1986

FINITE-DIFFERENCE ALGORITHMS FOR INVISCID, INCOMPRESSIBLE FLOW OVER AN ARBITRARY SYMMETRIC PROFILE.

GEORGE WILLIAM. GROSSMAN
University of Windsor

FINITE-DIFFERENCE ALGORITHMS FOR INVISCID, INCOMPRESSIBLE FLOW OVER AN ARBITRARY SYMMETRIC PROFILE

by

© George William Grossman

A Dissertation
submitted to the
Faculty of Graduate Studies and Research
through the Department of
Mathematics and Statistics in Partial Fulfillment
of the requirements for the Degree
of Doctor of Philosophy at
The University of Windsor

Windsor, Ontario, Canada
1986

ISBN 0-315-29307-1

ABSTRACT

FINITE-DIFFERENCE ALGORITHMS FOR INVISCID, INCOMPRESSIBLE FLOW OVER AN ARBITRARY SYMMETRIC PROFILE

by

George William Grossman

This thesis studies steady, two dimensional flow of an inviscid, incompressible fluid over an arbitrary symmetric profile. Flows with zero and variable vorticity are considered. In the present work a numerical algorithm is given for a class of flows that can also be solved by perturbation techniques. However, reliable solutions by the perturbation technique, especially in the case of rotational flows, require complicated analytical methods even in the case of the circle. Thus, one of the goals of this thesis is to provide a fast and efficient algorithm from which a solution to several standard problems can be obtained with less effort.

The equations of motion based on a transformation of coordinate systems are derived. The approach is new in that the computational domain consists of the streamlines $\psi(x,y) = $ constant and an arbitrary family of curves $\phi(x,y) = $ constant such that the (ϕ,ψ) coordinate system forms a curvilinear net. To solve the flow the transformed equations are simplified based on the flow assumptions. Boundary conditions of the mixed type are then applied to the computational domain. Results are presented for several aerodynamic profiles and compared with those obtained by other methods. The proposed method is found to be fast, efficient and reliable. Accurate results can be obtained with a minimum of numerical calculation.

A stability analysis of the ADI (Alternating-Direction-Implicit) iteration method is carried out, based on a Fourier series method. A new equation for the error is obtained. It is found possible to obtain a precise interval where convergence is optimized for a certain class of elliptic partial differential equations.

To My Parents

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Dr. R.M. Barron, for his guidance and support during my years as a Ph.D. student. I am especially thankful for the patience and wisdom shown by him during the writing of this dissertation. I am appreciative of the kind advice and encouragement of Drs. A.C. Smith, O.P. Chandna, P.N. Kaloni and K.L. Duggal.

Special thanks are extended to Dr. Y.S. Wong of the Mathematics Department of the University of Alberta, Drs. G.W. Rankin and K. Sridhar of the Mechanical Engineering Department of the University of Windsor for their constructive criticism of the dissertation. I am also grateful to the Department of Mathematics for providing a modern computer typing facility.

LIST OF SYMBOLS

ϵ	grid generation curvilinear coordinate
η	grid generation curvilinear coordinate
x	cartesian coordinate
y	cartesian coordinate
Ψ	curvilinear coordinate
Φ	curvilinear coordinate
α	local angle of inclination of streamline with x-axis
ρ	constant fluid density, ADI parameter
μ	constant viscosity
$\underline{\nabla}$	del operator
∇^2	Laplace's operator
Δ	Laplace's operator, increment, e.g., Δx
ψ	stream function, curvilinear coordinate
ϕ	curvilinear coordinate
θ	angle between coordinate curves in (ϕ,ψ) net (Ch.5)
$\Gamma^k_{i,j}$	Christoffel symbols; i, j, k are integers
u_∞	constant flow speed at infinity
x_{LE}	value of x at the leading edge
x_{TE}	value of x at the trailing edge
κ	curvature
r	radial component of polar coordinate
θ	angular component of polar coordinate (Chapter 2)
$D\phi$	grid spacing in ϕ direction
$D\psi$	grid spacing in ψ direction

κ	constant
u	velocity component in x direction
v	velocity component in y direction
p	pressure function
q	speed function
h	energy function
ω	vorticity function
E, F, G	coefficients of First Fundamental Form
J	Jacobian
$W = \pm J$	
T	$\frac{\ln E}{2}$
U	$\frac{\ln G}{2}$
L	characteristic length

LIST OF TABLES

Table I:

New Algorithm. Irrotational, inviscid, incompressible flow (I.I.I.). Values of speed on surface of NACA-0012-64 airfoil using present algorithm are given in column 3. ϕ values are presented in column 1 while x values are given in column 2 (cf. Chapter 3). Note all speed values are measured in non-dimensional variables. The title of all tables is underlined.

Table II:

Theodorsen's Results (I.I.I.). Values of speed on surface of NACA-0012-64 airfoil using Theodorsen's results (conformal mapping) are presented in column 2. x values as a percentage of the airfoil chord c are given in column 1 (cf. Chapter 1).

Table III:

Grid Generation Results (I.I.I.). Values of speed on surface of NACA-0012-64 airfoil using grid generation techniques are given in column 2. x values are in column 1 (cf. Chapter 1).

Table IV:

Parabolic Shear for Circle of Radius .5. Values of speed on surface of circle $y=(.25-x^2)^{1/2}$ for parabolic shear flow. The J value corresponds to grid point location, i.e., (J,K). The second column represents Van Dyke's second order perturbation solution. The third column represents the angle of inclination of the velocity vector to the surface of the airfoil measured in radians. The fourth column is the solution by the present method. Table IV corresponds with algorithm A (cf. Chapter4).

Table V:

Parabolic Shear for Circle of Radius .5. Columns have same meaning as in Table IV except column 4 represents a solution by present method with slightly different boundary conditions than Table IV. Table V corresponds with algorithm B (cf. Chapter 4).

Table VI:

Parabolic Shear for Circle of Radius .5. Columns have same meaning as Table IV. Table VI corresponds with algorithm C (cf. Chapter 4).

LIST OF FIGURES AND LEGENDS

LIST OF APPENDICES

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION

1.1 SUMMARY OF OTHER APPROACHES

Several numerical and analytic techniques are available to solve the problem under consideration. Numerical grid generation (N.G.G.) and conformal mapping involve mappings between coordinate systems. Finite element methods require the construction of a polygonal mesh and the minimization of certain integrals. The motivation of these techniques is the construction of a grid to remove difficulties involving derivative boundary conditions and curved boundaries. If a grid can be made orthogonal on the boundary then the incorporation of normal conditions is facilitated.

The algorithm of this thesis closely follows some of the principles of numerical grid generation, i.e., the use of boundary conforming coordinates, the solution of elliptic partial differential equations and the generation of a curvilinear grid system. However, in the present work physical signifigance has been attached to the coordinate curves $\psi(x,y)$ = constant, and $\phi(x,y)$ = constant as opposed to the ξ, η system used in N.G.G., which is usually independent of any flow considerations. This comprises the main difference in the two methods.

A comprehensive text by J. F. Thompson and others [1] describes in detail the basic methods of generating an arbitrary curvilinear grid. The techniques include elliptic, hyperbolic, parabolic, algebraic and adaptive grid generation. This text also contains a discussion of conformal mapping and a means of constructing orthogonal grids. Through numerical experimentation it can easily be seen that the physical solution depends with varying degree on grid sizes, spacing, and the type of grid coordinate system chosen to represent the particular problem at hand. Clearly the first two properties will always cause some degree of uncertainty in any numerical solution of finite difference equations. The last property may be used to advantage to eliminate inherent difficulties in choosing a grid if one of the coordinate curves is a curve with physical signifigance for the flow under consideration, for example a streamline or pressure curve. This choice may cause initial difficulties in the sense that certain boundary conditions are difficult to implement, but the advantages, it is found, far outweigh the disadvantages once the problem is understood. A degree of arbitrariness is still left in the choice of the other coordinate but this can usually be determined so as to be consistent with flow characteristics.

There are many different techniques in N.G.G.. Elliptic grid generation is one of the more popular ones partially owing to its ease and reliability in use [2]. Elliptic grid generation has been established in order that arbitrary coordinate systems can be transformed in an orderly and efficient manner. Usually the physical region is described in cartesian coordinates and the transformed region in an arbitrary (ξ,η) coordinate system. A desirable feature of this approach is that each boundary of the physical region can be made to conform with a coordinate line of the transformed region. Poisson type equations of the form

$$\xi_{xx} + \xi_{yy} = P(\xi,\eta) \quad , \quad \eta_{xx} + \eta_{yy} = Q(\xi,\eta)$$

are considered. These equations may be inverted so as to solve for x,y as functions of ξ,η; x and y satisfy elliptic P.D.E.'s of the form [1]

$$G(x_{\xi\xi} + Px_{\xi}) - 2Fx_{\xi\eta} + E(x_{\eta\eta} + Qx_{\eta}) = 0 \qquad (1.1.1a)$$

$$G(y_{\xi\xi} + Py_{\xi}) - 2Fy_{\xi\eta} + E(y_{\eta\eta} + Qy_{\eta}) = 0 \qquad (1.1.1b)$$

Equations (1.1.1) can also be written in tensor notation where the Laplacian represents the divergence of the gradient of a vector. In (1.1.1) E, F, G are coefficients of the first fundamental form, i.e.,

$$dx^2 + dy^2 = Ed\xi^2 + 2Fd\xi d\eta + Gd\eta^2 \quad ,$$

where

$$E = x_{\xi}^2 + y_{\xi}^2 \quad , \quad F = x_{\xi}x_{\eta} + y_{\xi}y_{\eta} \quad ,$$

$$G = x_{\eta}^2 + y_{\eta}^2 \ .$$

P and Q are functions that control the grid spacing in different regions. A commonly used form is given by [1]

$$P = - \frac{\underset{\sim}{r}_{\xi} \cdot \underset{\sim}{r}_{\xi\xi}}{\left|\underset{\sim}{r}_{\xi}\right|^2} \quad ,$$

$$Q = - \frac{\underset{\sim}{r}_\eta \cdot \underset{\sim}{r}_{\eta\eta}}{\left|\underset{\sim}{r}_\eta\right|^2} ,$$

where $\underset{\sim}{r} = (x,y)$. x and y are specified monotonically as functions of ξ,η on the boundaries of the transformed region which is a rectangular grid with numbering

$$\xi_I = 1,...,IMAX \qquad \eta_J = 1,...,JMAX.$$

P and Q are determined on the boundaries by the values of x and y. They are then linearly interpolated in the η and ξ directions respectively. Numerical differencing of (1.1.1) is facilitated by the fact that one can choose the grid spacing $D\xi=D\eta=1$.

In cartesian coordinates either the grid sizes would change monotonically in either the x or y direction for x and y to match on the profile or, if both x and y were specified at infinity arbitrarily they would not necessarily intersect on the profile and the grid sizes in both directions would jump immediately adjacent to the profile. Neither of these possibilities is desirable since staggered grid spacing may result in a loss of accuracy and differencing becomes extremely cumbersome. With a staggered grid the following expression is derived for the second order derivative Ψ_{xx} [3]; at the (I,J) grid point using (I,J) ordering:

$$\psi_{xx}\Big|_{I} = \tag{1.1.2}$$

$$2\left[(x_I - x_{I-1})\psi_{I+1} - (x_{I+1} - x_{I-1})\psi_I + (x_{I+1} - x_I)\psi_{I-1}\right]$$

$$\times\left[(x_{I+1} - x_I)(x_I - x_{I-1})(x_{I+1} - x_{I-1})\right]^{-1}$$

$$+ O\left[\frac{3\psi_{xxx}((x_{I+1} - x_I)^2 - (x_I - x_{I-1})^2)}{(x_{I+1} - x_{I-1})}\right] .$$

By observation it can be seen that (1.1.2) involves at least five multiplications and one division. Hence, solving the Poisson equation for the streamfunction, i.e., $\nabla^2\psi = f(x,y)$, using the above differencing, involves a dozen such operations per grid point not counting evaluations in $f(x,y)$. It is also seen that (1.1.1) requires eighteen operations not counting the evaluation of the control functions. Six multiplications are required for the streamfunction $\Psi(\xi,\eta)$ which in transformed coordinates satisfies

$$G(\Psi_{\xi\xi} + P\Psi_{\xi}) - 2F\Psi_{\xi\eta} + E(\Psi_{\eta\eta} + Q\Psi_{\eta}) = -J^2\, f(x,y) \tag{1.1.3}$$

If point SOR is applied to (1.1.1), (1.1.2) and (1.1.3) the number of operations becomes thirty-six and ten respectively. The convergence of (1.1.1), (1.1.3) is in general much faster than using (1.1.2) because of the uniform grid. In this case the truncation error is independent of the grid spacing leading to smooth convergence properties. A staggered grid may have the opposite effect as the truncation error depends strongly on the grid spacing.

The speed on the airfoil surface can be determined from [2]

$$q_I = \left(\Psi_\eta \frac{E^{1/2}}{W}\right)_I \quad , \quad 1 \le I \le IMAX \quad , \quad J = 1 \qquad (1.1.4)$$

following the solution of (1.1.1), (1.1.3). A one sided second order differencing is used for Ψ_η [1],

$$\left[\Psi_\eta\right]_I = \frac{1}{2}\left(- \Psi_{I,3} + 4\Psi_{I,2} - 3\Psi_{I,1} \right) . \qquad (1.1.5)$$

Similar differencing notation is used in the calculation of x_η and y_η in W. Central differencing is used for E. The velocity can be determined from the solution of (1.1.2) by the relation [3]

$$\Psi_x \Big|_I = \left(\frac{\Psi_{I+1} - \Psi_{I-1}}{x_{I+1} - x_{I-1}} \right) = - \rho v \quad ,$$

and the flow tangency condition $v/u = f'(2X)$ where ρ is the fluid density, u, v, are the velocity components and $y=f(x)$ defines the airfoil surface. The speed is given by

$$q = \left(u^2 + v^2 \right)^{1/2} \quad . \qquad (1.1.6)$$

Equations (1.1.1) through (1.1.6) represent two possible methods for the solution of incompressible, inviscid flow past a symmetric profile. Although the grid generation approach has a relatively high operational count for such a problem it is clearly preferred to manually constructing a grid and applying (1.1.2). It is shown in later sections how the new approach being proposed greatly reduces the number of operations, at least in the irrotational problem.

In conformal mapping an arbitrary two dimensional profile is mapped to a near circular shape. The near circle is mapped to a true circle by complex analytic and Fourier series techniques. Through these various transformations velocity profiles can be determined [4]. Results obtained by this approach, referred to as Theodorson's method, are used for a comparison with the new technique and that of grid generation.

The source-sink method for irrotational flows can be applied in terms of perturbation techniques (thin airfoil theory) [5] or matrix equations with known constant coefficients and unknown doublet densities, $K(\varepsilon)$. This approach uses the superposition of a uniform flow and a distribution of sources and sinks having zero total strength to obtain an expression for the streamfunction [6],

$$\Psi = u_\infty y - \int_a^b \frac{K(\varepsilon) y}{(x-\varepsilon)^2 + y^2} \, d\varepsilon \quad . \qquad (1.1.7)$$

The evaluation of $K(\varepsilon)$ is used to determine the source densities. This is accomplished by specifying n points on a prescribed shape and approximating (1.1.7) as

$$\Psi_P = u_\infty y_P - \sum_{j=1}^{n} \frac{K_j \Delta \varepsilon y_P}{(x_P - \varepsilon_j)^2 + y_P^2} \quad . \qquad (1.1.8)$$

The subscript P denotes a point P on the surface, $K_j \cdot \Delta\varepsilon$ is the total doublet strength at the j^{th} segment of the airfoil, ε_j is the distance from the origin (x=0) to the centre of the doublet and y_P is the desired value of y. The doublet strength on the body may be calculated from the solution of the matrix equation determined by (1.1.8). The distribution of the doublet $K(\varepsilon_j)$ yields Ψ_x at every point ε_j along the profile surface. The velocity component v can then be calculated and using the flow tangency condition u can be found.

1.2 OUTLINE OF THE PRESENT WORK

In the present thesis several standard numerical techniques are utilized, such as ADI (Alternating-Direction-Implicit) and SLOR (Successive-Line Over Relaxation) to yield solutions, by new methods, to several fluid dynamic flows. The results obtained compare favourably with those obtained by existing methods such as the grid generation approach. The goal of the present research is to open up the possibility of solving a series of two dimensional fluid dynamics problems such as transonic flow, circulatory flow and compressible flow. The partial achievement of this goal therefore depends on showing that the present method is successful for several fundamental problems in fluid dynamics, outlined in the abstract.

The precise formulation of the problem is based on the equations derived by Martin [7] in a study of viscous, incompressible fluids. This study is based on a transformation of the governing flow equations such that the physical variables, i.e., energy, pressure, speed and vorticity, and the metrics of a natural coordinate system are dependent variables. In fact, for irrotational flow a single elliptic partial differential equation in two metric quantities is obtained from Gauss' equation of differential geometry [8] for two dimensions based on the orthogonality of the coordinate system. If the profile is taken to be symmetric with respect to the incoming flow then zero circulation yields a specific ratio between these two metrics. The number of unknowns is subsequently reduced by one. Line SOR can be used to solve the resulting partial differential equations with central differencing applied in the interior of the computational domain. Von Neumann boundary conditions are applied on the vanishing streamline which is coincident with the profile for such flows, (cf. Figure 1). On this coordinate line integrability conditions are incorporated into a one sided second order differencing [3]. Use of integrability conditions is found to be essential for stability and convergence.

Given an arbitrary symmetric profile it is shown that a solution can be achieved by solving only two linear elliptic partial differential equations. The solution is valid over the entire portion of the profile although for certain types of airfoils a boundary condition correction is required. This can be obtained in a simple way. Velocity profiles are presented for several airfoils and compared wherever possible with exact analytic solutions, and the grid generation solution which is described in detail in [1].

In a similar manner flow equations are derived for the rotational problem. However, in this case the metrics E and G, with F zero, are related in a non-linear fashion. Hence there is an extra unknown. The numerical solution involves a solution of a first order and second order PDE but a similar algorithm as applied in the first problem can be applied here. The CPU time is considerably greater. The algorithm employs non-linear boundary conditions.

The ADI technique was used to solve the rotational problem but was found to fall short of CPU time requirements and numerical instabilities occurred. This led to a stability analysis of this method as mentioned in the abstract. Through a rigorous analysis several interesting relationships can be established to relate the convergence properties of a PDE of the form

$$AT_{\psi\psi} + BT_{\phi\phi} - 2CT_{\psi\phi} = 0$$

with variable coefficients A, B and C. That is, an interval I is sought for which the choice of one of A, B or C lying in this interval will optimize convergence. Thus, it will be better understood why instabilities occur for values chosen outside of this interval.

To conclude the thesis an inverse problem is considered, i.e., suppose the velocity profile on the airfoil surface is specified. Then the inverse problem is to determine the resultant, corresponding airfoil profile. This problem is only considered for irrotational flows.

CHAPTER 2 FLOW EQUATIONS AND AN EXACT SOLUTION

2.1 MARTIN'S FLOW EQUATIONS

The system of equations governing the steady two dimensional flow of a viscous, incompressible fluid is given by

$$\underset{\sim}{\nabla}\cdot(\rho\underset{\sim}{u}) = 0 \qquad \text{(continuity)},$$

$$\rho(\underset{\sim}{u}\cdot\nabla)\underset{\sim}{u} + \underset{\sim}{\nabla} p = \mu\nabla^2\underset{\sim}{u} \ . \qquad \text{(Navier-Stokes)} \qquad (2.1.1)$$

In (2.1.1) $\underset{\sim}{u}$ is the velocity vector with components u, v in the x, y directions respectively, p is the pressure function, ρ is the constant density and μ the constant coefficient of kinematic viscosity. u, v, and p are functions of x, y. Expanding (2.1.1) by use of the del-operator,

$$\underset{\sim}{\nabla} = \left(\frac{\partial}{\partial x} \ , \ \frac{\partial}{\partial y} \right)$$

yields

$$(\rho u)_x + (\rho u)_y = 0 \qquad (2.1.2a)$$

$$\rho(uu_x + vu_y) + p_x = \mu(u_{xx} + u_{yy}) \ ,$$

$$\rho(uv_x + vv_y) + p_y = \mu(v_{xx} + v_{yy}) \qquad (2.1.2b)$$

where subscripts denote partial derivatives. Equations (2.1.2) form a system of three equations (2nd order) in three unknowns u, v, and p. The order of the system can be reduced to one by introducing an energy function h, and a vorticity function ω, defined by

$$h = \frac{\rho}{2}(u^2 + v^2) + p \qquad (2.1.3a)$$

$$\omega = v_x - u_y \quad . \qquad (2.1.3b)$$

Taking the derivatives of h with respect to x and y, using the vorticity equation and (2.1.2b) yield a new form for the Navier-Stokes équations. The resulting new system is given by, (cf. [7]),

$$(\rho u)_x + (\rho v)_y = 0 \qquad \text{(continuity)} \qquad (2.1.4a)$$

$$h_x - \rho\omega v = - \mu\omega_y$$

$$h_y + \rho\omega u = \mu\omega_x \qquad \text{(Navier-Stokes)} \qquad (2.1.4b)$$

$$\omega = v_x - u_y \quad . \qquad \text{(vorticity)} \qquad (2.1.4c)$$

Equations (2.1.4) represent a system of four equations (1st order) in four unknown functions u, v, ω, h of x,y. The equation of continuity implies the existence of a stream-function $\psi(x,y)$ such that

$$\rho u = \psi_y \quad , \quad \rho v = - \psi_x \quad . \qquad (2.1.5)$$

Thus the vorticity function can be rewritten as

$$- \rho\omega = \psi_{xx} + \psi_{yy} \quad . \qquad (2.1.6)$$

Consider a curvilinear coordinate system (ϕ,ψ) such that ψ and ϕ are functions of x and y. To find x and y as functions of ϕ and ψ one takes the derivatives of x and y to yield the following relations

$$1 = x_\phi \, \phi_x + x_\psi \, \psi_x \quad , \quad 0 = x_\phi \, \phi_y + x_\psi \, \psi_y \quad ,$$

$$1 = y_\phi \, \phi_y + y_\psi \, \psi_y \quad , \quad 0 = y_\phi \, \phi_x + y_\psi \, \psi_x \quad . \qquad (2.1.7a)$$

Solving the system of equations (2.1.7a) for the first partials of x and y gives

$$x_\phi = \frac{\psi_y}{j} \quad , \quad x_\psi = - \frac{\phi_y}{j} \quad ,$$

$$y_\phi = \frac{\psi_x}{j} \quad , \quad y_\psi = - \frac{\phi_x}{j} \qquad (2.1.7b)$$

where $x = x(\phi,\psi)$, $y = y(\phi,\psi)$ and,

$$j^{-1} = J = \frac{\partial(x,y)}{\partial(\phi,\psi)} = x_\phi y_\psi - x_\psi y_\phi \neq 0 \ . \qquad (2.1.8)$$

Consider a transformation of variables defined by

$x = x(\phi,\psi)$, $y = y(\phi,\psi)$

such that (2.1.8) is satisfied. Using the relations

$$dx = x_\phi \, d\phi + x_\psi \, d\psi \quad , \quad dy = y_\phi \, d\phi + y_\psi \, d\psi$$

we find the first fundamental form of differential geometry is given by

$$dx^2 + dy^2 = ds^2 \qquad (2.1.9)$$

$$= E(\phi,\psi)d\phi^2 + 2F(\phi,\psi)d\phi d\psi + G(\phi,\psi)d\psi^2$$

in which the three coefficients E, F, G satisfy

$$E = x_\phi{}^2 + y_\phi{}^2 \ , \ G = x_\psi{}^2 + y_\psi{}^2 \ , \ F = x_\phi x_\psi + y_\phi y_\psi \ . \qquad (2.1.10)$$

The Jacobian J satisfies

$$J = \pm \left(EG - F^2\right)^{1/2}$$

and it is convenient to define

$$W = \left(EG - F^2\right)^{1/2}.$$

Let (x_ϕ, y_ϕ) denote the tangent vector to the coordinate line ψ=constant. The magnitude of (x_ϕ, y_ϕ) is

$$\left|(x_\phi\ ,\ y_\phi)\right| = E^{1/2}\quad ,$$

therefore, see Figure 2,

$$x_\phi = E^{1/2}\cos\alpha\ ,\ y_\phi = E^{1/2}\sin\alpha \tag{2.1.11a}$$

where α is the angle of inclination of the coordinate curve ψ=constant to the x axis. Solving for x_ψ , y_ψ from (2.1.11a) and (2.1.10) we have, (cf. [7]),

$$x_\psi = \frac{F}{E^{1/2}}\cos\alpha - \frac{J}{E^{1/2}}\sin\alpha\ ,$$

$$y_\psi = \frac{F}{E^{1/2}}\sin\alpha + \frac{J}{E^{1/2}}\cos\alpha\ . \tag{2.1.11b}$$

The integrability conditions

$$x_{\phi\psi} = x_{\psi\phi}\quad ,\quad y_{\phi\psi} = y_{\psi\phi}$$

yields a pair of equations,

$$\frac{E_\psi}{2E^{1/2}}\cos\alpha - E^{1/2}\sin\alpha\ \alpha_\psi$$

$$= \frac{F_\phi}{E^{1/2}}\cos\alpha - \frac{FE_\phi}{2EE^{1/2}}\cos\alpha - \frac{F}{E^{1/2}}\sin\alpha\ \alpha_\phi$$

$$- \frac{J_\phi}{E^{1/2}}\sin\alpha + \frac{JE_\phi}{2EE^{1/2}}\sin\alpha - \frac{J}{E^{1/2}}\cos\alpha\ \alpha_\phi$$

$$\frac{E_\psi}{2E^{1/2}}\sin\alpha + E^{1/2}\cos\alpha\ \alpha_\psi$$

$$= \frac{F_\phi}{E^{1/2}} \sin\alpha - \frac{FE_\phi}{2EE^{1/2}} \sin\alpha + \frac{F}{E^{1/2}} \cos\alpha \; \alpha_\phi$$

$$+ \frac{J_\phi}{E^{1/2}} \cos\alpha - \frac{JE_\phi}{2EE^{1/2}} \cos\alpha - \frac{J}{E^{1/2}} \sin\alpha \; \alpha_\phi \quad .$$

Multiplying the first equation by $\cos\alpha$ and the second equation by $\sin\alpha$ and adding, then respectively by $\sin\alpha$ and $-\cos\alpha$ and adding, Martin [7] obtained

$$\alpha_\phi = \frac{J}{E}\,\Gamma^2_{11} \;, \quad \alpha_\psi = \frac{J}{E}\,\Gamma^2_{12} \tag{2.1.12a}$$

where

$$\Gamma^2_{11} = \frac{-FE_\phi + 2EF_\phi - EE_\psi}{2W^2} \tag{2.1.12b}$$

$$\Gamma^2_{12} = \frac{EG_\phi - FE_\psi}{2W^2} \tag{2.1.12c}$$

$$\Gamma^2_{22} = \frac{EG_\psi - 2FF_\psi + FG_\phi}{2W^2} \tag{2.1.12d}$$

$$W = \left(EG - F^2\right)^{1/2} \quad . \tag{2.1.12e}$$

The Gaussian curvature K is obtained from the integrability condition for α,

$$\alpha_{\psi\phi} = \alpha_{\phi\psi}$$

to yield

$$K = \frac{1}{W}\left[\left(\frac{W}{E}\,\Gamma^2_{11}\right)_\psi - \left(\frac{W}{E}\,\Gamma^2_{12}\right)_\phi\right] = 0 \quad . \tag{2.1.13a}$$

Equation (2.1.13a) therefore represents a necessary and sufficient condition that E, F and G are coefficients of the first fundamental form since if E, F and G satisfy (2.1.13a) then the following relations, (cf. [7])

$$Z = x + i y = \int x_\phi \, d\phi + x_\psi \, d\psi \qquad (2.1.13b)$$

$$= \int \frac{e^{i\alpha}}{E^{1/2}} \Big(E d\phi + (F + i J) d\psi \Big) \ , \ J = \pm W ,$$

$$\alpha = \int \alpha_\phi \, d\phi + \int \alpha_\psi \, d\psi \qquad (2.1.13c)$$

$$= \int \frac{J}{E} \Big(\Gamma^2_{11} \, d\phi + \Gamma^2_{12} \, d\psi \Big)$$

determine $x(\phi,\psi)$, $y(\phi,\psi)$ and $\alpha(\phi,\psi)$. The geometrical results just formulated will be used to transform the governing flow equations (2.1.2). For that purpose several more results are quoted [7],

$$\left(\frac{E}{2W^2}\right)_\psi = \frac{F}{W^2}\Gamma^2_{12} - \frac{E}{W^2}\Gamma^2_{22} \qquad (2.1.13d)$$

$$\left(\frac{E}{2W^2}\right)_\phi = \frac{F}{W^2}\Gamma^2_{11} - \frac{E}{W^2}\Gamma^2_{12} \qquad (2.1.13e)$$

$$\left[\frac{F}{W}\right]_\phi - \left[\frac{E}{W}\right]_\psi = \frac{G}{W}\Gamma^2_{11} - \frac{2F}{W}\Gamma^2_{12} + \frac{E}{W}\Gamma^2_{22} \qquad (2.1.13f)$$

The following equations (2.1.14)-(2.1.17) are derived in [7]. Their derivations as given in [7] are summarized to suit our purposes. Equations (2.1.5), (2.1.7b) and the equation of continuity imply the following two equations

$$\mathcal{X}_\phi = \rho J u \quad , \quad \mathcal{Y}_\phi = \rho J v \tag{2.1.14}$$

for the functions $\mathcal{X}$, $\mathcal{Y}$, u and v. Conversely if the four functions satisfy (2.1.14) then $u(\phi,\psi)$, $v(\phi,\psi)$ satisfy the equation of continuity (2.1.2a) by (2.1.7b) so that (2.1.14) is equivalent to (2.1.2a). From (2.1.14) it can be seen that if J is positive then the fluid flows towards higher values of ϕ, taking u to be always positive. Similarly if J is negative the opposite effect occurs. By introducing polar coordinates in the hodograph plane u and v can be written as [7]

$$u = q\cos\theta \quad , \quad v = q\sin\theta \quad . \tag{2.1.15a}$$

θ is the direction of flow in the physical plane and

$$\theta = \alpha \ , \ J > 0 \quad ; \quad \theta = \alpha + \pi \ , \ J < 0 \quad .$$

By definition of θ and W, (2.1.14) can be written as

$$\mathcal{X}_\phi = \rho q W\cos\alpha \quad , \quad \mathcal{Y}_\phi = \rho q W \sin\alpha \tag{2.1.15b}$$

which is equivalent to the equation of continuity such that the curves ψ=constant of the curvilinear coordinate system are the streamlines. Upon squaring and adding (2.1.15b) we get

$$q = \frac{E^{1/2}}{\rho W} \tag{2.1.15c}$$

where q is the speed. Let θ denote the angle of intersection of the coordinate curves in the (ϕ,ψ) net. Then

$$F = (EG)^{1/2}\cos\theta \ , \ W = (EG)^{1/2}\sin\theta \ , \ q = \frac{1}{\rho G^{1/2}\sin\theta} \ . \tag{2.1.15d}$$

The last equation in (2.1.15d) is equivalent to (2.1.15c) which is equivalent to the equation of continuity.

The vorticity equation can also be expressed in the ϕ,ψ system. Expanding (2.1.3b) by the chain rule for derivatives and using (2.1.7b) yields

$$J\omega = x_\psi u_\phi - x_\phi u_\psi + y_\psi v_\phi - y_\phi v_\psi . \tag{2.1.16a}$$

Equation (2.1.16a) can be written as

$$E^{1/2} W\omega = Fq_\phi - Eq_\psi + Jq\alpha_\phi \tag{2.1.16b}$$

using (2.1.11), (2.1.15a). Assuming the result (2.1.15d), conservation of mass, (2.1.16b) can be written as

$$\rho\omega = \frac{G}{W} \Gamma^2_{11} - \frac{2F}{W} \Gamma^2_{12} + \frac{E}{W} \Gamma^2_{22}$$

where, by (2.1.13d-e),

$$q_\phi = \frac{F\Gamma^2_{11}}{W\rho E^{1/2}} - \frac{E\Gamma^2_{12}}{W\rho E^{1/2}}$$

$$q_\psi = \frac{F\Gamma^2_{12}}{W\rho E^{1/2}} - \frac{E\Gamma^2_{22}}{W\rho E^{1/2}}$$

or, by (2.1.13f) and (2.1.6),

$$- \rho\omega = \nabla^2\psi = \frac{1}{\rho W} \left[\left(\frac{F}{W}\right)_\phi - \left(\frac{E}{W}\right)_\psi \right] . \tag{2.1.16c}$$

In a similar manner the Navier-Stokes equations (2.1.4b) can be transformed so that ϕ,ψ form the independent variables. This is accomplished by using the chain rule for derivatives, (2.1.7b), and (2.1.14).

Using the previous results we have the following theorem [7]. Let ψ=constant denote the streamlines of the flow in question. The flow equations (2.1.4) of a viscous incompressible fluid, with curvilinear coordinates (ϕ,ψ) as independent variables and functions E, F, G, h, ω, p as dependent variables, transforms to the system of equations,

$$Gh_\phi - F(h_\psi + \omega) = - J\mu\omega_\psi \quad ,$$

$$- Fh_\phi + E(h_\psi + \omega) = J\mu\omega_\phi \qquad \text{(Navier-Stokes)} \quad (2.1.17a)$$

$$\omega = \frac{1}{\rho W}\left[\left(\frac{F}{W}\right)_\phi - \left(\frac{E}{W}\right)_\psi\right] \qquad \text{(vorticity)} \quad (2.1.17b)$$

$$h = \frac{E}{2\rho W^2} + p \qquad \text{(energy)} \quad (2.1.17c)$$

$$K = \frac{1}{W}\left[\left(\frac{W}{E}\,\Gamma^2_{11}\right)_\psi - \left(\frac{W}{E}\,\Gamma^2_{12}\right)_\phi\right] = 0 \qquad \text{(Gauss)} \quad (2.1.17d)$$

with E, F, G satisfying (2.1.10), and where

$$W = \left(EG - F^2\right)^{1/2} \; , \; J = \pm\, W \; ,$$

and ϕ=constant are an arbitrary family of curves. The continuity equation is equivalent to

$$q = \frac{E^{1/2}}{\rho W} \quad . \qquad (2.1.18)$$

x, y and α can be determined from (2.1.13b) and (2.1.13c) given a solution to system (2.1.17). This theorem, stated by Martin [7], gives the form of the governing flow equations and serves as the starting point for the present work.

Equations (2.1.17a-c) and (2.1.18) can be written in non-dimensional form as

$$Gh_{\phi} - F(h_{\psi} + \omega) = - WRe^{-1}\omega_{\psi} \quad ,$$

$$-Fh_{\phi} + E(h_{\psi} + \omega) = WRe^{-1}\omega_{\phi} \qquad \text{(Navier-Stokes)} \quad (2.1.19a)$$

$$\omega = \frac{1}{W}\left[\left(\frac{F}{W}\right)_{\phi} - \left(\frac{E}{W}\right)_{\psi}\right] \qquad \text{(vorticity)} \quad (2.1.19b)$$

$$h = \frac{E}{2W^{2}} + p \qquad \text{(energy)} \quad (2.1.19c)$$

$$q = \frac{E^{1/2}}{W} \qquad \text{(speed)} \quad (2.1.19d)$$

where the Jacobian is taken to be positive

$$W = \left(EG - F^{2} \right)^{1/2} > 0$$

so that the parameter ϕ increases in the flow direction. Non-dimensional (unbarred) variables are related to physical variables by:

$$\bar{h} = \rho u_{\infty}^{2} h \quad , \quad \bar{\omega} = \frac{u_{\infty}}{L}\,\omega \quad , \quad \bar{u} = u_{\infty}\, u \quad ,$$

$$\bar{v} = u_{\infty}\, v \quad , \quad \bar{x} = Lx \quad , \quad \bar{y} = Ly \quad .$$

$Re = \rho \dfrac{u_{\infty} L}{\mu}$ is the flow Reynold's number, u_{∞} is the speed at infinity and L is a characteristic length.

By inspection, (2.1.19a-b) constitutes three equations in five unknowns E, F, G, h, ρ. Having determined these unknowns, pressure p and speed q are found from (2.1.19c,d). Thus at least one more equation is required, and this is the Gauss' equation. One extra unknown is still left which may be determined by the physical nature of the problem or the curvilinear coordinates (ϕ,ψ) may be chosen to be orthogonal (F=0). Given the characteristics of a particular flow with governing equations given by (2.1.19) it is necessary to put these in a form suitable for numerical computation, as well as to derive the essential Von Neumann or Dirichlet boundary conditions. The approach being considered is such that the boundary conditions are not everywhere specified at the outset.

To obtain a correct solution it is essential to use Gauss' equation. Now the integrability conditions for X depend on α_ϕ. The surface $y=f(X)$ is coincident with a section of the boundary ψ=constant (taken to be zero) for the flows being considered. The integrability condition therefore depends on the airfoil profile, in particular the angle of inclination of the $\psi=0$ streamline at a particular point on the profile. Hence it is required to specify the values of α here or find them by some other means. Knowing $X(\phi,0)$ then we can find $f(X)$ and hence $\tan\alpha$ from $f'(X)$; Gauss' equation can then be solved, using the first equality of (2.1.12a) derived from the integrability condition for X along the portion of the boundary where the profile is coincident with the streamline $\psi=0$.

In many cases the profile under consideration can be considered as a parameterized curve in cartesian coordinates which coincides with particular coordinate curves ψ=constant or ψ=constant in the (ϕ,ψ) plane. As an example, consider a parameterized curve in the notation of [8],

$$\beta(t) = \Gamma\,[\phi(t)\ ,\ \psi(t)] \qquad (2.1.20)$$

in the $\mathcal{X}$-$\mathcal{Y}$ plane. The arc length s(t) satisfies, by (2.1.9),

$$\left(\frac{ds}{dt}\right)^2 = E\left(\frac{d\phi}{dt}\right)^2 + 2F\,\frac{d\phi}{dt}\,\frac{d\psi}{dt} + G\left(\frac{d\psi}{dt}\right)^2 . \qquad (2.1.21a)$$

Suppose

$$\mathcal{X} = t,\ \mathcal{Y} = f[\mathcal{X}(t)] = f(t)\ ,$$

and $[t, f(t)]$ coincides with a particular curve, say,

$$\psi = \text{constant} = 0.$$

Then,

$$\begin{aligned}\frac{d\psi}{dt} &= \frac{\partial\psi}{\partial\mathcal{X}}\frac{d\mathcal{X}}{dt} + \frac{\partial\psi}{\partial\mathcal{Y}}\frac{d\mathcal{Y}}{dt}\\ &= \frac{\partial\psi}{\partial\mathcal{X}} + \frac{\partial\psi}{\partial\mathcal{Y}}\frac{d\mathcal{Y}}{d\mathcal{X}} \qquad (2.1.21b)\\ &= \frac{\partial\psi}{\partial\mathcal{X}} + \frac{\partial\psi}{\partial\mathcal{Y}}\,f'(\mathcal{X})\\ &= \frac{\partial\psi}{\partial\mathcal{X}} + \frac{\partial\psi}{\partial\mathcal{Y}}\tan\alpha(\mathcal{X})\ ,\quad f'(\mathcal{X})=\tan\alpha(\mathcal{X})\\ &= 0\ .\end{aligned}$$

Equation (2.1.21b) corresponds to the flow tangency condition for inviscid flows. Note that the vectors $[1, f'(t)]$ and

$$\left(\frac{\partial\psi}{\partial\mathcal{X}}\ ,\ \frac{\partial\psi}{\partial\mathcal{Y}}\right)$$

are orthogonal since $\underset{\sim}{\nabla}\psi$ is orthogonal to the coordinate curves ψ=constant. Therefore,

$$\frac{ds}{d\mathcal{X}} = \frac{\left(d\mathcal{X}^2+d\mathcal{Y}^2\right)^{1/2}}{d\mathcal{X}}$$

$$= \left(1 + f'(\mathcal{X})^2 \right)^{1/2}$$

$$= \sec\alpha(\mathcal{X}) \quad . \qquad (2.1.22a)$$

Using (2.1.22a) in (2.1.21a) yields

$$\frac{d\phi}{d\mathcal{X}} = \frac{1}{E^{1/2}\cos\alpha} \qquad \alpha \neq \pm \frac{\Pi}{2} \qquad (2.1.22b)$$

which can be integrated when E and $f'(\mathcal{X})$ are known, to yield ϕ as a function of $\mathcal{X}$,

$$\phi = \phi(\mathcal{X},\mathcal{Y})_{\mathcal{Y}=f(\mathcal{X})} = \phi[\mathcal{X},\mathcal{Y}(\mathcal{X})]$$

or equivalently, $\mathcal{X}$ as a function of ϕ,

$$\mathcal{X} = \mathcal{X}(\phi,\psi)_{\psi=0} = \mathcal{X}(\phi,0) \quad .$$

Then, since $\mathcal{Y}=f(\mathcal{X})$, we have $\mathcal{Y}$ as a function of ϕ also,

$$\mathcal{Y} = \left(f[\mathcal{X}(\phi,\psi)]\right)_{\psi=0}$$

$$= f[\mathcal{X}(\phi,0)] \quad .$$

Conversely given $f(\mathcal{X})$, if $\phi(\mathcal{X})$ or $\mathcal{X}(\phi)$ and hence $f[\mathcal{X}(\phi)]$ are known, E can be determined along the coordinate line ψ=0.

Consider the boundary condition (2.1.21b) in light of inviscid flow with the flow tangency condition holding. Using (2.1.7b) the inverse relations for $\mathcal{X}$, $\mathcal{Y}$, in (2.1.21b) we have

$$\frac{\mathcal{Y}_\phi}{\mathcal{X}_\phi} = \tan\alpha \quad . \qquad (2.1.23)$$

Differentiating (2.1.23) with respect to ψ and ϕ one can show the resultant equations are equivalent to (2.1.12a) such that

$$\frac{dy}{dx} = \tan\alpha = \frac{y_\phi}{x_\phi} \quad , \quad \psi = 0 \quad .$$

Thus the flow tangency condition is consistent with (2.1.12a). Now we consider several profiles which will be used in later sections.

2.2 PROFILES

Consider the governing equations (2.1.19) for the flow of a viscous, incompressible fluid in an infinite domain. It is the purpose of the present work to solve these equations in a finite domain in which an arbitrary symmetric profile has been inserted. The entire focus of previous work using Martin's approach has been to find exact or analytical solutions without applying boundary conditions, see for example, [9] and [10]. Owing to the non-linearity of the equations, exact solutions are difficult to find in most cases except when some simplyfying assumptions are made, i.e., the form of the streamfunction is assumed, or the geometry is specified, such as spiral, vortex, or parallel flow. In the present work no such assumptions are made on the streamlines except at the regions of the computational domain far from the profile where the flow is assumed known, taken to correspond to the flow at infinity in the physical plane.

The problem then is to accurately predict the flow in a neighbourhood of the profile under consideration by the numerical solution of the finite differenced form of the governing fluid flow equations.

