is known at various points. By replacing the derivatives in partial differential equations with their finite-difference approximations, a set of equations can be generated where the only unknowns are the values of the function itself at various grid locations. And, if the boundary conditions imposed on the region are known or specified in which the partial difference equation is assumed to be valid, the set of equations can be solved for the value of the function at the various grid locations within the region.

B. Finite-Difference Approximations for the Two-Dimensional Laminar Boundary Layer

The Crank-Nicolson (17) implicit finite-difference method was used to represent the x-momentum equation, 3.6a, in the two-dimensional flat plate region. Using the dummy variable H, the various derivatives in equation 3.6a were written about the point i,m+1,n+1/2. The position specified by the index m+1 is the line 4-5 as shown in Fig. 3.4.

$$H_x = (H(i,m+1,n+1) - H(i,m+1,n))/\Delta x \tag{B.1a}$$

$$H_y = \frac{1}{2(\Delta y_+ + \Delta y_-)} \left\{ \frac{\Delta y_-}{\Delta y_+} [H(i+1,m+1,n) - H(i,m+1,n) + H(i+1,m+1,n+1) \right.$$

$$- H(i,m+1,n+1)] + \frac{\Delta y_+}{\Delta y_-} [H(i,m+1,n+1) - H(i-1,m+1,n+1)$$

$$\left. + H(i,m+1,n) - H(i-1,m+1,n)] \right\} \tag{B.1b}$$

$$H_{yy} = \frac{1}{(\Delta y_+ + \Delta y_-)} \left\{ \frac{1}{\Delta y_+} [H(i+1,m+1,n+1) + H(i+1,m+1,n) - H(i,m+1,n+1) \right.$$

$$- H(i,m+1,n)] - \frac{1}{\Delta y_-} [H(i,m+1,n+1) + H(i,m+1,n)$$

$$\left. - H(i-1,m+1,n+1) - H(i-1,m+1,n] \right\} \tag{B.1c}$$

The non-linear coefficients in equations 3.6a were assumed to be the space average of the previous nodal location and the new nodal location in the x-direction at each y node in order to linearize the equation. Thus, the non-linear coefficients were calculated as follows:

$$\bar{H} = \frac{H(i,m+1,n+1) + H(i,m+1,n)}{2} \tag{B.1d}$$

The above equation assumes that the value of H at the new nodal location in the x-direction was known but the entire problem was to determine what the value of H was at this nodal location. Therefore, as a first approximation, H(i,m+1,n+1) was assigned the value of

H(i,m+1,n) and an iterative procedure was used to determine successively better approximations to H(i,m+1,n+1).

In order to present the finite-difference equations in a more more concise form, the following definitions were introduced for the coefficients which make up the matrix.

$$r_1 = \frac{\bar{v}}{\bar{u}} \cdot \frac{\Delta x}{2} \cdot \frac{\Delta y_+}{\Delta y_-} \cdot \frac{1}{(\Delta y_+ + \Delta y_-)} \tag{B.2a}$$

$$r_2 = \frac{\bar{v}}{\bar{u}} \cdot \frac{\Delta x}{2} \cdot \frac{\Delta y_-}{\Delta y_+} \cdot \frac{1}{(\Delta y_+ + \Delta y_-)} \tag{B.2b}$$

$$s_1 = \frac{1}{\bar{u}} \cdot \frac{\Delta x}{\Delta y_-} \cdot \frac{1}{(\Delta y_+ + \Delta y_-)} \tag{B.2c}$$

$$s_2 = \frac{1}{\bar{u}} \cdot \frac{\Delta x}{\Delta y_+} \cdot \frac{1}{(\Delta y_+ + \Delta y_-)} \tag{B.2d}$$

Using the equations B.1a-d, the finite-difference approximation of equation 3.6a is as follows:

$$\begin{aligned}&(r_1 - s_1)u(i-1,m+1,n+1) + (1.-r_1-r_2+s_1+s_2)u(i,m+1,n+1)\\ &+ (r_2 - s_2)u(i+1,m+1,n+1) = (r_1+s_1)u(i-1,m+1,n)\\ &+ (1.-r_1+r_2-s_1-s_2)u(i,m+1,n) + (s_2-r_2)u(i+1,m+1,n)\end{aligned} \tag{B.3}$$

Klinksiek (18) has shown that the finite-difference approximation used for the continuity equation 3.6b was important from a stability consideration. The finite-difference form of the continuity equation recommended by Klinksiek (18) was written about the point (i-1/2,m+1,n+1/2).

$$\begin{aligned}v(i,m+1,n+1) = v(i-1,m+1,n+1) - \frac{\Delta y_-}{2\Delta x} \{&u(i-1,m+1,n+1)\\ &+ u(i,m+1,n+1) - u(i-1,m+1,n) - u(i,m+1,n)\}\end{aligned} \tag{B.4}$$

The equation B.4 contains a backward difference approximation for the u_x derivative and therefore has a truncation error of $0(\Delta x)$.

To further condense equation B.3, the following notation was employed:

$$A_1(i) = r_1 - s_1 \tag{B.5a}$$

$$B_1(i) = 1. + r_1 - r_2 + s_1 + s_2 \tag{B.5b}$$

$$C_1(i) = r_2 - s_2 \tag{B.5c}$$

$$D_1(i) = (r_2 + s_2)u(i-1,m+1,n) + (1.-r_1+r_2-s_1-s_2) \cdot u(i,m+1,n) + (s_2-r_2)u(i+1,m+1,n) \tag{B.5d}$$

Using the notation above, the following algebraic equation resulted and was solved in conjunction with the finite-difference representation of the continuity equation to yield the u and v-profiles for the two-dimensional boundary layer developing on the Y = 0 wall.

x-momentum

$$A_1(i)u(i-1,m+1,n+1) + B_1(i)u(i,m+1,n+1) + C_1(i)u(i+1,m+1,n+1) = D_1(i) \qquad i = 2,3..... \ell \tag{B.6}$$

Inspection of equation B.6 reveals that a tridiagonal matrix was generated. The solution technique used to solve this type of matrix is discussed in the last portion of this Appendix.

C. Finite-Difference Approximations for the Laminar Corner Region

The Alternating Direction Implicit (ADI) finite-difference method was used to solve the x, y, and z momentum equations 3.5a-c in the

corner region. In accordance with the ADI method, the derivatives in one of the transverse directions were written about the point i,j,n+1/4 and the derivatives of the second transverse direction about the point i,j,n. Then, the derivatives in the second transverse direction were written about the point i,j,n+3/4 and the derivatives of the first transverse direction about the point i,j,n+1/2 using the half-step values. Because of the method used to write the derivatives with the ADI method, half-step values of the desired variable are generated at the point i,j,n+1/2. The values generated at this point, however, are not the true representation of the variable at this point since to arrive at this point, the calculations are executed explicitly in one of the coordinate directions. These half-step values are only used for calculations in the next half step for the desired value of the variable at the end of the full step. Explicit stepping continuously in one of the coordinate directions can cause instabilities to develop which are circumvented in the ADI method by stepping implicitly in the next half step in the coordinate direction which was done explicitly in the previous half step. For a detailed study and analysis of the ADI method, consult Peaceman and Rachford (15).

Using the dummy variable H, the various terms of equations 3.5a-c were written in the following finite-difference form for the first half-step in the x-direction using the equations A.4a-d as models.

$$H_x = (H^*(i,j) - H(i,j,n))/(\Delta x/2) \tag{C.1a}$$

$$H_y = \frac{\frac{\Delta y_-}{\Delta y_+}(H^*(i+1,j) - H^*(i,j)) + \frac{\Delta y_+}{\Delta y_-}(H^*(i,j) - H^*(i-1,j))}{\Delta y_+ - \Delta y_-} \tag{C.1b}$$

$$H_z = \frac{\frac{\Delta z_-}{\Delta z_+}(H(i,j+1,n) - H(i,j,n)) + \frac{\Delta z_+}{\Delta z_-}(H(i,j,n) - H(i,j-1,n))}{\Delta z_+ + \Delta z_-} \tag{C.1c}$$

$$H_{yy} = \frac{\frac{1}{\Delta y_+}(H^*(i+1,j) - H^*(i,j)) - \frac{1}{\Delta y_-}(H^*(i,j) - H^*(i-1,j))}{(\Delta y_+ + \Delta y_-)/2} \tag{C.1d}$$

$$H_{zz} = \frac{\frac{1}{\Delta z_+}(H(i,j+1,n) - H(i,j,n)) - \frac{1}{\Delta z_-}(H(i,j,n) - H(i,j-1,n))}{(\Delta z_+ + \Delta z_-)/2} \tag{C.1e}$$

And, for the second half step in the x-direction, the following finite-difference equations were used.

$$H_x = (H(i,j,n+1) - H^*(i,j))/(\Delta x/2) \tag{C.1f}$$

$$H_y = \frac{\frac{\Delta y_-}{\Delta y_+}(H^*(i+1,j) - H^*(i,j)) + \frac{\Delta y_+}{\Delta y_-}(H^*(i,j) - H^*(i-1,j))}{\Delta y_+ + \Delta y_-} \tag{C.1g}$$

$$H_z = \frac{\frac{\Delta z_-}{\Delta z_+}(H(i,j+1,n+1) - H(i,j,n+1) + \frac{\Delta z_+}{\Delta z_-}(H(i,j,n+1)}{\Delta z_+ + \Delta z_-}$$

$$\underline{- H(i,j-1,n+1))} \tag{C.1h}$$

$$H_{yy} - \frac{\frac{1}{\Delta y_+}(H^*(i+1,j) - H^*(i,j)) - \frac{1}{\Delta y_-}(H^*(i,j) - H^*(i-1,j))}{(\Delta y_+ + \Delta y_-)/2} \tag{C.1i}$$

$$H_{zz} = \frac{\frac{1}{\Delta z_+}(H(i,j+1,n+1) - H(i,j,n+1)) - \frac{1}{\Delta z_-}(H(i,j,n+1)}{(\Delta z_+ + \Delta z_-)/2}$$

$$\underline{- H(i,j-1,n+1))} \qquad \text{(C.1j)}$$

The symbol (*) denotes the intermediate value of the variable at the point i, j, n+1/2.

The non-linear coefficients in equations 3.5a-c were assumed to be the space average of the previous nodal location and the new nodal location in the x-direction at each nodal location in order to linearize the equations. The half-step values of u, v, and w were not used for the value of these non-linear variables since the half-step values of u, v, and w were to be used only in the calculation of u, v, and w at the new x-location. Thus, the non-linear coefficients were calculated as follows:

$$\overline{H} = \frac{H(i,j,n+1) + H(i,j,n)}{2} \qquad \text{(C.1k)}$$

The value of H(i,j,n+1) for the above equation was obtained in the same manner as the value for H(i,j,n+1) was obtained for equation B.1d.

In order to present the finite-difference equations in a more concise form, the following definitions of the coefficients which make up the matrixes for the first half step were introduced in addition to the definitions introduced in section B.

$$r_3 = \frac{\bar{w}}{\bar{u}} \cdot \frac{\Delta x}{2} \cdot \frac{\Delta z_+}{\Delta z_-} \cdot \frac{1}{(\Delta z_+ + \Delta z_-)} \tag{C.2a}$$

$$r_4 = \frac{\bar{w}}{\bar{u}} \cdot \frac{\Delta x}{2} \cdot \frac{\Delta z_-}{\Delta z_+} \cdot \frac{1}{(\Delta z_+ + \Delta z_-)} \tag{C.2b}$$

$$s_3 = \frac{1}{\bar{u}} \cdot \frac{\Delta x}{\Delta z_-} \cdot \frac{1}{(\Delta z_+ + \Delta z_-)} \tag{C.2c}$$

$$s_4 = \frac{1}{\bar{u}} \cdot \frac{\Delta x}{\Delta z_+} \cdot \frac{1}{(\Delta z_+ + \Delta z_-)} \tag{C.2d}$$

Using the above notation and the notation given by equations B.2a-d, the finite-difference approximations for equations 3.5a-c for the first half x step were as follows:

x-momentum

$$(-r_1 - s_1)\, u^*(i-1,j) + (1.+r_1-r_2 + s_1+s_2)\, u^*(i,j) + (r_2-s_2)\, u^*(i+1,j)$$
$$= (r_3+s_3)u(i,j-1,n) + (1.-r_3+r_4-s_3-s_4)u(i,j,n) + (s_4-r_4)u(i,j+1,n) \tag{C.3a}$$

y-momentum

$$(-r_1-s_1)v^*(i-1,j) + (1.+r_1-r_2+s_1+s_2)v^*(i,j) + (r_2-s_2)v^*(i+1,j)$$
$$= (r_3+s_3)v(i,j-1,n) + (1.-r_3+r_4-s_3-s_4)v(i,j,n) + (s_4-r_4)v(i,j+1,n) \tag{C.3b}$$

z-momentum

$$(-r_1-s_1)w^*(i-1,j) + (1.+r_1-r_2+s_1+s_2)w^*(i,j) + (r_2-s_2)w^*(i=1,j)$$
$$= (r_3+s_3)w(i,j-1,n) + (1.-r_3+r_4-s_3-s_4)w(i,j,n) + (s_4-r_4)w(i,j+1,n) \tag{C.3c}$$

To further condense the equations, the following notation was introduced:

$$A_2(i) = -r_1 - s_1 \tag{C.4a}$$

$$B_2(i) = 1. + r_1 - r_2 + s_1 + s_2 \tag{C.4b}$$

$$C_2(i) = r_2 - s_2 \tag{C.4c}$$

$$D_2(i) = (r_3 + s_3)u(i,j-1,n) + (1. - r_3 + r_4 - s_3 - s_4) \cdot u(i,j,n) + (s_4 - r_4)u(i,j+1,n) \tag{C.4d}$$

$$D_3(i) = (r_3 + s_3)v(i,j-1,n) + (1. -r_3 + r_4 - s_3 - s_4) \cdot v(i,j,n) + (s_4 - r_4)v(i,j+1,n) \tag{C.4e}$$

$$D_4(i) = (r_3 + s_3)w(i,j-1,n) + (1. - r_3 + r_4 - s_3 - s_4) \cdot w(i,j,n) + (s_4 - r_4)w(i,j+1,n) \tag{C.4f}$$

Using the above notation, the following algebraic equations resulted. They were solved simultaneously for the first half x step and yielded the intermediate values of u, v, and w, i.e., u*, v*, and w*.

