

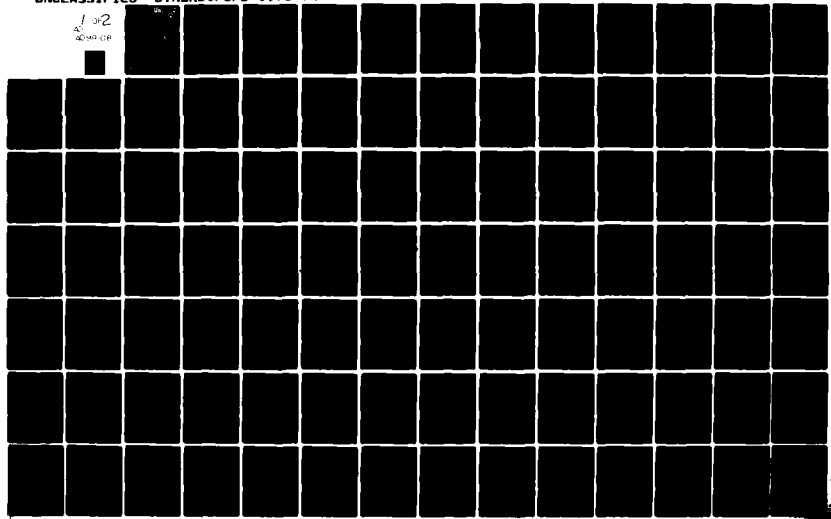
AD-A099 108

DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CE--ETC F/S 20/4
A USER'S MANUAL FOR THE DTNSRDC MOMENTUM INTEGRAL BOUNDARY-LAYE--ETC(U)
DEC 80 T J LANGAN; C VON KERCEK
DTNSRDC/SPD-0972-01

UNCLASSIFIED

NL

1 of 2
AD-A099 108



~~LEVEL~~ II yw 12

AD A099108

DTNSRDC/SPD-0972-01

A USER'S MANUAL FOR THE DTNSRDC MOMENTUM INTEGRAL BOUNDARY-LAYER CODE FOR SHIP HULLS

**DAVID W. TAYLOR NAVAL SHIP
RESEARCH AND DEVELOPMENT CENTER**



Bethesda, Maryland 20884

A USER'S MANUAL FOR THE DTNSRDC MOMENTUM INTEGRAL
BOUNDARY-LAYER CODE FOR SHIP HULLS

by

Thomas J. Langan
Christian von Kerczek

DTIC
ELECTE
MAY 19 1981
S **E**

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

SHIP PERFORMANCE DEPARTMENT
DEPARTMENTAL REPORT

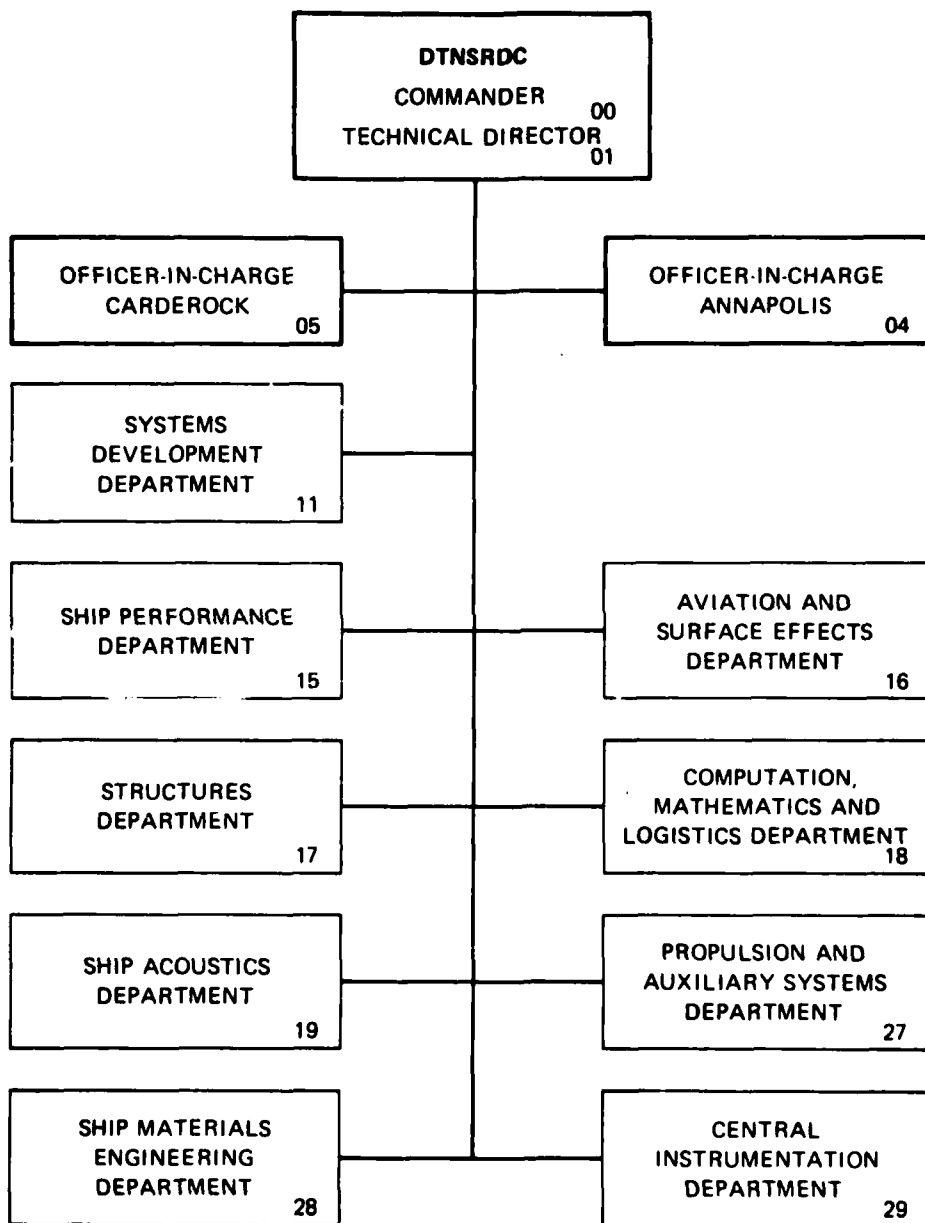
December 1980

DTNSRDC/SPD-0972-01

DTIC FILE COPY

81 3 18 130

MAJOR DTNSRDC ORGANIZATIONAL COMPONENTS



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 14 DTNSRDC/SPD-0972-01	2. GOVT ACCESSION NO. 9 <i>Supplemental Rept.</i>	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) 6 A User's Manual for the DTNSRDC Momentum Integral Boundary-Layer Code for Ship Hulls.		5. TYPE OF REPORT & PERIOD COVERED
7. AUTHOR(s) 10 Thomas J. Langan Christian von Kerczek		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS David W. Taylor Naval Ship Research and Development Center Bethesda, Maryland 20084		8. CONTRACT OR GRANT NUMBER(s)
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Sea Systems Command (SEA-035) Washington, D.C. 20362		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS See Reverse Side
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) 12 <i>164</i>		12. REPORT DATE 11 Dec 1980
		13. NUMBER OF PAGES
		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved For Public Release: Distribution Unlimited 16 <i>SPD-0972-01</i> 17 <i>SR-234-L-1</i>		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Boundary Layers Three-Dimensional Momentum Integral Turbulent		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report provides the instructions for using a suite of computer codes to calculate approximately the three-dimensional boundary layer on ship hulls. There are two sets of codes for computing boundary layers. One is for computing a first-order approximation and is based on a slender-body potential and the small crossflow assumption. The second set of codes uses a general three-dimensional calculation method for the potential flow and solves the three-dimensional momentum integral boundary-layer equations		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

389694

JP

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block #20

without making restrictive assumptions on the magnitude of the crossflow.
An example problem is presented for both methods.