For particular flow problems we consider profiles of the form,

$$y = f(x) = \pm \epsilon \left(a^2 - x^2 \right)^b \qquad a,\ b,\ \epsilon \in R^+ . \tag{2.2.1}$$

The constant ϵ usually represents the thickness parameter in a singular perturbation problem, a is a positive constant such that

$$x_{LE}^2 = x_{TE}^2 = a^2 , \tag{2.2.2}$$

b is also a positive constant and can be chosen so that f ranges from parabolic to circular shapes. Two other forms that are considered are the second order approximation to a symmetrical Joukowski airfoil [5],

$$y(x) = \pm\epsilon(1-x)\left(1 - x^2 \right)^{1/2} \tag{2.2.3}$$

and the NACA 0012-64 symmetrical airfoil given by [4],

$$\pm y_t = 5t(.2969x^{1/2}-.126x-.3516x^2+.2843x^3-.1015x^4), \tag{2.2.4}$$

where t is the maximum thickness expressed as a fraction of the chord, given by

$$r_t = 1.1019t^2$$

where r_t is the leading edge radius having value

$$r_t = .01582. \tag{2.2.5}$$

Now (2.1.12a) can be written in terms of the curvature $\kappa(x)$ of the surface of the airfoil. This can be seen by expanding α_ϕ by the chain rule for partial derivatives to get

$$\alpha_\phi = \alpha_x\, x_\phi + \alpha_y\, y_\phi \quad . \tag{2.2.6}$$

The angle α depends on x and y. On the streamline $\psi=0$ $y=f(x)$. Therefore,

$$\begin{aligned}\frac{d\alpha}{dx} &= \alpha_x + \alpha_y \frac{dy}{dx}\\ &= \alpha_x + \alpha_y \tan\alpha\\ &= \frac{d}{dx}\left[\tan^{-1} f'(x)\right] = \frac{f''(x)}{1 + f'^2(x)} \quad .\end{aligned} \tag{2.2.7a}$$

Using (2.1.11a) and (2.2.7a) in (2.2.6) yields

$$\begin{aligned}\alpha_\phi &= \left(\frac{f''(x)}{1+f'^2(x)} - \alpha_y \tan\alpha\right)E^{\frac{1}{2}}\cos\alpha + \alpha_y\, E^{\frac{1}{2}}\sin\alpha\\ &= \frac{f''(x)E^{\frac{1}{2}}\cos\alpha}{1 + f'^2(x)}\\ &= \frac{f''(x)E^{\frac{1}{2}}}{\left(1 + f'^2(x)\right)^{3/2}} \quad .\end{aligned} \tag{2.2.7b}$$

Applying the chain rule for partial derivatives to α_ψ gives

$$\alpha_\psi = \alpha_x\, x_\psi + \alpha_y\, y_\psi \quad . \tag{2.2.8a}$$

Using (2.2.7a), (2.1.11b) in (2.2.8a) gives

$$\alpha_\psi = \left[\frac{d\alpha}{dx} - \alpha_y \tan\alpha\right] x_\psi + \alpha_y y_\psi$$

$$= \frac{f''(\chi)\ \chi_\psi}{\left[1 + f'^2(\chi)\right]} + \alpha_y\left[y_\psi - \tan\alpha\chi_\psi\right]$$

$$= \frac{f''(\chi)}{\left[1 + f'^2(\chi)\right]^{1.5}}\left(\frac{F\cos\alpha}{\sqrt{E}} - \frac{J\sin\alpha}{\sqrt{E}}\right)$$

$$+ \alpha_y\frac{J\sec\alpha}{\sqrt{E}}. \qquad (2.2.8b)$$

Let $f(\chi)$ be a continuously differentiable function in the neighbourhood of a point $\chi_\circ$. By definition the signed curvature of f at the point

$$(\chi_\circ, f(\chi_\circ))$$

is given by [8],

$$\kappa = \frac{f''(\chi_\circ)}{\left(1 + f'(\chi_\circ)^2\right)^{3/2}} . \qquad (2.2.9)$$

From (2.2.1),

$$\frac{df(\chi)}{d\chi} = f'(\chi) = \mp 2\epsilon b\left(a^2 - \chi^2 \right)^{b-1} \chi \qquad (2.2.10)$$

$$\frac{d^2 f(\chi)}{d\chi^2} = f''(\chi) \qquad (2.2.11)$$

$$= \mp 2\epsilon b\left(a^2 - \chi^2 \right)^{b-1} \pm 4\epsilon b(b - 1)\left(a^2 - \chi^2 \right)^{b-2} \chi^2 .$$

Using (2.2.10), (2.1.11) in (2.2.9) we obtain

$$\pm\kappa = \qquad (2.2.12)$$

$$\frac{-2\epsilon b(a^2-\chi^2)^{b-1}(1-2(b-1)(a^2-\chi^2)^{-1}\chi^2)}{(1+(2\epsilon b\chi(a^2-\chi^2)^{b-1})^2)^{3/2}}$$

From (2.2.3),

$$f'(x) = \mp \epsilon \left\{ \left[1 - x^2 \right]^{1/2} + \frac{x(1-x)}{\left(1-x^2 \right)^{1/2}} \right\}$$

$$= \mp \epsilon \left[\frac{1 - x^2 + x - x^2}{(1-x^2)^{1/2}} \right]$$

$$= \mp \epsilon \left\{ \frac{2x^2 - x - 1}{(1-x^2)^{1/2}} \right\} \qquad (2.2.13)$$

$$f''(x) = \mp \epsilon \left[\frac{4x-1}{\left(1-x^2 \right)^{1/2}} + \frac{x(2x^2-x-1)}{\left(1-x^2 \right)^{3/2}} \right]$$

after some simplification we get

$$f''(x) = \mp \epsilon \frac{(x-1)(1-2x-2x^2)}{\left(1-x^2 \right)^{3/2}} . \qquad (2.2.14)$$

Using (2.2.13) and (2.2.14) in (2.2.9) we find the curvature κ is given by

$$\pm\kappa = \frac{\epsilon(2x^2 + 2x-1)(1 - x)}{\left[1 - x^2 + \epsilon^2 \left(2x^2-x-1 \right)^2 \right]^{3/2}} . \qquad (2.2.15)$$

Likewise from (2.2.4),

$$y_t' = 5t(.14845x^{-1/2}-.126-.7032x+.8529x^2-.4060x^3) \qquad (2.2.16)$$

$$y_t'' = 5t(-.074225x^{-3/2} - .7032 + 1.7058x - 1.218x^2)$$

$$\kappa = \frac{5t\left\{-.074225x^{-3/2}-.7032+1.7058x-1.218x^2\right\}}{\left[1+\left\{ 5t\ (.14845x^{-1/2}-.126-.7012x+.8529x^2-.406x^3) \right\}^2 \right]^{3/2}}$$

The NACA-0012 can be translated to the origin by the translation $x \rightarrow x+.5$. Of the profiles considered the

NACA-0012 is the most relevant from a research and experimental standpoint and the class of profiles given by (2.2.1) is studied extensively in [5]. The profiles so far considered are also chosen to be symmetrical with respect to the x axis. Moreover, in the problems considered the angle of attack is taken to zero. Further research problems include asymmetric profiles, various angles of attack and the effects of circulation which are not attempted in the thesis.

As a starting point we consider irrotational, inviscid, incompressible flow over a circle of unit radius. The exact solution in polar coordinates r,θ with ψ,ϕ as dependent variables can be inverted so that ϕ,ψ serve as the independent variables. The coefficients E, F, G are then determimed in terms of ϕ and ψ on the profile surface ($\psi=0$); subsequently velocity and pressure can be calculated and used as a check for the accuracy of the numerical solution.

2.3 FLOW OVER A CIRCLE

Consider the inviscid, irrotational, incompressible flow around a circle of unit radius. The streamfunction, velocity potential function and non-dimensional variables are given by [4],

$$\psi^* = \left(r^* - \frac{1}{r^*}\right)\sin\theta^* \quad ,$$

$$\phi^* = \left(r^* + \frac{1}{r^*}\right)\cos\theta^* \qquad (2.3.1)$$

$$\phi^* = \frac{\phi}{\rho U_\infty\, a}\ , \quad \psi^* = \frac{\psi}{\rho U_\infty\, a}\ , \quad \theta^* = \theta\ , \quad r^* = \frac{r}{a}\ .$$

In (2.3.1) U_∞ and a are positive constants denoting the upstream velocity and radius of an arbitrary circle centred at the origin. (r,θ) denotes polar coordinates. Deleting *, the velocity components in the r and θ directions respectively are given by

$$\psi_r = -v(r,\theta) = \left[1+ \frac{1}{r^2}\right]\sin\theta \quad ,$$

$$\frac{\psi_\theta}{r} = u(r,\theta) = \left[1 - \frac{1}{r^2}\right]\cos\theta \quad . \qquad (2.3.2)$$

The velocity $q(r,\theta)$ at any point on the surface of the circle is given by

$$q(r,\theta)\Big|_{r=1} = \left(\psi_r^{\,2} + \frac{1}{r^2}\,\psi_\theta^{\,2}\right)^{\frac{1}{2}}\Big|_{r=1} \qquad (2.3.3)$$

$$= \left[(1+\frac{1}{r})^2\sin^2\theta+\left(1-\frac{1}{r^2}\right)^2\cos^2\theta\right]^{\frac{1}{2}}\Big|_{r=1}$$

$$= 2\ \sin\theta \quad .$$

$\alpha(\phi,\psi)$ by definition satisfies

$$\alpha(\phi,\psi)\Big|_{\psi=0} = \theta - \frac{\pi}{2}\ . \qquad (2.3.4a)$$

Thus, on the airfoil surface, $\psi=0$, and considering q as a function of ϕ, ψ, we get using (2.3.4a) in (2.3.3)

$$q(\phi,0) = 2\cos(\alpha(\phi,0))\ . \qquad (2.3.4b)$$

Consider r and θ as functions of ϕ and ψ. The following inverse relations hold,

$$r_\phi = \frac{\psi_\theta}{j}\ ,\ \theta_\phi = -\frac{\psi_r}{j}\ ,$$

$$r_\psi = -\frac{\phi_\theta}{j}\ ,\ \theta_\psi = \frac{\phi_r}{j} \qquad (2.3.5a)$$

where the Jacobian j satisfies

$$j = \phi_r\psi_\theta - \phi_\theta\psi_r \neq 0\ . \qquad (2.3.5b)$$

For polar coordinates one has the standard relations

$$x = r\cos\theta\ ,\ y = r\sin\theta\ ,\ r = \left(x^2 + y^2\right)^{\frac{1}{2}}\ ,\ \theta = \tan^{-1}\frac{y}{x}\ ,$$

$$x_r = \cos\theta\ ,\ y_r = \sin\theta\ ,$$

$$x_\theta = -\, r\sin\theta\ ,\ y_\theta = r\cos\theta\ . \qquad (2.3.6)$$

The coefficients E, F, G can be determined in terms of r,θ by considering ϕ,ψ as functions of r,θ. One obtains by use of the chain rule for derivatives and (2.3.6),

$$E(\phi,\psi) = E[\phi,(r,\theta),\ \psi(r,\theta)] = x_\phi{}^2 + y_\phi{}^2$$

$$= (x_r\, r_\phi + x_\theta\, \theta_\phi)^2 + (y_r\, r_\phi + y_\theta\, \theta_\phi)^2$$

$$= (x_r{}^2 + y_r{}^2)r_\phi{}^2 + 2\theta_\phi\, r_\phi(x_r\, x_\theta + y_r\, y_\theta) + (x_\theta{}^2 + y_\theta{}^2)\theta_\phi{}^2$$

$$= r_\phi{}^2 + r^2\theta_\phi{}^2\ . \qquad (2.3.7)$$

Similarly,

$$G(\phi,\psi) = r_\psi{}^2 + r^2\theta_\psi{}^2\ . \qquad (2.3.8)$$

By (2.3.1) we also have

$$\psi_r = \left[1 + \frac{1}{r^2}\right]\sin\theta \ , \ \phi_r = \left[1 - \frac{1}{r^2}\right]\cos\theta$$

$$\psi_\theta = \left[r - \frac{1}{r}\right]\cos\theta \ , \ \phi_\theta = -\left[r + \frac{1}{r}\right]\sin\theta \ . \qquad (2.3.9)$$

Thus from (2.3.5b) and (2.3.9),

$$j = r\left(1 - \frac{1}{r^2}\right)^2\cos^2\theta + r\left(1 + \frac{1}{r}\right)^2\sin^2\theta$$

$$E(r,\theta) = \frac{1}{j^2}\left[\psi_\theta{}^2 + r^2\psi_r{}^2\right]$$

$$= \frac{1}{j^2}\left[\left(r-\frac{1}{r}\right)^2\cos^2\theta + r^2\left(1+\frac{1}{r^2}\right)^2\sin^2\theta\right] \qquad (2.3.10a)$$

$$G(r,\theta) = \frac{1}{j^2}\left[\phi_\theta{}^2 + r^2\phi_r{}^2\right] \qquad (2.3.10b)$$

$$= \frac{1}{j^2}\left[\left(r+\frac{1}{r}\right)^2\sin^2\theta + r^2\left(1-\frac{1}{r^2}\right)^2\cos^2\theta\right] \ .$$

It is observed from (2.3.10) that

$$E(r,\theta) \neq G(r,\theta) \qquad \forall \ r,\theta \in \mathcal{D}$$

where $\mathcal{D}$ is the flow region exterior to the circle r=1. For the metric F, using (2.3.6) again

$$F(\phi,\psi) = F\left(\phi(r,\theta),\ \psi(r,\theta)\right)$$

$$= x_\phi x_\psi + y_\phi y_\psi$$

$$= (x_r r_\phi + x_\theta\theta_\phi)(x_r r_\psi + x_\theta\theta_\psi) + (y_r r_\phi + y_\theta\theta_\phi)(y_r r_\psi + y_\theta\theta_\psi)$$

$$= (x_r{}^2 + y_r{}^2)(r_\phi r_\psi) + (y_\theta{}^2 + x_\theta{}^2)(\theta_\phi\theta_\psi)$$

$$+ (x_\theta x_r + y_\theta y_r)(\theta_\phi r_\psi + r_\phi\theta_\psi)$$

$$= r_\phi r_\psi + r^2\theta_\phi\theta_\psi$$

$$= -\frac{1}{j^2}\left[\psi_\theta\ \phi_\theta + r^2\psi_r\ \phi_r\right] \qquad (2.3.11)$$

$$= 0 \quad .$$

From the equation of continuity (2.1.18), by use of (2.3.10) and (2.3.11) we find

$$q = \frac{E^{½}}{\rho W} = \frac{1}{\rho E^{½}} = \frac{1}{E^{*½}} \quad . \qquad (2.3.12)$$

From (2.3.1), (2.3.4) and (2.3.12) we find

$$E(\phi,0) = \frac{1}{4\cos^2\alpha} = \frac{1}{4\left[1-\sin^2\alpha\right]}$$

$$= \frac{1}{4\left[1-\frac{\phi^2}{4}\right]} = \frac{1}{4-\phi^2} \quad , \quad |\phi| < 2 \quad . \qquad (2.3.13)$$

The result (2.3.13) gives the coefficient $E(\phi,0)$ of (2.1.10) along the coordinate line ψ=0 and coincident with the profile under consideration.

The individual partial derivatives for x and y can be determined in terms of r, θ. Assuming,

$$x = x(\phi(r,\theta),\psi(r,\theta)) \quad , \quad y = y(\phi(r,\theta),\psi(r,\theta))$$

we have, by use of the chain rule for partial derivatives,

$$y_\theta = y_\phi\phi_\theta + y_\psi\psi_\theta = x \quad ,$$

$$x_\theta = x_\phi\phi_\theta + x_\psi\psi_\theta = -y \quad ,$$

$$y_r = y_\phi\ \phi_r + y_\psi\ \psi_r = \frac{y}{r} \quad ,$$

$$x_r = x_\phi\ \phi_r + x_\psi\ \psi_r = \frac{x}{r} \quad . \qquad (2.3.14)$$

Solving for x_ϕ, x_ψ, y_ϕ, y_ψ from (2.3.14) (2.3.9) gives

$$x_\phi = \frac{x_\theta\ \psi_r - x_r\ \psi_\theta}{\phi_\theta\ \psi_r - \phi_r\ \psi_\theta}$$

$$= \left[r\sin\theta\left(1 + \frac{1}{r^2}\right)\sin\theta + \cos\theta\left(r - \frac{1}{r}\right)\cos\theta \right] \times$$

$$\left[\left(r + \frac{1}{r}\right)\sin\theta\left(1 + \frac{1}{r^2}\right)\sin\theta + \left(1 - \frac{1}{r^2}\right)\cos\theta\left(r - \frac{1}{r}\right)\cos\theta \right]^{-1}$$

$$= \frac{r^4 - r^2\cos 2\theta}{r^4 + 1 - 2r^2\cos 2\theta} \quad . \tag{2.3.15}$$

ϕ as given in (2.3.1) denotes the velocity potential. Inspection of (2.1.7b) and (2.3.10-11) we find

$$x_\phi = y_\psi$$

and thus y_ψ is given by (2.3.15). This can also be verified by a similar procedure as above. Solving (2.3.14) for y_ϕ

$$y_\phi = x_\psi = \frac{\phi_\theta\ x_r - \phi_r\ x_\theta}{\phi_\theta\ \psi_r - \phi_r\ \psi_\theta} \quad . \tag{2.3.16}$$

After some simplification we get

$$y_\phi = x_\psi = \frac{r^2\sin 2\theta}{r^4 - 2r^2\cos^2\theta + 1} \quad . \tag{2.3.17}$$

From (2.3.15) and (2.3.17) we find

$$E(r,\theta) = x_\phi{}^2 + y_\phi{}^2$$

$$= \frac{\left(r^4 - r^2\cos 2\theta\right)^2 + r^4\sin^2 2\theta}{\left(r^4 - 2r^2\cos 2\theta + 1\right)^2}$$

$$= \frac{r^4}{r^4 - 2r^2\cos 2\theta + 1} \tag{2.3.18}$$

which is identical to (2.3.10a). Along the coordinate curve, $\psi=0$,

$$\theta = \begin{cases} 0 & , \; r > 1 \\ \Pi & , \; -r < -1 \\ \tan^{-1}\left(\frac{y}{x}\right) & , \; r=1 \;. \end{cases} \qquad (2.3.19)$$

Therefore from (2.3.18) and (2.3.19) the expression for E along $\psi=0$ is given by

$$E(r,0) = E(r,\Pi)$$

$$= \frac{r^4}{\left[\; r^2-1 \;\right]^2} \;, \quad r \neq 1 \qquad (2.3.20)$$

$$E(1,\theta) = \frac{1}{2\left[1-\cos 2\theta\right]}$$

$$= \frac{1}{4\sin^2\theta} \;, \quad r = 1 \;. \qquad (2.3.21)$$

Equation (2.3.21) agrees with (2.3.4b).

Next, it is necessary to find r and θ in terms of ϕ,ψ in order to completely determine E as a function of ϕ and ψ. Squaring and adding ϕ,ψ as given in (2.3.1) yields

$$\phi^2 + \psi^2 = \left[r-\frac{1}{r}\right]^2 \sin^2\theta + \left[r+\frac{1}{r}\right]^2 \cos^2\theta$$

$$= \left(r^2+\frac{1}{r^2}\right)\left(\cos^2\theta+\sin^2\theta\right) + 2\left(\cos^2\theta-\sin^2\theta\right)$$

$$= r^2 + \frac{1}{r^2} + 2\cos 2\theta \;. \qquad (2.3.22)$$

Similarily squaring and subtracting yields

$$\phi^2 - \psi^2 = \left[r+\frac{1}{r}\right]^2\cos^2\theta - \left[r-\frac{1}{r}\right]^2\sin^2\theta$$

$$= \left(r^2+\frac{1}{r^2}\right)\left(\cos^2\theta-\sin^2\theta\right)+2\left(\cos^2\theta+\sin^2\theta\right)$$

$$= \left(r^2 + \frac{1}{r^2}\right) \cos 2\theta + 2 \quad . \tag{2.3.23}$$

Eliminating $\cos 2\theta$ from (2.3.22) and (2.3.23)

$$\phi^2 - \psi^2 = \frac{1}{2}\left(r^2+\frac{1}{r^2}\right)\left(\phi^2+\psi^2-r^2-\frac{1}{r^2}\right)+2 \; .$$

That is

$$R^2-R(\phi^2+\psi^2)-4+2(\phi^2-\psi^2)=0 \tag{2.3.24}$$

where

$$R = r^2 + \frac{1}{r^2} \quad . \tag{2.3.25}$$

Solving (2.3.24) for R by quadratic formula yields

$$R = \frac{1}{2}\left[\phi^2 + \psi^2 \pm \left((\phi^2 + \psi^2)^2 - 4(2(\phi^2 - \psi^2) - 4)\right)^{\frac{1}{2}}\right]$$

$$= \frac{1}{2}\left[\phi^2 + \psi^2 \pm \left(\phi^4 + 2\phi^2\psi^2 + \psi^4 + 8\psi^2 - 8\phi^2 + 16\right)^{\frac{1}{2}}\right] . \tag{2.3.26}$$

By (2.3.25) and (2.3.26) if

$$\psi = 0 \quad , \quad r = 1$$

$$R = 2 = \frac{1}{2}\left[\phi^2 \pm (\phi^2-4)\right] \tag{2.3.27}$$

so that minus sign is taken in (2.3.26). For

$$\psi \geq 0 \quad , \quad r > 1$$

take the plus sign in (2.3.27) since taking the minus sign requires that,

$$\left(\phi^2 + \psi^2\right)^2 > \left(\phi^2 - 4\right)^2 + \left(\psi^2 + 4\right)^2 + 2\phi^2\psi^2 - 16$$

since $R > 0$. Simplifying the above inequality,

$$\phi^2 - \psi^2 > 2$$

which cannot be satisfied for

$$|\phi| < 2^{\frac{1}{2}} \quad , \quad \psi \geq 0 \quad .$$

The change in sign in the quadratic corresponds to singularity at the leading and trailing edge evident from (2.3.21) when θ vanishes or equals π.

For ψ=0, r>1 (2.3.26) yields

$$R = r^2 + \frac{1}{r^2} = \frac{1}{2}\left[\phi^2 + \left(\phi^2 - 4\right)\right]$$

$$= \phi^2 - 2 \quad . \tag{2.3.28}$$

Solving the quadratic (2.3.28) for r and substituting into (2.3.20) we find

$$E(r,0) = E(r,\Pi) \tag{2.3.29}$$

$$= \frac{1}{\left[1 - \frac{1}{2}\left[\phi^2 - 2 + \left[\left[\phi^2 - 2\right]^2 - 4\right]^{\frac{1}{2}}\right]\right]^2}$$

$$= E(\phi,0) \quad , \quad \frac{1}{2}\left(\phi^2 - 2 + \left(\left(\phi^2 - 2\right)^2 - 4\right)^{\frac{1}{2}}\right) > 1 \quad .$$

Equations (2.3.13) and (2.3.29) give the exact value for $E(\phi,0)$ and hence the speed at every point $(\phi,0)$. The pressure distribution p as a function of ϕ is given by

$$c_p = \frac{p - p_\infty}{\frac{1}{2}\,\rho U_\infty^2} = 1 - q^2 \quad . \tag{2.3.30}$$

Equation (2.3.30) is equivalent to the energy equation (2.1.17c) and is also known as Bernoulli's equation. To find the solution at any point ϕ,ψ it is required to solve (2.3.26) for r and then substitute this expression into (2.3.18). cos2θ can be found from (2.3.22) or (2.3.23). A complicated expression would arise. Thus it suffices to check the numerical solution with the exact solution along the coordinate line ψ=0 provided the continuity of the former is maintained. The leading and trailing edges constitute the only singular points of the exact solution, i.e., the polar form of the Jacobian vanishes. This will constitute a singular point of the transformation to ϕ,ψ coordinates except that the Jacobian will be infinite here. This can be seen from (2.3.10a) and (2.3.13). At these two points the velocity vanishes. This singular point thus corresponds to the physical requirement of zero velocity.

CHAPTER 3 IRROTATIONAL, INVISCID FLOW

3.1 SIMPLIFICATION OF GOVERNING EQUATIONS

In this chapter the governing equations with boundary conditions are formulated and solved for steady, inviscid, irrotational, incompressible flow over an arbitrary symmetric profile. A numerical algorithm is subsequently developed which is suitable for any high speed computer such as an IBM 3081. This algorithm, as will be seen, involves the inversion of large tridiagonal matrices.

One of the concerns mentioned in the introduction was the reduction of operational count as compared with the elliptic grid generation approach. The high count was due primarily to the calculation of the metrics E, F, and G, which involved two multiplications apiece per grid point. One of the advantages of the present approach, as will be shown, is that the metrics need not be calculated from grid generation equations. The governing equation is shown to be a linear elliptic partial differential equation with constant coefficients. Equations (2.1.19) are, therefore, considered as a starting point. We show, based on the flow assumptions, that E=G and F=0 throughout the flow region.

Existence and uniqueness of a solution can be established depending on the type of boundary conditions applied and the type of partial differential equation employed. Existence and uniqueness theorems [12] exist for the linear elliptic equations developed in the present chapter for either type of boundary condition.

As stated previously the addition of Gauss' equation (2.1.19d) is necessary and leaves one unknown more than number of equations. For flows of zero vorticity h is easily determined from (2.1.19a) to be constant, say h_∞. It is left for (2.1.17d) and (2.1.19b) to determine E, F, and G. The pressure coefficient is determined from (2.3.30) with q replaced by (2.1.19d). Other restrictions on the metrics are determined by the physical nature of the problem which implies the existence of a velocity potential $\phi(\mathcal{X},\mathcal{Y})$ provided $\psi(\mathcal{X},\mathcal{Y})$ is harmonic with continuous first order partial derivatives. By the inverse relations (2.1.7b) and the fact that $\phi(\mathcal{X},\mathcal{Y})$ and $\psi(\mathcal{X},\mathcal{Y})$ are harmonic conjugates, i.e., Cauchy-Riemann's conditions hold, it follows that

$$F = 0 \quad . \tag{3.1.1}$$

Equation (3.1.1) simplifies the problem but also narrows the freedom of specification of other variables; for example the value of $\mathcal{X}(\phi,0)$ over the profile cannot arbitrarily be assigned. However, this course of action seems intuitively sensible and is followed. Another physical requirement because of the flow assumptions is that of zero circulation over any closed loop ℓ containing the profile

$$C = 0 = \int_\ell \underset{\sim}{q} \bullet d\underset{\sim}{r} \quad . \tag{3.1.2}$$

From (3.1.1) and relations (2.1.11) one obtains

$$\underset{\sim}{q} \bullet d\underset{\sim}{r} = \frac{1}{G^{1/2}} \urcorner(\cos\alpha, \sin\alpha) \bullet (d\mathcal{X}, d\mathcal{Y})$$

$$= \frac{1}{G^{1/2}}(\cos\alpha, \sin\alpha)(\sqrt{E}\cos\alpha d\phi - \sqrt{G}\sin\alpha d\psi, \sqrt{E}\sin\alpha d\phi + \sqrt{G}\cos\alpha d\psi)$$

$$= \frac{E^{1/2}}{G^{1/2}}\, d\phi \quad . \qquad (3.1.3)$$

If (3.1.2) is integrated around the boundary of the finite computational domain (see Figure 3) then using (3.1.3),

$$0 = \frac{E_\infty}{G_\infty}\left[\mathcal{L}_2 - \mathcal{L}_1\right] - \int_{\mathcal{L}_1}^{\mathcal{L}_2} \frac{E^{1/2}}{G^{1/2}}\, d\phi \qquad (3.1.4)$$

$$\mathcal{L}_2 = \phi_{\max} \quad , \quad \mathcal{L}_1 = \phi_{\min} \quad , \quad \frac{E_\infty}{G_\infty} = \text{constant} = C_\circ \; .$$

Now (3.1.4) is independent of ϕ since part of the loop may enclose any streamline and therefore

$$\frac{E^{1/2}}{G^{1/2}} = g'(\phi) \quad , \quad g(\mathcal{L}_2) = C_\circ\, \mathcal{L}_2 \quad , \quad g(\mathcal{L}_1) = C_\circ\, \mathcal{L}_1 \qquad (3.1.5)$$

for some differentiable function $g(\phi)$. The same result can be obtained from the vorticity (2.1.19b) assuming $F=\omega=0$. The function g, because of zero circulation, is taken to satisfy the condition

$$\lim_{|\phi|\to\infty} g'(\phi) < \infty \quad .$$

Using (3.1.1) in (2.1.17d) yields

$$\left[\frac{E_\psi}{(EG)^{1/2}}\right]_\psi + \left[\frac{G_\phi}{(EG)^{1/2}}\right]_\phi = 0 . \qquad (3.1.6)$$

By (3.1.4), (3.1.6) becomes

$$g'(\phi)\left[\frac{G_\psi}{G}\right]_\psi + \left[\frac{G_\phi}{g'(\phi)G}\right]_\phi = 0 . \qquad (3.1.7)$$

Two possible approaches considered to find the function g are given below:

(I) The first method is to follow the physics of the problem: It immediately follows that E=G throughout the flow since ϕ and ψ are harmonic conjugates;

$$x_\phi = y_\psi \quad , \quad x_\psi = - y_\phi \quad . \qquad (3.1.8)$$

Thus g'(ϕ) = 1. Inspection of (2.1.19d), however, reveals that

$$q \to 0 \leftrightarrow G \to \infty \quad \text{or} \quad \theta \to 0, \Pi$$

where θ denotes the angle of intersection of the curves ϕ=const and ψ=constant. The latter case is not in general true at stagnation points. Thus it follows E → ∞ at these points in order that E=G. Therefore W → ∞ also at the leading and trailing edge. The inverse relations given by (2.1.7b) when using differenced forms for the first partial derivatives of x and y are therefore, not necessarily valid in a neighbourhood of these two points since x and y are necessarily bounded monotonic functions of ϕ and ψ. This follows since x and y are harmonic conjugates as functions of ϕ and ψ. Thus the Min-Max Modulus theorem applies [13]. In the neighbourhood of a stagnation point, therefore, knowing values of x and y is not necessarily sufficient to achieving an accurate solution. This further points to using (3.1.8) to find E.

(II) The second approach is to calculate g'(ϕ) by iteration. However, this would involve knowing one of x or y along the profile itself, and for the reasons stated above instabilities result; a fact borne out by numerical experimentation. Iteration was tried unsuccessfully and was deemed to be unnecessary for this particular type of problem. For this reason only approach (I) was considered.

Using g'(ϕ)=1 (3.1.7) reduces to

$$\nabla^2 \ln E = 0 \quad , \quad \nabla^2 = \frac{\partial^2}{\partial \phi^2} + \frac{\partial^2}{\partial \psi^2} \quad . \tag{3.1.9}$$

It can easily be shown that α, x, y also satisfy Laplace's equation.

3.2 NUMERICAL ALGORITHM, BOUNDARY CONDITIONS

To solve the problem numerically, i.e., by finite difference methods, a grid needs to be constructed such that the solution of the flow equations on finitely many grid points will give a reasonable flow representation. In the present study the grid coordinate system has the streamlines as one set of curves and for the other, the orthogonal trajectories (see Figure 3). The grid points may be considered as ordered pairs (J,K) where J denotes ϕ direction and K denotes the ψ direction.

The computational domain is defined for our purpose as

$(\phi,\psi) \subseteq [\ -4,4\ ] \times [\ 0,\ 4\ ]$. (3.2.1)

The grid size is taken to be 85 x 43 or 64 x 43. The step size for the ϕ and ψ directions, respectively, is

$$\frac{\phi_{\max} - \phi_{\min}}{J_{\max} - J_{\min}} , \quad \frac{\psi_{\max} - \psi_{\min}}{K_{\max} - K_{\min}} .$$

Line SOR is used to solve (3.1.9) and other equations, which have a simple tridiagonal form when central differenced.

Integration of (2.1.11), to within arbitrary constants, subject to $E_\infty = G_\infty =$const yields

$$x_\infty(\phi,\psi) = \left[E_\infty\right]^{1/2} \cos\alpha_\infty \phi - \left[E_\infty\right]^{1/2} \sin\alpha_\infty \psi \qquad (3.2.2a)$$

$$y_\infty(\phi,\psi) = \left[E_\infty\right]^{1/2} \sin\alpha_\infty \phi + \left[E_\infty\right]^{1/2} \cos\alpha_\infty \psi \qquad (3.2.2b)$$

where α_∞ denotes the angle of the incoming flow. To express (3.2.2) in terms of speed we observe from (2.1.19d) that

$$q = \frac{1}{E^{1/2}} .$$

To describe the incoming flow it is only necessary to specify q_∞ and α_∞ . Uniform parallel flow gives

$$q_\infty = 1 \quad , \quad \alpha_\infty = 0 \quad , \quad x_\infty = \phi \quad , \quad y_\infty = \psi \quad . \qquad (3.2.3)$$

Along the streamline $\psi = 0$ it is necessary to use derivative or Von Neumann boundary conditions. Numerical experimentation indicates that the first of equations (2.1.12a) is appropriate for (3.1.9) when the profile is given. This is necessary as (3.1.9) was derived from a integrability condition involving α_ϕ. By the results E=G, F=0, (2.1.12a) becomes

$$\alpha_\phi = -\frac{E_\psi}{2E} \tag{3.2.4a}$$

$$\alpha_\psi = \frac{E_\phi}{2E} \quad . \tag{3.2.4b}$$

To apply boundary condition (3.2.4a) on the coordinate line $\psi=0$ there is the option of specifying $\alpha(\phi_J,0)$ for J = 1,...,JMAX and determining the resultant profile by integration of (3.2.4a) using either (2.2.7b) or by using a differenced form for α_ϕ. We also have

$$\left[\frac{dy}{dx}\right]_J = \tan\alpha_J \quad , \quad y(x_{LE}) = y(x_{TE}) = 0 \tag{3.2.5}$$

in which

$$(dx)_J = x(\phi_{J+1},0) - x(\phi_{J-1},0) \quad .$$

This approach may be advantageous in that a certain velocity profile may be obtained by merely altering the angle at various grid points $(\phi_J,0)$. The above two equations form a system of two ordinary differential equations for which a unique solution exists provided certain conditions are met. The method of solving them utilizes Newton's method for determining the zeros of vector systems as well as solving tridiagonal matrices (cf. [14)] and Chapter 6. Alternatively, given a profile, $\alpha(\phi_J,0)$ can be determined during iteration. Values of $x(\phi_J,0)$ are also required since

$$\alpha(\phi_J\ ,0) = \tan^{-1}[\mathcal{I}'(\mathcal{X}(\phi_J\ ,0))] \quad . \qquad (3.2.6)$$

A simple algorithm solves (3.1.9), (3.2.4a), and (3.2.6)

(i) Constuct a computational domain with grid size and grid spacing, i.e., $D\phi$, $D\psi$, JMAX, JMIN, KMAX, KMIN.

(ii) Initialize $\alpha(\phi_J,0)$ to 0 for J = 1,..., JMAX. Set T = $\frac{\ln E}{2}$, $T_{J,K}$ = 0 and set $\mathcal{X}_{J,K}$=0 for all grid points (J,K).

(iii) Central difference all derivatives and solve the Laplace equation $\nabla^2 T = 0$ by line-SOR with T_∞=0 , T_ψ=$-\alpha_\phi$ on ψ=0, in both the ϕ and ψ directions; once each per iteration. In the interior of the computational domain $\nabla^2 T=0$ by application of central differencing becomes

$$T_{J,K} = \frac{1}{4}\Big[\ T_{J+1,K} + T_{J-1,K} + T_{J,K+1} + T_{J,K-1}\ \Big]$$

such that $D\phi$ = $D\psi$. In the case of SLOR the tridiagonal matrix equation $B(-1,4,-1)\underset{\sim}{T}_J = \underset{\sim}{b}_J$ is directly inverted in the K (respectively J) direction for K=2,...,KMAX-1 where

$$\underset{\sim}{b}_J = T_{J+1,K} + T_{J-1,K} \text{ for } K > 1 \quad .$$

When using the index K,

$$\underset{\sim}{b}_K = T_{J,K+1} + T_{J,K-1} \text{ for } 2 \le J \le \text{JMAX}-1 \quad .$$

For K=1 or ψ=0, $T_{\psi\psi}$ is replaced by the second order differencing [3]

$$\Big[T_{\psi\psi}\Big]_{J,1} = \frac{2}{D\psi^2}\Big[\ T_{J,2} - T_{J,1} - D\psi T_\psi\Big] + O\Big(D\psi^2\Big) .$$

(iv) Specify $\chi(0,0) = \chi_0$. After each iteration, using a one sided differencing in (2.1.11a) find $(d\chi)_J$ for

$$J = \frac{JMAX+1}{2},\ldots,JMAX \quad \text{and} \quad 1,\ldots,\frac{JMAX+1}{2}.$$

From sums of these $(d\chi)_J$ determine $\chi(\phi_J,0)$ for J=1,...,JMAX. A simple one-sided difference given by

$$\chi_{J+1} = \chi_J + D\phi\ (E^{1/2}\cos\alpha)_J$$

for forward differencing, and

$$\chi_{J-1} = \chi_J - D\phi\ (E^{1/2}\cos\alpha)_J$$

for backward differencing is found sufficient. The truncation error can be signifigantly reduced by using central differencing alternately with foward and backward differencing, respectively.

(v) Calculate $\alpha(\phi_J,0)$ from (3.2.6), J = 1,...,JMAX, such that α=0 if

$$\chi \le \chi_{LE} \quad \text{or} \quad \chi \ge \chi_{TE} .$$

Continue iterating until convergence.

This algorithm is found to be convergent, but values for χ and hence α do not have sufficient accuracy. In order to more accurately determine χ on the airfoil surface it is necessary to solve for χ using $\nabla^2\chi = 0$ throughout the entire flow field with boundary conditions at ∞ given by (3.1.2a). Along the streamline ψ=0,

$$\chi_\psi = - E^{1/2}\sin\alpha$$

is used. Thus step (iv) in the above algorithm becomes identical to (iii) with T replaced by χ and the above boundary conditions. T and χ are then solved simultaneously. Although more calculations are required, better accuracy is obtained. A computer listing is available (cf. Appendix A).

3.3 RESULTS

The NACA 0012-64 profile was tested with the algorithm of the previous section. The leading and trailing edge was shifted to $\chi=-.5$, $\chi=.5$ respectively so that (2.2.4) becomes,

$$\pm y_t(\chi) = \frac{t}{.2}\left\{.2969(\chi+.5)^{\frac{1}{2}} - .126(\chi+.5)\right.$$

$$\left. - .3516(\chi+.5)^2 + .2843(\chi+.5)^3 - .1015(\chi+.5)^4\right\} .$$

The constant t denoting the maximum thickness expressed as a fraction of the chord for this profile is given by [4]

$$t = \left(\frac{.01582}{1.1019}\right)^{\frac{1}{2}} .$$

The results are given in Table I and compared with Theodorsen's method (Table II) and the grid generation method (Table III), as well as graphically in Figure 6.

Comparison of Table I and Table II reveals accuracy is of first order. By first order, it is meant that the solution is within $D\phi$ of the standard solution (Theodorsen). Our solution was obtained using SLOR with an optimal acceleration parameter iteratively adjusted [19]. The technique of obtaining the optimal iteration parameter for SLOR is explained in Chapter 4.

The grid size was 85 × 43 , $(\phi,\psi) \in [-1.5,1.5] \times [0,1.5]$ with a tolerance level of 1×10^{-5}. 115 iterations were required for convergence under the maximum norm, which means that the maximum modulus between $T^{n}_{J,K}$ and $T^{n+1}_{J,K}$ is less than a prescribed number for J=2,...,JMAX-1 and K=1,...,KMAX-1.

This problem can also be solved by SOR with a fixed iteration parameter and requires only 87 iterations. Also using SOR on a 50 × 50 O-type grid [1] with a circular outer boundary of radius 2 using elliptic grid generation, 152 iterations were required for x and y and 162 iterations for ψ, solved separately. The accuracy is of the same order as the method proposed in this investigation, but clearly the operational count (multiplication and divisions) indicates this new approach is much more efficient. Our approach required roughly one-sixth as many operations (from approximately 12M to 2M for SOR). It should be noted that the results obtained by the grid generation approach were obtained by the author using the method described in detail in [1]. The solution for the ellipse

$$y = \frac{1}{2}(1 - x^2)^{\frac{1}{2}}$$

as shown in Figure 7, compares favourably with the exact solution.

Through experimentation it was also found that an exact form of (3.2.4a) yielded a solution which apparently is valid only over the centre portion of the airfoil. This form is obtained from using (2.2.7b) in (3.2.4a), to get

$$T_\phi = \frac{-f''(x)\, e^T}{\left\{1 + f'^2(x)\right\}^{3/2}} = -\kappa\, e^T \qquad (3.3.1)$$

where κ denotes the signed curvature of the airfoil. Figures 4 and 5 give the velocity profiles for the circle and Joukowski airfoil using (3.3.1). The grid sizes were respectively

$$43 \times 43 \quad , \quad [\,-1,\ 1\,] \times [\,0,\ 2\,] \quad ,$$

and

$$85 \times 43 \quad , \quad [\,-2,\ 2\,] \times [\,0,\ 2\,] \quad ,$$

for

$$f(x) = \left\{.25 - x^2\right\}^{1/2} \quad , \quad f(x) = \frac{1}{2}\left\{1 - x^2\right\}^{1/2}(1-x) \quad .$$

The former equation in the above is obtained simply by setting ϵ=1, a=.5 and b=.5 in (2.2.1). The curvature of the circle is found to be 2 from (2.2.12). In using the exact form (3.3.1) the jump in T_ϕ is not accurately predicted in a neighbourhood of ~~the~~ LE or TE. This is because T_ϕ=0 at the grid points $\phi(J_{LE}-1,1)$ and $\phi(J_{TE}+1,1)$ which is not the case when the differenced form of α_ϕ in (3.2.4a) is used. This accounts for the much higher values for velocity in these regions since T is continuous across the LE and TE. However, accurate values were obtained over the main body of the airfoil which is not surprising since the relation is exact; however this may not always hold. In the case of the NACA-0012-64 less accurate results were produced by using (3.3.1). The inaccuracy in this case is believed due to the small extent of the profile in the ψ direction as compared with the circle and Joukowski airfoil. Hence the rate of change of curvature is high resulting in greater possibility of error in the determination of T_ψ. Thus, we see that in general, using the exact form (3.3.1) for the boundary conditions, results in less overall accuracy, but may be necessary in certain cases; while using the differenced form will generally produce good results except in extreme cases, i.e., singular points at the leading or trailing edge. By differencing α_ϕ across the LE and TE we assume that

$$T_\psi = \alpha_{LE}\ O\left[\frac{1}{D\phi}\right] \quad . \tag{3.3.3}$$

This is a reasonable approximation in a small neighbourhood of the LE(TE) provided α_{LE} is not 0. It is clear that using (3.3.1) as the boundary condition may be appropriate when the curvature of the profile varies greatly. In this case the differenced form of α_ϕ may introduce a large error because of the high curvature. For example the Joukowski airfoil with $\epsilon = \frac{1}{2}$ in (2.2.3) is given by

$$f(\chi) = \frac{1}{2}\left[1 - \chi^2\right]^{1/2}(1 - \chi) \quad . \qquad (3.3.4a)$$

It can easily be shown by L'Hopital's rule that

$$\lim_{\chi \to \chi_{TE}} \left|\kappa(\chi)\right| = +\infty \quad , \quad \alpha_{TE} = 0 \qquad (3.3.4b)$$

where the signed curvature from (2.2.15) is given by

$$\kappa(\chi) = \frac{\frac{1}{2}\,(2\chi^2+2\chi-1)\,(1-\chi)}{\left(1-\chi^2+\frac{1}{4}\left(2\chi^2-\chi-1\right)^2\right)^{1.5}}$$

and the tangent of the angle, by (2.2.13),

$$\tan\alpha = \pm\frac{1}{2}\left[\frac{2\chi^2 - \chi - 1}{(1-\chi^2)^{1/2}}\right] \quad .$$

We also have

$$\kappa(0) = -.326 \quad , \quad \kappa(.99)\; e^T \approx 5.15 \times 1.7$$

$$\frac{\alpha(.99) - \alpha_{TE}}{2D\phi} \approx -1.11 \quad .$$

We see that differencing α_ϕ in boundary condition (3.2.4a) is not as accurate in the case of the Joukowski airfoil because of the rapid change of curvature and because $\alpha_{TE} = 0$; however, the same problem does not occur for the NACA-0012-64 airfoil since $\kappa(\chi)$ is bounded. Thus, as a rule of thumb, if singularities occur at the leading and trailing edge, i.e., infinite curvature, of an arbitrary profile then the exact form, (3.3.1) can be used.

In summary, the proposed method enhances our ability to compute this particular class of flows. The solution given is direct, efficient, economical and relatively easy to code. The results compare favourably with existing methods. It has been indicated where a possible loss of accuracy may occur, for example infinite curvature and zero trailing edge angle. A possible correction has been given. The proposed method is general enough to solve for the flow over an arbitrary symmetric profile. Having understood the essential features and difficulties of this new method it is possible to venture into more complicated areas of Computational Fluid Dynamics to include flow problems with vorticity and compressibility effects.