<u>x-momentum</u>

$$A_2(i)u^*(i-1,j) + B_2(i)u^*(i,j) + C_2(i)u^*(i+1,j) = D_2(i)$$

$$i = 2,3,....k$$

$$j = 2,3,....m \tag{C.5a}$$

<u>y-momentum</u>

$$A_2(i)v^*(i-1,j) + B_2(i)v^*(i,j) + C_2(i)v^*(i+1,j) = D_3(i)$$

$$i = 2,3,....k$$

$$j = 2,3,....m \tag{C.5b}$$

z-momentum

$$A_2(i)w^*(i-1,j)+B_2(i)w^*(i,j) + C_2(i)w^*(i+1,j) = D_4(i)$$

$$i = 2,3,....k$$

$$j = 2,3,....m \qquad \text{(C.5c)}$$

Inspection of equations C.5a-c reveals that all three equations generate matrixes of coefficients which are tridiagonal. The solution of this type of equations is presented later in this Appendix.

Similar to the first half x step, the second half x step matrix coefficients generated by the finite-difference approximations were condensed for ease of notation. The the same notation for the coefficients of the finite-difference approximations given by equations B.2a-d and C.2a-d yields the following for the three momentum equations for the second half x step.

x-momentum

$$(-r_3-s_3)u(i,j-1,n+1) + (1.+r_3-r_4+s_3+s_4)u(i,j,n+1)$$
$$+ (r_4-s_4)u(i,j+1,n+1) = (r_1+s_1)u^*(i-1,j)$$
$$+ (1.-r_1+r_2-s_1-s_2)u^*(i,j) + (s_2-r_2)u^*(i+1,j) \qquad \text{(C.6a)}$$

y-momentum

$$(-r_3-s_3)v(i,j-1,n+1) + (1.+r_3-r_4+s_3+s_4)v(i,j,n+1)$$
$$+ (r_4-s_4)v(i,j+1,n+1) = (r_1+s_1)v^*(i-1,j)$$
$$+ (1.-r_1+r_2-s_1-s_2)v^*(i,j) + (s_2-r_2)v^*(i+1,j) \qquad \text{(C.6b)}$$

z-momentum

$$(-r_3-s_3)w(i,j-1,n+1) + (1.+r_3-r_4+s_3+s_4)w(i,j,n+1)$$
$$+ (r_4-s_4)w(i,j+1,n+1) = (r_1+s_1)w^*(i-1,j)$$
$$+ (1.-r_1+r_2-s_1-s_2)w^*(i,j) + (s_2-r_2)w^*(i+1,j) \qquad (C.6c)$$

Again, to further condense the equations, the following notation was employed:

$$A_3(j) = -r_3-s_3 \qquad (C.7a)$$

$$B_3(j) = 1.+r_3-r_4+s_3+s_4 \qquad (C.7b)$$

$$C_3(j) = r_4-s_4 \qquad (C.7c)$$

$$D_5(j) = (r_1+s_1)u^*(i-1,j) + (1.-r_1+r_2-s_1-s_2)$$
$$u^*(i,j) + (s_2-r_2)u^*(i+1,j) \qquad (C.7d)$$

$$D_6(j) = (r_1+s_1)v^*(i-1,j) + (1.-r_1+r_2-s_1-s_2)$$
$$v^*(i,j) + (s_2-r_2)v^*(i+1,j) \qquad (C.7e)$$

$$D_7(j) = (r_1+s_1)w^*(i-1,j) + (1.-r_1+r_2-s_1-s_2)$$
$$w^*(i,j) + (s_2-r_2)w^*(i+1,j) \qquad (C.7f)$$

Using the notation above, the following algebraic equations resulted, which were solved simultaneously to yield the values of u, v, and w at the next full x step.

x-momentum

$$A_3(j)u(i,j-1,n+1) + B_3(j)u(i,j,n+1) + C_3(j)u(i,j+1,n+1) = D_5(j)$$
$$j = 2,3,....k$$
$$i = 2,3,....m \qquad (C.8a)$$

y-momentum

$$A_3(j)v(i,j-1,n+1) + B_3(j)v(i,j,n+1) + C_3(j)v(i,j+1,n+1) = D_6(j)$$

$$j = 2,3,....k$$

$$i = 2,3,....m \qquad \text{(C.8b)}$$

z-momentum

$$A_3(j)w(i,j-1,n+1) + B_3(j)w(i,j,n+1) + C_3(j)w(i,j+1,n+1) = D_7(j)$$

$$j = 2,3,....k$$

$$i = 2,3,....m \qquad \text{(C.8c)}$$

Again, as for the first half x step, the equations C.8a-c generated matrixes of coefficients which were tridiagonal. The solution for equation of this type is presented later in this Appendix.

D. Finite-Difference Approximations for the Two-Dimensional Turbulent Boundary Layer

The Crank-Nicolson (17) implicit finite-difference method was used to represent the x-momentum equation, 3.14a, in the two-dimensional flat plate region. Using the dummy variable H, the representation of the H_x and H_y derivatives are the same as equations B.1a-b. The term on the right hand side of equation 3.14a was written about the point (i, m+1, n+1/2) where the position specified by the index m+1 is the line 4-5 shown in Figure 3.4.

$$[(1.+\varepsilon_n)H_y]_y = \frac{1}{(\Delta y_+ + \Delta y_-)} \{[1.+\bar{\varepsilon}_n(i+1/2,m+1,n+1)]$$

$$\cdot \frac{H(i+1,m+1,n+1) - H(i,m+1,n+1)}{\Delta y_+} - [1.+\bar{\varepsilon}_n(i-1/2,m+1,n+1)]$$

$$\cdot \frac{H(i,m+1,n+1) - H(i-1,m+1,n+1)}{\Delta y_-} + [1.+\bar{\varepsilon}_n(i+1/2,m+1,n)]$$

$$\cdot \frac{H(i+1,m+1,n) - H(i,m+1,n)}{\Delta y_+} - [1.+\bar{\varepsilon}_n(i-1/2,m+1,n)]$$

$$\cdot \frac{H(i,m+1,n) - H(i-1,m+1,n)}{\Delta y_-} \Big\} \qquad (D.1)$$

where:

$$\bar{\varepsilon}_n(i+1/2,m+1,n+1) = \frac{\varepsilon_n(i+1,m+1,n+1) + \varepsilon_n(i,m+1,n+1)}{2} \qquad (D.2)$$

Similar space averages were used for evaluating the other values of ε_n in the above equations.

The non-linear coefficients in equations 3.14a were determined by equation B.1d.

The following definitions for the coefficients which make up the matrix were used to present the finite-difference equations in a more concise form.

$$s_5 = \frac{(1.+\bar{\varepsilon}_n(i+1/2,m+1,n+1))}{\bar{u}} \left\{\frac{\Delta x}{(\Delta y_+)(\Delta y_+ + \Delta y_-)}\right\} \qquad (D.2a)$$

$$s_6 = \frac{(1.+\bar{\varepsilon}_n(i-1/2,m+1,n+1))}{\bar{u}} \frac{\Delta x}{(\Delta y_-)(\Delta y_+ + \Delta y_-)} \qquad (D.2b)$$

$$s_7 = \frac{(1.+\bar{\varepsilon}_n(i+1/2,m+1,n))}{\bar{u}} \frac{\Delta x}{(\Delta y_-)(\Delta y_+ + \Delta y_-)} \qquad (D.2c)$$

$$s_8 = \frac{(1.+\bar{\varepsilon}_n(i-1/2,m+1,n))}{\bar{u}} \frac{\Delta x}{(\Delta y_-)(\Delta y_+ + \Delta y_-)} \qquad (D.2d)$$

Using the equations B.1a, b and d, B.2a and b and D.2a-d, the finite-difference approximation of equation 3.14a is given by:

$$(-r_2-s_6)u(i-1,m+1,n+1) + (1.-r_1+r_2+s_5+s_6)u(i,m+1,n+1) + (r_2-s_5)$$
$$u(i+1,m+1,n+1) = (r_2+s_8)u(i-1,m+1,n) + (1.+r_1-r_2-s_7-s_8)u(i,m+1,n)$$
$$+ (s_7-r_1)u(i+1,m+1,n) \qquad (D.3)$$

To further condense the equation D.3, the following notation was employed:

$$A_4(i) = -r_2-s_6 \qquad (D.4a)$$

$$B_4(i) = 1.-r_1+r_2+s_5+s_6 \qquad (D.4b)$$

$$C_4(i) = r_1-s_5 \qquad (D.4c)$$

$$D_8(i) = (r_2+s_8)u(i-1,m+1,n) + (1.+r_1-r_2-s_7-s_8)u(i,m+1,n) + (s_7-r_1)u(i+1,m+1,n) \qquad (D.4d)$$

The above notation results in the following algebraic equation. This equation was solved in conjunction with the continuity equation B.4 to predict the u and v profiles for the two-dimensional boundary layer developing on the Y=0 wall.

x-momentum

$$A_4(i)u(i-1,m+1,n+1) + B_4(i)u(i,m+1,n+1) + C_4(i)u(i+1,m+1,n+1) = D_8(i) \qquad i = 2,3,....\ell \qquad (D.5)$$

Again a tridiagonal matrix was generated and its solution technique is discussed in the last portion of this Appendix.

E. Finite-Difference Approximations for the Turbulent Corner Region

Again the ADI method was used to solve the x, y and z momentum equations 3.13a-c in the corner region.

Using the dummy variable H, the derivatives on the left hand side and nonlinear coefficients can be expressed by equations C.1a-c and C.1k for the first half step in the x-direction. The derivatives on the right hand side were expressed in finite-difference form as follows:

$$[(1+\varepsilon_n)H_y]_y = \left[\frac{2}{\Delta y_+ + \Delta y_-}\right] \{(1.+\bar{\varepsilon}_n(i+1/2,j,n+1/2))\frac{H^*(i+1,j) - H^*(i,j)}{\Delta y_+}$$

$$- (1.+\bar{\varepsilon}_n(i-1/2,j,n+1/2))\frac{H^*(i,j) - H^*(i-1,j)}{\Delta y_-} \qquad \text{(E.1a)}$$

$$[(1+\varepsilon_n)H_z]_z = \left[\frac{2}{\Delta y_+ + \Delta y_-}\right] \{(1.+\bar{\varepsilon}_n(i,j+1/2,n))\ \frac{H(i,j+1,n) - H(i,j,n)}{\Delta z_+}$$

$$- (1.+\bar{\varepsilon}_n(i,j-1/2,n))\ \frac{H(i,j,n) - H(i,j-1,n)}{\Delta z_-}\} \qquad \text{(E.1b)}$$

For the second half step, equations C.2a-c and C.1k were used for the left hand side and equation E.1a was used for the expression of H_{yy} in the momentum equations. The following finite-difference expressions were used for the right hand terms, H_{zz}.

$$[(1+\varepsilon_n)H_z]_z = \left[\frac{2}{\Delta z_+ + \Delta z_-}\right] \{(1+\bar{\varepsilon}_n(i,j+1/2,n+1))$$

$$\cdot \frac{H(i,j+1,n+1) - H(i,j,n+1)}{\Delta z_+} - (1+\bar{\varepsilon}_n(i,j-1/2,n+1))$$

$$\cdot \frac{H(i,j,n+1) - H(i,j-1,n+1)}{\Delta z_-}\} \qquad \text{(E.2)}$$

To place the finite-difference equations in a more workable form, the following definitions of the coefficients which make up the matrixes for the first half step were introduced in addition to equations B.2a-b and C.2a-b.

$$s_9 = \frac{(1.+\bar{\varepsilon}_n(i+1/2,j,n+1/2))}{\bar{u}} \left\{\frac{\Delta x}{(\Delta y_+)(\Delta y_+ + \Delta y_-)}\right\} \tag{E.3a}$$

$$s_{10} = \frac{(1.+\bar{\varepsilon}_n(i-1/2,j,n+1/2))}{\bar{u}} \left\{\frac{\Delta x}{(\Delta y_-)(\Delta y_+ + \Delta y_-)}\right\} \tag{E.3b}$$

$$s_{11} = \frac{(1.+\bar{\varepsilon}_n(i,j+1/2,n))}{\bar{u}} \left\{\frac{\Delta x}{(\Delta z_+)(\Delta z_+ + \Delta z_-)}\right\} \tag{E.3c}$$

$$s_{12} = \frac{(1.+\bar{\varepsilon}_n(i,j-1/2,n))}{\bar{u}} \left\{\frac{\Delta x}{(\Delta z_-)(\Delta z_+ + \Delta z_-)}\right\} \tag{E.3d}$$

Using equations C.1a-c, C.1k, B.2a-b, C.2a-b, E.1a-b and E.3a-d, the finite-difference approximations of equations 3.13a-c can be written as follows for the first half x-step:

x-momentum

$$\begin{aligned}&(r_1-s_{10})u^*(i-1,j) + (1.+r_1-r_2+s_9+s_{10})u^*(i,j)\\ &+ (r_2-s_9)u^*(i+1,j) = (r_3+s_{12})u(i,j-1,n)\\ &+ (1.-r_3+r_4-s_{11}-s_{12})u(i,j,n) + (s_{11}-r_4)u(i,j+1,n)\end{aligned} \tag{E.4a}$$

y-momentum

$$\begin{aligned}&(r_1-s_{10})v^*(i-1,j) + (1.+r_1-r_2+s_9+s_{10})v^*(i,j)\\ &+ (r_2-s_9)v^*(i+1,j) = (r_3+s_{12})v(i,j-1,n)\\ &+ (1.-r_3+r_4-s_{11}-s_{12})v(i,j,n) + (s_{11}-r_4)v(i,j+1,n)\end{aligned} \tag{E.4b}$$

z-momentum

$$(r_1-s_{10})w^*(i-1,j) + (1.+r_1-r_2+s_9+s_{10})w^*(i,j)$$
$$+ (r_2-s_9)w^*(i+1,j) = (r_3+s_{12})w(i,j-1,n)$$
$$+ (1.-r_3+r_4-s_{11}-s_{12})w(i,j,n) + (s_{11}-r_4)w(i,j+1,n) \qquad \text{(E.4c)}$$