Block #10

Project Element 61153N 23
Project Number SR 02301
Task Area SR 023-01-01
Work Unit 1552-070

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A	

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

Page

LIST OF FIGURES	iv
ABSTRACT	1
ADMINISTRATIVE INFORMATION	1
1. INTRODUCTION	1
2. THE PROGRAMS	4
2.1 MAP22	4
2.2 TBL18	9
2.3 TBL3D	11
2.4 CBL	13
2.5 DOUBDD	13
2.6 TBLIPC	14
2.7 TBL SOL	15
3. OPERATING INSTRUCTIONS WITH AN EXAMPLE	15
3.1 MAP22	16
3.2 TBL18	29
3.3 TBL3D	37
3.4 CBL	42
3.5 DOUBDD	45
3.6 TBLIPC	45
3.7 TBL SOL	47
4. CONCLUDING REMARKS	52
5. REFERENCES	53
APPENDIX - PROGRAM LISTINGS	55

LIST OF FIGURES

Page

1 - Mapping of Unit Circle onto a Section	6
2 - The Bottom and Side Views of the (ϕ, θ) Coordinate Net on the SSPA 720 Model	6

LIST OF TABLES

1 - Input to MAP22	17
2 - Hull Geometry Matrix	22
3 - File ANS	22
4 - Printer File for MAP22	23
5 - File TBLINP	30
6 - Terminal Display for TBL18	31
7 - Printer File for TBL18	35
8 - File TBL3DINP	37
9 - Terminal Display for TBL3D	39
10 - File TBL3DOPT	40
11 - Terminal Display for CBL	43
12 - File CBL3DOPT	44
13 - File DOUBDOPT	46
14 - Terminal Display for TBLIPC	47
15 - File TBLIPCOPT	48
16 - Terminal Display for TBL SOL	50
17 - Printer File for TBL SOL	51

ABSTRACT

This report provides the instructions for using a suite of computer codes to calculate approximately the three-dimensional boundary layer on ship hulls. There are two sets of codes for computing boundary layers. One is for computing a first-order approximation and is based on a slenderbody potential and the small crossflow assumption. The second set of codes uses a general three-dimensional calculation method for the potential flow and solves the three-dimensional momentum integral boundary-layer equations without making restrictive assumptions on the magnitude of the crossflow. An example problem is presented for both methods.

ADMINISTRATIVE INFORMATION

The work supported herein was supported by the General Hydromechanics Research Program under Task Area SR 023-01-01 and Work Unit 1552-070.

I. INTRODUCTION

This manual provides the instructions for using a suite of computer codes to calculate approximately the three-dimensional turbulent boundary layer on ship hulls. The development of the methods programmed into this suite of boundary-layer codes, from here on designated BLC for the entire system of programs, is given in five references (1-5).

The package BLC has two sets of codes. The first set of codes is for computing a first-order approximation to the three-dimensional boundary layer on fairly fine ships such as a Series 60, Block 60 hull. These codes use slender-body-theory potential flow¹, at zero Froude number, to obtain the input pressure distribution. Secondly, the three-dimensional momentum integral equations with the assumption of small crossflow in the boundary layer are solved in streamline coordinates². This is another reason restricting these codes to slender ships. This set of codes is very easy to use and inexpensive to run on a computer so that for a first estimate of the ship boundary layer, it may be useful to employ them. Run time is less than two minutes for this code, whereas, twenty to thirty minutes is required by the second set of codes.

The second set of codes in BLC is more general in that the slender-body-theory

potential flow calculation method is replaced by a general three-dimensional calculation method. Furthermore, the three-dimensional momentum integral boundary-layer equations are solved in a general orthogonal surface coordinate system without making restrictive assumptions on the magnitude of the crossflow.

We shall briefly describe below the three main segments of BLC which make up the second, more general, boundary-layer calculation method. The geometry program represents the ship hull by a surface equation and is common to both the first and second set of programs in BLC.

The code for representing the ship hull by a surface equation from which all the necessary geometrical data are computed is based on the theory published by von Kerczek and Tuck³. The potential flow data is computed by a code based on the surface doublet distribution that was developed by Chang and Pien⁴, and the three-dimensional turbulent boundary layer is computed by the momentum integral method described by von Kerczek and Langan⁵. We refer the reader to references (3) - (5) for further details. We emphasize at the outset that the BLC provides only approximate results that are valid for (i) ships moving straight ahead at zero Froude number, i.e., the free surface is assumed to be plane, no waves are taken into account; (ii) the B.L. flow is unseparated and in fact very thin compared with the local radius of curvature in any direction on the body, so that the stern region is not properly modeled; and (iii) the boundary layer is fully turbulent, i.e., the laminar and transitional parts of the boundary layer near the bow are not modeled.

The underwater part of the ship hull is represented by a surface equation that consists of the conformal mapping of a unit circle onto the ship section and the polynomial interpolation of these sectionwise mappings along the length of the ship (see reference (1)). This type of surface representation can be used for most conventional ship hull shapes but it excludes two important geometrical features often found on ships. The first exclusion is a forward protruding bulb. The second exclusion is the rear portion of cruiser-type sterns. These exclusions of geometric forms are acceptable since the boundary-layer calculation method is not applicable in these regions of the ship.

The potential flow code⁴ is general and will handle any nonlifting fully-submerged body with a vertical plane of symmetry. The restriction to submerged bodies means, for the present situation, a double ship hull in which free surface effects are neglected. The code is based on the calculation of the doublet strength density

on the surface of the ship that will result in potential flow tangency on the surface. The method consists of distributing constant strength doublets on an assemblage of quadrilateral surface elements that approximate the actual surface of the ship and then solving the resulting algebraic equations that approximate the integral equation.

The numerical solution of the boundary-layer equations is carried out by a three-dimensional momentum integral-entrainment method⁵. In this method, the boundary-layer crossflow can be large but the accuracy of the computer results is limited by the very rough approximations that are made for the crossflow profile shape, the rate of entrainment coefficient and the relationships assumed for the crossflow parameters as functions of the streamwise flow parameters.

It is possible to give only rough estimates of the level of accuracy of the computed results of this program because the accuracy varies not only from one hull to another, but also on various parts of one hull. However, we have shown that on one model hull, the SSPA-720 Model, block coefficient 0.675, for which detailed experimental boundary layer data⁶ were available, the computed values of streamwise momentum and displacement thicknesses and surface shear stresses that were computed by the second system of codes in BLC were mostly within 10 percent of the experimental data⁵. Larger discrepancies between the computed and experimental results occurred only in localized regions of the hull. For example, very close to the stern (within the last 5 percent of the ship length), the experimentally determined boundary layer becomes very thick and the surface shear stress approaches the value of zero; the computed values of shear stress tend to overpredict the experimental values by as much as 100 percent in this region. Furthermore, there is a qualitative discrepancy between the computed and experimental values of displacement and momentum thicknesses and surface shear stresses along a streamline that lies very close to the keel of the hull. However, in general, the computed results from the second set of boundary-layer codes in BLC will be qualitatively correct and quantitatively will give values of about 90 percent of the experimental values for the streamwise displacement and momentum thicknesses and surface shear stresses on about 80 percent of the hull length, exclusive of the ends.

The crossflow angle of β that measures the difference in direction of the surface shear stress vector and the local direction of the inviscid surface streamline at the surface is not well-predicted by the methods of BLC. Over most of the hull

surface, where β is fairly small, the predictions of β from the BLC are qualitatively correct. However, near the stern of the model tested by Larsson⁶, where the experimental values of β become a little more substantial, on the order of 10-20 deg, the predicted values of BLC are substantially different, sometimes even having the wrong sign.

The numerical methods that are used in BLC are well-established and have been tested for accuracy as a function of step or panel sizes. The recommended step and panel sizes given in Chapter 3 in conjunction with the descriptions of the input schemes have been ascertained to yield at least a 2-3 percent level of numerical accuracy. Further reductions of step and panel sizes do reduce this error accordingly.

We need to mention here that empirical boundary-layer data are needed to start the calculations at some station downstream of the bow stern. The effects of inaccuracies in this starting data on the subsequent boundary-layer development are not known, but it seems that, if the starting station is located in a favorable pressure gradient region, then the downstream development of the boundary layer is not too sensitive to the starting conditions. Some experimentation will be necessary to ascertain this for individual cases.

The rest of this manual is organized as follows. Chapter 2 gives a complete description of each element of BLC and its relationship to the rest of the programs in BLC. In Chapter 3 we present a detailed discussion of the operation of the programs in conjunction with an example. The example is the SSPA Model 720, for which extensive boundary layer data exist⁶. A listing of the program is given in the appendix.

