AD-A042 984 AIR FORCE FLIGHT DYNAMICS LAB WRIGHT-PATTERSON AFB OHIO F/G 20/4
INTEGRAL EQUATION METHODS FOR TWO DIMENSIONAL INCOMPRESSIBLE FL--ETC(U)
APR 77 C L KELLER
UNCLASSIFIED AFFDL-TR-77-27 NL

1 OF 2
AD A042 984

AFFDL-TR-77-27

(13)

AD A042984

INTEGRAL EQUATION METHODS FOR TWO DIMENSIONAL INCOMPRESSIBLE FLOWS FOR MULTIELEMENT AIRFOILS

Analysis and Optimization Branch
Structural Mechanics Division

April 1977

TECHNICAL REPORT AFFDL-TR-77-27

Final Report for Period December 1974 to January 1977

DDC
RECEIVED
AUG 16 1977
C

AD NO.
DDC FILE COPY

Approved for public release; distribution unlimited

AIR FORCE FLIGHT DYNAMICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
Air Force Systems Command
Wright-Patterson Air Force Base, Ohio 45433

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Information Office (IO) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Charles L. Keller

CHARLES L. KELLER
Project Engineer
Applied Mathematics Group
Analysis & Optimization Branch

Richard D. Krobusek

RICHARD D. KROBUSEK, Major, USAF
Chief, Analysis & Optimization Branch
Structural Mechanics Division

FOR THE COMMANDER

Howard L. Farmer

HOWARD L. FARMER, Colonel, USAF
Chief, Structural Mechanics Division

Copies of this report should not be returned unless return is required by security considerations, contractural obligations, or notice on a specific document.

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFFDL-TR-77-27	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): INTEGRAL EQUATION METHODS FOR TWO DIMENSIONAL INCOMPRESSIBLE FLOWS FOR MULTIELEMENT AIRFOILS	5. TYPE OF REPORT & PERIOD COVERED: Final Report, Dec 1974 - Jan 1977
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Charles L. Keller	8. CONTRACT OR GRANT NUMBER(s): In-house research
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Applied Mathematics Group/FBRD, Air Force Flight Dynamics Laboratory (AFSC), Wright-Patterson AFB, Ohio 45433	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: DoD Element 61102F, 2304N102
11. CONTROLLING OFFICE NAME AND ADDRESS: Analysis & Optimization Branch, Structural Mechanics Division, Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio 45433	12. REPORT DATE: April 1977
	13. NUMBER OF PAGES: 113
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): 113 p.	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

2304, N 7071

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

N1

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Two Dimensional Incompressible Flow
Fredholm Integral Equation of Second Kind
Multielement Airfoil Profile
Logarithmic Potential

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The potential or the stream function of the flow is expressed by potential theoretic methods. Once either of these functions is determined in principle, the velocity along the airfoil profile is determined and the pressure coefficient can be obtained using Bernoulli's equation.

The velocity along the profile either depends upon the solution to an integral equation or is determined as the solution to an integral equation. To insure well conditioned integral equations the profile elements in the physical plane are transformed (using a Karman Trefftz-like transformation)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

012 070

to an auxiliary near circle plane. In the near circle plane the integrands can be expressed in terms of very smooth functions and consequently the integrals can be evaluated with high accuracy by the trapezoidal rule. Most current methods use panel methods in the numerical evaluation of the integrals.

The present method is chosen because it is independent of the number of profile elements. In contrast procedures based on conformal mapping methods break down completely when the number of profile elements exceeds two.

FOREWORD

This report describes work performed in the Aerospace Research Laboratories and in the Air Force Flight Dynamics Laboratory under the Defense Research Science Program, Project 7071, Research in Applied Mathematics. This is a final report on work carried out between December 1974 and January 1977.

The author wishes to express his thanks to Dr. Karl G. Guderley, now retired, for his many helpful suggestions and encouragement. He also thanks Mrs. Rhonda Baab for her assistance in preparing the draft copy.

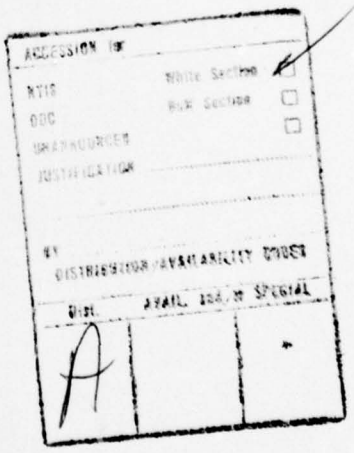


TABLE OF CONTENTS

Section		Page
I	Introduction	1
II	The Integral Equations	9
III	Numerical Approximations	21
IV	Discussions and Conclusions	31
Appendix		
A	Logarithmic Potentials	47
B	The Karman Trefftz-like Transportation	55
C	Numerical Determination of the Densities	61
D	Numerical Determination of the Derivatives of the Densities	63
E	Numerical Determination of the Tangential Velocities	65
F	Curve Fitting and Smoothing	67
G	The Computer Programs and Listings	71
References		110

PRECEDING PAGE BLANK-NOT FILMED

LIST OF ILLUSTRATIONS

Figure		Page
1	Pressure Coefficient for the "Smoothed" Profile	37
2	The "Smoothed" Profile	38
3	The Clark Y Profile	39
4	Cubic Spline Fit of the Near Circle Data and its Derivative	40
5	The Derivative of the Cubic Spline Fit of the Near Circle Data and its Smoothed Replacement	41
6	Graph of the Imaginary Part of $z + 0.2/(1 + z) + 0.2/(1 - z) = 0$	42
7	The Angles at the Trailing Edge	43
8	The Image of the Clark Y Profile Data under the Karman Trefftz-like Mapping	44
9	Mapping of Two Parallel Slits into Two Near Circles	45
10	Biplane Profile Data Transformed into Two Near Circles	46

SECTION I

INTRODUCTION

A basic problem in the two dimensional flow of an ideal fluid is the determination of the stationary flow around a body or bodies. The flow field is completely specified by either the velocity potential function $\Phi(x,y)$ or the stream function $\Psi(x,y)$. If either is known the velocity field can be determined, the pressure coefficient can be calculated from Bernoulli's equation and the forces acting on the body or bodies can be evaluated.

Both the velocity potential function $\Phi(x,y)$ and the stream function $\Psi(x,y)$ satisfy Laplace's equation at all points exterior to the body or bodies. The normal derivative of $\Phi(x,y)$ vanishes on the surface of the bodies and the stream function $\Psi(x,y)$ is a constant on the surfaces. The gradients of both functions tend to finite limits at infinity. These properties characterize the functions $\Phi(x,y)$ and $\Psi(x,y)$. Thus the basic problem reduces to that of determining a solution of Laplace's equation in the region exterior to one or more contours and satisfying conditions prescribed in advance (boundary conditions) on these contours and at infinity.

Functions which satisfy Laplace's equation are called harmonic functions. In potential theory the functions of an extensive class of harmonic functions are expressed as either the logarithmic potential of a single layer of density k(s) or as the logarithmic potential of a double layer of moment (density) m(s) distributed over the contours.

In this report we use each representation (separately) in an expression for the desired solution of Laplace's equation. The solution for the basic problem, is obtained in principle when the density function is determined. The density function is the solution of an integral equation.

The velocity tangent to the airfoil may be considered as the end product, for the determination of the pressure coefficient is routine after the tangential velocity is known. Thus in SECTION II we express, in turn, the velocity potential function $\Phi(x,y)$ and the stream function $\Psi(x,y)$ as a linear combination of certain simple harmonic functions and the logarithmic potential of a single and double layer respectively. From these representations of the velocity potential and the stream function we determine expressions for the velocity tangent to the contour. We see there that the tangential velocity can be obtained conveniently in two different ways.

The tangential velocity is determined as the solution to a Fredholm integral equation of the second kind or depends upon the solution to such an equation. In SECTION III we consider the reduction of these integral equations to systems of linear equations and give the reasons for our preference of method for achieving this reduction. In SECTION IV we describe and discuss our numerical experiments and give our conclusions.

The methods described in this report are integral equation methods. Integral equation methods have been carried out successfully in the past. Indeed, practically all methods, even conformal mapping

methods, have an integral equation in their development. There are, however, some variants in the present report which we believe are worth notice.

When the contour is a circle the kernel in the integral equation reduces to a constant. In this case the correct solution of the inhomogeneous problem is obtained in one iteration step, the reason being that here all eigenvalues of the kernel are zero except for one which is known to be +1. This suggests that the problem for a nearly circular contour also has a favorable distribution of eigenvalues which makes a rapidly converging iteration process feasible.

Speaking in more physical terms, one can say that by mapping a profile onto a contour which is nearly circular, a near circle, one suppresses the interference between the upper and lower sides which is generated by the form in which the solution is expressed (superposition of sources or vortices). Therefore, the first step of the Theodorsen method, namely, transformation of the profile into a near circle, is carried over in the present work. The solution to the potential problem is then computed for the near circle.

The benefits derived from the transformation into a near circle are closely related. One obtains a contour which may be considered an analytic contour. The integrand can be represented by a very smooth periodic function. The integral can then be evaluated very accurately by one of the simplest of all quadrature formulas, namely, the trapezoidal rule. The suppression of the interference between the lower and upper sides makes greater accuracy possible and reduces

the number of quadrature points required to achieve a given accuracy. The resulting equations may be solved by iteration. Lastly, the Kutta condition may be applied directly to the trailing edge.

This raises the question whether each element of a multielement profile can be transformed into a near circle. The results of [1] and our own experience with the Karman Trefftz-like mapping, described in Appendix B, leads us to believe that multielement profiles can be transformed into near circles rather well. Thus the interference between the upper and lower sides of a profile element can usually be suppressed, but the interference between portions of elements close to one another may still cause difficulties.

In many cases the integral equation formulation is either passed over quickly or bypassed by formulating an approximate expression for the tangential velocity directly from physical considerations. The author believes that it is preferable to formulate the integral equation explicitly before approximations are introduced for then all approximations are made at once, and it might be possible to achieve high accuracy without much additional effort.

For instance, in the present case one must make certain assumptions when evaluating numerically the integral in the integral equation. Frequently, the assumption is made that the density is approximated by piecewise linear functions. From a mathematical point of view the idea of interpolation by trigonometric polynomials is attractive and we use it in the present approach. (As far as the computations are concerned this interpolation need not be carried out.)

Obviously this is a much better representation of the density function than an approximation by piecewise linear functions. With this approximation the integrals to be evaluated have very smooth integrands and the trapezoidal rule gives much better results than practically all higher order methods. The computation to be carried out is extremely simple, actually the same as an approximation of the integrands by piecewise linear functions. But the rationale given shows that great accuracy can be achieved and sample calculations bear this out. These refinements are perhaps not possible, or in any case, not clearly recognizeable if one introduces approximations at an earlier stage.

If one has a single profile then a conformal mapping of the profile on a circle is practical because a solution for the potential field around a circle is known. If one proceeds to a bielement airfoil then a mapping of the flow field on the inside of a ring is possible, in principle, and the solution in this normalized form is again known, but only in principle. The procedure for the bielement profile is closely connected with the use of elliptic functions which are not always available in a computer. If an airfoil has more than two elements then the idea of conformal mapping onto a standard region breaks down completely.

In contrast the integral equation method can be readily extended to a multielement section. It must be admitted that for the integral equation method one must solve the integral equation several times to allow for angle of attack and circulation. However, the kernels

of the integral equations do not change in this procedure, only the right hand sides.

In the first example treated, the author started with the coordinates for profile points as given in the paper by Theodorsen [2]. These profile points were transformed into a near circle. To satisfy the condition that the normal velocity vanish, one must obtain the direction of the normal to the near circle. For this purpose an interpolation by cubic splines of the nearly circular contour was used. The graph of the derivatives of the spline interpolating polynomials was ragged. This raggedness is reflected in the pressure coefficient. One might say that the smoothness of the pressure coefficient reflects the smoothness (or lack thereof) of the curve of tangential directions.

If the profile is given merely by its coordinates, then one deals with an ill-posed problem, for the steps which lie between the data and the resulting pressures include a differentiation. This means that all high accuracy solutions of the problem are bound to give poor and ragged pressure coefficients whenever the initial data are poor. The author rejected the idea of a posteriori smoothing of the pressure coefficient in favor of a step in which the original profile data are smoothed. Such a step is a necessary preliminary to the determination of the pressure coefficient.

In this paper some smoothing has been carried out in the near circle plane and after the smoothing the profile coordinates have been recomputed so that one knows exactly which problem has finally

been solved. Actually, the choice of the profile and the smoothing procedure is not a mathematical, but rather an engineering decision. The graph of the pressure coefficient, Figure 1, illustrates what can be achieved with a very simple smoothing process.

We have noted above the benefits resulting from transforming profile elements into near circles. The use of such preliminary mappings is restricted to the two dimensional case. Thus the methods proposed here for determining the pressure coefficient on airfoil profile elements are limited to the two dimensional case.

SECTION II

THE INTEGRAL EQUATIONS

From a practical point of view the primary objective is the pressure distribution on an airfoil profile. The pressure coefficient can be obtained from Bernoulli's equation when the velocity on the profile is known. In this section we obtain expressions for the theoretical velocity at a point on the profile. We examine four possible ways for developing expressions for this theoretical velocity. It is shown that there are two essentially different expressions for the velocity.

The theoretical velocity is the solution of an integral equation or depends upon the solution to an integral equation. In either event the properties of the integral equations and their kernels are well established and can be used to advantage for the determination of the numerical values of the velocity at the profile. The two expressions for the theoretical velocity are examined for possible advantages and disadvantages.

The notation and conventions used throughout this paper are given in this section. The pertinent properties of logarithmic potential functions used in the following developments are given in Appendix A. Proofs and additional details on logarithmic potential functions may be found in [3,4,5]. The reader who is acquainted with the integral equation methods for determining the flow field around a body or bodies may skip this section.

PRECEDING PAGE BLANK-NOT FILMED

Profile contours are usually quite slender, at least in the vicinity of the trailing edge where there is a sharp corner. In order to obtain better conditioned integral equations which may be solved by iteration the profile contour is mapped onto a nearly circular contour. The equation for this nearly circular contour may be considered analytic. The mapping function is conformal at all points on and exterior to the profile except the trailing edge. Such a mapping function is described in Appendix B. If the airfoil profile consists of several elements, each element is mapped into a near circle by repeated application of the mapping function. The ability to solve iteratively is not crucial for a single element profile but could be advantageous for multielement profiles where the number of equations and unknowns may be quite large.

Let C_i denote a simple closed and nearly circular contour. The coordinates of a point on C_i are given as functions of the length of arc t measured from some convenient point on C_i. (There should be no difficulty in writing simply t rather than t_i for the length of arc on the i^{th} contour C_i.) Hence let the equations for C_i be

$$\xi_i = \xi_i(t)$$
$$\eta_i = \eta_i(t) \text{ for } 0 \leq t \leq L_i$$

where L_i denotes the total length of C_i. It is assumed that the point $(\xi_i(t),\eta_i(t))$ traces C_i in a counterclockwise direction with increasing t. The unit tangent vector τ_i to C_i has components $(d\xi_i/dt, d\eta_i/dt)$ and the unit outer normal has components $(d\eta_i/dt,$

$-d\xi_i/dt$). The functions $\xi_i(t)$ and $\eta_i(t)$ are smooth periodic functions of t. Lastly, let $(\tilde{x}_i, \tilde{y}_i)$ denote the coordinates (or approximations to the coordinates) of the centroid of C_i.

Let r denote the distance of the point (x, y) from the point $(\xi_i(t), \eta_i(t))$, $r = \sqrt{[(x - \xi_i)^2 + (y - \eta_i)^2]}$. A real valued function f(r) is then a function of the four variables x, y, ξ, η. In forming directional derivatives of f(r) it is helpful if one knows, without depending upon context, which quantities are regarded as constants. Thus we shall write for the derivative of f(r) in the direction n, $\partial f/\partial n_1$ when (ξ, η) is fixed and $\partial f/\partial n_2$ when (x, y) is fixed.

The velocity field could be determined if we had either the velocity potential function $\Phi(x, y)$ or the stream function $\Psi(x, y)$. We consider first the determination of $\Phi(x, y)$. The properties which characterize $\Phi(x, y)$ are: $\Phi(x, y)$ is harmonic outside C_i, the normal derivative vanishes on C_i and the derivatives $\partial\Phi/\partial x$ and $\partial\Phi/\partial y$ tend to well defined limits as the quantity $x^2 + y^2$ increases without bound.

It is natural to represent $\Phi(x, y)$ as a linear combination of functions which have properties desired for $\Phi(x, y)$. Thus we may write $\Phi(x, y)$ either as

$$\Phi(x, y) = \phi(x, y) + \sum_i \int_{C_i} k(t)[\log r]\, dt \qquad (1)$$

or

$$\Phi(x, y) = \phi(x, y) + \sum_i \int_{C_i} m(t)(\partial[\log r]/\partial n_2)\, dt \qquad (2)$$

where i goes from 1 to N_c, N_c is the number of contours C_i. Here $\phi(x, y)$ denotes a harmonic function which provides certain properties

desired for $\Phi(x, y)$ and the integral provides the capability for satisfying the boundary conditions. In Eq. (1) the integral represents the contribution to $\Phi(x, y)$ of a continuous layer of sources (or sinks) of strength $k(t)$ distributed along the contour C_i, while in Eq. (2) the integral represents the contribution due to a continuous layer of doublets of strength or moment $m(t)$.

We consider first the representation of $\Phi(x, y)$ as given by Eq. (1). We have

$$\partial\Phi(x, y)/\partial n = \partial\phi(x,y)/\partial n + \sum_i \int_{C_i} k(t)(\partial[\log r]/\partial n_1)\, dt \quad (3)$$

Let s_j satisfy the condition $0 \leq s_j \leq L_j$. Then the point $x = \xi_j(s_j)$, $y = \eta_j(s_j)$ lies on the contour C_j. We will write $\Phi(s_j)$ for the value of $\Phi(x, y)$ at the point $(\xi_j(s_j), \eta_j(s_j))$. This convention will be used for denoting the value of a function at a particular point on a particular contour.

We obtain from Eq. (3) by Eq. (A12)

$$-\partial\phi(s_j)/\partial n = \pi k(s_j) + \sum_i \int_{C_i} k(t)(\partial[\log r]/\partial n_1)\, dt \quad (4)$$

since the $\partial\Phi_e(s_j)/\partial n = 0$. The Eq. (4) is a Fredholm integral equation of the second kind with inhomogeneous term, $-\partial\Phi(s_j)/\partial n$. The Eq. (4) must be solved for the density $k(t)$ on the contour C_i. The integrands are well defined and continuous except for the integral around C_j, where the integrand is indeterminate when $t = s_j$. The value of this integrand at $t = s_j$ is given in Appendix C.

Once the density $k(t)$ is determined the velocity potential func-

tion $\Phi(x, y)$ is known. The velocity at a point $(\xi_j(s_j), \eta_j(s_j))$ on the contour C_j is given by $\partial\Phi(s_j)/\partial\tau$, the value of the derivative of $\Phi(x, y)$ in the direction of the tangent vector to C_j at $(\xi_j(s_j), \eta_j(s_j))$. We have

$$\partial\Phi(s_j)/\partial\tau = \partial\phi(s_j)/\partial\tau + \sum_i \int_{C_i} k(t)(\partial[\log r]/\partial\tau_1)\, dt \qquad (5)$$

Thus the evaluation of a line integral is required for each point at which the value of the velocity is desired.

The integrands occurring in Eq. (5) are well defined and continuous except for the integral around the contour C_j, where the integrand has a first order singularity for $t = s_j$. In this case one must take for the value of the integral its Cauchy principal value. In the numerical evaluation of the integral around the contour C_j we use the derivative of the density $k(t)$ with respect to the integration variable. The determination of these derivatives is discussed in Appendix D. The numerical evaluation of the singular integral occurring in Eq. (5) is discussed in Appendix E.

For our particular problem we take

$$\phi(x, y) = Ux + Vy + \sum_\ell \omega_\ell \arctan[(y - \tilde{y}_\ell)/(x - \tilde{x}_\ell)] \qquad (6)$$

where ℓ runs through the values $1, 2, \ldots, N_c$. The first two terms $Ux + Vy$ enable $\Phi(x, y)$ to satisfy the required conditions at a great distance from the contours C_i. The terms following the summation sign provide the circulation. We write the density $k(t)$ as a sum also

$$k(t) = -Uk_1(t) + Vk_2(t) + \sum_\ell \omega_\ell k_{3\ell}(t) \qquad (7)$$

When the value of $\partial\phi(s_j)/\partial n$ as obtained from Eq. (6) and the value of the density, Eq. (7) are substituted in Eq. (4) one obtains $N_c + 2$ integral equations for the densities $k_1(t)$, $k_2(t)$ and $k_{3\ell}(t)$, $\ell = 1, \ldots, N_c$.

It is clear that these densities need to be determined only once. Then for any prescribed undisturbed flow at infinity (or any angle of attack) and prescribed velocity at a point on each C_i, appropriate values for the coefficients U, V and ω_ℓ, $\ell = 1, \ldots, N_c$, will give a potential function $\Phi(x, y)$, Eq. (1) which satisfies the prescribed conditions. In particular the coefficients ω_ℓ can be determined to enforce the Kutta condition at the image of the trailing edge on each contour C_i. The ω_ℓ are determined by a system of linear equations. The coefficients U and V must take into account the distortion at infinity (if any) arising from mapping the profile elements into near circles.

Now consider Eq. (2),

$$\Phi(x, y) = \phi(x, y) + \sum_i \int_{C_i} m(t)(\partial[\log r]/\partial n_2)\, dt$$

Suppose $\phi(x, y)$ harmonic at points (x, y) on and inside the contour C_i, $i = 1, \ldots, N_c$. It follows, since then $\Phi(x, y)$ is harmonic inside C_i and $\partial\Phi/\partial n = 0$ by hypothesis that $\Phi(x, y)$ is constant inside C_i. Hence the limit $\Phi_i(s_j)$ of (x, y) as the point (x, y) tends to $(\xi_j(s_j), \eta_j(s_j))$ on C_j is a constant. Then by Eq. (A8) we have

$$\Phi_e(s_j) = \phi(s_j) - \pi m(s_j) + \sum_i \int_{C_i} m(t)(\partial[\log r]/\partial n_2)\, dt \qquad (8)$$

By Eq. (A6)

$$m(t) = (\text{constant} - \Phi_e(t))/2\pi \tag{9}$$

Substitute for m(t) in Eq. (8) its value as given by Eq. (9). Then using Eq. (A5) and the fact that

$$\int_{C_i} (\partial[\log r]/\partial n_2)\, dt = 0 \text{ if } (x, y) \text{ lies outside } C_i$$
$$= \pi \text{ if } (x, y) \text{ lies on } C_i.$$

Eq. (8) simplifies to

$$2\pi\phi(s_j) = \pi\Phi_e(s_j) + \sum_i \int_{C_i} \Phi_e(t)[\partial\beta/\partial t]\, dt \tag{10}$$

The integral equation, Eq. (10), could be solved for $\Phi_e(t)$ and in this way one could determine the potential function $\Phi(x, y)$. Instead, differentiate Eq. (10) with respect to s_j. After an integration by parts we have

$$2\pi\phi's(s_j) = \pi\partial\Phi_e/\partial s_j - \sum_i \int_{C_i} (\partial\Phi_e/\partial t)(\partial[\log r]/\partial n_1)\, dt \tag{11}$$

The quantity $|\partial\Phi_e/\partial s_j|$ is the speed of the flow past the body bounded by the contours C_i. Set $v(t) = \partial\Phi_e/\partial t$, then Eq. (11) can be rewritten as

$$2\pi\phi'(s_j) = \pi v(s_j) - \sum_i \int_{C_i} v(t)(\partial[\log r]/\partial n_1)\, dt \tag{12}$$

The Eq. (12) is of the same form as Eq. (A11). From properties of harmonic functions we conclude that $\phi'(s)$ satisfies the condition expressed by Eq. (A13). Hence Eq. (12) is solvable for v(t); however,

the solution v(t) is not unique. To any solution of Eq. (12) one may add a solution of the homogeneous equation. The Kutta condition is used to determine the solution of the homogeneous equation to be added to the solution of Eq. (12).

In the method associated with Eq. (1) one provides for circulation by means of the functions $\omega_\ell \arctan [(y - \tilde{y}_\ell)/(x - \tilde{x}_\ell)]$. The basic assumptions for the derivation of the methods associated with Eq. (2) did not permit the use of these functions. By contrast, the circulation is provided for in a natural way by the eigenfunctions associated with the eigenvalue π. We postpone further comparison until the possibilities for other method are exhausted.

The stream function $\Psi(x, y)$ is characterized by the properties: $\Psi(x, y)$ is harmonic at points (x, y) exterior to the contours C_i, $\Psi(x, y)$ is constant on C_i and the first order derivatives tend to finite limits as the quantity $x^2 + y^2$ increases without bound. We may attempt to represent $\Psi(x, y)$ in either the form

$$\Psi(x, y) = \psi(x, y) + \sum_i \int_{C_i} m(t)(\partial[\log r]/\partial n_2)\, dt \qquad (13)$$

or

$$\Psi(x, y) = \psi(x, y) + \sum_i \int_{C_i} k(t)[\log r]\, dt \qquad (14)$$

The tangential velocity to the contours C_i is not readily computed from $\Psi(x, v)$ as given by Eq. (13). For this reason we do not consider the representation of $\Psi(x, y)$ which contains the logarithmic potential of a double layer further.