As before, to further condense the equations, the following additional notation was introduced:

$$A_5(i) = -r_1-s_{10} \qquad \text{(E.5a)}$$

$$B_5(i) = 1.+r_1-r_2+s_9+s_{10} \qquad \text{(E.5b)}$$

$$C_5(i) = r_2-s_9 \qquad \text{(E.5c)}$$

$$D_9(i) = (r_3+s_{12})u(i,j-1,n) + (1.-r_3+r_4-s_{11}-s_{12})$$
$$\cdot u(i,j,n) + (s_{11}-r_4)u(i,j+1,n) \qquad \text{(E.5d)}$$

$$D_{10}(i) = (r_3+s_{12})v(i,j-1,n) + (1.-r_3+r_4-s_{11}-s_{12})$$
$$\cdot v(i,j,n) + (s_{11}-r_4)v(i,j+1,n) \qquad \text{(E.5e)}$$

$$D_{11}(i) = (r_3+s_{12})w(i,j-1,n) + (I.-r_3+r_4-s_{11}-s_{12})$$
$$\cdot w(i,j,n) + (s_{11}-r_4)w(i,j+1,n) \qquad \text{(E.5f)}$$

With the above notation the following algebraic equations resulted which, when solved, yielded values for u*, v*, and w* at the first half step.

x-momentum

$$A_5(i)u^*(i-1,j) + B_5(i)u^*(i,j) + C_5(i)u^*(i+1,j) = D_9(i)$$
$$i = 2,3,....k$$
$$j = 2,3,....m \qquad \text{(E.6a)}$$

y-momentum

$$A_5(i)v^*(i-1,j) + B_5(i)v^*(i,j) + C_5(i)v^*(i+1,j) = D_{10}(i)$$

$$i = 2,3,....k$$

$$j = 2,3,....m \qquad \text{(E.6b)}$$

z-momentum

$$A_5(i)w^*(i-1,j) + B_5(i)w^*(i,j) + C_5(i)w^*(i+1,j) = D_{11}(i)$$

$$i = 2,3,....k$$

$$j = 2,3,....m \qquad \text{(E.6c)}$$

Again, the above three equations yield tridiagonal matrixes. The solution technique is presented in the last portion of this Appendix.

Similar to the first half x-step, the second half x-step matrix coefficients were condensed for ease of notation. With the addition of two more equations,

$$s_{13} = (1.+\bar{\varepsilon}_n(i,j+1/2,n+1))\left\{\frac{\Delta x}{(\Delta z_+)(\Delta z_+ + \Delta z_-)}\right\} \qquad \text{(E.7a)}$$

$$s_{14} = (1.+\bar{\varepsilon}_n(i,j-1/2,n+1))\left\{\frac{\Delta x}{(\Delta z_-)(\Delta z_+ + \Delta z_-)}\right\} , \qquad \text{(E.7b)}$$

the following equations were generated for the three momentum equations for the second half step.

x-momentum

$$(-r_3-s_{14})u(i,j-1,n+1) + (1.+r_3-r_4+s_{13}+s_{14})u(i,j,n+1)$$
$$+ (r_4-s_{13})u(i,j+1,n+1) = (r_1+s_{10})u^*(i-1,j)$$
$$+ (1.-r_1+r_2-s_{10}-s_{11})u^*(i,j) + (s_9-r_2)u^*(i+1,j) \qquad \text{(E.8a)}$$

y-momentum

$$(-r_3-s_{14})v(i,j-1,n+1) + (1.+r_3-r_4+s_{13}+s_{14})v(i,j,n+1)$$
$$+ (r_4-s_{13})v(i,j+1,n+1) = (r_1+s_{10})v^*(i-1,j)$$
$$+ (1.-r_1+r_2-s_{10}-s_{11})v^*(i,j) + (s_9-r_2)v^*(i+1,j) \qquad \text{(E.8b)}$$

z-momentum

$$(-r_3-s_{14})w(i,j-1,n+1) + (1.+r_3-r_4+s_{13}+s_{14})w(i,j,n+1)$$
$$+ (r_4-s_{13})w(i,j+1,n+1) = (r_1+s_{10})w^*(i-1,j)$$
$$+ (1.-r_1+r_2-s_{10}-s_{11})w^*(i,j) + (s_9-r_2)w^*(i+1,j) \qquad \text{(E.8c)}$$

Again the equations were further condensed by employing the following notations:

$$A_6(j) = -r_3-s_{14} \qquad \text{(E.9a)}$$

$$B_6(j) = 1.+r_3-r_4+s_{13}+s_{14} \qquad \text{(E.9b)}$$

$$C_6(j) = r_4-s_{13} \qquad \text{(E.9c)}$$

$$D_{12}(j) = (r_1+s_{10})u^*(i-1,j) + (1.-r_1+r_2-s_{10}-s_{11})u^*(i,j)$$
$$+ (s_9-r_2)u^*(i+1,j) \qquad \text{(E.9d)}$$

$$D_{13}(j) = (r_1+s_{10})v^*(i-1,j) + (1.-r_1+r_2-s_{10}-s_{11})v^*(i,j)$$
$$+ (s_9-r_2)v^*(i+1,j) \qquad \text{(E.9e)}$$

$$D_{14}(j) = (r_1+s_{10})w^*(i-1,j) + (1.-r_1+r_2-s_{10}-s_{11})w^*(i,j)$$
$$+ (s_9-r_2)w^*(i+1,j) \qquad \text{(E.9f)}$$

With the above notation, the following algebraic equations resulted which were solved for u, v, and w at the next full step in the x direction.

$$A_6(j)u^*(i,j-1) + B_6(j)u^*(i,j) + C_6(j)u^*(i,j+1) = D_{12}(j)$$

$$j = 2,3,....k$$

$$i = 2,3,....m \qquad \text{(E.10a)}$$

$$A_6(j)v^*(i,j-1) + B_6(j)v^*(i,j) + C_6(j)v^*(i,j+1) = D_{13}(j)$$

$$j = 2,3,....k$$

$$i = 2,3,....m \qquad \text{(E.10b)}$$

$$A_6(j)w^*(i,j-1) + B_6(j)w^*(i,j) + C_6(j)w^*(i,j+1) = D_{14}(j)$$

$$j = 2,3,....k$$

$$i = 2,3,....m \qquad \text{(E.10c)}$$

Again these equations also yielded tridiagonal matrixes and were solved by the Thomas Algorithm which is discussed last in the Appendix.

F. Finite-Difference Approximation of Continuity Equation in the Corner Region

The continuity equation for the corner region was written in finite-difference form about the point (i+1/2,j+1/2,n+1/2) so that truncation error would be of $0(\Delta x)^2$, $0(\Delta y)^2$, and $0(\Delta z)^2$ for the truncation error. Figure A.3 illustrates the grid used for the equation which can be written as follows:

$$\{[u(i+1,j+1,n+1) + u(i+1,j,n+1) + u(i,j+1,n+1) + u(i,j,n+1)$$
$$- u(i+1,j+1,n) - u(i+1,j,n) - u(i,j+1,n) - u(i,j,n)]/dx$$
$$+ [v(i+1,j+1,n+1) + v(i+1,j,n+1) + v(i+1,j+1,n) + v(i+1,j,n)$$
$$- v(i,j+1,n+1) - v(i,j,n+1) - v(i,j+1,n) - v(i,j,n)]/dy$$
$$+ [w(i+1,j+1,n+1) + w(i+1,j+1,n) + w(i,j+1,n+1) + w(i,j+1,n)$$
$$- w(i+1,j,n+1) - w(i+1,j,n) - w(i,j,n+1) - w(i,j,n)]/dz\}/4. = 0 \qquad \text{(F.1)}$$

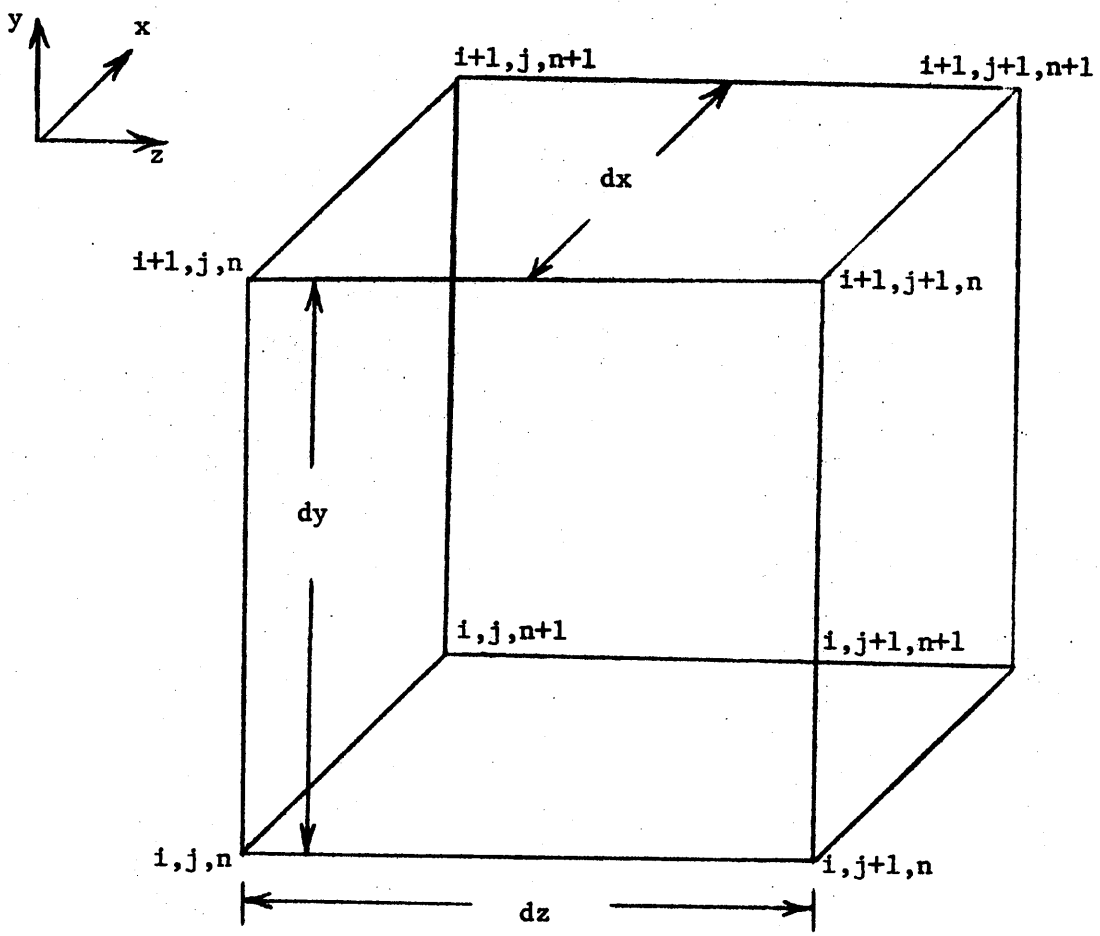

Fig. A.3. Grid used for finite-difference approximation for continuity equation in the corner region.

The above equation was used to obtain the residual of the continuity equation at the points (i+1/2,j+1/2,n+1/2) throughout the corner region.

G. Mixing Length for the Corner Region

Pletcher's (22) mixing length model for the two-dimensional flow was adapted for use in the corner region in the following manner. Figure A.3 illustrates the corner region and the grid lines.

When attempting to estimate the eddy-viscosity at each nodal position, a modified mixing length was used. For each nodal position, the line A-A was passed through the node and the grid position where the two-dimensional boundary lines intersected, point C. It was further assumed that the isovel 0.999 continued in a circular arc B-B from one of the two-dimensional boundaries to the other. The distance along the line A-A from the wall to where the circular arc B-B intersected the line A-A was considered the corner boundary thickness for the mixing length model. The distance along the line A-A from the intersection with the wall to the nodal position in question was considered the Y distance. Together, the two positions on the line A-A were used in Pletcher's mixing length model to generate a modified mixing length at each nodal position. The shear velocity, u_s, which was necessary for Pletcher's formulation, was taken as equal to the two-dimensional region shear velocity since experiments by Gersten (12) have shown little or no change in total skin friction for planes at ninety degrees to a flat plate with an equivalent wetted area.