2. PROGRAMS

2.1 MAP22

The program MAP22 fits an analytic function to a set of hull offsets which are specified at K stations. It fits each of the K cross sections with an N-parameter map of the unit circle given by the equation²:

$$Z = \sum_{n=1}^N a_n \zeta^{3-2n} \quad (1)$$

Figure 1 shows the relationship between the complex Z and ζ -planes and also shows a typical map; Figure 2 shows the hull coordinate system. In the program the mapping is split into its real and imaginary parts; so at the I-th station it takes the form

$$\begin{aligned} \bar{X} &= \sum_{J=1}^N AN(I,J) \cos(3-2J)\theta \\ Y &= \sum_{J=1}^N AN(I,J) \sin(3-2J)\theta \end{aligned} \quad (2)$$

Once the AN's are computed, the program fits a polynomial in S to the parameters for J=1, 2, ..., N. It computes the coefficients BN(L,J) of these polynomials by solving the linear system

$$AN(I,J) = \sum_{L=1}^{KPK} BN(L,J) * S(I) ** (L-1) \quad (3)$$

Each polynomial has KPK coefficients

The hull is represented by the system of equations

$$\begin{aligned} X(S,\theta) &= \sum_{I=1}^{KPK} \sum_{J=1}^N BN(I,J) * S^{(I-1)} \cos(3-2J)\theta \\ Y(S,\theta) &= \sum_{I=1}^{KPK} \sum_{J=1}^N BN(I,J) * S^{I-1} \sin(3-2J)\theta \end{aligned} \quad (4)$$

This representation is used throughout the suite of programs.

The number of offsets given at each of the K stations need not be the same; for the moment let P offsets be given at station I. There are then P points of the unit circle in the fourth quadrant that will be mapped onto the P points of the I contour corresponding to these offsets. Since these points can be arbitrarily chosen, there are P values of θ ; namely θ_p ($p=1, \dots, P$), to be determined in addition to the N parameters AN(I,J). There are then 2P equations to be solved for

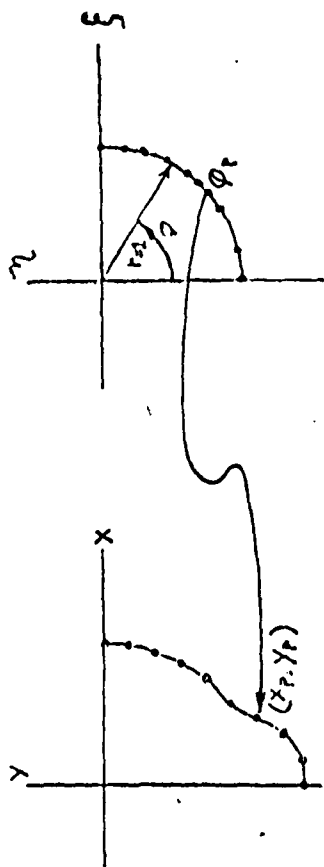


Figure 1 - Mapping of Unit Circle onto a Section

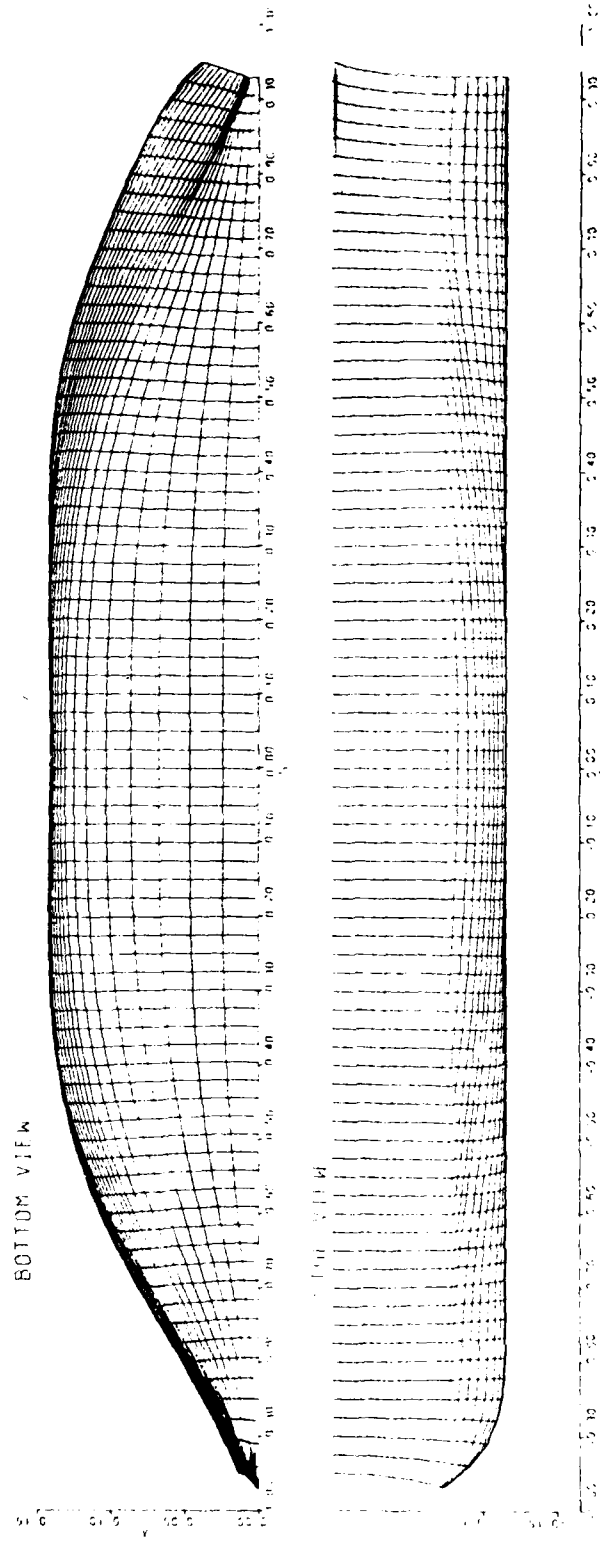


Figure 2 - The Bottom and Side Views of the (ϕ, θ) Coordinate Net on the SSPA 720 Model

the N values AN(I,J) (J=1, ..., N) and the P values of θ_p ; namely,

$$\begin{aligned} XMS(I,p) &= \sum_{J=1}^N AN(I,J) \cos(3-2J)\theta_p \\ YMS(I,p) &= \sum_{J=1}^N AN(I,J) \sin(3-2J)\theta_p \end{aligned} \quad (5)$$

where XMS and YMS are respectively the (x,y) offset of the P-th point. If N=P, unacceptable wiggles occur in the approximating cross section between the specified points; instead, N is taken to be of the order of 1/2 to 2/3 P and Equation (5) is solved in a least squares sense. Specifically, the AN's and θ_p are chosen to minimize the error

$$E = \sum_{p=1}^P (E_p'{}^2 + E_p''{}^2) \quad (6)$$

where

$$\begin{aligned} E_p' &= XMS(I,p) - \sum_{J=1}^N AN(I,J) \cos(3-2J)\theta_p \\ E_p'' &= YMS(I,p) - \sum_{J=1}^N AN(I,J) \sin(3-2J)\theta_p \end{aligned}$$

Not only does this approach result in a smooth form; it also fits the data points in a manner that minimizes the squared distance between the data points and the generated form. The relations

$$\begin{aligned} 0 &= \frac{\partial E}{\partial AN(I,J)} = L(AN; \theta_p) \\ 0 &= \frac{\partial E}{\partial \theta_p} = N(\theta_p; AN) \end{aligned} \quad (7)$$

are necessary conditions for E to have a minimum, where L and N are given by von Kerczek and Tuck². The system L=0 is linear in AN and can be solved provided θ_p is given. The system N=0 is totally nonlinear in both variables; however, it

can be used as a system of equations for θ_p provided the AN are assumed.