CHAPTER 4 FLOW WITH VARIABLE VORTICITY

4.1 INTRODUCTION

The formulation numerical algorithm that is developed in Chapters 2 and 3 for steady, incompressible, inviscid, irrotational flows is extended to include flows with vorticity over arbitrary symmetric profiles. The vorticity ω is constant on individual streamlines, so that $\omega_{\phi}=0$ because of the inviscid property of the flow. The choice of vorticity in the far field approximation is arbitrary provided symmetry is preserved about the x-axis. In particular parabolic shear flow is considered over an arbitrary symmetrical shape for which a numerical algorithm is given. It is found that a similar approach as taken in chapter 2 can also be followed for problems of this type. However, as will be seen, its application requires several more considerations, due to the increased number of unknowns and the non-linearity of the problem. Other considerations include stability aspects of the algorithm and the handling of the boundary conditions.

In the present chapter parabolic shear flow over an arbitrary, symmetric two dimensional profile is considered. The flow problem is posed by Van Dyke in [5] and solved by perturbation methods for the case of a circle. Van Dyke's solution was facilitated by a transformation of the governing equations to polar coordinates in which the circle becomes a coordinate line of the transformation. The stream-function obtained by Van Dyke is given by

$$\psi = U_\infty(r-\frac{a^2}{r})\sin\theta+\epsilon U_\infty(\frac{r^3}{6a^2}\sin^3\theta-\frac{r}{2}\log r\sin\theta+x)+O(\epsilon^2) \qquad (4.1.1)$$

in which

$$x = -\frac{a^2}{8r}\sin\theta + \frac{a^4}{24r^3}\sin 3\theta + c(r - \frac{a^2}{r})\sin\theta$$

is used to minimize the disturbance of the flow upstream and where

$$c = \frac{1}{4}(\log\frac{4}{\epsilon} - 2\gamma + 1) \quad , \quad \gamma = .577 \quad .$$

Equation (4.1.1) is the solution of the boundary value problem,

$$\psi_{rr} + \frac{\psi_r}{r} + \frac{\psi_{\theta\theta}}{r^2} = \frac{\epsilon\psi}{a^2} \quad , \quad \psi(a,\theta) = 0 \quad .$$

$$\psi(r,\theta) \simeq U_\infty r \sin\theta + \frac{U_\infty \epsilon r^3}{6a^2}\sin^3\theta + O(\epsilon^2) \text{ as } r \to \infty \ .$$

The velocity components are determined from the derivative relations for $u(r,\theta)$ and $v(r,\theta)$ given in (2.3.2). Pressure can be found from Bernoulli's equation [15],

$$\frac{p}{\rho} + \Omega + \frac{q^2}{2} + \omega\psi = \text{constant}. \qquad (4.1.2)$$

In (4.1.1) and (4.1.2) U_∞ is the upstream velocity along the X axis, p is the pressure, ρ is the density and Ω is the body force potential. The body force is generally taken to be zero for this type of flow.

The transformation to cylindrical coordinates simplifies the boundary conditions since one coordinate r, is constant on the surface of the circle. Vectors normal and tangent to the surface of the cylinder are normal and tangent to the zero streamline because of the flow tangency condition. Thus, the velocity vector is tangential to the coordinate curve r = constant. Generating an orthogonal curvilinear coordinate system about an arbitrary airfoil profile is difficult. Several orthogonal systems are discussed in [16]. The approach taken for the flow problem at hand is along similar conceptual lines as grid generation and the approach taken for the problem in the previous chapter, the essential difference being the addition of the vorticity term. The similarities are that the streamlines are again utilized as one of a pair of coordinate curves while the other coordinate curve is taken to intersect the streamlines at constant angle usually, $\alpha=90^{\circ}$. In the present problem, however, there is no velocity potential to work with. In the following section we begin with the system of equations (2.1.19) and Gauss' equation.

4.2 EQUATIONS OF MOTION

In non-dimensional form, the system of equations governing the steady, plane flow of a viscous, incompressible fluid with the curvilinear coordinate system (ϕ,ψ) taken as the independent variables and (x,y) as the dependent variables, where ψ=const denote the streamlines, is given by [7],

$$h_\phi = \frac{1}{Re}\frac{F}{W}\omega_\phi - \frac{1}{Re}\frac{E}{W}\omega_\psi \quad ,$$

$$h_\psi = -\omega + \frac{1}{Re}\frac{G}{W}\omega_\phi - \frac{1}{Re}\frac{F}{W}\omega_\psi \tag{4.2.1a}$$

$$\omega = \frac{1}{W}\left[\left(\frac{F}{W}\right)_\phi - \left(\frac{E}{W}\right)_\psi\right] \tag{4.2.1b}$$

$$K = \frac{1}{W}\left[\left(\frac{W}{E}\Gamma^2_{11}\right)_\psi - \left(\frac{W}{E}\Gamma^2_{12}\right)_\phi\right] = 0 \tag{4.2.1c}$$

$$h = \frac{E}{2W^2} + p \tag{4.2.1d}$$

$$u^2 + v^2 = \frac{E}{W^2} \tag{4.2.1e}$$

where $W = EG - F^2 > 0$.

Equations (4.2.1a) are determined by solving (2.1.19a) for h_ϕ and h_ψ. The above denote respectively the Navier-Stokes, vorticity, Gauss, and energy equations. h, ω and p are respectively the energy, vorticity and pressure functions. Martin [7] has shown the continuity equation

$$\underset{\sim}{\nabla} \cdot \underset{\sim}{u} = 0$$

is equivalent to (4.2.1e), where $\underset{\sim}{u}$ is the two dimensional velocity vector. Taking the Jacobian J positive in the region of interest means we consider the streamlines to be directed in the sense of increasing ϕ. E, F and G are taken to be the coefficients of the first fundamental form

$$ds^2 = dx^2 + dy^2 = d\underset{\sim}{r}\cdot d\underset{\sim}{r} \tag{4.2.2a}$$

$$= E(\phi,\psi)d\phi^2 + 2F(\phi,\psi)d\phi d\psi + G(\phi,\psi)d\psi^2$$

such that

$$\underset{\sim}{r} = \underset{\sim}{r}(\phi,\psi) \tag{4.2.2b}$$

is the position vector of a point $\underset{\sim}{r}=(x,y)$ in Cartesian coordinates. Summarizing other geometrical results as outlined in section 2.1,

$$W = EG - F^2 = \left| \underset{\sim}{r}_\phi \times \underset{\sim}{r}_\psi \right| \tag{4.2.2c}$$

$$E = \underset{\sim}{r}_\phi\cdot\underset{\sim}{r}_\phi \ , \ F = \underset{\sim}{r}_\phi\cdot\underset{\sim}{r}_\psi \ , \ G = \underset{\sim}{r}_\psi\cdot\underset{\sim}{r}_\psi \tag{4.2.2d}$$

$$\underset{\sim}{r}_\phi = W(\psi_y \ , \ -\psi_x) = E^{\frac{1}{2}}(\cos\alpha,\sin\alpha) \tag{4.2.2e}$$

$$\underset{\sim}{r}_\psi = W(-\phi_y \ , \ \phi_x) = \frac{F}{E^{\frac{1}{2}}}(\cos\alpha,\sin\alpha) - \frac{W}{E^{\frac{1}{2}}}(\sin\alpha,-\cos\alpha) \tag{4.2.2f}$$

$$\alpha_\phi = \frac{W}{E}\Gamma^2_{11} \quad , \quad \alpha_\psi = \frac{W}{E}\Gamma^2_{12} \tag{4.2.2g}$$

$$\Gamma^2_{11} = \frac{1}{2W^2}\left[- FE_\phi + 2EF_\phi - EE_\psi \right] \tag{4.2.2h}$$

$$\Gamma^2_{12} = \frac{1}{2W^2}\left[EG_\phi - FE_\psi \right] . \qquad (4.2.2i)$$

$\alpha(\phi,\psi)$ represents the local angle of inclination of the tangent to the coordinate line ψ=constant directed in the sense of increasing ϕ. Non-dimensional (unbarred) variables are related to the physical (barred) variables by the relations,

$$\bar{h} = \rho\, u_\infty^2\, h \quad , \quad \bar{\omega} = \frac{u_\infty}{L}\,\omega \quad , \quad \bar{u} = u_\infty\, u \quad ,$$

$$\bar{v} = u_\infty\, v \quad , \quad \bar{x} = Lx \quad , \quad \bar{y} = Ly$$

for characteristic length L and speed u_∞. Re $= \rho \dfrac{u_\infty L}{\mu}$ is the Reynold's number.

In the previous problem it was seen that the coefficient F vanished necessarily because of the existence of a velocity potential ϕ. In the present problem a potential ϕ does not exist.

Once again, the system (4.2.1) contains one more unknown than the number of equations because of the arbitrariness of the curves ϕ=constant. Taking F=0, the system of equations (4.2.1) is determinant and becomes

$$h_\phi = -\frac{1}{Re}\frac{E}{W}\,\omega_\psi \quad ,$$

$$h_\psi = -\,\omega + \frac{1}{Re}\frac{G}{W}\,\omega_\phi \qquad (4.2.3a)$$

$$\omega = -\frac{1}{W}\left(\frac{E}{W}\right)_\psi \qquad (4.2.3b)$$

$$\left(\frac{E_\psi}{W}\right)_\psi + \left(\frac{G_\phi}{W}\right)_\phi = 0 \qquad (4.2.3c)$$

$$h = \frac{1}{2G} + p \qquad (4.2.3d)$$

$$u^2 + v^2 = \frac{1}{G} \qquad (4.2.3e)$$

such that $W = (EG)^{½}$.

4.3 INVISCID ROTATIONAL FLOWS

In this section we look at one of two shear flows, derive the boundary conditions, transform the governing equations into a more suitable form and investigate the stability of the equations with respect to the numerical method that will be used. We consider flows with upstream velocity profile given by

$$\left(\psi_y\right)_\infty = u(x_\infty\ ,y) = u_\infty \cosh(Cy) \qquad (4.3.1a)$$

or

$$\left(\psi_y\right)_\infty = u(x_\infty\ ,y) = u_\infty\left(1 + \frac{C^2y^2}{2}\right) \ . \qquad (4.3.1b)$$

Equation (4.3.1b) is truly parabolic in nature. For the case of a circle Van Dyke [5] has furnished a pair of solutions for upstream profiles (4.3.1a), (4.3.1b), i.e., (4.1.1) with the x term and without the x term respectively. In Van Dyke's solution

$$C = \frac{\varepsilon^{1/2}}{a} = \text{constant}$$

where a is the circle radius, ε is a perturbation parameter. The vorticity profiles of (4.3.1a), (4.3.1b) are found by taking the Laplacian,

$$\nabla^2\psi = -\omega = u_\infty C \sinh(Cy) \tag{4.3.2a}$$

$$\nabla^2\psi = -\omega = u_\infty\, C^2 \quad . \tag{4.3.2b}$$

We consider only (4.3.1a) which leads to simpler expressions for $\omega(\psi)$. Integrating (4.3.1a) with respect to y and using (4.3.2a) we obtain

$$\psi = -C^{-2}\omega \quad . \tag{4.3.3}$$

The constant of integration vanishes due to symmetry. From (4.3.2a) and (4.3.3) we also get

$$y_\infty = C^{-1}\sinh^{-1}\left[\frac{C\psi}{u_\infty}\right] \quad . \tag{4.3.4a}$$

By the assumption of parallel flow upstream we can take

$$x_\infty = \phi \quad . \tag{4.3.4b}$$

From (4.3.4) and the relations (4.2.2d-g) we get

$$E_\infty = x_{\infty\phi}{}^2 + y_{\infty\phi}{}^2 = 1 \tag{4.3.4c}$$

$$G_\infty = x_{\infty\psi}{}^2 + y_{\infty\psi}{}^2 \tag{4.3.4d}$$

$$= \frac{1}{u_\infty{}^2}\left[1 + \frac{C^2\psi^2}{u_\infty^2}\right]^{-1} .$$

Equation (4.3.4) are the essential far field approximations.

to $\mathcal{X}$, $\mathcal{Y}$, E, and G. As in Chapter 2, matters can be simplyfied by the introduction of a transformation,

$$T = \frac{1}{2}\ln E \quad , \quad U = \frac{1}{2}\ln G \quad . \tag{4.3.5}$$

Equations (4.3.4) is replaced by

$$T_\infty = 0 \tag{4.3.6a}$$

$$U_\infty = -\ln \mathcal{U}_\infty - \frac{1}{2}\ln\left(1 + \frac{C^2\psi^2}{\mathcal{U}_\infty^2}\right) . \tag{4.3.6b}$$

We now transform the governing equations (4.2.3) according to (4.3.5) and the inviscid assumption ($Re^{-1}=0$), derive the equation for the pressure coefficient and establish the Von Neumann boundary conditions on the airfoil surface to be used in the numerical calculations. Equations (4.2.3) are replaced by

$$h_\phi = 0 \tag{4.3.7a}$$

$$h_\psi = -\omega \tag{4.3.7b}$$

$$U_\psi - T_\psi = e^{2U}\omega \tag{4.3.7c}$$

$$e^{T-U}T_{\psi\psi} + e^{U-T}U_{\phi\phi} + e^{T-U}T_\psi\,(T_\psi - U_\psi) \tag{4.3.7d}$$

$$+ e^{U-T}U_\phi\,(U_\phi - T_\phi) = 0$$

$$h = \frac{1}{2}e^{-2U} + p \tag{4.3.7e}$$

$$u^2 + v^2 = e^{-2U} \quad . \tag{4.3.7f}$$

Integrating (4.3.7a-b) and using (4.3.3) yields

$$h(\phi,\psi) = \frac{1}{2}(C\psi)^2 + h_\circ \quad . \qquad (4.3.8a)$$

It follows that $h_\circ=0$ by symmetry. Using (4.3.8a) in the energy (4.3.7e) we obtain

$$\frac{1}{2}(C\psi)^2 = \frac{1}{2G} + p \quad . \qquad (4.3.8b)$$

Letting ϕ tend to ∞ yields

$$\frac{1}{2}(C\psi)^2 = \frac{1}{2G_\infty} + p_\infty \quad . \qquad (4.3.8c)$$

Subtracting (4.3.8b) and (4.3.8c) we obtain

$$2(p-p_\infty) = G_\infty^{-1} - G^{-1}$$

$$= \left\{ \mathcal{U}_\infty^2 + C^2\psi^2 \right\} - \frac{1}{G} \quad . \qquad (4.3.8d)$$

From (4.2.2g-i) we find

$$\alpha_\phi = - e^{T-U} T_\psi \qquad (4.3.9a)$$

$$\alpha_\psi = e^{U-T} U_\phi \quad . \qquad (4.3.9b)$$

Taking the total derivative of $\alpha(\mathcal{X},\mathcal{Y})$ with respect to $\mathcal{X}$ and expanding α_ϕ in (4.3.9a) by the chain rule, one can obtain, using the flow tangency condition and (4.2.2f-g),

$$T_\psi = \frac{- e^U \mathcal{f}''(\mathcal{X})}{\left[1 + \mathcal{f}'^2(\mathcal{X}) \right]^{3/2}} \quad . \qquad (4.3.9c)$$

Equation (4.3.9a) or (4.3.9c) is incorporated into the solution procedure along the coordinate curve $\psi=0$. In the iterative process $\mathcal{f}'(\mathcal{X})$, $\mathcal{f}''(\mathcal{X})$ are determined from the respective profile.

Equations (4.3.7c), (4.3.7d) represent two non-linear equations in two unknowns and the question arises as to existence and uniqueness of a solution. From (4.2.2e), (4.2.2f), (4.3.9a), (4.3.9b) (F=0), by taking cross derivatives, to eliminate the other variable one can arrive at self-adjoint second order non-linear partial differential equations of the form

$$\left[e^{U-T}\alpha_{\phi}\right]_{\phi} + \left[e^{T-U}\alpha_{\psi}\right]_{\psi} = 2\omega e^{2U}U_{\phi} \qquad (4.3.10a)$$

$$\left[e^{U-T}\chi_{\phi}\right]_{\phi} + \left[e^{T-U}\chi_{\psi}\right]_{\psi} = 0 \qquad (4.3.10b)$$

for which existence and uniqueness theorems exist [12] in the case of linearized coefficients e^{T-U}, e^{U-T}, and either Dirichlet or Von Neumann boundary conditions. Thus any solution to (4.3.7c), (4.3.7d) leads to a unique solution of (4.3.10a) or (4.3.10b). Furthermore T is determined uniquely by U in (4.3.7c) as a boundary value problem provided certain conditions are met in a closed domain, i.e., boundedness of U and Lipschitz condition on a closed region, [14]. Using (4.3.7c) in (4.3.7d) we obtain

$$e^{T-U}U_{\psi\psi} + e^{U-T}U_{\phi\phi} - 3\omega U_{\psi}e^{T+U} - \overset{U}{\omega}_{\psi}e^{T+U} \qquad (4.3.11a)$$

$$+ \omega^{2}\, e^{T+3U} + e^{U-T}U_{\phi}(U_{\phi}-T_{\phi}) = 0$$

$$e^{T-U}\chi_{\psi\psi} + e^{U-T}\chi_{\phi\phi} - e^{T+U}\chi_{\psi}\omega + e^{U-T}\chi_{\phi}(U_{\phi}-T_{\phi}) = 0 \;. \qquad (4.3.11b)$$

The intention is to follow as closely as possible the algorithm of Chapter 2. Equations (4.3.11) thus represent respective PDEs in U and χ. However, as will be seen, their non-linear nature increases the difficulty of finding an accurate, stable solution and also increases the number of unknowns by one since T and U, and hence E and G are related in a non-linear fashion. Nevertheless, it is worthwhile to utilize fully our previous knowledge and a solution to the problem is thus attempted in a similar manner as the previous problem. The inherent difficulties in convergence and stability can therefore be analysed bearing in mind the limitations of the technique used. It would thus seem prudent to invesigate stability aspects of the particular numerical technique chosen.

4.4 SUMMARY OF ADI TECHNIQUE

Two numerical techniques were used in the solution of the problem: SLOR with fixed or optimal acceleration parameter β and ADI (Alternating Direction Implicit) with parameter ρ. It was found that SLOR was more stable and reliable for such a problem even with only a fixed parameter β. The ADI was erratic in converging and highly sensitive to changes in ρ. Moreover where SLOR converged ADI in many cases diverged especially when using boundary conditions with a high gradient at one or more locations. ADI was found to be less successful in solving equations with non-linear, Von Neumann boundary conditions and more successful when solving only equations with exact boundary conditions. For these reasons a stability analysis based on a trigonometric approximation to the error is presented in Chapter 5.

The ADI technique is discussed in [17] with an attempt to find optimal parameters. However, the matter of coupled equations is not discussed nor do the examples include derivative boundary conditions. In simplest terms the ADI technique utilizes a positive parameter ρ to increase the diagonal dominance of the system of matrix equations determined by finite difference approximation to the respective system of PDE. The model equation that is used in the study in [17] is a linear self-adjoint PDE,

$$G(x,y)u - \frac{\partial}{\partial x}\left[A(x,y)\frac{\partial u}{\partial x}\right] - \frac{\partial}{\partial y}\left[C(x,y)\frac{\partial u}{\partial y}\right]$$
$$= S(x,y). \qquad (4.4.1a)$$

Equation (4.4.1a) is subsequently transformed to an algebraic system of equations,

$$(H + V + \Sigma)\underset{\sim}{u} = \underset{\sim}{\kappa} \qquad (4.4.1b)$$

where

$$Hu(x,y) = -a(x,y)u(x+h,y) + 2b(x,y)u(x,y) - c(x,y)u(x-h,y)$$
$$Vu(x,y) = -\alpha(x,y)u(x,y+k) + 2\beta(x,y)u(x,y) - \gamma(x,y)u(x,y-k)$$
$$a = \kappa A(x+\frac{h}{2},y)/h \ , \ c = \kappa A(x-\frac{h}{2},y)/h \ , \ 2b = a + c$$
$$\alpha = hC(x,y+\frac{k}{2})/k \ , \ \pi = hC(x,y-\frac{k}{2})/k \ , \ 2\beta = \alpha + \gamma$$
$$\Sigma = hkG(x,y) \ , \ \underset{\sim}{\kappa} = hkS(x,y) \ . \qquad (4.4.1c)$$

Equation (4.4.1b) is then rewritten as two vector equations

$$(H + \Sigma + D)\underset{\sim}{u} = \underset{\sim}{K} - (V-D)\underset{\sim}{u}$$

$$(V + \Sigma + E)\underset{\sim}{u} = \underset{\sim}{K} - (H-E)\underset{\sim}{u}$$

taking $H+\Sigma+D$ and $V+\Sigma+E$ to be non-singular. The stationary ADI process is then defined to be [17],

$$(H + \Sigma + D)u_{n+\frac{1}{2}} = K - (V-D)u_n$$

$$(V + \Sigma + E)u_{n+1} = K - (H-E)u_{n+\frac{1}{2}} \qquad (4.4.1d)$$

and is convergent provided $\Sigma+D+E$ is symmetric and positive definite and $2H+\Sigma+D-E$, $2V+\Sigma+E-D$ are positive definite. With the same hypotheses and,

$$\rho_1 , \rho_2 > 0 \quad ; \quad \theta_1 , \theta_2 \in [0,2] \quad ;$$

$$\theta_3 = 2 - \theta_1 \quad ; \quad \theta_4 = 2 - \theta_2$$

the following iterative system is convergent,

$$(H + \frac{\theta_1\Sigma}{2} + \rho_1 I)u_{n+\frac{1}{2}} = K - (V + \frac{\theta_3\Sigma}{2} - \rho_1 I)u_n$$

$$(V + \frac{\theta_2\Sigma}{2} + \rho_2 I)u_{n+1} = K - (H + \frac{\theta_4\Sigma}{2} + \rho_2 I)u_{n+\frac{1}{2}} . \qquad (4.4.1e)$$

Equation (4.4.1e) is the model used in the solution of (4.3.11a). Differences are in the non-linearity of (4.3.11a). For this reason no approximations are made for the coefficients exp(U+T), exp(U-T), etc. at half grid locations as in (4.4.1c). Instead, the equation is solved as is. Some terms, it will be seen, can be dropped for economy of computational count.

The precise matrix forms are given in a later section of this chapter. Some of the information from that section is now used to examine stability aspects of (4.3.11a) with respect to the ADI method.

It is first necessary to view (4.3.11a) in terms of a linear system of algebraic equations. Hence a coefficient matrix A is sought. This is found by Taylor series approximations of the various derivatives in (4.3.11a) in the ψ and ϕ directions with respective grid sizes $D\psi$ and $D\phi$. Generally the grid spacing is the same in both directions. Franklin [18] has shown that a matrix A is stable if and only if every solution to the differential equation

$$\frac{d\chi(t)}{dt} = A\chi(t) \qquad (4.4.2a)$$

approaches zero as t → ∞. If A has n distinct eigenvalues λ_i with associated eigenvectors ω_i then the solution to (4.4.2a) is given by [18],

$$\chi(t) = e^{\lambda_i t}\omega_i \quad . \qquad (4.4.2b)$$

The solution is easily seen to be stable if and only if λ_i is negative for each i. Applying the ADI technique in the ψ direction to (4.3.11a) leads to a tridiagonal matrix,

$$\text{TRID}[A,B,C] = \qquad (4.4.2c)$$

$$\text{TRID}\left[-e^{T-U}-1.5\omega e^{T+U}D\psi, \rho+2e^{T-U}, -e^{T-U}+1.5\omega e^{T+U}D\psi\right]$$

for which the eigenvalues are known to be of the form [3]

$$\lambda_J = -b + 2(ac)^{1/2} \cos\left[\frac{\Pi J}{p}\right] \quad , \qquad J = 1, \ldots, p-1$$

$$\simeq -\rho - 2e^{T-U} + 2e^{T-U} \cos\left[\frac{\Pi J}{p}\right]$$

$$\leq -\rho - 2e^{T-U} + 2e^{T-U} \quad . \qquad (4.4.2d)$$

From (4.4.2d) it can be concluded that if

$$\rho > 0 \qquad (4.4.2e)$$

the coefficient matrix (4.4.2c) is stable and this agrees with the result obtained in [17]. In practice however the use of (4.3.9a) as a boundary condition led to divergence in most cases for a given choice of ρ when the simultaneous solution of (4.3.7c) and (4.3.11a) together with (4.3.10b) was attempted. If (4.3.9a) was replaced by (4.3.9c) then convergence occurred if ρ was chosen carefully. However convergence was slow and instabilities often entered when the convergence criteria was made more demanding. Moreover relaxation parameters were required for the ADI. Nevertheless promising results were obtained for some profiles.

4.5 NUMERICAL ALGORITHM

After some numerical experimentation the following algorithm was found to be convergent when using SLOR, for the profiles discussed in section 2.2,

(1) Set the grid sizes, JMIN, JMAX, KMIN, KMAX,

ϕ_{max} , ϕ_{min}

similarily for ψ. Determine grid spacing

$$D\phi_{a} = \frac{\phi_{\max} - \phi_{\min}}{\text{JMAX} - \text{JMIN}} \quad , \quad D\psi = \frac{\psi_{\max} - \psi_{\min}}{\text{KMAX} - \text{KMIN}} .$$

KMIN and JMIN were taken to be unity. Determine the grid points on the rectangular computational domain in terms of ϕ, ψ, usually by some linear interpolating formula for J=1,...,JMAX, K=1,...,KMAX.

(2) Initialize χ_∞, T_∞, U_∞ according to (4.3.4b), (4.3.6). Set χ = T = U = 0 in the interior of the computational domain.

(3) Using ADI or SLOR solve for U everywhere in the flow field using (4.3.11a) and employing (4.3.7c) as the boundary condition along the coordinate line ψ=0 or K=1, for J=2,...,JMAX-1.

(4) Using SLOR, solve for T everywhere using (4.3.7c) with (4.3.9c) or (4.3.9a) as the boundary condition for K=1; J=2,...,JMAX-1, such that for the former boundary condition $T_\psi=0$, K=1, $\chi < \chi_{LE}$, $\chi > \chi_{TE}$. This step could also be iterated until convergence of T between every iteration of U and χ. Convergence of T was found to be extremely fast by SLOR.

(5) Using ADI or SLOR, solve for χ from (4.3.11b) using (4.2.2f) for the boundary condition: $\chi_\psi=-(G)^{1/2}\sin\alpha(\phi,0)$.

(6) Determine $\alpha(\phi,0)$ by

$$\alpha(\chi(\phi),0) = \begin{cases} \tan^{-1}[f'[\chi(\phi),0]] \ , & \chi_{LE} \le \chi \le \chi_{TE} \\ 0 \ , & \text{otherwise} \end{cases}$$

(7) Choose a particular error norm $\|\cdot\|$ and set a tolerance level ε; for each unknown, determine whether

$\|\cdot\|_{J,K} \leq \varepsilon$; J = 2,..., JMAX-1 ; K = 1,..., KMAX-1.

where the dot $\cdot$ denotes unknown.

(8) Iterate steps (3) to (7) until convergence sweeping in both directions ϕ and ψ. The roles of ψ and ϕ are reversed when sweeping in the ϕ direction,

K = 1,...,KMAX-1 , J = 2,...,JMAX-1.

Let algorithm A denote the use of ADI for the solution of U, χ and, SLOR for T with an optimal acceleration parameter β^{n}_{opt}. Let algorithm B denote the use of SLOR throughout in the above algorithm with fixed β and boundary condition (4.3.9a). Let algorithm C be the same as algorithm B except that boundary condition (4.3.9c) is used. A computer listing is available (cf. Appendix B).

It was found that the use of over-relaxation parameters complemented the use of the ADI. Care had to be taken in their choice as the algorithm was sensitive to variations in their size. In certain cases, when using ADI the results were misleading as the maximum error decreased to what looked like an acceptable tolerance level, a minimum, and then began to slowly increase again due to some instability inherent in the use of ADI in the algorithm. SLOR produced results using both boundary conditions with an error bound, the maximum norm, having a value of .0001 in the case of the circle while the same error bound could not be achieved for boundary condition (4.3.9a) using ADI. Optimal relaxation parameters could be used when solving by SLOR. This was found to be indispensible for convergence, otherwise the algorithm converged much too slowly when using ADI for χ and U and SLOR for T. For example, after the nth iteration T, χ and U were updated by

$$T^{n+1}_{J,K} = T^{n}_{J,K} + \alpha^{*} \left[T^{n+1}_{J,K} - T^{n}_{J,K}\right]$$

$$\chi^{n+1}_{J,K} = \chi^{n}_{J,K} + \beta^{*} \left[\chi^{n+1}_{J,K} - \chi^{n}_{J,K}\right] \qquad (4.5.1)$$

$$U^{n+1}_{J,K} = U^{n}_{J,K} + \beta^{n}_{opt}\left[U^{n+1}_{J,K} - U^{n}_{J,K}\right]$$

for J = 2, ..., JMAX-1; K = 1, ..., KMAX-1. Using a fixed acceleration parameter $\beta^{*} \in [1.5,2]$ was found to accelerate convergence. β^{n}_{opt} was determined after each successive double sweep according to the technique described by Carre [19]. This is outlined in the following variation of Carre's algorithm which may be added to step (6) of the previous algorithm :

(6a) For n < 10 set β^{n}=1 and for any n> if

MOD[n,10]≠0 set $\beta^{n+1} = \beta^{n}$.

(6b) After each iteration calculate,

$$S^{n+1} = \sum_{J}\sum_{K}\left|T^{n+1}_{J,K} - T^{n}_{J,K}\right| \quad ,$$

$$R^{n+1} = \frac{S^{n+1}}{S^{n}} \quad .$$

(6c) If $n \geq 10$ calculate $\mathrm{ERR}^{n+1} = \dfrac{\left|\beta^{n+1} - \beta^{n}\right|}{2-\beta^{n}}$.

(6d) If $\mathrm{ERR}^{n+1} \leq .05$ then set $\beta^{n+1} = \beta^{n}$.

(6e) If $n \geq 10$, MOD(n,10)=0, set

$$\beta^{n+1} = \mathrm{EE}^{n+1} - \frac{2-\mathrm{EE}^{n+1}}{4}$$

where

$$\mathrm{EE}^{n+1} = \frac{2}{1+\sqrt{\dfrac{1-(R^{n+1}+\beta^{n}-1)^{2}}{\beta^{n2}R^{n+1}}}} \quad .$$

Steps (6a) - (6e) are the technical details of the approach. The theoretical motivations are explained in [19]. It is also interesting to note that the rate of convergence depends on the error norm used in step (7). The relative error norm,

$$\mathrm{MAX}\left[\frac{\left|T^{n+1}_{J,K} - T^{n}_{J,K}\right|}{\left|T^{n+1}_{J,K}\right|}\right]$$

used in conjunction with steps (6a) - (6e) was found to give a slower rate of convergence than the maximum norm.

4.6 DIFFERENCE NOTATION

In the numerical differencing of (4.3.7c), (4.3.11a), (4.3.11b) central differencing was used wherever possible. No stretching functions were introduced into the equations to pack lines in any one region although this is one area that could be investigated. To illustrate the method of differencing used in the ADI and SLOR methods we consider (4.3.7c) and (4.3.11a). Applying central difference approximation to (4.3.11a) with equal grid spacing yields

$$\begin{aligned}
& e^{(T-U)_{J,K}} (U_{J,K+1} - 2U_{J,K} + U_{J,K-1}) \\
& + e^{(U-T)_{J,K}} (U_{J+1,K} - 2U_{J,K} + U_{J-1,K}) \\
& - \frac{3}{2} \omega_K (U_{J,K+1} - U_{J,K-1}) e^{(T+U)_{J,K}} D\psi \\
& - (\omega_\psi)_K e^{(T+U)_{J,K}} D\psi^2 + (\omega^2)_K e^{(T+3U)_{J,K}} D\psi^2 \\
& + .25e^{(U-T)_{J,K}} (U_{J+1,K}-U_{J-1,K}) \\
& \qquad \times (U_{J+1,k}-U_{J-1,K}-T_{J+1,K}+T_{J-1,K}) = 0 \qquad (4.6.1)
\end{aligned}$$

where J=2,...,JMAX-1 , K=2,...,KMAX-1. Equation (4.6.1) is used to form a tridiagonal matrix,

$$\text{TRID(AA,BB,CC)} \; \underset{\sim}{U} = \underset{\sim}{\text{RHS}} \qquad (4.6.2)$$

to be inverted in either the J(ϕ) or K(ψ) directions. When K=1, $U_{\psi\psi}$ is replaced by [3],

$$U_{\psi\psi}\Big|_{K=1} = \frac{2(U_{J,2}-U_{J,1})}{D\psi^2} - \frac{2U_\psi}{D\psi} \qquad (4.6.3)$$

in which U_ψ is determined by (4.3.7c) and T_ψ is determined by (4.3.9a) or (4.3.9c). For example if (4.6.2) is solved in the K direction we have, by inspection from (4.6.1) for K=1,

$$AA(K) = 0 \qquad (4.6.4a)$$

$$BB(K) = \rho + 2e^{(T-U)_{J,K}} + 3\omega_K\, D\psi\, e^{(T+U)_{J,K}} \quad ,$$

$$CC(K) = -\,e^{(T-U)_{J,K}} - 3\omega_K\, D\psi\, e^{(T+U)_{J,K}} \quad ,$$

$RHS(K) =$

$$e^{(U-T)_{J,K}}\left(U_{J+1,K} - 2U_{J,K} + U_{J-1,K}\right) + \rho U_{J,K}$$

$$+ .25e^{(U-T)_{J,K}}\left(U_{J+1,K} - U_{J-1,K}\right)$$

$$\times\left(U_{J+1,k} - U_{J-1,K} - T_{J+1,K} + T_{J-1,K}\right)$$

$$+ \omega_K^2\, e^{(T+3U)_{J,K}}\, D\psi^2 + \frac{2D\psi\, f''\, e^{T_{J,K}}}{(1+f'^2)^{3/2}}$$

$$-\, 2\omega_K\, e^{T+U}\, D\psi^2$$

For K > 1,

$$AA(K) = -\,e^{(T-U)_{J,K}} + 1.5\omega_K\, D\psi\, e^{(T+U)_{J,K}} \quad , \qquad (4.6.4b)$$

$$BB(K) = \rho + 2e^{(T-U)_{J,K}} \quad ,$$

$$CC(K) = - e^{(T-U)_{J,K}} - 1.5\omega_K\, D\psi\, e^{(T+U)_{J,K}} ,$$

$$RHS(K) =$$

$$e^{(U-T)_{J,K}} \left(U_{J+1,K} - 2U_{J,K} + U_{J-1,K}\right) + \rho U_{J,K}$$

$$+ .25e^{(U-T)_{J,K}} \left(U_{J+1,K} - U_{J-1,K}\right)$$

$$\times \left(U_{J+1,k} - U_{J-1,K} - T_{J+1,K} + T_{J-1,K}\right)$$

$$+ \omega_K^2\, e^{(T+3U)_{J,K}}\, D\psi^2 - (\omega_\psi)_K\, e^{(T+U)_{J,K}}\, D\psi^2$$

where $\rho \in [2,4]$ is the ADI parameter. Equations (4.6.4) are solved for J=2,...,JMAX-1. In the subsequent sweep the directions are changed and similar equations as (4.6.4) are solved, only in the J direction for K=1,...,KMAX-1. Similar equations apply as well for Υ.

In the sequence of operations, step (4) of the algorithm, SLOR is used to calculate T. The numerical differencing of (4.3.7c) results in a matrix equation of the form,

$$\begin{bmatrix} -1 & 1 & 0 & . & . & 0 \\ -1 & 0 & 1 & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & -1 & 0 \end{bmatrix} \begin{bmatrix} T_{J,1} \\ T_{J,2} \\ . \\ . \\ . \\ T_{J,K1} \end{bmatrix}$$

$$= RHS_{J,K}$$

where K1=KMAX-1 and,

$$RHS_{J,1} = -\frac{e^{U_{J,1}} f''(\mathcal{X}_{J,1})\mathcal{D}\psi}{(1+f'^{2}(\mathcal{X}_{J,1}))^{3/2}}$$

or

$$-\frac{1}{2}\left[\alpha_{J+1,1} - \alpha_{J-1,1}\right] e^{u_{J,1}-T_{J,1}} \quad \text{for } J = 2,\ldots,JMAX-1$$

such that the former equality holds for

$$\mathcal{X}_{LE} < \mathcal{X} < \mathcal{X}_{TE}$$

or

$$RHS_{J,1} = 0 \quad \text{for} \quad \mathcal{X} \geq \mathcal{X}_{TE} \, , \, \mathcal{X} \leq \mathcal{X}_{LE} \; .$$

Otherwise

$$RHS_{J,K} = U_{J,K+1} - U_{J,K-1} - 2\mathcal{D}\psi \, \omega_K \, e^{2U_{J,K}}$$

for 1 < K < KMAX , J=2,...,JMAX-1. T was solved only in the K direction. T, U and $\mathcal{X}$ were updated after every iteration according to (4.5.1).

4.7 DISCUSSION OF RESULTS

Different combinations of boundary conditions and use of SLOR and ADI in the algorithm given in the previous section were applied to compute the flow over a circle of radius $\frac{1}{2}$ as well to the Joukowski and NACA-0012-64 airfoils. The parameter ε was given the value .1 while for all cases 'a' had the value .5? The results for the circle are presented in Table IV through VI and compared with the second order solution determined by Van Dyke [5]. The grid size in the J and K directions was respectively 64x43 with computational domain:

$$\{(\phi,\psi): \phi \in [-1.5,1.5],\ \psi \in [0,1.5]\}.$$

Equal sized grid spacing was used in the ϕ and ψ directions.

Table IV gives the results obtained by using algorithm A. T was not required to converge after every iteration of U and χ. The angle of inclination of the velocity vector along the profile surface is also given. In this case ρ was taken to be 3.25. The convergence criteria was the maximum norm with tolerance level .0005. The number of iterations required for convergence was 191 and the optimal acceleration parameter was determined to be 1.5384. CPU time was approximately 12 minutes on an IBM 3081 computer. Agreement is best over the main body of the airfoil with accuracy sharply decreasing towards the leading and trailing edge. The accuracy problem at the LE(TE) could be overcome by an extrapolation of the accurate values over the centre of the profile to the stagnation points at the LE(TE). In fact, this was done for a parabolic profile in the irrotational problem using boundary condition (3.3.1) by constructing a least squares polynomial to $\phi(\chi)$ and employing the known values of α in (2.1.22b) such that $\phi'(\chi)$ vanished at the LE(TE). This accounted for the stagnation points. For example $\phi(\chi)$ was taken to be of the form

$$a\chi + b\chi^1 + c\chi^2 + d\chi^3 + e\chi^4 + f\chi^5 + g\chi^6 + h\chi^7$$

for some constants a, b, c, d, e, f, h. The first five constants were determined by the least squares polynomial while g and h were calculated according to

$$\phi'(\chi_{LE}) = \phi'(\chi_{TE}) = 0 \quad .$$

The result was to give more accurate results near the leading and trailing edge while maintaining accuracy over the centre of the profile.

Table V and Figure 8 gives the results obtained by using algorithm B. T was required to converge after every iteration of U and χ. The angle of inclination of the velocity vector along the profile surface is also given. In this case β was taken to be 1.68. The convergence criteria was the maximum norm with tolerance level .0001. The number of iterations required for convergence was 108. CPU time was approximately 16 minutes. The CPU time can be greatly reduced by not requiring T to converge. By discarding terms such as

$$e^{U-T}U_\phi(U_\phi-T_\phi), \; e^{U-T}\chi_\phi(U_\phi-T_\phi)$$

in (4.3.11) the computational time could also be substantially reduced since the effect of these terms is small. However, one of the primary objectives was to retain as many terms as possible so as not to sacrifice accuracy. Agreement is not as good as the previous example over the main body of the airfoil with accuracy much improved towards the leading and trailing edge. The computer used was the IBM-3081. The results by the new approach are found to be roughly $D\phi$ less than the results of Van Dyke. This suggests making the grid finer or packing lines or increasing the grid size could improve the solution in the sense of making it closer to perturbation results, which however are themselves only approximate solutions.

Table VI gives the results obtained by using algorithm C. T was required to converge after every iteration of U and χ. The angle of inclination of the velocity vector along the profile surface is also given. In this case β was taken to be 1.68. The convergence criteria was the maximum norm with tolerance level .0001. Only 95 iterations were required for convergence. CPU time was 10 minutes. This is substantially less than for algorithm B. One of the reasons is that T converged 2 to 3 times faster between each iteration of χ and U. The results follow the same pattern as algorithm A and are roughly $D\phi$ greater than Van Dyke's results, over the top of the airfoil.

In summary, the algorithm of Chapter 3 can be successfully extended to flows with variable vorticity. Two boundary conditions, one exact and the other differenced form of the same equation, can be employed to generate solutions. The validity of the former solution can be questioned near the leading and trailing edge. The latter represents the flow accurately in these regions as well as over the top of the profile. However, the exact form can be used to advantage when singularities arise as in Chapter 3.

CHAPTER 5 ERROR EQUATION FOR ADI METHOD

5.1 STABILITY ANALYSIS

In the numerical solution of partial differential equations various iterative schemes are derived, for example SOR, usually based on the finite difference approximations to these equations. In order to give them a mathematically symbolic form it is necessary to take into account the setting in which they were derived which is generally a rectangular grid. From the Taylor series approximations to the PDE we assume the dependence of the value of the unknown T say, on its neighbours, usually represented as subscripts. Moreover, an iterative process involves an iteration level n which is denoted by a superscript. Simply then, $T^n_{J,K}$ can be expressed as a linear combination of its neighbours all evaluated at some iteration level $n_0 \leq n$. Therefore we write the general form,

$$T^{n+m+i}_{J,K} = \sum_{q,L,M} C^{n+q+i}_{J+L,K+M} \, T^{n+q+i}_{J+L,K+M} + S_{J,K} \qquad (5.1.1)$$

$$\text{for} \quad i = 1,\ldots,\ell$$

where q, L, M are finite integers ranging through both positive and negative values and T^{n+m+i} is the most recent approximate solution. $S_{J,K}$ is a function independent of the iteration level. i denotes the fact that there may be more than one equation for only one unknown. This also indicates one complete iteration is comprised of several steps. For example, in SLOR there are two distinct equations if the iterative process is done in both coordinate directions. These two equations may differ according to the boundary conditions being used. As will be seen, it is essential to look at one single equation comprised of the individual steps in order to get an overall picture of stability. Let

$$n + q_{MAX} = n + m \quad ,$$

$$n + q_{MIN} = n - m \quad . \tag{5.1.2}$$

By (5.1.1) and (5.1.2) we observe (5.1.1) is similar to a polynomial of order 2m, in the sense that the superscripts denote powers. This suggests finding an approximation to the error that utilizes the iteration level n as an index, hence a polynomial can be found whose roots determine limitations on one or more constants in the PDE, such as grid size, or ADI parameter. To analyse stability certain criteria are required, for example boundedness. Let $\epsilon>0$ be arbitrarily chosen and

$$\| T^{n+1} - T^{n} \|_{J,K} \leq \varepsilon_n < \varepsilon \quad , \quad n = 0,1,2,... \tag{5.1.3}$$

where ε_n is a <u>sufficiently small</u> positive number and $\|\cdot\|$ is a suitable norm such as Euclidean norm. Equation (5.1.3) means that the error in the solution between the n^{th} and $n+1^{st}$ iteration level is bounded by a reasonably small number. Theoretically it may approach zero but never in fact become zero, due to rounding and truncation error which are related to the limitations of the machine being used. In addition it is required for stability purposes that (5.1.3) hold for arbitrary grid size, $\Delta\phi$, $\Delta\psi$ where Δ denotes increment in this chapter. We note that (5.1.3) holding for large n and arbitrary ε does not always mean that the solution converges to the true solution of the PDE under consideration unless the total truncation error of the representation (5.1.1) can be made arbitrarily small depending on the initial guess and step size.