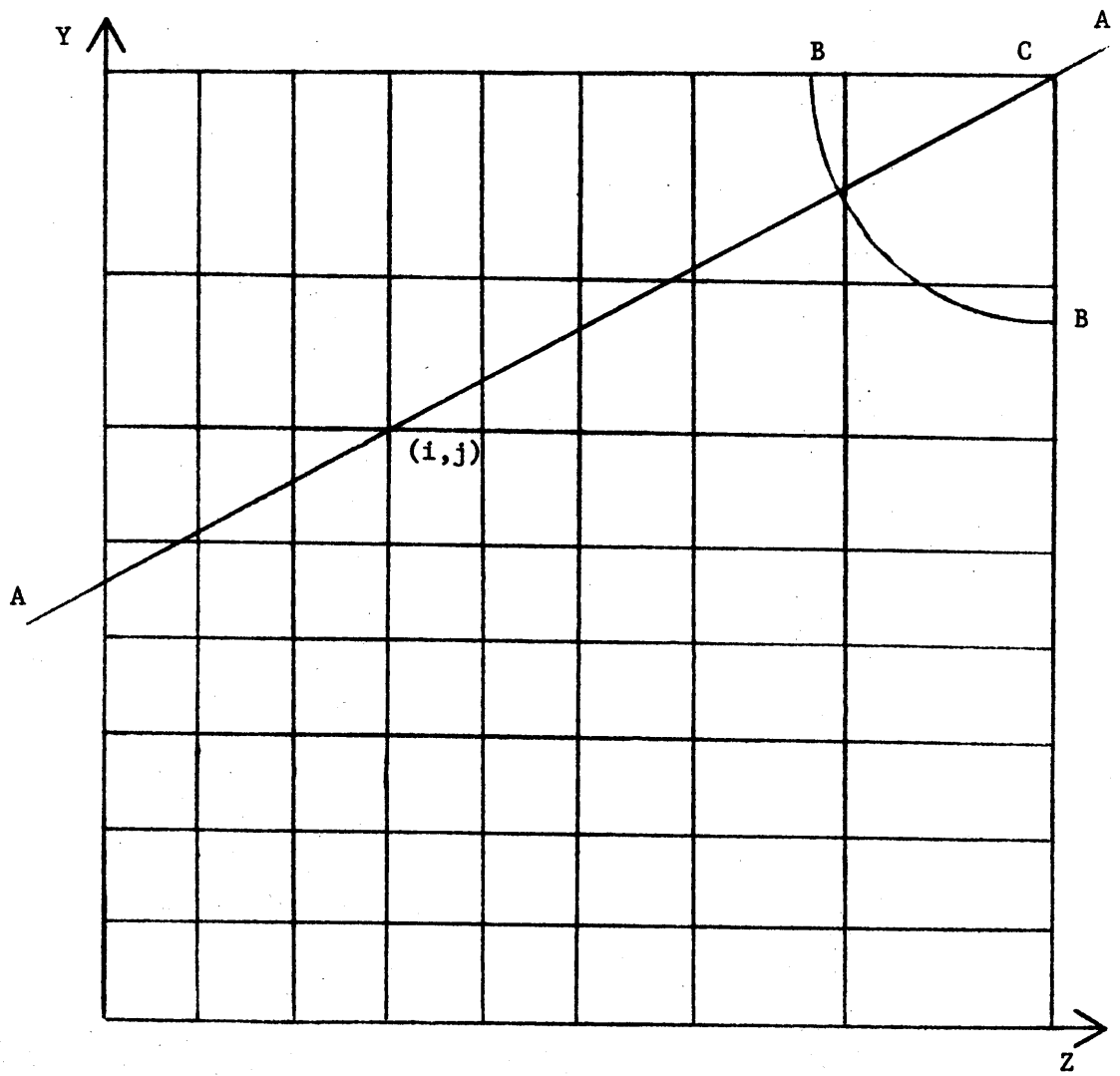

Fig. A.4. Corner Grid.

H. Thomas Algorithm

The following describes the Thomas algorithm as applied to a general tridiagonal matrix as the one given below for n equations for the dummy variable h.

$$
\begin{aligned}
b_1h_1 + c_1h_2 &= d_1 \\
a_2h_1 + b_2h_2 + c_2h_3 &= d_2 \\
a_3h_2 + b_3h_3 + c_3h_4 &= d_3 \\
&\cdots\cdots\cdots\cdots \\
a_{n-1}h_{n-2} + b_{n-1}h_{n-1} + c_{n-1}h_n &= d_{n-1} \\
a_nh_{n-1} + b_nh_n &= d'_n
\end{aligned}
$$

Where: $a_1 = 0$ or $h_o = 0$ and $d'_n = d_n - c_nh_{n+1}$

The matrix is then transformed into upper bidiagonal matrix by a Gaussian elemination generating the following coefficients:

$$
\begin{aligned}
e_1 &= b_1 \\
f_1 &= d_1/b_1 \\
e_1 &= b_i - a_iC_{i-1}/e_i \qquad i = 2,3,\ldots.n \\
f_i &= (d_i - a_if_{i-1})/e_i \qquad i = 2,3,\ldots.n
\end{aligned}
$$

The transformed matrix is as follows

$$
\begin{aligned}
h_1 + e_1h_2 &= f_1 \\
h_2 + e_2h_3 &= f_2 \\
&\cdots\cdots\cdots\cdots
\end{aligned}
$$

$$h_{n-1} + e_{n-1} h_n = f_{n-1}$$

$$h_n = f_n$$

Thus, once f_n is determined, h_n is known and the remaining values of the dummy variable h can be obtained from the following equation.

$$h_i = f_i - c_i u_{i+1}/e_i \qquad i = n-1, n-2, \ldots .1$$

The above method was considered most appropriate for the solution of the system of equations generated by this finite-difference technique. The method greatly reduced the number of calculations necessary to solve the ever increasing large system of equations generated as the solution proceeded in the x-direction. Without this method, matrix inversion methods would have been necessary which generally produce an even larger number of equations for solution and thus an even larger number of calculations for each x-step.

VIII. APPENDIX B

The following contains a listing of the computer programs. The first program is for the laminar case and the second is for the turbulent case. If desired, different initial u, v, and w profiles can be used by adding the appropriate data cards to the BLOCK DATA subprogram.