It is not practical to compute the best N-term fit directly, for this approach allows too much freedom in the AN's and results in a relatively bad fit. Instead, the best 4-term, 5-term, ..., N-term fit is determined successively starting with the 3-term Lewis form. Moreover, the values of AN (I,1) and AN (I,2) are further restricted to conform more closely with the Lewis forms. The best (J-1) term map serves as the initial guess to the j-th term map, which is solved iteratively. The first step of this iteration is to substitute the present values of the θ_p together with AN(I,1) and AN(I,2) into the system L=0. This system is solved by linear methods for the j-2 terms AN(I,3), ..., AN(I,j). Next the program computes AN(I,1) and AN(I,2) using the equations

$$A(I,1) = 1/2 \{ B(I) + H(I) - \sum_{J=3}^j [1 - (-1)^J] AN (I,J) \}$$

$$A(I,2) = 1/2 \{ B(I) - H(I) - \sum_{J=3}^j [1 + (-1)^J] AN (I,J) \}$$

where B(I) is the beam at station I and H(I) is the draft. System N=0 is then solved iteratively for the θ_p . If the resultant value for E has not changed from the previous value by more than a given tolerance TOL, the resulting AN's and θ_p are taken as the value for the best j-term fit; otherwise, the first step of the iteration is entered again.

MAP22 uses a least squares method to solve Equation (3) for the KPK coefficients BN(L,J), ..., BN(KPK,J). It has the facility to constrain the polynomial at any of the K stations at which the offsets are specified. The BN can be determined such that at station I the approximating polynomial has the same value as AN(I,J), zero slope, both zero slope and curvature, or a combination. An additional feature of the program is the provision made to partition the hull surface into a forebody, a parallel middlebody, and an afterbody. The forebody and afterbody are each constrained at their junction with the middlebody to have zero slope and curvature in the longitudinal direction and to have the same shape as the middlebody cross section.

2.2 TBL18

This program is based on the method of von Kerczek ² for computing a first-order approximation to the three-dimensional boundary layer on a fairly fine ship hull. The program uses the analytic surface representation, Equation (4), for the hull geometry and applies the slender-body potential derived by Tuck and von Kerczek to calculate the inviscid flow. It applies the Cumpsty-Head-Smith ^{7,8} three-dimensional turbulent boundary-layer calculation method to compute the entrainment, streamwise momentum, and skin friction along given streamlines. It is assumed that the crossflow is small in these calculations.

The Tuck and von Kerczek formula for the complex potential $f(s, \zeta)$ is given by

$$f(s, \zeta) = CN(1) \log AN(1) \zeta + BO - \sum_{I=1}^{N-1} \frac{CN(I+1)}{2I} \quad (8)$$

This is the potential for the inviscid flow around the analytic surface

$$AN(J) = \sum_{L=1}^{KPK} BN(L, J) S^{L-1} \quad (9)$$

which agrees with (4) for $S = S(I)$. The functions CN are given by

$$CN(1) = \sum_{J=1}^N (3-2J) AN(J) * ANP(J) \quad (10)$$

and

$$CN(J) = \sum_{L=1}^N (3-2L) AN(L) [ANP(L+J) + ANP(L-J)] \quad (11)$$

where ANP is the derivative of AN with respect to S (ANP(J) \equiv 0 for $J < 1$ or $J > N$). Finally the function BO of S is given as

$$BO = -\frac{1}{2} CN(1) * \text{LOG} [4(1-S^2)] + \frac{1}{2} \int_{-1}^1 \frac{C_o(s) - C_o(\xi)}{|S - \xi|} \quad (12)$$

where $C_o(S) = CN(1)$ evaluated at S. The integral is evaluated as a sum of terms containing the integrals

$$\int_{-1}^1 \frac{S^j - \xi^j}{|S - \xi|} d\xi = 2 \left(\sum_{k=1}^j \frac{1}{k} \right) S^j - \sum_{k=1}^j \frac{1 + (-1)^k}{k} S^{j-k} \quad (13)$$

for $J = 1, \dots, N$.

The integration of the boundary-layer equations begin at station $S(1)$, where empirical boundary-layer data are given as initial data for the differential equations. These boundary layer equations are

$$\frac{d\theta_{11}}{d\alpha} + \theta_{11}(2 + H) \frac{1}{U_s} \frac{dU_s}{d\alpha} - \theta_{11} K_1 = C_{f_1}$$

$$\frac{d\theta_{21}}{d\alpha} + 2\theta_{21} \left(\frac{1}{U_s} \frac{dU_s}{d\alpha} - K_1 \right) + \theta_{11} K_2 (1 + H) = C_{f_2} \quad (14)$$

$$\frac{d(\theta_{11}G)}{d\alpha} + \theta_{11}G \left(\frac{1}{U_s} \frac{dU_s}{d\alpha} - K_1 \right) = F(G)$$

where G is given in terms of the boundary-layer shape parameter H by

$$G = 1.535 (H - 0.7)^{-2.715} + 3.3 \quad (15)$$

and

$$F(G) = 0.0306 (G - 3.0)^{-0.1653} \quad (16)$$

$d\alpha$ is the differential element of arc along a streamline; K_1 and K_2 are geodesic curvatures; and U_s is the inviscid fluid velocity at the wall.

$$\theta_{11} = \int_0^\delta \left(1 - \frac{u}{U_0} \right) \frac{u}{U_0} d\lambda \quad (17)$$

$$\theta_{21} = -\int_0^\delta \frac{uv}{U_0^2} d\lambda \quad (18)$$

where u is the streamwise velocity in the boundary layer, and v is the cross-flow velocity. The friction coefficients C_{f_1} and C_{f_2} are in the streamwise and cross-flow directions respectively.

An equal step size Euler method is used in the numerical solution of Equation (14).

2.3 TBL3D

This program is a modification of the potential portion of TBL18 and is based on the same hull representation, Equation (4). Its primary purpose is the computation of the surface coordinates, distance metrics, and geodesic curvatures used in the three-dimensional momentum integral equations.

The intended application of BLC calls for the use of the inviscid velocities obtained from using a doublet distribution over the hull. Additionally, BLC has been provided with a quick way to estimate the solution to the three-dimensional momentum integral equation. This solution uses the Tuck and von Kerczek slender body potential to compute the inviscid velocities at the grid points. The potential portion of these computations is carried out in TBL3D.

If $\underline{r}(S, \theta)$ is the vector from the origin of the (X, Y, S) coordinate system,

$$\underline{r}(S, \theta) = X(S, \theta)\underline{i} + Y(S, \theta)\underline{j} + S\underline{k} \quad (19)$$

(\underline{i} , \underline{j} , \underline{k}) are the unit tangent vectors to the (X, Y, S) coordinate systems respectively. The surface coordinate lines $\theta = \text{constant}$, run along the length of the hull surface; the orthogonal trajectories to these coordinate lines form the coordinate lines $\phi = \text{constant}$. These latter coordinate lines are nearly parallel to the hull cross-sections. The arc length increments along the ϕ and θ coordinates are given by von Kerczek and Langan⁵ as

$$d\phi = (d\underline{r} \cdot d\underline{r})^{\frac{1}{2}}_{\theta = \text{constant}} = H(S, \theta) dS \quad (20)$$

and

$$d\theta = (d\underline{r} \cdot d\underline{r})^{\frac{1}{2}}_{\phi = \text{constant}} = G(S, \theta) d\theta \quad (21)$$

respectively, where

$$H(S, \theta) = \left[\left(\frac{\partial X}{\partial S} \right)^2 + \left(\frac{\partial Y}{\partial S} \right)^2 + 1 \right]^{\frac{1}{2}} \quad (22)$$

$$G(S, \theta) = \left[\left(\frac{\partial X}{\partial \theta} + F \right)^2 + \left(\frac{\partial Y}{\partial \theta} + F \right)^2 + F^2 \right]^{\frac{1}{2}} \quad (23)$$

and

$$F = \frac{dS}{d\theta} \Big|_{\phi} = \frac{-\left(\frac{\partial X}{\partial \theta} \frac{\partial X}{\partial S} + \frac{\partial Y}{\partial \theta} \frac{\partial Y}{\partial S} \right)}{\left[1 + \left(\frac{\partial X}{\partial \theta} \right)^2 + \left(\frac{\partial Y}{\partial \theta} \right)^2 \right]} \quad (24)$$

The program uses the unit orthogonal vectors \underline{e}_ϕ , \underline{e}_n , \underline{e}_θ , to simplify the computation of the geodesic curvatures K_ϕ and K_θ . Since the ϕ -coordinates are generated by holding θ constant in Equation 19 for \underline{r} , the unit vector in the ϕ -direction is given by

$$\underline{e}_\phi = (EP1, EP2, EP3) = H^{-1} \left(\frac{\partial X}{\partial S}, \frac{\partial Y}{\partial S}, 1 \right) \quad (25)$$