Lastly, consider the stream function as given by Eq. (14). Since $\Psi(x, y)$ is constant along the contour C_i, the derivative of $\Psi(x, y)$ at a point on C_i in the direction of the tangent vector to C_i vanishes. Thus we have

$$\partial\Psi(s_j)/\partial s_j = 0 = \partial\psi(s_j)/\partial s_j + \sum_i \int_{C_i} k(t)(\partial[\log r]/\partial s_j)\, dt \qquad (15)$$

This is an integral equation of the first kind for $k(t)$. As we have already noted the integrand for the line integral around the j^{th} contour C_j has a first order singularity for $t = s_j$. The Eq. (15) is not readily solved.

The condition on the function values of $\Psi(x, y)$ can be exchanged for a condition on the normal derivative of $\Psi(x, y)$ on C_i. If $\psi(x, y)$ is harmonic inside the contours and continuous across the contours then $\Psi(x, y)$ as defined by Eq. (14) is harmonic inside the C_i also. Since $\Psi(x, y)$ is harmonic inside the C_i and constant on the C_i it follows from properties of harmonic functions that $\Psi(x, y)$ is constant inside the contours C_i. Hence $\partial\Psi_i(s_j)/\partial n = 0$ and we have by Eq. (A10)

$$k(t) = (1/2\pi)\partial\Psi_e(t)/\partial n \qquad (16)$$

replacing the density function by its value as given by Eq. (16) we have from Eq. (A12)

$$\partial\Psi_e(s_j)/\partial n = \partial\psi(s_j)/\partial n + (1/2)\partial\Psi_e(s_j)/\partial n + (1/2\pi)\sum_i \int_{C_i} (\partial\Psi_e(t)/\partial n)(\partial[\log r]/\partial n_1)\, dt$$

Setting $\partial\Psi_e(t)/\partial n = -v(t)$ and combining like terms we obtain

$$2\pi(\partial\psi(s_j)/\partial n) = -\pi v(s_j) + \sum_i \int_{C_i} v(t)(\partial[\log r]/\partial n_1)\, dt \quad (17)$$

These same equations for determining the velocity at the contour can be derived in yet another way from more elementary considerations. Namely, one can go from the knowledge of the velocity components due to a point source or induced by a point vortex at the origin to the velocity components due to a line distribution of sources or induced by a line distribution of vortices. Usually, however, in such an approach certain approximations are made and accepted at an early stage and, hence, the equations given above are not explicitly obtained.

The Eqs. (12) and (17) are essentially the same. Hence previous remarks concerning Eq. (12) apply equally well to Eq. (17). The derivation of the equation for the tangential velocity starting with Eq. (2) and concluding with Eq. (12) is often referred to as the method of doublets. This last derivation starting with Eq. (14) and leading to Eq. (17) is called the method of vortices or Prager's method [6]. We will not consider the method of doublets further. The first method which gave Eq. (4) for the density function and Eq. (5) for the tangential velocity we will call the method of sources.

In the preceding pages we considered two representations for the velocity potential and two representations for the stream function. These four different starting points led to two different equations or expressions for the velocity at the contour. Now we want to con-

sider the determination of the tangential velocities further. Our object here is to obtain an estimate of work involved using Eq. (17) and compare with a corresponding estimate of work using Eqs. (4) and (5).

A first impression is that the method of vortices is far superior to the method of sources, since the tangential velocity is obtained directly as the solution of the integral equation, Eq. (17). For the method of sources, on the other hand, the solution of the integral equation, Eq. (4) is just the first step. An integration is required to obtain the tangential velocity.

For the method of vortices, let us note that along with Eq. (17) we need to consider the homogeneous equation,

$$-\pi v_{3\ell}(s_j) + \sum_i \int_{C_i} v_{3\ell}(t)(\partial[\log r]/\partial n_1)\, dt = 0 \qquad (18)$$

We need N_c independent solutions $v_{3\ell}(t)$, $\ell = 1, \ldots, N_c$, of Eq. (18) in order to be able to enforce the Kutta condition at prescribed points on the contours C_i. Also in order to accomodate any given undisturbed flow at infinity write v(t) in Eq. (17) as

$$v(t) = Uv_1(t) + Vv_2(t) \qquad (19)$$

Then with $\psi(x, y) = Uy - Vx$ one obtains from Eq. (17) one equation of the same form as Eq. (17) for $v_1(t)$ and also for $v_2(t)$.

Hence for the method of vortices the integral equation, Eq. (17) has to be solved for two different inhomogeneous terms. The homogeneous equation, Eq. (18) is solved by iteration. Note that an

iteration is an integration. The number of iterates depends upon the tolerance which one sets. The total number of integrations required is then the sum of the number of iterates for the determination of the eigenfunctions, that is, the solutions of the homogeneous equation, Eq. (18).

For the method of sources the integral equation, Eq. (4), has to be solved for $N_c + 2$ different inhomogeneous terms. Next, $N_c + 2$ integrations have to be performed. In the method of vortices the integral equation does not have to be solved as many times as in the method of sources. This decided advantage is offset or diminished by the number of iterates required for the determination of the eigenfunctions.

We conclude that the method of vortices, in general, requires less computer resources than the method of sources. Each method was programmed so that the system of simultaneous equations obtained could be solved either directly or iteratively. When the number of contours N_c is small and high accuracy is desired the difference in cost is negligible. As the number of contours is increased we expect the method of vortices based on iterative procedures to exhibit a cost advantage.

It is clear that many different computer programs for the speed of the flow past the profiles are possible based on Eqs. (4) and (5) or Eqs. (17) and (18), respectively. The different programs are the result of different ways of numerically approximating and solving these equations. Four programs are listed in Appendix G. These programs are based on the numerical approximations and solution methods discussed in Section III.

SECTION III

NUMERICAL APPROXIMATIONS

In this section we describe the numerical procedures we use to obtain the tangential velocities. Then we mention briefly several alternative numerical procedures. In the previous section we found that whether one used a method based on sources or doublets or vortices an integral equation needed to be solved first. Note that this integral equation, before any modification, has the same kernel, regardless of method. Accordingly, different numerical methods for expressing the value of the integral account for some of the differences in solution procedure.

We assume that the contours C_i are very smooth. The expressions for the equations for the C_i may be considered analytic. Then the integrands of the integrals, Eqs. (4), (17) and (18) are smooth continuous periodic functions of t for each value of s_j. These integrals, which are of the form

$$\int_{C_i} k(t)(\partial[\log r]/\partial n_1)\, dt$$

are evaluated with high accuracy using the trapezodial rule and equally spaced increments of the integration variable [7].

Let $f(\theta)$ be a periodic function of θ of period 2π. Set

$$\Delta\theta = 2\pi/N, \text{ N an integer } \geq 1$$

$$\theta(I) = (I - 1)\Delta\theta \text{ for } I = 1, \ldots, N+1$$

and

$$f(I) = f(\theta(I)).$$

Then the value of the integral of $f(\theta)$ over one period is given approximately by the trapezoidal rule by the expression

$$\sum_{I=1}^{N} f(I)\Delta\theta \tag{20}$$

Now let $f(t)$ denote a function defined on the simple closed contours C_i. We found it convenient to express each contour C_i in polar coordinates with pole at (or near) the centroid of C_i. Then with $\Delta\theta(I)$ constant on C_i but not necessarily the same constant on different contours and with appropriate indexing one has the approximation

$$\sum_i \int_{C_i} f(t)\, dt = \sum_{I=1}^{NT} f(I)[dt(I)/d\theta]\Delta\theta(I) \tag{21}$$

where NT denotes the total number of stations on all the contours C_i. Thus the expression for the numerical value of a line integral around one or several contours is basically the same.

Expressing the values of the integrals in Eqs. (4), (17) and (18) for appropriate values of s_j by the trapezoidal rule one obtains systems of linear equations. Using the conventions just introduced the elements in the coefficient matrix A(I, J) of the system of equations stemming from Eq. (4) are, for $J \neq I$, given by the expression

$$\frac{(\xi(I)-\xi(J))(d\eta(I)/dt)-(\eta(I)-\eta(J))(d\xi(I)/dt}{(\xi(I) - \xi(J))^2 + (\eta(I) - \eta(J))^2} \left(\frac{dt(J)}{d\theta}\right) \Delta\theta(J) \tag{22}$$

and the diagonal elements are given by

$$\pi + \frac{(d^2\eta(I)/d\theta^2)(d\xi(I)/d\theta) - (d^2\xi(I)/d\theta^2)(d\eta(I)/d\theta)}{2[(d\xi(I)/d\theta)^2 + (d\eta(I)/d\theta)^2]} \Delta\theta(I) \tag{23}$$

The right hand sides as determined from Eq (6) is

$$(d\eta(I)/d\theta)/(dt(I)/d\theta),\ I = 1, \ldots, NT \tag{24}$$

$$(d\xi(I)/d\theta)/(dt(I)/d\theta),\ I = 1, \ldots, NT \tag{25}$$

and

$$\frac{(\xi(I) - \tilde{x}(\ell))(d\xi(I)/d\theta) + (\eta(I) - \tilde{y}(\ell))(d\eta(I)/d\theta)}{[(\xi(I) - \tilde{x}(\ell))^2 + (\eta(I) - \tilde{v}(\ell))^2](dt(I)/d\theta)} \tag{26}$$

for $I = 1, \ldots, NT$, $\ell = 1, \ldots, N_c$.

The off diagonal elements for the matrix A(I, J) associated with Eq. (18) are given by Eq. (22). The diagonal elements are given by Eq. (23) with the term π deleted. The eigenvectors associated with the eigenvalue π are determined by iteration. For the initial guess or starting value we take an eigenvector determined from an eigenfunction corresponding to the eigenvalue π of the adjoint kernel.

The set of functions defined on the contours C_i and which have the value 1 on one of the contours and 0 on all other contours is a complete set of eigenfunctions corresponding to the eigenvalue π of the adjoint kernel. Accordingly the vector with components equal to 1 at the stations on one contour and zero otherwise is taken as the initial value for the iteration process. For our test problem the maximum difference between the components of successive iterates was less than 10^{-8} after six to eight iterations.

Denote the components of the eigenvectors just determined by CC(I, J), $I = 1, \ldots, NT$ and $J = 1, \ldots, N_c$. Denote the components

of the eigenvectors defined by the known eigenfunctions of the adjoint kernel by CA(I,J), I + 1, ... , NT, J = 1, ... , N_c. One should note that the eigenvectors CA(I, J) are not eigenvectors of the transpose of the matrix A(I, J). Instead, they are the eigenvectors of the matrix in the linear system approximation, to the homogeneous integral equation with the adjoint of the kernel in Eq. (18). This difference between the eigenfunctions of the adjoint integral equation and the eigenfunctions determined by the transpose of the matrix A is the result of using the integration variable θ.

Once the integral equation is reduced to a system of equations one could forget the origin of the system of equations. We believe that it is preferable, in general, to use relations involving the original equations. In the present problem it is advantageous to proceed as though we were dealing directly with the integral equations rather then linear system approximations, since the eigenfunctions of the adjoint kernel are known whereas the eigenvectors of the transpose of the matrix A(I,J) are not known.

Once the eigenvectors of the matrix A(I,J) are determined we may regard the eigenfunctions for the integral equation, Eq. (18) as known. Since now we know the eigenfunctions corresponding to the eigenvalue π for the kernel of Eq. (17) and its adjoint we can modify Eq. (17) in the manner discussed in Appendix A, Eq. (A21). The entries for the matrix $A_M(I,J)$ in the linear system approximation of the integral equation with the modified kernel are, for $I \neq J$, given by the expression

$$-A(I, J) + \pi \sum_{K=1}^{N_c} CC(I, K)\ CA(I, K)(dt(J)/d\theta)\Delta\theta(J) \qquad (27)$$

and the diagonal elements are given by

$$-A(I, I) + \pi[1 + \sum_{K=1}^{N_c} CC(I, K)\ CA(I, K)(dt(I)/d\theta)\Delta\theta(I)] \qquad (28)$$

where the A(I,J) are the elements of the matrix associated with the integral equation, Eq. (18). One is assured that the matrix $A_M(I,J)$ is nonsingular. Hence the associated linear system can be solved by direct means.

The linear system associated with the matrix $A_M(I,J)$ has to be solved for two right hand sides. The components for these vectors are

$$2\pi(d\xi(I)/d\theta)/(dt(I)/d\theta),\ I = 1, \ldots, NT \qquad (29)$$

and

$$2\pi(d\eta(I)/d\theta)/(dt(I)/d\theta),\ I = 1, \ldots, NT \qquad (30)$$

The tangential derivative at a station on one of the contours C_i is then given as a linear combination of the eigenfunctions CC(I,J) and of the solutions corresponding to the right hand sides given by the expressions (29) and (30).

The system of linear equations obtained from the integral equations, either Eq. (4) or Eq. (17), is well suited to solving by iterative means. In vector matrix notation the linear system is of the form

$$(\lambda I \pm A)\ x = b \qquad (31)$$

where λ is the maximum eigenvalue of A. Note that the term πI is incorporated in the matrix defined by Eqs. (22) and (23). Divide Eq. (31) by λ and consider first the case with the plus sign in front of the matrix A. Thus Eq. (31) becomes $(I + A/\lambda)\, x = b/\lambda$ and a linear system of this form represents the integral equation, Eq. (4). Formally one has

$$x = b/\lambda - (A/\lambda)(b/\lambda) + (A/\lambda)^2(b/\lambda) - + \ldots \tag{32}$$

If the vector b has no component in space spanned by the eigenvectors corresponding to the eigenvalue λ the right-hand side of Eq. (32) will converge to a solution of Eq. (31). (We have assumed that $-\lambda$ is not an eigenvalue of the matrix A.)

To insure that the right-hand side has no component in the space spanned by the eigenvectors corresponding to λ write x as

$$x = \tilde{x} + \alpha(b/\lambda) \tag{33}$$

Then we can exchange the system of equations for the unknown x for the following system of equations for the unknown $\tilde{x}$

$$(I + A/\lambda)\tilde{x} = b/\lambda - \alpha(b/\lambda) - \alpha(A/\lambda)(b/\lambda) \tag{34}$$

Observe that if $\alpha = 1/2$ the right-hand side of Eq. (34) has no component in the space spanned by the eigenvectors of A corresponding to the eigenvalue λ. Thus, replacing (b/λ) in Eq. (32) by the right-hand side of Eq. (34) gives $\tilde{x}$ and then Eq. (33) gives x.

The case of the minus sign in Eq. (31) is similar, up to a

point, to that of the plus sign. The equation $(I - A/\lambda)\, x = b/\lambda$ represents the integral equation Eq. (17). As for the plus sign, this equation has solutions only if the right-hand side is orthogonal to every eigenfunction corresponding to λ of the adjoint kernel. Hence, if the right-hand side b/λ has no component in the space spanned by the eigenfunctions of A corresponding to λ, the expansion

$$x = (b/\lambda) + (A/\lambda)(b/\lambda) + (A/\lambda)^2(b/\lambda) + \ldots \qquad (35)$$

should converge.

If the right-hand side b/λ has a component, in the space of the eigenvectors of the matrix A, sufficiently large to affect the convergence to x in Eq. (35) the procedure given by Eqs. (33) and (34) cannot be used to remove this component. Considerable additional computations are required. First one would have to obtain the eigenvectors of A^T. Next, these eigenvectors of A^T would be used to obtain a system of linear equations for the component of b/λ in the space spanned by the eigenvectors of A. Then this system of equations would have to be solved for this component.

The tangential velocities can be obtained from either the velocity potential function $\Phi(x,y)$ or the stream function $\Psi(x,y)$. Using a continuous distribution of sources in the representation for the potential function led to an integral equation which had to be solved in order to determine the tangential velocities. Using a continuous distribution of doublets to represent the potential function or a continuous distribution of vortices to represent the stream function

led to an essentially different integral equation. Thus there are two basic approaches in the determination of the tangential velocities.

In the above pages of this section we described two ways for determining a numerical solution of each integral equation. In all cases the value of the integral was expressed by the trapezoidal rule with equal increments of the integration variable. Every different method for expressing the value of the integral in either integral equations may be considered a different procedure for determining the tangential velocities.

Instead of using continuous distributions of sources, doublets or vortices one might, for example, approximate the potential function with finitely many sources, with strengths to be determined, distributed along the contours C_i. In order to determine the unknown strengths the expression for the normal velocity at points on the C_i midway between two sources is equated to zero. In this way one obtains a system of equations for the source strength. One could proceed in an analogous way using either vortices or doublets, [8, pp. 195-6]. Using a discrete rather than a continuous distribution gives the appearance of avoiding the integral equation and the problem of evaluating the integral associated therewith. However, one pays a price for this computational simplicity. The quantities of interest are the velocities at points along the contours. The lumping of the singularities into discrete points on the boundaries C_i adversely affects the accuracy of this method in the vicinity of the

C_i. It is clear that at points where the sources (vortices and doublets) are located the velocities are infinite.

Another method for evaluating numerically the line integrals occurring in Eqs. (4) and (17) is the "panel" method. In this method the contours C_i are replaced by polygonal contours and the density $k(t)$ (method of sources) or the tangential velocity $v(t)$ (method of vortices) is taken as constant on each side of the polygonal contours. (Different constant on different sides, of course.) Instead of constant functions on the sides of the polygonal contours one can use linear functions. Thus for the side of length ℓ between the Ith and the $(I + 1)$ th stations take

$$k(t) = k(I) + [k(I + 1) - k(I)](t/\ell)$$

For $0 \leq t \leq \ell$

The resulting line integrals along the sides of the polygonal contours are well defined and finite for all points except the end points of the sides of the polygonal contours.

The details for evaluation of such an integral with uniform density are given in [8, pp. 191-3] for the special case of a segment of the x-axis. Using a rotation of the xy-plane one can reduce the evaluation of the integral over an arbitrarily oriented side to this special case.

Details (which include the case of a constant density) for the case of a density linear on a side are given in [9, pp. 21-32]. One can give expressions for the velocity components directly in

terms of sums of integrals of the form being considered here. In this way one appears to have avoided the integral equations of Section II. In actuality one has gone directly to an approximate solution of one of the integral equations. However, in the process of going directly to an approximate solution it may not be clear what approximations were made. This method of evaluating the integrals in Eqs. (4) and (17) or for deriving an approximate solution is attractive because the values of the integrals are given in terms of readily evaluated functions. One avoids the indeterminate case occurring in the use of the trapezoidal rule. The determination of the tangential velocities by the panel method involves the evaluation of essentially the same integrals as in the determination of the densities. These attractive features are offset by the fact that this method of evaluating the integrals requires a rather large number of stations on the contours C_i to achieve a given accuracy. Application of the Kutta condition requires a modified treatment.

SECTION IV

DISCUSSIONS AND CONCLUSIONS

The procedures described in Section III were applied to the data for a single element profile given by Theodorsen [2]. Figure 1 is the graph of the pressure coefficient P using 40 points around the near circle contour. (All the figures in this report were machine drawn and not retouched.) The appearance of Figure 1 could be improved by computing more values of the pressure coefficient in the vicinity of the leading edge.

The present results and those of Theodorsen are fairly close. This shows mainly, that the program is trustworthy. As no exact results are available, one cannot say that one method gives greater accuracy than the other. Actually both methods can be perfected to give a high degree of accuracy. The present method is advocated because it extends readily to multielement profiles rather than because of its inherent accuracy.

The distribution of pressures shown by Figure 1 has been determined for the profile, Figure 2. This profile is obtained from the original profile coordinates Figure 3 by a smoothing process. One observes that the "smoothed" profile, Figure 2, is thicker than the original profile, Figure 3. It is, of course, clear to the reader that this thickening is a result of an imperfect smoothing process and not of the transformation to the near circle. A smoothing process is needed because pressure results will reflect irregularities

in the profile slope. Such irregularities occur rather easily if the profile is described by its coordinates.

In Figure 4 we give graphs of the quantities R-1 versus θ and $dR/d\theta$ versus θ, where R is the near circle radius before smoothing. One observes that the graph of $dR/d\theta$ is rather wavy. Theory asserts and computational experience shows that a high accuracy method will produce similar irregularities in the pressure coefficient. Frequently the irregularities occurring in the pressure coefficient are removed rather arbitrarily either on the drawing board or by truncating the expression for the pressure coefficient. Thus cruder methods may appear better because they are less likely to pick up irregularities in the profile data.

In Figure 5 we give graphs of $dR/d\theta$ and smoothed values of $dR/d\theta$. From the smoothed values of $dR/d\theta$ we redetermined R, and the profile shown in Figure 2. From this revised data we determined the pressure coefficient shown in Figure 1. We feel that the appearance of Figure 1 implies that this is the proper order of proceeding. We can assert that the data for the profile of Figure 2 produced the values for the pressure coefficient exhibited in Figure 1. Admittedly, from a practical point of view, the differences between smoothing the original data and the arbitrary smoothing of the final results in most cases will not be significant.

Our objective in applying a smoothing process was a pressure coefficient graph, Figure 1, with no undue irregularities or waviness. We wanted to illustrate what a preliminary smoothing could

accomplish. Resorting to a smoothing process is an admission of some disatisfaction with the original data or of the fitting process or both. Thus one would expect the resulting profile after smoothing the derivatives to differ some from the original. The extent of the difference which is acceptable between the original profile and the profile after smoothing the derivatives is an engineering decision. It seems clear that, in general, one should be able to determine a smoothing for which the resulting profile is acceptable.

Our next objective is a test of the accuracy of the methods we proposed for solving the integral equations and computing the tangential velocities. From the above discussion it is clear that a practical problem is not well suited to this task. For this purpose we considered the complex potential function

$$F(z) = z + 0.2/(1-z) + 0.2/(1+z) \tag{36}$$

This function describes a flow around two nearly circular obstacles. The flow at infinity is parallel to the x-axis and of unit speed. A problem of this kind would be encountered in the determination of the distribution of the pressure coefficient on biplane profiles.

The "profile" data was determined from the equation

$$\Psi(x,y) = \text{Im}\,[F(z)] = 0 \tag{37}$$

That is, we determined 80 points, 40 around each contour, which satisfied Eq. (37). Figure 6 is the graph of $\Psi(x,y) = 0$. The "true

values" of the tangential velocities at these 80 points can be determined from either Re[F(z)] or Im[F(z)], Eq. (36). The maximum difference between the tangential velocities determined by the numerical procedures described in Section III and the true values was less than 0.5 E-08.

The maximum difference between the true values and the computed values using only 20 points, instead of 40, around each contour was less than 0.5 E-08 also. The maximum differences using 20 points was larger than the corresponding maximum difference using 40 points, in general. Normally, one would expect that doubling the number of points would produce a greater decrease in the maximum difference. We believe that limitations in the accuracy of the initial data accounts for the present results.

We have given two basic ways for computing tangential velocities. In each of these procedures we had to solve a system of linear equations (several times, in fact). Utilizing some special properties of this system of equations we are able to solve this system of equations iteratively as well as directly. Thus we may say that we have four ways for computing tangential velocities. All four ways are equally accurate. The method based on a vortex distribution (Prager's method) required slightly less central processing time than did the method of sources. With 40 points per contour iterative methods used less central processing time than direct methods. For 20 points per contour the direct methods used less central processing time.

Using a source distribution and solving the system of linear equations directly took 1.129 sec. of central processing time. With a vortex distribution and solving iteratively 1.010 sec. of central processing time was used. These times are for the case of 40 points per contour. This central processing time difference is not significant. We expect that with a greater number of profile elements the method using a vortex distribution and solving the linear systems iteratively will improve its advantage over the other three methods. We note also that solving a system of equations iteratively is more flexible than solving the same system directly, in the sense that the solving process can be shortened by reducing the accuracy requirement.

Thus we conclude that all four methods proposed for the task of providing the basic integral equation, solving this equation and determining the tangential velocities in a transformed plane are good and very accurate. The method based on a vortex distribution and which solves the linear system of equations iteratively has a slight cost advantage.

As we have mentioned the complete procedure requires some additional steps: First, the mapping of the profile data from the physical plane to the plane of near circles; secondly, the fitting of the near circle data and smoothing, Appendix F. As a result of these two steps one removes the difficulties associated with the sharp trailing edge and the leading edge. One has to deal with very smooth functions. This simplifies solving the associated integral equation

and enables one to obtain a very accurate solution from relatively few quadrature points. The Kutta condition can be invoked at the trailing edge in a straight forward manner.

The disadvantages of these two steps are clear. In general, they are not fully automated. In particular, the smoothing may require some drawing and reading of points with a digitizer. On the other hand, for example, the panel methods which bypass these first two steps require considerably more data points, especially along the leading edge. Some times other special precautions have to be taken to avoid ill conditioned or singular coefficient matrices in the associated systems of linear equations.