By definition of convergence of an iterative process, convergence of (5.1.1) implies (5.1.3) is satisfied [14]. Moreover the limit is unique. One can also note that the space $\mathbb{R}^k$ k=0,1,... is complete and for each J,K varying $T_{J,K}$ is a vector in $\mathbb{R}^k$. Therefore, that $\left\{T^{n+1}\right\}$ represent a Cauchy sequence is necessary; that it is sufficient can be seen in light of (5.1.1). Therefore convergence of (5.1.1) is guaranteed if and only if the left hand side is a Cauchy sequence in the sense that a unique limit exists as n increases. In the notation of [20] let ϵ denote the error, ω the true solution and T the approximate solution to the difference relation (5.1.1)

$$T_{J,K} = \omega_{J,K} + \epsilon_{J,K}$$

By linearity of (5.1.1) the function T is improved as the sum of improved functions ω and ϵ. The improvement $\omega = \omega^n$, i.e., the improvement method does not change the true solution. Thus,

$$T^n_{J,K} = \omega_{J,K} + \epsilon^n_{J,K} \quad . \tag{5.1.4}$$

We have from (5.1.4),

$$\left| T^n_{J,K} - T_{J,K} \right| = \left| \epsilon^n_{J,K} - \epsilon_{J,K} \right| \quad .$$

Since $\left\{ T^n \right\}$ is a Cauchy sequence $\left\{ \epsilon^n \right\}$ is also. Using (5.1.4) in (5.1.1) we get

$$\epsilon^{n+m+i}_{J,K} = \sum_{q,L,M} C^{n+q+i}_{J+L,K+M} \, \epsilon^{n+q+i}_{J+L,K+M} \tag{5.1.5}$$

for $i = 1, \ldots, \ell$. For our purposes it is simpler and more convenient to write (5.1.5) as a pair of linear operators;

$$C_i \, \epsilon^n = \epsilon^{n+1} \quad , \quad i = 1,2 \quad . . \tag{5.1.6}$$

We can think of C_i in (5.1.6) not only as a linear operator summing previous values at lower iteration levels (SOR), or a linear matrix operator (SLOR or ADI), but also as having eigenvalues ϵ with eigenvectors $\lambda^{r,s}_{J,K}$,

$$C_i \, \epsilon^n_{J,K} = \lambda^{r,s}_{J,K} \, \epsilon^n_{J,K} \tag{5.1.7}$$

where J and K generally vary from 0 or 1 to JMAX,KMAX respectively. r,s are integers which vary from 0 or 1 to JMAX-1,KMAX-1 respectively. As n increases a sequence of operations is obtained,

$$[C\ldots C]\ \epsilon^{n}_{J,K} = \left[\lambda^{r,s}_{J,K}\right]^{\ell}\ \epsilon^{n}_{J,K}$$

$$= \epsilon^{n+\ell}_{J,K}\ . \qquad (5.1.8)$$

From (5.1.7) we note it is necessary that each of the two operators C_i have similarities in form in order to have the same eigenvalue structure. Also in order that $C^{\ell} \rightarrow 0$ as ℓ approaches infinity the matrix C must be convergent so that necessarily the eigenvalues of C are less than one in magnitude. Also, since $\left[\epsilon^{n}_{J,K}\right]$ is a Cauchy sequence, the ratio,

$$\frac{\epsilon^{n+\ell}_{J,K}}{\epsilon^{n}_{J,K}}\ .$$

converges, as does,

$$\lim_{\ell\to\infty}\left(\lambda^{\ell}+n\ \ldots\ \lambda^{n}\right)\ . \qquad (5.1.9)$$

For our purposes there are two linear operators. The question is whether to examine them separately or combine them into one operator. By combining them together, a quadratic in n is obtained, since there are three iteration levels in total. By examining each separately as in Peaceman and Rachford's analysis [21] of a linear ADI iterative sequence, two symmetrically opposite expressions were obtained for the error, namely,

$$\frac{A_{r,s,2n+1}}{A_{r,s,2n}} = \frac{\rho - 4\sin^2(\beta_s\,\Delta y/2)}{\rho + 4\sin^2(\beta_r\,\Delta x/2)} ,$$

$$\frac{A_{r,s,2n+2}}{A_{r,s,2n+1}} = \frac{\rho - 4\sin^2(\beta_r\,\Delta x/2)}{\rho + 4\sin^2(\beta_s\Delta y/2)} \qquad (5.1.10)$$

where ρ is the ADI parameter

$\beta_r = (2r + 1)\pi/2$, and $\beta_s = (2s + 1)\pi/2$,

$r = 0, \ldots,$ JMAX-1; $q = 0, \ldots,$ KMAX-1.

Individually each ratio of (5.1.10) may exceed unity depending on the choice of r,s or ρ but their product is always less than one in magnitude. Nevertheless it does not mean convergence will occur for arbitrary choice of ρ since for some choice of the ADI parameter, based on experience, divergence will occur. This suggests that looking at each equation separately is not sufficient or thorough enough, rather it is necessary to examine them as a quadratic in the iteration parameter n. As a simple example consider the two equations,

$$E_1^n(\lambda) = \lambda^{n+1} + \lambda^n , \qquad (5.1.11a)$$

$$E_2^n(\lambda) = \lambda^n - \lambda^{n-1} . \qquad (5.1.11b)$$

From the first equation we have

$$\lim_{n\to\infty}\left|E_1^n(\lambda)\right| = \lim_{n\to\infty}\left|\lambda^n(\lambda+1)\right| \to 0 \Leftrightarrow -1 \le \lambda < 1 \qquad (5.1.12a)$$

$$\lim_{n\to\infty}\left|E_2^n(\lambda)\right| = \lim_{n\to\infty}\left|\lambda^{n-1}(\lambda-1)\right| \to 0 \Leftrightarrow -1 < \lambda \le 1. \qquad (5.1.12b)$$

But,

$$\lim_{n\to\infty} \left| \left[E_1 + \widehat{E_2} \right](\lambda) \right| \doteq \lim_{n\to\infty} \left| \lambda^{n-1}\left(\lambda^2+2\lambda-1\right) \right| \to 0 \ ,$$

only if λ is strictly less than one in magnitude or $\lambda=-(1+2^{\frac{1}{2}})$. Thus the sum of the two operators is more restrictive than the product in certain intervals, i.e., [-1,1]. However, because λ can take on negative values the sum could have values outside of the union of the two intervals in (5.1.12). The two operators will therefore be analysed from the viewpoint of a sum to give slightly more accurate information about the stability. To only view them as a product may be inaccurate and misleading.

Next we consider various forms for the eigenfunctions ϵ for the linear operator C_i which as yet is undefined. The type of linear operator we consider is related to a tridiagonal matrix in the case of SLOR or ADI and denoted by TRID[a,b,c] where a,b,c are real numbers. There are several forms for the error ϵ as presented in the literature. One such form is given in [23],

$$\epsilon^n_{J,K} = \epsilon^n \, A_{r,s} \, e^{i(r\Delta\phi\Pi J+s\Delta\psi\Pi K)} \quad . \qquad (5.1.13)$$

r and s are integers ranging over the number of interior points in the ϕ and ψ directions respectively and $i=(-1)^{\frac{1}{2}}$. An alternate form is given in [22],

$$\epsilon_{J,K} = A^J B^K \left[\sin\frac{\Pi rJ}{P} \sin\frac{\Pi sK}{Q} \right], \ P = \text{JMAX}-1, \ Q = \text{KMAX}-1 \qquad (5.1.14)$$

where

$$r = 0,\ldots,P \;;\quad s = 0,\ldots,Q \;;\quad JMIN = 0 \;;\; KMIN = 0 \quad .$$

As an example consider Richardson's method [22], similar to point-Jacobi,

$$\epsilon^{n+1}_{J,K} = \epsilon^{n}_{J,K} + \lambda L \epsilon^{n}_{J,K}$$

where L is an operator corresponding to the central difference approximation to Laplace's equation with equal grid spacing. For Richardson's method, the eigenvalues derived from (5.1.14), with the term $A^J B^K$ equal to one, are given by (cf. [22]),

$$\lambda^{r,s} = -4 \left[\sin^2\left(\frac{\pi r}{2P}\right)+\sin^2\left(\frac{\pi s}{2Q}\right)\right].$$

Another variation is given in [21],

$$\epsilon_{J,K,n} = \sum_{r,s=0}^{N-1} A_{r,s,n} \cos(\beta_r x) \cos(\beta_s y) \qquad (5.1.15)$$

where r and s are as defined in (5.1.10). Inspection of (5.1.14)-(5.1.15) shows that the two forms are similar in that one can be transformed into the other by trigonometric identities. The coefficient functions have different significance depending on the problem being solved.

The eigenvalues of TRID(a,b,c) are given by

$$b + 2(ac)^{1/2}\cos\left[\frac{\pi J}{P}\right] \qquad J = 1,\ldots,P \;. \qquad (5.1.16)$$

The determinant of (TRID(a,b,c) − λI) is given by a recursive relation,

$$\begin{vmatrix} b-\lambda & c & \cdots \\ a & b-\lambda & \cdots \\ \cdot & . & \end{vmatrix}_p = (b-\lambda)C(\lambda)_{p-1} - acC(\lambda)_{p-2} \qquad (5.1.17)$$

where p is the order of the matrix and

$$C_0(\lambda) = 1 \quad , \quad C_1(\lambda) = b - \lambda \quad ,$$

$$C_2(\lambda) = (b-\lambda)^2 - ac \quad .$$

The result (5.1.17) is easily verified by induction. By a change of variable $t=b-\lambda$ it can be seen that $C_{2p+1}(t)$ is odd and C_{2p} is even, $p=0,1,\ldots,$ with respect to $\lambda=b$ since $t>0$ for $b>\lambda$ and $t\leq 0$ for $b<\lambda$. Thus the roots come in pairs. In the case of ADI C is simply

$$\text{TRID}(-1,2+\rho,-1)^{-1}$$

for Laplace's equation. From (5.1.16) we see the eigenvalues λ are in the interval (0,4). The addition of a positive constant $\rho>1$ is observed to ensures that the eigenvalues of C, λ^{-1}, are less than one in magnitude. Trigonometric approximations to the eigenvalue behaviour of C are essential because of the oddness and evenness of the characteristic polynomials (5.1.17). This then is a necessary condition.

Consider the pair of iterative equations [23],

$$T^{n+\frac{1}{2}}_{J,K} = T^{n}_{J,K} - \frac{1}{\rho_A E_{J,K}}$$

$$\times \left[\Delta^2_x \, T^{n+\frac{1}{2}}_{J,K} + \Delta^2_y \, T^{n}_{J,K} - q_{J,K}\right] \qquad (5.1.18a)$$

$$T^{n+1}_{J,K} = T^{n+\frac{1}{2}}_{J,K} - \frac{1}{\rho_A E_{J,K}}$$

$$\times \left[\Delta^2_x T^{n+\frac{1}{2}}_{J,K} + \Delta^2_y T^{n+1}_{J,K} - q_{J,K}\right] \qquad (5.1.18b)$$

which represents the ADI method and where ρ_A is a positive constant taken in the interval (0,1),

$$\Delta^2_x T_{J,K} = F_{J,K} T_{J+1,K} + EX_{J,K} T_{J,K} + D_{J,K} T_{J-1,K}$$

such that $EX_{J,K}$ is a portion of $E_{J,K}$ corresponding to the x-direction. Similarily for the y direction,

$$\Delta^2_y T_{J,K} = B_{J,K} T_{J,K-1} + EY_{J,K} T_{J,K} + H_{J,K} T_{J,K+1} \cdot$$

Peaceman and Rachford [21] have examined the stability of a pair of similar equations,

$$\frac{T^{2n+1}_{J,K} - T^{2n}_{J,K}}{\Delta t} = \frac{T^{2n+1}_{J-1,K} - 2T^{2n+1}_{J,K} + T^{2n+1}_{J+1,K}}{(\Delta x)^2}$$

$$+ \frac{T^{2n}_{J,K-1} - 2T^{2n}_{J,K} + T^{2n}_{J,K+1}}{(\Delta y)^2}$$

$$\frac{T^{2n+2}_{J,K} - T^{2n+1}_{J,K}}{\Delta t} = \frac{T^{2n+1}_{J-1,K} - 2T^{2n+1}_{J,K} + T^{2n+1}_{J+1,K}}{(\Delta x)^2}$$

$$+ \frac{T^{2n+2}_{J,K-1} - 2T^{2n+2}_{J,K} + T^{2n+2}_{J,K+1}}{(\Delta y)^2} \qquad (5.1.19)$$

and shown that the ratios (5.1.10), for an arbitrary choice of parameters ρ do not exceed unity when considering their product where

$$\frac{1}{\rho} = \frac{\Delta t}{\Delta x^2} = \frac{\Delta t}{\Delta y^2} \;.$$

Equation (5.1.18) is in a more general form than that of (5.1.19). Comparing the respective notations of [21], [23] we find

$$\rho = \rho_A \, E_{J,K} \qquad (5.1.20)$$

Let x, y correspond to ϕ and ψ respectively. Consider a second order PDE of the form (this equation arises in the next chapter),

$$K^{-1}T_{\psi\psi} + KT_{\phi\phi} - 2\cos\theta \; T_{\psi\phi} = 0 \qquad (5.1.21a)$$

where θ is a measure of the angle of intersection of the coordinate curves ψ=constant and ϕ=constant taken as functions of x and y and where K is independent of ψ and ϕ but varies continuously with θ. Assume that K=1 corresponds to θ=90 degrees and that as θ decreases to 0 degree K increases. In the subsequent stability analysis of (5.1.18), the continuous dependence of convergence rate on θ is shown. A critical angle θ^c exists, lying in the degree interval (0,90) for which the convergence rate is maximized where

$$R = - \log \, [\rho(C)]$$

is the rate of convergence and $\rho(C)$ is the spectral radius. It is also interesting to observe that as θ was decreased only slightly below θ^c, divergence occurred. According to our previous reasoning it is necessary to substitute (5.1.18a) into (5.1.18b) to yield, with $x \to \phi$ and $y \to \psi$,

$$T^{n+1}_{J,K} = T^{n}_{J,K} - \frac{2}{\rho_A E_{J,K}} \tag{5.1.21b}$$

$$\times \left(\Delta^2_\phi T^{n+½}_{J,K} + \frac{\Delta^2_\psi}{2} T^{n+1}_{J,K} + \frac{\Delta^2_\psi}{2} T^{n}_{J,K} - q_{J,K}\right).$$

Using the relation (5.1.4) in (5.1.21b) yields

$$\epsilon^{n+1}_{J,K} = \epsilon^{n}_{J,K} - \frac{2}{\rho_A E_{J,K}} \tag{5.1.21c}$$

$$\times \left(\Delta^2_\phi \epsilon^{n+½}_{J,K} + \frac{\Delta^2_\psi}{2} \epsilon^{n+1}_{J,K} + \frac{\Delta^2_\psi}{2} \epsilon^{n}_{J,K}\right).$$

By inspection, (5.1.21c) has three iteration levels and will involve a quadratic in ϵ. Applying central difference approximations to (5.1.21a) yields

$$\kappa^{-1} \left(\epsilon_{J,K-1} -2\, \epsilon_{J,K} + \epsilon_{J,K+1}\right) + \kappa \left(\epsilon_{J-1,K} -2\, \epsilon_{J,K} + \epsilon_{J+1,K}\right)$$

$$- .5 \cos\theta \left(\epsilon_{J+1,K+1} - \epsilon_{J+1,K-1} - \epsilon_{J+1,K-1} + \epsilon_{J-1,K-1}\right)$$

$$= 0 \tag{5.1.21d}$$

where $D\psi = D\phi$. In the notation of (5.1.18), comparing with the coefficients of (5.1.21d),

$$E_{J,K} = E = 2\left[\kappa^2+1\right] \qquad B = B_{J,K} = -1 \quad ,$$

$$F_{J,K} = F = -\kappa^2 \qquad H = H_{J,K} = -1 \quad ,$$

$$D_{J,K} = D = \kappa^2 \quad . \tag{5.1.22}$$

The cross derivative term has been taken to the RHS in (5.1.21d) since the terms correspond to elements far from the main diagonal of the tridiagonal, and is not included in the stability analysis. Using (5.1.22), (5.1.13) in (5.1.21c), yields

$$\epsilon^{n+1}A_{r,s}e^{i(r\Delta\phi\Pi J+s\Delta\psi\Pi K)} = \tag{5.1.23a}$$

$$\epsilon^{n}A_{r,s}e^{i(r\Delta\phi\Pi J+s\Delta\psi\Pi K)} - \frac{2}{\rho_A E_{J,K}}\Bigl\{A_{r,s}K^2\epsilon^{n+½} \times$$

$$\Bigl(-e^{i(r\Delta\phi\Pi(J-1)+s\Delta\psi\Pi K)}+2e^{i(r\Delta\phi\Pi J+s\Delta\psi\Pi K)}-e^{i(r\Delta\phi\Pi(J+1)+s\Delta\psi\Pi K)}\Bigr)$$

$$+\frac{1}{2}\Bigl(\epsilon^n+\epsilon^{n+1}\Bigr)A_{r,s}\times$$

$$\Bigl(-e^{i(r\Delta\phi\Pi J+Q\Delta\psi\Pi(K-1))}+2e^{i(r\Delta\phi\Pi J+s\Delta\psi\Pi K)}-e^{i(r\Delta\phi\Pi J+s\Delta\psi\Pi(K+1))}\Bigr)\Bigr\}.$$

Multiplying (5.1.23a) by

$$\epsilon^{-n}A^{-1}_{r,s}e^{-i(r\Delta\phi\Pi J+s\Delta\psi\Pi K)}$$

yields

$$\epsilon - 1 = -\Bigl[\rho_A(K^2+1)\Bigr]^{-1}x \tag{5.1.23b}$$

$$\Bigl(K^2\epsilon^{½}(-e^{-ir\Delta\phi\Pi}+2-e^{ir\Delta\phi\Pi}+\frac{1}{2}(\epsilon+1)(-e^{-is\Delta\psi\Pi}+2-e^{is\Delta\psi\Pi})\Bigr)$$

$$= -\frac{1}{\rho_A(K^2+1)}\Bigl[K^2\epsilon^{½}(2-2\cos r\Delta\phi\Pi)+\frac{1}{2}(\epsilon+1)(2-2\cos s\Delta\psi\Pi)\Bigr].$$

Replacing

$\frac{1}{\rho_A(K^2+1)}$ by τ , $r\Delta\phi\Pi$ by θ_ϕ , $s\Delta\psi\Pi$ by θ_ψ ,

$2 - 2\cos(s\Delta\psi\Pi)$ by $4\sin^2\frac{\theta_\psi}{2}$

and

$2- 2\cos(r\Delta\phi\Pi)$ by $4\sin^2\frac{\theta_\phi}{2}$

in (5.1.23b) yields

$$\epsilon - 1 = -4\tau\left[K^2\ \epsilon^{1/2}\ \sin^2\frac{\theta_\phi}{2} + \frac{1}{2}(\epsilon+1)\ \sin^2\frac{\theta_\psi}{2}\right] .$$

Replacing ϵ by ϵ^2 gives

$$\epsilon^2\left[1+2\tau\sin^2\frac{\theta_\psi}{2}\right]+4K^2\tau\epsilon\sin^2\frac{\theta_\phi}{2}-(1-2\tau\sin^2\frac{\theta_\psi}{2})=0. \qquad (5.1.24)$$

Solving the quadratic (5.1.24) for ϵ yields

$$\epsilon = \frac{-2\tau K^2\sin^2\eta\pm\left\{4\tau^2K^4\sin^4\eta+1-4\tau^2\sin^4\varepsilon\right\}^{1/2}}{1+2\tau\sin^2\varepsilon}$$

$$\text{where } \varepsilon = \frac{\theta_\psi}{2} \quad , \quad \eta = \frac{\theta_\phi}{2} \quad . \qquad (5.1.25a)$$

We consider only the case ϵ is real. For stability purposes we are interested when $|\epsilon| \leq 1$ in (5.1.25a). Taking the positive sign and squaring yields

$$4\tau^2K^4\sin^4\frac{\theta_\phi}{2} + 1 - 4\tau^2\sin^4\frac{\theta_\psi}{2} \leq$$

$$\left[1+2\tau\sin^2\frac{\theta_\psi}{2}+2\tau K^2\sin^2\frac{\theta_\phi}{2}\right]^2 .$$

Simplifying further gives

$$(1+2\tau\sin^2\frac{\theta_\phi}{2})(1-2\tau\sin^2\frac{\theta_\psi}{2}) \leq$$

$$(1+2\tau\sin^2\frac{\theta_\psi}{2})^2 + 4\tau K^2\sin^2\frac{\theta_\phi}{2}(1+2\tau\sin^2\frac{\theta_\psi}{2}) .$$

Dividing through by $1 + 2\tau\sin^2\frac{\theta_\psi}{2}$ (> 0) gives

$$1-2\tau\sin^2\frac{\theta_\psi}{2} \leq 1 + 2\tau\sin^2\frac{\theta_\psi}{2} + 4\tau K^2\sin^2\frac{\theta_\phi}{2}$$

which holds necessarily unless θ_ϕ, θ_ψ vanish in which case equality holds. On the other side, $-1 < \epsilon$,

$$\left(-(1+2\tau\sin^2\frac{\theta_\psi}{2}) + 2\tau K^2\sin^2\frac{\theta_\phi}{2}\right)^2 \leq$$

$$4\tau^2K^4\sin^4\frac{\theta_\phi}{2} + 1 - 4\tau^2\sin^4\frac{\theta_\psi}{2} . \qquad (5.1.25b)$$

Simplifying brings

$$-4\tau K^2\sin^2\frac{\theta_\phi}{2}\left(1+2\tau\sin^2\frac{\theta_\psi}{2}\right) + \left(1+2\tau\sin^2\frac{\theta_\psi}{2}\right)^2 \leq$$

$$\left(1+2\tau\sin^2\frac{\theta_\psi}{2}\right)\left(1-2\tau\sin^2\frac{\theta_\psi}{2}\right) .$$

That is

$$-4\tau K^2\sin^2\frac{\theta_\phi}{2} + 1 + 2\tau\sin^2\frac{\theta_\psi}{2} - 1 + 2\tau\sin^2\frac{\theta_\psi}{2} \leq 0$$

or

$$4\tau(K^2\sin^2\frac{\theta_\phi}{2} - \sin^2\frac{\theta_\psi}{2}) \geq 0 .$$

Thus,

$$K \geq \frac{\left|\sin\frac{\theta_\psi}{2}\right|}{\left|\sin\frac{\theta_\phi}{2}\right|} . \qquad (5.1.26a)$$

On the other hand, if the negative sign is taken in (5.1.25a), then by a similar analysis,

$$K < \frac{\left|\sin\frac{\theta_\psi}{2}\right|}{\left|\sin\frac{\theta_\phi}{2}\right|} . \qquad (5.1.26b)$$

Equation (5.1.26b) also holds if the quantity on the LHS of inequality (5.1.25) is so large in magnitude as to exceed the RHS. An interval for κ is sought which accelerates convergence of the iterative solution of (5.1.21d) by the ADI method. Thus, specific bounds are determined for $\sin\frac{\theta_\phi}{2}$, $\sin\frac{\theta_\psi}{2}$. We next determine several cases under which the discriminant of (5.1.25a) is positive, that is,

$$4\tau^2\left[\kappa^4\sin^4\frac{\theta_\phi}{2} - \sin^4\frac{\theta_\psi}{2}\right]+1 \geq 0 \quad .$$

If (5.1.26a) holds then necessarily,

$$\kappa^4\sin^4\frac{\theta_\phi}{2} - \sin^4\frac{\theta_\psi}{2} + \frac{\rho_A^2}{4}(\kappa^2+1)^2 \geq 0 \quad . \tag{5.1.27}$$

Equation (5.1.27) also holds, independently of κ if,

$$\rho_A \geq 2\sin^2\frac{\theta_\psi}{2} \quad .$$

Let us suppose

$$\rho_A \leq 2\sin^2\left(\frac{\theta_\psi}{2}\right) \quad . \tag{5.1.28}$$

Equation (5.1.27) can also be written as

$$\kappa^4\left[4\sin^4\frac{\theta_\phi}{2}+\rho_A^{\ 2}\right]+2\kappa^2\rho_A^{\ 2}+\rho_A^{\ 2}-4\sin^4\frac{\theta_\psi}{2} \geq 0$$

the roots being

$$\kappa^2 = \frac{-\rho_A^{\ 2}\pm(4\rho_A^{\ 2}(\sin^4\epsilon-\sin^4\eta)+16\sin^4\epsilon\sin^4\eta)^{1/2}}{\rho_A^{\ 2}+4\sin^4\eta}$$

$$= \frac{-1\pm 2(\sin^4\epsilon-\sin^4\eta+4({\rho_A}^2)^{-1}\sin^4\epsilon\sin^4\eta)^{1/2}}{1+({\rho_A}^2)^{-1}\sin^4\eta} \quad . \tag{5.1.29a}$$

where $\epsilon = \frac{\theta_\psi}{2}$, $\eta = \frac{\theta_\phi}{2}$.

By inspection (5.1.27) is of the form,

$$aK^4 + bK^2 + c \geq 0 \qquad a,b > 0$$

and only has one critical point at $K=0$ which is easily verified. By taking $\rho_A \leq 2\sin^2\frac{\theta_\psi}{2}$ we have added a restriction to ρ_A. In the same manner a restriction can be applied to ρ_A in (5.1.29a) to yield

$$4\sin^4\frac{\theta_\psi}{2} \geq {\rho_A}^2 \geq \frac{4\sin^4\frac{\theta_\psi}{2}\sin^4\frac{\theta_\phi}{2}}{\sin^4\frac{\theta_\psi}{2} - \sin^4\frac{\theta_\phi}{2}} \quad . \tag{5.1.29b}$$

The discriminant of (5.1.29a) is positive if

$$\sin^4\frac{\theta_\psi}{2} - \sin^4\frac{\theta_\phi}{2} > 0 \quad .$$

Assuming the above relation holds implies $K \geq 1$. Also solving the discriminant in (5.1.29a) for ρ_A yields an expression which is trivially satisfied with respect to (5.1.28). Taking the modulus yields one of two possibilities for an expression of the form

$$\rho_A^2(a + b)(a - b) \leq (\geq) \; 4a^2b^2 \quad , \quad a,b > 0 \quad .$$

Equation (5.1.28) suggests the above inequalities (5.1.29b). In [23] ρ is taken to lie in [0,1]. However this same restriction is not applied here. Simplifying (5.1.29b) yields

$$\sin^4\frac{\theta_\psi}{2} - \sin^4\frac{\theta_\phi}{2} \geq \sin^4\frac{\theta_\phi}{2} \quad .$$

Thus, dividing by $\sin^4\frac{\theta_\phi}{2}$ and taking the fourth root yields

$$\frac{\left|\sin\frac{\theta_\psi}{2}\right|}{\left|\sin^4\frac{\theta_\phi}{2}\right|} \geq 2^{\frac{1}{4}} \quad . \tag{5.1.30}$$

The above result is true if (5.1.29b) holds,

$$\rho_A \leq 2\sin^2\frac{\theta_\psi}{2} \leq 2 \tag{5.1.31}$$

and (5.1.25a) has real roots. Thus if equality holds in (5.1.26a), (5.1.30) gives a lower bound for an <u>optimal</u> κ. In the case of (5.1.25a) we find

$$\left|\epsilon\left(\frac{\sin\xi}{\sin\eta}\right)\right| = \left|\frac{1-2\tau\sin^2\xi}{1+2\tau\sin^2\xi}\right| \tag{5.1.32}$$

$$= \left|\left\{1-\left[2\sin^2\xi\left(\rho_A\left(\frac{\sin^2\xi}{\sin^2\eta} + 1\right)\right)^{-1}\right]\right\}\right.$$

$$\left.\times\left\{1+\left[2\sin^2\xi\left(\rho_A\left(\frac{\sin^2\xi}{\sin^2\eta}+1\right)\right)^{-1}\right]\right\}^{-1}\right|$$

where $\xi = \frac{\theta_\psi}{2}$, $\eta = \frac{\theta_\phi}{2}$.

Equation (5.1.32) is less than one in magnitude for every choice of non-zero θ_ψ, θ_ϕ. We can check for critical points.

Differentiating with respect to $\sin\frac{\theta_\phi}{2}$ we get

$$\frac{\partial \epsilon}{\partial \sin\frac{\theta_\phi}{2}} =$$

$$\left[\frac{-4\sin^2\epsilon\sin\eta}{\rho_A(\sin^2\eta+\sin^2\epsilon)}+\frac{4\sin^2\epsilon\sin^3\eta}{\rho_A(\sin^2\eta+\sin^2\epsilon)^2}\right]$$

$$\times\left\{\left[1+\frac{2\sin^2\epsilon\sin^2\eta}{\rho_A(\sin^2\eta+\sin^2\epsilon)}\right]^{-1}\right.$$

$$+\left[1+\frac{2\sin^2\epsilon\sin^2\eta}{\rho_A(\sin^2\eta+\sin^2\epsilon)}\right]^{-2}$$

$$\left.\times\left[1-\frac{2\sin^2\epsilon\sin^2\eta}{\rho_A(\sin^2\eta+\sin^2\epsilon)}\right]\right\} .$$

where $\epsilon = \frac{\theta_\psi}{2}$, $\eta = \frac{\theta_\phi}{2}$. The latter terms in the bracket are non-zero. The former terms become after cross-multiplying by

$$\left[\sin^2 \frac{\theta_\phi}{2} + \sin^2 \frac{\theta_\phi}{2}\right]^2$$

and equating the numerator to zero,

$$\left\{\left(\sin^2\frac{\theta_\phi}{2}+\sin^2\frac{\theta_\psi}{2}\right)-\sin^2\frac{\theta_\phi}{2}\right\}\sin\frac{\theta_\phi}{2}\sin^2\frac{\theta_\psi}{2} = 0$$

$$\Leftrightarrow \sin\frac{\theta_\psi}{2} = 0 \Rightarrow \frac{\theta_\psi}{2} = 0 , \Pi .$$

Similarly for $\frac{\theta_\phi}{2}$. By symmetry the same result holds when differentiating with respect to $\sin\frac{\theta_\phi}{2}$ and we see (0,0) corresponds to a maximum. Subsequently in the next section it is checked if $\epsilon(K)$ has any critical points. The function

$$\epsilon\left[\left(\sin\frac{\theta_\psi}{2}\right)\times\left(\sin\frac{\theta_\phi}{2}\right)^{-1}\right]$$

behaves therefore as an decreasing function of $\sin^2\frac{\theta_\psi}{2}$ and $\sin^2\frac{\theta_\phi}{2}$. We also observe from (5.1.32), $\epsilon \to 0$ as

$$\rho_A^{-1} \to \frac{\sin^2\frac{\theta_\psi}{2} + \sin^2\frac{\theta_\phi}{2}}{2\sin^2\frac{\theta_\psi}{2}\sin^2\frac{\theta_\phi}{2}} = \rho_B. \qquad (5.1.33)$$

It also follows that $\rho_B^{-1} < \rho_B$ since for any two positive numbers a,b in the interval (0,1),

$$2ab < a + b .$$

It is thus accurate to consider ρ_A bounded below by ρ_B^{-1} and above by (5.1.31). ρ_B is taken as an intermediate value. Assuming (5.1.31), (5.1.33) hold we have

$$\sin^2\frac{\theta_\psi}{2} + \sin^2\frac{\theta_\phi}{2} \le 4\sin^4\frac{\theta_\psi}{2}\sin^2\frac{\theta_\phi}{2} .$$

or

$$0 \le 4\sin^4\theta_\psi\sin^2\frac{\theta_\phi}{2} - \sin^2\frac{\theta_\psi}{2} - \sin^2\frac{\theta_\phi}{2} . \qquad (5.1.34)$$

The roots of (5.1.34) in terms of θ_ϕ are

$$\sin^2\frac{\theta_\psi}{2} = \frac{1 \pm \left(1+16\sin^4\frac{\theta_\phi}{2}\right)^{1/2}}{8\sin^2\frac{\theta_\phi}{2}} \; . \qquad (5.1.35)$$

From (5.1.35) we see $\sin\frac{\theta_\psi}{2} \leq 1$ requires that

$$\left[\left(1+16\sin^4\frac{\theta_\phi}{2}\right)^{1/2}\right]^2 \leq \left[8\sin^2\frac{\theta_\phi}{2}-1\right]^2$$

so that

$$0 \leq -\frac{1}{3}\sin^2\frac{\theta_\phi}{2} + \sin^4\frac{\theta_\phi}{2}$$

and

$$\frac{1}{3} \leq \sin^2\theta_\phi \leq 1 \; . \qquad (5.1.36a)$$

Hence,

$$\sin^2\frac{\theta_\psi}{2} \in \left[\frac{1+17^{1/2}}{8} , 1\right] \qquad (5.1.36b)$$

by (5.1.35) and (5.1.36a). Now assuming

$$\kappa^2 = \frac{\left|\sin\frac{\theta_\psi}{2}\right|^2}{\left|\sin\frac{\theta_\phi}{2}\right|^2}$$

by (5.1.35) and (5.1.36),

$$.64 \leq \kappa^2 \leq 3$$

but by (5.1.30),

$$2^{1/2} \leq \kappa^2 \leq 3 \; . \qquad (5.1.37)$$

Since $\rho = 2\rho_A(\kappa^2+1)$ we have

$$2(2^{1/2}+1) \le \frac{\rho}{\rho_A} \le 8 \quad . \tag{5.1.38}$$

From (5.1.33) it can be shown, using (5.1.36), that

$$.5 \le \rho_A \le 2 \quad . \tag{5.1.39}$$

By taking ρ_A=.5 we have

$$(2^{1/2}+1) \le \rho \le 4 \quad . \tag{5.1.40}$$

Thus we conclude that <u>good</u> values of ρ,κ can be found by (5.1.37) and (5.1.40). Table VII gives the results of solving (5.1.21d). For ψ=0 the boundary condition was (3.2.4a) with α calculated by determining $f'(\chi)$ for the circle of radius $\frac{1}{2}$. χ was found from differencing χ_ϕ along ψ=0 as described in the algorithm of Chapter 2. The computational domain was as previously described in section 3.3. ρ was taken to be 3.25, well within the interval given in (5.1.40). The results show agreement with (5.1.37) and convergence is fastest near $\kappa=\sqrt{3}$. When equality held in (5.1.26a) we saw the monotonic variation of ϵ. This suggests finding $\epsilon'(\kappa)$ which is done in the next section.

5.2 CRITICAL POINTS OF ERROR FUNCTION

In this section we check the derivative $\epsilon'(\kappa)$ for critical points. To simplify notation we set

$$\alpha = \sin\frac{\theta_\psi}{2} \quad , \quad \beta = \sin\frac{\theta_\phi}{2} \quad , \quad \varepsilon = \kappa^2\tau$$

where

$$\frac{1}{\rho_A(K^2+1)} = \tau \quad .$$

Thus we have

$$\varepsilon'(K) = \frac{d}{dK}\left[\frac{-2\varepsilon\beta^2+(4\varepsilon^2\beta^4+1-4\tau^2\alpha^4)^{1/2}}{1+2\tau\alpha^2}\right]$$

$$= \left[-2\varepsilon'\beta^2+\frac{4\varepsilon\varepsilon'\beta^4-4\tau\tau'\alpha^4}{(4\varepsilon^2\beta^4+1-4\tau^2\alpha^4)^{1/2}}\right]\left(\frac{1}{1+2\tau\alpha^2}\right)$$

$$- \left[\frac{-2\varepsilon\beta^2+(4\varepsilon^2\beta^4+1-4\tau^2\alpha^4)^{1/2}}{(1+2\tau\alpha^2)^2}\right](2\tau'\alpha^2) \quad .$$

Multiplying the above expression by

$$(1+2\tau\alpha^2)^2\,(4\varepsilon^2\beta^4+1-4\alpha^4\tau^2)^{1/2}$$

and equating to 0 gives

$$\left(1 + 2\tau\alpha^2\right)\ \left(4\varepsilon^2\beta^4 + 1 - 4\alpha^4\tau^2\right)^{1/2}$$

$$\left[-2\varepsilon'\beta^2+\frac{4\varepsilon\varepsilon'\beta^4-4\tau\tau'\alpha^4}{(4\varepsilon^2\beta^4+1-4\tau^2\alpha^4)^{1/2}}\right]$$

$$-\ (2\tau'\alpha^2)\ (4\varepsilon^2\beta^4+1-4\alpha^4\tau^2)^{1/2}\left[-2\varepsilon\beta^2+(4\varepsilon^2\beta^4+1-4\alpha^4\tau^2)^{1/2}\right]$$

$$= (4\varepsilon^2\beta^4+1-4\alpha^4\tau^2)^{1/2}\left[-2\varepsilon'\beta^2(1+2\tau\alpha^2)+4\tau'\varepsilon\alpha^2\beta^2\right]$$

$$+\ (1+2\tau\alpha^2)(4\varepsilon\varepsilon'\beta^4-4\tau\tau'\alpha^4)-2\tau'\alpha^2(4\varepsilon^2\beta^4-4\tau^2\alpha^4+1)$$

$$= 0 \quad .$$

Now we note

$$\varepsilon' = \left(\frac{K^2}{\rho_A(K^2+1)}\right)' = \frac{2K}{\rho_A(K^2+1)} - \frac{2K^3}{\rho_A(K^2+1)^2} \quad .$$

$$= \frac{2\kappa}{\rho_A(\kappa^2+1)^2}$$

$$\tau' = \frac{-2\kappa}{\rho_A(\kappa^2+1)^2} .$$

Thus the above expression can be written upon squaring both sides

$$\left(\frac{4\kappa^4\beta^4}{\rho_A^{\,2}(\kappa^2+1)^2}+1-\frac{4\alpha^4}{\rho_A^{\,2}(\kappa^2+1)^2}\right) \times$$

$$\left[\frac{-8\kappa\alpha^2\beta^2(\kappa^2+1)}{\rho_A^{\,2}(\kappa^2+1)^3}-\frac{4\kappa\beta^2}{\rho_A(\kappa^2+1)^2}\right]^2 =$$

$$\left(\left(1+\frac{2\alpha^2}{\rho_A(\kappa^2+1)}\right)\left(\frac{8\kappa^3\beta^4}{\rho_A^{\,2}(\kappa^2+1)^3}+\frac{8\kappa\alpha^4}{\rho_A^{\,2}(\kappa^2+1)^3}\right)\right.$$

$$\left.+\ \frac{4\kappa\alpha^2}{\rho_A(\kappa^2+1)^2}\left(\frac{4\kappa^4\beta^4}{\rho_A^{\,2}(\kappa^2+1)^2}+1-\frac{4\alpha^4}{\rho_A^{\,2}(\kappa^2+1)^2}\right)\right)^2 .$$

That is

$$\left(\frac{4}{\rho_A^{\,2}(\kappa^2+1)^2}(\beta^4\kappa^4-\alpha^4)+1\right) \times$$

$$\left[\frac{-8\kappa\alpha^2\beta^2(\kappa^2+1)}{\rho_A^{\,2}(\kappa^2+1)^3}-\frac{4\kappa\beta^2}{\rho_A(\kappa^2+1)^2}\right]^2 .$$

$$= \left\{\left(1+\frac{2\alpha^2}{\rho_A(\kappa^2+1)}\right)\left(\frac{8\kappa(\kappa^2\beta^4+\alpha^4)}{\rho_A^{\,2}(\kappa^2+1)^3}\right)\right.$$

$$+\ \frac{4K\alpha^2}{\rho_A(K^2+1)^2}\left(\frac{4}{\rho_A^2(K^2+1)^2}(\beta^4K^4-\alpha^4)+1\right)\Bigg\}^2$$

or multiplying by $(K^2+1)^4(16K^2)^{-1}$ yields

$$\left(\frac{4}{\rho_A^2(K^2+1)^2}(\beta^4K^4-\alpha^4)+1\right)\left[\frac{2\alpha^2}{\rho_A^2}+\frac{1}{\rho_A}\right]^2\beta^4$$

$$=\left\{\left(1+\frac{2\alpha^2}{\rho_A(K^2+1)}\right)\frac{2(\beta^4K^2+\alpha^4)}{\rho_A^2(K^2+1)}\right.$$

$$\left.+\ \frac{\alpha^2}{\rho_A}\left(\frac{4}{\rho_A^2(K^2+1)^2}(\beta^4K^4-\alpha^4)+1\right)\right\}^2$$

$$=\left[\frac{4K^2\alpha^2\beta^4(K^2+1)}{\rho_A^3(K^2+1)^2}\right.$$

$$\left.+\ \frac{2(K^2\beta^4+\alpha^4)}{\rho_A^2(K^2+1)}+\frac{\alpha^2}{\rho_A}\right]^2$$

or multiplying by $\left[\dfrac{1}{\rho_A^2(K^2+1)^2}\right]^{-1}$ gives

$$\left[4(\beta^4K^4-\alpha^4)+\rho_A^2(K^2+1)^2\right]\left(1+\frac{2\alpha^2}{\rho_A}\right)^2\frac{\beta^4}{\rho_A^2}$$

$$=\left[\frac{4\alpha^2\beta^4K^2}{\rho_A^2}+\frac{2}{\rho_A}(K^2\beta^4+\alpha^4)+\alpha^2(K^2+1)\right]^2$$

$$=\left[K^2\left(\frac{4\alpha^2\beta^4}{\rho_A^2}+\frac{2\beta^4}{\rho_A}+\alpha^2\right)+\left(\alpha^2+\frac{2\alpha^4}{\rho_A}\right)\right]^2$$

$$= \kappa^4\left(\frac{4\alpha^2\beta^4}{\rho_A^2}+\frac{2\beta^4}{\rho_A}+\alpha^2\right)^2$$

$$+ \; 2\kappa^2\left(\alpha^2+\frac{2\alpha^4}{\rho_A}\right)\left(\frac{4\alpha^2\beta^4}{\rho_A^2}+\frac{2\beta^4}{\rho_A}+\alpha^2\right)$$

$$+ \; \left(\alpha^2+\frac{2\alpha^4}{\rho_A}\right)^2 \quad .$$

Collecting coefficients of like powers of κ gives

$$\kappa^4\Bigg[\left(4\beta^4+\rho_A^2\right)\frac{\beta^4}{\rho_A^2}\left(1+\frac{2\alpha^2}{\rho_A}\right)^2 - \left(\frac{4\alpha^2\beta^4}{\rho_A^2}+\frac{2\beta^4}{\rho_A}+\alpha^2\right)^2\Bigg]$$

$$+ \; 2\kappa^2\Bigg\{\frac{\rho_A^2}{\rho_A^2}\;\beta^4\left(1+\frac{2\alpha^2}{\rho_A}\right)^2-\alpha^2\left(1+\frac{2\alpha^2}{\rho_A}\right) \times \left(\frac{4\alpha^2\beta^4}{\rho_A^2}+\frac{2\beta^4}{\rho_A}+\alpha^2\right)\Bigg\}$$

$$+ \; \left(\rho_A^2 - 4\alpha^4\right)\frac{\beta^4}{\rho_A^2}\left(1 + \frac{2\alpha^2}{\rho_A}\right)^2 - \alpha^4\left(1 + \frac{2\alpha^2}{\rho_A}\right)^2 \quad . \qquad (5.2.1)$$

Equation (5.2.1) is of the form

$$a\kappa^4 + b\kappa^2 + c = 0 \quad .$$

The coefficients can be simplified. Expanding a in (5.2.1):

$$\left(4\beta^4+\rho_A{}^2\right)\ \frac{\beta^4}{\rho_A{}^2}\ \left(1+\frac{4\alpha^2}{\rho_A}+\frac{4\alpha^4}{\rho_A{}^2}\right)$$

$$-\left[\frac{16\alpha^4\beta^8}{\rho_A{}^4}+\frac{16\alpha^2\beta^8}{\rho_A{}^3}+\frac{8\alpha^4\beta^4}{\rho_A{}^2}\right.$$

$$\left.+\frac{4\beta^8}{\rho_A{}^2}+\frac{4\alpha^2\beta^4}{\rho_A}+\alpha^4\right]$$

$$=\frac{\beta^4}{\rho_A{}^2}\left(4\beta^4+\frac{16\alpha^2\beta^4}{\rho_A}+\frac{16\alpha^4\beta^4}{\rho_A{}^2}+\rho_A{}^2+4\alpha^2\rho_A+4\alpha^4\right)$$

$$-\left(\frac{16\alpha^4\beta^8}{\rho_A{}^4}+\frac{16\alpha^2\beta^8}{\rho_A{}^3}+\frac{8\alpha^4\beta^4}{\rho_A{}^2}\right.$$

$$\left.+\frac{4\beta^8}{\rho_A{}^2}+\frac{4\alpha^2\beta^4}{\rho_A}+\alpha^4\right)$$

$$=\beta^4-\alpha^4-\frac{4\alpha^4\beta^4}{\rho_A{}^2}\ .$$

Next we consider b: factoring out $2\left(1+\frac{2\alpha^2}{\rho_A}\right)$ gives

$$2\left(1+\frac{2\alpha^2}{\rho_A}\right)\left[\beta^4+\frac{2\alpha^2\beta^4}{\rho_A}\right.$$

$$\left.-\left(\frac{4\alpha^4\beta^4}{\rho_A{}^2}+\frac{2\alpha^2\beta^4}{\rho_A}+\alpha^4\right)\right]$$

$$=2\left(1+\frac{2\alpha^2}{\rho_A}\right)\left[\beta^4-\alpha^4-\frac{4\alpha^4\beta^4}{\rho_A{}^2}\right].$$

Equation (5.2.1) can be rewritten as

$$\kappa^4+2\left(1+\frac{2\alpha^2}{\rho_A}\right)\kappa^2+\left(1+\frac{2\alpha^2}{\rho_A}\right)^2$$

$$\times\ \frac{\left[\dfrac{\beta^4}{\rho_A{}^2}(\rho_A{}^2-4\alpha^4)-\alpha^4\right]}{\beta^4-\alpha^4-\dfrac{4\alpha^4\beta^4}{\rho_A{}^2}} = 0$$

or

$$\kappa^4 + 2\left(1+\frac{2\alpha^2}{\rho_A}\right)\kappa^2 + \left(1+\frac{2\alpha^2}{\rho_A}\right)^2 = 0.$$

Since

$$\left(\kappa^2+\left(1+\frac{2\alpha^2}{\rho_A}\right)\right)^2 > 0 \qquad (5.2.2)$$

κ is positive there are no critical points. The only possible factor in the preceding analysis that can vanish is,

$$0 = \beta^4 - \alpha^4 - \frac{4\alpha^4\beta^4}{\rho_A{}^2}$$

or

$$\rho_A{}^2 = \frac{4\sin^4\frac{\theta_\psi}{2}\sin^4\frac{\theta_\phi}{2}}{\sin^4\frac{\theta_\phi}{2} - \sin^4\frac{\theta_\psi}{2}}.$$

Using this result in (5.1.29) gives

$$\kappa^2 < 0$$

which is not possible. Thus we have that the error formula (5.1.25a) is a decreasing or increasing function of κ. From (5.1.25a)

$$\epsilon(1) = \left\{\left[1+\frac{1}{\rho_A^2}\sin^4\frac{\theta_\phi}{2}-\frac{1}{\rho_A^2}\sin^4\frac{\theta_\psi}{2}\right]^{1/2} - \frac{1}{\rho_A}\sin^2\frac{\theta_\psi}{2}\right\} \times \left\{1+\frac{1}{\rho_A}\sin^2\frac{\theta_\psi}{2}\right\}^{-1}$$

$$\epsilon(\infty) = \left[1+\frac{4}{\rho_A^2}\sin^4\frac{\theta_\phi}{2}\right]^{1/2} - \frac{2}{\rho_A}\sin^2\frac{\theta_\phi}{2} . \qquad (5.2.3)$$

It is clear by inspection of the above equations that $\epsilon(\kappa)$ is an increasing or decreasing function of κ depending on the choice of θ_ϕ, θ_ψ. This also implies that convergence is fastest at one end of the interval for κ determined in the previous section because of the monotonic variation of ϵ.