The unit vector normal to the surface is determined from the cross product $\frac{\partial \underline{r}}{\partial \theta} \times \frac{\partial \underline{r}}{\partial S}$

which is normal to the surface of the hull; hence,

$$\begin{aligned} \underline{e}_n &= (EN1, EN2, EN3) \\ &= D^{-1} \left(\frac{\partial Y}{\partial \theta}, -\frac{\partial X}{\partial \theta}, \frac{\partial X}{\partial \theta} \frac{\partial Y}{\partial S} - \frac{\partial X}{\partial S} \frac{\partial Y}{\partial \theta} \right) \end{aligned} \quad (26)$$

where

$$D = \left[\left(\frac{\partial Y}{\partial \theta} \right)^2 + \left(\frac{\partial X}{\partial \theta} \right)^2 + \left(\frac{\partial X}{\partial \theta} \frac{\partial Y}{\partial S} - \frac{\partial X}{\partial S} \frac{\partial Y}{\partial \theta} \right)^2 \right]^{1/2} \quad (27)$$

From the definition of orthonormal

$$\underline{e}_\theta = \underline{e}_n \times \underline{e}_\phi; \quad |\underline{e}_\theta| = 1$$

that is

$$\begin{aligned} \underline{e}_\theta &= (ET1, ET2, ET3) \\ &= (EN2 * EP3 - EN3 * EP2, EN3 * EP1 \\ &\quad - EN1 * EP3, EN1 * EP2 - EN2 * EP1) \end{aligned} \quad (28)$$

The geodesic curvatures are calculated using

$$K_\phi = \underline{e}_\phi \cdot \frac{d\underline{e}_\theta}{d\ell_\theta} = -\frac{1}{h_\theta h_\phi} \frac{\partial h_\theta}{\partial \phi} \quad (29)$$

and

$$K_\theta = \underline{e}_\theta \cdot \frac{d\underline{e}_\phi}{d\ell_\phi} = -\frac{1}{h_\phi h_\theta} \frac{\partial h_\phi}{\partial h_\theta} \quad (30)$$

2.4 CBL

The program CBL lays out the panels for the doublet distribution on the analytic hull. It merely repeatedly applies Equation (9) to obtain the coordinates for the panel corners and formats this data for the potential flow program DOUBLD.

If one is simply computing the potential flow around a double hull model, one needs only to use MAP22, CBL, and DOUBLD.

2.5 DOUBDD

The program DOUBDD computes the potential and velocities for inviscid flow around a double hull model. Pien and Chang⁴ developed the original form of this program; with their permission we have documented their program here to publicize it and to have a complete discussion of the program system BLC. The program solves the Neumann problem for flow around a body by distributing a layer of doublets on the surface of the body, in our case the double hull, and solving numerically the resulting integral equation by a panel method. The magnitude of the doublet distribution on the panels, SIG(I), is obtained directly as the solution of the linear system

$$\sum_{J=1}^{NP} VPP(I,J) * SIG(J) = -4\pi \bar{S}(I) \quad (31)$$

for $I=1, \dots, NP$, which approximates the integral equation

$$4\pi S(P) = \iint_{\substack{\text{Double} \\ \text{Hull}}} \sigma(Q) G(P,Q) ds_Q$$

SIG(J) is the strength of the doublets distributed on the J-th panel, VPP(I,J) is the influence of this doublet distribution on the I-th panel, and $\bar{S}(I)$ is an average station location for the I-th panel. In the integral equation $G(P,Q)$ is the three-dimensional distribution $1/r$, where r is the distance between the points P and Q on the hull surface, and ds_Q is the increment of surface area used in the integration.

The doublet distribution is the perturbation potential due to the inviscid flow around the double hull.

In the actual mechanics of the solution DOUBDD replaces the curva-linear panels, that are constructed in CBL, by plane panels. These planes are chosen to minimize

the distance between the corners of the plane panels and those of the curvilinear ones. If the longitudinal length of a plane panel exceeds its traverse width by a multiple of two or more the panel length is divided into subpanels which satisfy the criterion that their length is less than twice their width.

The off-diagonal elements of the influence matrix VPP are computed in a straight forward manner. This matrix can be ill conditioned, and a straight forward computation of the diagonal elements V(I, I) can lead to serious error. A different approach is taken in the computation of the diagonal elements to avoid the problem. Since the equation

$$\sum_{J=1}^{NP} VPP(I,J) = -4\pi \quad (32)$$

corresponds to Greens identity

$$\iint_{SS} \frac{\partial}{\partial n} G(P,\theta) dS_Q = -4\pi$$

one can use

$$VPP(I,I) = -4\pi - \sum_{\substack{J=1 \\ J \neq I}}^{NP} VPP(I,J) \quad (33)$$

to compute the diagonal elements. This calculation method does not produce the errors in the solution that results from computing the diagonal elements in a straight forward manner. It is therefore used in DOUBDD.

2.6 TBLIPC

This program organizes the geometric and inviscid flow data for use in the turbulent boundary-layer calculations. These calculations are made on a surface grid (ϕ_i, θ_j) given by

$$\begin{aligned} \phi_i &= \phi_{i-1} + (i-1)\Delta\phi & i &= 2, \dots, NS \\ \theta_j &= (j-1)\Delta\theta & j &= 1, \dots, NTH \end{aligned}$$

The initial value ϕ_1 is the value of ϕ at the station at which the boundary-layer calculations are initiated. Empirical values of the momentum integral θ_{11} , the crossflow angle β , and of the shape parameter H are given at $\phi = \phi_1$ as initial conditions; the calculations proceed station by station downstream from ϕ_1 . The bound-

ary-layer equations are first solved for θ_{11} , $t = \tan \beta$ and H at $\phi = \phi_2$; these calculations are then repeated at $\phi = \phi_3, \phi_4, \dots, \phi_{NS}$. Because the total amount of geometric and inviscid flow data is prohibitively large for storage in main memory and since it is only used station by station, TBLIPC organizes it as a function of θ for each value ϕ_i and stores it on disk for $i = 1, \dots, NS$.

Typically, DOUBDD computes the inviscid flow potential at 250 to 500 points approximately evenly distributed over the portion of the hull having $0 \leq \theta \leq \frac{\pi}{2}$; the boundary layer calculations require the inviscid velocities and their derivatives at 1500 or more points on the grid (ϕ_i, θ_j) . TBLIPC uses spline on spline interpolation to determine the potential at the grid points. It uses the derivatives of these spline functions to compute the inviscid velocity tangent to the surface in the ϕ -direction, u_ϕ , and the corresponding velocity in the θ -direction, u_θ .

2.7 TBSOL

von Kerczek and Langan⁵ expressed the momentum integral equations and the entrainment equation in the form

$$A(W(\phi, \theta)) \frac{\partial W}{\partial \phi} + B(W(\phi, \theta)) \frac{\partial W}{\partial \theta} = C(W)$$

where

$$W = \begin{pmatrix} \theta \\ t \\ H \end{pmatrix}$$

The program TBSOL uses an implicit marching method to solve this system of nonlinear equations. In marching from station ϕ_i to ϕ_{i+1} ..., it uses the value of W at (ϕ_i, θ_i) to compute the coefficients A , B , and C at (ϕ_{i+1}, θ_i) . The program uses Gaussian reduction to solve the resulting system of linear equations.

The coefficient matrices are described in von Kerczek and Langan⁵ together with the theoretical details of the solution method.

3. OPERATING INSTRUCTIONS WITH AN EXAMPLE

The suite of programs BLC has been installed on the Burroughs 7700 at DTNSRDC as an interactive software system. Each individual program was initially designed for batch operation on the CDC 6700 computer system. A version of DOUBDD is pre-

sently available on the Naval Research Laboratory Advanced Scientific Computer; it uses the large array storage and pipeline capabilities of that machine. The individual programs required minimum modification when moved from one machine to the other; they are written in a fairly basic FORTRAN. The operating instructions and program listings presented in this report apply to the interactive software system as installed on the Burroughs 7700.

The Swedish Model SSPA 720 is used in the sample calculations. The offsets used for this ship are given as input to MAP22. The input to each program is given in detail, unless it is generated as output from one of the programs in the system. Since the output from the programs would require too much space and can be easily generated if needed, only partial listings of the output will be given.