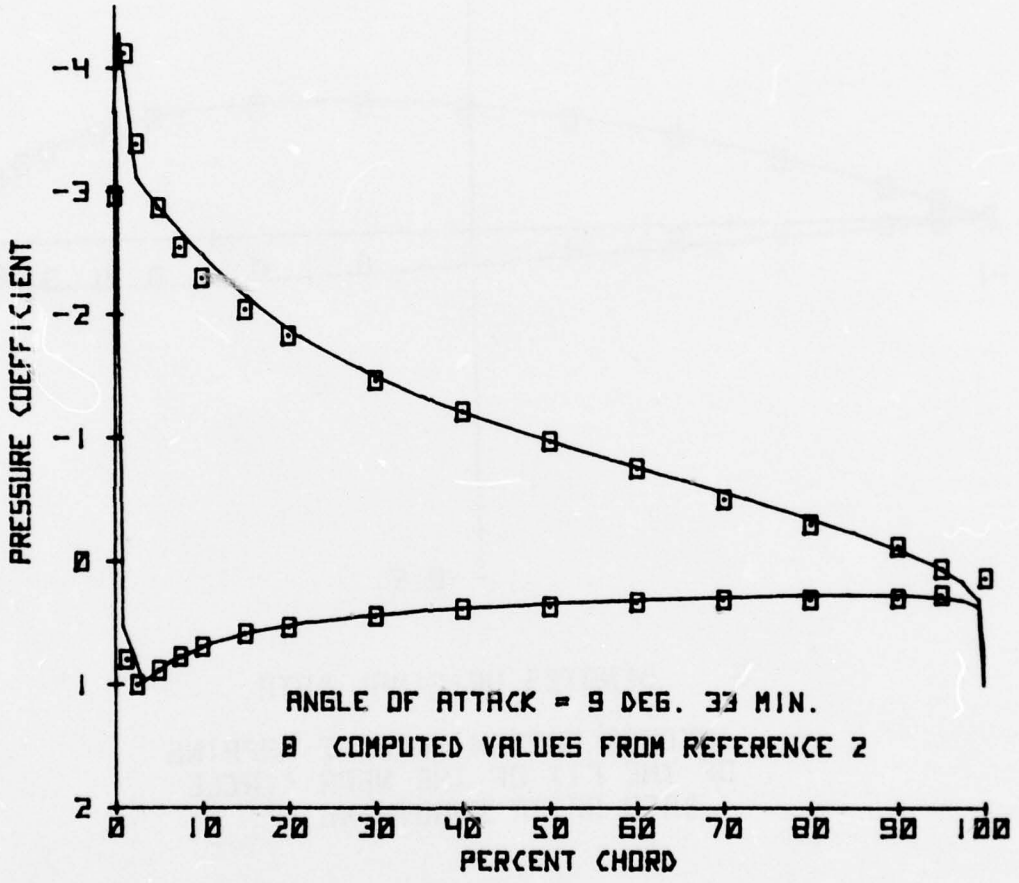
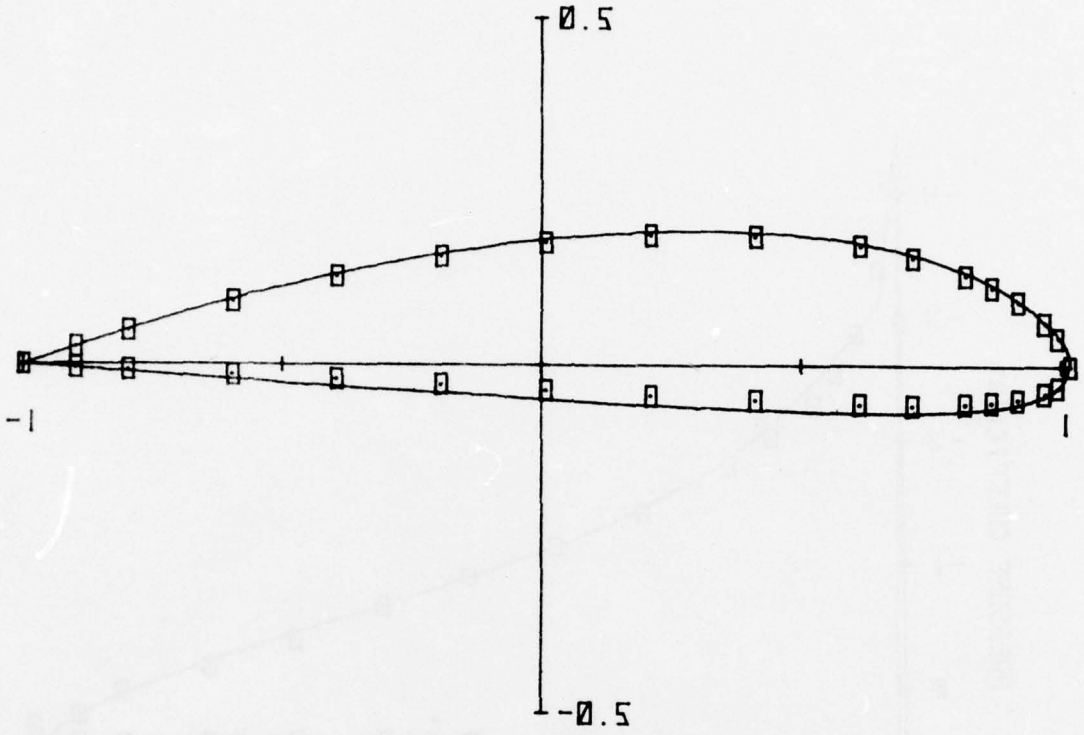


Figure 1.

Pressure Coefficient for the "Smoothed" Profile



⊡ DENOTES ORIGINAL DATA.

—— INVERSE KARMAN TREFFTZ MAPPING OF THE FIT OF THE NEAR CIRCLE DATA AFTER SMOOTHING.

Figure 2.

The "Smoothed" Profile

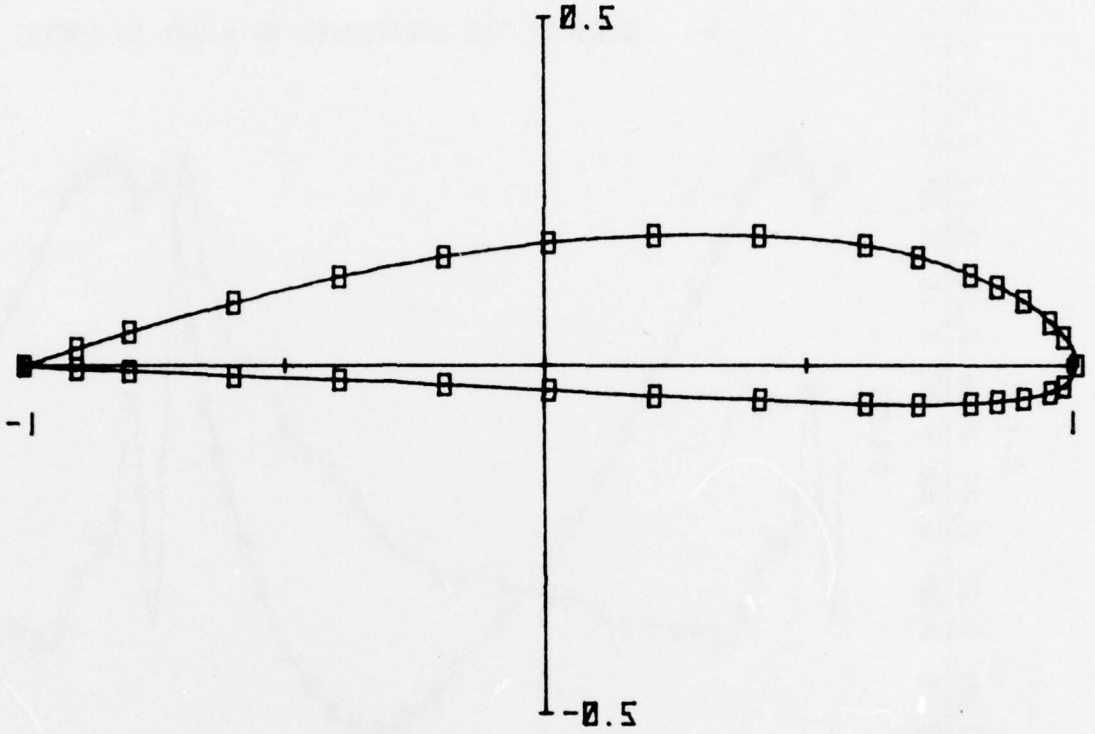


Figure 3.

The Clark Y Profile

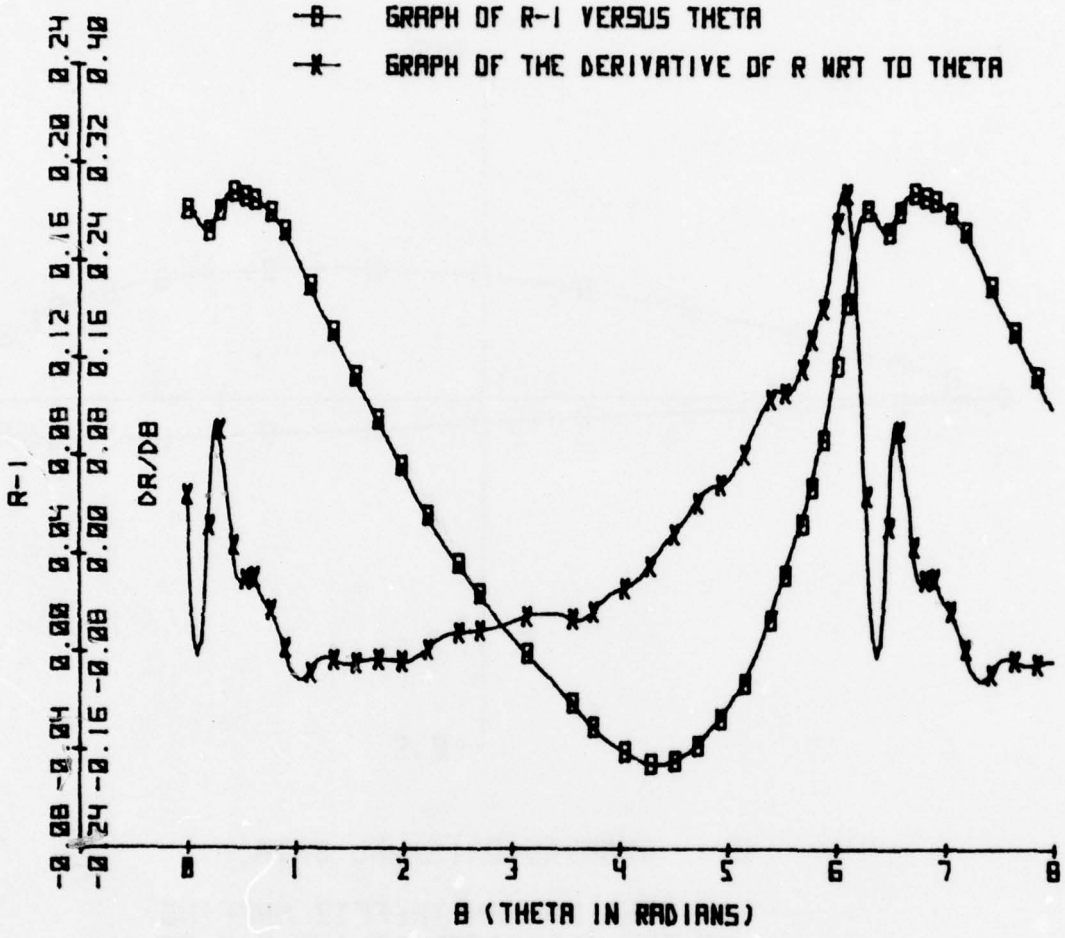


Figure 4.

Cubic Spline Fit of the Near Circle Data and Its Derivative

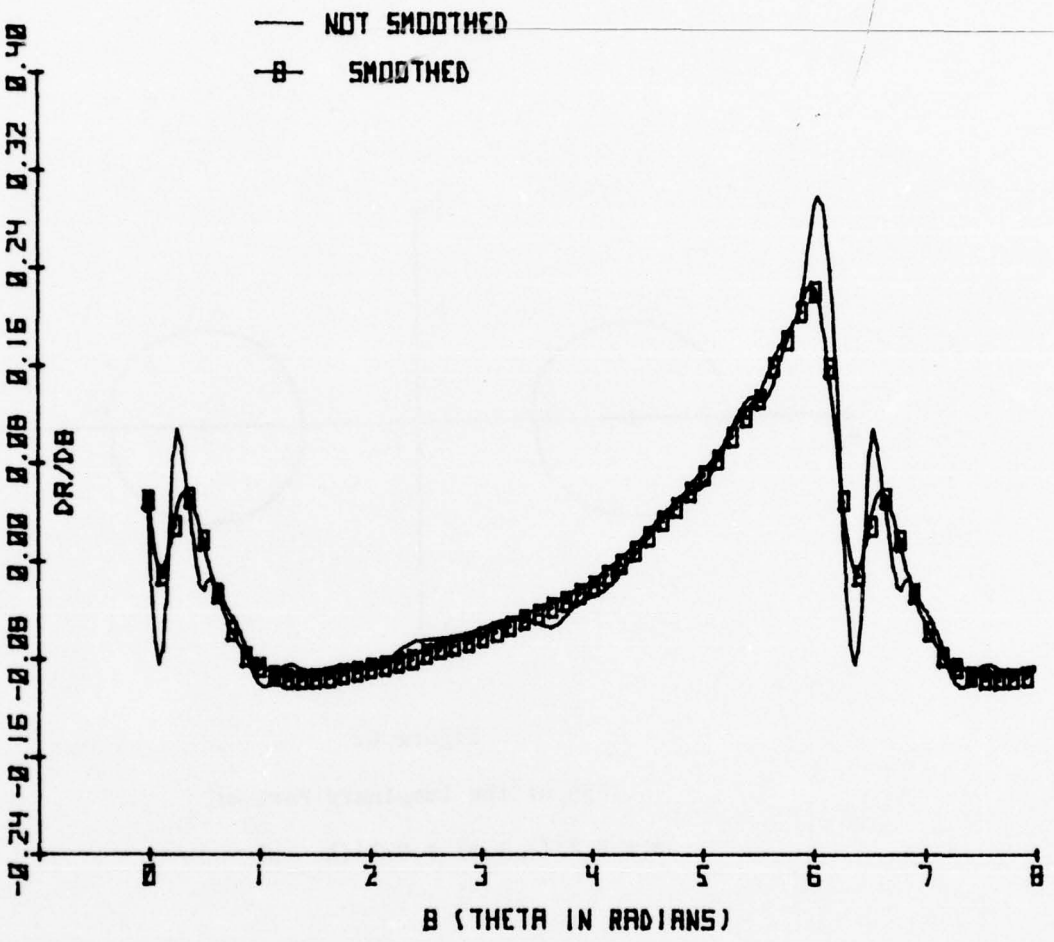


Figure 5.

The Derivative of the Cubic Spline Fit of the Near Circle Data and Its Smoothed Replacement

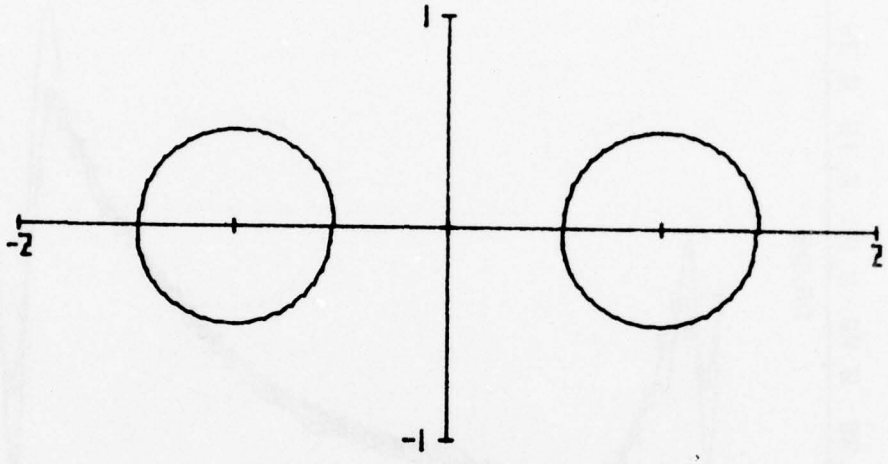


Figure 6.

Graph of the Imaginary Part of

$z + 0.2/(1 + z) + 0.2/(1 - z) = 0$

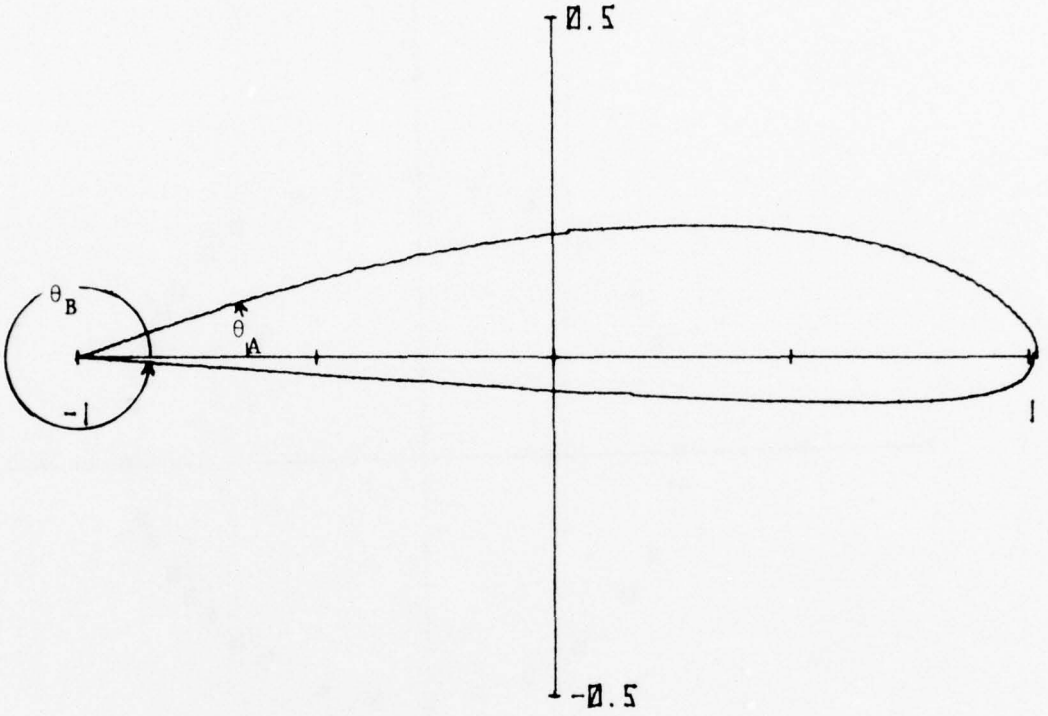


Figure 7.

The Angles at the Trailing Edge

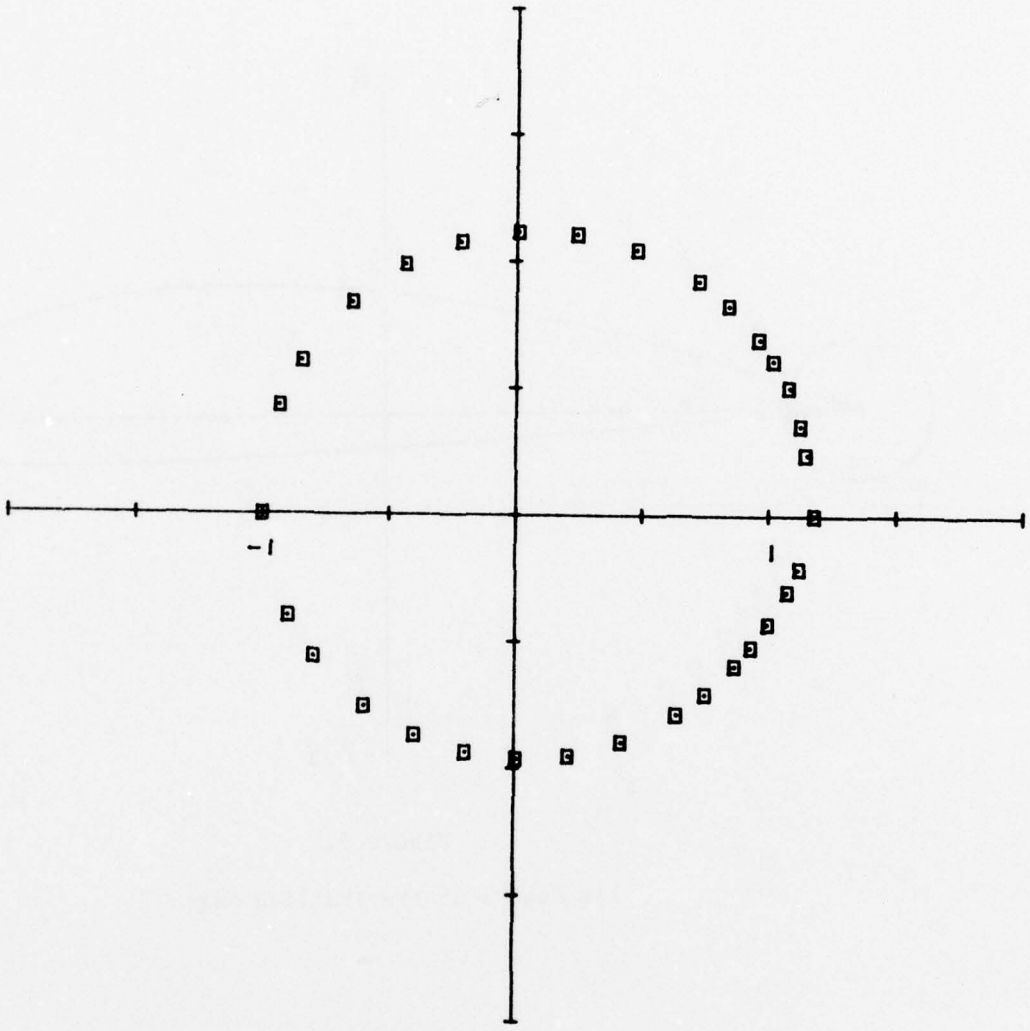


Figure 8.