CHAPTER 6 INVERSE PROBLEM WITH METRIC F NON-ZERO

6.1 FORMULATION OF GAUSS EQUATION

In previously considered flow problems the curvilinear coordinate system was chosen with the coefficient F of (2.1.10) zero. In the present chapter Gauss' equation is formulated under similar flow assumptions as the problem considered in Chapter 2 except with non-zero F. An equation similar to that examined in Chapter 5 is derived for which the previous stability analysis applies. The algorithm of Chapter 2 is utilized, in this case using the ADI (Alternating Direction Implicit) method as well as SLOR. ADI is well known to have a faster rate of convergence than SLOR for certain classes of PDEs. The goal in using ADI is to achieve a reduction in computational count and maintain the accuracy of SLOR.

Several airfoil profiles were considered and their accuracy compared with previous results. The variation in choice of F and convergence rate was also examined for a particular problem. To begin with, we reformulate the problem by replacing F in Gauss'equation in terms of E, G and θ, the angle of intersection of the coordinate curves ψ=constant and ϕ=constant. Using (2.1.10) and using vector notation yields

$$F = x_\phi x_\psi + y_\phi y_\psi$$

$$= \underset{\sim}{r}_\phi \bullet \underset{\sim}{r}_\psi$$

$$= |\underset{\sim}{r}_\phi| \bullet |\underset{\sim}{r}_\psi| \cos\theta$$

$$= (EG)^{½}\cos\theta \qquad (6.1.1)$$

where θ is a function of position,

$$\theta = \theta(\phi(x,y),\psi(x,y)) \quad . \qquad (6.1.2)$$

Use of (6.1.1) and (6.1.2) in (2.1.12b) yields

$$\Gamma^2_{11} = \frac{-FE_\phi + 2EF_\phi - EE_\psi}{2W^2}$$

$$= \frac{-(EG)^{½}\cos\theta E_\phi + 2E((EG)^{½}\cos\theta)_\phi - EE_\psi}{2EG\sin^2\theta} \qquad (6.1.3)$$

such that

$$W = \left\{ EG - F^2 \right\}^{½}$$

$$= (EG)^{½}\left[1 - \cos^2\theta\right]^{½}$$

$$= (EG)^{½}\sin\theta \quad .$$

Also,

$$\Gamma^2_{12} = \frac{EG_\theta - (EG)^{½}\cos\theta E_\psi}{2EG\sin^2\theta} \quad . \qquad (6.1.4)$$

Using (6.1.3) and (6.1.4), Gauss' equation becomes

$$\left\{\frac{(EG)^{½}\sin\theta}{E}\left(\frac{-(EG)^{½}\cos\theta E_\phi + 2E((EG)^{½}\cos\theta)_\phi - EE_\psi}{2EG\sin^2\theta}\right)\right\}_\psi$$

$$- \left\{\frac{(EG)^{½}\sin\theta}{E}\left(\frac{EG_\phi - (EG)^{½}\cos\theta E_\psi}{2EG\sin^2\theta}\right)\right\}_\phi = 0 \ ,$$

or, on simplification,

$$\left\{\frac{1}{2(EG)^{1/2}\sin\theta}\left(-\left(\frac{G}{E}\right)^{1/2}\cos\theta E_{\phi}+2((EG)^{1/2}\cos\theta)_{\phi}-E_{\psi}\right)\right\}_{\psi}$$

$$-\left\{\frac{1}{2(EG)^{1/2}\sin\theta}\left(G_{\phi}-\left(\frac{G}{E}\right)^{1/2}\cos\theta E_{\psi}\right)\right\}_{\psi} = 0. \qquad (6.1.5)$$

Considering the irrotational case the vorticity equation becomes

$$\left(\frac{(EG)^{1/2}\cos\theta}{(EG)^{1/2}\sin\theta}\right)_{\phi} - \left(\frac{E}{(EG)^{1/2}\sin\theta}\right)_{\psi} = 0 ,$$

which reduces to

$$(\cot\theta)_{\phi} - \left(\left(\frac{E}{G}\right)^{1/2}\csc\theta\right)_{\psi} = 0 . \qquad (6.1.6)$$

For simplicity, the incoming flow is considered to be described by linear functions of ϕ, ψ with constants A, B, C and D,

$$x_{\infty}(\phi,\psi) = A\phi + B\psi \quad ; \quad A,\ B \in \Re \quad ;$$

$$y_{\infty}(\phi,\psi) = C\phi + D\psi \quad ; \quad C,\ D \in \Re \quad . \qquad (6.1.7)$$

It follows using (2.1.10), that at infinity

$$E_{\infty} = x_{\phi}^{2} + y_{\phi}^{2} = A^{2} + C^{2} ,$$

$$F_{\infty} = x_{\phi}x_{\psi} + y_{\phi}y_{\psi} = AB + CD = \cos\theta\left[(A^{2}+C^{2})(B^{2}+D^{2})\right] ,$$

$$G_{\infty} = x_{\psi}^{2} + y_{\psi}^{2} = B^{2} + D^{2} \qquad (6.1.8)$$

$$\frac{y_{\phi}}{x_{\phi}} = \tan\alpha_{\infty} = \frac{C}{A} . \qquad (6.1.9)$$

From the vorticity equation (6.1.6), assuming constant θ,

$$E = g^2(\phi)G$$

so that from the above relations,

$$g(\phi) = \left(\frac{A^2 + C^2}{B^2 + D^2}\right)^{½} = \kappa^{-1} \quad , \ \kappa = \text{constant.} \qquad (6.1.10)$$

From (6.1.7) – (6.1.10) we observe that it is sufficient to specify A, B, C, D to find $\cos\theta$, $g(\phi)$, α_∞. For example, if

$$x_\infty = \phi + \psi \quad ,$$

$$y_\infty = \psi$$

then

$$\cos\theta = \frac{1}{2^{½}} \quad \text{or} \quad \theta = 45^\circ$$

$$g(\phi) = \frac{1}{2^{½}} \quad , \quad \alpha_\infty = 0^\circ \quad .$$

The physics of the irrotational problem dictated the existence of a velocity potential implying E=G when F was chosen to be zero. For F≠0, a uniform flow with angle of attack α_∞ at infinity, and a constant angle between the coordinate curves in the (ϕ,ψ) net, we have seen that

$$E = \kappa^{-2}G \quad , \quad \kappa^{-2} = \text{constant.} \qquad (6.1.11)$$

Using (6.1.11) in the Gauss equation (6.1.5) gives

$$\left[\frac{1}{2E\sin\theta}\left(-\cos\theta E_\phi + 2(E\cos\theta)_\phi - \kappa^{-1}E_\psi\right)\right]_\psi$$

$$- \left[\frac{1}{2E\sin\theta}(\kappa E_\phi - \cos\theta E_\psi)\right]_\phi = 0. \qquad (6.1.12)$$

Using the relation $T=\frac{\ln E}{2}$ in (6.1.12) gives

$$\left[\frac{1}{\sin\theta}(T_\phi\cos\theta - \kappa^{-1}T_\psi)\right]_\psi$$

$$-\left[\frac{1}{\sin\theta}(\kappa T_\phi - \cos\theta T_\psi)\right]_\phi = 0$$

or

$$\left[T_\phi\cot\theta - \kappa^{-1}T_\psi\csc\theta\right]_\psi - \left[\kappa T_\phi\csc\theta - \cot\theta T_\psi\right]_\phi = 0$$

or

$$T_{\phi\psi}\cot\theta - \kappa^{-1}T_{\psi\psi}\csc\theta - \kappa T_{\phi\phi}\csc\theta + \cot\theta T_{\phi\psi} = 0 .$$

Let

$$\nabla^2 = \kappa\frac{\partial^2}{\partial\phi^2} + \kappa^{-1}\frac{\partial^2}{\partial\psi^2} \qquad (6.1.13)$$

then by (6.1.13), (6.1.12) becomes

$$\csc\theta\nabla^2 T - 2\cot\theta T_{\phi\psi} = 0 . \qquad (6.1.14)$$

A solution is required for (6.1.14) subject to far field boundary conditions

$\theta_\infty = \theta_\circ$, $T_\infty = 0$ at $\phi = \phi_\infty$, $\psi = \psi_\infty$.

For boundary conditions along the coordinate curve $\psi=0$ we use (2.1.12a), (6.1.3) and (6.1.11) to get

$$\alpha_\phi = T_\phi\cot\theta - \kappa^{-1}T_\psi\csc\theta \qquad (6.1.15a)$$

$$\alpha_\psi = \kappa T_\phi\csc\theta - \cot\theta T_\psi \qquad (6.1.15b)$$

in which α is determined from

$$\alpha = \tan^{-1}[f'(X)] \quad . \qquad (6.1.16a)$$

As in the algorithm in Chapter 2, x may be determined by differencing

$$x_\phi = E^{\frac{1}{2}}\cos\alpha \tag{6.1.16b}$$

along $\psi=0$ subject to $x(0)=x_0$. Alternatively, x can be calculated throughout the entire flow field. To accomplish this, second order PDEs in x and y are required. Referring to equations (2.1.11b) and (6.1.1) we find, using elementary trigonometric identities,

$$x_\psi = \frac{(EG)^{\frac{1}{2}}}{E^{\frac{1}{2}}}\cos\theta\cos\alpha - \frac{(EG)^{\frac{1}{2}}}{E^{\frac{1}{2}}}\sin\theta\sin\alpha \tag{6.1.17a}$$

$$= G^{\frac{1}{2}}\cos(\theta+\alpha) = G^{\frac{1}{2}}(\cos\theta\cos\alpha - \sin\theta\sin\alpha)$$

$$y_\psi = \frac{(EG)^{\frac{1}{2}}}{E^{\frac{1}{2}}}\cos\theta\sin\alpha + \frac{(EG)^{\frac{1}{2}}}{E^{\frac{1}{2}}}\sin\theta\cos\alpha \tag{6.1.17b}$$

$$= G^{\frac{1}{2}}\sin(\theta+\alpha) = G^{\frac{1}{2}}(\sin\theta\cos\alpha + \cos\theta\sin\alpha) .$$

We see (6.1.17a-b), by use of (6.1.11) together with (2.1.11a) can be written as two pairs of equations,

$$\cos\theta y_\psi - \sin\theta x_\psi = G^{\frac{1}{2}}\sin\alpha = \kappa E^{\frac{1}{2}}\sin\alpha ,$$

$$y_\phi = E^{\frac{1}{2}}\sin\alpha \tag{6.1.18}$$

$$\sin\theta y_\psi + \cos\theta x_\psi = G^{\frac{1}{2}}\cos\alpha = \kappa E^{\frac{1}{2}}\cos\alpha ,$$

$$x_\phi = G^{\frac{1}{2}}\cos\alpha \quad . \tag{6.1.19}$$

By inspection of (6.1.18), (6.1.19) we have

$$\kappa y_\phi = y_\psi\cos\theta - \sin\theta x_\psi \tag{6.1.20a}$$

$$\kappa x_{\phi} = y_{\psi}\sin\theta + \cos\theta x_{\psi} \quad . \tag{6.1.20b}$$

Solving (6.1.20a-b) for x_{ψ} and y_{ψ} yields

$$x_{\psi} = \frac{\begin{vmatrix} \cos\theta & \kappa y_{\phi} \\ \sin\theta & \kappa x_{\phi} \end{vmatrix}}{\begin{vmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{vmatrix}}$$

$$= \kappa(\cos\theta x_{\phi} - \sin\theta y_{\phi}) \tag{6.1.21a}$$

$$y_{\psi} = \frac{\begin{vmatrix} \kappa y_{\phi} & -\sin\theta \\ \kappa x_{\phi} & \cos\theta \end{vmatrix}}{\begin{vmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{vmatrix}}$$

$$= \kappa(y_{\phi}\cos\theta + x_{\phi}\sin\theta) \quad . \tag{6.1.21b}$$

Rewriting (6.1.21a-b) gives

$$\frac{1}{\kappa}\, y_{\psi} = y_{\phi}\cos\theta + x_{\phi}\sin\theta \tag{6.1.22a}$$

$$\frac{1}{\kappa}\, x_{\psi} = x_{\phi}\cos\theta + y_{\phi}\sin\theta \quad . \tag{6.1.22b}$$

Taking the derivative of (6.1.20a), (6.1.22a) with respect to ϕ, ψ respectively and adding yields

$$\kappa x_{\phi\phi} + \frac{1}{\kappa}\, x_{\psi\psi} = 2\cos\theta x_{\phi\psi} \quad . \tag{6.1.23a}$$

For y; one also obtains in an identical way,

$$\kappa y_{\phi\phi} + \frac{1}{\kappa}\, y_{\psi\psi} = 2\cos\theta\; y_{\phi\psi} \quad . \tag{6.1.23b}$$

Let

$$b = -2\cos\theta \quad , \quad a = \kappa^{-1} \quad , \quad c = \kappa$$

then

$$b^2 - 4ac = - 4\sin^2\theta < 0 \ , \ \theta \neq 0 \ .$$

Thus, (6.1.23) are by definition elliptic PDE as is (6.1.14). Similarily, it can be shown from (6.1.15) solving for T_ϕ and T_ϕ

$$T_\phi = \kappa^{-1}\alpha_\psi \csc\theta - \alpha_\phi \cot\theta \tag{6.1.24a}$$

$$T_\psi = \cot\theta \ \alpha_\psi - \alpha_\phi \kappa \csc\theta \ . \tag{6.1.24b}$$

Taking the cross-derivative of (6.1.24) and subtracting yields

$$\kappa\alpha_{\phi\phi} + \frac{1}{\kappa}\alpha_{\psi\psi} = 2\cos\theta\alpha_{\phi\psi} \quad . \tag{6.1.25}$$

An inverse problem could be attempted using (6.1.25) and boundary condition (6.1.24a), whereby the speed is specified apriori, the angle α determined, subsequently χ can be determined from (6.1.23a) with boundary condition (6.1.17a). The integration of a pair of ODE described in section 6.3 is then made to determine the profile.

6.2 VARYING THETA

In this section the convergence characteristics of the numerical scheme of Chapter 2 is investigated with respect to varying κ. θ is taken constant. Gauss' equation (6.1.14) with boundary conditions (6.1.15a) is also used As well, χ is determined from (6.1.23a). To vary κ one or more of the constants A, B, C and D of the previous section are slightly altered. By inspection, there is more computational work to be done simply because of the extra terms involving the cross-derivative terms $T_{\phi\psi}$, $x_{\phi\psi}$, $y_{\phi\psi}$. It can be seen that elliptic PDE of the form (6.1.23) can be transformed by linear transformations to Laplace's equation (by taking ϕ and ψ as linear functions of variables ξ and η). The existence and uniqueness of a solution for Laplace's equation with Dirichlet or Von Neumann boundary conditions is verified in [12].

Central differencing (6.1.14) yields

$$K^{-1}\left(T_{J,K+1} - 2T_{J,K} + T_{J,K-1}\right) + K\left(T_{J+1,K} - 2T_{J,K} + T_{J-1,K}\right)$$

$$- .5\cos\theta\left(T_{J+1,K+1}-T_{J+1,K-1}-T_{J-1,K+1}+T_{J-1,K-1}\right) = 0 \qquad (6.2.1)$$

such that for K=1 or ψ=0,

$$T_{\psi\psi} = 2\left[\frac{T_{J,2} - T_{J,1}}{D\psi^2} - \frac{T_\psi}{D\psi}\right] \qquad (6.2.2a)$$

$$T_\psi = - K\sin\theta\alpha_\phi + K\cos\theta T_\phi \qquad (6.2.2b)$$

$$T_{\phi\psi} = \frac{1}{2D\phi D\psi}\left[T_{J+1,K+1}-T_{J+1,K}-T_{J-1,K+1}+T_{J-1,K}\right] \cdot \qquad (6.2.2c)$$

To verify (6.2.2c), expand the RHS terms in a Taylor Series,

$T_{J\pm1,K\pm1}$

$$= T_{J\pm1,K} \pm D\psi\, T^{1}_{J\pm1,K} + \frac{D\psi^2}{2!}\, T^{11}_{J\pm1,K} + \ldots$$

$$= T_{J,K} \pm D\phi\, T^{0}_{J,K} + \frac{D\phi^2}{2!}\, T^{00}_{J,K} + \ldots$$

$$\pm\, D\psi\left(T^{1}_{J,K} \pm D\phi T^{01}_{J,K} + \ldots\right)$$

$$+\, \frac{D\psi^{2}}{2!}\left(T^{11}_{J,K} + \ldots\right) \qquad (6.2.3)$$

$$T_{J\pm 1,K} = T_{J,K} \pm D\phi T^{0}_{J,K} + \frac{D\phi^{2}}{2!}T^{00}_{J,K} + \ldots .$$

Using (6.2.3) in (6.2.2c) we find, (note that all terms on right hand side have subscript J,K which has been dropped for convenience)

$$T_{J+1,K+1} - T_{J+1,K} - T_{J-1,K+1} + T_{J-1,K}$$

$$= T + D\phi T^{0} + \frac{D\phi^{2}}{2}T^{00} \pm D\psi(T^{1} + D\phi T^{01} + \ldots)$$

$$+ \frac{D\psi^{2}}{2}(T^{11} + \ldots) - T - D\phi T^{0} - \frac{D\phi^{2}}{2!}T^{00}$$

$$- \left[T - D\phi T^{0} + \frac{D\phi^{2}}{2!}T^{00} + D\psi(T^{1} - D\phi T^{01} + \ldots) + \frac{D\psi^{2}}{2!}(T^{11} + \ldots)\right]$$

$$+ T - D\phi T^{0} + \frac{D\phi^{2}}{2!}T^{00} + \ldots .$$

$$= 2T - 2T + 2D\phi(T^{0} - T^{0}) + D\psi(T^{1} - T^{1}) + D\phi^{2}(T^{00} - T^{00})$$

$$+ \frac{D\psi^{2}}{2!}(T^{11} - T^{11}) + 2D\phi D\psi T^{01} + O(D\phi D\psi^{2})$$

$$= 2D\phi D\psi T^{01} + \text{H.O.T.}$$

where 0, 1 denote the derivate in the ϕ,ψ direction respectively. Thus use of (6.2.2c) is validated. From (6.1.7) – (6.1.9)

$$\kappa = \left[\frac{B^2 + D^2}{A^2 + C^2}\right]^{\frac{1}{2}} . \qquad (6.2.4a)$$

$$\cos\theta = \frac{AB + CD}{\left[(A^2 + C^2)(B^2 + D^2)\right]^{\frac{1}{2}}} ,$$

$$\sin\theta = \frac{AD + BC}{\left[(A^2 + C^2)(B^2 + D^2)\right]^{\frac{1}{2}}} ,$$

$$\tan\theta = \frac{AD - BC}{AB + CD} . \qquad (6.2.4b)$$

To investigate the effect on convergence of taking a non-orthogonal grid in the ϕ,ψ system it is necessary to take F non-zero. This investigation can be carried out in one of two ways : by an Fourier series approach, as in Chapter 5 whereby it was shown precisely for which angles the numerical procedure converges fastest, or simply by numerical experimentation or trial and error, to establish an interval of convergence for a solution to exist for a given angle θ and a non-orthogonal grid. The results should agree with the stability analysis of the previous chapter. For example, at outer regions of the computational domain, we let the linear functions of ψ and ϕ be

$$x_\infty = \phi + A_n \psi , \quad n = 0, 1, 2, \ldots, N \qquad (6.2.5)$$

$$y_\infty = \psi$$

for some positive integer N, where A_n is a positive constant. To solve the problem numerically the case θ=0 is excluded since then there is no curvilinear net. Various values of θ based on (6.2.5) and (6.2.4b) were used. The profile taken was the circle of radius .5. ADI, described in Chapter 4, was used. Equations (6.2.1) and (6.1.23a) are solved with respective boundary conditions (6.2.2) and (6.1.17a) for K=1 or ψ=0.

Inspection of Table VIII reveals fast convergence for some values of n and slow convergence for others as predicted by the analysis of Chapter 5. The behaviour also suggests, as predicted in Chapter 5, the existence of a angle θ which optimizes convergence. A possible explanation for the fast convergence at a value of κ close to but not equal to three is that the error function, as shown in Chapter 5, is a decreasing function for certain values of κ, i.e., equality in (5.1.26a). Thus the eigenvalues of the error reduction matrix of the ADI process are minimized for values κ close to the upper bounds of the interval determined in Chapter 5.

Various examples of non-orthogonal grids for grid size [-1.5,1.5] x [0,2], (64 x 43) grid points and tolerance level

$$\max_{J,K} \left| T^{N+1}_{J,K} - T^{N}_{J,K} \right| < .0001$$

were considered.

For the circle of radius one-half the maximum velocity is 2 and occurs at the top of the circle. The variation of accuracy with respect to the exact solution that was evident in Table VIII (column 5) suggests that as θ varies the problem changes in some sense. In the next section the ADI method is utilized for solving problems when the angle α or metric E is specified on the airfoil surface apriori and the resultant profile determined through the integration of a second order ODE.

6.3 SPEED SPECIFIED: THE INVERSE PROBLEM

We have seen that by specifying a profile $\mathcal{Y}$ as a function of $\mathcal{X}$ and knowing the value of the leading and trailing edge in terms of $\mathcal{X}$, say $\mathcal{X}_{LE}$, $\mathcal{X}_{TE}$, that a solution to the irrotational, inviscid, incompressible problem can be obtained using the algorithm of Chapter 3. It is clear that knowing the stagnation points in terms of $\mathcal{X}$ simplifies the problem.

In this section a method of solution for the inverse problem is considered in that the speed q is specified at grid points ϕ_J, ψ=0. It is required to find the resulting shape having this velocity profile. An algorithm is developed and the computer listing is given (cf. Appendix C). Not a great amount of published work is readily available on this particular problem. In a recent article, Jones and Eggleston [25] used an optimization method in the design of supercritical airfoils.

One of the problems that arises is that there could be infinitely many airfoil surfaces $y=f(x)$ corresponding to the particular velocity profile which is specified. Thus it is necessary to find the angle distribution which uniquely corresponds with the choice of grid points chosen to represent the function y in the transformed plane. To find the angle α one of two approaches can be taken. In the first it is necessary to prescribe values for $T_{J,1}$ (T=lnE/2) for J=1,...,JMAX, then to solve for α from (6.1.25) throughout the computational domain with boundary condition (6.1.24a) used on $\psi=0$ or K=1 and α_∞ determined from (6.1.9).

The other approach is to specify grid points ϕ_1, ϕ_2 for which α vanishes and which satisfy

$$\phi_1 < \phi_{LE} < \phi < \phi_{TE} < \phi_2 .$$

α can be determined simply by forward and backward differencing (6.1.15a) from the leading edge and trailing edge respectively towards the center of the computational domain after finding values for T_ψ and T_ϕ on $\psi=0$ or K=1. Finding these values can be accomplished by solving the Dirichlet problem (6.2.1) having specified values for T on the boundaries. Once α is determined accurately x is determined as in the algorithm in Chapter 3. Subsequently the increments Dx_J along the profile can be found. These values are required for the determination of the function $y=f(x)$ as seen below.

Consider the system of ordinary differential equations,

$$\frac{du_1}{dx} = u^2 = f_1(x\ ,\ u_1\ ,\ u_2) \tag{6.3.1a}$$

$$\frac{du_2}{dx} = \left[1 + u_2^2\right]^{1.5} \alpha_\phi\ e^{-T} = f_2(x\ ,\ u_1\ ,\ u_2) \tag{6.3.1b}$$

$$u_1(x_{LE}) = U_1 \quad , \quad u_2(x_{TE}) = U_2 \quad .$$

Consider a curvilinear net defined by

$$x = x(\phi,\psi) \quad , \quad y = y(\phi,\psi) \quad .$$

Let

$$y = f(x) = u_1 \quad .$$

Consider the bounded intervals [a,b] , [c,d] for real numbers a, b, c, d. Suppose we are given an arbitrary set of values $\{\alpha_\phi e^{-T}\}$ defined on [a,b] × [c,d]. Then provided this set is bounded and continuous a unique solution to (6.3.1) exists [14]. Thus the product $\alpha_\phi e^{-T}$ uniquely determines y as a function of x. It is also known that given a function $y=f(x)$, T and α are uniquely determined according to the algorithm of Chapter 3. From (2.1.22a) values of T and α determine x. Therefore we can conclude, that if from the values of T, specified apriori, unique values of α can be obtained then $y=f(x)$ is determined uniquely subject to two initial conditions

$$y(x_{LE}) = y_{LE} \quad , \quad y(x_{TE}) = y_{TE}$$

and the O.D.E. (6.3.1). Suppose then we are given a set of values for $T_{J,1}$ either randomly distributed or determined by a set of rules. For example a linear interpolating formula

$$T(\phi_J) = \frac{\phi_J - \phi_{J_{TE}}}{\phi_{LE} - \phi_{TE}} T_{LE}$$

$$+ \frac{\phi_J - \phi_{J_{LE}}}{\phi_{TE} - \phi_{LE}} T_{TE} \qquad (6.3.2)$$

where $J_{LE} \leq J \leq J_{TE}$. Assume that a similar linear interpolating formula describes T for $J < J_{LE}$ and $J > J_{TE}$. We wish to determine the resulting airfoil profile by one of the two approaches mentioned above. A method that is well--suited to the problem and is used is the Non-Linear Finite Difference Algorithm which also uses Newton's method to find zero's of vector systems [11]

$$G(\vec{X}) = \vec{X} - J^{-1}(\vec{X})\ F(\vec{X}) \qquad (6.3.3)$$

where

$$J(\vec{X}) = \qquad (6.3.4)$$

$$\begin{bmatrix} \frac{\partial f_1}{\partial X_1} & \cdots & \frac{\partial f_1}{\partial X_n} \\ \cdot & \cdot & \\ \cdot & \cdot & \\ \frac{\partial f_n}{\partial X_1} & \cdots & \frac{\partial f_n}{\partial X_n} \end{bmatrix}$$

is the Jacobian matrix. Equation (6.3.3) is based on the one dimensional relation

$$\mathcal{G}(x) = x - \phi(x)\mathcal{F}(x) \tag{6.3.5}$$

in which ϕ is determined so that the fixed point x of $\mathcal{G}(x)$ is found.

The differencing method that is used for

$$\frac{dy}{dx} \; , \; \frac{d^2y}{dx^2}$$

is, from [3],

$$\left(\frac{dy}{dx}\right)_J = \frac{y_{J+1} - y_{J-1}}{x_{J+1} - x_{J-1}} \tag{6.3.6}$$

$$\left(\frac{d^2y}{dx^2}\right)_J = \tag{6.3.7}$$

$$\left\{2\left[(x_J - x_{J-1})y_{J+1} - (x_{J+1} - x_{J-1})y_J + (x_{J+1} - x_J)y_{J-1}\right]\right\}$$
$$\times \left\{(x_{J+1} - x_J)(x_{J+1} - x_{J-1})(x_J - x_{J-1})\right\}^{-1}$$

Employing (6.3.6) and (6.3.7) in (6.3.1b) yields

$$\left\{2\left[(x_J - x_{J-1})y_{J+1} - (x_{J+1} - x_{J-1})y_J + (x_{J+1} - x_J)y_{J-1}\right]\right\}$$
$$\times \left\{(x_{J+1} - x_J)(x_{J+1} - x_{J-1})(x_J - x_{J-1})\right\}^{-1}$$
$$- \left[1 + \left(\frac{y_{J+1} - y_{J-1}}{x_{J+1} - x_{J-1}}\right)^2\right]^{3/2}$$
$$\times \frac{\alpha_{J+1} - \alpha_{J-1}}{2\mathcal{D}\phi} e^{-T_J} = 0 \quad . \tag{6.3.8}$$

In our case, according to the notation of (6.3.4), x_i

corresponds to y_i. For computer ordering purposes let

$$J = 1 \quad , \quad J = M$$

correspond to the values

$$J = J_{LE} + 1 \quad , \quad J = J_{TE} - 1$$

respectively. Also let

$$C_{6J} = x_{J+1} - x_J$$

$$C_{7J} = x_{J+1} - x_{J-1}$$

$$C_{8J} = x_J - x_{J-1}$$

$$C_{3J} = .5\ (x_{J+1} - x_J)(x_{J+1} - x_{J-1})(x_J - x_{J-1})$$

$$C_{5J} = \frac{(\alpha_{J+1} - \alpha_{J-1})e^{-T_J}}{2D\phi}$$

$$C_{9J} = \frac{C_{3J}}{C_{7J}}$$

$$C_{4J} = \frac{y_{J+1} - y_{J-1}}{C_{7J}}$$

and for compactness rewrite (6.3.8) as

$$C_{8J}y_{J+1} - C_{7J}y_J + C_{6J}y_{J-1}$$

$$- C_{3J}C_{5J}\left[1 + \left(C_{4J}\right)^2\right]^{3/2} = 0 \ . \qquad (6.3.9)$$

From (6.3.9) we can readily calculate $J(\vec{y})$ to be

$$J(\vec{y}) = \mathrm{TRID}(AA,BB,CC)_J$$

where

$$AA_J = -C_{6J} - 3C_{9J}\, C_{4J}\, C_{5J} \left[1 + \left(C_{4J}\right)^2\right]^{1/2} ,$$

$$BB_J = C_{7J} \quad , \qquad\qquad (6.3.10)$$

$$CC_J = -C_{8J} + 3C_{9J}\, C_{4J}\, C_{5J} \left[1 + \left(C_{4J}\right)^2\right]^{1/2} .$$

For J = 1

$$AA_1 = 0$$

and for J = M

$$CC_M = 0 \quad .$$

In employing Newton's method to solve the system (6.3.1) at each iteration the MxM linear system,

$$T(AA,BB,CC)(W_1 \;,\ldots,\; W_M) \qquad\qquad (6.3.11)$$

$$= \left(C_{81}y_2 - C_{71}y_1 + C_{61}y_{LE} - (C_3C_5)_1\left[1 + (C_{41})^2\right]^{3/2},\ldots\right.$$

$$\left.\ldots,C_{8M}y_{TE} - C_{7M}y_M + C_{6M}y_{M-1} - (C_3C_5)_M\left[1 + (C_{4M})^2\right]^{3/2}\right)$$

is solved for $W_1,\ldots,W_M$ such that

$$y_J^{(K)} = y_J^{(K)} + W_J^{(K)} \qquad J_{LE+1} \le J \le J_{TE-1} \quad .$$

K denotes the iteration level and

$$\left(W_1^{(0)} \;,\ldots,\; W_M^{(0)}\right) = (0,\ldots,0)$$

is the initial guess.

The values of $x_{J,1}$ and thence increments $Dx_{J,1}$ can be determined by solving the full equation for x, (6.1.23a) with far-field boundary conditions determined from (6.1.7) and derivative boundary condition from (6.1.17a). The following algorithm solves the inverse problem using the first approach as described in the beginning of this section.

(1) Construct a grid as in the previous algorithms of Chapter 3 and 4. Specify $\mathcal{X}_\infty$ and y_∞ as in (6.1.7). Initialize unknowns T_∞ and α_∞ according to (6.1.8) and (6.1.9).

(2) Specify $T_{J,1}$ according to predetermined criteria along the coordinate line ψ=0 such as a linear interpolating formula. Solve for α throughout the flow field by (6.1.25) using SLOR or ADI with boundary condition (6.1.24a) on K=1. For α_∞ use the value determined in step (1).

(3) Knowing T and α along ψ=0, solve for $\mathcal{X}$ throughout by using (6.1.23a) and SLOR or ADI with boundary condition (6.1.17a), K=1. The increments $D\mathcal{X}$ can then be determined for the solution of (6.3.1).

(4) Having values for $D\mathcal{X}$, T, α integrate (6.3.1) by Newton's method described above.

Using a linear interpolating form for T the algorithm was tested to yield various profiles with varying θ. $\mathcal{X}_\infty$ and y_∞ were taken to satisfy (6.2.5). The ADI method was used. The convergence characteristics followed those of section 2. Figure 9(e-f) gives the various profiles generated by choosing particular values for θ.

The grid size was 65 x 43 with $\phi_{MAX} = 2 = -\phi_{MIN}$. Equal sized grid spacing was used. The tolerance level was .0001 with maximum norm. The profile was taken to lie on the grid points ϕ_{23} to ϕ_{43} , ψ=0. T was a linear function of ϕ linearly connecting the following points

$$T_{23} = .55 \ , \ T_{24} = .50 \ , \ T_{33} = -.2 \ ,$$

$$T_{42} = .5 \ , \ T_{43} = .55 \ , \ T_{1} = 0 \ , \ T_{65} = 0 \ .$$

As θ decreases from 90 degrees and the algorithm becomes more stable and the profile increases in size. To check the validity of the resultant profile it would be necessary to represent the profile as polynomial in χ and then to solve the problem according to the algorithm of Chapter 3.

CONCLUSION

The algorithms presented in the thesis deal with a curvilinear coordinate system and the solution of elliptic partial differential equations to generate this system. Unlike previous grid generation systems the flow equations were interlocked with the fluid flow equations, i.e., one of a pair of coordinate lines of the computational domain was taken to be the streamlines. This then, represents an original approach to solving fluid dynamic flows by finite difference techniques using high speed computers. Among the features of this new approach were its directness and ability to solve the flows considered in a fast and efficient manner. However, it has yet to be tested for the more complicated, relevant flows of modern Computational Fluid Dynamics. However, one of the goals of the thesis was to solve several standard problems in a new manner, thereby opening up the possibility of solving more complicated flows, such as those with compressibility and circulatory effects.

To investigate flows with circulation and lift would require use of the entire computational domain. Questions that might be considered are : what stagnation point corresponds to a particular angle of attack and circulation, or given a stagnation point and angle of attack what is the corresponding lift. To solve compressible problems requires the governing equations to be reformulated with variable density. Other types of problems at a more advanced level are : viscous flows with various Reynold's numbers, flows with separated vortices. For the latter flow it might be feasible to construct a separate conmputational domain for the reversing flow and match with the regular computational domain at a particular streamline. In conclusion, it is promising that a series of CFD flows can be solved.

REFERENCES

1. Thompson, J. F., Warsi, Z. U. A. and Mastin, C. W. Numerical Grid Generation, Foundations and Applications, North-Holland, New York, 1985.

2. Thames, F. C., Thompson, J. F., Mastin, C. W., and Walker, R. L. Numerical Solutions for Viscous and Potential Flow about Arbitrary Two Dimensional Bodies, J. of Comput. Physics Vol.24, pp.245-247, 1977.

3. Steger, J. L. Numerical Methods in Fluid Mechanics : Transonic and Viscous Boundary Layer Flow, Lecture Notes, Department of Aeronautics and Astronautics, Stanford University, Stanford, California, 1982.

4. Abbott, I. H. and Von Doenhoff, A. E. Theory of Wing Sections, Dover Publications, N. Y., 1959.

5. Van Dyke, M. D. Perturbation Methods in Fluid Mechanics, Parabolic Press, Stanford, CA., 1975.

6. Kuethe, A. M. and Chow, Chuen-Yen Foundations of Aerodynamics : Basis of Aerodynamic Design, 3rd edn, John Wiley & Sons, N. Y., 1976.

7. Martin, M. H. The Flow of a Viscous Fluid. I, Archives Rat. Mech. Analysis 41, pp. 266-286, 1971.

8. Docarmo, M. P.
Differential Geometry of Curves and Surfaces,
Prentice-Hall, Inc., Engle-Wood Cliffs, N. J., 1976.

9. Chandna, O. P., Barron, R. M. and Chew, K. T.
Martin's Method Applied to Variably-Inclined Plane
Viscous MHD Flows, Int. J. Engng. Sci., Vol.21,
No.4, pp.373-393, 1983.

10. Kaloni, P. N. and Siddiqui, A. M.
The Flow of a Second Grade Fluid, Int. J. Engng.
Sci., Vol.21, No.10, pp.1157-1169, 1983.

11. Olmstead, J. M. H.
Calculus with Analytic Geometry,
Appleton-Century-Crofts, N. Y., 1966.

12. Garabedian, P. R.
Partial Differential Equations, John Wiley & Sons,
N. Y., 1964.

13. Spiegel, M. R.
Complex Variables, with an Introduction to
Conformal Mapping and its Applications,
McGraw-Hill Book Co., N. Y., 1974.

14. Burden, R. L., Faires, J. D. and Reynolds, A. C.
Numerical Analysis, 2nd edn, Prindle,
Weber & Schmidt, Boston, 1981.

15. Milne-Thomson, C. B. E.
Theoretical Hydrodynamics, 4th Ed., Macmillan & Co.,
London, 1962.

16. Spiegel, M. R.
Mathematical Handbook of Formulas and Tables,
McGraw-Hill Book Co., N. Y., 1968.

17. Birkhoff, G., Varga, R. S. and Young, D.
Alternating Direction Implicit Methods,
Advances in Computers, Vol.3, pp.190-273,
Academic Press, N. Y., 1962.

18. Franklin, J. N.
Matrix Theory, Prentice-Hall Inc., Englewood Cliffs,
N. J., 1968.

19. Carre, B. A.
The Determination of the Optimum Accelerating
Factor for Successive Over-Relaxation, Comput. J.
Vol.4,pp.73-78, 1961.

20. Shortley, G. H. and Weller, R.
The Numerical Solution of Laplace's Equation,
J. of Applied Physics, Vol.9, pp.334-348, 1938.

21. Peaceman, D. W. and Rachford, H. H.
The Numerical Solution of Parabolic and Elliptic
Differential Equations, J. Soc. Indust. Appl.
Math.,Vol.3, pp.28-41, 1955.

22. Frankel, S. P.
Convergence Rates of Iterative Treatments of
Partial Differential Equations, Math. Tables Aid
Comput. Vol.4, pp.66-75, 1950.

23. Stone, H. L.

Iterative Solution of Implicit Approximations of Multidimensional Partial Differential Equations, SIAM J. of Numerical Analysis, Vol.5, pp.530-558, 1968.

24. Isaacson, E. and Keller, H. B.

Analysis of Numerical Methods, John Wiley & Sons, Inc., N. Y., 1966.

25. Eggleston, B. and Jones, D. J.

The Design of Lifting Supercritical Airfoils using a Numerical Optimisation Method. Canadian Aeronautics and Space Journal, Vol. 23, No. 3, 1977.