3.1 MAP22

This is the first program of the system. Its primary output is the hull geometry representation matrix BN. This matrix forms the basis of all further calculations in the system. The input data is stored in file PFN before running the program. The input data file PFN for the SSPA Model 720 is shown in Table 1. It is read by MAP22 with the following read statements.

```
READ(5,24) (TITLE(J),J=1,12)
READ(5,3) IST1,IST2,KP1,KP2
READ(5,1) K,N,TOL,BL
READ(5,3) (IFS(I),I=1,K)
READ(5,3) (JFS(I),I=1,K)
READ(5,25) (KT(I),I=1,K)
READ(5,4) (IX(J),IY(J),J=1,M)
READ(5,2) (S(I),I=1,K)
32 READ(5,2) (AN(I,J),J=1,N)
```

The formats are

```
1 FORMAT(2I10,2F10.5)
4 FORMAT(6(1X,2I6))
2 FORMAT (7F10.5)
3 FORMAT(14I5)
25 FORMAT(7I10)
24 FORMAT(12A6)
```

TABLE 1 - INPUT TO MAP22

#FILE (CHXL)PFN ON DTNSRDC																
100																
200	25	0	11	0												
300		25		7	0.0001	1.0										
400	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
500	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
600	1															
700																
800		4		15		17		19		20		23		26		
900		29		33		20		21		21		21		21		
1000		22		21		19		18		17		16		18		
1100		20		21		23		9								
1200	*		*		- 500*		- 1000*		- 1636							
1300		*800439-00005*800407-00272*800381-00511*800349-00766*800331-01002*800309-01248														
1400		*800287-01503*800264-01753*800242-01992*800217-02236*800183-02494*800142-02743														
1500		*800097-02981*800064-03088*-00001-03091														
1600		*800859-00002*800823-00271*800791-00512*800760-00767*800736-01007*800706-01250														
1700		*800686-01508*800659-01750*800628-01993*800585-02235*800531-02496*800475-02735														
1800		*800405-02984*800320-03229*800223-03476*800119-03713*-00008-03718														
1900		*801338-00003*801297-00272*801263-00514*801226-00769*801191-01009*801168-01253														
2000		*801132-01506*801094-01754*801051-01996*801000-02241*800932-02493*800850-02736														
2100		*800747-02987*800635-03237*800501-03478*800348-03718*800252-03853*800137-03984														
2200		*-00007-03985														
2300		*801837-00004*801803-00274*801773-00513*801732-00771*801699-01005*801660-01253														
2400		*801616-01508*801565-01756*801506-01996*801441-02238*801358-02497*801270-02735														
2500		*801140-02988*800990-03233*800806-03480*800583-03720*800427-03873*800311-03972														
2600		*800188-04079*-00001-04095														
2700		*802850-00007*802820-00277*802788-00513*802754-00774*802724-01007*802686-01257														
2800		*802643-01520*802593-01757*802539-02000*802468-02238*802370-02504*802250-02740														
2900		*802083-02995*801872-03231*801719-03374*801581-03478*801418-03588*801242-03693														
3000		*800983-03822*800738-03924*800506-04006*800252-04089*-00004-04090														
3100		*803738-00003*803715-00276*803688-00515*803668-00783*803647-00808*803622-01257														
3200		*803595-01514*803556-01760*803508-02001*803438-02240*803335-02504*803201-02741														
3300		*803023-03002*802793-03244*802638-03370*802463-03487*802219-03616*801967-03729														
3400		*801718-03825*801478-03897*801243-03952*800988-04012*800741-04049*800502-04080														
3500		*800251-04091*-00004-04089														
3600		*804407-00002*804400-00281*804391-00529*804383-00785*804374-01015*804365-01265														
3700		*804350-01525*804335-01770*804302-02002*804243-02249*804162-02505*804048-02749														
3800		*803886-03005*803699-03218*803459-03421*803200-03581*802955-03698*802717-03784														
3900		*802463-03863*802221-03928*801961-03976*801718-04015*801480-04046*801243-04072														
4000		*800987-04091*800739-04097*800503-04095*800251-04096*800001-04093														
4100		*804800-00000*804797-00282*804796-00521*804794-00777*804798-01009*804802-01260														
4200		*804798-01522*804793-01769*804776-02001*804744-02248*804690-02506*804613-02751														
4300		*804495-03006*804337-03247*804201-03393*804084-03490*803946-03598*803695-03728														
4400		*803460-03821*803199-03886*802953-03943*802651-03982*802455-04010*802215-04032														
4500		*801963-04051*801719-04066*801475-04082*801244-04093*800988-04093*800745-04095														

TABLE 1 - (continued)

4600 *800501-04093*800257-04096*-00007-04098
4700 *804954-00008*804955-00285*804959-00777*804958-01267*804960-01768*804927-02249
4800 *804867-02751*804757-03139*804578-03462*804310-03705*803944-03888*803568-03986
4900 *803204-04031*802714-04067*802219-04085*801715-04092*801239-04090*800741-04092
5000 *800253-04094*-00008-04089
5100 *804954-00004*804956-00286*804956-00781*804959-01264*804956-01770*804959-02249
5200 *804954-02752*804902-03250*804683-03665*804434-03872*804193-03993*803939-04062
5300 *803460-04072*802954-04078*802461-04083*801964-04090*801479-04087*800990-04089
5400 *800499-04090*800250-04093*-00008-04091
5500 *804957-00004*804953-00286*804961-00776*804956-01258*804962-01768*804957-02249
5600 *804954-02754*804935-03249*804752-03646*804459-03917*804158-04041*803702-04071
5700 *803201-04073*802714-04077*802220-04084*801714-04090*801242-04088*800741-04092
5800 *800503-04092*800250-04095*-00003-04091
5900 *804957-00004*804953-00286*804961-00776*804956-01258*804962-01768*804957-02249
6000 *804954-02754*804935-03249*804752-03646*804459-03917*804158-04041*803702-04071
6100 *803201-04073*802714-04077*802220-04084*801714-04090*801242-04088*800741-04092
6200 *800503-04092*800250-04095*-00003-04091
6300 *804957-00004*804953-00286*804961-00776*804956-01258*804962-01768*804957-02249
6400 *804954-02754*804935-03249*804752-03646*804459-03917*804158-04041*803702-04071
6500 *803201-04073*802714-04077*802220-04084*801714-04090*801242-04088*800741-04092
6600 *800503-04092*800250-04095*-00003-04091
6700 *-04962800003*-04961-00245*-04974-00735*-04967-01237*-04963-01719*-04964-02209
6800 *-04965-02717*-04932-03107*-04820-03440*-04658-03682*-04471-03849*-04238-03982
6900 *-03925-04059*-03526-04064*-03003-04065*-02495-04072*-01988-04081*-01500-04089
7000 *-00976-04085*-00491-04091*-00243-04092*-00009-04091
7100 *-04963-00000*-04964-00240*-04970-00734*-04963-01225*-04960-01720*-04933-02190
7200 *-04868-02679*-04777-03074*-04619-03423*-04439-03650*-04205-03818*-03750-04003
7300 *-03363-04055*-02996-04070*-02507-04077*-01988-04082*-01499-04082*-00975-04089
7400 *-00498-04089*-00249-04087*-00008-04089
7500 *-04964800001*-04953-00241*-04916-00736*-04874-01232*-04818-01720*-04731-02215
7600 *-04591-02717*-04396-03115*-04152-03445*-03848-03688*-03486-03861*-02994-03991
7700 *-02504-04062*-01993-04079*-01495-04085*-00977-04084*-00498-04091*-00241-04094
7800 *-00006-04088
7900 *-04771-00000*-04744-00243*-04669-00739*-04573-01234*-04470-01654*-04333-02084
8000 *-04175-02473*-03974-02843*-03724-03189*-03441-03456*-03065-03700*-02678-03867
8100 *-02267-03984*-01745-04068*-01253-04087*-00748-04093*-00253-04093*-00002-04088
8200 *-04424-00004*-04345-00250*-04165-00743*-03975-01241*-03765-01729*-03527-02226
8300 *-03237-02717*-03014-03023*-02799-03263*-02503-03526*-02161-03741*-01742-03916
8400 *-01348-04020*-00941-04082*-00502-04090*-00209-04091*-00008-04092
8500 *-03849-00001*-03691-00250*-03479-00595*-03260-00939*-03040-01302*-02755-01764
8600 *-02518-02162*-02335-02490*-02140-02834*-01945-03171*-01712-03503*-01353-03813
8700 *-00958-03994*-00548-04088*-00254-04093*-00009-04091
8800 *-03087-00001*-02881-00212*-02640-00466*-02415-00698*-02143-00995*-01945-01247
8900 *-01747-01530*-01567-01845*-01402-02214*-01278-02592*-01168-02973*-01081-03344
9000 *-00981-03594*-00841-03814*-00660-03974*-00428-04083*-00211-04089*-00008-04092