The Image of the Clark Y Profile Data

Under the Karman Trefftz-like Mapping

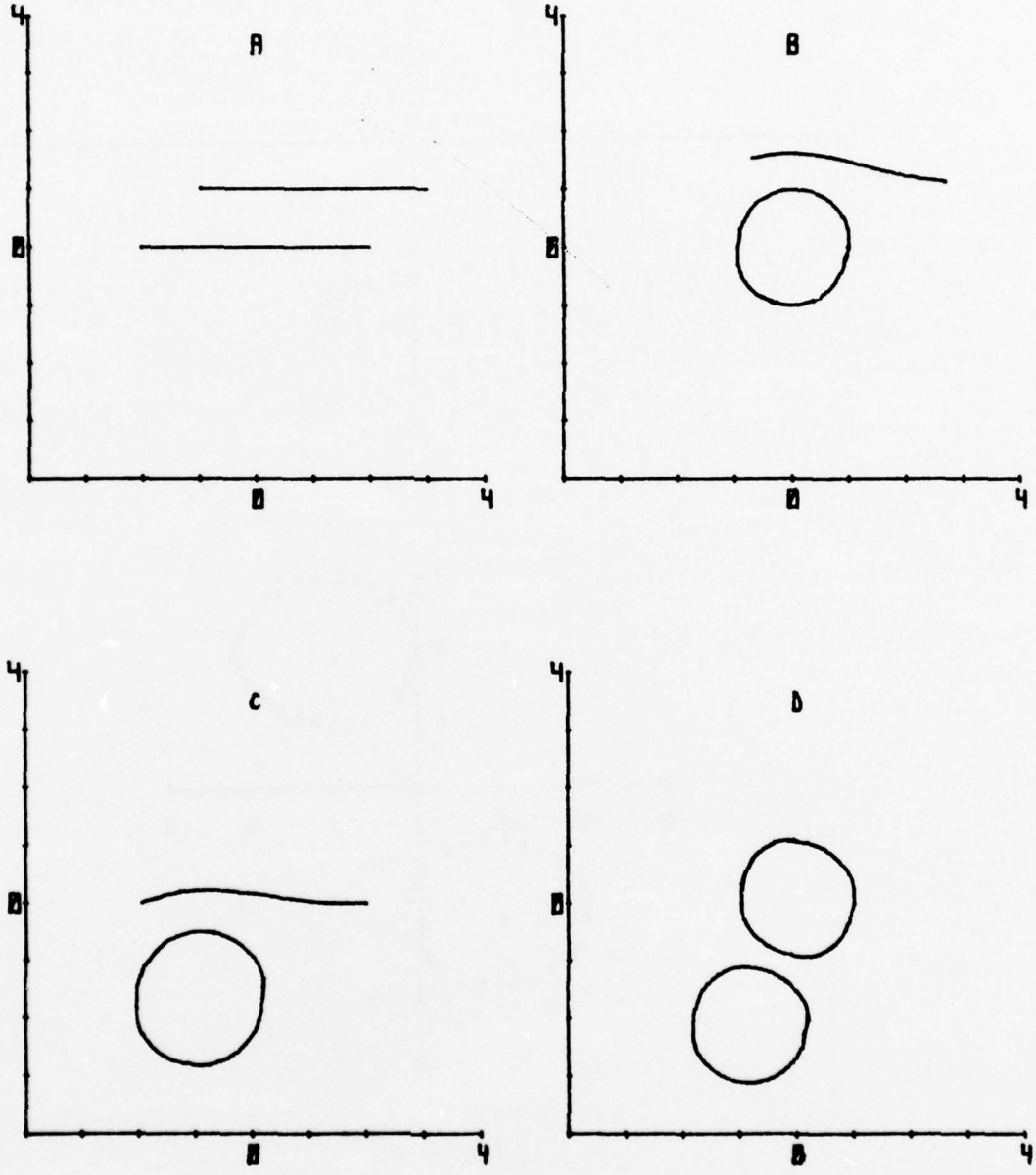
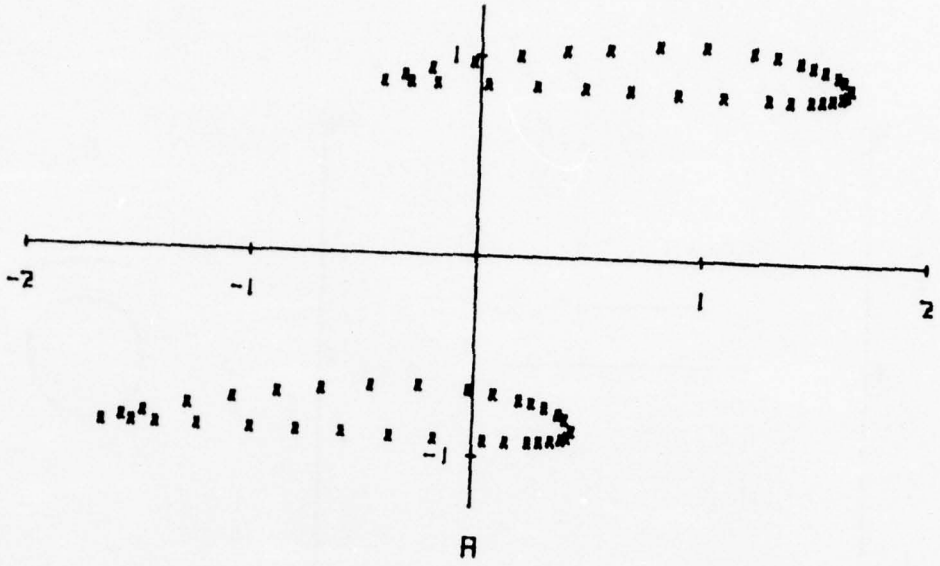
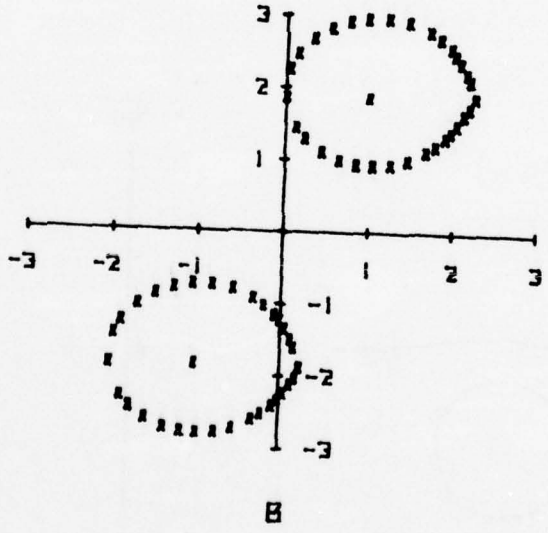


Figure 9.

Mapping of Two Parallel Slits into Two Near Circles



A



B

Figure 10.

Biplane Profile Data Transformed Into Two Near Cirlces

APPENDIX A

LOGARITHMIC POTENTIALS

In this appendix we give some properties of harmonic functions expressed as the logarithmic potential of a single or double layer. These properties are the basis for the derivatives presented in Section II. For additional details and proofs see [3, 4, 5]. We discuss also the solvability of some integral equations in this background material and a modification of these equations which removes the solvability restriction. Our notation, conventions and basic assumptions are the same as in Section II.

The function

$$w(x,y) = \int_C m(t)(\partial[\log r]/\partial n_2)\, dt \qquad (A1)$$

defined by the line integral is called the logarithmic potential of a double layer of density (or moment) m(t) distributed over the contour C. The density function m(t) is supposed continuous. The integral in Eq. (A1) defines a function w(x,y) which is continuous, has continuous derivatives of all orders and satisfies Laplace's equation

$$\Delta w = \partial^2 w/\partial x^2 + \partial^2 w/\partial y^2 = 0$$

at all points (x,y) not on C.

The integrand in Eq. (A1) may be written in other ways, such as

$$m(t)(1/r)\partial r/\partial n_2 \tag{A2}$$

$$m(t)((y - \eta(t))(d\xi/dt) - (x - \xi(t))(d\eta/dt))/r^2 \tag{A3}$$

$$m(t)(\cos(r,n))/r \tag{A4}$$

$$m(t)d\beta/dt \tag{A5}$$

In Eq. (A4) $\cos(r,n)$ denotes the cosine of the angle between the normal and the radius vector to the point (ξ,η) from the point (x,y). In Eq. (A5) β is the angle $\theta - \pi$ where $\theta = \arctan(y - \eta)/(x - \xi)$.

The function $w(x,y)$ does not vary in a continuous fashion as the point (x,y) passes through the contour C. Let s denote an arbitrary but fixed value satisfying $0 \leq s \leq L$. Let $w_i(s)$ and $w_e(s)$ denote the limits of $w(x,y)$ as the point with coordinates (x,y) tends to the point $(\xi(s), \eta(s))$ on C through points lying only inside C and points lying only outside C, respectively. Then $w_i(s)$, $w_e(s)$ and $m(s)$, the value of $m(t)$ at $(\xi(s), \eta(s))$ on C, satisfy the following conditions:

$$w_i(s) - w_e(s) = 2\pi m(s) \tag{A6}$$

$$w_i(s) = \pi m(s) + \int_C m(t)(\partial[\log r]/\partial n_2)\, dt \tag{A7}$$

$$w_e(s) = -\pi m(s) + \int_C m(t)(\partial[\log r]/\partial n_2)\, dt \tag{A8}$$

at $(\xi(s), \eta(s))$. Equations (A6), (A7) and (A8) exhibit the connection between the density function $m(t)$ and the boundary values of the function $w(x,y)$.

In the process of deriving the relations (A6),(A7) and (A8) it is seen that π is an eigenvalue belonging to the kernel $\partial[\log r]/\partial n_2$

and that m(t) equal to a constant is an eigenfunction corresponding to the eigenvalue π. The Eqs. (A7) and (A8) can be rewritten so that the kernel is $(1/\pi)\partial[\log r]/\partial n_2$. Then by an argument similar to that of [5, pp. 309-310] one can show that all the eigenvalues belonging to the "normalized" kernel $(1/\pi)\partial[\log r]/\partial n_2$ are real and lie between -1 and 1. It is shown also in [4, p. 104] that -1 is not an eigenvalue of the normalized kernel.

Given the boundary values, either $w_i(s)$ or $w_e(s)$, Eqs. (A7) and (A8) are Fredholm integral equations of the second kind for the density function m(t). From the Fredholm theory it is known that Eq. (A7) is uniquely solvable for any prescribed inhomogeneous term $w_i(s)$. The Eq. (A8), on the other hand, is not solvable unless $w_e(s)$ satisfies an orthogonality condition. Even when solvable the solution is not unique, but any two solutions differ at most by a constant.

The function

$$v(x,y) = \int_C k(t) \log r \, dt \tag{A9}$$

is called the logarithmic potential of a single layer of density k(t) distributed over C. This function defined by the line integral, Eq. (A9), is continuous, has continuous derivatives of all orders and satisfies Laplace's equation at all points (x,y) not on C. The function v(x,y) is continuous even as the point (x,y) passes through C, but the normal derivative $\partial v/\partial n_1$ is discontinuous as the point (x,y) passes through C.

Let s have a fixed value satisfying the condition $0 \leq s \leq L$. Let the point (x,y) be confined to the normal line to C at the point $(\xi(s), \eta(s))$. Let $\partial v_i(s)/\partial n_1$ and $\partial v_e(s)/\partial n_1$ denote the limits of $\partial v/\partial n_1$ as the point (x,y) tends to $(\xi(s), \eta(s))$ through points (x,y) inside C and outside C respectively. The values $\partial v_i(s)/\partial n_1$, $\partial v_e(s)/\partial n_1$ and k(s) satisfy the following conditions:

$$\partial v_e(s)/\partial n_1 - \partial v_i(s)/\partial n_1 = 2\pi k(s) \tag{A10}$$

$$\partial v_i(s)/\partial n_1 = -\pi k(s) + \int_C k(t)(\partial[\log r]/\partial n_1)dt \tag{A11}$$

$$\partial v_e(s)/\partial n_1 = \pi k(s) + \int_C k(t)(\partial[\log r]/\partial n_1)\, dt \tag{A12}$$

The kernel $\partial[\log r]/\partial n_1$ is the adjoint of $\partial[\log r]/\partial n_2$. Hence it has exactly the same eigenvalues as $\partial[\log r]/\partial n_2$. Given normal derivatives, either $\partial v_i(s)/\partial n_1$ or $\partial v_e(s)/\partial n_1$, the Eqs. (A11) and (A12) are Fredholm integral equations of the second kind for the density function k(t). Equation (A12) is solvable for arbitrary inhomogeneous terms $\partial v_e(s)/\partial n_1$. Equation (A11) is solvable provided

$$\int_C (\partial v_i(t)/\partial n_1)\, dt = 0 \tag{A13}$$

and when solvable, the solution is not unique.

Observe that the first order derivatives of w(x,y) and v(x,y) tend to zero as the quantity $x^2 + y^2$ increases without bound.

Next we want to make some observations and remarks concerning the solutions and the determination of solutions of the integral equations, Eqs. (A8) and (A11). When the solvability condition is

satisfied each of these equations is analogous to a consistent but dependent system of linear equations. As is well known, such a system of linear equations does not have an unique solution but a linear manifold of solutions. Occasionally, one has available information which can be used to change the consistent dependent system of equations into a consistent system of equations which determines a particular solution in the linear manifold of solutions of the dependent system. Then the linear manifold of solutions of the dependent system is obtained by adjoining to the particular solution the null space of the dependent system.

It is desireable to change a dependent system into an independent system because linear equation solvers are designed primarily for independent systems. Frequently this change is accomplished by deleting an appropriate number of variables and equations. The question then arises as to which variables and equations to delete as some sets may not span as well as others.

In numerical work a consistent dependent system may manifest itself as an independent system which may be ill conditioned. Even though one may get by treating such a system directly [10] we believe a procedure which is correct in principle is preferable. Given a dependent but consistent system of linear equations the process of deleting variables and equations is correct in principle. However, in the present case in which the linear system is obtained from an integral equation we believe an alternative procedure to be described next is preferable.

We have noted that the kernels of Eqs. (A8) and (A11) are adjoint to one another, π is a simple eigenvalue belonging to these kernels and for any constant $c \neq 0$, $f(t) = c$ is an eigenfunction or solution of the homogeneous equation Eq. (A8). Usually we take $c = 1$. Let $g(t)$ denote an eigenfunction or solution of the homogeneous equation Eq. (A11). The function $g(t)$ is not explicitly known.

From the Fredholm theory we know that for any continuous function $k(t)$ the image $v(s)$ where

$$v(s) = \pi k(s) - \int_C k(t)(\partial[\log r]/\partial n_1)\, dt \tag{A14}$$

satisfies the orthogonality condition

$$\int_C v(s)\, f(s)\, ds = \int_C v(s)\, ds = 0 \tag{A15}$$

and

$$\int_C g(s)\, f(s)\, ds = \int_C g(s)\, ds \neq 0 \tag{A16}$$

Consider now the Fredholm integral equation

$$\begin{aligned} u(s) &= \pi k(s) - \int_C k(t)(\partial[\log r]/\partial n_1)\, dt - \pi \int_C k(t)\, dt \\ &= v(s) - \pi \int_C k(t)\, dt \end{aligned} \tag{A17}$$

According to the Fredholm alternative either Eq. (A17) has a unique solution for any inhomogeneous term $u(s)$ or the homogeneous equation has non trivial solutions. It is readily seen that assuming the homogeneous equation has non trivial solutions leads to absurdities. Hence it follows Eq. (A17) has a unique solution for every inhomo-

geneous term u(s).

Next suppose the inhomogeneous term u(s) in Eq. (A17) satisfies the orthogonality condition

$$\int_C u(t)\, dt = 0$$

Then the solution k(s) of Eq. (A17) satisfies this orthogonality condition also. (Assuming otherwise gives rise to an absurdity as above.) Hence k(t) satisfies the integral equation, Eq. (A11) and the manifold of solutions of Eq. (A11) is given by

$$\hat{k}(s) = k(s) + \text{const. } g(s) \qquad \text{(A18)}$$

Since Eq. (A8) does not arise in the investigations carried out in the remainder of this paper we could omit further consideration here. However, for completeness and to increase understanding of the principles involved we shall discuss Eq. (A8) and its modification briefly. The modification appears to be the same as for Eq. (A11); that is, we include in the right-hand side of Eq. (A8) a term of the form

$$-\pi \int_C m(t)\, dt$$

The modification in both instances consists of adding to the original kernel a degenerate kernel of rank 1. This fact is hidden or may be overlooked because the degenerate kernel is the product of two constant functions. While the degenerate kernel is the same in both cases, its relation to the original kernels differs. In the

one case the degenerate kernel consists of the self product of the eigenfunction $f(t) = 1$ of the adjoint kernel while in the other case it is the self product of the eigenfunction $f(t) = 1$ of the kernel itself. The factors in the degenerate kernel are determined by the information available.

The modification term is of the form

$$\int_C f_1(s)\ f_2(t)\ k(t)\ dt \tag{A20}$$

Ideally, the function $f_1(t)$ is an eigenfunction of the kernel being modified and $f_2(t)$ is an eigenfunction of the adjoint kernel. (It is understood that f_1 and f_2 correspond to the same eigenvalue.) In general the desired eigenfunctions are not known. This modification technique can still be used if one knows functions made up from one or the other of the desired eigenfunctions.

If the eigenvalue is of multiplicity p then the modification consists of a sum of p terms

$$\sum_{j=1}^{p} \int_C f_{1j}(s)\ f_{2j}(t)\ k(t)\ dt \tag{A21}$$

The functions $f_{1j}(t)$, $j=1, \ldots, p$ must be linearly independent and so also the functions $f_{2j}(t)$.

APPENDIX B

THE KARMAN TREFFTZ-LIKE TRANSFORMATION

In this section we display the Karman Trefftz-like transformation. We noted that the first step in our procedure is the conformal transformation from the physical plane to the near circle plane. The purpose of this transformation is to remove the sharp trailing edge and to reduce the interference between the sources over those portions of the profile where the upper and lower surfaces are close together. Once this transformation into a near circle has been achieved no further transforming is carried out. By contrast, in the Theodorsen method [2] a transformation into a near circle is a first step in the transformation into a circle.

The Karman Trefftz transformation is a familiar tool of hydrodynamics. In this section it is broken down into a sequence of elementary transformations. In this form one can recognize rather easily which branch of the multivalued complex function one must take.

In the example which we treat, the airfoil profile has been scaled so that the trailing edge is at (-1,0) and the point (1,0) is approximately half the distance between the leading edge and the center of curvature of the leading edge. The portion of the horizontal axis between -1 and 1 lies inside the profile, Figure 7. This configuration simplifies the mapping procedure somewhat. Highly cambered profiles or profiles which differ considerably from

the type illustrated may require a slightly modified treatment.

Let $z_k = x_k + iy_k$ for k = 1 through 4. The z_1-plane is the physical plane, that is, the plane of the airfoil profile. Let

$$z_2 = (1 + z_1)/(1 - z_1) \tag{B1}$$

Then

$$dz_2/dz_1 = 2/(1 - z_1)^2 \tag{B2}$$

and

$$\lim_{z_1 \to \infty} (1 + z_1)/(1 - z_1) = -1 \tag{B3}$$

Thus the image of the point at infinity in the z_1-plane under the mapping (B1) is $z_2 = -1$. The inverse mapping is

$$z_1 = (z_2 - 1)/(z_2 + 1) \tag{B4}$$

and

$$dz_1/dz_2 = 2/(1 + z_2)^2 \tag{B5}$$

The next mapping is a rotation which causes the image of the upper surface of the profile at the trailing edge to be tangent to the real axis. We have

$$z_3 = z_2 \exp(-i\theta_A) \tag{B6}$$

where θ_A is the angle which the upper surface of the profile makes with the x_1-axis at the trailing edge, Figure 7. Then

$$dz_3/dz_2 = \exp(-i\theta_A) \tag{B7}$$

The point $z_2 = -1$ goes into the point

$$z_3 = \exp i(\pi - \theta_A) \tag{B8}$$

The inverse mapping is

$$z_2 = z_3 \exp(i\theta_A) \tag{B9}$$

and

$$dz_2/dz_3 = \exp(i\theta_A) \tag{B10}$$

Let θ_B denote the angle which the lower surface of the profile makes with the x_1-axis, Figure 7. Our object is to eliminate the sharp corner at the trailing edge. This is accomplished by the mapping

$$z_4 = z_3^{\,g} \tag{B11}$$

where $g = \pi/(\theta_B - \theta_A)$. Then

$$dz_4/dz_3 = g z_3^{\,g-1} \tag{B12}$$

The point $z_3 = \exp i(\pi - \theta_A)$ goes into the point

$$z_{44} = \exp i\pi(\pi - \theta_A)/(\theta_B - \theta_A) \tag{B13}$$

The inverse mapping is given by

$$z_3 = z_4^{\,1/g} \tag{B14}$$

and

$$dz_3/dz_4 = (1/g)z_4^{(1-g)/g} \tag{B15}$$

Finally a linear transformation

$$z = (z_4 + z_{44})/(z_4 - z_{44}) \tag{B16}$$

is carried out which maps z_{44}, the image of the point at infinity in the physical plane, into infinity in the z-plane and the image of the trailing edge, $z_4 = 0$, into $z = -1$. Then

$$dz/dz_4 = -2z_{44}/(z_4 - z_{44})^2 \tag{B17}$$

The inverse transformation is

$$z_4 = z_{44}(z + 1)/(z - 1) \tag{B18}$$

and

$$dz_4/dz = -2z_{44}/(z - 1)^2 \tag{B19}$$

The sequence of mappings defined by equations (B1), (B6), (B11) and (B16) determine z as a function of z_1. Replacing z_4 by its value in terms of z_3 and so on gives

$$z = \frac{(1 + z_1)^g + (1 - z_1)^g \exp(ig\pi)}{(1 + z_1)^g - (1 - z_1)^g \exp(ig\pi)} \tag{B20}$$

This is the Karman Trefftz-like transformation $z = F(z_1)$ which we use to map profiles into near circles. Figure 8 shows the near circle image of the Clark Y profile data under the transformation

just described.

In order to determine the values of U and V, Eq. (6) the value of dF/dz_1 at $z_1 = \infty$ is needed. By the chain rule

$$dz/dz_1 = (dz_2/dz_1)(dz_3/dz_2)(dz_4/dz_3)(dz/dz_4) \qquad (B21)$$

As z_1 tends to infinity dz/dz_1, as expressed by Eq (B21), is indeterminate, but it can be shown that the limiting value is 1/g. Thus we have at $z_1 = \infty$

$$dF/dz_1 = 1/g \qquad (B22)$$

The image of a highly cambered profile under a Karman Trefftz transformation may still be close to a circle, but not necessarily close to a circle with center at the origin of the z-plane. In this event it is practical to make a judicious shift of the origin to the center of gravity of the contour.

The Joukowski transformation, which is a special case of the Karman Trefftz transformation, applied to two line segments shows, Figure 9, the effects of the blow up of one profile on another. Figure 9, panel A, shows two parallel line segments. Panel B shows the image of these line segments under the inverse of the first Joukowski transformation. The lower line segment in panel A has been unfolded to produce the unit circle in panel B. Panel C shows the image of panel B under a transformation of the form Az + B. The endpoints of the upper contour are transformed into (-2,0) and (2,0) respectively and the unit circle is transformed into a circle.

Finally, panel D shows the image of panel C under the inverse of a second Joukowski transformation. This time the upper contour of panel C is unfolded into a nearly circular contour and the circle of panel C is transformed into a near circle also.

In Figure 10 biplane data, given in reference [11], is transformed into two near circles by a composite mapping consisting of two Karman Trefftz transformations and three linear transformations.

APPENDIX C

NUMERICAL DETERMINATION OF THE DENSITIES

The unknown density function k(t) or the tangential velocity v(t) is determined by an integral equation of the form

$$f(s) = \pm\pi k(s) + \int_C k(t)K(s, t)\, dt \tag{C1}$$

The integrand k(t)K(s, t) is a periodic function of t for each value of s. Such integrals can be evaluated with high accuracy by the trapezoidal rule provided that the integrands are sufficiently smooth, [7].

Since the contour C is described as r = T(θ), here r denotes the radius vector from the "center" of the contour C to a point on C, θ instead of length of arc t is taken for the integration variable. Partition the interval [0, 2π] into n equal subintervals and denote the endpoints of the i^{th} subinterval by $[\theta_i, \theta_{i+1}]$. For brevity write f_i for $f(\theta_i)$, K_{ij} for $K(\theta_i, \theta_j)$ and so on.

Let the contour C be given by the parametric equations

$$x = x(\theta),\ y = y(\theta) \tag{C2}$$

and let t(θ) denote the length of the contour C as a function of θ. Derivatives of functions of θ will be denoted by primes. Then the kernel

$$K_{ij} = \frac{(x_i - x_j)y_i' - (y_i - y_j)x_i'}{((x_i - x_j)^2 + (y_i - y_j)^2)t_i'} \tag{C3}$$

is indeterminate when $j = i$. Applying L'Hospital's Rule one obtains for the limit as θ_j tends to θ_i

$$K_{ii} = \frac{y_i'' x_i' - x_i'' y_i'}{2[(x_i')^2 + (y_i')^2]t_i'} \qquad \text{(C4)}$$

APPENDIX D

NUMERICAL DETERMINATION OF THE DERIVATIVES OF THE DENSITIES

The derivatives of the density functions are needed for the numerical determination of the tangential velocities, Eq. (5). These derivatives can be obtained by numerical differentiation of the density functions. They can be obtained also from the integral equation as shown in this appendix.

Differentiating Eq. (C1) with respect to arc length s gives

$$df/ds = \pi(dk/ds) + \int_C k(t)K_2(s, t)\, dt \qquad (D1)$$

where $K_2(s, t) = \partial K(s, t)/\partial s$. Set

$$A = (x - x_j)(d^2y/ds^2) - (y - y_j)(d^2x/ds^2),$$
$$B = (x - x_j)(dy/ds) - (y - y_j)(dx/ds),$$
$$C = (x - x_j)(dx/ds) + (y - y_j)(dy/ds),$$
$$R = (x - x_j)^2 + (y - y_j)^2$$

Then

$$K_2(s, t) = (RA - 2BC)/R^2$$

The kernel $K_2(s, t)$ is indeterminate for $t = s$.

Let us write, for example, $\overset{4}{R}$ for d^4R/dt^4. Using this convention, one has after four applications of L'Hospital's Rule

$$K_2(s,s) = \frac{\overset{4}{R}A + 4\overset{31}{R}A + 6\overset{22}{R}A + 4\overset{13}{R}A + \overset{4}{R}A - 2(BC + 4\overset{31}{R}BC + 6\overset{22}{R}BC + 4\overset{13}{R}BC + \overset{4}{R}BC)}{6\overset{22}{R}R + 8\overset{13}{R}R + 2\overset{4}{R}R}$$

which simplifies to

$$K_2(s, s) = (1/3)(RA^{13} - 2BC^{31} - 3RC^{22}) \tag{D2}$$

The right-hand side of Eq. (D2) written out is

$$\begin{aligned}&(2/3)((dx/ds)(d^3y/ds^3) - (dy/ds)(d^3x/ds^3))+\\&((dx/ds)(d^2x/ds^2) + (dy/ds)(d^2y/ds^2))((dy/ds)(d^2x/ds^2) -\\&(dx/ds)(d^2y/ds^2))\end{aligned} \tag{D3}$$

We are now ready to evaluate numerically the integral in Eq. (D1). We abide by the same conventions as in Appendix C and take θ as the integration variable. Then

$$(df/ds)_i = \pi(dk/ds)_i + \Delta\theta \sum_{j=1}^{n} k_j K_{2ij} (dt/d\theta)_j \tag{D4}$$

for $i = 1, \ldots, n$. Now $(dk/ds)_i$ is the value of dk/ds at the station θ_i. Solving Eq. (D4) for $(dk/ds)_i$ and multiplying by $(ds/d\theta)_i$ gives the value of $dk/d\theta$ at the station θ_i.

APPENDIX E

NUMERICAL DETERMINATION OF THE TANGENTIAL VELOCITIES

In this appendix we discuss the evaluation of the integral in Eq. (5) for the tangential velocities. The equation for the tangential velocities is quite similar to Eqs. (C1) and (D1). The difference is that the kernel in Eq. (5) has a first order singularity for t = s. Hence the integral must be interpreted in the sense of its Cauchy principle value.