Any time the subprogram OUTPUT is called, the matrixes containing the u, v, and w components are printed. Thus, the programer can have these matrixes printed by controlling with one or more parameters and obtain information only when necessary or desired by calling the subprogram OUTPUT.

```
C   THIS PROGRAMME CALULATES THE VELOCITY COMPONENTS FOR FLOW ALONG A CORNER
C   FORMED BY TWO 90 DEGREE PLATES
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER BLTP,COUNT,CBL
      DIMENSION DY(70),D(70)
      COMMON X,DX,VK,DEN,VD,UINF,PLI,LA
C   ***************************************************************************
C   LABEL COMMON USED TO ENSURE CORRECT TRANSFER OF VARIABLES BETWEEN MAIN AND
C   SUBROUTINES
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2)
      COMMON /VELTEM/ UT(70,70),VT(70,70),WT(70,70)
      COMMON /DIMENS/ YM(70),YP(70),YMYP(70),YPYM(70),DYD(70)
      COMMON /COFF/ A(70),B(70),C(70),E(70),F(70)
      COMMON /YLENG/ Y(71)
      DATA UINFD/ 1.DO /
      DATA ZE,HALF,ONEE,TWOO/0.DO,.5DO,1.DO,2.DO/
 1001 FORMAT(7D11.4/D11.4,5I3)
 1002 FORMAT('1',20X,'THE VALUE OF JZ IS NOW > THE INDEX,',I4,' .')
 1003 FORMAT(15X,'$$$$SUBROUTINE CHECK HAS RUN THE 2-D BOUNDARY FURTHER
     1THAN THE INDEX WHICH IS ',I3,' .')
C   ***************************************************************************
C   X      --- STARTING POINT OF FLOW, INCHES
C   DX     --- STARTING STEP SIZE IN THE MAIN FLOW DIRECTION, INCHES
C   VK     --- KINEMATIC VISCOSITY
C   DEN    --- DENSITY
C   VD     --- DYNAMIC VISCOSITY
C   UINF   --- VELOCITY AT INFINITY IN THE X DIRECTION
C   PLI    --- PLATE LENGTH, INCHES
C   DY(1)  --- FIRST STEP SIZE IN Y AND Z DIRECTION, INCHES
C   BLTP   --- ESTIMATION OF BOUNDARY LAYER THICKNESS, NUMBER OF STEPS
C   CBL    --- ESTIMATION OF NUMBER OF STEPS IN 2-D FLOW
C   COUNT  --- COUNT OF NUMBER OF STEPS TAKEN IN THE X DIRECTION
```

```
C  LA     --- PARAMETER FOR TYPE OF FLOW: LAMINAR=1, TURBULENT=2
C  INDEX --- SIZE OF MATRIX
C  ********************************************************************************
      READ(5,1001) X,DX,VK,DEN,VD,UINF,PLI,DY(1),BLTP,CBL,COUNT,LA,INDEX
      CALL TITLE (DY(1))
      REC=819.D3
      JZ=BLTP+CBL
C  DO LOOP TO SET WALL VELOCITIES TO ZERO
      DO 9 I=1,INDEX
       U(I,1,1)=ZE
      UT(1,I)=ZE
      UT(I,1)=ZE
    9  U(1,I,1)=ZE
      TWE=ONEE/12.DO
      PLF=PLI*TWE
      DISD=DSQRT(UINF/(VK*PLF))
      VELD=DSQRT(UINF*VK/PLF)
C  ********************************************************************************
C  THE NEXT 21 STATEMENTS DETERMINE THE VARIABLE GRID SPACING IN THE Y & Z
C   DIRECTION AND NON-DIMENSIONALIZE THE COEFFICIENTS FOR THE TRIDIAGONAL MATRIX
C   AND THE GRID SPACING
      DO 1 I=1,5
      DY(I)=DY(1)
    1 Y(I+1)=Y(I)+DY(I)
      DO 2 I=6,20
      DY(I)=DY(I-1)*1.05DO
    2 Y(I+1)=Y(I)+DY(I)
      DO 3 I=21,40
      DY(I)=DY(I-1)*1.1DO
    3 Y(I+1)=Y(I)+DY(I)
      DO 4 I=41,INDEX
      DY(I)=DY(I-1)*1.15DO
```

```
    4 Y(I+1)=Y(I)+DY(I)
      DYD(1)=DY(1)*DISD*TWE
      DO 6 I=2,INDEX
      IL1=I-1
      DYD(I)=DY(I)*DISD*TWE
      SUMY=1.DO/(DYD(IL1)+DYD(I))
      YM(I)=SUMY/DYD(IL1)
      YP(I)=SUMY/DYD(I)
      YMYP(I)=HALF*SUMY*DYD(IL1)/DYD(I)
    6 YPYM(I)=HALF*SUMY*DYD(I)/DYD(IL1)
C   ***************************************************************************
C   NON-DIMENSIONALIZE THE STEP SIZE IN THE X DIRECTION
    7 DXD=DX/PLI
C   TRANSPOSE PREVIOUS VALUES OF THE VELOCTY COMPONENTS TO THE NEXT STEP AS
C    FIRST GUESS
      DO 11 I=1,INDEX
      DO 11 J=1,INDEX
       U(I,J,2)=U(I,J,1)
       V(I,J,2)=V(I,J,1)
   11  W(I,J,2)=W(I,J,1)
C   ***************************************************************************
C   START SOLUTION OF 2-D BOUNDARY LAYER. BOUNDARY LAYER THICKNESS IS INCREASED
C    UNTIL U AT EDGE IS 0.99999
      ISTOP=BLTP-1
  101 DO 40 IT=1,3
      DO 20 N=2,ISTOP
      N1=N-1
      UB=   (U(N,JZ,2)+U(N,JZ,1))*HALF
      VB=   (V(N,JZ,2)+V(N,JZ,1))*HALF
      IF(V(N,JZ,1).EQ.ZE) VB=TWOO*VB
      SPE=DXD/UB
      R=VB*SPE*YMYP(N)
```

```
      RR=VB*SPE*YPYM(N)
      S=SPE*YP(N)
      SS=SPE*YM(N)
      A(N)=-RR-SS
      B(N)=   -R+RR+S+SS+ONEE
       C(N)=R-S
   20 D(N)=(RR+SS)*U(N1,JZ,1)+(ONEE+R-RR-S-SS)*U(N,JZ,1)+(S-R)*U(N+1,JZ,
     11)
      D(2)=D(2)-A(2)*U(1,JZ,1)
      D(ISTOP)=D(ISTOP)-C(ISTOP)*UINFD
      CALL SOLVE(D,ISTOP)
      U(ISTOP,JZ,2)=F(ISTOP)
      L=ISTOP-1
      DO 30 I=2,L
      K=(L+2)-I
   30 U(K,JZ,2)=F(K)-C(K)*U(K+1,JZ,2)/E(K)
      DO 40 N=2,BLTP
      N1=N-1
   40 V(N,JZ,2)=V(N1,JZ,2)-   DYD(N1)*(U(N,JZ,2)+U(N1,JZ,2)-U(N,JZ,1)-
     1U(N1,JZ,1))/DXD*HALF
      DO 50 M=1,ISTOP
      IF(U(M,JZ,2).GT..99999D0.AND.U(M,JZ,2).LE.ONEE) GO TO 60
   50 CONTINUE
      ISTOP=BLTP
      BLTP=BLTP+1
      IF(JZ.GT.INDEX) GO TO 1000
      JZ=JZ+1
      GO TO 101
C  TRANSPOSE SOLUTION OF U, V, & W AT 2-D STATIONS TO ALL STATIONS WITHIN THE
C  2-D BOUNDARY LAYER AWAY FROM THE CORNER
   60 DO 70 K=JZ,INDEX
      DO 70 I=2,BLTP
```

```
      U(I,K,2)=U(I,JZ,2)
      U(K,I,2)=U(I,JZ,2)
      V(I,K,2)=V(I,JZ,2)
   70 W(K,I,2)=V(I,JZ,2)
      LUS=BLTP+1
      DO 110 K=LUS,INDEX
      DO 10 I=LUS,INDEX
      V(I,K,2)=V(BLTP,JZ,2)
      W(K,I,2)=V(BLTP,JZ,2)
      IF(K.EQ.I) GO TO 110
   10 CONTINUE
  110 CONTINUE
      DO 80 I=2,BLTP
      ID=I
      IF(U(I,JZ,2).GT..99D0) GO TO 100
   80 CONTINUE
  100 CALL VELSOL(JZ,BLTP,UINFD,DXD)
      IF(COUNT.LT.20) GO TO 120
      CALL CHECK (BLTP,JZ,INC,INDEX,IQUIT,ID)
      IF(IQUIT.EQ.INDEX) GO TO 2000
      IF(INC.EQ.0) GO TO 100
      IF(JZ.EQ.INDEX) GO TO 2000
  120 X=X+DX
C   ***************************************************************************
C   CONTROLLED OUTPUT AT DESIRED REYNOLDS NUMBERS. (ANY OTHER TYPE OF CONTROL
C    IS POSSIBLE)
      RE=UINF*X*TWE/VK
      IF(RE.GT.REC) CALL OUTPUT (JZ,BLTP,ID,COUNT,RE,X)
      IF(RE.GT.REC) CALL CONTIN (JZ,COUNT,DXD)
      IF(RE.GT.REC) REC= TWO0*REC
      IF(COUNT.GT.20) DX=1.1D0*DX
      IF(DX.GT.Y(ID)) DX=Y(ID)
```

```
C  DO LOOP TO STORE NEW VALUES OF U, V, & W IN OLD POSITION SO THAT THE
C   CALCULATIONS CAN ADVANCE TO THE NEXT X STEP
      DO 8 I=1,INDEX
      DO 8 J=1,INDEX
      U(I,J,1)=U(I,J,2)
      V(I,J,1)=V(I,J,2)
    8 W(I,J,1)=W(I,J,2)
      COUNT=COUNT+1
      IF(COUNT.LE.1000) GO TO 7
 1000 WRITE(6,1002) INDEX
      STOP
 2000 WRITE(6,1003) INDEX
      STOP
      END
```

```
      SUBROUTINE SOLVE (D,K)
C   SUBPROGRAM TO CALCULATE THE SOLUTION TO THE TRIDIAGONAL MATRIX
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /COFF/ A(70),B(70),C(70),E(70),F(70)
      DIMENSION D(70)
      E(2)=B(2)
      F(2)=D(2)/B(2)
      DO 1 I=3,K
       I1=I-1
      E(I)=B(I)-A(I)*C(I1)/E(I1)
    1 F(I)=(D(I)-A(I)*F(I1))/E(I)
      RETURN
      END
```

```
      SUBROUTINE VELSOL(JZ,BLTP,UINFD,DXD)
C  SUBPROGRAM TO SOLVE THE X, Y, & Z MOMENTUM EQUATIONS
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER  BLTP
      DIMENSION D(70) ,DV(70),DW(70)
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2)
      COMMON /VELTEM/ UT(70,70),VT(70,70),WT(70,70)
      COMMON /DIMENS/ YM(70),YP(70),YMYP(70),YPYM(70),DYD(70)
      COMMON /COFF/ A(70),B(70),C(70),E(70),F(70)
      DATA ZE,HALF,ONEE,TWOO/0.D0,.5D0,1.D0,2.D0/
      JZZ=JZ-1
C  MOMENTUM EQUATIONS ARE ITERATED THREE TIMES
      DO 9 IT=1,3
      CALL SETBOU (BLTP,JZ)
      K=JZZ
C  STARTING DO LOOP FOR EXPLICIT STRIPPING IN THE Z DIRECTION
      DO 4 J=2,JZZ
      JP1=J+1
      JM1=J-1
      R3=YPYM(J)
      R4=YMYP(J)
      S3=YP(J)
      S4=YM(J)
      R5=R4-R3
      S5=S3+S4
      IF(J.GT.BLTP) K=K-1
      KP1=K+1
      L=K-1
C  STARTING DO LOOP FOR IMPLICIT STRIPPING IN THE Y DIRECTION
      DO 1 I=2,K
      UB=    (U(I,J,2)+U(I,J,1))*HALF
      VB=    (V(I,J,2)+V(I,J,1))*HALF
```

```
      WB=    (W(I,J,2)+W(I,J,1))*HALF
      IF(V(I,J,1).EQ.ZE) VB=TWOO*VB
      IF(W(I,J,1).EQ.ZE) WB=TWOO*WB
      SPE=DXD/UB
      R1=VB*SPE*YPYM(I)
      R2=VB*SPE*YMYP(I)
      S1=SPE*YM(I)
      S2=SPE*YP(I)
C  COEFFICIENTS FOR THE TRIDIAGONAL MATRIX FOR EACH EQUATION
      A(I)=-R1-S1
      B(I)=    R1-R2+S2+S1+ONEE
      C(I)=R2-S2
      P=(WB*R3+S4)*SPE
      Q=ONEE+(WB*R5-S5)*SPE
      T=(S3-WB*R4)*SPE
      D (I)=P*U(I,JM1,1)+Q*U(I,J,1)+T*U(I,JP1,1)
      DV(I)=P*V(I,JM1,1)+Q*V(I,J,1)+T*V(I,JP1,1)
    1 DW(I)=P*W(I,JM1,1)+Q*W(I,J,1)+T*W(I,JP1,1)
      D (2)=D (2)-A(2)*U(1,JM1,1)
      DV(2)=DV(2)-A(2)*V(1,JM1,1)
      DW(2)=DW(2)-A(2)*W(1,JM1,1)
      D (K)=D (K)-C(K)*UT(KP1,J)
      DV(K)=DV(K)-C(K)*VT(KP1,J)
      DW(K)=DW(K)-C(K)*WT(KP1,J)
      CALL SOLVE (D,K)
C  SOLVE FOR TEMPORARY U VALUES
      UT( K ,J)=F( K )
      DO 2 M=2,L
      N=(L+2)-M
    2 UT(N,J)=F(N)-C(N)*UT(N+1,J)/E(N)
C SOLVE FOR TEMPORARY V VALUES
      CALL SOLVE (DV,K)
```

```
      VT( K ,J)=F( K )
      DO 3 M=2,L
      N=(L+2)-M
    3 VT(N,J)=F(N)-C(N)*VT(N+1,J)/E(N)
C SOLVE FOR TEMPORARY W VALUES
      CALL SOLVE (DW,K)
      WT( K ,J)=F( K )
      DO 4 M=2,L
      N=(L+2)-M
    4 WT(N,J)=F(N)-C(N)*WT(N+1,J)/E(N)
      K=JZZ
C  STARTING DO LOOP FOR EXPLICIT STRIPPING IN Y DIRECTION
      DO 8  I=2,JZZ
      IM1=I-1
      IP1=I+1
      IF(I.GT.BLTP)K=K-1
      KP1=K+1
      L=K-1
      R3=YPYM(I)
      R4=YMYP(I)
      S3=YP(I)
      S4=YM(I)
      R5=R4-R3
      S5=S3+S4
C  STARTING DO LOOP FOR IMPLICIT STRIPPING IN Z DIRECTION
      DO 5 J=2,K
      UB=   (U(I,J,2)+U(I,J,1))*HALF
      VB=   (V(I,J,2)+V(I,J,1))*HALF
      WB=   (W(I,J,2)+W(I,J,1))*HALF
      IF(V(I,J,1).EQ.ZE) VB=TWOO*VB
      IF(W(I,J,1).EQ.ZE) WB=TWOO*WB
      SPE=DXD/UB
```

```
      R1=WB*SPE*YPYM(J)
      R2=WB*SPE*YMYP(J)
      S1=SPE*YM(J)
      S2=SPE*YP(J)
C  COEFFICIENTS FOR THE TRIDIAGONAL MATRIX
      A(J)=-R1-S1
      B(J)=   R1-R2+S1+S2+ONEE
      C(J)=R2-S2
      P=(VB*R3+S4)*SPE
      Q=ONEE+(VB*R5-S5)*SPE
      T=(S3-VB*R4)*SPE
      D (J)=P*UT(IM1,J)+Q*UT(I,J)+T*UT(IP1,J)
      DV(J)=P*VT(IM1,J)+Q*VT(I,J)+T*VT(IP1,J)
    5 DW(J)=P*WT(IM1,J)+Q*WT(I,J)+T*WT(IP1,J)
      D (K)=D (K)-C(K)*U(I,KP1,2)
      DV(K)=DV(K)-C(K)*V(I,KP1,2)
      DW(K)=DW(K)-C(K)*W(I,KP1,2)
C  SOLVE FOR NEW U VALUES
      CALL SOLVE(D,K)
      U(I, K ,2)=F( K )
      DO 6 M=2,L