TABLE I New Algorithm

ϕ	χ	q/u_α (speed)
-.5	-.4673	.8869
-.4286	-.4019	1.1192
-.3517	-.3395	1.1462
-.2857	-.2772	1.1468
-.2143	-.2144	1.1390
-.1428	-.1511	1.1275
-.0714	-.0816	1.1142
0.	-.0224	1.0998
.0714	.0432	1.0849
.1428	.1096	1.0695
.2143	.1770	1.0534
.2857	.2454	1.0360
.3571	.3150	1.0156
.4286	.3861	.9887
.5	.4596	.9411
.5357	.4982	.8887

TABLE II Theodorsen's Results

x (percent)	q/u_∞
0	0
1.25	1.035
2.5	1.127
5.0	1.153
7.5	1.151
10	1.150
15	1.146
20	1.141
30	1.139
40	1.140
50	1.131
60	1.115
70	1.090
80	1.050
90	.996
95	.943
100	0

TABLE III Grid Generation Results

x	q/u_∞
-.5	.0213
-.4583	1.205
-.4167	1.191
-.375	1.182
-.2916	1.166
-.2083	1.145
-.125	1.112
-.0416	1.110
0.0	1.090
.0833	1.069
.1667	1.047
.2083	1.036
.250	1.024
.3333	.996
.4167	.959
.4583	.9510
.500	.7667

TABLE IV

Parabolic Shear for Circle of Radius .5
Comparison of Van Dyke's Perturbation Solution, ϵ=.1,
with Numerical Solution Using Algorithm A

J	EXACT SPEED	ANGLE OF INCL.	NUMER. SPEED
20	0.44261836D 00	0.13619851D 01	1.78837129
21	0.10706203D 01	0.10505036D 01	1.88482938
22	0.13771839D 01	0.88371697D 00	1.95816958
23	0.15878872D 01	0.75760763D 00	2.0177196
24	0.17469128D 01	0.65202466D 00	2.0673800
25	0.18718802D 01	0.55893096D 00	2.10915288
26	0.19717819D 01	0.47418155D 00	2.1442386
27	0.20518551D 01	0.39529977D 00	2.1734122
28	0.21153969D 01	0.32065023D 00	2.1972034
29	0.21645772D 01	0.24906688D 00	2.21599282
30	0.22008472D 01	0.17966225D 00	2.230042237
31	0.22251624D 01	0.11171907D 01	2.2395263
32	0.22381063D 01	0.44622538D-01	2.244553031
33	0.22399593D 01	-0.22186436D-01	2.24516783
34	0.22307297D 01	-0.89247870D-01	2.24135844
35	0.22101556D 01	-0.15711602D 00	2.233064993
36	0.21776786D 01	-0.22639517D 00	2.22015994
37	0.21323810D 01	-0.29778458D 00	2.20244427
38	0.20728713D 01	-0.37214250D 00	2.17964646
39	0.19970791D 01	-0.45058743D 00	2.15137505
40	0.19018794D 01	-0.53467429D 00	2.1170786
41	0.17823546D 01	-0.62673324D 00	2.0759679
42	0.16301728D 01	-0.73060979D 00	2.026843
43	0.14293494D 01	-0.85360317D 00	1.96768665
44	0.11414282D 01	-0.10132502D 01	1.8945960
45	0.60356149D 00	-0.12843909D 01	1.79822155

TABLE V

Parabolic Shear for Circle of Radius .5
Comparison of Van Dyke's Perturbation Solution, ϵ=.1,
with Numerical Solution Using Algorithm B

J	EXACT SPEED	ANGLE OF INCL.	NUMER. SPEED
18	0.64006280D 00	0.12697396D 01	0.56674550
19	0.10738306D 01	0.10542176D 01	1.00312270
20	0.13215281D 01	0.92139656D 01	1.27874290
21	0.15040109D 01	0.81569060D 01	1.46928750
22	0.16503059D 01	0.72363828D 00	1.61499750
23	0.17715991D 01	0.64006502D 00	1.7326068
24	0.18735535D 01	0.56235334D 00	1.8298563
25	0.19595365D 01	0.48892730D 00	1.91096808
26	0.20317396D 01	0.41873059D 00	1.9785290
27	0.20916649D 01	0.35100252D 00	2.0342652
28	0.21403725D 01	0.28516450D 00	2.0793609
29	0.21786191D 01	0.22075555D 00	2.11465222
30	0.22069410D 01	0.15739247D 00	2.1407271
31	0.22257052D 01	0.94743412D-01	2.1579769
32	0.22351408D 01	0.32509109D-01	2.1666501
33	0.22353568D 01	-0.29591617D-01	2.1668660
34	0.22263514D 01	-0.91834081D-01	2.1586197
35	0.22080109D 01	-0.15449930D 00	2.1417817
36	0.21801026D 01	-0.21788620D 00	2.1160976
37	0.21422558D 01	-0.28232610D 00	2.0811698
38	0.20939313D 01	-0.34820142D 00	2.0363944
39	0.20343701D 01	-0.41597202D 00	1.9809355
40	0.19625114D 01	-0.48621494D 00	1.9135899
41	0.18768540D 01	-0.55968863D 00	1.8326363
42	0.17752103D 01	-0.63744599D 00	1.7354694
43	0.16542221D 01	-0.72105739D 00	1.6178612
44	0.15082466D 01	-0.81313054D 00	1.4720581
45	0.13261510D 01	-0.91881544D 00	1.2813137
46	0.10791247D 01	-0.10514744D 01	1.0053778
47	0.64800457D 00	-0.12659160D 01	0.5688508

TABLE VI

Parabolic Shear for Circle of Radius .5
Comparison of Van Dyke's Perturbation Solution, ϵ=.1,
with Numerical Solution Using Algorithm C

J	EXACT SPEED	ANGLE OF INCL.	NUMER. SPEED
21	0.73299866D 00	0.12247777D 01	1.87745130
22	0.12139827D 01	0.98028912D 00	1.97829420
23	0.14863582D 01	0.82630122D 00	2.0553168
24	0.16781709D 01	0.70511991D 00	2.1179665
25	0.18236590D 01	0.60142053D 00	2.17015778
26	0.19375326D 01	0.50859229D 00	2.2138735
27	0.20275588D 01	0.42308099D 00	2.2502858
28	0.20983311D 01	0.34270216D 00	2.2801533
29	0.21527489D 01	0.26597986D 00	2.3039829
30	0.21927007D 01	0.19183832D 00	2.3221306
31	0.22194138D 01	0.11943878D 01	2.33483339
32	0.22336440D 01	0.48083492D-01	2.3422495
33	0.22357787D 01	-0.22847173D-01	2.3444680
34	0.22258853D 01	-0.93941516D-01	2.3415146
35	0.22037217D 01	-0.16579722D 00	2.3333629
36	0.21687085D 01	-0.23906366D 00	2.3199273
37	0.21198578D 01	-0.31449418D 00	2.3010405
38	0.20556371D 01	-0.39302038D 00	2.2764679
39	0.19737224D 01	-0.47587074D 00	2.2458586
40	0.18705361D 01	-0.56478018D 00	2.2087050
41	0.17403196D 01	-0.66240311D 00	2.1642492
42	0.15730254D 01	-0.77324855D 00	2.1113260
43	0.13485058D 01	-0.90627062D 00	2.0479507
44	0.10138233D 01	-0.10850729D 01	1.9702337
45	0.16489334D 01	-0.14953991D 01	1.8687467

TABLE VII Varying Theta

angle θ	total	max.err	No of	Loc.of max	κ
	error ≈	$\times 10^{-3}$ ≈	iter.	err(J,K)	
89.2°	.47	.497	136	(29,1)	1
89.1°	.47	.497	136	(29,1)	1
88.2°	.47	.497	136	(29,2)	1
86.4°	.47	.497	136	(29,2)	1.001
75.96°	.46	.495	134	(29,2)	1.03
64.43°	.44	.496	128	(28,2)	1.12
58.2°	.425	.497	124	(28,2)	1.176
53.1°	.408	.497	119	(28,2)	1.25
48.8°	.393	.4998	112	(28,2)	1.33
45°	.34	.499	128	(26,2)	1.414
41.6°	.314	.494	113	(26,2)	1.505
38.66°	.267	.488	103	(25,2)	1.6
38.3°	.261	.492	100	(28,1)	1.61
36.03°	.092	.47	84	(28,1)	1.70
34.87°	∞	∞	∞	(--)	1.75
33.7°	∞	∞	∞	(--)	1.803

TABLE VIII Varying Theta

angle θ	tot(I+T)	max.err	No of	2-max(T)	κ
	error ≈	$\times 10^{-3}$ ≈	iter.	err(J,K)	
90.0°	.71	1.0	>250	.37	1.
76.0°	.75	2.3	>250	.318	1.03
63.5°	.795	.930	>250	.227	1.12
53.1°	.923	1.1	>250	.134	1.25
45°	.682	.776	>250	.083	1.41
42.2°	.719	8.68	>250	.077	1.49
39.8°	.308	.374	>250	.10	1.56
37.6°	.135	.099	234	.116	1.56
36.5°	.052	.099	230	.168	1.68
35.5°	.05	.099	171	.222	1.72
35.3°	.047	.099	154	.217	1.73
34.6°	.026	.091	116	.239	1.76
33.6°	1.6	9.8	>250	.412	1.8

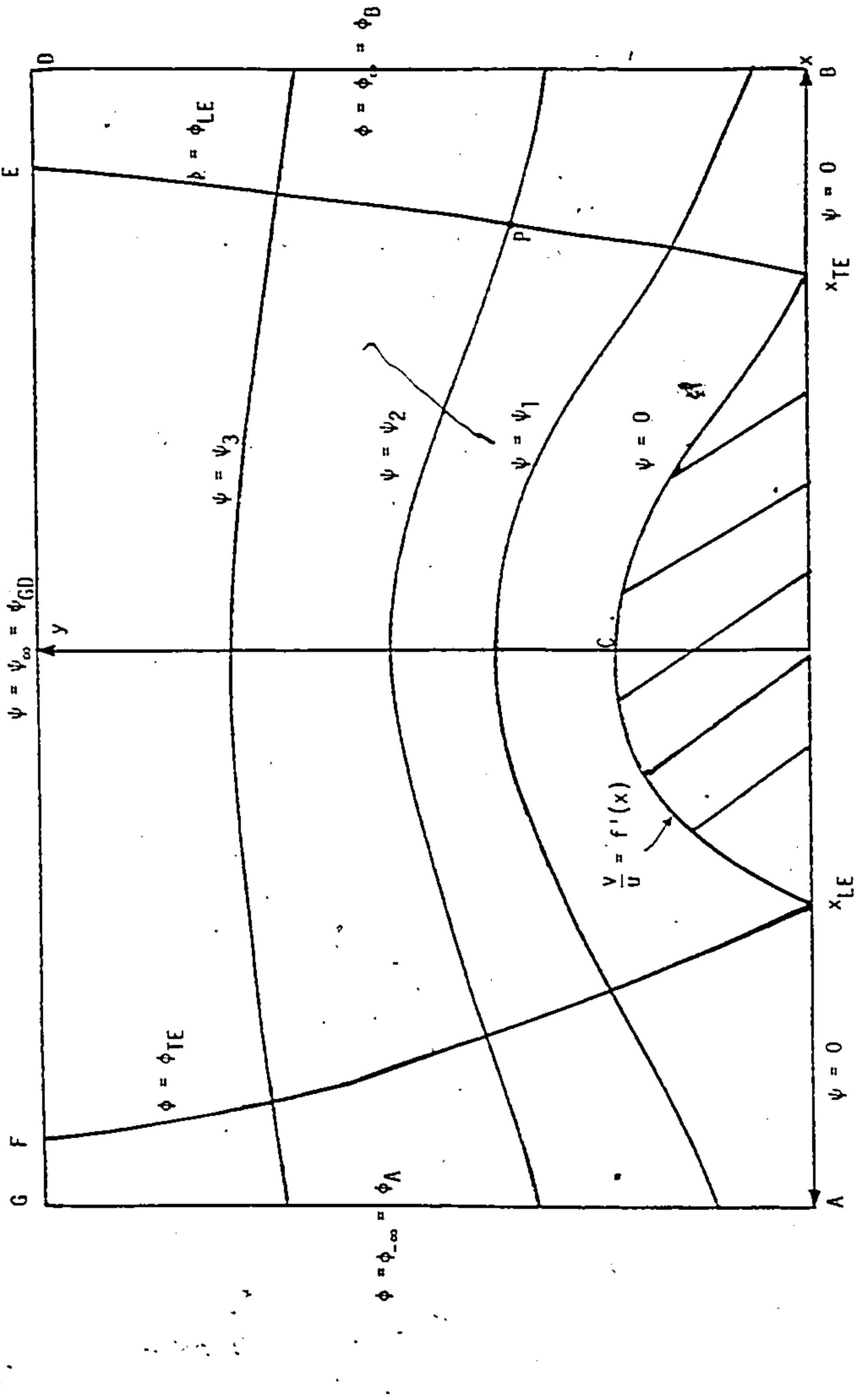

Fig. 1 Physical (x,y) Plane

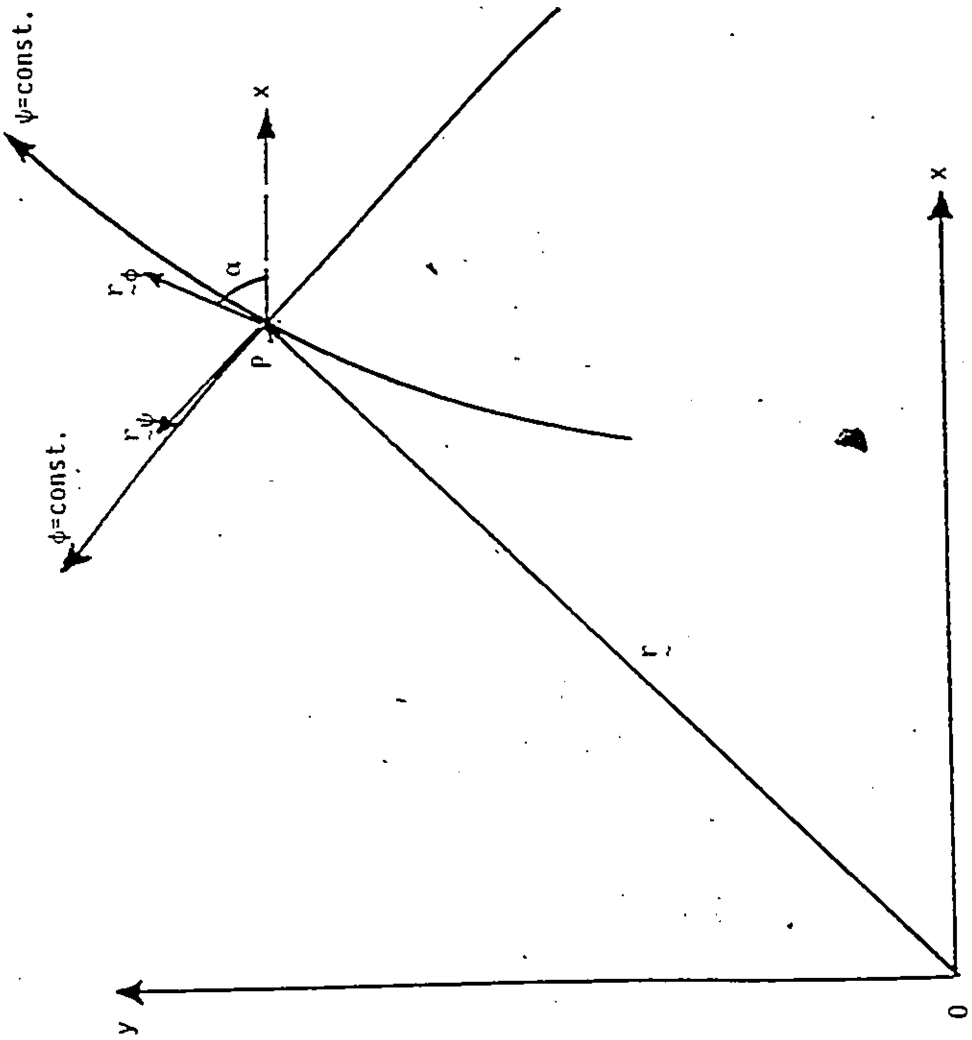

Fig.2 (φ,ψ) Coordinate System

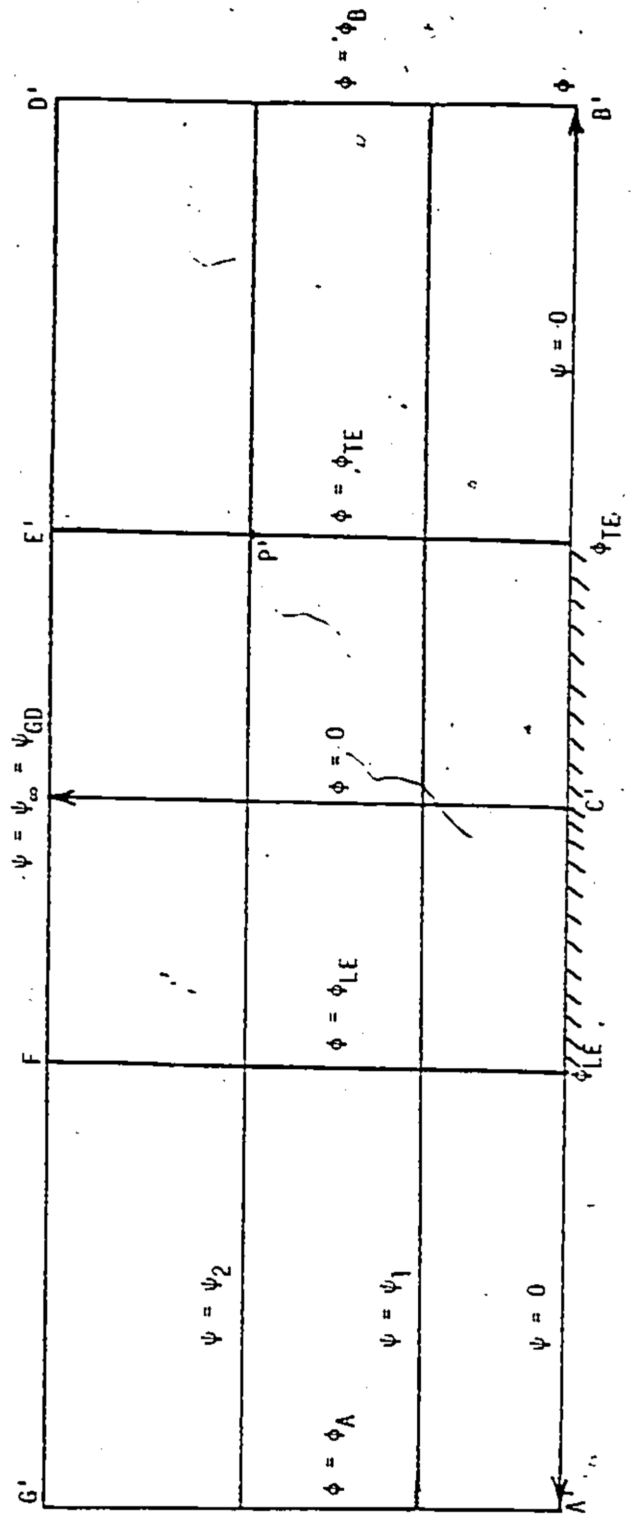

Fig. 3 Computational (ϕ, ψ) Plane

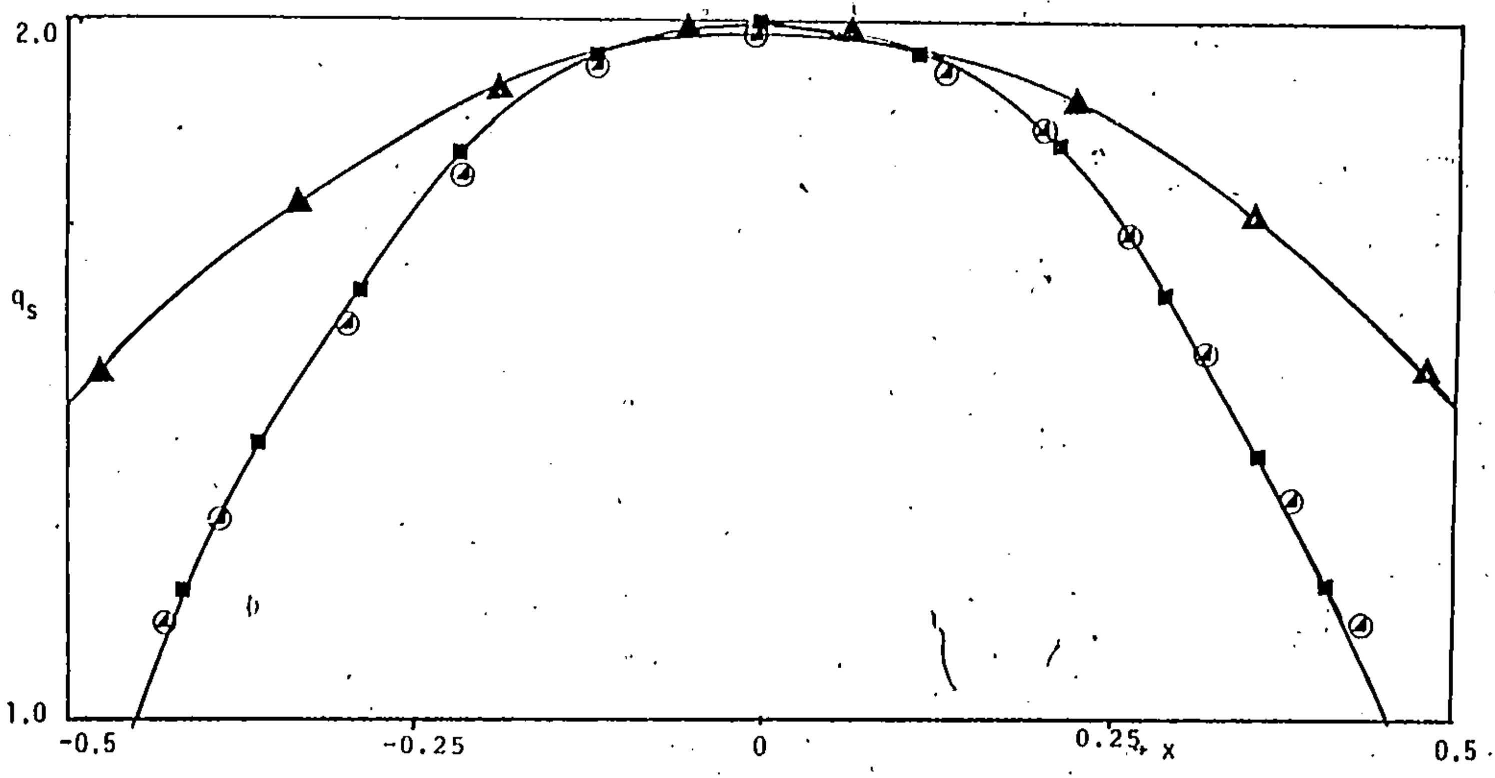

Fig. 4

Surface Speed on Circle; $y(x)=\sqrt{.25-x^2}$

○ ○ :Exact Solution

▲—▲ : Present Solution with Boundary Condition (3.3.1)

■—■ : Present Solution with Boundary Condition (3.2.4a)

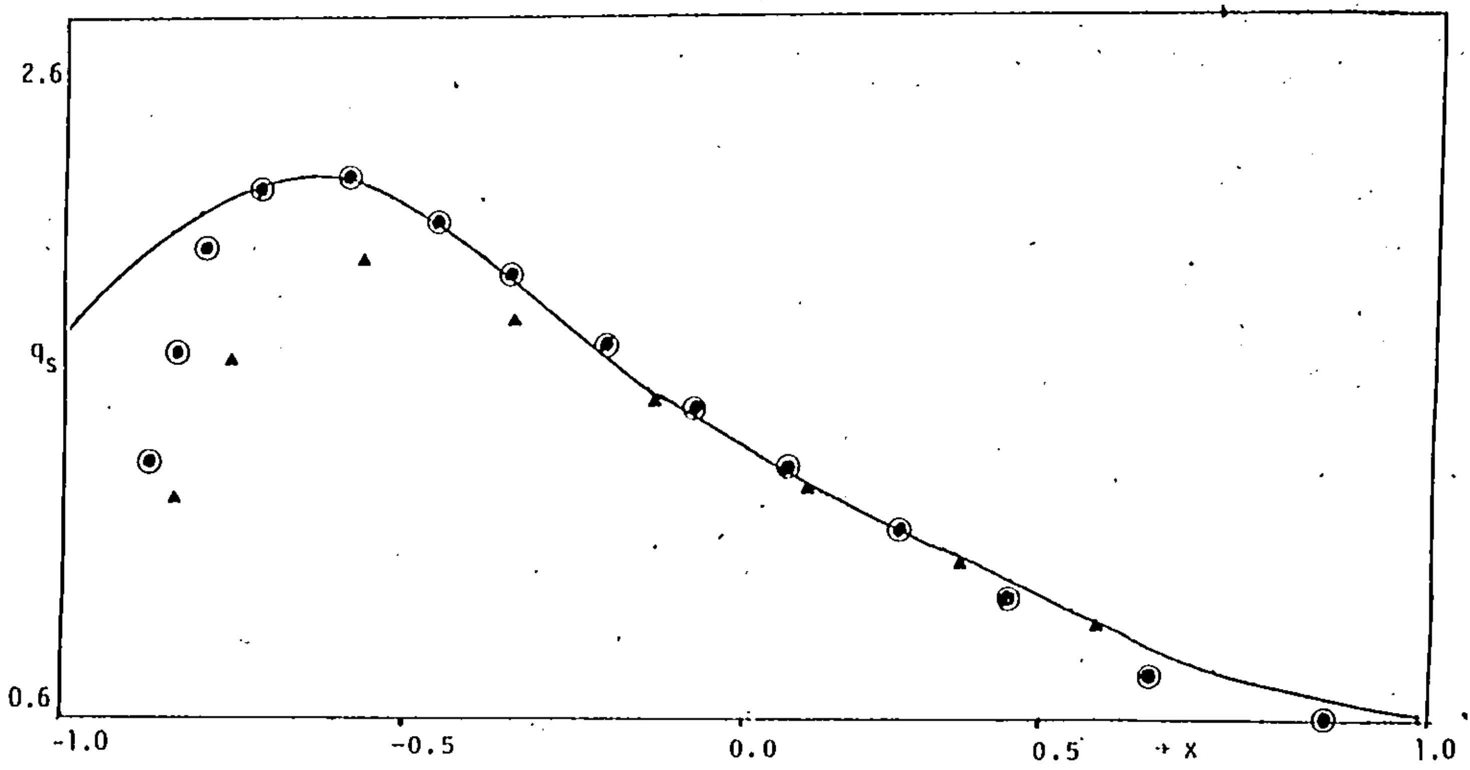

Fig. 5 . Surface Speed on Joukowski Airfoil : $y(x)=(1-x)\sqrt{1-x^2}$

⊙ ⊙ :Van Dyke's Perturbation Solution

——— :Present Solution with Boundary Condition (3.3.1)

▲ ▲ :Present Solution with Boundary Condition (3.2.4a)

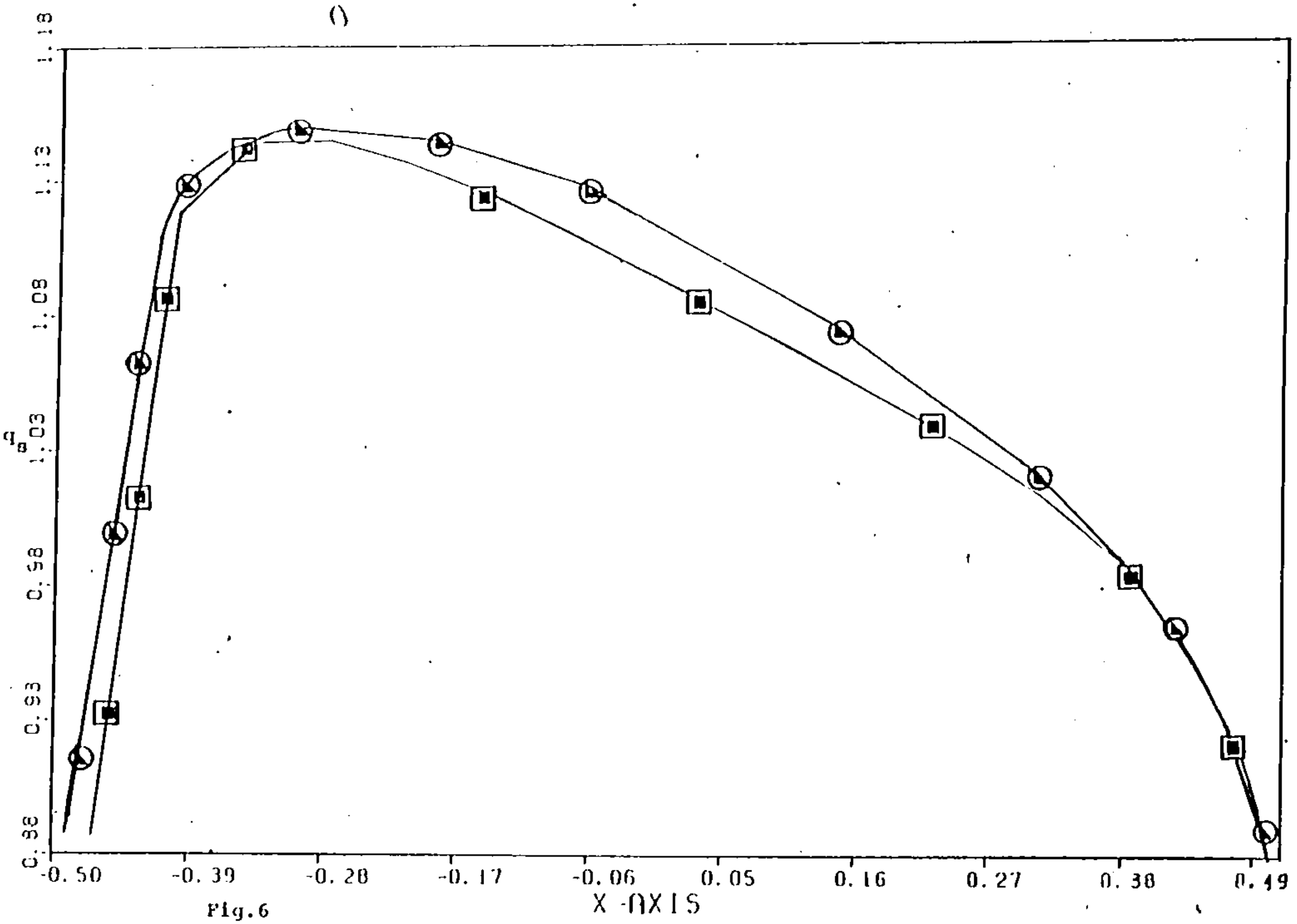

Fig.6

Surface Speed on NACA-0012-64 Airfoil

○—○ : Theodorsen's Solution (2nd order)

■—■ : Present Solution with Boundary Condition (3.2.4a)

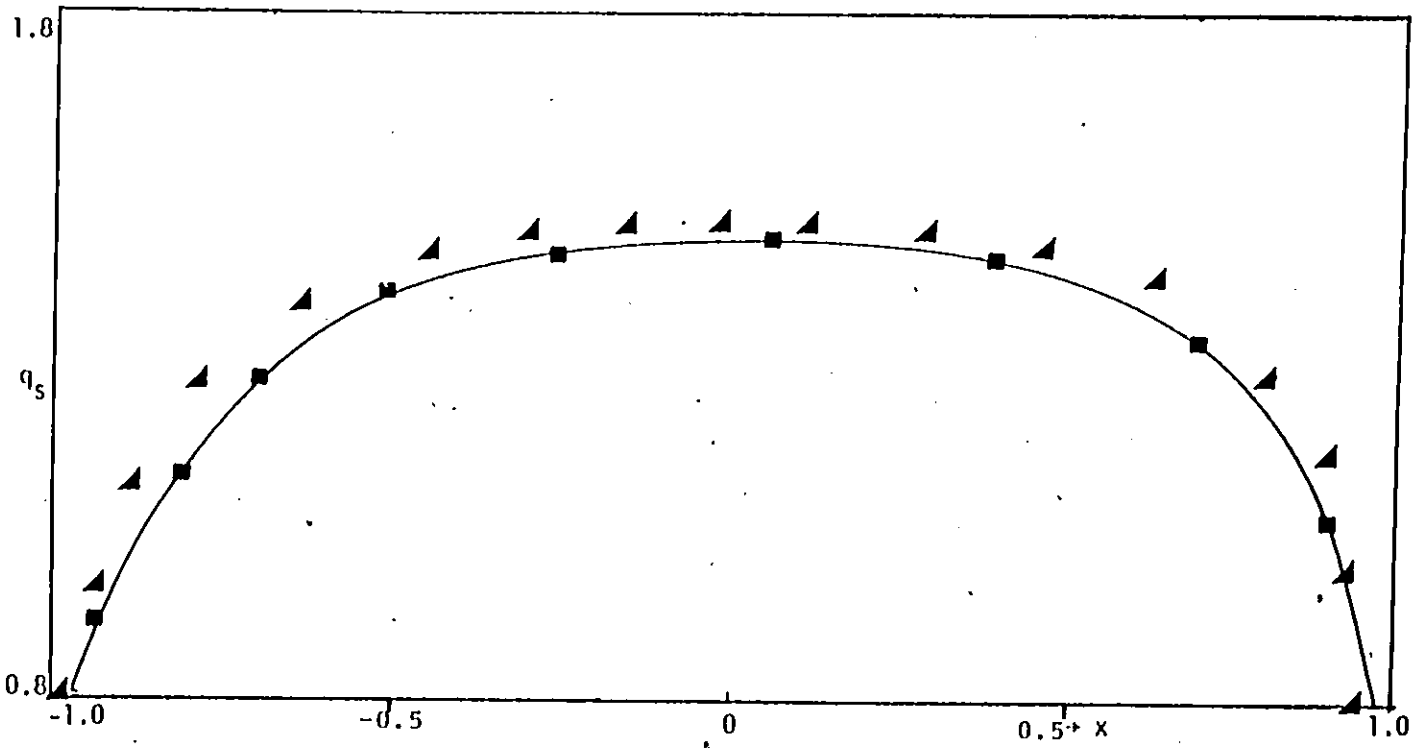

Fig.7. Surface Speed on Ellipse :$y(x)=.5\sqrt{1-x^2}$

◢ ◢ : Exact Speed

—■——■—.: Present Method with Boundary Condition (3.2.4a)

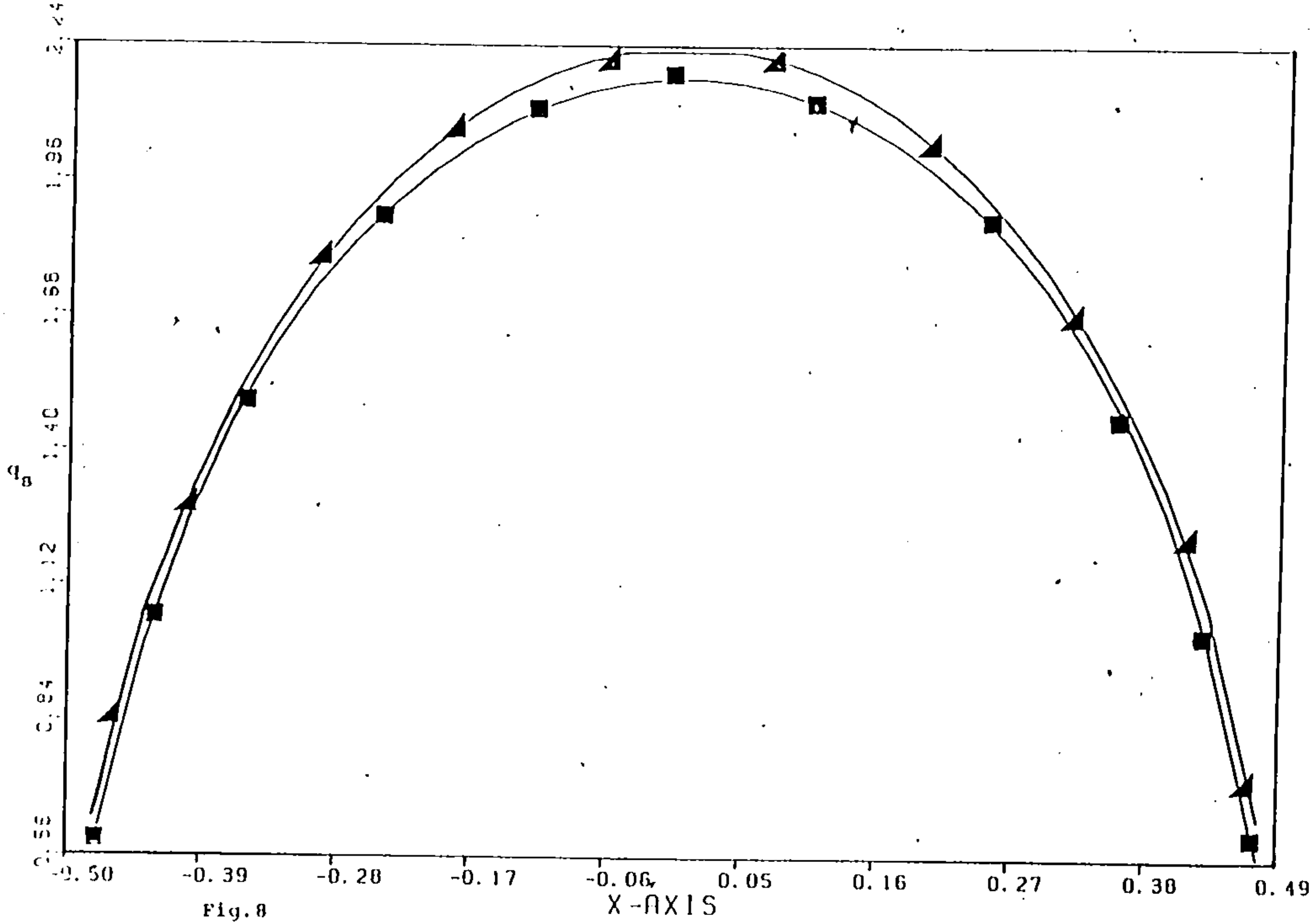

Fig.8

Surface Speed on Circle :$y(x)=\sqrt{.25-x^2}$ for Parabolic Shear Flow

▲——▲: Van Dyke's Perturbation Solution (2nd order)

■——■: Present Method with Boundary Condition (4.3.9a)

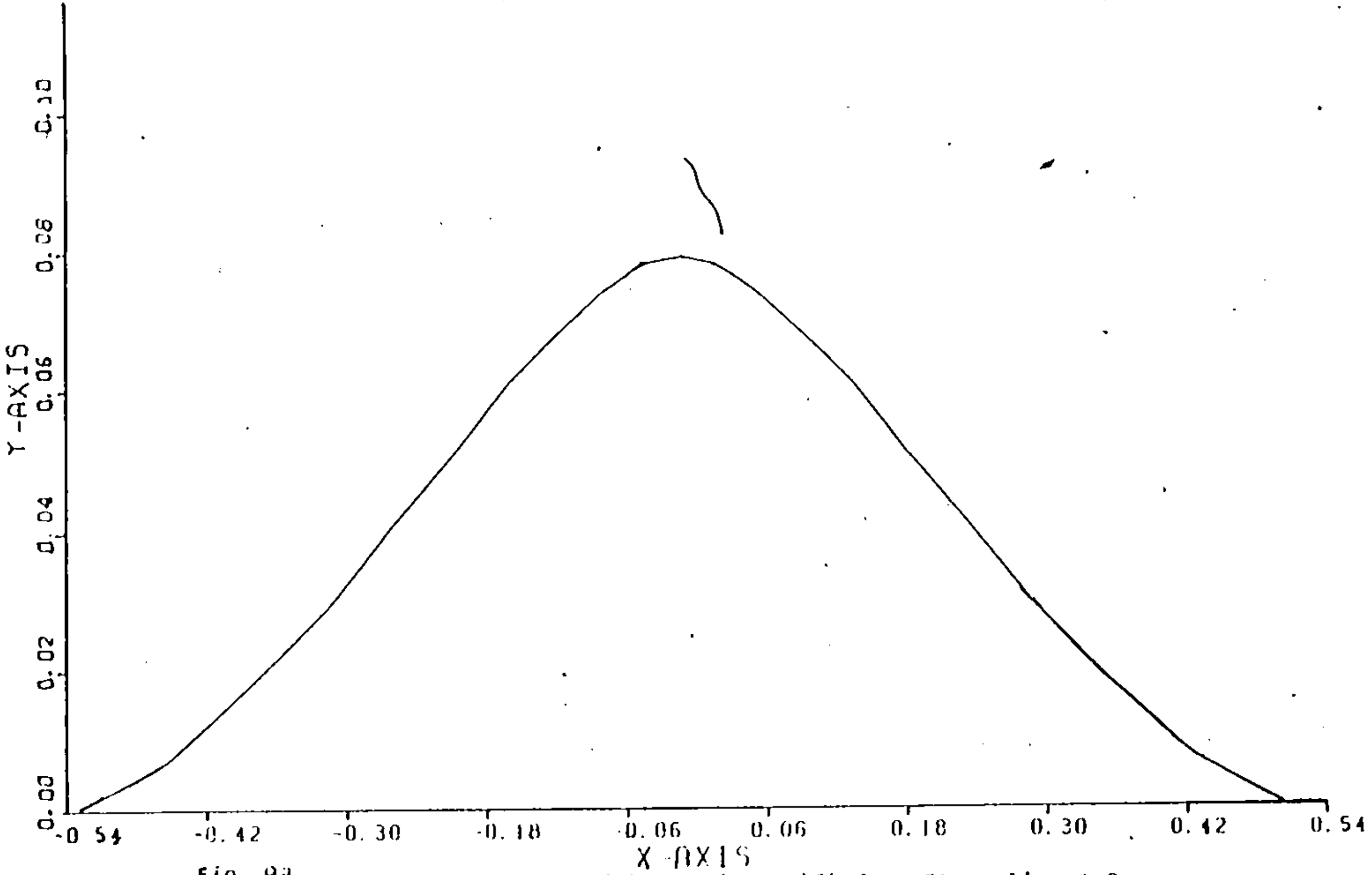

Fig. 9a Inverse Problem with Speed Specified on Streamline $\psi=0$

$K=1$, $\sin\theta=1/K$

y-axis represents Airfoil Profile.

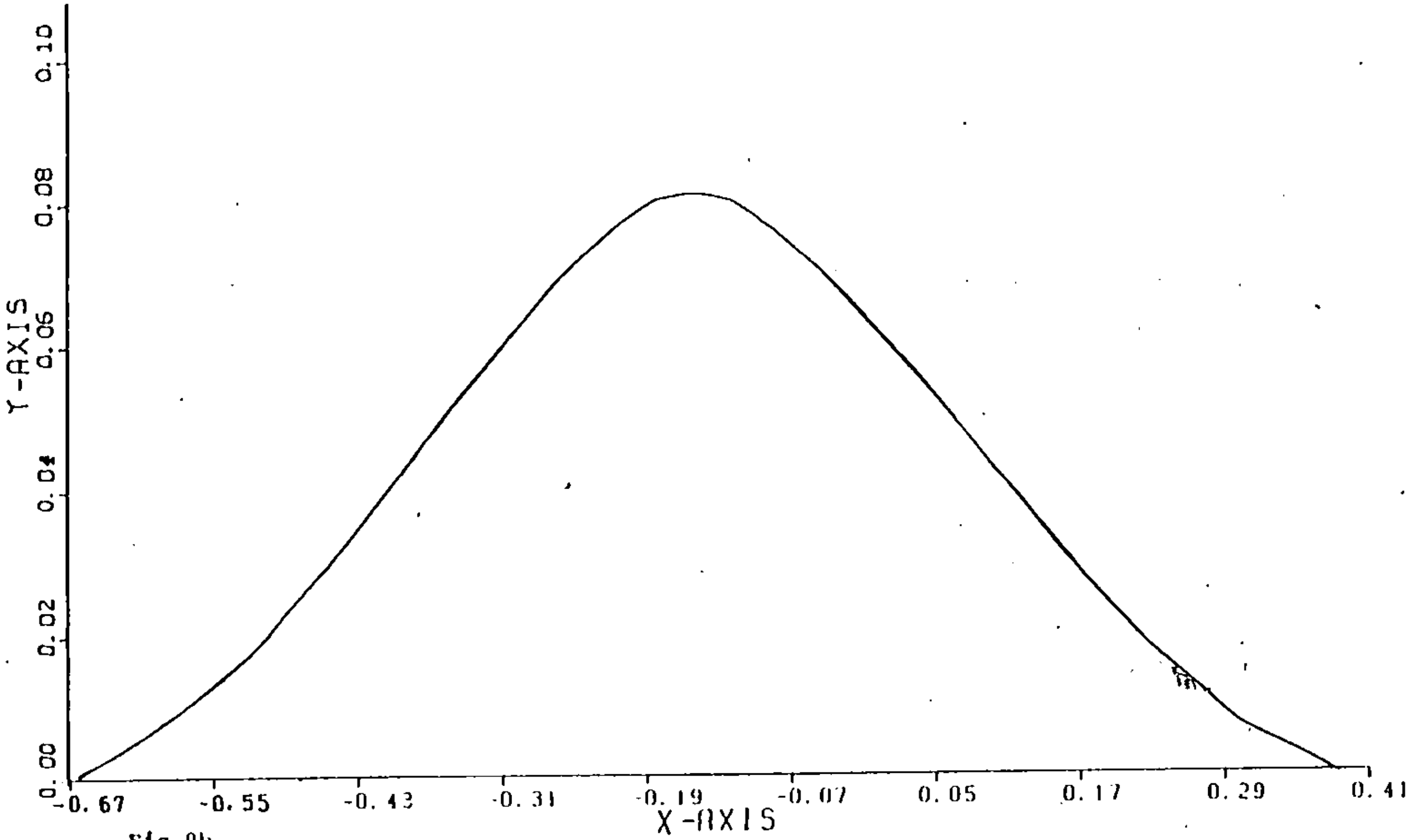

Fig.9b

Inverse Problem with Speed Specified on Streamline $\psi=0$

$K=\sqrt{1.04}$, $\sin\theta = 1/K$

y-axis represents Airfoil Profile.