Denote the integrand in Eq. (5) by $H(\theta)$. We have

$$H(\theta) = \frac{(x(\hat{\theta})-x(\theta))/(dx(\hat{\theta})/dt)+(y(\hat{\theta})-y(\theta))(dy(\hat{\theta})/dt)}{(x(\hat{\theta}) - x(\theta))^2 + (y(\hat{\theta}) - y(\theta))^2} k(\theta)(dt/d\theta)$$

and $H(\theta)$ has a first order singularity at $\theta = \theta$. Take $\Phi = \theta - \hat{\theta}$. Then we can write

$$H(\theta) = (1/2)[H(\hat{\theta}+\phi) + H(\hat{\theta}-\phi)] + (1/2)[H(\hat{\theta}+\phi) - H(\hat{\theta}-\phi)] \qquad (E1)$$

that is, as the sum of an even and odd function of ϕ. It is clear that the singularity is in the odd part of $H(\theta)$. Then

$$\text{CPV}\int_0^{2\pi} H(\theta)d\theta = \int_{-\pi}^{\pi} (1/2)[H(\hat{\theta}+\phi) + H(\hat{\theta}-\phi)]\, d\phi \qquad (E2)$$

since the Cauchy principle value of the integral of the odd part of $H(\theta)$ is zero.

Needed for the evaluation of the right-hand side of Eq. (E2) is the value of

$$\lim_{\phi\to 0} (1/2)[H(\hat{\theta} + \phi) + H(\hat{\theta} - \phi)] \qquad (E3)$$

If $H(\theta)$ were a continuous function of θ then, as noted above, we would have

$$\int_0^{2\pi} H(\theta)\, d\theta = \Delta\theta \sum_j H(\theta_j) \tag{E4}$$

Suppose now that the first order singularity of $H(\theta)$ is at the i^{th} station θ_i. Then the sum

$$\Delta\theta \sum_j{}' H(\theta_j) + \Delta\theta \lim_{\varepsilon \to 0} (1/2)[H(\theta_i + \varepsilon) + H(\theta_i - \varepsilon)] \tag{E5}$$

where $\sum'$ denotes the sum over all values of $j \neq i$, is equivalent to evaluating the even part, that is, the right-hand side of Eq. (E2), by the trapezoidal rule.

Set

$$A = (dx/dt)(dx/d\theta) + (dy/dt)(dy/d\theta) \tag{E6}$$

$$B = k(\theta)(d^2t/d\theta^2) + (dk/d\theta)(dt/d\theta) \tag{E7}$$

$$C = 0.5[(dx/dt)(d^2x/d\theta^2) + (dy/dt)(d^2y/d\theta^2)]k(\theta)(dt/d\theta) \tag{E8}$$

$$D = (dx/d\theta)^2 + (dy/d\theta)^2 \tag{E9}$$

$$E = [(dx/d\theta)(d^2x/d\theta^2) + (dy/d\theta)(d^2y/d\theta^2)]k(\theta)(dt/d\theta) \tag{E10}$$

Then the limiting value of $0.5[H(\hat{\theta} + \phi) + H(\hat{\theta} - \phi)]$ as ϕ tends to zero is given by the expression

$$-(AB + C)/D + AE/D^2 \tag{E11}$$

evaluated at $\hat{\theta}$.

APPENDIX F

CURVE FITTING AND SMOOTHING

We have assumed that the contour C is given as a function of a continuous parameter. In practice, the airfoil profile frequently is given as a discrete set of points. In this case some curve fitting process is needed to determine the profile as a function of a continuous variable. After the profile data is transformed to the near circle plane the rapid change at the leading edge and the sudden change at the trailing edge have been removed. For these reasons it is expected that the near circle data can be interpolated easier than the profile data. We use a cubic spline to fit the near circle data. The general equation for the cubic spline and the procedure for fitting the near circle data is given next.

Let there be given n data points (x_i, y_i) with $x_i < x_{i+1}$. We approximate the curve between two points (x_i, y_i) and (x_{i+1}, y_{i+1}) with a polynomial $p_i(x)$ of the third degree and of the form

$$p_i(x) = \frac{(x_{i+1} - x)^3 \, y''_i}{6(x_{i+1} - x_i)} - \frac{(x_i - x)^3 \, y''_{i+1}}{6(x_{i+1} - x_i)}$$

$$+ (x_{i+1} - x)\,(y_i/(x_{i+1} - x_i) - y''_i(x_{i+1} - x_i)/6) \qquad \text{(F1)}$$

$$+ (x - x_i)(y_{i+1}/(x_{i+1} - x_i) - y''_{i+1}(x_{i+1} - x_i)/6),$$

where the coefficients are expressed in terms of the quantities y''_i and y''_{i+1}. The resulting curve has continuous curvature over the whole interval (x_1, x_n).

In the present case y is a periodic function of x. Using this fact we obtain the following system of equations for the y_i'''s:

$$(x_{i+1} - x_i)y_i'' + 2(x_{i+2} - x_i)y_{i+1}'' + (x_{i+2} - x_{i+1})y_{i+2}'' = 6\left(\frac{y_{i+2} - y_{i+1}}{x_{i+2} - x_{i+1}} - \frac{y_{i+1} - y_i}{x_{i+1} - x_i}\right) \tag{F2}$$

for $i = 1, \ldots, n-3$. For $i = n - 2$ and $n - 1$ we have, respectively

$$(x_n - x_{n-1})y_1'' + (x_{n-1} - x_{n-2})y_{n-2}'' + 2(x_n - x_{n-2})y_{n-1}'' = 6\left(\frac{y_1 - y_{n-1}}{x_n - x_{n-1}} - \frac{y_{n-1} - y_{n-2}}{x_{n-1} - x_{n-2}}\right), \tag{F3}$$

$$2(x_n - x_{n-1} + x_2 - x_1)y_1'' + (x_2 - x_1)y_2'' + (x_n - x_{n-1})y_{n-1}'' = 6\left(\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_{n-1}}{x_n - x_{n-1}}\right) \tag{F4}$$

Note that because of the periodicity we have n - 1 instead of n equations and unknowns.

The airfoil profile is transformed into a near circle in the z-plane. If the profile is given by a set of n discrete points then the near circle consists of a set of discrete points z_i, $i = 1, \ldots, n$ with $z_n = z_1$. Let r_i and θ_i denote the polar coordinates of the point z_i. Taking θ as the independent variable we set up the system of equations (F2), (F3) and (F4) with r_i playing the role of y_i and θ_i playing the role of x_i.

In Figure 3 we show the cubic spline fit of the near circle data transformed into the physical plane by the inverse of the Karman Trefftz-like transformation. First impressions, from this figure, are that the cubic spline fit is more than adequate. However, the graph of the derivative of this cubic spline fit, Figure 4, shows considerable waviness. Thus inaccuracies in the profile data may produce rapidly varying slopes and, in a high precision method, the pressure results will reflect these variations in slope. This may be the reason practitioners recommend limited accuracy in such computations. Obviously, the waviness of the results obtained with high accuracy computations is not caused by the method but by roughness in the data to which the method is applied. We mention this fact, because if it is not clearly recognized a good method might be rejected because the results are sensitive to inadequacies in the data.

Smoothing the pressure results caused by inaccuracies in the profile data introduces an element of arbitrariness. It seems preferable to smooth the profile data. One can proceed as follows: After r is determined as a function of the continuous variable θ, the values of $dr/d\theta$ are computed and plotted. If the graph of $dr/d\theta$ oscillates too much some of these oscillations should be smoothed out.

The graphs of $r - 1$ versus θ and $dr/d\theta$ versus θ are given in Figure 4. Using a commercial reader (a digitizer) values of $dr/d\theta$ are read at equal increments of θ from a smoothed graph of $dr/d\theta$.

These values of $dr/d\theta$ are used to determine a trigonometric interpolating polynomial for $dr/d\theta$. From this trigonometric interpolating polynomial r is redetermined as a function of θ, $r = T(\theta)$. Figure 5 shows the graph of $dr/d\theta$ (not smoothed), the values of $dr/d\theta$ which were read at equal increments of θ from a smoothed graph of $dr/d\theta$ and the smoothed graph of $dr/d\theta$ as given by the trigonometric interpolating polynomial. The smoothed graph as given by the trigonometric interpolating polynomial does not pass through the smoothed data because the constant term must be set equal to zero. (Otherwise the curve will not close to the same value of r after one complete cycle.)

To check the smoothing process the points $(T(\theta),\theta)$ for $0 \leq \theta \leq 2\pi$ are transformed to the physical plane by the inverse of the Karman Trefftz transformation. In this way a new airfoil profile is determined. If this new profile is technically unacceptable then the smoothed values of $dr/d\theta$ need adjusting. In Figure 2 we show the graph of the image of $(T(\theta),\theta)$ for $0 \leq \theta \leq 2\pi$ in the physical plane and the plot of the original data.

The pressure distribution is computed for this new profile. Of course, if the fitting of the original data and the smoothing is performed with sufficient skill, the new profile will be close to the original profile. After the fitting and smoothing process is satisfactory the next step is the determination of the density functions.

APPENDIX G

THE COMPUTER PROGRAMS AND LISTINGS

In the first part of this appendix we review the conditions which must be satisfied in order to use the computer programs listed at the end. These conditions and instructions are given in the comment statements also. SECTIONS II, III and the preceding appendices provide the theoretical basis and details for the creation of these programs. These programs fall into two categories. In the first category are four programs for determining contributions to the tangential velocities (usually in the plane of the near circles). In the second are programs concerned with mapping an airfoil profile into a near circle.

The program which maps an airfoil profile into a near circle is the first step in determining the equations for the near circle contour. The equations for the near circle contour (or contours) are needed in the programs for computing the contributions to the tangential velocities. The point which we wish to make here is that the listed programs only provide intermediate results. These are working programs but they are not finished products.

The four programs MTVLPI, MTVELP, MTVLSI and MTVELS for the contributions to the tangential velocities are much alike and instructions for using them are the same in each case. The programs MTVLPI and MTVELP are based on Prager's method with vortices distributed along the contours. The programs MTVLSI and MTVELS assume

a distribution of sources along the contours. The programs MTVLPI and MTVLSI solve certain systems of equations iteratively. Note that the letter S in the program name indicates a distribution of sources around the contours, the letter P a distribution of vortices and the letter I means that systems of linear equations are solved by an iterative process.

These programs were developed on the CDC Cyber 74. The programming is standard FORTRAN IV except for the use of NAMELIST in conjunction with the READ and WRITE statements on unit 5. NAMELIST is used only for separating and locating records on the file associated with unit 5. These nonstandard statements occur only in the main or control programs, that is, in PROGRAMS MTVLPI, MTVELP, MTVLSI and MTVELS. A user will have to replace the programming noted with programming which is compatible with the available processor and which provides the capability of reading previously prepared records and writing new records. Also in developing these programs it was convenient to write formatted records. For general use format free records are probably preferable.

The information which must be supplied to these programs is the number of contours NC, the number of points N(I) around the Ith contour at which contributions to the tangential velocities are to be computed and the equations of the near circle contours. The equations of the near circle contours are given in terms of a local polar coordinate system. The radius vector from the (approximate) center of the near circle contour to a point on the contour is

assumed given by a trigonometric interpolating polynomial. Thus the near circle contours are determined by their (approximate) centers and the coefficients of the trigonometric interpolating polynomials.

The integers NC and N(I) for I = 1, ... , NC are supplied at input, FORMAT(I5). Using NAMELIST to locate the proper record, the coefficient number M(J), FORMAT(I5), the coordinates (XX(J), YY(J)) of the "center" and the coefficients A(I, J), for I = 1, ... , M(J)+1, B(I, J) for I = 1, ... , M(J)-1 of the trigonometric interpolating polynomial of the Jth contour, FORMAT(5E23.14), for J = 1, ... , NC are read from the file associated with unit 5. A user may have to provide some alternate way to supply the contour equations to these programs.

The principle outputs from the programs MTVLPI and MTVELP are the arrays AA(NT), BB(NT) and CC(NT,NC). Here NT = $\sum_{I} N(I)$. The tangential velocity at points on the near circle contours is a linear combination of the entries in the arrays AA, BB, and CC. The coefficients for the linear combination are determined by the angle of attack and the Kutta condition. In the programs MTVLSI and MTVELS the tangential velocity is obtained as a linear combination of the entries in the arrays TAA(NT), TBB(NT) and TCC(NT, NC). These arrays are written on the file associated with unit 5. The user may have to provide another way for saving and recalling this information for computing tangential velocities and the pressure coefficient.

The program for mapping profile data into near circle data by a Karman Trefftz-like transformation is called KATR. The Karman Trefftz-like transformation is multivalued. The program listing of KATR is intended only for profiles which have the form indicated by Figure 7. That is, if one draws a horizontal line through the trailing edge, then, except for the leading and trailing edges, the upper surface of the profile lies entirely above this horizontal line and the lower surface lies entirely below. The Karman Trefftz transformation is not limited to profiles for which the upper and lower surfaces satisfy the above condition.

The profile data is separated into two sets. One set consists of the points on the upper and the other, the points on the lower surface. The leading and trailing edges are included in both sets. NA(NB) denotes the number of data points on the upper(lower) surface. Beginning with the leading edge, the points on the upper (lower) surface are indexed from 1 to NA(NB). It is assumed that the profile data is scaled so that the trailing edge has coordinates (-1, 0). The distance from the point (1,0) to the leading edge is approximately half the radius of curvature at the leading edge.

The input data to the program KATR is the integers NA and NB, FORMAT(215). Next the coordinates, FORMAT(6E10.0), of the points on the upper and lower surfaces are read from input. We used NAMELIST in conjunction with WRITE statements (as noted above) to place this input information on the permanent file associated with unit 5. A user may have to replace this coding with coding compatible with

his central processor.

The output from KATR is the four parameters TA($=\theta_A$), TB($=\theta_B$),G and T which are used in defining the Karman Trefftz-like transformation and its inverse, and the entries in the arrays TH(K), RO(K) and R2(K), where K = NA+NB-1. The entries in the arrays RO and TH are, respectively, the polar coordinates, radius vector and polar angle, of the near circle images of the profile data. The entries in the array R2 are the second derivatives of the cubic spline fit (the second derivatives of the radius vector with respect to the polar angle) of the near circle data. RO(K) = RO(1), R2(K) = R2(1) and TH(K) = TH(1) + 2π.

```
      PROGRAM MTVLPI(TAPE5,INPUT,OUTPUT)
      DIMENSION AAA1(2,2),IP1(2),AA1(2),B31(2)
      COMMON/BK3/S1(80),S2(80),X(80),X1(80),
     2 X2(80),Y(80),Y1(80),Y2(80),X1P(80),
     3 Y1P(80),TI(2),SD(80)
      COMMON/BK4/AA(80),BB(80),CC(80,2),CI(80),CA(80,2)
      COMMON/BK5/AAA(80,80),IP(80)
      COMMON/BK6/N(2),NC,NT
      COMMON/BK7/NI(2),XME(2),NA(2),XAM(2)
      COMMON/BK8/XO(2),YO(2),M(2),A(51,2),B(49,2)
**************************************************
**************************************************
C  MTVLPI IS THE CONTROL PROGRAM FOR DETERMIN-
C  ING THE TANGENTIAL VELOCITIES INDUCED BY A
C  DISTRIBUTION OF VORTICES ON THE CONTOURS.
C  IN THIS PROGRAM MOST OF THE LINEAR SYSTEMS
C  OF EQUATIONS ARE SOLVED BY ITERATIVE
C  PROCESSES.
C  NC              NUMBER OF CONTOURS
C  N(I)            NUMBER OF SUBDIVISIONS OF THE ITH CONTOUR
C  NT              TOTAL NUMBER OF SUBDIVISIONS ALL CONTOURS
C                  NT=SUM N(I), I=1,NC
C  WE HAVE USED FIXED RATHER THAN VARIABLE DIMENSIONS.  IF THE
C  NUMBER OF CONTOURS OR IF THE NUMBER OF SUBDIVISIONS IS IN-
C  CREASED THEN THE DIMENSION STATEMENTS WILL HAVE TO BE CHANGED
C  AND THE PROGRAM MTVLPI AND SUBROUTINES  WHICH IT CALLS WILL
C  HAVE TO BE RECOMPILED TO REFLECT THE NEW VALUES OF NT AND NC
C  XO(K),YO(K)     APPROXIMATE "CENTER" OF THE KTH CONTOUR
C  A(L,K),B(L,K)   FOURIER COSINE AND SINE COEFFICIENTS FOR THE
C                  RADIUS VECTOR R FROM XO(K),YO(K) TO A POINT ON
C                  THE KTH CONTOUR.
C  THE CENTERS (XO(K),YO(K)) AND THE FOURIER
C  COEFFICIENTS A(L,K) AND B(L,K) ARE READ
C  FROM THE FILE, TAPE5.  THESE QUANTITIES
C  MUST BE PREPARED AND STORED IN ADVANCE
C  IN THE PROPER FORMAT.
C  AA,BB           CONTRIBUTIONS TO THE TANGENTIAL VELOCITY
C                  RESULTING FROM THE VELOCITY COMPONENTS
C                  AT INFINITY.
C  CC              COMPONENTS OF THE EIGENVECTORS.  THESE
C                  PROVIDE THE CIRCULATION.
C  CA              THE EIGENVECTORS OF THE ADJOINT
C                  PROBLEM OR FOR THE TRANSPOSED MATRIX
C  X(I),Y(I)       COORDINATES OF THE ITH STATION ON THE CON-
C                  TOURS.  X1(I),Y1(I) ARE THE FIRST DERIVATIVES
C                  W.R.T.THE CENTRAL ANGLE THETA AND SO ON
C                  X1P(I),Y1P(I) ARE THE FIRST DERIVATIVES
C                  W.R.T. ARC LENGTH AND SO ON
C  S1,S2           DENOTE FIRST AND SECOND DERIVATIVES
C                  OF THE ARC LENGTH W.R.T. THE CENTRAL ANGLE
C  SD               CHANGE OF INTEGRATION VARIABLE FACTOR
C  MTVLPI CALLS THE FOLLOWING SUBPROGRAMS
C  MVALP   THIS SUBROUTINE USES THE QUANTITIES IN COMMON BLOCK
C          8 TO COMPUTE THE QUANTITIES,PRIMARILY IN BK3.
```