      N=(L+2)-M
    6 U(I,N,2)=F(N)-C(N)*U(I,N+1,2)/E(N)
C  SOLVE FOR NEW V VALUES
      CALL SOLVE(DV,K)
      V(I, K ,2)=F( K )
      DO 7 M=2,L
      N=(L+2)-M
    7 V(I,N,2)=F(N)-C(N)*V(I,N+1,2)/E(N)
C  SOLVE FOR NEW W VALUES
      CALL SOLVE(DW,K)
      W(I, K ,2)=F( K )
```

```
      DO 8 M=2,L
      N=(L+2)-M
    8 W(I,N,2)=F(N)-C(N)*W(I,N+1,2)/E(N)
    9 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SETBOU (K,L)
C   SUBPROGRAM WHICH FIXES THE TEMPORARY U, V, & W ON 2-D BOUNDARIES AT AVERAGE
C    OF LAST STEP AND NEW STEP U, V, & W
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2)
      COMMON /VELTEM/ UT(70,70),VT(70,70),WT(70,70)
      HALF=.5D0
      DO 10 J=2,K
      UT(L,J)=(U(L,J,2)+U(L,J,1))*HALF
      UT(J,L)=(U(J,L,2)+U(J,L,1))*HALF
      VT(J,L)=(V(J,L,2)+V(J,L,1))*HALF
   10 WT(L,J)=(W(L,J,2)+W(L,J,1))*HALF
      KK=K+1
      LL=L-1
      N=L
      DO 20 J=KK,LL
      N=N-1
      VT(N,J)=(V(N,J,2)+V(N,J,1))*HALF
   20 WT(N,J)=(W(N,J,2)+W(N,J,1))*HALF
      RETURN
      END
```

```
      SUBROUTINE CHECK (L,J,K,M,N,ID)
C   THIS SUBROUTINE CHECKS THE SLOPE OF THE U ISOVELS AT THE TWO-DIMENSIONAL
C    BOUNDARY.
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2)
      COMMON /DIMENS/ YM(70),YP(70),YMYP(70),YPYM(70),DYD(70)
      JP=J+1
      JL=J-1
      DY=DYD(J)+DYD(JL)
      SLOPE=DABS((U(ID+1,J,2)-U(ID-1,J,2))/(DYD(ID)+DYD(ID-1)))
      DO 10 I = 2,ID
      DEVY=DABS((U(I,JP,2)-U(I,JL,2))/DY)
      IF(DEVY.GT.SLOPE) GO TO 20
   10 CONTINUE
      GO TO 30
   20 J=J+1
      K=0
      N=0
      IF(J.GT.M) N=M
      RETURN
   30 K=1
      RETURN
      END
```

```
      SUBROUTINE TITLE (DY)
C   SUBPROGRAM TO PRINT THE PARAMETERS OF THE FLOW
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON X,DX,VK,DEN,VD,UINF,PLI,LA
 1001 FORMAT('1',15X,'PHYSICAL PARAMETERS OF PLATE:'///20X,'THE FLOW STARTS AT '
     1RTS AT ',F8.3,' INCHES FROM THE LEADING EDGE.'//20X,'THE PLATE IS
     2',F8.3,' INCHES IN DEPTH.'//20X,'THE INITIAL STARTING STEP SIZE IN
     3 THE X DIRECTION IS ',F6.4,' INCHES'//20X,'THE STEP SIZE IN THE Y
     4AND Z DIRECTION IS ',F6.4,' INCHES.'////15X,'PARAMETERS OF THE FLU
     5ID:'///20X,'THE VELOCITY OF THE MAIN STREAM FLOW IN THE X DIRECTIO
     6N IS ',F8.1,' FT/SEC'//20X,'KINEMATIC VISCOSITY IS ',F13.9,' SQ FT
     7/SEC'//20X,'DYNAMIC VISCOSITY IS ',F13.9,'LBF-SEC/SQ FT'//20X,'DENSI
     8SITY IS ',F13.9,'LBF-SQ SEC/FT TO FOURTH')
 1002 FORMAT(/  20X,'THE FLOW IS LAMINAR. *****')
 1003 FORMAT(/  15X,'THE FLOW IS TURBULENT. *****')
 1004 FORMAT(///20X,'THE COMPONENTS OF VELOCITY ARE U IN THE X DIRECTION
     1, V IN THE Y DIRECTION AND W IN THE Z DIRECTION.'/20X,'FOR EACH I
     2AND J POSITION AT THE INDICATED X POSITION, VALUES FOR U, V, & W A
     3RE PRINTED. THE FIRST VALUE IS U,'/20X,'THE SECOND IS V, AND THE T
     4HIRD IS W. POSITIVE U IS INTO THE PAPER, POSITIVE V IS UP THE PAPE
     5R, & POSITIVE W IS'/20X,'IS TO THE RIGHT.')
      WRITE(6,1001) X,PLI,DX,DY,UINF,VK,VD,DEN
      IF(LA.EQ.0) WRITE(6,1002)
      IF(LA.EQ.1) WRITE(6,1003)
      WRITE(6,1004)
      RETURN
      END
```

```
      SUBROUTINE OUTPUT(JZ,BLTP,ID,COUNT,RE,X)
C   SUBPROGRAM TO PRINT U, V, & W AT GIVE REYNOLDS NUMBERS
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER BLTP,COUNT
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2)
      COMMON /YLENG/ Y(71)
 1001 FORMAT('1'//////////20X,'AT ',I5,' STEPS DOWN THE PLATE OR ',F8.4
     1,'INCHES FROM THE LEADING EDGE,'/20X,'THE REYNOLDS NUMBER HAS A VA
     2LUE OF ',F10.0)
 1002 FORMAT(////20X,'THE 2-D BOUNDARY LAYER AT THIS POSITION IS ',F12.8
     1,' INCHES.'/20X,'THE POSITION WHERE U/UINF IS > 0.99999 IS ',F12.8
     2,' INCHES.')
 1003 FORMAT(4X,10(1X,D11.4))
 1004 FORMAT(I4,10(1X,D11.4),I4)
 1005 FORMAT(4X,10(1X,D11.4)//)
 1006 FORMAT( 8X,9(I2,10X),I2/)
 1007 FORMAT('1', 8X,9(I2,10X),I2/)
 1008 FORMAT(////20X,'THE POSITION AT WHICH THE FLOW IS 2-D IS ',F12.8,'
     1 INCHES FROM THE CORNER.'////20X,'THE INDEX FOR THE ABOVE POSTION
     2IS ',I3,' .'//20X,'THE 2-D INDEX IS ',I3,' .')
      DIMENSION ETA(70)
      IN=(JZ+9)/10*10
      CONS=DSQRT(.5D0*RE)/X
      DO 3 K=1,IN
    3 ETA(K)=Y(K)*CONS
      WRITE(6,1001) COUNT,X,RE
      WRITE(6,1002) Y(ID),Y(BLTP)
      WRITE(6,1008) Y(JZ),JZ,BLTP
      DO 2 M=1,JZ,10
      N=M+9
      DO 2 K=1,JZ,10
      WRITE(6,1007)(J,J=M,N)
```

```
      L=((JZ+10-K)/10)*10
      KK=L-9
      LL=L+KK
      DO 1 II=KK,L
      I=LL-II
      WRITE(6,1003)(U(I,J,2),J=M,N)
      WRITE(6,1004)I,(V(I,J,2),J=M,N),I
    1 WRITE(6,1005)(W(I,J,2),J=M,N)
      WRITE(6,1006)(J,J=M,N)
    2 WRITE(6,1003) (ETA(J),J=M,N)
      RETURN
      END
```

```
      SUBROUTINE CONTIN (N,M,X)
C   THIS SUBROUTINE DETERMINES THE VALUE OF THE CONTINUITY EQUATION. IT ALSO
C    EQUATION FOR THE REGION OF THE CORNER
C    DETERMINES THE ABSOLUTE AVERAGE AND REAL AVERAGE OF THE CONTINUITY
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2)
      COMMON /VELTEM/ UT(70,70),VT(70,70),WT(70,70)
      COMMON /DIMENS/ YM(70),YP(70),YMYP(70),YPYM(70),DYD(70)
      DATA QUA / .25D0 /
 1001 FORMAT('1',////////////////////3X,'**********************************
     1**************************************************************
     2'////15X,'THE FOLLOWING IS A CONTINUITY CHECK AT ',I5,' STEPS.'///
     4/3X,'************************************************************
     5************************************'/////6X,'AVERAGE OF THE CON
     6TINUITY CHECK IS ',F10.3/6X,'AVE REAL IS ',F10.3)
 1004 FORMAT(I4,10(1X,D11.4),I4///)
 1006 FORMAT( 8X,9(I2,10X),I2)
 1007 FORMAT('1', 8X,9(I2,10X),I2//)
      DO 10 J=2,N
      JL1=J-1
      DO 10 I=2,N
      IL1=I-1
      WT(I,J)=((U(I,J,2)-U(I,J,1)+U(I,JL1,2)-U(I,JL1,1)
     1  +U(IL1,J,2)-U(IL1,J,1)+U(IL1,JL1,2)-U(IL1,JL1,1))/X       +
     2         (V(I,J,2)+V(I,J,1)+V(I,JL1,2)+V(I,JL1,1)
     4  -V(IL1,J,2)-V(IL1,J,1)-V(IL1,JL1,2)-V(IL1,JL1,1))/DYD(IL1)+
     3         (W(I,J,2)+W(I,J,1)-W(I,JL1,2)-W(I,JL1,1)
     5  +W(IL1,J,2)+W(IL1,J,1)-W(IL1,JL1,2)-W(IL1,JL1,1))/DYD(JL1))*QUA
   10 CONTINUE
      SUM=0.D0
      SUMQ=0.D0
      DO 50 I=2,N
```

```
      DO 50 J=2,N
      SUMQ=SUMQ+WT(I,J)
   50 SUM=SUM+DABS(WT(I,J))
      DIV=(N-1)*(N-1)
      AVE=SUM/DIV
      AVEQ=SUMQ/DIV
      WRITE(6,1001) M,AVE,AVEQ
      DO 30 K=1,N,10
      L=K+9
      DO 30 KK=1,N,10
       WRITE(6,1007)(J,J=K,L)
      LL=((N+10-KK)/10)*10
      LLL=LL-9
      LLLL=LL+LLL
      DO 20 KKK=LLL,LL
      I=LLLL-KKK
   20 WRITE(6,1004) I,(WT(I,J),J=K,L),I
   30 WRITE(6,1006) (J,J=K,L)
      DO 40 J=2,N
      DO 40 I=2,N
   40 WT(I,J)=0.D0
      RETURN
      END
```

```
      BLOCK DATA
C   BLOCK DATA PROGRAM TO SET THE U & U TEMPORARY MATRIX TO ONE AND SET THE
C    V, W, W TEMPORARY, & V TEMPORARY MATRIXES TO ZERO
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2)
      COMMON /VELTEM/ UT(70,70),VT(70,70),WT(70,70)
      COMMON /YLENG/ Y(71)
      DATA U,V,W / 9800 * 1.D0 , 19600 * 0.D0 /
      DATA UT,VT,WT / 4900 * 1.D0 , 9800 * 0.D0 /
      DATA Y / 71 * 0.D0 /
      END
```

```
C   THIS PROGRAMME CALULATES THE VELOCITY COMPONENTS FOR FLOW ALONG A CORNER
C   FORMED BY TWO 90 DEGREE PLATES
      IMPLICIT REAL*8(A-H,O-Z)
      LOGICAL CHECK
      INTEGER BLTP,COUNT,CBL
      COMMON X,DX,VK,DEN,VD,UINF,PLI,LA
C   ***************************************************************************
C   LABEL COMMON USED TO ENSURE CORRECT TRANSFER OF VARIABLES BETWEEN MAIN AND
C   SUBROUTINES
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2),EM(70,70,2)
      COMMON /VELTEM/ UT(70,70),VT(70,70),WT(70,70)
      COMMON /DIMENS/ YM(70),YP(70),YMYP(70),YPYM(70),DYD(70)
      COMMON /COFF/ A(70),B(70),C(70),E(70),F(70)
      COMMON /YLENG/ Y(71),DY(70)
      DIMENSION RETB(5),D(70)
      DATA RETB / 2740.D0,3060.D0,3370.D0,3640.D0,0.D0 /
      DATA UINFD/ 1.D0 /
      DATA ZE,HALF,ONEE,TWOO/0.D0,.5D0,1.D0,2.D0/
 1001 FORMAT(7D11.4/D11.4,5I3)
 1002 FORMAT('1',20X,'THE VALUE OF JZ IS NOW > THE INDEX,',I4,' .')
 1003 FORMAT(15X,'$$$$SUBROUTINE CHECK HAS RUN THE 2-D BOUNDARY FURTHER
     1THAN THE INDEX WHICH IS ',I3,' .')
C   ***************************************************************************
C   X      --- STARTING POINT OF FLOW, INCHES
C   DX     --- STARTING STEP SIZE IN THE MAIN FLOW DIRECTION, INCHES
C   VK     --- KINEMATIC VISCOSITY
C   DEN    --- DENSITY
C   VD     --- DYNAMIC VISCOSITY
C   UINF   --- VELOCITY AT INFINITY IN THE X DIRECTION
C   PLI    --- PLATE LENGTH, INCHES
C   DY(1)  --- FIRST STEP SIZE IN Y AND Z DIRECTION, INCHES
C   BLTP   --- ESTIMATION OF BOUNDARY LAYER THICKNESS, NUMBER OF STEPS
```

```
C   CBL    --- ESTIMATION OF NUMBER OF STEPS IN 2-D FLOW
C   COUNT --- COUNT OF NUMBER OF STEPS TAKEN IN THE X DIRECTION
C   LA     --- PARAMETER FOR TYPE OF FLOW: LAMINAR=0, TURBULENT=1
C   INDEX --- SIZE OF MATRIX
C   ************************************************************************
       READ(5,1001) X,DX,VK,DEN,VD,UINF,PLI,DY(1),BLTP,CBL,COUNT,LA,INDEX
       CALL TITLE (DY(1))
        IFIRST=1
       JZ=CBL+BLTP
C   DO LOOP TO SET WALL VELOCITIES TO ZERO
       DO 9 I=1,INDEX
       UT(1,I)=ZE
       UT(I,1)=ZE
        U(I,1,1)=ZE
     9  U(1,I,1)=ZE
       REC=100.DO
       TWE=ONEE/12.DO
       CVT=VK/TWE
       CVTI=ONEE/CVT
       PLF=PLI*TWE
       FIXED=(UINF*PLF/VK)**1.5DO/(PLI*PLI)
       DISD=DSQRT(UINF/(VK*PLF))
       VELD=DSQRT(UINF*VK/PLF)
C   ************************************************************************
C   THE NEXT 21 STATEMENTS DETERMINE THE VARIABLE GRID SPACING IN THE Y & Z
C    DIRECTION AND NON-DIMENSIONALIZE THE COEFFICIENTS FOR THE TRIDIAGONAL MATRIX
C    AND THE GRID SPACING
       DO 1 I=1,5
       DY(I)=DY(1)
     1 Y(I+1)=Y(I)+DY(I)
       DO 2 I=6,20
       DY(I)=DY(I-1)*1.05DO
```

```
    2 Y(I+1)=Y(I)+DY(I)
      DO 3 I=21,40
      DY(I)=DY(I-1)*1.1D0
    3 Y(I+1)=Y(I)+DY(I)
      DO 4 I=41,INDEX
      DY(I)=DY(I-1)*1.15D0
    4 Y(I+1)=Y(I)+DY(I)
      DYD(1)=DY(1)*DISD*TWE
      DIL=Y(BLTP)
      DO 6 I=2,INDEX
      IL1=I-1
      DYD(I)=DY(I)*DISD*TWE
      SUMY=1.D0/(DYD(IL1)+DYD(I))
      YM(I)=SUMY/DYD(IL1)