TABLE 1 - (continued)

9100	*-02616-00001*-02474-00122*-02324-00255*-02107-00438*-01785-00742*-01540-00998							
9200	*-01317-01306*-01161-01577*-01059-01843*-00949-02190*-00884-02487*-00851-02741							
9300	*-00819-02988*-00788-03224*-00740-03476*-00677-03713*-00558-03949*-00362-04085							
9400	*-00145-04091*-00012-04095							
9500	*-02042-00006*-01935-00089*-01741-00248*-01497-00447*-01261-00668*-01071-00858							
9600	*-00863-01130*-00677-01446*-00591-01689*-00539-01952*-00511-02237*-00496-02490							
9700	*-00494-02737*-00496-02984*-00496-03233*-00494-03477*-00470-03711*-00401-03923							
9800	*-00258-04067*-00096-04092*-00007-04090							
9900	*-01484&00002*-01356-00088*-01254-00167*-01124-00258*-01001-00366*-00835-00500							
10000	*-00655-00670*-00529-00816*-00414-01000*-00311-01250*-00260-01502*-00230-01753							
10100	*-00214-01991*-00207-02238*-00206-02488*-00197-02739*-00199-02939*-00189-03228							
10200	*-00188-03476*-00183-03716*-00179-03961*-00151-04092*-00008-04095							
10300	*-00007&00000*-00004-00511*-00005-01004*-00004-01506*-00006-01989*-00006-02491							
10400	*-00005-02977*-00011-03480*-00010-04095							
10500	-1.0	-0.95	-0.9	-0.85	-0.8	-0.7	-0.6	
10600	-0.5	-0.4	-0.3	-0.2	-0.1	0.0	0.1	
10700	0.2	0.3	0.4	0.5	0.6	0.7	0.8	
10800	0.85	0.9	0.95	1.0				
10900	0.34641	-0.27440	0.00758	0.00463	0.00198	0.00211	0.00032	
11000	0.44878	-0.29447	0.01009	0.00223	0.00258	0.00369	0.00050	
11100	0.52025	-0.27080	0.01227	-0.00004	0.00391	0.00368	0.00081	
11200	0.58204	-0.22798	0.01064	-0.00164	0.00543	0.00172	0.00060	
11300	0.70588	-0.12377	-0.01246	-0.00275	0.00587	0.00137	0.00117	
11400	0.82395	-0.03395	-0.03654	-0.00304	0.00176	0.00157	0.00080	
11500	0.92477	0.03544	-0.06407	-0.00478	-0.00266	0.00103	-0.00015	
11600	0.99266	0.07721	-0.09170	-0.00561	-0.00276	-0.00075	-0.00013	
11700	1.03039	0.09126	-0.11683	-0.00364	-0.00079	-0.00033	-0.00007	
11800	1.04960	0.09273	-0.14022	-0.00500	0.00252	-0.00063	0.00099	
11900	1.05300	0.09338	-0.14479	-0.00569	0.00431	-0.00028	0.00067	
12000	1.05300	0.09338	-0.14479	-0.00569	0.00431	-0.00028	0.00067	
12100	1.05300	0.09338	-0.14479	-0.00569	0.00431	-0.00028	0.00067	
12200	1.04944	0.09401	-0.13950	-0.00489	0.00276	-0.00121	0.00100	
12300	1.03303	0.08996	-0.11985	-0.00025	0.00009	-0.00150	0.00033	
12400	0.99788	0.07939	-0.08535	0.00724	0.00040	0.00129	0.00069	
12500	0.93959	0.05219	-0.04667	0.01319	0.00088	0.00356	0.00033	
12600	0.85550	0.00524	-0.00262	0.02280	0.00562	0.00547	0.00101	
12700	0.75006	-0.06606	0.04397	0.03767	0.00606	0.00397	0.00128	
12800	0.63970	-0.15695	0.07969	0.05623	0.00548	-0.00072	-0.00031	
12900	0.58626	-0.21333	0.08316	0.06482	0.01088	-0.00076	-0.00297	
13000	0.53095	-0.27523	0.07389	0.06886	0.01946	-0.00034	-0.00540	
13100	0.47381	-0.33047	0.06755	0.05733	0.02485	0.00962	-0.00313	
13200	0.41167	-0.41260	0.00234					

On Table 1, line 100 is simply an alpha numeric title to identify the data in subsequent runs; this title is carried throughout the calculations.

Line 200: IST1 is the number of stations at which offsets are given on the forebody, if the ship is being cut into a fore and afterbody or a parallel middle body is being used. It is the total number of stations, if the ship is being represented by a single polynomial. IST2 is the number of stations on the afterbody; it is zero if there is no separate afterbody representation. KP1 is the degree minus one of the polynomial in S for the forebody or for the full length of the ship if IST2=0. KP2 is the degree minus one of the polynomial for the afterbody or zero if IST2=0.

Line 300: The first number is K, the total number of stations at which offsets are given. N is the number of parameters in the mapping of the unit circle onto the cross section; it should be chosen to be of the order of 1/2 to 2/3 the number of offsets at each of the stations. In the present case N=7 is between 1/3 to 1/2 the number of offsets at the different stations. TOL is the tolerance to which an interaction is solved. BL is the beam-to-length ratio.

Lines 400-500: IFS(I) determines the type of constraints to be used at station I.

- IFS(I) = 1. No constraints
- 2. Surface at station I passes through offsets
- 3. Slope with respect to S is zero
- 4. Combination of 2 and 3
- 5. Slope and curvature with respect to S is zero
- 6. Combination of 2, 3 and 5

Lines 600-700: JFS(I) is zero if AN is to be read at station I; it is 1 if AN is to be computed.

Lines 900-1100: KT(I) is the number of offsets at station I.

Lines 1300-10400: IX(J) is the x-coordinate of the j-th offset at station I in integer form, as would be obtained from a digitizer. IY(J) is the corresponding Y-coordinate.

Lines 10500-10800: S(I) is the value of S at station I.

Lines 10900-13200: AN(I,J) is the jth value of AN at station I grouped by stations. J=1,...,N.

MAP22 has three output files:

File BNS Table 2 contains the hull geometry matrix.

Line 100: N KPK BL

Lines 200-1500: BN(I,J) grouped by values of J=1, 2, ..., N.

I = 1, ..., KPK

File ANS Table 3, contains starting values of AN for additional calculations.

Lines 100-2600: AN(I,J) I=1, KPK, J=1, ..., N grouped by values of I.

File PRINTER is shown as File OPT in Table 4. It contains the data as follows:

Line 100: Ignore the 1 in this line

Line 200: Title

Line 300: "The Input Offsets," this is a heading for several pages of input effects. In the present example, the input effects ran from lines 900 through 62500; Table 4 shows output for stations $S(1) = -1.00$
 $S(2) = -0.95$ and $S(25) = 1.00$. XMS is the x-coordinate and YMS is the y-coordinate.

Lines 62900-63500: show the progression of the calculation for AN(I,J), which is typical of the iterative calculations of AN.

Line 63600: In this case, the iteration is not convergent from fourth to fifth iteration, so the results for the fourth iteration are used.

Line 63900: Is typical of lines 63900 through 70800. The AN's for stations 2 through 25 are given in this example and line 63900 gives the values of AN at station I = 2.

Line 71100: MP, KPK, and IT

Lines 71500-72400: Gives intermediate values of a matrix BN used in the computation.

Lines 72700-7400: Gives the intermediate matrix BS

Lines 74300-75900: "The Polynomial Coefficients," these are the final values of the matrix BN(I,J).

Lines 76200-85500: are typical of the computed offsets at the different stations.

Lines 201300-203900: are a printed version of the matrix A(I,J).