PRECEDING PAGE BLANK-NOT FILMED

```
C  MPMATI  THIS SUBROUTINE USES THE VALUES IN COMMON BLOCK 3
C          TO COMPUTE THE ENTRIES FOR THE MATRIX AAA.  THEN
C          THE EIGENVECTORS CORRESPONDING TO THE EIGEN-
C          VALUE PI OF AAA AND ITS TRANSPOSE ARE COMPUTED
C          THE MATRIX ENTRIES ARE STORED IN COMMON BLOCK 5.
C  SCV     THIS SUBROUTINE GIVES THE PRODUCT OF A VECTOR
C          BY A SCALAR.
C  PMAT2   THIS SUBROUTINE COMPUTES THE COEFFICIENT
C          MATRIX USED IN REMOVING THE COMPONENTS IN
C          IN THE DIRECTION OF THE EIGENVECTORS FROM
C          A GIVEN VECTOR.
C  LU3     A ROUTINE FOR THE DIRECT SOLUTION OF SYSTEMS
C          OF LINEAR EQUATIONS TAKEN FROM THE ARL
C          LINEAR ALGEBRA LIBRARY.  IT INCLUDES LEQS3.
C  PITER   THIS SUBROUTINE SOLVES FOR THE CONTRIBUTIONS TO
C          THE TANGENTIAL VELOCITIY BY AN ITERATION
C          PROCESS.  THIS ROUTINE IS APPLICABLE ONLY
C          TO VERY SPECIAL SYSTEMS OF LINEAR EQUATIONS
C  THE INPUT DATA TO RUN THIS PROGRAM IS AS FOLLOWS
C  NOTE-THE LETTER B DENOTES A BLANK IN THE INPUT DATA
C  *EOR
C  BBBNC
C  BBBN(1)
C  ETC.
C  BBBN(NC)
C  *EOR
***********************************************
      NAMELIST/NAM/LLL/NAM1/LLL/NAM2/LLL/NAM3/LLL/NAM4/LLL/NAM5/
     2 LLL/NAM6/LLL/NAM7/LLL/NAM8/LLL/NAM9/LLL
      LLL=1
      PI=3.1415926535898
      READ 900,NC
      NT=0
      DO 10 I=1,NC
      READ 900,N(I)
 10   NT=NT+N(I)
      READ(5,NAM4)
      DO 20 J=1,NC
      READ(5,900)M(J)
      MP=M(J)+1
      MM=M(J)-1
      READ(5,901)XO(J),YO(J)
      READ(5,901)(A(I,J),I=1,MP)
 20   READ(5,901)(B(I,J),I=1,MM)
      CALL MVALP
      C=1/PI
      CALL SCV(NT,C,AA)
      CALL SCV(NT,C,BB)
      CALL MPMATI
C  ANY COMPONENTS ALONG THE EIGENVECTORS CORRESPONDING
C  TO THE EIGENVALUE PI ARE REMOVED NEXT.
      CALL PMAT2(NT,NC,AAA1,CA,CC)
      DO 24 J=1,NC
      AA1(J)=0
```

```
      BB1(J)=0
      DO 22 I=1,NT
      AA1(J)=AA1(J)+CA(I,J)*AA(I)
22    BB1(J)=BB1(J)+CA(I,J)*BB(I)
24    CONTINUE
      CALL LU3(NC,AAA1,2,IP1)
      CALL LEQS3(NC,AAA1,2,IP1,AA1)
      CALL LEQS3(NC,AAA1,2,IP1,BB1)
      DO 26 I=1,NT
      DO 25 J=1,NC
      AA(I)=AA(I)-AA1(J)*CC(I,J)
25    BB(I)=BB(I)-BB1(J)*CC(I,J)
26    CONTINUE
C  THE COMPONENTS IF ANY ARE REMOVED
      WRITE(5,NAM5)
C  STORE THE EIGENVECTORS OF THE TRANSPOSE OF AAA.
      DO 40  K1=1,NC
      I1=1
      I2=0
      WRITE(5,900)NA(K1)
      WRITE(5,901)XAM(K1)
      DO 30 K=1,NC
      I2=I2+N(K)
      WRITE(5,900)N(K)
      WRITE(5,901)(CA(I,K1),I=I1,I2)
30    I1=I1+N(K)
40    CONTINUE
      WRITE(5,NAM6)
      CALL PITER(NT,K,EM,AAA,AA,L)
      IF (L) 42,42,44
42    PRINT 902,EM
      GO TO 100
44    WRITE(5,900)K
      WRITE(5,901)EM
      I1=1
      I2=0
      DO 50 K=1,NC
      I2=I2+N(K)
      WRITE(5,900)N(K)
      WRITE(5,901)(AA(I),I=I1,I2)
50    I1=I1+N(K)
      CALL PITER(NT,K,EM,AAA,BB,L)
      IF (L) 52,52,54
52    PRINT 902,EM
      GO TO 100
54    WRITE(5,900)K
      WRITE(5,901)EM
      I1=1
      I2=0
      DO 60 K=1,NC
      I2=I2+N(K)
      WRITE(5,900)N(K)
      WRITE(5,901)(BB(I),I=I1,I2)
60    I1=I1+N(K)
```

```
      DO 80 K1=1,NC
      I1=1
      I2=0
      WRITE(5,900)NI(K1)
      WRITE(5,901)XME(K1)
      DO 70 K=1,NC
      I2=I2+N(K)
      WRITE(5,900)N(K)
      WRITE(5,901)(CC(I,K1),I=I1,I2)
70    I1=I1+N(K)
80    CONTINUE
100   STOP
900   FORMAT(I5)
901   FORMAT(5E23.14)
902   FORMAT(*0EM=*E23.14,*, ITERATION PROCESS DID NOT CONVERGE*)
      END



      SUBROUTINE MVALP
      DIMENSION CH(2)
      COMMON/BK3/S1(80),S2(80),X(80),X1(80),
     2 X2(80),Y(80),Y1(80),Y2(80),X1P(80),
     3 Y1P(80),TI(2),SD(80)
      COMMON/BK4/AA(80),BB(80),CC(80,2),CI(80),CA(80,2)
      COMMON/BK5/N(2),NC,NT
      COMMON/BK8/XO(2),YO(2),M(2),A(51,2),B(49,2)
      REAL MFOU,MDFOU,MD2FOU
*****************************************************************
C  THIS SUBROUTINE PREPARES TABLES OF VALUES TO BE USED
C  IN LATER COMPUTATIONS.  SPECIFICALLY, IT COMPUTES THE
C  COORDINATES AT STATIONS ALONG THE CONTOUR C,THE
C  DERIVATIVES OF THESE COORDINATES W.R.T. THE
C  POLAR VARIABLE THETA=T,AND WITH RESPECT TO ARC
C  LENGTH.  THESE ARE DENOTED E.G.,BY X1 AND X1P,RESP.
C  IT ALSO COMPUTES THE DERIVATIVES OF THE ARC  LENGTH
C  W.R.T. THE POLAR VARIABLE.  THESE DERIVATIVES ARE
C  DENOTED BY S1,S2,ETC.
C  THIS SUBROUTINE USES THE FOLLOWING REAL
C  FUNCTION PROGRAMS:
C  MFOU    EVALUATES A TRIGONOMETRIC POLYNOMIAL WITH
C          FOURIER COEFFICIENTS A(I,K) AND B(I,K) AT
C          A GIVEN ANGLE (IN RADIANS).
C  MDFOU   EVALUATES THE FIRST DERIVATIVE OF THE
C          TRIGONOMETRIC POLYNOMIAL.
C  MD2FOU  EVALUATES THE SECOND DERIVATIVE OF THE
C          TRIGONOMETRIC POLYNOMIAL.
*****************************************************************
      PI=3.1415926535898
      DO 10 K=1,NC
10    CH(K)=0.
      II=0
      DO 40 K=1,NC
```

```
      TI(K)=2.*PI/N(K)
      T=-TI(K)
      I2=N(K)
      DO 30 I=1,I2
      T=T+TI(K)
      R=MFOU(T,M(K),A,B,K)
      R1=MDFOU(T,M(K),A,B,K)
      R2=MD2FOU(T,M(K),A,B,K)
      S1(I+II)=SQRT(R**2+R1**2)
      SD(I+II)=S1(I+II)*TI(K)
      U=R+R2
      V=R1*U
      S2(I+II)=V/S1(I+II)
      D2=S1(I+II)**2
      CO=COS(T)
      SI=SIN(T)
      X(I+II)=CO*R+XO(K)
      Y(I+II)=SI*R+YO(K)
      X1(I+II)=CO*R1-R*SI
      Y1(I+II)=SI*R1+R*CO
      X2(I+II)=CO*R2-2.*SI*R1-CO*R
      Y2(I+II)=SI*R2+2.*CO*R1-SI*R
      X1P(I+II)=X1(I+II)/S1(I+II)
      AA(I+II)=2.*PI*X1P(I+II)
      Y1P(I+II)=Y1(I+II)/S1(I+II)
      BB(I+II)=2.*PI*Y1P(I+II)
30    CONTINUE
      II=II+N(K)
40    CONTINUE
      DO 70 K=1,NC
      I1=1
      I2=0
      CH(K)=1
      DO 60 J=1,NC
      I2=I2+N(J)
      DO 50 I=I1,I2
50    CC(I,K)=CH(J)
60    I1=I2+1
70    CH(K)=0
      RETURN
      END



      SUBROUTINE SCV(N,C,V)
      DIMENSION V(80)
**********************************************
C  THIS SUBROUTINE MULTIPLIES A VECTOR BY A SCALAR.
**********************************************
      DO 10 I=1,NT
 10   V(I)=C*V(I)
      RETURN
      END
```

```
      SUBROUTINE MPMATI
      COMMON/BK3/S1(80),S2(80),X(80),X1(80),
     2 X2(80),Y(80),Y1(80),Y2(80),X1P(80),
     3 Y1P(80),TI(2),SD(80)
      COMMON/BK4/AA(80),BB(80),CC(80,2),CI(80),CA(80,2)
      COMMON/BK5/AAA(80,80),IP(80)
      COMMON/BK6/N(2),NC,NT
      COMMON/BK7/NI(2),XME(2),NA(2),XAM(2)
**********************************************************
**************************************
C  THIS SUBROUTINE DETERMINES THE MATRIX ELEMENTS NEEDED FOR THE
C  COMPUTATION OF THE TANGENTIAL VELOCITIES.
C  THEN THE EIGENFUNCTIONS ARE DETERMINED.
**************************************

      I2=NT-1
      DO 30 I=1,I2
      J1=I+1
      DO 20
      U=(X(I)-X(J))*Y1P(I)-(Y(I)-Y(J))*X1P(I)
      V=(X(I)-X(J))**2+(Y(I)-Y(J))**2
20    AAA(I,J)=(U/V)*SD(J)
30    CONTINUE
      DO 50 I=2,NT
      J2=I-1
      DO 40 J=1,J2
      U=(X(I)-X(J))*Y1P(I)-(Y(I)-Y(J))*X1P(I)
      V=(X(I)-X(J))**2+(Y(I)-Y(J))**2
40    AAA(I,J)=(U/V)*SD(J)
50    CONTINUE
      DO 60 I=1,NT
      U=Y2(I)*X1P(I)-X2(I)*Y1P(I)
      V=X1(I)**2+Y1(I)**2
60    AAA(I,I)=0.5*(U/V)*SD(I)
      DO 82 K=1,NC
      DO 80 J=1,NT
80    CA(J,K)=CC(J,K)
82    CONTINUE
**********************************************
C  THE EIGENFUNCTIONS ARE DETERMINED BY ITERATION.  XM
C  GIVES THE MAXIMUM DEVIATION OF THE COMPONENTS OF THE CURRENT
C  KTH ITERATE FROM THOSE OF THE PREVIOUS K-1 TH ITERATE.
C  THE ITERATION PROCESS TERMINATES IF EITHER K=20 OR
C  XM L.T.E. 1.*10**-8.  THE EIGENFUNCTIONS
C  CORRESPONDING TO THE EIGENVALUE PI OF THE
C  OF THE TRANSPOSE OF AAA ARE DETERMINED FIRST.
**********************************************

      DO 140 K1=1,NC
      DO 120 K=1,20
      XM=0
      DO 100 J=1,NT
      CI(J)=0
      DO 90 I=1,NT
90    CI(J)=CI(J)+CA(I,K1)*AAA(I,J)
```

```
      CI(J)=CI(J)/PI
      XM1=ABS(CA(J,K1)-CI(J))
      XM=AMAX1(XM,XM1)
100   CONTINUE
      DO 110 J=1,NT
110   CA(J,K1)=CI(J)
      IF (1.E-08-XM) 120,130,130
120   CONTINUE
130   XAM(K1)=XM
      NA(K1)=K
140   CONTINUE
C  THE EIGENFUNCTIONS CORRESPONDING TO PI OF THE
C  MATRIX AAA ARE DETERMINED NEXT.
      DO 230 K1=1,NC
      DO 210 K=1,20
      XM=0
      DO 190 I=1,NT
      CI(I)=0.
      DO 180 J=1,NT
 180  CI(I)=CI(I)+AAA(I,J)*CC(J,K1)
      CI(I)=CI(I)/PI
      XM1=ABS(CC(I,K1)-CI(I))
      XM=AMAX1(XM,XM1)
 190  CONTINUE
      DO 200 I=1,NT
 200  CC(I,K1)=CI(I)
      IF(1.E-08-XM) 210,220,220
 210  CONTINUE
 220  XME(K1)=XM
      NI(K1)=K
 230  CONTINUE
**************************************************************
C  THE EIGENFUNCTIONS HAVE BEEN DETERMINED.  NI(K1) GIVES THE
C  NUMBER OF ITERATIONS IN THE DETERMINATION OF THE K1TH EIGEN-
C  FUNCTION.  XME(K1) GIVES THE MAXIMUM DIFFERENCE BETWEEN THE
C  COMPONENTS OF THE LAST TWO ITERATES OF THE K1TH EIGENFUNCTION.
**************************************************************
      RETURN
      END



      SUBROUTINE PITER(N,K,EM,A,B,L)
      DIMENSION A(80,80),B(80),V(80),BC(80)
**************************************************************
C    THIS SUBROUTINE SOLVES THE LINEAR SYSTEM OF EQUATIONS
C    (PI*I-A)X=B  BY AN ITERATION PROCESS.  HERE PI IS AN EIGEN-
C    VALUE OF THE MATRIX A.  FIRST, REWRITE THE SYSTEM OF EQUA-
C    TIONS IN THE FORM (I-A/PI)X=B/PI AND SET AC=A/PI AND BD=B/PI.
C    THE MATRIX AC HAS 1 FOR AN EIGENVALUE.  WE SUPPOSE REMOVED
C    FROM THE VECTOR BD ANY EIGENVECTORS CORRESPONDING TO THE
C    EIGENVALUE 1.  ONE OBTAINS X=BD+AC*BD+AC*(AC*BD)+ ETC.
C    PITER CALLS THE FOLLOWING SUBPROGRAM:
```

```
C  MATV   THIS SUBROUTINE GIVES THE PRODUCT OF A VECTOR BY
C        A MATRIX AND DIVIDES THE RESULTING VECTOR BY PI.
*********************************************************
      PI=3.1415926535898
      DO 5 I=1,N
 5    V(I)=B(I)
      K=1
 15   CALL MATV(N,A,V)
      DO 20 I=1,N
 20   B(I)=B(I)+V(I)
      EM=0
      DO 30 I=1,N
      U=ABS(V(I))
      E=AMAX1(EM,U)
 30   EM=E
      IF (1.E-08-EM) 40,40,70
 40   IF(20-K) 60,60,50
 50   K=K+1
      GO TO 15
 60   L=0
      GO TO 90
 70   L=1
 90   RETURN
      END



      SUBROUTINE PMAT2(N,M,A,U,V)
      DIMENSION A(2,2),U(80,2),V(80,2)
*********************************************************
C    THIS SUBROUTINE CONSTRUCTS THE COEFFICIENT MATRIX USED IN
C    DETERMINING THE LINEAR COMBINATION OF EIGENVECTORS
C    CORRESPONDING TO THE EIGENVALUE PI IN A VECTOR.
*********************************************************
      DO 30 I=1,M
      DO 20 J=1,M
      DO 10 K=I,N
10    S=S+U(K,I)*V(K,J)
20    A(I,J)=S
30    CONTINUE
      RETURN
      END



      SUBROUTINE MATV(N,A,V)
      DIMENSION A(80,80),V(80),S(80)
*********************************************************
C  THIS SUBROUTINE MULTIPLIES A VECTOR BY A MATRIX AND THEN
C  DIVIDES THE RESULTING VECTOR BY PI.
*********************************************************
      PI=3.1415926535898
```

```
      DO 20 I=1,N
      S(I)=0
      DO 10 J=1,N
10    S(I)=S(I)+A(I,J)*V(J)
20    CONTINUE
      DO 30 I=1,N
30    V(I)=S(I)/PI
      RETURN
      END



      PROGRAM MTVELP(TAPE5,INPUT,OUTPUT)
      COMMON/BK3/S1(80),S2(80),X(80),X1(80),
     2 X2(80),Y(80),Y1(80),Y2(80),X1P(80),
     3 Y1P(80),TI(2),SD(80)
      COMMON/BK4/AA(80),BB(80),CC(80,2),CI(80),CA(80,2)
      COMMON/BK5/AAA(80,80),IP(80)
      COMMON/BK6/N(2),NC,NT
      COMMON/BK7/NI(2),XME(2),NA(2),XAM(2)
      COMMON/BK8/XO(2),YO(2),M(2),A(51,2),B(49,2)
**************************************************
**************************************************
C  MTVELP IS THE CONTROL PROGRAM FOR DETERMINING
C  THE TANGENTIAL VELOCITIES INDUCED BY A DISTRI-
C  BUTION OF VORTICES ON THE CONTOURS.
C  IN THIS PROGRAM SYSTEMS OF LINEAR EQUATIONS
C  ARE SOLVED DIRECTLY.
C  NC              NUMBER OF CONTOURS
C  N(I)            NUMBER OF SUBDIVISIONS OF THE ITH CONTOUR
C  NT              TOTAL NUMBER OF SUBDIVISIONS ALL CONTOURS
C                  NT=SUM N(I), I=1,NC
C  WE HAVE USED FIXED RATHER THAN VARIABLE DIMENSIONS.  IF THE
C  NUMBER OF CONTOURS OR IF THE NUMBER OF SUBDIVISIONS IS IN-
C  CREASED THEN THE DIMENSION STATEMENTS WILL HAVE TO BE CHANGED
C  AND THE PROGRAM MTVELP AND SUBROUTINES  WHICH IT CALLS WILL
C  HAVE TO BE RECOMPILED TO REFLECT THE NEW VALUES OF NT AND NC
C  XO(K),YO(K)     APPROXIMATE "CENTER" OF THE KTH CONTOUR
C  A(L,K),B(L,K)   FOURIER COSINE AND SINE COEFFICIENTS FOR THE
C                  RADIUS VECTOR R FROM XO(K),YO(K) TO A POINT ON
C                  THE KTH CONTOUR.
C  THE CENTERS (XO(K),YO(K)) AND THE FOURIER
C  COEFFICIENTS A(L,K) AND B(L,K) ARE READ FROM
C  THE FILES ,TAPE5.  THESE QUANTITIES MUST BE
C  PREPARED AND STORED THERE IN ADVANCE AND IN
C  THE PROPER FORMAT.
C  AA,BB           CONTRIBUTIONS TO THE TANGENTIAL VELOCITY
C                  RESULTING FROM THE VELOCITY COMPONENTS
C                  AT INFINITY.
C  CC              COMPONENTS OF THE EIGENVECTORS.  THESE
C                  PROVIDE THE CIRCULATION.
C  CA              EIGENVECTORS OF THE ADJOINT PROBLEM
C                  OR FOR THE TRANSPOSED MATRIX.
```

```
C  X(I),Y(I)        COORDINATES OF THE ITH STATION ON THE CONTOURS
C                   X1(I),Y1(I) ARE THE FIRST DERIVATIVES W.R.T.
C                   THE CENTRAL ANGLE THETA AND SO ON
C                   X1P(I),Y1P(I) ARE THE FIRST DERIVATIVES W.R.T.
C                   ARC LENGTH AND SO ON
C  S1,S2            DENOTE THE 1ST, 2ND,ETC. DERIVATIVES
C                   OF THE ARC LENGTH W.R.T. THE CENTRAL ANGLE
C  SD                CHANGE OF INTEGRATION VARIABLE FACTOR
C  MTVELP CALLS THE FOLLOWING SUBPROGRAMS
C  MVALP   THIS SUBROUTINE USES THE QUANTITIES IN COMMON BLOCK
C          8 TO COMPUTE THE QUANTITIES,PRIMARILY IN BK3.
C  MPMAT   THIS SUBROUTINE USES THE VALUES IN COMMON BLOCK 3 TO
C          COMPUTE THE MATRIX ENTRIES FOR DETERMINING THE
C          TANGENTIAL VELOCITIES AND THE EIGENVECTORS.
C          THE MATRIX ENTRIES ARE STORED IN COMMON BLOCK 5.
C  LU3     A ROUTINE FOR THE DIRECT SOLUTION
C          OF SYSTEMS OF LINEAR EQUATIONS TAKEN
C          FROM THE ARL LINEAR ALGEBRA LIBRARY.
C          LU3 INCLUDES LEQS3.
C  THE INPUT DATA TO RUN THIS PROGRAM IS AS FOLLOWS
C  *EOR
C  BBBNC
C  BBBN(1)
C  ETC.
C  BBBN(NC)
C  *EOR
*********************************************
      NAMELIST/NAM/LLL/NAM1/LLL/NAM2/LLL/NAM3/LLL/NAM4/LLL/NAM5/
     2 LLL/NAM6/LLL/NAM7/LLL/NAM8/LLL/NAM9/LLL
      LLL=1
      PI=3.14159265358898
      READ 900,NC
      NT=0
      DO 10 I=1,NC
      READ 900,N(I)
 10   NT=NT+N(I)
      READ(5,NAM4)
      DO 20 J=1,NC
      READ(5,900)M(J)
      MP=M(J)+1
      MM=M(J)-1
      READ(5,901)XO(J),YO(J)
      READ(5,901)(A(I,J),I=1,MP)
 20   READ(5,901)(B(I,J),I=1,MM)
      CALL MVALP
      CALL MPMAT
      CALL LU3(NT,AAA,80,IP)
      CALL LEQS3(NT,AAA,80,IP,AA)
      CALL LEQS3(NT,AAA,80,IP,BB)
      READ(5,NAM5)
      DO 40  K1=1,NC
      I1=1
      I2=0
      WRITE(5,900)NA(K1)
```

```
      WRITE(5,901)XAM(K1)
      DO 30 K=1,NC
      I2=I2+N(K)
      WRITE(5,901)(CA(I,K1),I=I1,I2)
      WRITE(5,900)N(K)
30    I1=I1+N(K)
40    CONTINUE
      WRITE(5,NAM6)
      I1=1
      I2=0
      DO 50 K=1,NC
      I2=I2+N(K)
      WRITE(5,901)(AA(I),I=I1,I2)
      WRITE(5,900)N(K)
50    I1=I1+N(K)
      I1=1
      I2=0
      DO 60 K=1,NC
      I2=I2+N(K)
      WRITE(5,901)(BB(I),I=I1,I2)
      WRITE(5,900)N(K)
60    I1=I1+N(K)
      DO 80 K1=1,NC
      I1=1
      I2=0
      WRITE(5,900)NI(K1)
      WRITE(5,901)XME(K1)
      DO 70 K=1,NC
      I2=I2+N(K)
      WRITE(5,901)(CC(I,K1),I=I1,I2)
      WRITE(5,900)N(K)
70    I1=I1+N(K)
80    CONTINUE
      STOP
900   FORMAT(I5)
901   FORMAT(3E23.14)
      END



      SUBROUTINE MPMAT
      COMMON/BK3/S1(80),S2(80),X(80),X1(80),
     2 X2(80),Y(80),Y1(80),Y2(80),X1P(80),
     3 Y1P(80),TI(2),SD(80)
      COMMON/BK4/AA(80),BB(80),CC(80,2),CI(80),CA(80,2)
      COMMON/BK5/AAA(80,80),IP(80)
      COMMON/BK6/N(2),NC,NT
      COMMON/BK7/NI(2),XME(2),NA(2),XAM(2)
*****************************************************
*****************************************
C  THIS SUBROUTINE DETERMINES THE MATRIX ELEMENTS NEEDED FOR THE
C  COMPUTATION OF THE TANGENTIAL VELOCITIES.  THE MATRIX FOR
C  COMPUTING THE EIGENFUNCTIONS IS DETERMINED FIRST.
```

```
*****************************************
      PI=3.14159265358938
      I2=NT-1
      DO 30 I=1,I2
      J1=I+1
      DO 20 J=J1,NT
      U=(X(I)-X(J))*Y1P(I)-(Y(I)-Y(J))*X1P(I)
      V=(X(I)-X(J))**2+(Y(I)-Y(J))**2
20    AAA(I,J)=(U/V)*SD(J)
30    CONTINUE
      DO 50 I=2,NT
      J2=I-1
      DO 40 J=1,J2
      U=(X(I)-X(J))*Y1P(I)-(Y(I)-Y(J))*X1P(I)
      V=(X(I)-X(J))**2+(Y(I)-Y(J))**2
40    AAA(I,J)=(U/V)*SD(J)
50    CONTINUE
      DO 60 I=1,NT
      U=Y2(I)*X1P(I)-X2(I)*Y1P(I)
      V=X1(I)**2+Y1(I)**2
60    AAA(I,I)=0.5*(U/V)*SD(I)
*****************************************************************
C  THE EIGENFUNCTIONS ARE DETERMINED BY ITERATION.  XM
C  GIVES THE MAXIMUM DEVIATION OF THE COMPONENTS OF THE CURRENT
C  KTH ITERATE FROM THOSE OF THE PREVIOUS K-1 TH ITERATE.
C  THE ITERATION PROCESS TERMINATES IF EITHER K=20 OR
C  XM L.T.E.1.*10**-8.  AFTER THE EIGENFUNCTIONS ARE DETERMINED
C  THE MATRIX MUST BE MODIFIED TO DETERMINE THE CONTRIBUTIONS
C  TO THE TANGENTIAL VELOCITY DUE TO A UNIT X AND Y COMPONENT
C  OF THE VELOCITY AT INFINITY.
*****************************************************
*****************************************************
C  COMPUTE THE EIGENVECTORS CORRESPONDING TO THE
C  EIGENVALUE PI FOR THE ADJOINT PROBLEM.
*****************************************************
      DO 82 K=1,NC
      DO 80 J=1,NT
80    CA(J,K)=CC(J,K)
      XAM(K)=0
      NA(K)=0
82    CONTINUE
*****************************************************
C  COMPUTE THE EIGENVECTORS CORRESPONDING TO THE
C  EIGENVALUE PI FOR THE MATRIX AAA.
*****************************************************
      DO 230 K1=1,NC
      DO 210 K=1,20
      XM=0
      DO 190 I=1,NT
      CI(I)=0.
      DO 180 J=1,NT
180   CI(I)=CI(I)+AAA(I,J)*CC(J,K1)
      CI(I)=CI(I)/PI
      XM1=ABS(CC(I,K1)-CI(I))
```

```
      XM=AMAX1(XM,XM1)
  190 CONTINUE
      DO 200 I=1,NT
  200 CC(I,K1)=CI(I)
      IF(1.E-08-XM) 210,220,220
  210 CONTINUE
  220 XME(K1)=XM
      NI(K1)=K
  230 CONTINUE
**************************************************************
C  THE EIGENFUNCTIONS HAVE BEEN DETERMINED.  NI(K1) GIVES THE
C  NUMBER OF ITERATIONS IN THE DETERMINATION OF THE K1TH EIGEN-
C  FUNCTION.  XME(K1) GIVES THE MAXIMUM DIFFERENCE BETWEEN THE
C  COMPONENTS OF THE LAST TWO ITERATES OF THE K1TH EIGENFUNCTION.
**************************************************************
**************************************************************
**************************************************
C  THE MATRIX ELEMENTS FOR THE MODIFIED
C  MATRIX ARE COMPUTED NEXT.
**************************************************
      DO 310 I=1,NT
      DO 300 J=1,NT
      U=0
      DO 290 K=1,NC
  290 U=U+CC(I,K)*CA(J,K)
  300 AAA(I,J)=-AAA(I,J)+PI*U*SD(J)
  310 CONTINUE
      DO 320 I=1,NT
  320 AAA(I,I)=PI+AAA(I,I)
      RETURN
  900 FORMAT(4I5)
  901 FORMAT(6E23.14)
      END
```

```
      PROGRAM MTVELS(TAPE5,INPUT,OUTPUT)
      COMMON/BT1/AA(80),BB(80),CC(80,2)
     2 /BT2/TI(2),SD(80),S1(80),S2(80),S3(80),X(80),
     3 X1(80),X2(80),X3(80),Y(80),Y1(80),Y2(80),Y3(80),X1P(80),
     4 X2P(80),X3P(80),Y1P(80),Y2P(80),Y3P(80)
     5 /BT3/DAA(80),DBB(80),DCC(80,2)
     6 /BT4/C(2),XX(2),YY(2),M(2),A(51,2),B(49,2)
     7 /BT5/AAA(80,80),IP(80)
     8 /BT6/TAA(80),TBB(80),TCC(80,2)
     9 /BT7/N(2),NC,NT
      NAMELIST/NAM/LLL/NAM1/LLL/NAM2/LLL/NAM3/LLL/NAM4/LLL/NAM5/
     2 LLL/NAM6/LLL/NAM7/LLL/NAM8/LLL/NAM9/LLL
************************************************************
************************************************************
C  MTVELS IS THE CONTROL PROGRAM FOR DETERMIN-
C  ING THE TANGENTIAL VELOCITIES DUE TO A DISTRIBUTION
C  OF SOURCES ON THE CONTOURS.
C  IN THIS PROGRAM THE LINEAR SYSTEMS OF EQUATIONS
C  ARE SOLVED DIRECTLY.
C  NC              NUMBER OF CONTOURS
C  N(I)            NUMBER OF SUBDIVISIONS OF THE ITH CONTOUR
C  NT              TOTAL NUMBER OF SUBDIVISIONS ALL CONTOURS
C                  NT=SUM N(I), I=1,NC
C  WE HAVE USED FIXED RATHER THAN VARIABLE DIMENSIONS.  IF THE
C  NUMBER OF CONTOURS OR IF THE NUMBER OF SUBDIVISIONS IS IN-
C  CREASED THEN THE DIMENSION STATEMENTS WILL HAVE TO BE CHANGED
C  AND THE PROGRAM MTVELS AND SUBROUTINES  WHICH IT CALLS WILL
C  HAVE TO BE RECOMPILED TO REFLECT THE NEW VALUES OF NT AND NC
C  XX(K),YY(K)     APPROXIMATE "CENTER" OF THE KTH CONTOUR
C  A(L,K),B(L,K)   FOURIER COSINE AND SINE COEFFICIENTS FOR THE
C                  RADIUS VECTOR R FROM XX(K),YY(K) TO A POINT ON
C                  THE KTH CONTOUR.
C  THE CENTERS (XX(K),YY(K)) AND THE FOURIER
C  COEFFICIENTS A(L,K) AND B(L,K) ARE READ
C  FROM THE FILE, TAPE5.  THESE QUANTITIES
C  MUST BE PREPARED AND STORED IN ADVANCE
C  IN THE PROPER FORMAT.
C  AA,BB,CC        FICTITIOUS DENSITY FUNCTIONS RESULTING FROM THE
C                  VELOCITY COMPONENTS AT INFINITY AND CIRCULATION
C                  ABOUT EACH CONTOUR
C  DAA,DBB,DCC     DERIVATIVES OF THE DENSITY FUNCTIONS W.R.T. THE
C                  INTEGRATION VARIABLE THETA
C  TAA,TBB,TCC     THE CONTRIBUTIONS TO THE TANGENTIAL VELOCITY
C  X(I),Y(I)       COORDINATES OF THE ITH STATION ON THE CONTOURS
C                  X1(I),Y1(I) ARE THE FIRST DERIVATIVES W.R.T. THE
C                  CENTRAL ANGLE THETA AND SO ON
C                  X1P(I),Y1P(I) ARE THE FIRST DERIVATIVES W.R.T.
C                  ARC LENGTH AND SO ON
C  S1,S2,S3        DENOTE THE 1ST,2ND ETC,DERIVATIVES OF THE
C                  ARC LENGTH W.R.T. THE CENTRAL ANGLE
C  SD              CHANGE OF INTEGRATION VARIABLE FACTOR
C  MTVELS CALLS THE FOLLOWING SUBPROGRAMS
C  MVALS   THIS SUBROUTINE USES THE QUANTITIES IN COMMON BLOCK 4
C          TO COMPUTE THE QUANTITIES IN COMMON BLOCKS 1 AND 2
```

```
C  MDENS    THIS SUBROUTINE USES THE VALUES IN COMMON BLOCK 2 TO
C          COMPUTE THE MATRIX ENTRIES FOR DETERMINING THE DEN-.
C          SITIES.  THE MATRIX ENTRIES ARE STORED COMMON BLOCK 5.
C  LU3      A ROUTINE FOR THE DIRECT SOLUTION
C          OF SYSTEMS OF LINEAR EQUATIONS TAKEN
C          FROM THE ARL LINEAR ALGEBRA LIBRARY.
C          LEQS3 IS INCLUDED IN LU3
C  MDDENS   THIS SUBROUTINE COMPUTES THE DERIVATIVES OF THE DENSITIES
C  MTVS     THIS SUBROUTINE COMPUTES THE CONTRIBUTIONS TAA,TBB AND TCC
C          TO THE TANGENTIAL VELOCITIES.
C  THE INPUT DATA TO RUN THIS PROGRAM IS AS FOLLOWS
C  NOTE-THE LETTER B DENOTES A BLANK IN THE INPUT DATA.
C  *EOR
C  BBBNC
C  BBBN(1)
C  ETC.
C  BBBN(NC)
C  *EOR
**************************************************************
**************************************************************
      LLL=1
      PI=3.14159265359979
      READ 900,NC
      NT=0
      DO 10 I=1,NC
      READ 900,N(I)
10    NT=NT+N(I)
      READ(5,NAM4)
      DO 20 J=1,NC
      READ(5,900)M(J)
      M1P=M(J)+1
      M1M=M(J)-1
      READ(5,901)XX(J),YY(J)
      READ(5,901)(A(I,J),I=1,M1P)
20    READ(5,901)(B(I,J),I=1,M1M)
      CALL MVALS
      CALL MDENS
      CALL LU3(NT,AAA,80,IP)
      CALL LEQS3(NT,AAA,80,IP,AA)
      CALL LEQS3(NT,AAA,80,IP,BB)
      DO 25 J=1,NC
25    CALL LEQS3(NT,AAA,80,IP,CC(1,J))
      READ(5,NAM5)
      WRITE(5,900) NT
      WRITE(5,901)(AA(I),I=1,NT)
      WRITE(5,900)NT
      WRITE(5,901)(BB(I),I=1,NT)
      DO 30 J=1,NC
      WRITE(5,900)N(J)
30    WRITE(5,901)(CC(I,J),I=1,NT)
      WRITE(5,NAM6)
      CALL MDDENS
      WRITE(5,900)NT
      WRITE(5,901)(DAA(I),I=1,NT)
```