      YP(I)=SUMY/DYD(I)
      YMYP(I)=HALF*SUMY*DYD(IL1)/DYD(I)
    6 YPYM(I)=HALF*SUMY*DYD(I)/DYD(IL1)
C   ***************************************************************************
C   NON-DIMENSIONALIZE THE STEP SIZE IN THE X DIRECTION
    7 DXD=DX/PLI
C   TRANSPOSE PREVIOUS VALUES OF THE VELOCTY COMPONENTS TO THE NEXT STEP AS
C    FIRST GUESS
      DO 11 I=1,INDEX
      DO 11 J=1,INDEX
       EM(I,J,2)=EM(I,J,1)
       U(I,J,2)=U(I,J,1)
       V(I,J,2)=V(I,J,1)
   11  W(I,J,2)=W(I,J,1)
C   ***************************************************************************
C   START SOLUTION OF 2-D BOUNDARY LAYER. BOUNDARY LAYER THICKNESS IS INCREASED
C    UNTIL U AT EDGE IS 0.99999
      ISTOP=BLTP-1
```

```
 101 DO 700 IT=1,2
     SHEAR=DSQRT(U(2,JZ,2)*UINF*CVT/DY(1))
     DO 6000 N=2,ISTOP
     RATIO=Y(N)/DIL
     CHECK=RATIO.GT..6D0
     IF(CHECK) GO TO 4000
     YPLUS=Y(N)*SHEAR*CVTI
     IF(YPLUS.GE.3125.D0) YPLUS=3125.D0
     TUR=.41D0*(1.D0-DEXP(-YPLUS*4.D-2))*Y(N)
     CHECK=RATIO.LT..1D0
     IF(CHECK) GO TO 5000
     R1=RATIO-.1D0
     R2=R1*R1
     TUR=TUR-1.53506D0*R2*DIL+2.75625D0*R1*R2*DIL-1.88425D0*R2*R2*DIL
     GO TO 5000
4000 TUR=0.089*DIL
5000 EM(N,JZ,2)=TUR*TUR*FIXED*2.D0*(YMYP(N)*(U(N+1,JZ,2)-U(N,JZ,2))+
    1YPYM(N)*(U(N,JZ,2)-U(N-1,JZ,2)))
6000 CONTINUE
     DELO=DIL
     DO 20 N=2,ISTOP
     NP1=N+1
     N1=N-1
     UB=    (U(N,JZ,2)+U(N,JZ,1))*HALF
     VB=    (V(N,JZ,2)+V(N,JZ,1))*HALF
     IF(V(N,JZ,1).EQ.ZE) VB=TWOO*VB
     SPE=DXD/UB
     R=VB*SPE*YMYP(N)
     RR=VB*SPE*YPYM(N)
     EVP2=(EM(NP1,JZ,2)+EM(N,JZ,2))*HALF
     EVP1=(EM(NP1,JZ,1)+EM(N,JZ,1))*HALF
     EVM2=(EM(N,JZ,2)+EM(N1,JZ,2))*HALF
```

```
      EVM1=(EM(N,JZ,1)+EM(N1,JZ,1))*HALF
      S1=(1.D0+EVP2)*SPE*YP(N)
      S2=(1.D0+EVM2)*SPE*YM(N)
      S4=(1.D0+EVP1)*SPE*YP(N)
      S5=(1.D0+EVM1)*SPE*YM(N)
      A(N)=-S2-RR
      B(N)=ONEE-R+RR+S1+S2
      C(N)=R-S1
   20 D(N)=(RR+S5)*U(N1,JZ,1)+(ONEE+R-RR-S4-S5)*U(N,JZ,1)+(S4-R)*U(NP1,J
     1Z,1)
      D(ISTOP)=D(ISTOP)-C(ISTOP)
      CALL SOLVE(D,ISTOP)
      U(ISTOP,JZ,2)=F(ISTOP)
      L=ISTOP-1
      DO 30 I=2,L
      K=(L+2)-I
   30 U(K,JZ,2)=F(K)-C(K)*U(K+1,JZ,2)/E(K)
      DO 40 N=2,BLTP
      N1=N-1
   40 V(N,JZ,2)=V(N1,JZ,2)-    DYD(N1)*(U(N,JZ,2)+U(N1,JZ,2)-U(N,JZ,1)-
     1U(N1,JZ,1))/DXD*HALF
      DO 901 N=2,BLTP
      M=N
      IF(U(N,JZ,2).GT..999D0) GO TO 902
  901 CONTINUE
  902 K=M-1
      DIL=Y(K)+DY(K)*(.999D0-U(K,JZ,2))/(U(M,JZ,2)-U(K,JZ,2))
      IF(IT.EQ.1) GO TO 700
      DELC=DIL/DELO
      IF(DELC.LT..99D0.OR.DELC.GT.1.01D0) GO TO 101
  700 CONTINUE
      DO 50 M=1,ISTOP
```

```
      IF(U(M,JZ,2).GT..99999D0.AND.U(M,JZ,2).LE.ONEE) GO TO 60
   50 CONTINUE
      ISTOP=BLTP
      BLTP=BLTP+1
      JZ=JZ+1
      IF(JZ.GT.INDEX) GO TO 1000
      GO TO 101
C   TRANSPOSE SOLUTION OF U, V, & W AT 2-D STATIONS TO ALL STATIONS WITHIN THE
C   2-D BOUNDARY LAYER AWAY FROM THE CORNER
   60 DO 70 K=JZ,INDEX
      DO 70 I=2,BLTP
      EM(I,K,2)=EM(I,JZ,2)
      EM(K,I,2)=EM(I,JZ,2)
      U(I,K,2)=U(I,JZ,2)
      U(K,I,2)=U(I,JZ,2)
      V(I,K,2)=V(I,JZ,2)
   70 W(K,I,2)=V(I,JZ,2)
      LUS=BLTP+1
      DO 110 K=LUS,INDEX
      DO 10 I=LUS,INDEX
      V(I,K,2)=V(BLTP,JZ,2)
      W(K,I,2)=V(BLTP,JZ,2)
      IF(K.EQ.I) GO TO 110
   10 CONTINUE
  110 CONTINUE
      DO 80 I=2,BLTP
      ID=I
      IF(U(I,JZ,2).GT..99D0) GO TO 100
   80 CONTINUE
  100 CALL VELSOL(JZ,BLTP,UINFD,DXD,CVT,CVTI,UINF,FIXED,DIL)
      IF(COUNT.LT.20) GO TO 120
      CALL CHECKK(BLTP,JZ,INC,INDEX,IQUIT,ID)
```

```
      IF(IQUIT.EQ.INDEX) GO TO 2000
      IF(INC.EQ.0) GO TO 100
      IF(JZ.EQ.INDEX) GO TO 2000
  120 X=X+DX
C  ***************************************************************************
C  CONTROLLED OUTPUT AT DESIRED REYNOLDS NUMBERS. (ANY OTHER TYPE OF CONTROL
C   IS POSSIBLE)
      DELSTA=0.0D0
      THETA=0.0D0
      DO 900 NT=2,I
      NTL1=NT-1
      FX1=ONEE-U(NTL1,JZ,2)
      FX2=ONEE-U(NT,JZ,2)
      DELSTA=DELSTA+.5D0*(FX1+FX2)*DY(NTL1)
  900 THETA=THETA+HALF*(U(NTL1,JZ,2)*FX1+U(NT,JZ,2)*FX2)*DY(NTL1)
      DELSTA=DELSTA+.5*(FX2+1.D-1)*(DIL-Y(I))
      THETA=THETA+HALF*(U(I,JZ,2)*FX2+.99D-2)*(DIL-Y(I))
      RE=UINF*X*TWE/VK
      RET=UINF*THETA*CVTI
      IF(RE.GT.REC) CALL OUTPUT (JZ,BLTP,ID,COUNT,UINF,X,CVTI,RE,RET)
      IF(RE.GT.REC) CALL CONTIN (JZ,COUNT,DXD)
      IF(RE.GT.REC) REC= TWOO*REC
       IF(RET.LT.RE TB(IFIRST)) GO TO 111
      IF(RET.GT.RE TB(IFIRST)) CALL OUTPUT (JZ,BLTP,ID,COUNT,UINF,X,
     1CVTI,RE,RET)
      IF(RET.GT.RE TB(IFIRST)) CALL CONTIN (JZ,COUNT,DXD)
      IF(RET.GT.RE TB(IFIRST)) IFIRST=IFIRST+1
      IF(RETB(IFIRST).LT.1.DO) STOP
  111 IF(COUNT.GT.20) DX=1.1D0*DX
      IF(DX.GT.Y(ID)) DX=Y(ID)
C  DO LOOP TO STORE NEW VALUES OF U, V, & W IN OLD POSITION SO THAT THE
C   CALCULATIONS CAN ADVANCE TO THE NEXT X STEP
```

```
      DO 8 I=1,INDEX
      DO 8 J=1,INDEX
      EM(I,J,1)=EM(I,J,2)
      U(I,J,1)=U(I,J,2)
      V(I,J,1)=V(I,J,2)
    8 W(I,J,1)=W(I,J,2)
      COUNT=COUNT+1
      IF(COUNT.LE.1000) GO TO 7
 1000 WRITE(6,1002) INDEX
                    CALL OUTPUT (JZ,BLTP,ID,COUNT,UINF,X,CVTI,RE,RET)
      STOP
 2000 WRITE(6,1003) INDEX
                    CALL OUTPUT (JZ,BLTP,ID,COUNT,UINF,X,CVTI,RE,RET)
      STOP
      END
```

```
      SUBROUTINE SOLVE (D,K)
C   SUBPROGRAM TO CALCULATE THE SOLUTION TO THE TRIDIAGONAL MATRIX
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /COFF/ A(70),B(70),C(70),E(70),F(70)
      DIMENSION D(70)
      E(2)=B(2)
      F(2)=D(2)/B(2)
      DO 1 I=3,K
       I1=I-1
      E(I)=B(I)-A(I)*C(I1)/E(I1)
    1 F(I)=(D(I)-A(I)*F(I1))/E(I)
      RETURN
      END
```

```
      SUBROUTINE VELSOL(JZ,BLTP,UINFD,DXD,CVT,CVTI,UINF,FIXED,DIL)
C   SUBPROGRAM TO SOLVE THE X, Y, & Z MOMENTUM EQUATIONS
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER  BLTP
      DIMENSION D(70) ,DV(70),DW(70)
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2),EM(70,70,2)
      COMMON /VELTEM/ UT(70,70),VT(70,70),WT(70,70)
      COMMON /DIMENS/ YM(70),YP(70),YMYP(70),YPYM(70),DYD(70)
      COMMON /COFF/ A(70),B(70),C(70),E(70),F(70)
      COMMON /YLENG/ Y(71),DY(70)
      DATA ZE,QU,HALF,ONEE,TWOO/0.D0,.25D0,.5D0,1.D0,2.D0 /
      JZZ=JZ-1
C   MOMENTUM EQUATIONS ARE ITERATED THREE TIMES
      DO 9 IT=1,3
      CALL SETBOU (BLTP,JZ)
      CALL EDDY (JZ,BLTP,CVT,CVTI,DIL,UINF,FIXED)
      K=JZZ
C   STARTING DO LOOP FOR EXPLICIT STRIPPING IN THE Z DIRECTION
      DO 4 J=2,JZZ
      JP1=J+1
      JM1=J-1
      R3=YPYM(J)
      R4=YMYP(J)
      R5=R4-R3
      IF(J.GT.BLTP) K=K-1
      KP1=K+1
      L=K-1
C   STARTING DO LOOP FOR IMPLICIT STRIPPING IN THE Y DIRECTION
      DO 1 I=2,K
      IP=I+1
      IL=I-1
      UB=    (U(I,J,2)+U(I,J,1))*HALF
```

```
      VB=    (V(I,J,2)+V(I,J,1))*HALF
      WB=    (W(I,J,2)+W(I,J,1))*HALF
      IF(V(I,J,1).EQ.ZE) VB=TWOO*VB
      IF(W(I,J,1).EQ.ZE) WB=TWOO*WB
      SPE=DXD/UB
      R1=VB*SPE*YPYM(I)
      R2=VB*SPE*YMYP(I)
      EMAP=(EM(IP,J,2)+EM(I,J,2)+EM(IP,J,1)+EM(I,J,1))*QU
      EMAM=(EM(IL,J,2)+EM(I,J,2)+EM(IL,J,1)+EM(I,J,1))*QU
      S1=SPE*YM(I)*(ONEE+EMAM)
      S2=SPE*YP(I)*(ONEE+EMAP)
C  COEFFICIENTS FOR THE TRIDIAGONAL MATRIX FOR EACH EQUATION
      A(I)=-R1-S1
      B(I)=   R1-R2+S2+S1+ONEE
      C(I)=R2-S2
      EMAM=(EM(I,JM1,1)+EM(I,J,1))*HALF
      EMAP=(EM(I,JP1,1)+EM(I,J,1))*HALF
      S3=YP(J)*(ONEE+EMAP)
      S4=YM(J)*(ONEE+EMAM)
      S5=S3+S4
      P=(WB*R3+S4)*SPE
      Q=ONEE+(WB*R5-S5)*SPE
      T=(S3-WB*R4)*SPE
      D (I)=P*U(I,JM1,1)+Q*U(I,J,1)+T*U(I,JP1,1)
      DV(I)=P*V(I,JM1,1)+Q*V(I,J,1)+T*V(I,JP1,1)
    1 DW(I)=P*W(I,JM1,1)+Q*W(I,J,1)+T*W(I,JP1,1)
      D (2)=D (2)-A(2)*U(1,JM1,1)
      DV(2)=DV(2)-A(2)*V(1,JM1,1)
      DW(2)=DW(2)-A(2)*W(1,JM1,1)
      D (K)=D (K)-C(K)*UT(KP1,J)
      DV(K)=DV(K)-C(K)*VT(KP1,J)
      DW(K)=DW(K)-C(K)*WT(KP1,J)
```

```
      CALL SOLVE (D,K)
C  SOLVE FOR TEMPORARY U VALUES
      UT( K ,J)=F( K )
      DO 2 M=2,L
      N=(L+2)-M
    2 UT(N,J)=F(N)-C(N)*UT(N+1,J)/E(N)
C SOLVE FOR TEMPORARY V VALUES
      CALL SOLVE (DV,K)
      VT( K ,J)=F( K )
      DO 3 M=2,L
      N=(L+2)-M
    3 VT(N,J)=F(N)-C(N)*VT(N+1,J)/E(N)
C SOLVE FOR TEMPORARY W VALUES
      CALL SOLVE (DW,K)
      WT( K ,J)=F( K )
      DO 4 M=2,L
      N=(L+2)-M
    4 WT(N,J)=F(N)-C(N)*WT(N+1,J)/E(N)
      K=JZZ
C   STARTING DO LOOP FOR EXPLICIT STRIPPING IN Y DIRECTION
      DO 8  I=2,JZZ
      IM1=I-1
      IP1=I+1
      IF(I.GT.BLTP)K=K-1
      KP1=K+1
      L=K-1
      R3=YPYM(I)
      R4=YMYP(I)
      R5=R4-R3
C   STARTING DO LOOP FOR IMPLICIT STRIPPING IN Z DIRECTION
      DO 5 J=2,K
      JL=J-1
```

```
      JP=J+1
      UB=    (U(I,J,2)+U(I,J,1))*HALF
      VB=    (V(I,J,2)+V(I,J,1))*HALF
      WB=    (W(I,J,2)+W(I,J,1))*HALF
      IF(V(I,J,1).EQ.ZE) VB=TWOO*VB
      IF(W(I,J,1).EQ.ZE) WB=TWOO*WB
      SPE=DXD/UB
      R1=WB*SPE*YPYM(J)
      R2=WB*SPE*YMYP(J)
      EMAM=(EM(I,JL,2)+EM(I,J,2))*HALF
      EMAP=(EM(I,JP,2)+EM(I,J,2))*HALF
      S1=SPE*YM(J)*(ONEE+EMAM)
      S2=SPE*YP(J)*(ONEE+EMAP)
C   COEFFICIENTS FOR THE TRIDIAGONAL MATRIX
      A(J)=-R1-S1
      B(J)=   R1-R2+S1+S2+ONEE
      C(J)=R2-S2
      EMAM=(EM(IM1,J,2)+EM(I,J,2)+EM(IM1,J,1)+EM(I,J,1))*QU
      EMAP=(EM(IP1,J,2)+EM(I,J,2)+EM(IP1,J,1)+EM(I,J,1))*QU
      S3=YP(I)*(ONEE+EMAP)
      S4=YM(I)*(ONEE+EMAM)
      S5=S3+S4
      P=(VB*R3+S4)*SPE
      Q=ONEE+(VB*R5-S5)*SPE
      T=(S3-VB*R4)*SPE
      D (J)=P*UT(IM1,J)+Q*UT(I,J)+T*UT(IP1,J)
      DV(J)=P*VT(IM1,J)+Q*VT(I,J)+T*VT(IP1,J)
    5 DW(J)=P*WT(IM1,J)+Q*WT(I,J)+T*WT(IP1,J)
      D (K)=D (K)-C(K)*U(I,KP1,2)
      DV(K)=DV(K)-C(K)*V(I,KP1,2)
      DW(K)=DW(K)-C(K)*W(I,KP1,2)
C   SOLVE FOR NEW U VALUES
```

```
      CALL SOLVE(D,K)
      U(I, K ,2)=F( K )
      DO 6 M=2,L
      N=(L+2)-M
    6 U(I,N,2)=F(N)-C(N)*U(I,N+1,2)/E(N)
C   SOLVE FOR NEW V VALUES
      CALL SOLVE(DV,K)
      V(I, K ,2)=F( K )
      DO 7 M=2,L
      N=(L+2)-M
    7 V(I,N,2)=F(N)-C(N)*V(I,N+1,2)/E(N)
C   SOLVE FOR NEW W VALUES
      CALL SOLVE(DW,K)
      W(I, K ,2)=F( K )
      DO 8 M=2,L
      N=(L+2)-M
    8 W(I,N,2)=F(N)-C(N)*W(I,N+1,2)/E(N)
    9 CONTINUE
      RETURN
      END
      SUBROUTINE SETBOU (K,L)