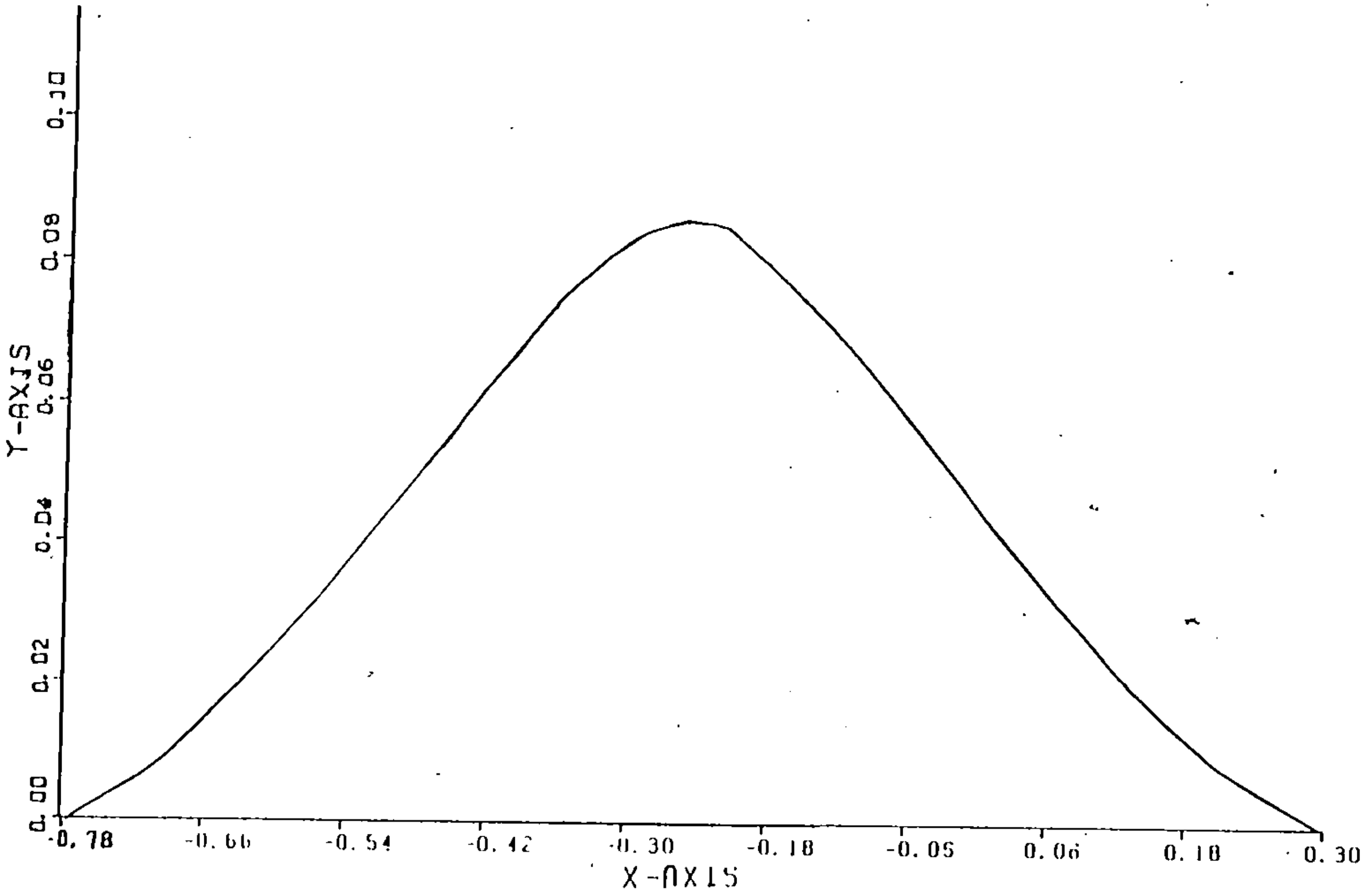

Fig.9c Inverse Problem with Speed Specified on Streamline $\psi=0$

K=√1.16 , sin0=1/K

y-axis represents Airfoil Profile.

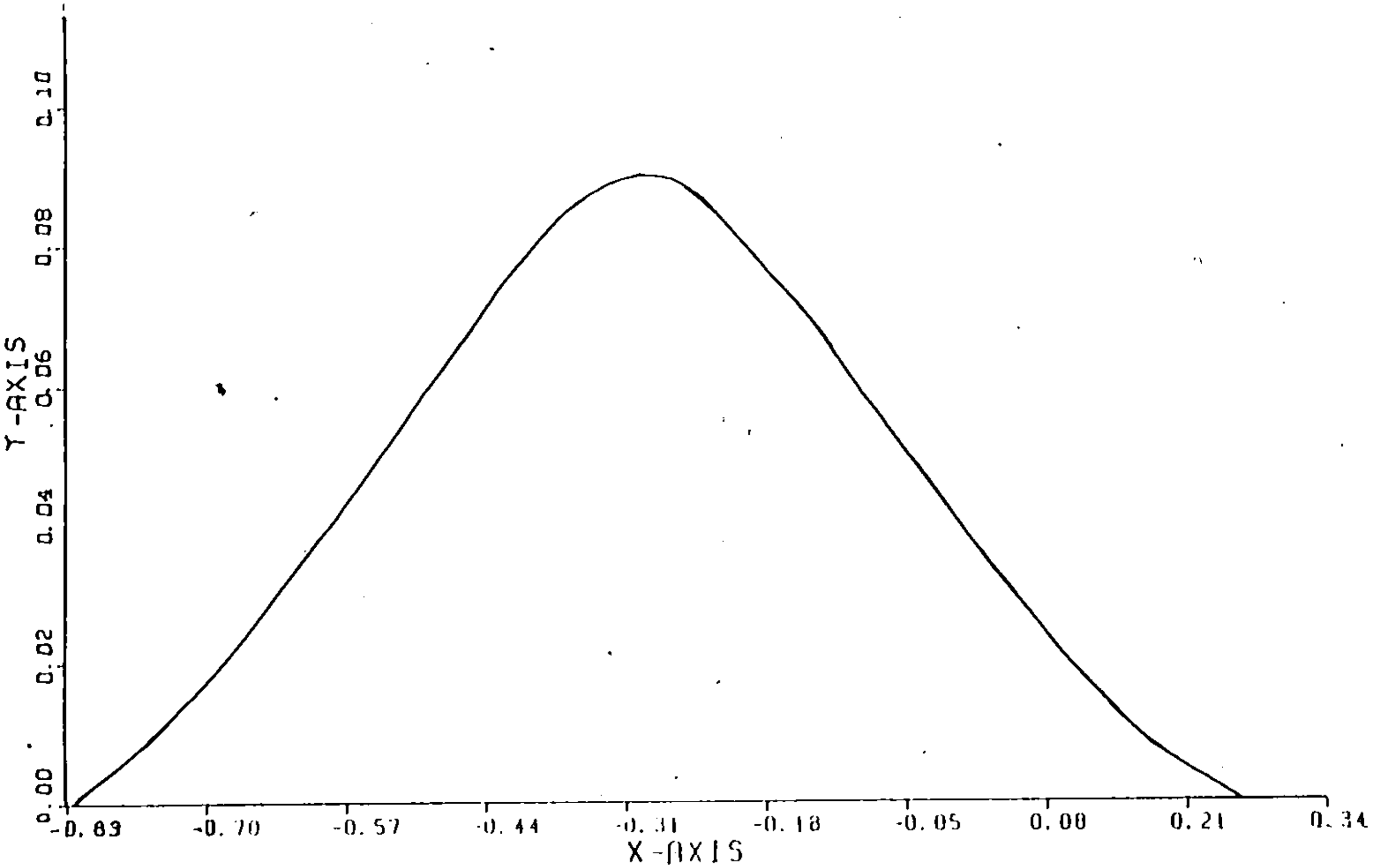

Fig.9d

Inverse Problem with Speed Specified on Streamline $\psi=0$

$K=\sqrt{1.36}$, $\sin\theta=1/K$

y-axis represents Airfoil Profile .

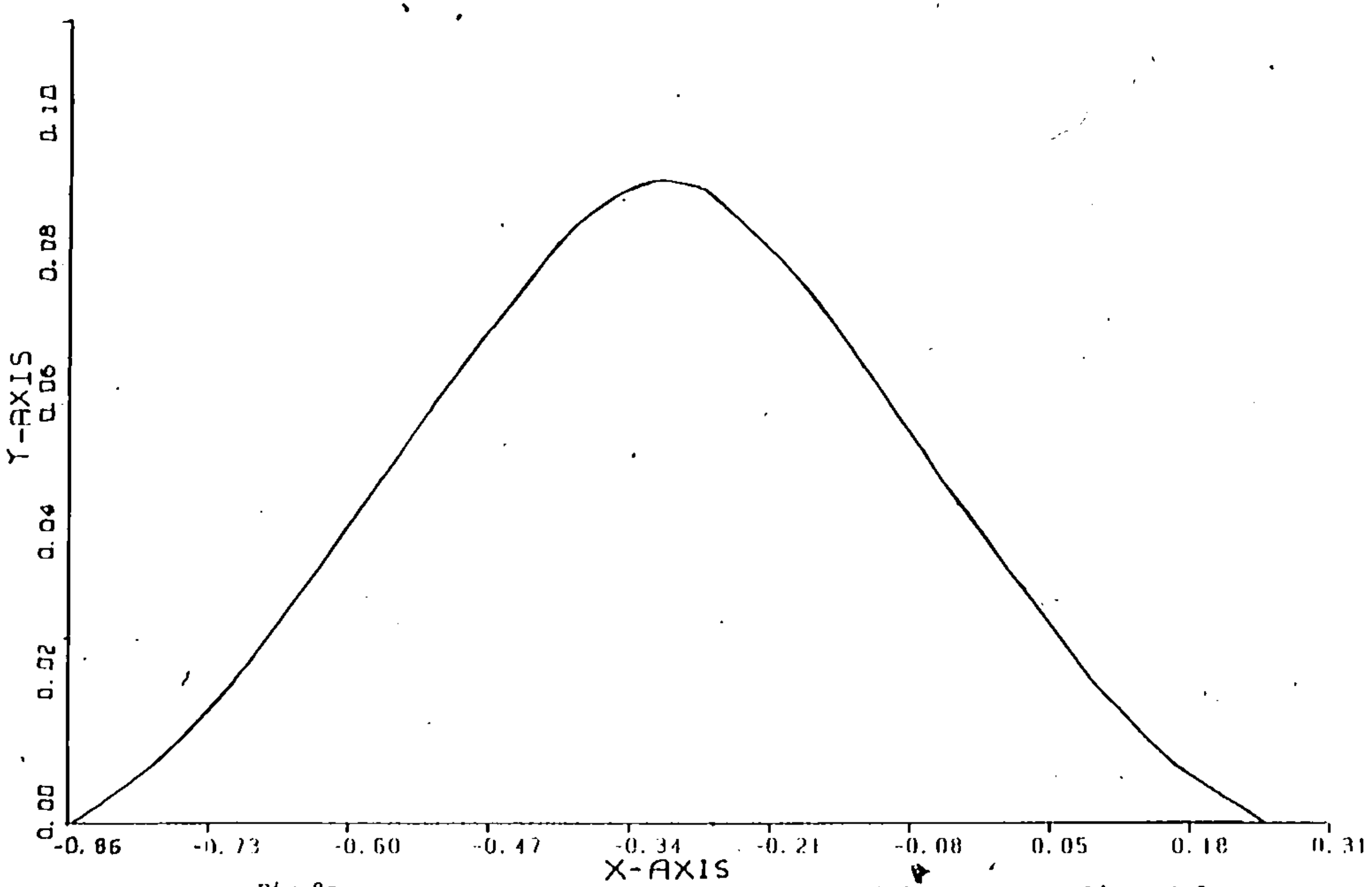

Fig.9e

Inverse Problem with Speed Specified on Streamline $\psi=0$

$K=\sqrt{1.64}$, $\sin\theta=1/K$

y-axis represents Airfoil Profile

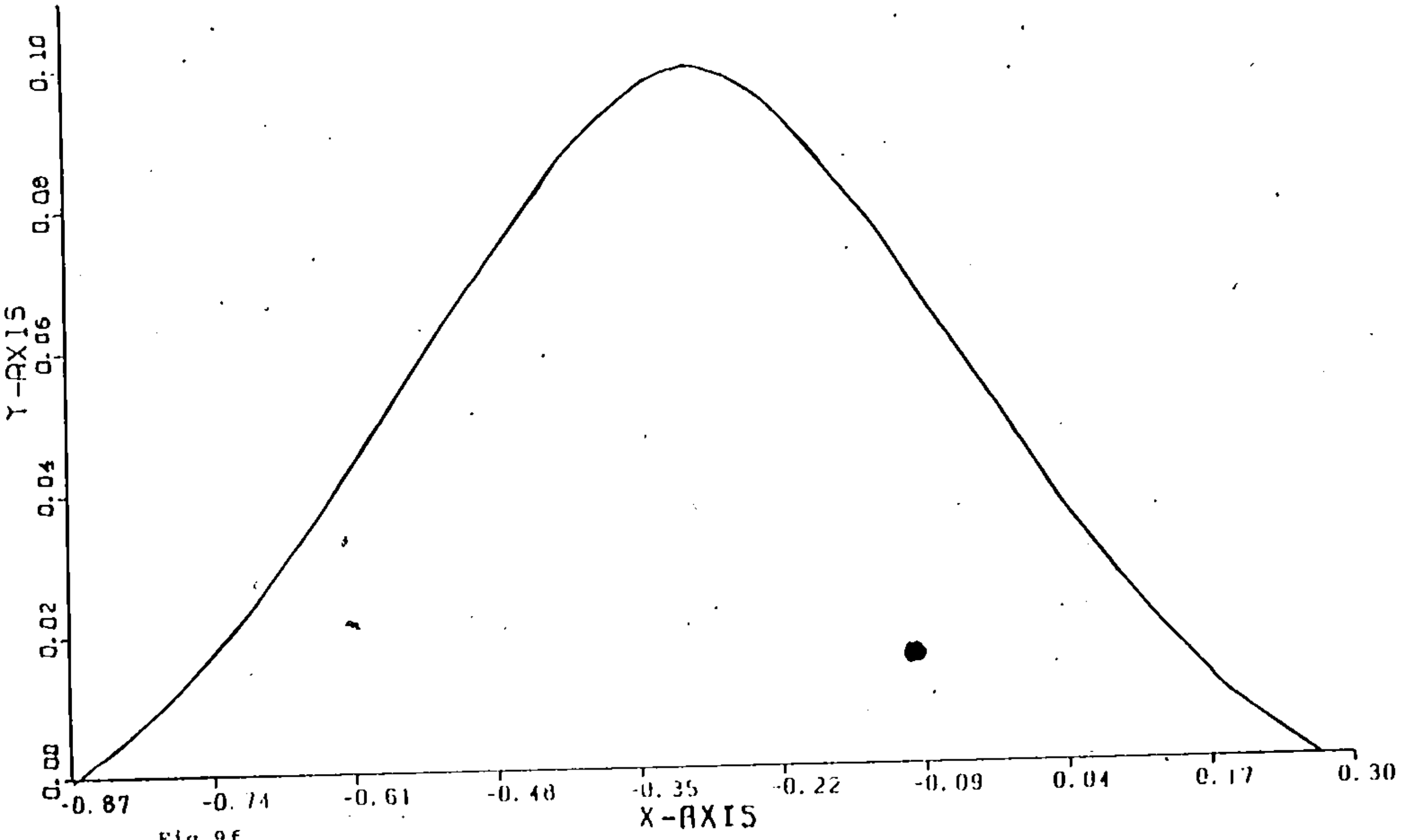

Fig.9f

Inverse Problem with Speed Specified on Streamline $\psi=0$

$K=\sqrt{1.81}$, $\sin\theta=1/K$

y-axis represents Airfoil Profile.

APPENDIX A

```
C|-----------------------------------------------------------------------|
C|                                                                       |
C|             IRROTATIONAL,INVISCID,INCOMPRESSIBLE FLOW PROBLEM         |
C|             ---------------------------------------------------       |
C| SLOR ALTERNATING-DIRECTION IMPLICIT PROGRAM FOR IRROTATIONAL,         |
C| INVISCID,INCOMPRESSIBLE FLOW OVER NACA-0012-64 AIRFOIL                |
C|                                                                       |
C|                        BY GEORGE GROSSMAN                             |
C|                           UNIVERSITY OF WINDSOR                       |
C|                                                                       |
C|-----------------------------------------------------------------------|
C
C REPRESENTATION OF VARIABLES: ET,T-SQRD ELT OF ARC LENGTH.
C X,Y-CARTESIAN COORDINATES.AA,BB,CC-ELTS OF TRIDIAGONAL MATRIX.
C (BX(J),WX(K))-GRID COORDINATES IN TRANSFORMED PLANE.
C WP(J)-ANGLE OF INCLINATION OF AERO-FOIL SURFACE.
C WM- ACCELERATION PARAMETER.EE,AL,AM,SR-PARAMETERS ASSOCIATED
C WITH WM.
C
     BLOCK DATA
     COMMON /STAT/ JM,KM,J1,K1,PM,PS,WW,WS
     DATA JM/85/,KM/43/,J1/84/,K1/42/,PM/1.5/,PS/-1.5/,WW/1.5/,
   $ WS/0./
     COMMON /TITAN/X,XX,Y,TT/GORGO/WX,BX
   $ /SPAR/T,OT,ET,ETT/TROY/AL,AM,SR,XXX
   $ /SANDAL/AA,BB,CC,DD,RHS/ARROW/XRX,WP,EE,WM
     DIMENSION X(85,43),XX(85,43),Y(85),TT(85,43),
   & WX(85),BX(85),T(85,43),OT(85,43),ET(85,43),ETT(85,43),
   & AL(200),AM(200),SR(200),XXX(85,43),AA(85),
   & BB(85),CC(85),DD(85),RHS(85),XRX(85,43),WP(85),
   & EE(200),WM(200)
     END
C
C MAIN PROGRAM
C
     CALL SOLV
     CALL PRIN
     STOP
     END
C
C  CALCULATIONS:BOUNDARY AND INTERIOR CONDITIONS,MATRIX ITERATIONS
C ,OPTIMAL ACCELERATION PARAMETER SEARCH.
C
     SUBROUTINE SOLV
     COMMON /STAT/ JM,KM,J1,K1,PM,PS,WW,WS
     COMMON /TITAN/X,XX,Y,TT/GORGO/WX,BX
   $ /SPAR/T,OT,ET,ETT/TROY/AL,AM,SR,XXX
   $ /SANDAL/AA,BB,CC,DD,RHS/ARROW/XRX,WP,EE,WM
     DIMENSION X(85,43),XX(85,43),Y(85),TT(85,43),
   & WX(85),BX(85),T(85,43),OT(85,43),ET(85,43),ETT(85,43),
   & AL(200),AM(200),SR(200),XXX(85,43),AA(85),
   & BB(85),CC(85),DD(85),RHS(85),XRX(85,43),WP(85),
   $ EE(200),WM(200)
C
C GRID SPACING
C
     DP=(PM-PS)/J1
     DW=(WW-WS)/K1
C
C BOUNDARY CONDITIONS,GRID SPECIFICATION,INITIAL VALUES IN INTERIOR.
```

```
C
              DO 10 J=1,JM
              DO 20 K=1,KM
                    X(J,K)=0.
                    XRX(J,K)=0.
                    T(J,K)=0.
                    ET(J,K)=0.
                    TT(J,K)=0.
  20          CONTINUE
  10          CONTINUE
                    C1=5.*SQRT(.01582/1.1019)
              DO 30 J=1,JM
                    Y(J)=0.
                    BX(J)=(J-1)*DP-1.5
                    WP(J)=0.
                    X(J,KM)=BX(J)
  30          CONTINUE
                    WM(1)=1.00
                    WM(2)=1.00
                    SUM=0.
              DO 40 K=1,KM
                    WX(K)=(K-1)*DW
                    X(1,K)=-1.5
                    X(JM,K)=1.5
  40          CONTINUE
C
C MATRIX FORMS;SOLUTION BY MATRIX ITERATION;ALTERNATING SLOR
C
                    MM=0
   45               MM=MM+1
              IF(MM-200) 50,50,60
  60          WRITE(6,70)
  70          FORMAT(' ',2X,'NO OF IT EX 100')
              GO TO 75
  50          DO 80 J=2,J1
              DO 90 K=2,K1
                    AA(K)=1.
                    BB(K)=-4.
                    CC(K)=1.
                    RHS(K)=-(T(J-1,K)+T(J+1,K))
  90         CONTINUE
                    K=1
                    AA(K)=0.
                    BB(K)=-4.
                    CC(K)=2.
                    RHS(K)=-(T(J-1,K)+T(J+1,K))
               IF(ABS(X(J,K))  .LT.  .5)
          &         RHS(K)=-2.*DW*EXP(T(J,K))*C1*(-.074225/(X(J,1)+.5)**1.5-
          &      .7032+1.7053*(X(J,1)+.5)-1.213*(X(J,1)+.5)**2)
          &      /(1.+C1**2*(.14845/
          & (X(J,1)+.5)**.5-.126-.7032*(X(J,1)+.5)+.8529*(X(J,1)+.5)**2
          &      -.406*(X(J,1)+.5)**3)**2)**1.5+RHS(K)
               CALL TRID(1,42)
               DO 100 K=1,K1
                    OT(J,K)=T(J,K)
  100               T(J,K)=T(J,K)+WM(MM)*(RHS(K)-T(J,K))
  80            CONTINUE
               DO 110 K=1,K1
               DO 120 J=2,J1
                    AA(J)=1.
```

```
                   BB(J)=-4.
                   CC(J)=1.
            IF(K .GT. 1) RHS(J) =-(T(J,K+1)+T(J,K-1))
            IF(K .EQ. 1) RHS(J) =-2.*T(J,2)
            IF( K .EQ. 1 .AND. ABS(X(J,1)) .LT. .5)
     &         RHS(J)=-2.*DW*EXP(T(J,K))*C1*(-.074225/(X(J,1)+.5)**1.5-
     &       .7032+1.7058*(X(J,1)+.5)-1.218*(X(J,1)+.5)**2)
     &       /(1.+C1**2*(.14845/
     & (X(J,1)+.5)**.5-.126-.7032*(X(J,1)+.5)+.8529*(X(J,1)+.5)**2
     &       -.406*(X(J,1)+.5)**3)**2)**1.5-2.*T(J,2)
  120     CONTINUE
            CALL TRID(2,34)
            DO 130 J=2,J1
                   OT(J,K)=T(J,K)
  130              T(J,K)=T(J,K)+WM(MM)*(RHS(J)-T(J,K))
  110       CONTINUE
C CALCULATION OF X THROUGHOUT FLOW FIELD
            DO 140 J=2,J1
            DO 150 K=2,K1
                   AA(K)=1.
                   BB(K)=-4.
                   CC(K)=1.
                   RHS(K)=-(X(J+1,K)+X(J-1,K))
  150       CONTINUE
                   K=1
                   AA(K)=0.
                   BB(K)=-4.
                   CC(K)=2.
                   RHS(K)=-(X(J-1,K)+X(J+1,K))
     &  -2.*DW*SIN(WP(J))*EXP(T(J,1))
                   RHS(K1)=RHS(K1)-CC(K1)*X(J,KM)
            CALL TRID(1,42)
            DO 160 K=1,K1
                   XXX(J,K)=X(J,K)
  160              X(J,K)=X(J,K)+WM(MM)*(RHS(K)-X(J,K))
                   Y(J)=0.
         IF(ABS(X(J,1)) .LT. .5) Y(J)=C1*(.2969*
     &  (X(J,1)+.5)**.5-.126*(X(J,1)+.5)-.3516*(X(J,1)+.5)**2
     &     +.2343*(X(J,1)+.5)**3-.1015*(X(J,1)+.5)**4)
                   WP(J)=0.
             IF(Y(J) .GT. 0.) WP(J)= ATAN(C1*(.14845/SQRT(X(J,1)+.5)
     & -.126-.7032*(X(J,1)+.5)+.8529*(X(J,1)+.5)**2-.406*(X(J,1)+.5)
     &  **3))
  140        CONTINUE
            DO 170 K=1,K1
            DO 180 J=2,J1
                   AA(J)=1.
                   BB(J)=-4.
                   CC(J)=1.
            IF(K .GT. 1) RHS(J) =-(X(J,K+1)+X(J,K-1))
          IF(K .EQ. 1)
     &      RHS(J)=-2.*X(J,2)-SIN(WP(J))*EXP(T(J,1))*2.*DW
  180     CONTINUE
                   RHS(J1)=RHS(J1)-CC(J1)*X(JM,K)
                   RHS(2)=RHS(2)-AA(2)*X(1,K)
            CALL TRID(2,34)
            DO 200 J=2,J1
                   XXX(J,K)=X(J,K)
  200              X(J,K)=X(J,K)+WM(MM)*(RHS(J)-X(J,K))
  170       CONTINUE
```

```
C DETERMINATION OF Y AND ALPHA ALONG SURFACE OF AIRFOIL
            DO 210 J=2,J1
                 Y(J)=0.
         IF(ABS(X(J,1)) .LT. .5) Y(J)=C1*(.2969*
      &   (X(J,1)+.5)**.5-.126*(X(J,1)+.5)-.3516*(X(J,1)+.5)**2
      &   +.2843*(X(J,1)+.5)**3-.1015*(X(J,1)+.5)**4)
                 WP(J)=0.
            IF(Y(J) .GT. 0.) WP(J)= ATAN(C1*(.14845/SQRT(X(J,1)+.5)
      & -.126-.7032*(X(J,1)+.5)+.8529*(X(J,1)+.5)**2-.406*(X(J,1)+.5)*
      &   *3))
  210       CONTINUE
C
C  FINDING OF OPTIMAL ITERATION PARAMETER
C
                 SUM=0.
           DO 220 K=1,K1
           DO 220 J=2,J1
                 XRX(J,K)=ABS(T(J,K)-OT(J,K))
                 SUM=SUM+XRX(J,K)
  220      CONTINUE
                 SR(MM)=SUM
           IF(MM .GT. 1) AL(MM)=SR(MM)/SR(MM-1)
           IF(MM .GT. 10) T1=ABS(WM(MM)-
      &     WM(MM-10))/(2.-WM(MM))
           IF(MM .GT. 10 .AND. T1 .LT. .05)
      &     WM(MM+1)=WM(MM)
           IF(MM .GT. 10 .AND. T1 .LT. .05)
      &     GO TO 250
           IF(MOD(MM,10) .NE. 0) GO TO 250
           EE(MM)=2./(1.+SQRT(ABS((1.-(AL(MM)+WM(MM)-1.)**2/(WM(MM)**2
      &      *AL(MM))))))
             WM(MM+1)=EE(MM)-(2.-EE(MM))/4.
           WRITE(6,7000)WM(MM+1),T1,T2,AL(MM),SR(MM)
  7000     FORMAT(' ',6(F8.5))
  250      CONTINUE
           IF(MOD(MM,10) .NE. 0 .AND. MM .GE. 10)WM(MM+1)=WM(MM)
           IF(MM .LE. 9)WM(MM+1)=1.00
           IF(MM .GE. 10 .AND. MM .LE. 19) WM(MM)=1.35
C TOLERANCE LEVEL CHECK
           DO 270 J=2,J1
           DO 270 K=1,K1
                 T2=ABS(T(J,K)-OT(J,K))
           IF(T2 .GT. .0005) GO TO 45
  270      CONTINUE
           WRITE(6,7100)MM
  7100     FORMAT(' ','NO OF IT=',I3)
           WRITE(6,7200)WM(MM)
  7200     FORMAT(' ','OPTIMAL ITERATION PARAMETER *1 =',F8.5)
           DO 280 J=1,JM
           DO 280 K=1,K1
                 T(J,K)=EXP(T(J,K))
                 XXX(J,K)=0.
  280      CONTINUE
                 WM(1)=1.
                 WM(2)=1.
                 T2=0.
                 T1=0.
C DETERMINING SECOND SET OF VALUES FOR E(=EXP(T))
                 ML=0
  310            ML=ML+1
```

```
               IF(ML-100) 320,320,330
      330      WRITE(6,340)
      340      FORMAT(' ',2X,'NO OF IT EX 100')
               GO TO 350
      320      DO 360 J=2,J1
               DO 370 K=2,K1
                      AA(K)=1.
                      BB(K)=-4.
                      CC(K)=1.
                      RHS(K)=-(ET(J-1,K)+ET(J+1,K))
      370       CONTINUE
                      K=1
                      AA(K)=0.
                      BB(K)=-4.
                      CC(K)=2.
                      RHS(K)=-(ET(J-1,K)+ET(J+1,K))
                      RHS(K)=-(WP(J+1)-WP(J-1))+RHS(K)
               CALL TRID(1,42)
               DO 380 K=1,K1
                      ETT(J,K)=ET(J,K)
      380             ET(J,K)=ET(J,K)+WM(ML)*(RHS(K)-ET(J,K))
      360        CONTINUE
               DO 390 K=1,K1
               DO 400 J=2,J1
                      AA(J)=1.
                      BB(J)=-4.
                      CC(J)=1.
              IF(K .GT. 1) RHS(J) =-(ET(J,K+1)+ET(J,K-1))
               IF(K .EQ. 1) RHS(J) =-2.*ET(J,2)
         &     -(WP(J+1)-WP(J-1))
       400     CONTINUE
               CALL TRID(2,34)
               DO 410 J=2,J1
                      ETT(J,K)=ET(J,K)
      410             ET(J,K)=ET(J,K)+WM(ML)*(RHS(J)-ET(J,K))
      390       CONTINUE
   C OPTIMAL ITERATION PARAMETER
                      SUM=0.
               DO 450 K=1,K1
               DO 450 J=2,J1
                      ERX(J,K)=ABS(ET(J,K)-ETT(J,K))
                      SUM=SUM+ERX(J,K)
      450      CONTINUE
                      SR(ML)=SUM
               IF(ML .GT. 1) AL(ML)=SR(ML)/SR(ML-1)
               IF(ML .GT. 10) T1=ABS(WM(ML)-
         &      WM(ML-10))/(2.-WM(ML))
               IF(ML .GT. 10 .AND. T1 .LT. .05)
         &      WM(ML+1)=WM(ML)
               IF(ML .GT. 10 .AND. T1 .LT. .05)
         &      GO TO 460
               IF(MOD(ML,10) .NE. 0) GO TO 460
               RR(ML)=2./(1.+SQRT(ABS((1.-(AL(ML)+WM(ML)-1.)**2/(WM(ML)**2
         &        *AL(ML))))))
                         WM(ML+1)=RR(ML)-(2.-RR(ML))/4.
               WRITE(6,7300) WM(ML+1),T1,AL(ML),T2,SR(ML)
      7300     FORMAT(' ',6(F8.5))
      460      CONTINUE
               IF(MOD(ML,10) .NE. 0 .AND. ML .GE. 10) WM(ML+1)=WM(ML)
               IF(ML .LE. 9) WM(ML+1)=1.00
```

```
            IF(WM(ML) .GT. 1.95) WM(ML+1)=WM(1)
            DO 420 J=2,J1
            DO 420 K=1,K1
                  T2=ABS(ET(J,K)-ETT(J,K))
                  IF(T2 .GT. .0005) GO TO 310
  420       CONTINUE
            WRITE(6,430) ML
  430       FORMAT(' ','NO OF IT=',I3)
            WRITE(6,7700) WM(ML)
  7700      FORMAT(' ',' OPTIMAL ITERATION PARAMETER #2= ',F8.5)
            DO 800 JK=2,J1
                  JKK=JM-JK+1
            IF(X(JK,1) .LE. .5) JJ=JK
  800       IF(X(JKK,1) .GT. -.5) KK=JKK
            WRITE(6,805) KK,JJ
  805       FORMAT(' ','KK=',1X,I2,10X,'JJ=',1X,I2)
            DO 440 J=1,JM
            DO 440 K=1,K1
                  ET(J,K)=EXP(ET(J,K))
  440.      CONTINUE
  350       CONTINUE
  75        CONTINUE
            RETURN
            END
C TRIDIAGONAL SOLVER
            SUBROUTINE TRID(NL,NU)
            COMMON/SANDAL/A,B,C,D,G
            DIMENSION A(85),B(85),C(85),D(85),G(85)
                  D(NL)=C(NL)/B(NL)
                  G(NL)=G(NL)/B(NL)
                  NLP1=NL+1
            DO 290 N=NLP1,NU
                  Z=1./(B(N)-A(N)*D(N-1))
                  D(N)=C(N)*Z
  290             G(N)=(G(N)-A(N)*G(N-1))*Z
                  NUPNL=NU+NL
            DO 300 NN=NLP1,NU
                  N=NUPNL-NN
  300            G(N)=G(N)-D(N)*G(N+1)
            RETURN
            END
C PRINTING OF RESULTS:X,E,ET,TT,WM...
            SUBROUTINE PRIN
            COMMON /STAT/ JM,KM,J1,K1,PM,PS,AW,WS
            COMMON /TITAN/X,XX,Y,TT/GORGO/WX,BX
       &    /SPAR/T,OT,ET,ETT/TPOY/AL,AM,SR,XXI
       &    /SANDAL/AA,BB,CC,DD,RHS/ARROW/XRX,WP,RR,WM
            DIMENSION X(85,43),XX(85,43),Y(85),TT(85,43),
       &    WX(85),BX(85),T(85,43),OT(85,43),ET(85,43),ETT(85,43),
       &    AL(200),AM(200),SR(200),XXX(85,43),AA(85),
       &    BB(85),CC(85),DD(85),RHS(85),XRX(85,43),WP(85),
       &    RR(200),WM(200)
            PRINT 1500,(K,K=1,8)
  1500      FORMAT(//8X,'K',3X,8(5X,I2,2X))
            WRITE(6,1600) (WX(K),K=1,8)
  1600      FORMAT(' ',6X,'WX(K)',8(1X,D13.6))
            WRITE(6,1700)
  1700      FORMAT(//,'J',2X,'BX(J)',4X,'ET(J,K)')
            DO 1800 J=1,JM
            WRITE(6,1850) J,BX(J),(ET(J,K),K=1,8)
```

```
 1800 CONTINUE
 1850 FORMAT(' ',I2,1X,D13.6,8(1X,D13.6))
      PRINT 2002,(K,K=1,8)
 2002 FORMAT(//9X,'K',3X,8(5X,I2,2X))
      WRITE(6,2102) (WX(K),K=1,8)
 2102 FORMAT(' ',6X,'WX(K)',8(1X,D13.6))
      WRITE(6,2202)
 2202 FORMAT(//,'J',2X,'BX(J)',4X,'T(J,K)')
      DO 2302 J=1,JM
      WRITE(6,2402) J,BX(J),(T(J,K),K=1,8)
 2302 CONTINUE
 2402 FORMAT(' ',I2,1X,D13.6,8(1X,D13.6))
      PRINT 2004,(K,K=1,8)
 2004 FORMAT(//9X,'K',3X,8(5X,I2,2X))
      WRITE(6,2104) (WX(K),K=1,8)
 2104 FORMAT(' ',6X,'WX(K)',8(1X,D13.6))
      WRITE(6,2204)
 2204 FORMAT(//,'J',2X,'BX(J)',4X,'X(J,K)')
      DO 2304 J=1,JM
      WRITE(6,2404) J,BX(J),(X(J,K),K=1,8)
 2304 CONTINUE
 2404 FORMAT(' ',I2,1X,D13.6,8(1X,D13.6))
      RETURN
      END
/*
//GO.SYSIN DD *
//
```