AD-A042 984 AIR FORCE FLIGHT DYNAMICS LAB WRIGHT-PATTERSON AFB OHIO F/G 20/4
INTEGRAL EQUATION METHODS FOR TWO DIMENSIONAL INCOMPRESSIBLE FL--ETC(U)
APR 77 C L KELLER
UNCLASSIFIED AFFDL-TR-77-27 NL

2 OF 2
AD
A042 984

END
DATE
FILMED
9 -77
DDC

```
      WRITE(5,900)NT
      WRITE(5,901)(DBB(I),I=1,NT)
      DO 40 J=1,NC
      WRITE(5,900)N(J)
40    WRITE(5,901)(DCC(I,J),I=1,NT)
      WRITE(5,NAM7)
      CALL MTVS
      WRITE(5,900)NT
      WRITE(5,901)(TAA(I),I=1,NT)
      WRITE(5,900)NT
      WRITE(5,901)(TBB(I),I=1,NT)
      DO 50 J=1,NC
      WRITE(5,900)N(J)
50    WRITE(5,901)(TCC(I,J),I=1,NT)
      WRITE(5,NAM8)
      STOP
900   FORMAT(I5)
901   FORMAT(5E23.14)
      END



      SUBROUTINE MVALS
      COMMON/BT1/AA(80),BB(80),CC(80,2)
     2 /BT2/TI(2),SD(80),S1(80),S2(80),S3(80),X(80),
     3 X1(80),X2(80),X3(80),Y(80),Y1(80),Y2(80),Y3(80),X1P(80),
     4 X2P(80),X3P(80),Y1P(80),Y2P(80),Y3P(80)
     6 /BT4/C(2),XX(2),YY(2),M(2),A(51,2),B(49,2)
     8 /BT7/N(2),NC,NT
      REAL MFOU,MDFOU,MD2FOU,MD3FOU
**************************************************
C  THIS SUBROUTINE PREPARES TABLES OF VALUES TO BE USED
C  IN LATER COMPUTATIONS.  SPECIFICALLY, IT COMPUTES THE
C  COORDINATES AT STATIONS ALONG THE CONTOUR C, THE
C  DERIVATIVES OF THESE COORDINATES W.R.T. THE
C  POLAR VARIABLE THETA=T, AND WITH RESPECT TO ARC
C  LENGTH.  THESE ARE DENOTED E.G., BY X1 AND X1P RESP.
C  IT ALSO COMPUTES THE DERIVATIVES OF THE ARC LENGTH
C  W.R.T. THE POLAR VARIABLE.  THESE DERIVATIVES ARE
C  DENOTED BY S1,S2, ETC.
C  THIS SUBROUTINE USES THE FOLLOWING REAL
C  FUNCTION PROGRAMS.
C  MFOU      EVALUATES A TRIGONOMETRIC POLYNOMIAL WITH
C            FOURIER COEFFICIENTS A(L,K) AND B(L,K) AT
C            A GIVEN ANGLE (IN RADIANS).
C  MDFOU     THE DERIVATIVE OF MFOU W.R.T. THETA.
C  MD2FOU    THE DERIVATIVE OF MDFOU W.R.T. THETA.
C  MD3FOU    THE DERIVATIVE OF MD2FOU W.R.T. THETA.
**************************************************
      PI=3.14159265358898
      T=0.
      II=1
      IL=0
```

```
      DO 35 J=1,NC
      IL=IL+N(J)
      TI(J)=2.*PI/N(J)
      DO 30 I=II,IL
      R=MFOU(T,M(J),A,B,J)
      R1=MDFOU(T,M(J),A,B,J)
      R2=MD2FOU(T,M(J),A,B,J)
      R3=MD3FOU(T,M(J),A,B,J)
      S1(I)=SQRT(R**2+R1**2)
      SD(I)=S1(I)*TI(J)
      U=R+R2
      V=R1*U
      S2(I)=V/S1(I)
      W=R2*U+R1*(R1+R3)
      D2=S1(I)**2
      S3(I)=(S1(I)*W-V*S2(I))/D2
      CO=COS(T)
      SI=SIN(T)
      X(I)=XX(J)+CO*R
      Y(I)=YY(J)+SI*R
      X1(I)=CO*R1-R*SI
      Y1(I)=SI*R1+R*CO
      X2(I)=CO*R2-2*SI*R1-CO*R
      Y2(I)=SI*R2+2*CO*R1-SI*R
      X3(I)=R3*CO-3.*R2*SI-3.*R1*CO+SI*R
      Y3(I)=R3*SI+3.*R2*CO-3.*R1*SI-CO*R
      D3=S1(I)*D2
      D5=D2*D3
      X1P(I)=X1(I)/S1(I)
      Y1P(I)=Y1(I)/S1(I)
      U=S1(I)*X2(I)-X1(I)*S2(I)
      X2P(I)=U/D3
      V=S1(I)*Y2(I)-Y1(I)*S2(I)
      Y2P(I)=V/D3
      Z=S1(I)*(S1(I)*X3(I)-X1(I)*S3(I))
      W=S1(I)*(S1(I)*Y3(I)-Y1(I)*S3(I))
      X3P(I)=(Z-3.*U*S2(I))/D5
      Y3P(I)=(W-3.*V*S2(I))/D5
      AA(I)=Y1P(I)
      BB(I)=X1P(I)
      DO 20 K=1,NC
      U=(XX(K)-X(I))*X1P(I)+(YY(K)-Y(I))*Y1P(I)
      U=-U
      V=(XX(K)-X(I))**2+(YY(K)-Y(I))**2
   20 CC(I,K)=U/V
      T=T+TI(J)
   30 CONTINUE
      II=II+N(J)
      T=0.
   35 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MDENS
      COMMON/BT2/TI(2),SD(80),S1(80),S2(80),S3(80),X(30),
     3 X1(80),X2(80),X3(80),Y(80),Y1(80),Y2(80),Y3(80),X1P(80),
     4 X2P(80),X3P(80),Y1P(80),Y2P(80),Y3P(80)
     6 /BT4/C(2),XX(2),YY(2),M(2),A(51,2),B(49,2)
     7 /BT5/AAA(80,80),IP(80)
     9 /BT7/N(2),NC,NT
*************************************************
C  THIS SUBROUTINE COMPUTES THE COEFFICIENT MATRIX FOR
C  DETERMINING THE SOURCE DENSITIES.
*************************************************
      PI=3.14159265358979
      I2=NT-1
      DO 30 I=1,I2
      J1=I+1
      DO 20 J=J1,NT
      U=(X(I)-X(J))*Y1P(I)-(Y(I)-Y(J))*X1P(I)
      V=(X(I)-X(J))**2+(Y(I)-Y(J))**2
20    AAA(I,J)=(U/V)*SD(J)
30    CONTINUE
      DO 50 I=2,NT
      J2=I-1
      DO 40 J=1,J2
      U=(X(I)-X(J))*Y1P(I)-(Y(I)-Y(J))*X1P(I)
      V=(X(I)-X(J))**2+(Y(I)-Y(J))**2
40    AAA(I,J)=(U/V)*SD(J)
50    CONTINUE
      DO 60 I=1,NT
      U=Y2(I)*X1P(I)-X2(I)*Y1P(I)
      V=X1(I)**2+Y1(I)**2
60    AAA(I,I)=PI+0.5*(U/V)*SD(I)
      RETURN
      END



      SUBROUTINE MDDENS
      DIMENSION SM3(2)
      COMMON/BT1/AA(80),BB(80),CC(80,2)
     2 /BT2/TI(2),SD(80),S1(80),S2(80),S3(80),X(80),
     3 X1(80),X2(80),X3(80),Y(80),Y1(80),Y2(80),Y3(80),X1P(80),
     4 X2P(80),X3P(80),Y1P(80),Y2P(80),Y3P(80)
     5 /BT3/DAA(80),DBB(80),DCC(80,2)
     6 /BT4/C(2),XX(2),YY(2),M(2),A(51,2),B(49,2)
     9 /BT7/N(2),NC,NT
*************************************************
C    THIS SUBROUTINE COMPUTES THE DERIVATVES OF THE
C    DENSITIY FUNCTIONS W.R.T. THE INTEGRATION
C    VARIABLE T (= THETA)
*************************************************
      PI=3.14159265358979
      SM1=SM2=0
      DO 5 K=1,NC
```

```
5       SM3(K)=0
        I=1
        GO TO 30
 10     SM1=SM2=0
        DO 15 K=1,NC
15      SM3(K)=0
        J2=I-1
        DO 20 J=1,J2
        U=X(I)-X(J)
        V=Y(I)-Y(J)
        Z=U*U+V*V
        W=Z*(U*Y2P(I)-V*X2P(I))
        W=W-2.*(U*Y1P(I)-V*X1P(I))*(U*X1P(I)+V*Y1P(I))
        W=(W*SD(J))/(Z*Z)
        SM1=SM1+W*AA(J)
        SM2=SM2+W*BB(J)
        DO 17 K=1,NC
17      SM3(K)=SM3(K)+W*CC(J,K)
 20     CONTINUE
        IF(I-NT) 30,50,50
 30     J1=I+1
        DO 40 J=J1,NT
        U=X(I)-X(J)
        V=Y(I)-Y(J)
        Z=U*U+V*V
        W=Z*(U*Y2P(I)-V*X2P(I))
        W=W-2.*(U*Y1P(I)-V*X1P(I))*(U*X1P(I)+V*Y1P(I))
        W=(W*SD(J))/(Z*Z)
        SM1=SM1+W*AA(J)
        SM2=SM2+W*BB(J)
        DO 35 K=1,NC
35      SM3(K)=SM3(K)+W*CC(J,K)
 40     CONTINUE
 50     Z=X1P(I)**2+Y1P(I)**2
        W1=X1P(I)*Y3P(I)-Y1P(I)*X3P(I)
        U=X1P(I)*Y2P(I)-Y1P(I)*X2P(I)
        V=X1P(I)*X2P(I)+Y1P(I)*Y2P(I)
        W=SD(I)*(2.*Z*W1-3.*U*V)/(6.*Z*Z)
*****************************************************
C    THE VALUES OF THE INTEGRALS ARE
*****************************************************
        SM1=(SM1+W*AA(I))
        SM2=(SM2+W*BB(I))
        DO 52 K=1,NC
52      SM3(K)=(SM3(K)+W*CC(I,K))
        D=S1(I)**2
        W1=(S1(I)*Y2(I)-Y1(I)*S2(I))/D
        W2=(S1(I)*X2(I)-X1(I)*S2(I))/D
        DAA(I)=(W1-S1(I)*SM1)/PI
        DBB(I)=(W2-S1(I)*SM2)/PI
        DO 55 K=1,NC
        Z=(X(I)-XX(K))**2+(Y(I)-YY(K))**2
        Z1=2.*((X(I)-XX(K))*X1(I)+(Y(I)-YY(K))*Y1(I))
        U=(Y(I)-YY(K))*Y1P(I)+(X(I)-XX(K))*X1P(I)
```

```
      V=(Y(I)-YY(K))*(S1(I)*Y2(I)-Y1(I)*S2(I))
      V=V+(X(I)-XX(K))*(S1(I)*X2(I)-X1(I)*S2(I))
      U1=(V/D)+Y1(I)*Y1P(I)+X1(I)*X1P(I)
      W1=(Z*U1-U*Z1)/(Z*Z)
55    DCC(I,K)=(W1-S1(I)*SM3(K))/PI
      I=I+1
      IF (I-NT) 10,10,60
 60   RETURN
      END



      SUBROUTINE MTVS
      DIMENSION SM3(2),W3(2)
      COMMON/BT1/AA(80),BB(80),CC(80,2)
     2 /BT2/TI(2),SD(80),S1(80),S2(80),S3(80),X(80),
     3 X1(80),X2(80),X3(80),Y(80),Y1(80),Y2(80),Y3(80),X1P(80),
     4 X2P(80),X3P(80),Y1P(80),Y2P(80),Y3P(80)
     5 /BT3/DAA(80),DBB(80),DCC(80,2)
     6 /BT4/C(2),XX(2),YY(2),M(2),A(51,2),B(49,2)
     8 /BT6/TAA(80),TBB(80),TCC(80,2)
     9 /BT7/N(2),NC,NT
      PI=3.14159265358979
*****************************************************************
C  THIS SUBROUTINE COMPUTES THE CONTRIBUTIONS TAA,TBB
C  AND TCC TO THE TANGENTIAL VELOCITIES AT STATIONS
C  AROUND THE CONTOURS.
C  FIRST WE COMPUTE THE VALUE OF THE INTEGRAND FI AT THE
C  STATIONS J NOT EQUAL TO I.  THE COMPUTATION OF
C  FI SEEMS TO SEPARATE NATURALLY INTO 3 CASES: I=1
C  1 L.T. I L.T. N AND I=N.  THE COMPUTATION OF FI
C  IS FOLLOWED BY THE EVALUATION OF THE INTEGRAL.
      I=1
      SM1=SM2=0
      DO 10 K=1,NC
10    SM3(K)=0
      DO 50 J=2,NT
      U=(X(I)-X(J))*X1P(I)+(Y(I)-Y(J))*Y1P(I)
      V=(X(I)-X(J))**2+(Y(I)-Y(J))**2
      FI=SD(J)*(U/V)
      SM1=SM1+AA(J)*FI
      SM2=SM2+BB(J)*FI
      DO 20 K=1,NC
20    SM3(K)=SM3(K)+CC(J,K)*FI
 50   CONTINUE
      GO TO 250
 60   SM1=SM2=0
      DO 62 K=1,NC
62    SM3(K)=0
      J2=I-1
      DO 70 J=1,J2
      U=(X(I)-X(J))*X1P(I)+(Y(I)-Y(J))*Y1P(I)
      V=(X(I)-X(J))**2+(Y(I)-Y(J))**2
```

```
      FI=SD(J)*(U/V)
      SM1=SM1+AA(J)*FI
      SM2=SM2+BB(J)*FI
      DO 68 K=1,NC
68    SM3(K)=SM3(K)+CC(J,K)*FI
 70   CONTINUE
      J1=I+1
      DO 80 J=J1,NT
      U=(X(I)-X(J))*X1P(I)+(Y(I)-Y(J))*Y1P(I)
      V=(X(I)-X(J))**2+(Y(I)-Y(J))**2
      FI=SD(J)*(U/V)
      SM1=SM1+AA(J)*FI
      SM2=SM2+BB(J)*FI
      DO 78 K=1,NC
78    SM3(K)=SM3(K)+CC(J,K)*FI
 80   CONTINUE
      GO TO 250
 90   SM1=SM2=0
      DO 92 K=1,NC
92    SM3(K)=0
      J2=NT-1
      DO 100 J=1,J2
      U=(X(I)-X(J))*X1P(I)+(Y(I)-Y(J))*Y1P(I)
      V=(X(I)-X(J))**2+(Y(I)-Y(J))**2
      FI=SD(J)*(U/V)
      SM1=SM1+AA(J)*FI
      SM2=SM2+BB(J)*FI
      DO 98 K=1,NC
98    SM3(K)=SM3(K)+CC(J,K)*FI
100   CONTINUE
      GO TO 250
C  COMPUTE INTEGRAND AT ITH STATION HERE.
250   Z=X1(I)**2+Y1(I)**2
      U=(X1(I)*X1P(I)+Y1(I)*Y1P(I))/Z
      V=X1(I)*X2(I)+Y1(I)*Y2(I)
      W=0.5*(X1P(I)*X2(I)+Y1P(I)*Y2(I))
      W1=-U*(S1(I)*DAA(I)+AA(I)*S2(I))
      W2=-U*(S1(I)*DBB(I)+BB(I)*S2(I))
      W1=W1+((AA(I)*S1(I))*(U*V-W)/Z)
      W2=W2+((BB(I)*S1(I))*(U*V-W)/Z)
      DO 260 K=1,NC
      W3(K)=-U*(S1(I)*DCC(I,K)+CC(I,K)*S2(I))
260   W3(K)=W3(K)+((CC(I,K)*S1(I))*(U*V-W)/Z)
C  THE VALUES OF THE INTEGRALS ARE GIVEN BY
      SM1=SM1+W1*SD(I)/S1(I)
      SM2=SM2+W2*SD(I)/S1(I)
      DO 270 K=1,NC
270   SM3(K)=SM3(K)+W3(K)*SD(I)/S1(I)
C  THE TANGENTIAL VELOCITIES ARE THEN
      DO 280 K=1,NC
      U=(X(I)-XX(K))*Y1P(I)-(Y(I)-YY(K))*X1P(I)
      V=(X(I)-XX(K))**2+(Y(I)-YY(K))**2
280   TCC(I,K)=(U/V)+SM3(K)
      TAA(I)=X1P(I)-SM1
```

```
      TBB(I)=Y1P(I)+SM2
      I=I+1
      IF(NT-I) 300,30,60
300   RETURN
901   FORMAT(5E23.14)
      END



      PROGRAM MTVLSI(TAPE5,INPUT,OUTPUT)
      COMMON/BT1/AA(80),BB(80),CC(80,2)
     2 /BT2/TI(2),SJ(80),S1(80),S2(80),S3(80),X(80),
     3 X1(80),X2(80),X3(80),Y(80),Y1(80),Y2(80),Y3(80),X1P(80),
     4 X2P(80),X3P(80),Y1P(80),Y2P(80),Y3P(80)
     5 /BT3/DAA(80),DBB(80),DCC(80,2)
     6 /BT4/C(2),XX(2),YY(2),M(2),A(51,2),B(49,2)
     7 /BT5/AAA(80,80),IP(80)
     8 /BT6/TAA(80),TBB(80),TCC(80,2)
     9 /BT7/N(2),NC,NT
      NAMELIST/NAM/LLL/NAM1/LLL/NAM2/LLL/NAM3/LLL/NAM4/LLL/NAM5/
     2 LLL/NAM6/LLL/NAM7/LLL/NAM8/LLL/NAM9/LLL
*************************************************************
*************************************************************
C  MTVLSI IS THE CONTROL PROGRAM FOR DETERMIN-
C  ING THE TANGENTIAL VELOCITIES DUE TO A
C  DISTRIBUTION OF SOURCES ON THE CONTOURS.
C  IN THIS PROGRAM THE LINEAR SYSTEMS OF EQUATONS
C  ARE SOLVED BY AN ITERATIVE PROCESS.
C  NC               NUMBER OF CONTOURS
C  N(I)             NUMBER OF SUBDIVISIONS OF THE ITH CONTOUR
C  NT               TOTAL NUMBER OF SUBDIVISIONS ALL CONTOURS
C                   NT=SUM N(I), I=1,NC
C  WE HAVE USED FIXED RATHER THAN VARIABLE DIMENSIONS.  IF THE
C  NUMBER OF CONTOURS OR IF THE NUMBER OF SUBDIVISIONS IS IN-
C  CREASED THEN THE DIMENSION STATEMENTS WILL HAVE TO BE CHANGED
C  AND THE PROGRAM MTVELS AND SUBROUTINES  WHICH IT CALLS WILL
C  HAVE TO BE RECOMPILED TO REFLECT THE NEW VALUES OF NT AND NC
C  XX(K),YY(K)      APPROXIMATE "CENTER" OF THE KTH CONTOUR
C  A(L,K),B(L,K)    FOURIER COSINE AND SINE COEFFICIENTS FOR THE
C                   RADIUS VECTOR R FROM XX(K),YY(K) TO A POINT ON
C                   THE KTH CONTOUR.
C  THE CENTERS (XX(K),YY(K)) AND THE FOURIER
C  COEFFICIENTS A(L,K) AND B(L,K) ARE READ FROM
C  THE FILE, TAPE5.  THESE QUANTITIES MUST BE PREPARED
C  AND STORED IN ADVANCE IN THE PROPER FORMAT.
C  AA,BB,CC         FICTITIOUS DENSITY FUNCTIONS RESULTING FROM THE
C                   VELOCITY COMPONENTS AT INFINITY AND CIRCULATION
C                   ABOUT EACH CONTOUR
C  DAA,DBB,DCC      DERIVATIVES OF THE DENSITY FUNCTIONS W.R.T. THE
C                   INTEGRATION VARIABLE THETA
C  TAA,TBB,TCC      THE CONTRIBUTIONS TO THE TANGENTIAL VELOCITY
C  X(I),Y(I)        COORDINATES OF THE ITH STATION ON THE CONTOURS
C                   X1(I),Y1(I) ARE THE FIRST DERIVATIVES W.R.T. THE
```

```
C                    CENTRAL ANGLE THETA AND SO ON
C                    X1P(I),Y1P(I) ARE THE FIRST DERIVATIVES W.R.T.
C                    ARC LENGTH AND SO ON
C  S1,S2,S3          DENOTE THE 1ST,2ND,ETC,DERIVATIVES OF THE
C                    ARC LENGTH W.R.T. THE CENTRAL ANGLE
C  SD                CHANGE OF INTEGRATION VARIABLE FACTOR
C  MTVLSI CALLS THE FOLLOWING SUBPROGRAMS
C  MVALS   THIS SUBROUTINE USES THE QUANTITIES IN COMMON BLOCK 4 TO
C          COMPUTE THE QUANTITIES IN COMMON BLOCKS 1 AND 2
C  MDENSI  THIS SUBROUTINE USES THE VALUES IN COMMON BLOCK 2 TO
C          COMPUTE THE MATRIX ENTRIES FOR DETERMINING THE DENSITIES.
C          THE MATRIX ENTRIES ARE STORED IN COMMON BLOCK 5.
C  SITER   THIS SUBROUTINE SOLVES THE SYSTEM OF LINEAR
C          EQUATIONS FOR THE DENSITIES BY AN ITERATION
C          PROCESS.  THIS ROUTINE IS APPLICABLE TO LINEAR
C          SYSTEMS OF EQUATIONS OF A PARTICULAR TYPE ONLY.
C  MDDENS  THIS SUBROUTINE COMPUTES THE DERIVATIVES OF THE DENSITIES
C  MTVS    THIS SUBROUTINE COMPUTES THE CONTRIBUTIONS TAA,TBB AND TCC
C          TO THE TANGENTIAL VELOCITIES.
C  THE INPUT DATA TO RUN THIS PROGRAM IS AS FOLLOWS
C  NOTE-THE LETTER B DENOTES A BLANK IN THE INPUT DATA.
C  BBBNC
C  BBBN(1)
C  ETC.
C  BBBN(NC)
*****************************************************
*****************************************************
      LLL=1
      PI=3.14159265358979
      READ 900,NC
      NT=0
      DO 10 I=1,NC
      READ 900,N(I)
10    NT=NT+N(I)
      READ(5,NAM4)
      DO 20 J=1,NC
      READ(5,900)M(J)
      M1P=M(J)+1
      M1M=M(J)-1
      READ(5,901)XX(J),YY(J)
      READ(5,901)(A(I,J),I=1,M1P)
20    READ(5,901)(B(I,J),I=1,M1M)
      CALL MVALS
      CALL MDENSI
      READ(5,NAM5)
      CALL SITER(NT,K,EM,AAA,AA,L)
      IF (L) 21,21,22
 21   PRINT 902,EM
      GO TO 100
 22   WRITE(5,900)NT,K
      WRITE(5,901)EM
      WRITE(5,901)(AA(I),I=1,NT)
      CALL SITER(NT,K,EM,AAA,BB,L)
      IF(L) 23,23,24
```

```
 23   PRINT 902,EM
      GO TO 100
 24   WRITE(5,900)NT,K
      WRITE(5,901)EM
      WRITE(5,901)(BB(I),I=1,NT)
      DO 30 J=1,NC
      CALL SITER(NT,K,EM,AAA,CC(1,J),L)
      IF(L) 26,26,27
 26   PRINT 902,EM
      GO TO 100
27    WRITE(5,900)N(J),K
      WRITE(5,901)EM
      WRITE(5,901)(CC(I,J),I=1,NT)
 30   CONTINUE
      WRITE(5,NAM6)
      CALL MDENS
      WRITE(5,900)NT
      WRITE(5,901)(DAA(I),I=1,NT)
      WRITE(5,900)NT
      WRITE(5,901)(DBB(I),I=1,NT)
      DO 40 J=1,NC
         E(5,900)N(J)
40       E(5,901)(DCC(I,J),I=1,NT)
         E(5,NAM7)
      CALL MTVS
      WRITE(5,900)NT
      WRITE(5,901)(TAA(I),I=1,NT)
      WRITE(5,900)NT
      WRITE(5,901)(TBB(I),I=1,NT)
      DO 50 J=1,NC
      WRITE(5,900)N(J)
50    WRITE(5,901)(TCC(I,J),I=1,NT)
      WRITE(5,NAM8)
100   STOP
900   FORMAT(4I5)
901   FORMAT(6E23.14)
902   FORMAT(*0EM=*E23.14,*, ITERATION PROCESS DID NOT CONVERGE*)
      END



      SUBROUTINE MDENSI
      COMMON/BT2/TI(2),S0(80),S1(80),S2(80),S3(80),X(80),
     3 X1(80),X2(80),X3(80),Y(80),Y1(80),Y2(80),Y3(80),X1P(80),
     4 X2P(80),X3P(80),Y1P(80),Y2P(80),Y3P(80)
     6 /BT4/C(2),XX(2),YY(2),M(2),A(51,2),B(49,2)
     7 /BT5/AAA(80,80),IP(80)
     9 /BT7/N(2),NC,NT
C*****************************************************