C   SUBPROGRAM WHICH FIXES THE TEMPORARY U, V, & W ON 2-D BOUNDARIES AT AVERAGE
C    OF LAST STEP AND NEW STEP U, V, & W
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2),EM(70,70,2)
      COMMON /VELTEM/ UT(70,70),VT(70,70),WT(70,70)
      HALF=.5D0
      DO 10 J=2,K
      UT(L,J)=(U(L,J,2)+U(L,J,1))*HALF
      UT(J,L)=(U(J,L,2)+U(J,L,1))*HALF
      VT(J,L)=(V(J,L,2)+V(J,L,1))*HALF
   10 WT(L,J)=(W(L,J,2)+W(L,J,1))*HALF
```

```
      KK=K+1
      LL=L-1
      N=L
      DO 20 J=KK,LL
      N=N-1
      VT(N,J)=(V(N,J,2)+V(N,J,1))*HALF
   20 WT(N,J)=(W(N,J,2)+W(N,J,1))*HALF
      RETURN
      END
```

```
      SUBROUTINE CHECKK(L,J,K,M,N,ID)
C   THIS SUBROUTINE CHECKS THE SLOPE OF THE U ISOVELS AT T&& 00--&& &
C    BOUNDARY.
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2),EM(70,70,2)
      COMMON /VELTEM/ UT(70,70),VT(70,70),WT(70,70)
      COMMON /DIMENS/ YM(70),YP(70),YMYP(70),YPYM(70),DYD(70)
      JP=J+1
      JL=J-1
      DY=DYD(J)+DYD(JL)
      SLOPE=DABS((U(ID+1,J,2)-U(ID-1,J,2))/(DYD(ID)+DYD(ID-1)))
      DO 10 I = 2,ID
      DEVY=DABS((U(I,JP,2)-U(I,JL,2))/DY)
      IF(DEVY.GT.SLOPE) GO TO 20
   10 CONTINUE
      GO TO 30
   20 J=J+1
      K=0
      N=0
      IF(J.GT.M) N=M
      RETURN
   30 K=1
      RETURN
      END
```

```
      SUBROUTINE EDDY (JL,JK,CVT,CVTI,DIL,UINF,FIXED)
C   THIS SUBROUTINE CALCULATES THE EDDY-VISCOSITY AT EACH NODAL POSITION
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2),EM(70,70,2)
      COMMON /DIMENS/ YM(70),YP(70),YMYP(70),YPYM(70),DYD(70)
      COMMON /YLENG/ Y(71),DY(70)
      LOGICAL CHECK
      RAD=Y(JL)-DIL
      ISTOP=JL-1
      KK=ISTOP
      II=2
      SHEAR=DSQRT(U(2,JL,2)*UINF*CVT/DY(1))
      DO 6001 J=2,ISTOP
      JM=J-1
      JP=J+1
      IF(J.GT.JK) KK=KK-1
      DO 6000 I=II,KK
      IM=I-1
       IP=I+1
      UY=4.DO*(YMYP(I)*(U(IP,J,2)-U(I,J,2))+YPYM(I)*(U(I,J,2)-U(IM,J,2))
     1)**2
      UZ=4.DO*(YMYP(J)*(U(I,JP,2)-U(I,J,2))+YPYM(J)*(U(I,J,2)-U(I,JM,2))
     1)**2
      T=DSQRT(UZ+UY)
      SLOPE=(Y(JL)-Y(I))/(Y(JL)-Y(J))
      BINC=Y(I)-SLOPE*Y(J)
      YLEN=DSQRT((Y(JL)-BINC)**2+Y(JL)*Y(JL))
      DELCL=YLEN-RAD
      YPOS=DSQRT((Y(I)-BINC)**2+Y(J)*Y(J))
      RATIO=YPOS/DELCL
      CHECK=RATIO.GE..6DO
      IF(CHECK) GO TO 4000
```

```
      YPLUS=YPOS*SHEAR*CVTI
      IF(YPLUS.GE.3125.D0) YPLUS=3125.D0
      TUR=.41D0*(1.D0-DEXP(-YPLUS*4.D-2))*YPOS
      CHECK=RATIO.LT..1D0
      IF(CHECK) GO TO 5000
      R=RATIO-.1D0
      RR=R*R
      TUR=TUR-1.53506D0*RR*DELCL+2.75625D0*R*RR*DELCL-1.88425D0*RR*RR*DE
     1LCL
      GO TO 5000
 4000 TUR=0.089*DELCL
 5000 EM(I,J,2)=TUR*TUR*FIXED*T
 6000 CONTINUE
      II=II+1
      IF(II.GT.KK) GO TO 7000
 6001 CONTINUE
 7000 KK=ISTOP
      JJ=3
      DO 8001 I=2,ISTOP
      IF(I.GT.JK) KK=KK-1
      DO 8000 J=JJ,KK
 8000 EM(I,J,2)=EM(J,I,2)
      JJ=JJ+1
      IF(JJ.GT.KK) GO TO 9000
 8001 CONTINUE
 9000 RETURN
      END
```

```
      SUBROUTINE TITLE (DY)
C   SUBPROGRAM TO PRINT THE PARAMETERS OF THE FLOW
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON X,DX,VK,DEN,VD,UINF,PLI,LA
 1001 FORMAT('1',15X,'PHYSICAL PARAMETERS OF PLATE:'///20X,'THE FLOW STARTS AT '
     1RTS AT ',F8.3,' INCHES FROM THE LEADING EDGE.'//20X,'THE PLATE IS
     2',F8.3,' INCHES IN DEPTH.'//20X,'THE INITIAL STARTING STEP SIZE IN
     3 THE X DIRECTION IS ',F6.4,' INCHES'//20X,'THE STEP SIZE IN THE Y
     4AND Z DIRECTION IS ',F6.4,' INCHES.'////15X,'PARAMETERS OF THE FLU
     5ID:'///20X,'THE VELOCITY OF THE MAIN STREAM FLOW IN THE X DIRECTIO
     6N IS ',F8.1,' FT/SEC'//20X,'KINEMATIC VISCOSITY IS ',F13.9,' SQ FT
     7/SEC'//20X,'DYNAMIC VISCOSITY IS ',F13.9,'LBF-SEC/SQ FT'//20X,'DENSI
     8SITY IS ',F13.9,'LBF-SQ SEC/FT TO FOURTH')
 1002 FORMAT(/  20X,'THE FLOW IS LAMINAR. *****')
 1003 FORMAT(/  15X,'THE FLOW IS TURBULENT. *****')
 1004 FORMAT(///20X,'THE COMPONENTS OF VELOCITY ARE U IN THE X DIRECTION
     1, V IN THE Y DIRECTION AND W IN THE Z DIRECTION.'/20X,'FOR EACH I
     2AND J POSITION AT THE INDICATED X POSITION, VALUES FOR U, V, & W A
     3RE PRINTED. THE FIRST VALUE IS U,'/20X,'THE SECOND IS V, AND THE T
     4HIRD IS W. POSITIVE U IS INTO THE PAPER, POSITIVE V IS UP THE PAPE
     5R, & POSITIVE W IS'/20X,'IS TO THE RIGHT.')
      WRITE(6,1001) X,PLI,DX,DY,UINF,VK,VD,DEN
      IF(LA.EQ.0) WRITE(6,1002)
      IF(LA.EQ.1) WRITE(6,1003)
      WRITE(6,1004)
      RETURN
      END
```

```
      SUBROUTINE OUTPUT(JZ,BLTP,ID,COUNT,UI,X,CVTI,RE,RET)
C   SUBPROGRAM TO PRINT U, V, & W AT GIVE REYNOLDS NUMBERS
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER BLTP,COUNT
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2),EM(70,70,2)
      COMMON /YLENG/ Y(71),DY(70)
 1001 FORMAT('1'//////////20X,'AT ',I5,' STEPS DOWN THE PLATE OR ',F8.4
     1,'INCHES FROM THE LEADING EDGE,'/20X,'THE REYNOLDS NUMBER HAS A VA
     2LUE OF ',F10.0/20X,'REYNOLDS NO. BASED ON THETA IS ',F10.2)
 1002 FORMAT(////20X,'THE 2-D BOUNDARY LAYER AT THIS POSITION IS ',F12.8
     1,' INCHES.'/20X,'THE POSITION WHERE U/UINF IS > 0.99999 IS ',F12.8
     2,' INCHES.')
 1003 FORMAT(4X,10(1X,D11.4))
 1004 FORMAT(I4,10(1X,D11.4),I4)
 1005 FORMAT(4X,10(1X,D11.4)//)
 1006 FORMAT( 8X,9(I2,10X),I2/)
 1007 FORMAT('1', 8X,9(I2,10X),I2/)
 1008 FORMAT(////20X,'THE POSITION AT WHICH THE FLOW IS 2-D IS ',F12.8,'
     1 INCHES FROM THE CORNER.'////20X,'THE INDEX FOR THE ABOVE POSTION
     2IS ',I3,' .'//20X,'THE 2-D INDEX IS ',I3,' .')
      DIMENSION ETA(70)
      DO 4 N=2,BLTP
      I=N-1
      IF(U(I,JZ,2).GT..99D0) GO TO 5
    4 CONTINUE
    5 DEL=Y(I)+DY(I)*(.99D0-U(I,JZ,2))/(U(I+1,JZ,2)-U(I,JZ,2))
      IN=(JZ+9)/10*10
      DO 3 K=1,IN
    3 ETA(K)=Y(K)
      WRITE(6,1001) COUNT,X,RE,RET
      WRITE(6,1002) DEL  ,Y(BLTP)
      WRITE(6,1008) Y(JZ),JZ,BLTP
```

```
      DO 2 M=1,JZ,10
      N=M+9
      DO 2 K=1,JZ,10
      WRITE(6,1007)(J,J=M,N)
       L=((JZ+10-K)/10)*10
      KK=L-9
      LL=L+KK
      DO 1 II=KK,L
      I=LL-II
      WRITE(6,1003)(U(I,J,2),J=M,N)
      WRITE(6,1004)I,(V(I,J,2),J=M,N),I
    1 WRITE(6,1005)(W(I,J,2),J=M,N)
      WRITE(6,1006)(J,J=M,N)
    2 WRITE(6,1003) (ETA(J),J=M,N)
      RETURN
      END
```

```
      SUBROUTINE CONTIN (N,M,X)
C   THIS SUBROUTINE DETERMINES THE VALUE OF THE CONTINUITY EQUATION. IT ALSO
C    EQUATION FOR THE REGION OF THE CORNER
C    DETERMINES THE ABSOLUTE AVERAGE AND REAL AVERAGE OF THE CONTINUITY
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2),EM(70,70,2)
      COMMON /VELTEM/ UT(70,70),VT(70,70),WT(70,70)
      COMMON /DIMENS/ YM(70),YP(70),YMYP(70),YPYM(70),DYD(70)
      DATA QUA / .25D0 /
 1001 FORMAT('1',////////////////////3X,'********************************
     1************************************************************
     2'////15X,'THE FOLLOWING IS A CONTINUITY CHECK AT ',I5,' STEPS.'///
     4/3X,'************************************************************
     5**************************************'/////6X,'AVERAGE OF THE CON
     6TINUITY CHECK IS ',F10.3/6X,'AVE REAL IS ',F10.3)
 1004 FORMAT(I4,10(1X,D11.4),I4///)
 1006 FORMAT( 8X,9(I2,10X),I2)
 1007 FORMAT('1', 8X,9(I2,10X),I2//)
      DO 10 J=2,N
      JL1=J-1
      DO 10 I=2,N
      IL1=I-1
      WT(I,J)=((U(I,J,2)-U(I,J,1)+U(I,JL1,2)-U(I,JL1,1)
     1  +U(IL1,J,2)-U(IL1,J,1)+U(IL1,JL1,2)-U(IL1,JL1,1))/X       +
     2          (V(I,J,2)+V(I,J,1)+V(I,JL1,2)+V(I,JL1,1)
     4  -V(IL1,J,2)-V(IL1,J,1)-V(IL1,JL1,2)-V(IL1,JL1,1))/DYD(IL1)+
     3          (W(I,J,2)+W(I,J,1)-W(I,JL1,2)-W(I,JL1,1)
     5  +W(IL1,J,2)+W(IL1,J,1)-W(IL1,JL1,2)-W(IL1,JL1,1))/DYD(JL1))*QUA
   10 CONTINUE
      SUM=0.D0
      SUMQ=0.D0
      DO 50 I=2,N
```

```
      DO 50 J=2,N
      SUMQ=SUMQ+WT(I,J)
   50 SUM=SUM+DABS(WT(I,J))
      DIV=(N-1)*(N-1)
      AVE=SUM/DIV
      AVEQ=SUMQ/DIV
      WRITE(6,1001) M,AVE,AVEQ
      DO 30 K=1,N,10
      L=K+9
      DO 30 KK=1,N,10
       WRITE(6,1007)(J,J=K,L)
      LL=((N+10-KK)/10)*10
      LLL=LL-9
      LLLL=LL+LLL
      DO 20 KKK=LLL,LL
      I=LLLL-KKK
   20 WRITE(6,1004) I,(WT(I,J),J=K,L),I
   30 WRITE(6,1006) (J,J=K,L)
      DO 40 J=2,N
      DO 40 I=2,N
   40 WT(I,J)=0.D0
      RETURN
      END
```

```
      BLOCK DATA
C   BLOCK DATA PROGRAM TO SET THE U & U TEMPORARY MATRIX TO ONE AND SET THE
C    V, W, W TEMPORARY, & V TEMPORARY MATRIXES TO ZERO
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /VEL/ U(70,70,2),V(70,70,2),W(70,70,2),EM(70,70,2)
      COMMON /VELTEM/ UT(70,70),VT(70,70),WT(70,70)
      COMMON /YLENG/ Y(71),DY(70)
      DATA U,V,W,EM/9800* 1.D0 , 29400 * 0.D0 /
      DATA UT,VT,WT / 4900 * 1.D0 , 9800 * 0.D0 /
      DATA Y / 71 * 0.D0 /
      END
```

The vita has been removed from the scanned document

AN IMPLICIT NUMERICAL SOLUTION FOR THE LAMINAR AND TURBULENT FLOW OF AN INCOMPRESSIBLE FLUID ALONG THE AXIS OF A 90-DEGREE CORNER

by

David Tillman Klinksiek

(ABSTRACT)

A method of solving the equations for the three-dimensional, incompressible laminar and turbulent flow along the intersection of two planes at ninety-degrees has been developed. The Alternating Direction Implicit (ADI) finite-difference method was applied for both types of flow. The turbulent stresses in the corner region were modeled with an eddy-viscosity model which was obtained from mixing length theory. The method was compared with other types of solutions for the laminar case and good agreement was achieved. For the turbulent case, the method was compared with experimental data and good agreement was obtained.

The three momentum equations were solved simultaneously and the continuity equation was used to verify the method. The method appeared to predict the velocity components correctly since the continuity equation residual approached zero as the solution proceeded from the leading edge in the mainstream flow direction.

No analysis was presented for the convergence or stability of the finite-difference equations and no convergence or stability problems

were encountered when the finite-difference equations were solved. The method predicted symmetry about the corner bisector in all cases and gave the expected u-velocity profile along the bisector for both the laminar and turbulent cases.