APPENDIX B

```
C|---------------------------------------------------------------|
C|                                                               |
C|               PARABOLIC SHEAR FLOW-PROBLEM                    |
C|               ----------------------------                    |
C|  ALTERNATING-DIRECTION IMPLICIT(ADI) PROGRAM FOR PARABOLIC    |
C|     SHEAR FLOW OVER CIRCLE OF RADIUS .5                       |
C|                                                               |
C|       BY GEORGE GROSSMAN                                      |
C|       UNIVERSITY OF WINDSOR                                   |
C|---------------------------------------------------------------|
C
C REPRESENTATION OF VARIABLES: TT,XX CORRESPOND WITH COEFFICIENTS
C OF SQUARED ELEMENT OF ARC LENGTH E AND G IN NORMAL NOTATION
C X,Y-CARTESIAN COORDINATES.AA,BB,CC-ELTS OF TRIDIAGONAL MATRIX.
C (BX,WX)-GRID COORDINATES IN TRANSFORMED PLANE.
C WP-ANGLE OF INCLINATION OF AERO-FOIL SURFACE.
C WM- ACCELERATION PARAMETER.EE,AL,AM,SR-PARAMETERS ASSOCIATED
C WITH WM.W2 IS MAGNITUDE OF THE VORTICITY.
C OTHER VARIABLES ARE USED OR UNUSED DUMMY VARIABLES
C
       BLOCK DATA
       INTEGER JM,KM,J1,K1
       REAL*8 PM/1.5D00/,PS/-1.5D00/,WW/2.D00/,WS/0.D00/
       DATA JM/64/,KM/43/,J1/63/,K1/42/
       COMMON /STAT/ JM,KM,J1,K1,PM,PS,WW,WS
       DOUBLE PRECISION X,XX,Y,TT,WX,BX,T,OT,ET,ETT,AL,AM,SR,XXX
    &  ,AA,BB,CC,DD,RHS,WP,EE,WM,SUM1,SUM2,SUM5,W2
       COMMON /TITAN/X,XX,Y,TT/GORGO/WX,BX
    &  /SPAR/T,OT,ET,ETT/TROY/AL,AM,SR,XXX
    &  /SANDAL/AA,BB,CC,DD,RHS/ARROW/WP,EE,WM
    &  /TRUE/SUM1,SUM2,SUM5,W2
       DIMENSION X(64,43),XX(64,43),Y(64),TT(64,43),
    &  WX(64),BX(64),T(64,43),OT(64,43),ET(64,43),ETT(64,43),
    &  AL(200),AM(200),SR(200),XXX(64,43),AA(64),
    &  BB(64),CC(64),DD(64),RHS(64),WP(64),
    &  EE(200),WM(200),SUM1(64),SUM2(64),SUM5(64),W2(43)
       END
C
C MAIN PROGRAM
C
       CALL SOLV
       CALL PRIN
       STOP
       END
C
C  CALCULATIONS:BOUNDARY AND INTERIOR CONDITIONS,MATRIX ITERATIONS
C ,OPTIMAL ACCELERATION PARAMETER SEARCH.
C
       SUBROUTINE SOLV
       REAL*8 DP,DW,ES,C1,T1,T2,T3,T4,T5,T6,T7,SUM,SUM100,SUM200
    &  ,SUM300,PM,PS,WW,WS
       COMMON /STAT/ JM,KM,J1,K1,PM,PS,WW,WS
       DOUBLE PRECISION X,XX,Y,TT,WX,BX,T,OT,ET,ETT,AL,AM,SR,XXX
    &  ,AA,BB,CC,DD,RHS,WP,EE,WM,SUM1,SUM2,SUM5,W2
       COMMON /TITAN/X,XX,Y,TT/GORGO/WX,BX
    &  /SPAR/T,OT,ET,ETT/TROY/AL,AM,SR,XXX
    &  /SANDAL/AA,BB,CC,DD,RHS/ARROW/WP,EE,WM
    &  /TRUE/SUM1,SUM2,SUM5,W2
       DIMENSION X(64,43),XX(64,43),Y(64),TT(64,43),
    &  WX(64),BX(64),T(64,43),OT(64,43),ET(64,43),ETT(64,43),
     &    AL(200),AM(200),SB(200),XXX(64,43),AA(64),
     &    BB(64),CC(64),DD(64),RHS(64),WP(64),
     &    EE(200),WM(200),SUM1(64),SUM2(64),SUM5(64),W2(43)
C
C GRID SPACING DP,DW
C
          DP=(PM-PS)/DFLOAT(J1)
          DW=(WM-WS)/DFLOAT(K1)
C
C BOUNDARY CONDITIONS,GRID SPECIFICATION,INITIAL VALUES IN INTERIOR.
C
          DO 20 J=1,JM
          DO 20 K=1,KM
                X(J,K)=0.D00
                T(J,K)=0.D00
                ET(J,K)=0.D00
                XX(J,K)=0.D00
                TT(J,K)=0.D00
                XXX(J,K)=0.
 20       CONTINUE
                ES=.1D00
          DO 30 J=1,JM
                XX(J,KM)=-.5D00*DLOG(1.D00+16.D00*ES)
                Y(J)=0.D00
                BX(J)=DFLOAT(J-1)*DP-1.5D00
                WP(J)=0.D00
                X(J,KM)=BX(J)
 30       CONTINUE
                WM(1)=1.D00
                WM(2)=1.D00
                SUM=0.D00
          DO 40 K=1,KM
                WX(K)=DFLOAT(K-1)*DW
                W2(K)=4.D00*ES*WX(K)
                XX(1,K)=-.5D00*DLOG(1.D00+4.D00*ES*WX(K)**2)
                XX(JM,K)=-.5D00*DLOG(1.D00+4.D00*ES*WX(K)**2)
                X(1,K)=-1.5D00
                X(JM,K)=1.5D00
 40       CONTINUE
C
C MATRIX FORMS;SOLUTION BY MATRIX ITERATION;ADI
C
          C1=2.D00
                MM=0
  45            MM=MM+1
          IF(MM-200) 50,50,60
 60       WRITE(6,70)
 70       FORMAT(' ',2X,'NO OF IT EX 400')
          GO TO 75
 50       DO 80 J=2,J1
          DO 90 K=2,K1
          AA(K)=-DEXP(TT(J,K)-XX(J,K))+1.5D00*DW*W2(K)*DEXP(TT(J,K)+
     &          XX(J,K))
            BB(K)=2.D00*DEXP(TT(J,K)-XX(J,K))+C1*DEXP(XX(J,K)-TT(J,K))
        CC(K)=-DEXP(TT(J,K)-XX(J,K))-1.5D00*DW*W2(K)*DEXP(TT(J,K)+
     &          XX(J,K))
      RHS(K)=DEXP(XX(J,K)-TT(J,K))*(XX(J-1,K)-2.D00*XX(J,K)+XX(J+1,K))
     &+DW**2*W2(K)**2*DEXP(3.D00*XX(J,K)+TT(J,K))+4.D00*ES*DW**2*DEXP
     &  (TT(J,K)+XX(J,K))+DEXP(XX(J,K)-TT(J,K))*(XX(J+1,K)-XX(J-1,K)
     & -TT(J+1,K)+TT(J-1,K))*(XX(J+1,K)-XX(J-1,K))*.25D00+C1*XX(J,K)
```

```
        & *DEXP(XX(J,K)-TT(J,K))
   90     CONTINUE
                K=1
          AA(K)=0.D00
                BB(K)=2.D00*DEXP(TT(J,K)-XX(J,K))+3.D00*DW*W2(K)*
        & DEXP(TT(J,K)+XX(J,K))+C1*DEXP(XX(J,K)-TT(J,K))
                CC(K)=-2.D00*DEXP(TT(J,K)-XX(J,K))
         RHS(K)=DEXP(XX(J,K)-TT(J,K))*(XX(J-1,K)-2.D00*XX(J,K)+XX(J+1,K))
        &+DW**2*W2(K)**2*DEXP(3.D00*XX(J,K)+TT(J,K))+4.D00*ES*DW**2*DEXP
        &  (TT(J,K)+XX(J,K))+DEXP(XX(J,K)-TT(J,K))*(XX(J+1,K)-XX(J-1,K)
        & -TT(J+1,K)+TT(J-1,K))*(XX(J+1,K)-XX(J-1,K))*.25D00
        & +XX(J,K)*C1*DEXP(XX(J,K)-TT(J,K))+DW*W2(K)*2.D00*DEXP(XX(J,K)+
        & TT(J,K))
          IF(DABS(X(J,1)) .LE. .5) RHS(K)=RHS(K)-DW*4.D00*DEXP(TT
        &   (J,K))
               RHS(K1)=RHS(K1)-XX(J,KM)*CC(K1)
                CC(K1)=0.D00
          CALL TRID(1,42)
          DO 100 K=1,K1
                OT(J,K)=XX(J,K)
  100           XX(J,K)=1.68*(RHS(K)-XX(J,K))+XX(J,K)
   80       CONTINUE
C  'SOLVE IMPLICITLY IN THE PHI DIRECTION.'
          DO 110 K=1,K1
          DO 120 J=2,J1
               AA(J)=DEXP(XX(J,K)-TT(J,K))*((XX(J+1,K)-XX(J-1,K)-TT(J+
        &          1,K)+TT(J-1,K))*.25D00-1.D00)
          BB(J)=2.D00*DEXP(XX(J,K)-TT(J,K))+C1*DEXP(TT(J,K)-XX(J,K))
              CC(J)=-DEXP(XX(J,K)-TT(J,K))*((XX(J+1,K)-XX(J-1,K)-TT(J+
        &          1,K)+TT(J-1,K))*.25D00+1.D00)
         IF(K .GT. 1) RHS(J)=W2(K)**2*DW**2*DEXP(3.D00*XX(J,K)+TT(J,K))
        &    +4.D00*ES*DW**2*DEXP(XX(J,K)+TT(J,K))+DEXP(TT(J,K)-XX(J,K))*
        &(XX(J,K-1)-2.D00*XX(J,K)+XX(J,K+1))+1.5D00*DW*W2(K)*DEXP(XX(J,K
        &    )+
        &TT(J,K))*(XX(J,K+1)-XX(J,K-1))+C1*XX(J,K)*DEXP(TT(J,K)-XX(J,K))
         IF(K .EQ. 1) RHS(J)=W2(K)**2*DW**2*DEXP(3.D00*XX(J,K)+TT(J,K))
        &    +4.D00*ES*DW**2*DEXP(XX(J,K)+TT(J,K))+DEXP(TT(J,K)-XX(J,K))*
        &(2.D00*(XX(J,2)-XX(J,1))+2.D00*DW*W2(K)*DEXP(2.D00*XX(J,K)))+3.
        & D00*DW*W2(K)
        &  *DEXP(XX(J,K)+TT(J,K))*(XX(J,K+1)-XX(J,K))+C1*XX(J,K)
        & *DEXP(TT(J,K)-XX(J,K))
          IF(DABS(X(J,1)) .LE. .5 .AND. K .EQ. 1) RHS(J)=RHS(J)-DW*
        &   4.D00*DEXP(TT(J,K))
  120     CONTINUE
                RHS(J1)=RHS(J1)-CC(J1)*XX(JM,K)
                RHS(2)=RHS(2)-AA(2)*XX(1,K)
                CC(J1)=0.D00
                AA(2)=0.D00
          CALL TRID(2,63)
          DO 130 J=2,J1
                OT(J,K)=XX(J,K)
  130           XX(J,K)=1.68*(RHS(J)-XX(J,K))+XX(J,K)
  110     CONTINUE
                MN=0
   47           MN=MN+1
          IF(MN-200)57,57,67
   67     WRITE(6,77)
   77     FORMAT(' ',2X,'NO OF IT EX 400')
          GO TO 75
C ' SOLVE FOR "E" '
```

```
57        DO 142 J=2,J1
          DO 152 K=2,K1
                AA(K)=-1.D00
                BB(K)=0.D00
                CC(K)=1.D00
         RHS(K)=(XX(J,K+1)-XX(J,K-1))+2.D00*DW*W2(K)*DEXP(2.*XX(J,K))
152       CONTINUE
                K=1
                AA(K)=0.D00
                BB(K)=-1.D00
                CC(K)=1.D00
                RHS(K)=0.D00
       IF(DABS(X(J,K)) .LE. .5) RHS(K)=2.D00*DW*DEXP(XX(J,K))
          CALL TRID(1,42)
          DO 162 K=1,K1
                XXX(J,K)=TT(J,K)
162             TT(J,K)=TT(J,K)+1.68*(RHS(K)-TT(J,K))
142       CONTINUE
         DO 167 J=2,J1
         DO 167 K=1,K1
                 T3=DABS(XXX(J,K)-TT(J,K))
                 IF(T3 .GT. .0001) GO TO 47
167      CONTINUE
          IF(MOD(MN,10) .EQ. 0) PRINT 6060,MN
6060      FORMAT(//8X,I4,//)
C ' SOLVE FOR "X" '
          DO 140 J=2,J1
          DO 150 K=2,K1
                AA(K)=-DEX2(TT(J,K)-XX(J,K))+DW*W2(K)*DEXP(TT(J,K)+
      &         XX(J,K))*.5D00
         BB(K)=2.D00*DEXP(TT(J,K)-XX(J,K))+C1*DEXP(XX(J,K)-TT(J,K))
                CC(K)=-DEXP(TT(J,K)-XX(J,K))-DW*W2(K)*DEXP(TT(J,K)+
      &         XX(J,K))*.5D00
        RHS(K)=DEXP(XX(J,K)-TT(J,K))*(X(J-1,K)-2.D00*X(J,K)+X(J+1,K))
      &+(X(J+1,K)-X(J-1,K))*DEXP(XX(J,K)-TT(J,K))*(XX(J+1,K)-XX(J-1,K)
      & -TT(J+1,K)+TT(J-1,K))*.25D00+C1*X(J,K)*DEXP(XX(J,K)-TT(J,K))
150       CONTINUE
                K=1
                AA(K)=0.D00
                BB(K)=2.D00*DEXP(TT(J,K)-XX(J,K))+1.D00*DW*W2(K)*
      & DEXP(TT(J,K)+XX(J,K))+C1*DEXP(XX(J,K)-TT(J,K))
                CC(K)=-2.D00*DEXP(TT(J,K)-XX(J,K))-1.D00*W2(K)*DW*
      & DEXP(TT(J,K)+XX(J,K))
        RHS(K)=DEXP(XX(J,K)-TT(J,K))*(X(J-1,K)-2.D00*X(J,K)+X(J+1,K))
      &+(X(J+1,K)-X(J-1,K))*DEXP(XX(J,K)-TT(J,K))*(XX(J+1,K)-XX(J-1,K)
      & -TT(J+1,K)+TT(J-1,K))*.25D00+C1*X(J,K)*DEXP(XX(J,K)-TT(J,K))
       IF(DABS(X(J,K)).LT..5D00)RHS(K)=RHS(K)+2.D00*DW*DSIN(WP(J))*DEX
      &   P(TT(J,K))
                RHS(K1)=RHS(K1)-CC(K1)*X(J,K1)
                 CC(K1)=0.D00
          CALL TRID(1,42)
                DO 160 K=1,K1
                 T(J,K)=X(J,K)
160             X(J,K)=1.68*(RHS(K)-X(J,K))+X(J,K)
                Y(J)=0.D00
          IF(DABS(X(J,1)) .LT. .5) Y(J)=(.25D00-X(J,1)**2)**.5
                WP(J)=0.D00
          IF(DABS(X(J,1)) .LT. .5) WP(J)=DATAN(-X(J,1)/Y(J))
          IF(X(J,1) .EQ. -5.D00) WP(J)=3.1415927/2.D00
          IF(X(J,1) .EQ. .5D00) WP(J)=-3.1415927/2.D00
```

```
 140      CONTINUE
C  'SOLVE IMPLICITLY IN THE PHI DIRECTION  '
            DO 170 K=1,K1
            DO 180 J=2,J1
                  AA(J) =DEXP(XX(J,K)-TT(J,K))*((XX(J+1,K)-XX(J-1,K)-TT(J+
      &             1,K)+TT(J-1,K))*.25D00-1.D00)
          BB(J)=2.D00*DEXP(XX(J,K)-TT(J,K))+C1*DEXP(TT(J,K)-XX(J,K))
                  CC(J) =-DEXP(XX(J,K)-TT(J,K))*((XX(J+1,K)-XX(J-1,K)-TT(J+
      &             1,K)+TT(J-1,K))*.25D00+1.D00)
       IF(K.GT.1)RHS(J)=DEXP(TT(J,K)-XX(J,K))*(X(J,K+1)-X(J,K)*2.D00+
      & X(J,K-1))+DW*W2(K)*(X(J,K+1)-X(J,K-1))*.5D00
      & *DEXP(TT(J,K)+XX(J,K))+C1*X(J,K)*DEXP(TT(J,K)-XX(J,K))
       IF(K .EQ. 1)RHS(J)=DEXP(TT(J,K)-XX(J,K))*(2.D00*(X(J,2)-X(J,1)
      & )+2.D00*DW*DSIN(WP(J))*DEXP(XX(J,K)))
      & +DW*W2(K)*(X(J,K+1)-X(J,K))*DEXP(TT(J,K)+XX(J,K))+C1*X(J,K)
      & *DEXP(TT(J,K)-XX(J,K))
 180     CONTINUE
                  RHS(J1)=RHS(J1)-CC(J1)*X(JM,K)
                  RHS(2)=RHS(2)-AA(2)*X(1,K)
                  AA(2)=0.D00
                  CC(J1)=0.D00
           CALL TRID(2,63)
           DO 200 J=2,J1
                  T(J,K)=X(J,K)
 200              X(J,K)=1.68*(RHS(J)-X(J,K))+X(J,K)
 170       CONTINUE
            DO 210 J=2,J1
                  Y(J)=0.D00
           IF(DABS(X(J,1)) .LT. .5) Y(J)=(.25D00-X(J,1)**2)**.5
                  WP(J)=0.D00
           IF(DABS(X(J,1)) .LT. .5) WP(J)=DATAN(-X(J,1)/Y(J))
           IF(X(J,1) .EQ. -5.D00) WP(J)=3.1415927/2.D00
           IF(X(J,1) .EQ. .5D00) WP(J)=-3.1415927/2.D00
 210        CONTINUE
C CALCULATION OF OPTIMAL ACCELLERATION PARAMETER
                  SUM=0.D00
           DO 220 K=1,K1
           DO 220 J=2,J1
                  ETT(J,K)=DABS(TT(J,K)-XXX(J,K))
                  SUM=SUM+ETT(J,K)
 220       CONTINUE
                  SR(MM)=SUM
           IF(MM .GT. 1) AL(MM)=SR(MM)/SR(MM-1)
           IF(MM .GT. 10) T1=DABS(WM(MM) -
      &     WM(MM-10))/(2.D00-WM(MM))
           IF(MM .GT. 10 .AND. T1 .LT. .05D00)
      &     WM(MM+1)=WM(MM)
           IF(MM .GT. 10 .AND. T1 .LT. .05D00)
      &     GO TO 250
           IF(MOD(MM,10) .NE. 0) GO TO 250
       ER(MM)=2.D00/(1.D00+DSQRT(DABS((1.D00-(AL(MM)+WM(MM)-1.D00)**2/
      &       (WM(MM)**2*AL(MM))))))
             WM(MM+1)=ER(MM)-(2.D00-ER(MM))/4.D00
           WRITE(6,7000)WM(MM+1),T1,AL(MM),SR(MM)
 7000      FORMAT(' ',4(D15.6)//)
 250       CONTINUE
           IF(MOD(MM,10) .NE. 0 .AND. MM .GE. 10) WM(MM+1)=WM(MM)
           IF(MM .LE. 9) WM(MM+1)=1.D00
           DO 270 J=2,J1
           DO 270 K=1,K1
```

```
            T2=DABS(OT(J,K)-XX(J,K))
            T4=DABS(T(J,K)-X(J,K))
            IF(T2 .GT. .0001D00) GO TO 45
            IF(T4 .GT. .0001D00) GO TO 45
  270       CONTINUE
            WRITE(6,7200) WM(MM)
  7200      FORMAT(' ','OPTIMAL ITERATION PARAMETER #1 =',D13.6)
  75        CONTINUE
            SUM100=0.D00
C CALULATION OF ERROR, TOTAL ERROR
            SUM200=0.D00
            SUM300=0.D00
            T5=DABS(OT(2,1)-XX(2,1))
            T6=DABS(T(2,1)-X(2,1))
            T7=DABS(XXX(2,1)-TT(2,1))
            DO 278 J=2,J1
            DO 278 K=1,K1
            T2=DABS(OT(J,K)-XX(J,K))
            T3=DABS(T(J,K)-X(J,K))
            T4=DABS(XXX(J,K)-TT(J,K))
            IF(T5 .LE. T2) T5=T2
            IF(T6 .LE. T3) T6=T3
            IF(T7 .LE. T4) T7=T4
            SUM100=T2+SUM100
            SUM200=T3+SUM200
            SUM300=T4+SUM300
   278      CONTINUE
            PRINT 6000,MM,SUM100,SUM200,SUM300,T5,T6,T7
  6000      FORMAT(//8X,I3,6(5X,D13.6),//)
            DO 274 J=2,J1
            DO 274 K=1,K1
                  XX(J,K)=DEXP(XX(J,K))
                  TT(J,K)=DEXP(TT(J,K))
  274       OT(J,K)=0.D00
            T1=2.
C CALCULATION OF VAN DYKE'S VALUES FOR SPEED
            DO 284 J=2,J1
            IF(DABS(X(J,1)) .LE. .5) SUM1(J)=2.*DCOS(WP(J))+ES*(
      & .5*DCOS(WP(J))**2+DCOS(3.*WP(J))/8.+.5*DLOG(T1)*DCOS(WP(J))
      & -.375*DCOS(WP(J))+.5*(DLOG(4./ES)-.154)*
      & DCOS(WP(J)))
            IF(DABS(X(J,1)) .LE. .5) SUM2(J)=ES*2.*(-.25*DCOS(WP(J))
      & **2*DSIN(WP(J))-.25*DLOG(T1)*DSIN(WP(J))+DSIN(WP(J))/16.
      & +DSIN(WP(J)*3.)/16.)
           SUM5(J)=DSQRT(SUM1(J)**2+SUM2(J)**2)
  294      CONTINUE
            DO 262 J=2,J1
            IF(DABS(X(J,1)) .LT. .5D00) PRINT 8989,SUM5(J),WP(J),J
  8989      FORMAT(//8X,2(10X,D16.8),10X,I3)
  292       CONTINUE
            RETURN
            END
C TRIDIAGCNAL SOLVER
            SUBROUTINE TRID(NL,NU)
                 DOUBLE PRECISION A,B,C,D,G
                 REAL*8 Z
            COMMON/SANDAL/A,B,C,D,G
            DIMENSION A(64),B(64),C(64),D(64),G(64)
                  D(NL)=C(NL)/B(NL)
                  G(NL)=G(NL)/B(NL)
                 NLP1=NL+1
            DO 290 N=NLP1,NU
                 Z=1.D00/(B(N)-A(N)*D(N-1))
                 D(N)=C(N)*Z
      290        G(N)=(G(N)-A(N)*G(N-1))*Z
                 NUPNL=NU+NL
            DO 300 NN=NLP1,NU
                 N=NUPNL-NN
      300       G(N)=G(N)-D(N)*G(N+1)
            RETURN
            END
C PRINTING OF RESULTS:X,XX,TT
            SUBROUTINE PRIN
            DOUBLE PRECISION X,XX,Y,TT,WX,BX,T,OT,ET,ETT,AL,AM,SR,XXX
         &  ,AA,BB,CC,DD,RHS,WP,EE,WM,SUM1,SUM2,SUM5,W2
            COMMON /TITAN/X,XX,Y,TT/GORGO/WX,BX
         &  /SPAR/T,OT,ET,ETT/TROY/AL,AM,SR,XXX
         &  /SANDAL/AA,BB,CC,DD,RHS/ARROW/WP,EE,WM
         &  /TRUE/SUM1,SUM2,SUM5,W2
            DIMENSION X(64,43),XX(64,43),Y(64),TT(64,43),
         &  WX(64),BX(64),T(64,43),OT(64,43),ET(64,43),ETT(64,43),
         &  AL(200),AM(200),SR(200),XXX(64,43),AA(64),
         &  BB(64),CC(64),DD(64),RHS(64),WP(64),
         &  EE(200),WM(200),SUM1(64),SUM2(64),SUM5(64),W2(43)
            END
            PRINT 1500,(K,K=1,8)
      1500  FORMAT(//8X,'K',3X,8(5X,I2,2X))
            WRITE(6,1600)(WX(K),K=1,8)
      1600  FORMAT(' ',6X,'WX(K)',8(1X,D13.6))
            WRITE(6,1700)
      1700  FORMAT(//,'J',2X,'BX(J)',4X,'XX(J,K)')
            DO 1800 J=1,JM
            WRITE(6,1850) J,BX(J),(XX(J,K),K=1,8)
      1800  CONTINUE
      1850  FORMAT(' ',I2,1X,D13.6,8(1X,D13.6))
            PRINT 2002,(K,K=1,8)
      2002  FORMAT(//8X,'K',3X,8(5X,I2,2X))
            WRITE(6,2102)(WX(K),K=1,8)
      2102  FORMAT(' ',6X,'WX(K)',8(1X,D13.6))
            WRITE(6,2202)
      2202  FORMAT(//,'J',2X,'BX(J)',4X,'TT(J,K)')
            DO 2302 J=1,JM
            WRITE(6,2402) J,BX(J),(TT(J,K),K=1,8)
      2302  CONTINUE
      2402  FORMAT(' ',I2,1X,D13.6,8(1X,D13.6))
            PRINT 2004,(K,K=1,8)
      2004  FORMAT(//8X,'K',3X,8(5X,I2,2X))
            WRITE(6,2104)(WX(K),K=1,8)
      2104  FORMAT(' ',6X,'WX(K)',8(1X,D13.6))
            WRITE(6,2204)
      2204  FORMAT(//,'J',2X,'BX(J)',4X,'X(J,K)')
            DO 2304 J=1,JM
            WRITE(6,2404) J,BX(J),(X(J,K),K=1,8)
      2304  CONTINUE
      2404  FORMAT(' ',I2,1X,D13.6,8(1X,D13.6))
            RETURN
            END
/*
//GO.SYSIN DD *
//
```

APPENDIX C

```
      C|------------------------------------------------------------------|
      C|                                                                  |
      C|               INVERSE FLOW PROBLEM                               |
      C|               --------------------                               |
      C|   PROGRAM FOR SOLVING INVERSE PROBLEM : THE SPEED Q IS           |
      C|   SPECIFIED ON THE VANISHING STREAMLINE . SUBSEQUENTLY THE       |
      C|   ANGLE ALPHA CAN BE DETERMINED BY SOLUTION OF A SECOND          |
      C|   ORDER PDE. THEN X IS DETERMINED IN A SIMILAR MANNER.           |
      C|   LASTLY HAVING VALUES FOR X,DX,ALPHA,EXPQ THE PROFILE CAN BE    |
      C|   BY INTEGRATION OF A PAIR OF SECOND ORDER ORDINARY DIFFER-      |
      C|   ENTIAL EQUATIONS.                                              |
      C|                                                                  |
      C|          BY GEORGE GROSSMAN                                      |
      C|           UNIVERSITY OF WINDSOR                                  |
      C|                                                                  |
      C|------------------------------------------------------------------|
      C
      C  VARIABLE REPRESENTATION :AN IS -LNQ/2 WHERE Q IS SPEED
      C      T IS ANGLE OF INCLINATION OF TANGENT TO STREAMLINE PSI
             X IS CARTESIAN COORDINATE
             U2 IS PROFILE FUNCTION AS A FUNCTION OF X
      C
             BLOCK DATA
             COMMON /STAT/ JM,KM,J1,K1,PM,PS,WW,WS
             DATA JM/65/,KM/43/,J1/64/,K1/42/,PM/2.0/,PS/-2.0/,WW/2.625/,
          &  WS/0./
             COMMON /TITAN/X,XX,Y,TT/GORGO/WX,BX
          &  /SPAR/T,OT,ET,ETT/TROY/AL,AM,SR,AN
          &  /SANDAL/AA,BB,CC,DD,RHS/ARROW/U2,UU1,WP,EE,WM
             DIMENSION X(65,43),XX(65,43),Y(65),TT(65,43),
          &  WX(65),BX(65),T(65,43),OT(65,43),ET(65,43),ETT(65,43),
          &  AL(200),AM(200),SR(200),AN(65,43),AA(65),
          &  BB(65),CC(65),DD(65),RHS(65),WP(65),
           &  EE(200),WM(200),U2(65),UU1(65)
             END
       C
       C MAIN PROGRAM
       C
             CALL SOLV
             CALL PRIN
             STOP
             END
       C
       C  CALCULATIONS:BOUNDARY AND INTERIOR CONDITIONS,MATRIX ITERATIONS
       C ,OPTIMAL ACCELERATION PARAMETER SEARCH.
       C
              SUBROUTINE SOLV
              COMMON /STAT/ JM,KM,J1,K1,PM,PS,WW,WS
              COMMON /TITAN/X,XX,Y,TT/GORGO/WX,BX
           &  /SPAR/T,OT,ET,ETT/TROY/AL,AM,SR,AN
           &  /SANDAL/AA,BB,CC,DD,RHS/ARROW/U2,UU1,WP,EE,WM
              DIMENSION X(65,43),XX(65,43),Y(65),TT(65,43),
           &  WX(65),BX(65),T(65,43),OT(65,43),ET(65,43),ETT(65,43),
           &  AL(200),AM(200),SR(200),AN(65,43),AA(65),
           &  BB(65),CC(65),DD(65),RHS(65),WP(65),
            &  EE(200),WM(200),U2(65),UU1(65)
        C LEADING EDGE AND TRAILING EDGE
              IWORK=23
              IY=43
              IP=(IY-IWORK)/2
```

```
C GRID SPACING
           DP=(PE-PS~~)/J1~~
           DW=(WE-WS)/K1
C SPECIFYING CONSTANTS FOR INCOMING FLOW
            C6=1.
            C7=1.45
            C8=0.
            C9=1.
C VALUE FOR 'K' AND COS THETA , SIN THETA
            T5=SQRT(C7**2+C9**2)/SQRT(C6**2+C8**2)
            T6=(C6*C7+C8*C9)/SQRT((C6**2+C8**2)*(C7**2+C9**2))
            T7=(C6*C9-C8*C7)/SQRT((C6**2+C8**2)*(C7**2+C9**2))
            C13=3.14159625/2.
            IF(T6 .NE. 0.)C13=ATAN(T7/T6)
C
C BOUNDARY CONDITIONS,GRID SPECIFICATION,INITIAL VALUES IN INTERIOR.
C
           DO 103 J=1,JM
 103                BX(J)=(J-1)*DP-2.0
           DO 10 J=1,JM
           DO 20 K=1,KM
                    SUM3(J,K)=0.
                  AN(J,K)=0.
                    T(J,K)=0.
                    ET(J,K)=0.
                    TT(J,K)=0.
 20        CONTINUE
 10        CONTINUE
            WX(KM)=2.625
            WX(1)=0.
           AN(23,1)=.55
           AN(24,1)=.5
           AN(33,1)=-.20
           AN(43,1)=.55
           AN(42,1)=.50
           DO 30 J=1,JM
                    Y(J)=0.
                    BX(J)=(J-1)*DP-2.0
                    W1(J)=0.
                    X(J,KM)=C6*BX(J)+C7*WX(KM)
C SPECIFYING T (LNE/2) ON VANISHING STREAMLINE AS A LINEAR FN
          IF(J .GT. 23 .AND. J .LE. 33) AN(J,1)=AN(24,1)*
     &    (BX(J)-BX(33))/(BX(24)-BX(33))+AN(33,1)*(BX(J)-BX(24))/
     &    (BX(33)-BX(24))
          IF(J .GE. 33 .AND. J .LT. 43) AN(J,1)=AN(42,1)*
     &    (BX(J)-BX(33))/(BX(42)-BX(33))+AN(33,1)*(BX(J)-BX(42))/
     &    (BX(33)-BX(42))
          IF(J .GT. 0 .AND. J .LT. 24) AN(J,1)=
     &    0.*(BX(J)-BX(23))/(BX(1)-BX(23))+AN(23,1)*(BX(J)-BX(1))/
     &    (BX(23)-BX(1))
          IF(J .GT. 42 .AND. J .LT. 66) AN(J,1)=
     &    0.*(BX(J)-BX(43))/(BX(65)-BX(43))+AN(43,1)*(BX(J)-BX(65))/
     &    (BX(43)-BX(65))
 30                 CONTINUE
                    WM(1)=1.00
                    WM(2)=1.00
                    SUM=0.
           DO 40 K=1,KM
                    WX(K)=(K-1)*DW
                    X(1,K)=C6*BX(1)+C7*WX(K)
                  T(JM,K)=C6*BX(JM)+C7*WX(K)
 40         CONTINUE
C
C MATRIX FORMS;SOLUTION BY MATRIX ITERATION;ALTERNATING SLOR
C
            C2=8.25
 45               MM=MM+1
            IF(MM-200)50,50,60
 60         WRITE(6,70)
 70         FORMAT(' ',2X,'NO OF IT EX MAX')
            GO TO 75
 50        CONTINUE
C SOLVING FOR AN (LNE/2) EVERYWHERE IN THE FLOW FIELD HAVING
C SPECIFIED IT AS LINEAR FUNCTION OF PHI
            DO 80 J=2,J1
            DO 90 K=2,K1
                  AA(K)=-1.
                  BB(K)=2.+C2
                  CC(K)=-1.
                RHS(K)=C2*AN(J,K)+T5**2*(AN(J-1,K)-2.*AN(J,K)+AN(J+1,K))
     & -.5*T5*T6*(AN(J+1,K+1)-AN(J+1,K-1)-AN(J-1,K+1)+AN(J-1,K-1))
 90        CONTINUE
                  RHS(K1)=RHS(K1)-CC(K1)*AN(J,KM)
                  CC(K1)=0
                  RHS(2)=RHS(2)-AA(2)*AN(J,1)
                  AA(2)=0.
            CALL TRID(2,42)
            DO 100 K=2,K1
                  OT(J,K)=AN(J,K)
 100              AN(J,K)=AN(J,K)+ 1.80*(RHS(K)-AN(J,K))
 80          CONTINUE
            DO 110 K=2,K1
            DO 120 J=2,J1
                  AA(J)=-T5**2
                  BB(J)=2.*T5**2+C2
                  CC(J)=-T5**2
          IF(K .GT.1) RHS(J)=C2*AN(J,K)+(AN(J,K+1)-2.*AN(J,K)+AN(J,K-1))
     & -.5*T5*T6*(AN(J+1,K+1)-AN(J+1,K-1)-AN(J-1,K+1)+AN(J-1,K-1))
 120       CONTINUE
           RHS(J1)=RHS(J1)-AN(JM,K)*CC(J1)
              CC(J1)=0.
           RHS(2)=RHS(2)-AN(1,K)*AA(2)
           AA(2)=0.
            CALL TRID(2,64)
            DO 130 J=2,J1
                  OT(J,K)=AN(J,K)
 130              AN(J,K)=AN(J,K)+ 1.80*(RHS(J)-AN(J,K))
 110       CONTINUE
C SOLVING FOR ALPHA EVERYWHERE IN THE FLOW FIELD USING VON NEUMANN
C BOUNDARY CONDITIONS
            DO 81 J=2,J1
            DO 91 K=2,K1
                  AA(K)=-1.
                  BB(K)=2.+C2
                  CC(K)=-1.
                RHS(K)=C2*T(J,K)+T5**2*(T(J-1,K)-2.*T(J,K)+T(J+1,K))
     & -.5*T5*T6*(T(J+1,K+1)-T(J+1,K-1)-T(J-1,K+1)+T(J-1,K-1))
 91        CONTINUE
                  RHS(K1)=RHS(K1)-CC(K1)*T(J,KM)
                  CC(K1)=0.
```

```
               K=1
               BB(K)=2.+C2
               CC(K)=-2.
             RHS(K)=C2*T(J,K)+T5**2*(T(J-1,K)-2.*T(J,K)+T(J+1,K))
     & -T5*T6*(T(J+1,K+1)-T(J+1,K)-T(J-1,K+1)+T(J-1,K))
     & -(T5*T7*(AN(J+1,1)-AN(J-1,1))+(T(J+1,1)-T(J-1,1))*T5*T6)
               AA(K)=0.
         CALL TRID(1,42)
         DO 101 K=1,K1
               TT(J,K)=T(J,K)
 101           T(J,K)=T(J,K)+ 1.90*(RHS(K)-T(J,K))
 81       CONTINUE
         DO 111 K=1,K1
         DO 121 J=2,J1
               AA(J)=-T5**2
               BB(J)=2.*T5**2+C2
               CC(J)=-T5**2
       IF(K .GT.1)  RHS(J)=C2*T(J,K)+(T(J,K+1)-2.*T(J,K)+T(J,K-1))
     & -.5*T5*T6*(T(J+1,K+1)-T(J+1,K-1)-T(J-1,K+1)+T(J-1,K-1))
         IF(K .EQ.1) RHS(J)=C2*T(J,K)+2.*(T(J,2)-T(J,1))
     & -T5*T6*(T(J+1,K+1)-T(J+1,K)-T(J-1,K+1)+T(J-1,K))
     & -(T5*T7*(AN(J+1,1)-AN(J-1,1))+(T(J+1,1)-T(J-1,1))*T5*T6)
 121    CONTINUE
        RHS(J1)=RHS(J1)-T(JM,K)*CC(J1)
             CC(J1)=0.
        RHS(2)=RHS(2)-T(1,K)*AA(2)
        AA(2)=0.
         CALL TRID(2,64).
         DO 131 J=2,J1
               TT(J,K)=T(J,K)
 131           T(J,K)=T(J,K)+ 1.80*(RHS(J)-T(J,K))
 111    CONTINUE
C TOLERANCE CHECK FOR UNKNOWNS AN AND T
         DO 219 J=2,J1
         DO 219 K=1,K1
          C10=0.
          IF(K .GT. 1)C10 =ABS(OT(J,K)-AN(J,K))
         IF(C10 .GT. .0001) GO TO 45
          C10 =ABS(T(J,K)-TT(J,K))
         IF(C10 .GT. .0001) GO TO 45
 219     CONTINUE
 75      CONTINUE
C DEPENDING ON APPROACH TAKEN WE CAN FIND ALPHA ON VANISHING
C STREAMLINE BY USING FORWARD AND BACKWARD DIFFERENCING
         J=23
         IF(J .EQ. 23) GO TO 888
            WP(J)=WP(J-1)-.5*(-AN(J-1,3)+4.*AN(J-1,2)-3.*AN(J-1,1))
     & /(T5*T7)+(AN(J,1)-AN(J-1,1))*T6/T7
          L=42
        WP(L+1)=WP(L+2)+.5*(-AN(L+1,3)+4.*AN(L+1,2)-3.*AN(L+1,1))
     & /(T5*T7)-(AN(L+2,1)-AN(L+1,1))*T6/T7
         DO 777 J=25,33,2
            WP(J)=WP(J-2)-(-AN(J-1,3)+4.*AN(J-1,2)-3.*AN(J-1,1))
     & /(T5*T7)+(AN(J,1)-AN(J-2,1))*T6/T7
            WP(J-1)=WP(J-2)-.5*(-AN(J-2,3)+4.*AN(J-2,2)-3.*AN(J-2,1))
     & /(T5*T7)+(AN(J-1,1)-AN(J-2,1))*T6/T7
 777       CONTINUE
         DO 79 J=33,41,2
                L=43+(33-J)-2
            WP(L)=WP(L+2)+(-AN(L+1,3)+4.*AN(L+1,2)-3.*AN(L+1,1))
        & /(T5*T7)-(AN(L+2,1)-AN(L,1))*T5/T7
           WP(L+1)=WP(L+2)+.5*(-AN(L+1,3)+4.*AN(L+1,2)-3.*AN(L+1,1))
        & /(T5*T7)-(AN(L+2,1)-AN(L+1,1))*T6/T7
 79         CONTINUE
C
 999        CONTINUE
            PRINT 2221,MM
 2221       FORMAT(//3X,'NO OF IT EQ',I3,//)
             DO 666 J=23,43
             WP(J)=T(J,1)
             PRINT 2222, WP(J)
 2222        FORMAT(/,7X,D15.8,/)
  666        CONTINUE
            C2=3.25
             MM=0
  47               MM=MM+1
            IF(MM-207)57,57,67
 67         WRITE(6,71)
 71         FORMAT(' ',2X,'NO OF IT EX MAX')
            GO TO 78
 57        CONTINUE
C   CALCULATE X EVERYWHERE IN THE FLOW FIELD
            DO 85 J=2,J1
            DO 95 K=2,K1
                  AA(K)=-1.
                  BB(K)=2.+C2
                  CC(K)=-1.
                  RHS(K)=C2*X(J,K)+T5**2*(X(J-1,K)-2.*X(J,K)+X(J+1,K))
        & -.5*T5*T6*(X(J+1,K+1)-X(J+1,K-1)-X(J-1,K+1)+X(J-1,K-1))
 95        CONTINUE
                  K=1
                  AA(K)=0.
                  BB(K)=2.+C2
                  CC(K)=-2.
                  C16=WP(J)+C13
                  RHS(K)=C2*X(J,K)+T5**2*(X(J-1,K)-2.*X(J,K)+X(J+1,K))
       & -T5*T6*(X(J+1,K+1)-X(J+1,K)-X(J-1,K+1)+X(J-1,K))
       & -2.*DW*EXP(AN(J,1))*COS(C16)*T5
           RHS(K1)=RHS(K1)-X(J,KM)*CC(K1)
               CC(K1)=0.
             CALL TRID(1,42)
            DO 105 K=1,K1
                  XX(J,K)=X(J,K)
 105              X(J,K)=X(J,K)+1.80*(RHS(K)-X(J,K))
 85          CONTINUE
            DO 115 K=1,K1
            DO 125 J=2,J1
                  AA(J)=-T5**2
                  BB(J)=2.*T5**2+C2
                  CC(J)=-T5**2
            IF(K .GT. 1) RHS(J)  =C2*X(J,K)+(X(J,K+1)-2.*X(J,K)+X(J,K-1))
       & -.5*T5*T6*(X(J+1,K+1)-X(J+1,K-1)-X(J-1,K+1)+X(J-1,K-1))
             IF(K .EQ. 1)  C16=WP(J)+C13
            IF(K .EQ. 1)RHS(J)=C2*X(J,K)+2.*(X(J,2)-X(J,1))
       & -T5*T6*(X(J+1,K+1)-X(J+1,K)-X(J-1,K+1)+X(J-1,K))
       & -2.*DW*EXP(AN(J,1))*COS(C16)*T5
 125       CONTINUE
           RHS(J1)=RHS(J1)-X(JM,K)*CC(J1)
           RHS(2)=RHS(2)-X(1,K)*AA(2)
             CC(J1)=0.
```

```
                 AA(2)=0.
                CALL TRID(2,64)
                DO 135 J=2,J1
                      XX(J,K)=X(J,K)
     135              X(J,K)=X(J,K)+1.90*(RES(J)-X(J,K))
     115        CONTINUE
    C OPTIMAL ACCELLERATION PARAMETER FOR SLOR
                      SUM=0.D00
                DO 222 K=1,K1
                DO 222 J=2,J1
                      ETT(J,K)=ABS(XX(J,K)-X(J,K))
                      SUM=SUM+ETT(J,K)
     222        CONTINUE
                      SE(MM)=SUM
                IF(MM .GT. 1)  AL(MM)=SE(MM)/SE(MM-1)
                IF(MM .GT. 10)  T1=ABS(WM(MM)-
         &       WM(MM-10))/(2.D00-WM(MM))
                IF(MM .GT. 10 .AND. T1 .LT. .05D00)
         &       WM(MM+1)=WM(MM)
                IF(MM .GT. 10 .AND. T1 .LT. .05D00)
         &       GO TO 250
                IF(MOD(MM,10) .NE. 0) GO TO 250
                C15=((1.D00-(AL(MM)+WM(MM)-1.D00)**2/
         &         (WM(MM)**2*AL(MM))))
                C14=ABS(C15)
            ZZ(MM)=2.D00/(1.D00+SQRT(C14))
                  WM(MM+1)=ZZ(MM)-(2.D00-ZZ(MM))/4.D00
                WRITE(6,7000)WM(MM+1),T1,AL(MM),SE(MM)
     7000       FORMAT(' ',4(D15.6)//)
     250        CONTINUE
                IF(MOD(MM,10) .NE. 0 .AND. MM .GE. 10) WM(MM+1)=WM(MM)
                IF(MM .LE. 19 .AND. MM .GE. 10) WM(MM)=1.35
                IF(MM .LE. 9) WM(MM+1)=1.D00
                IF(MM .LT. 10)  GO TO 47
    C TOLERANCE LEVEL CHECK
                DO 217 J=2,J1
                DO 217 K=1,K1
                 C10 =ABS(X(J,K)-XX(J,K))
                IF(C10 .GT. .0001)  GO TO 47
     217        CONTINUE
     78         CONTINUE
                WRITE(6,7200)WM(MM)
     7200       FORMAT(' ','OPTIMAL ITERATION PARAMETER #1 =',D13.6)
    C   ERROR
                SUM100=0.D00
                SUM200=0.D00
                T55=ABS(OT(2,1)-AN(2,1))
                T6=ABS(XX(2,1)-X(2,1))
                DO 278 J=2,J1
                DO 278 K=1,K1
                T2=ABS(OT(J,K)-AN(J,K))
                IF(K .EQ. 1) T2=0.
                T3=ABS(XX(J,K)-X(J,K))
                IF(K .GT. 1 .AND. T55 .LE. T2)  T55=T2
                IF(T6 .LE. T3)  T6=T3
                SUM100=T2+SUM100
                SUM200=T3+SUM200
      278       CONTINUE
                WRITE(6,7100)MM
     7100       FORMAT(' ','NO OF IT=',I3)
```

```
               PRINT 6000,NN,SUM100,SUM200,T55,T6,T7
      6000     FORMAT(//5X,I3,6(5X,D13.6),//)
C  INTEGRATION OF D2Y/D2X = CURV 'X' EXP(T)
                  LH=IY-IWORK+1
                  IRR=LH-3
                  NGMAX=LH-1
                  NGRID=LH-2
                   DO 5 I=1,LH
                        L=I+IWORK-1
                        UU1(I)=X(L,1)
      5                 U2(I)=0.
                  NN=0
      215         NN=NN+1
                  IF(NN-200) 220,220,225
      225      WRITE(6,230)
      230      FORMAT(' ',2X,'NO OF IT EX 200')
               GO TO 77
      220      DO 235 I=2,IRR
               C6=UU1(I+2)-UU1(I+1)
               C7=UU1(I+2)-UU1(I)
               C8=UU1(I+1)-UU1(I)
               C3=.5*(UU1(I+2)-UU1(I+1))*(UU1(I+1)-UU1(I))*
     &            (UU1(I+2)-UU1(I))
               C9=C3/C7
              C4=(U2(I+1)-U2(I-1))/C7
              AA(I)=-C6-C9*3.*C4*SQRT(1.+C4**2)*(WP(LH+3+I)-WP(I+LH+1))*
     &             EXP(-AN(I+LH+2,1))/(2.*DP)
              BB(I)=C7
              CC(I)=-C8+C9*3.*C4*SQRT(1.+C4**2)*(WP(LH+3+I)-WP(I+LH+1))*
     &             EXP(-AN(I+LH+2,1))/(2.*DP)
              RHS(I)=C6*U2(I-1)+C8*U2(I+1)-U2(I)*C7+C3
     &        *(WP(I+LH+3)-WP(I+LH+1))*SQRT((1.+C4**2)**3)
     &        *EXP(-AN(I+LH+2,1))/(-2.*DP)
      235     CONTINUE
              I=1
               C6=UU1(I+2)-UU1(I+1)
               C7=UU1(I+2)-UU1(I)
               C8=UU1(I+1)-UU1(I)
               C3=.5*(UU1(I+2)-UU1(I+1))*(UU1(I+1)-UU1(I))*
     &            (UU1(I+2)-UU1(I))
               C9=C3/C7
              C4=U2(2)/C7
              AA(I)=0.
              BB(I)=C7
              CC(I)=-C8+C9*3.*C4*SQRT(1.+C4**2)*(WP(LH+3+I)-WP(I+LH+1))*
     &             EXP(-AN(I+LH+2,1))/(2.*DP)
              RHS(I)=-C7*U2(I)+C8*U2(I+1)-C3
     &        *(WP(I+LH+3)-WP(I+LH+1))*SQRT((1.+C4**2)**3)
     &        *EXP(-AN(I+LH+2,1))/(2.*DP)
              I=NGRID
               C6=UU1(I+2)-UU1(I+1)
               C7=UU1(I+2)-UU1(I)
               C8=UU1(I+1)-UU1(I)
               C3=.5*(UU1(I+2)-UU1(I+1))*(UU1(I+1)-UU1(I))*
     &            (UU1(I+2)-UU1(I))
               C9=C3/C7
              C4=-U2(NGRID-1)/C7
              BB(I)=C7
              AA(I)=-C6-C9*3.*C4*SQRT(1.+C4**2)*(WP(LH+3+I)-WP(I+LH+1))*
     &             EXP(-AN(I+LH+2,1))/(2.*DP)
```

```
          CC(I)=0.
          RHS(I)=-C7*U2(I)+C6*U2(I-1)-C3
     &    *(WP(I+LH+3)-WP(I+LH+1))*SQRT((1.+C4**2)**3)
     &    *EXP(-AN(I+LH+2,1))/(2.*DP)
          CALL TRID(1,NGRID)
          DO 284 I=1,NGRID
           U2(I)=U2(I)+RHS(I)
 284      CONTINUE
          DO 289 I=1,NGRID
           T6=ABS(RHS(I))
           IF(T6 .GT. .00001) GO TO 215
 289      CONTINUE
          DO 280 J=1,JM
            DO 280 K=1,KM
           AN(J,K)=EXP(AN(J,K))/(T5*T7)
 280      CONTINUE
          PRINT 358,NN
 358      FORMAT(//8X,I3,//)
          DO 287 J=1,JM
                 U1(J)=U2(J)
                 U2(J)=0.
                 IF(J .GT. IWORK .AND. J .LT. IY) U2(J)=U1(J-IWORK)
          WRITE(6,7390) U2(J),X(J,1)
 287      CONTINUE
 7390     FORMAT(' '10X,2(5X,D15.6))
  77      CONTINUE
          RETURN
          END
C TRIDIAGONAL SOLVER
          SUBROUTINE TRID(NL,NU)
          COMMON/SANDAL/A,B,C,D,G
          DIMENSION A(65),B(65),C(65),D(65),G(65)
                D(NL)=C(NL)/B(NL)
                G(NL)=G(NL)/B(NL)
                NLP1=NL+1
          DO 290 N=NLP1,NU
                Z=1./(B(N)-A(N)*D(N-1))
                D(N)=C(N)*Z
 290            G(N)=(G(N)-A(N)*G(N-1))*Z
                NUPNL=NU+NL
          DO 300 NN=NLP1,NU
                N=NUPNL-NN
 300           G(N)=G(N)-D(N)*G(N+1)
          RETURN
          END
C PRINTING OF RESULTS:X,AN,T
          SUBROUTINE PRIN
          COMMON /STAT/ JM,KM,J1,K1,PM,PS,WW,WS
          COMMON /TITAN/X,XX,Y,TT/GORGO/WX,BX
     &    /SPAR/T,OT,ET,ETT/TROY/AL,AM,SR,AN
     &    /SANDAL/AA,BB,CC,DD,RHS/ARROW/U2,UU1,WP,ER,WM
          DIMENSION X(65,43),XX(65,43),Y(65),TT(65,43),
     &    WX(65),BX(65),T(65,43),OT(65,43),ET(65,43),ETT(65,43),
     &    AL(200),AM(200),SR(200),AN(65,43),AA(65),
     &    BB(65),CC(65),DD(65),RHS(65),WP(65),
     &     ER(200),WM(200),U2(65),UU1(65)
          PRINT 2000,(K,K=1,8)
 2000     FORMAT(//8X,'K',3X,8(5X,I2,2X))
          WRITE(6,2100) (WX(K),K=1,8)
 2100     FORMAT(' ',6X,'WX(K)',8(1X,D13.6))
```

```
          WRITE(6,2200)
2200      FORMAT(//,'J',2X,'BX(J)',4X,'AN(J,K)')
          DO 2300 J=1,JM
          WRITE(6,2400) J,BX(J),(AN(J,K),K=1,8)
2300      CONTINUE
2400      FORMAT(' ',I2,1X,D13.6,8(1X,D13.6))
          PRINT 2500,(K,K=1,8)
2500      FORMAT(//8X,'K',3X,8(5X,I2,2X))
          WRITE(6,2600) (WX(K),K=1,8)
2600      FORMAT(' ',6X,'WX(K)',8(1X,D13.6))
          WRITE(6,2700)
2700      FORMAT(//,'J',2X,'BX(J)',4X,'X(J,K)')
          DO 2800 J=1,JM
          WRITE(6,2900) J,BX(J),(X(J,K),K=1,8)
2800      CONTINUE
2900      FORMAT(' ',I2,1X,D13.6,8(1X,D13.6))
          PRINT 2505,(K,K=1,8)
2505      FORMAT(//8X,'K',3X,8(5X,I2,2X))
          WRITE(6,2605) (WX(K),K=1,8)
2605      FORMAT(' ',6X,'WX(K)',8(1X,D13.6))
          WRITE(6,2705)
2705      FORMAT(//,'J',2X,'BX(J)',4X,'T(J,K)')
          DO 2805 J=1,JM
          WRITE(6,2900) J,BX(J),(T(J,K),K=1,8)
2805      CONTINUE
2905      FORMAT(' ',I2,1X,D13.6,8(1X,D13.6))
          RETURN
          END
/*
//GO.SYSIN DD *
//
```

VITA AUCTORIS

The author was born in Ottawa, Canada in 1952. He received a B.A. in Honours Mathematics from York University, Toronto in 1980. In 1982 he received a M.Sc. in Applied Mathematics from the University of Windsor, Windsor, Ontario.