C  THIS SUBROUTINE COMPUTES THE MATRIX USED FOR
C  DETERMINING THE SOURCE DENSITIES.
C*****************************************************
      PI=3.14159265358979
```

```
      I2=NT-1
      DO 30 I=1,I2
      J1=I+1
      DO 20 J=J1,NT
      U=(X(I)-X(J))*Y1P(I)-(Y(I)-Y(J))*X1P(I)
      V=(X(I)-X(J))**2+(Y(I)-Y(J))**2
20    AAA(I,J)=(U/V)*SD(J)
30    CONTINUE
      DO 50 I=2,NT
      J2=I-1
      DO 40 J=1,J2
      U=(X(I)-X(J))*Y1P(I)-(Y(I)-Y(J))*X1P(I)
      V=(X(I)-X(J))**2+(Y(I)-Y(J))**2
40    AAA(I,J)=(U/V)*SD(J)
50    CONTINUE
      DO 60 I=1,NT
      U=Y2(I)*X1P(I)-X2(I)*Y1P(I)
      V=X1(I)**2+Y1(I)**2
60    AAA(I,I)=0.5*(U/V)*SD(I)
      RETURN
      END



      SUBROUTINE SITER(N,K,EM,A,B,L)
      DIMENSION A(80,80),B(80),V(80),BC(80)
**********************************************************
C    THIS SUBROUTINE SOLVES THE LINEAR SYSTEM OF EQUATIONS
C    (PI*I+A)X=B  BY AN ITERATION PROCESS.  HERE PI IS AN EIGENVALUE
C    OF THE MATRIX A.  FIRST, REWRITE THE SYSTEM OF EQUATIONS IN THE
C    FORM (I+A/PI)X=B/PI AND SET AC=A/PI AND BD=B/PI.  THE MATRIX AC
C    HAS 1 FOR AN EIGENVALUE.  NEXT WE WANT TO REMOVE FROM THE VECTOR
C    BD ANY EIGENVECTORS CORRESPONDING TO THE EIGENVALUE 1.  THIS IS
C    DONE BY WRITING X=(BD/2)+XC.ONE OBTAINS THE EQUATION (I+AC)XC=BC,
C    WHERE BC=(BD-AC*BD)/2,WHICH MUST BE SOLVED FOR XC.  ONE HAS THEN
C    XC=BC-AC*BC+AC*(AC*BD)-+ETC., AND FINALLY ONE OBTAINS X.
C SITER CALLS THE FOLLOWING SUBPROGRAM:
C  SCV      THIS SUBROUTINE GIVES THE PRODUCT OF A
C           VECTOR AND A SCALAR
C  MATV     THIS SUBROUTINE GIVES THE PRODUCT OF A VECTOR
C           BY A MATRIX AND DIVIDES THE RESULTING VECTOR
C           BY PI.
**********************************************************
      PI=3.1415926535898
      C=1/PI
      CALL SCV(N,C,B)
      DO 5 I=1,N
 5    V(I)=B(I)
      CALL MATV(N,A,V)
      DO 10 I=1,N
      V(I)=0.5*(B(I)-V(I))
 10   BC(I)=V(I)
      K=1
```

```
      SI=-1
 15   CALL MATV(N,A,V)
      DO 20 I=1,N
 20   BC(I)=BC(I)+SI*V(I)
      SI=-SI
      EM=0
      DO 30 I=1,N
      U=ABS(V(I))
      E=AMAX1(EM,U)
 30   EM=E
      IF (1.E-08-EM) 40,40,70
 40   IF(20-K) 60,60,50
 50   K=K+1
      GO TO 15
 60   L=0
      GO TO 90
 70   L=1
      DO 80 I=1,N
 80   B(I)=0.5*B(I)+BC(I)
 90   RETURN
      END



      REAL FUNCTION MFOU(X,N,A,B,L)
      DIMENSION A(51,2),B(49,2)
C**** **** **** **** **** **** **** **** **** **** **** **** ****
C**** **** **** **** **** **** **** **** **** **** **** **** ****
C     THIS PROGRAM EVALUATES THE TRIGONOMETRIC INTERPOLATING
C     POLYNOMIAL AT A GIVEN POINT X.
C**** **** **** **** **** **** **** **** **** **** **** **** ****
      MFOU=(A(1,L)+A(N+1,L)*COS(N*X))*0.5
      NM=N-1
      F=0
      DO 15 K=1,NM
15    F=F+A(K+1,L)*COS(K*X)+B(K,L)*SIN(K*X)
      MFOU=MFOU+F
      RETURN
      END



      REAL FUNCTION MDFOU(X,N,A,B,L)
      DIMENSION A(51,2),B(49,2)
C**** **** **** **** **** **** **** **** **** **** **** **** ****
C**** **** **** **** **** **** **** **** **** **** **** **** ****
C     THIS PROGRAM EVALUATES THE DERIVATIVE OF THE INTERPOLATING
C     POLYNOMIAL AT A GIVEN VALUE X.
C**** **** **** **** **** **** **** **** **** **** **** **** ****
      NM=N-1
      NP=N+1
      F=-.5*N*A(NP,L)*SIN(N*X)
```

```
      DO 15 K=1,NM
15    F=F-K*A(K+1,L)*SIN(K*X)+K*B(K,L)*COS(K*X)
      MDFOJ=F
      RETURN
      END
      REAL FUNCTION MD2FOU(X,N,A,B,L)
      DIMENSION A(51,2),B(49,2)
**********************************************
**********************************************
C     THIS PROGRAM EVALUATES THE SECOND
C     DERIVATIVE OF THE INTERPOLATING
C     POLYNOMIAL AT A GIVEN VALUE X.
**********************************************
      NM=N-1
      NP=N+1
      F=-.5*N*N*A(NP,L)*COS(N*X)
      DO 15 K=1,NM
15    F=F-K*K*(A(K+1,L)*COS(K*X)+B(K,L)*SIN(K*X))
      MD2FOU=F
      RETURN
      END



      REAL FUNCTION MD3FOU(X,N,A,B,L)
      DIMENSION A(51,2),B(49,2)
**********************************************
**********************************************
C     THIS PROGRAM EVALUATES THE THIRD DERI-
C     VATIVE OF THE INTERPOLATING POLYNOMIAL
C     AT A GIVEN VALUE OF X
**********************************************
      NM=N-1
      NP=N+1
      F=.5*(N**3)*A(NP,L)*SIN(N*X)
      DO 15 K=1,NM
15    F=F+(K**3)*(A(K+1,L)*SIN(K*X)-B(K,L)*COS(K*X))
      MD3FOU=F
      RETURN
      END
```

```
      PROGRAM KATR(TAPE5,INPUT,OUTPUT)
      COMMON/K1/XA(50,2),XB(50,2)/K2/TH(70),RO(70),R2(70)
      NAMELIST/NAM/LLL/NAM1/LLL/NAM2/LLL/NAM3/LLL/NAM4/LLL
*****************************************************
*****************************************************
C  THIS IS THE CONTROL PROGRAM FOR TRANSFORMING AN
C  AIRFOIL PROFILE IN TO A NEAR CIRCLE BY MEANS OF
C  A KARMAN TREFFETZ LIKE TRANSFORMATION.
C  THE END PRODUCTS OF THIS PROGRAM ARE THE
C  COORDINATES, IN POLAR FORM OF THE IMAGES
C  OF THE PROFILE DATA ON THE NEAR CIRCLE
C  AND THE SECOND DERIVATIVES AT THESE POINTS
C  OF THE FUNCTION DEFINED BY THESE NEAR
C  CIRCLE POINTS.  THESE QUANTITIES DETERMINE
C  THE CUBIC SPLINE FIT OF THE NEAR CIRCLE.
C  THE RESULT FROM THIS PROGRAM ARE WRITTEN ON
C  A FILE WHICH THE USER MAY NAME AND STORE.
C  NA(NB) - THE NUMBER OF DATA POINTS ON THE UPPER(LOWER)SURFACE.
C  XA(XB) - THE SET OF COORDINATES OF THE DATA POINTS ON THE
C  UPPER(LOWER) SURFACE OF THE AIRFOIL.  THE COORDINATES
C  OF THE LEADING AND TRAILING EDGES ARE
C  INCLUDED IN BOTH SETS.  THESE SETS ARE INDEXED
C  BEGINNING AT THE LEADING EDGE.
C  TH(I)    - THE ARGUMENT OF THE NEAR CIRCLE DATA
C  RO(I)    - THE MAGNITUDE OF THE NEAR CIRCLE DATA
C  R2(I)    - THE SECOND DERIVATIVE OF THE CUBIC
C             SPLINE FIT OF THE NEAR CIRCLE DATA.
C  TA(TB) - THE ANGLE THE UPPER(LOWER) SURFACE MAKES THE X-AXIS
C  AT THE TRAILING EDGE.
C  G  - THE EXPONENT USED IN THE SUB-
C     PROGRAM GP.
C  T  - THE IMAGE IN THE W1-PLANE OF THE POINT
C     AT INFINITY IN THE Z-PLANE IS EXP(IT).
C  THIS PROGRAM CALLS THE FOLLOWING SUBPROGRAMS
C  FL1    COMPUTES W=(1+Z)/(1-Z).  W IS IN POLAR FORM
C         AND Z IS A DATA POINT ON THE SURFACE.
C  REC    CHANGES A SET OF POINTS IN POLAR FORM TO
C         RECTANGULAR FORM.
C  GP     COMPUTES W1=(W*EXP(-I*TA))**G.  W1 IS IN
C         POLAR FORM.  G=PI/(TB-TA).
C  FL2    COMPUTES W3 =(W1+T)/(W1-T).  W3 IS IN POLAR FORM AND T
C         IS THE IMAGE IN THE W1 PLANE OF THE POINT AT INFINITY
C         IN THE Z-PLANE.
C  POLAR CONVERTS RECTANGULAR COORDINATES TO POLAR COORDINATES.
C  ICOF   GIVES A CUBIC SPLINE FIT OF THE RADIUS OF THE NEAR
C         CIRCLE AS A FUNTION OF THE POLAR ANGLE.
C  ICOF CALLS LU3.
C  LU3    A ROUTINE FOR THE DIRECT SOLUTION OF SYSTEMS
C         OF LINEAR EQUATIONS TAKEN FROM THE ARL
C         LINEAR ALGEBRA LIBRARY.  IT INCLUDES LEQS3.
C  THE NEAR CIRCLE IMAGES OF THE PROFILE DATA POINTS ARE INDEXED
C  FROM 1 TO K=NA+NB-1 WITH RO(K)=RO(1) AND TH(K)=TH(1)+2PI.
*************************************************************
*************************************************************
```

```
      LLL=1
      PI=3.1415926535898
      READ 900,NA,NB
      READ 901,((XA(I,J),J=1,2),I=1,NA)
      READ 901,((XB(I,J),J=1,2),I=1,NB)
      WRITE(5,900)NA,NB
      WRITE(5,902)((XA(I,J),J=1,2),I=1,NA)
      WRITE(5,902)((XB(I,J),J=1,2),I=1,NB)
      WRITE(5,NAM)
      WRITE(5,NAM1)
      UA=XA(NA-1,1)+1.
      VA=XA(NA-1,2)
      UB=XB(NB-1,1)+1.
      VB=XB(NB-1,2)
      CALL POLAR(UA,VA,RA,TA)
      CALL POLAR(UB,VB,RB,TB)
      G=PI/(TB-TA)
      CALL FL1(NA,NB)
C  REC IS CALLED TO NORMALIZE THE RESULTS
C  FROM FL1.
      CALL REC(NA,NB)
      CALL GP(NA,NB,G,TA)
      CALL REC(NA,NB)
      T=G*(PI-TA)
      CALL FL2(NA,NB,T)
      K=NA+NB-1
      TH(1)=0.
      TH(K)=2.*PI
      CALL ICOF(K)
      WRITE(5,903)TA,TB,G,T
      WRITE(5,NAM2)
      WRITE(5,900)K
      WRITE(5,903)(TH(I),I=1,K)
      WRITE(5,903)(RO(I),I=1,K)
      WRITE(5,903)(R2(I),I=1,K)
      WRITE(5,NAM3)
      STOP
900   FORMAT(4I5)
901   FORMAT(6E10.0)
902   FORMAT(6E17.8)
903   FORMAT(5E23.14)
      END



      SUBROUTINE FL1(N,M)
      COMMON/K1/XA(50,2),XB(50,2)/K2/TH(70),RO(70),R2(70)
*****************************************************
*****************************************************
C  THIS PROGRAM COMPUTES W=(1+Z)/(1-Z) AT A GIVEN SET
C  OF POINTS Z.  Z IS COMPLEX.
C  W IS IN POLAR FORM,THAT IS,
C  W(J)=RO(J)*EXP(I*TH(J)).  W IS THE
```

```
C  IMAGE OF THE POINTS ON THE UPPER AND
C  LOWER PROFILE SURFACES.  THESE IMAGE
C  POINTS ARE INDEXED CONSECUTIVELY
C  AROUND THE PROFILE CONTOUR, BEGINNING
C  AT THE LEADING EDGE OF THE UPPER SURFACE.
C  FLI CALLS THE SUBPROGRAM POLAR.
**********************************************************
**********************************************************
      NM=N-1
      DO 10 I=1,NM
      UA=-XA(I,1)-1
      VA=-XA(I,2)
      UB=-UA-2
      VB=-VA
      CALL POLAR(UA,VA,RA,TA)
      CALL POLAR(UB,VB,RB,TB)
      RO(I)=RA/RB
10    TH(I)=TA-TB
      RO(N)=0
      TH(N)=0
      MM=M-1
      NP=N+1
      NPM=N+MM
      J=0
      DO 20 I=NP,NPM
      J=J+1
      UA=-XB(M-J,1)-1
      VA=-XB(M-J,2)
      UB=-UA-2
      VB=-VA
      CALL POLAR(UA,VA,RA,TA)
      CALL POLAR(UB,VB,RB,TB)
      RO(I)=RA/RB
20    TH(I)=TA-TB
      RETURN
      END



      SUBROUTINE GP(N,M,G,TA)
      COMMON/K2/TH(70),RO(70),R2(70)
**********************************************************
**********************************************************
C  COMPUTES W1=(W*EXP(-I*TA))**G.  W1 IS IN RECTANGULAR FORM.
C  GP CALLS POLAR.
**********************************************************
**********************************************************
      J=1
      NM=N-1
5     DO 10 I=J,NM
      CALL POLAR(RO(I),TH(I),R,T)
      RO(I)=R**G
10    TH(I)=(T-TA)*G
```

```
      IF(J-1) 15,15,20
15    J=N+1
      NM=N+M-1
      GO TO 5
20    RETURN
      END



      SUBROUTINE REC(N,M)
      COMMON/K2/TH(70),RO(70),R2(70)
*****************************************************
*****************************************************
C  THIS PROGRAM CONVERTS A SET OF POINTS IN POLAR
C  FORM TO A SET OF POINTS  WITH RECTANGULAR
C  COORDINATES.  THE NEW COORDINATES ARE STORED
C  ON TOP OF THE OLD.
*****************************************************
*****************************************************
      K=N+M-1
      DO 10 I=1,K
      R=RO(I)
      B=COS(TH(I))
      C=SIN(TH(I))
      RO(I)=R*B
10    TH(I)=R*C
      RETURN
      END



      SUBROUTINE FL2(N,M,T)
      COMMON/K2/TH(70),RO(70),R2(70)
*****************************************************
*****************************************************
C  THIS PROGRAM COMPUTES W3=(W1+T)/(W1-T).  W3 IS IN
C  POLAR FORM AND T IS THE IMAGE IN THE W1-PLANE OF
C  THE POINT AT INFINITY IN THE Z-PLANE.
*****************************************************
*****************************************************
      K=N+M-1
      U=COS(T)
      V=SIN(T)
      DO 10 I=1,K
      UA=RO(I)+U
      VA=TH(I)+V
      UB=RO(I)-U
      VB=TH(I)-V
      CALL POLAR(UA,VA,RA,TA)
      CALL POLAR(UB,VB,RB,TB)
      R=RA/RB
      TT=TA-TB
```

```
      X=R*COS(TT)
      Y=R*SIN(TT)
 10   CALL POLAR(X,Y,RO(I),TH(I))
      RETURN
      END



      SUBROUTINE POLAR(X,Y,R,T)
**********************************************************
C  THIS PROGRAM CONVERTS RECTANGULAR COORDINATES TO
C  POLAR COORDINATES.
**********************************************************
**********************************************************
      S=3.1415926535898
      A=X*Y
      IF(A) 15,5,15
5     IF(X) 10,6,11
6     IF(Y) 8,7,9
7     R=0
      T=0
      GO TO 40
8     R=ABS(Y)
      T=(3.*S)/2.
      GO TO 40
9     R=Y
      T=S/2.
      GO TO 40
10    R=ABS(X)
      T=S
      GO TO 40
11    R=X
      T=0.
      GO TO 40
15    R=SQRT(X*X+Y*Y)
      U=ABS(Y/X)
      V=ATAN(U)
      IF (A) 20,30,25
20    IF(X) 21,30,22
21    T=S-V
      GO TO 40
22    T=2.*S-V
      GO TO 40
25    IF (X) 26,30,27
26    T=S+V
      GO TO 40
27    T=V
      GO TO 40
30    PRINT 31
31    FORMAT(*0SUBROUTINE POLAR PRODUCT XY IS BOTH ZERO AND
     1NOT ZERO *)
      STOP
40    RETURN
```

```
      END



      SUBROUTINE ICOF(N)
      DIMENSION A(70,70),IP(70)
      COMMON/K2/X(70),Y(70),Z(70)
      NAMELIST/NAM/LLL/NAM1/LLL/NAM2/LLL/NAM3/LLL/NAM4/LLL
*****************************************************************
*****************************************************************
C  THIS PROGRAM DETERMINES A CUBIC SPLINE FIT FOR A PERIODIC
C  FUNCTION.  THE FORMULA FOR DETERMINING THE VALUE OF THE
C  FUNCTION AT A POINT T DEPENDS UPON THE FUNCTION VALUES AT
C  THE ENDPOINTS OF THE INTERVAL ON WHICH T LIES AND THE
C  VALUES OF THE 2ND DERIVATIVES OF THE FUNCTION AT THESE
C  ENDPOINTS.  THIS PROGRAM DETERMINES THE VALUES OF THE 2ND
C  DERIVATIVES OF THE FUNCTION AT THE NODES WHERE THE
C  FUNCTION VALUES ARE GIVEN.
C  ICOF CALLS THE SUBPROGRAM LU3
C  LU3  IS A ROUTINE FOR THE DIRECT SOLUTION OF
C       OF SYSTEMS OF LINEAR EQUATIONS.
*****************************************************************
*****************************************************************
      LLL=1
      PI=3.14159265358979
      NM=N-1
      DO 10 I=1,NM
      DO 8 J=1,NM
8     A(I,J)=0
10    CONTINUE
      K=N-3
      DO 15 I=1,K
      J=I
      A(I,J)=X(I+1)-X(I)
      A(I,J+1)=2*(X(I+2)-X(I))
      A(I,J+2)=X(I+2)-X(I+1)
15    Z(I)=6*((Y(I+2)-Y(I+1))/A(I,I+2)-(Y(I+1)-Y(I))/A(I,I))
      A(N-2,1)=X(N)-X(N-1)
      A(N-2,N-2)=X(N-1)-X(N-2)
      A(N-2,N-1)=2*(X(N)-X(N-2))
      Z(N-2)=6*((Y(1)-Y(NM))/A(N-2,1)-(Y(NM)-Y(N-2))/A(N-2,N-2))
      A(NM,1)=2*(A(N-2,1)+A(1,1))
      A(NM,2)=A(1,1)
      A(NM,NM)=A(N-2,1)
      Z(NM)=6*((Y(2)-Y(1))/A(1,1)-(Y(1)-Y(NM))/A(N-2,1))
      CALL LU3(NM,A,70,IP)
      CALL LEQS3(NM,A,70,IP,Z)
      Z(N)=Z(1)
      RETURN
900   FORMAT(4I5)
903   FORMAT(5E23.14)
      END
```

REFERENCES

1. Williams, B.R. "An Exact Test Case for the Plane Potential Flow About Two Adjacent Lifting Aerofoils," Sep. 71, unclassified, Royal Aircraft Establishment Farnborough (England), RAE-TR-71197, TRC-BR-27241, AD-894-182.

2. Theodorsen, Theodore. "Theory of Wing Sections of Arbitrary Shape," T.R. No. 411, N.A.C.A., 1931.

3. Horn, J. "Partelle Differentialgleichungen," Berlin: Walter de Gruyter and Co., 1949. pp. 104-134.

4. Lovitt, Wm. V. "Linear Integral Equations," New York: Dover Reprinted Edition, 1950. pp. 85-115.

5. Kellogg, O.D. "Foundations of Potential Theory," New York: Dover Publications, Inc. 1929.

6. Prager, W. "Die Druckverteilung an Körpern in ebener Potentialztrömung," Physik. Zeitschr. XXIX, 1928, pp. 865-869.

7. Fettis, H.E. "Further Remarks Concerning the Relative Accuracy of Simpson's and the Trapezoidal Rule for a Certain Class of Functions." Z. Angew. Math. Mech. Bd. 38 Nr. 3/4 März/April 1958.

8. von Mises, R. "Theory of Flight," New York: Dover Publications, Inc. 1959.

9. Stevens, W.A., Goradia, S.H., and Braden, J.A., "Mathematical Model for Two-dimensional Multi-component Airfoils in Viscous Flow," NASA CR-1843 July 1971 (Lockheed-Georgia).

10. Seebohm, T. and Newman, B.G. "A Numerical Method for Calculating Viscous Flow Round Multiple-Section Aerofoils," The Aeronautical Quarterly, Vol. XXVI, August 1975, Part 3, p. 176.

11. Garrick, I.E. "Potential Flow about Arbitrary Biplane Wing Sections," TR No. 542, N.A.C.A., 1935.