For the entire ship starting with just the offsets MAP22 requires about ten minutes of CPU time on the Burroughs B7700; however, the example runs in less than a minute. It is imperative that the input offsets be compared with the output off-

TABLE 2 - HULL GEOMETRY MATRIX

100	7	11	1.00000					
200	1.05407	0.00946	-0.06386	-0.17969	-2.17726	1.30080	1.75289	
300	-2.35902	0.99482	1.35102	-1.27218				
400	0.09199	-0.01678	0.12962	0.23842	-1.63679	-0.65025	1.62619	
500	1.30540	-1.17223	-0.99895	0.67261				
600	-0.14817	-0.01482	0.27346	0.19518	0.85676	-0.34804	-2.45327	
700	0.50034	2.46250	-0.32998	-0.98946				
800	-0.00560	-0.00706	0.00830	0.22146	0.35303	-0.73910	-1.28967	
900	1.22848	1.99487	-0.70358	-1.06058				
1000	0.00609	-0.01300	-0.15698	0.25859	1.06743	-1.09828	-2.79267	
1100	1.64747	3.20478	-0.79404	-1.32815				
1200	0.00030	-0.01009	-0.05990	0.14915	0.62418	-0.50594	-1.98688	
1300	0.62018	2.24264	-0.25234	-0.92001				
1400	0.00096	0.00142	-0.01278	-0.01756	0.05647	0.09382	-0.03564	
1500	-0.17823	-0.08303	0.10064	0.07818				

TABLE 3 - FILE ANS

100	0.16513	-0.16512	-0.00003	0.00001	0.00001	-0.00001	0.00000
200	0.16513	-0.16512	-0.00003	0.00001	0.00001	-0.00001	0.00000
300	0.34641	-0.27440	0.00758	0.00463	0.00198	0.00211	0.00032
400	0.44878	-0.29447	0.01009	0.00223	0.00258	0.00369	0.00050
500	0.52025	-0.27080	0.01227	-0.00004	0.00391	0.00368	0.00081
600	0.58204	-0.22798	0.01064	-0.00164	0.00543	0.00172	0.00060
700	0.70588	-0.12377	-0.01246	-0.00275	0.00587	0.00137	0.00117
800	0.82395	-0.03395	-0.03654	-0.00304	0.00176	0.00157	0.00080
900	0.92477	0.03544	-0.06407	-0.00478	-0.00266	0.00103	-0.00015
1000	0.99266	0.07721	-0.09170	-0.00561	-0.00276	-0.00075	-0.00013
1100	1.03039	0.09126	-0.11683	-0.00364	-0.00079	-0.00033	-0.00007
1200	1.04960	0.09273	-0.14022	-0.00500	0.00252	-0.00063	0.00099
1300	1.05300	0.09338	-0.14479	-0.00569	0.00431	-0.00028	0.00067
1400	1.05300	0.09338	-0.14479	-0.00569	0.00431	-0.00028	0.00067
1500	1.05300	0.09338	-0.14479	-0.00569	0.00431	-0.00028	0.00067
1600	1.04944	0.09401	-0.13950	-0.00489	0.00276	-0.00121	0.00100
1700	1.03303	0.08996	-0.11985	-0.00025	0.00009	-0.00150	0.00033
1800	0.99788	0.07989	-0.08535	0.00724	0.00040	0.00129	0.00069
1900	0.93959	0.05219	-0.04667	0.01319	0.00088	0.00356	0.00033
2000	0.85550	0.00524	-0.00262	0.02280	0.00562	0.00547	0.00101
2100	0.75006	-0.06606	0.04397	0.03767	0.00606	0.00397	0.00128
2200	0.63970	-0.15695	0.07969	0.05623	0.00548	-0.00072	-0.00031
2300	0.58626	-0.21333	0.08316	0.06482	0.01088	-0.00076	-0.00297
2400	0.53095	-0.27523	0.07389	0.06886	0.01946	-0.00034	-0.00540
2500	0.47381	-0.33047	0.06755	0.05733	0.02485	0.00962	-0.00313
2600	0.41167	-0.41260	0.00234	0.00000	0.00000	0.00000	0.00000

TABLE 4 - PRINTER FILE FOR MAP22

100 1			
200	SSPA MODEL 720		
300	THE INPUT OFFSETS		
400			
500			
600			
700			
800			
900	XMS	YMS	S(1)=-1.00
1000			
1100			
1200	0.00000	0.00000	
1300	0.00000	-0.10093	
1400	0.00000	-0.20186	
1500	0.00000	-0.33024	
1600			
1700			
1800	XMS	YMS	S(2)=-0.95
1900			
2000			
2100	0.08862	-0.00101	
2200	0.08216	-0.05491	
2300	0.07691	-0.10315	
2400	0.07045	-0.15462	
2500	0.06681	-0.20226	
2600	0.06237	-0.25192	
2700	0.05793	-0.30339	
2800	0.05329	-0.35386	
2900	0.04885	-0.40210	
3000	0.04380	-0.45135	
3100	0.03694	-0.50343	
3200	0.02866	-0.55369	
3300	0.01958	-0.60174	
3400	0.01292	-0.62333	
3500	-0.00020	-0.62394	

TABLE 4 - (continued)

	XMS	YMS	S(25)= 1.00
61300			
61400			
61500			
61600			
61700	0.00141	0.00000	
61800	0.00081	-0.10315	
61900	0.00101	-0.20266	
62000	0.00081	-0.30400	
62100	0.00121	-0.40149	
62200	0.00121	-0.50283	
62300	0.00101	-0.60093	
62400	0.00222	-0.70246	
62500	0.00202	-0.82660	
62600			
62700			
62800			
62900			
63000			
63100			
63200			
63300			
63400			
63500			
63600			
63700			
53800			
53900			
64000			
71100			
71200			
71300			
71400			
71500			
71600			
71700			
71800			
71900			
72000			

INITIAL AN(I, J)=	0.00000	-0.16512	0.00000	A-EA COEFFICIENT=	-1.00000
DUT=	0.00000	AN= 0.16512	-0.16512	0.00000	
DUT=	0.00000	AN= 0.16513	-0.16513	-0.00001	0.00001
DUT=	0.00000	AN= 0.16513	-0.16513	-0.00002	0.00001
DUT=	0.00000	AN= 0.16513	-0.16512	-0.00003	0.00001
FIT IS GOING BAD DUT=0.00000					
AN GIVEN AT I= 2 AN= 0.34641 -0.27440 0.00758 0.00463 0.00198 0.00211 0.00032					
MP=	11	KPK=	11	INT=	0
THE VECTORS BN					
19.01675	0.60387	6.08532	0.50995	3.72070	0.45222
-1.94706	-0.10453	-2.21187	-0.17937	-1.89257	-0.21743
-0.89903	0.30259	0.17484	0.20837	0.19406	0.15325
				0.16108	0.11771
				0.13006	0.09322
				2.17005	0.38382
				-1.43171	-0.24976
				0.41271	1.81186
				-0.23813	-1.28060
				2.72657	-0.24976
				-1.63013	0.09322
				0.16108	0.11771
				0.15325	0.13006
				0.16108	0.11771

TABLE 4 - (continued)

72100	0.28630	0.27041	0.22098	0.18581	0.16476	0.13736	0.12724	0.10580	0.10091	0.08374	0.08163
72200	0.10726	0.04867	0.06766	0.03778	0.05217	0.03124	0.04160	0.02643	0.03396	0.02262	0.02822
72300	0.03199	0.00357	0.02218	0.00168	0.01615	0.00151	0.01279	0.00174	0.01062	0.00198	0.00906
72400	-0.00032	-0.01164	-0.00578	-0.00984	-0.00566	-0.00807	-0.00503	-0.00661	-0.00434	-0.00545	-0.00370
72500											
72600											
72700											
72800											
72900											
73000	25.00000	0.00000	10.95000	-0.00000	7.73963	0.00000	6.18129	-0.00000	5.22645	0.00000	4.57409
73100	0.00000	10.95000	-0.00000	7.73963	0.00000	6.18129	-0.00000	5.22645	0.00000	4.57409	0.00000
73200	10.95000	-0.00000	7.73963	0.00000	6.18129	-0.00000	5.22645	0.00000	4.57409	0.00000	4.10006
73300	-0.00000	7.73963	0.00000	6.18129	-0.00000	5.22645	0.00000	4.57409	0.00000	4.10006	0.00000
73400	7.73963	0.00000	6.18129	-0.00000	5.22645	0.00000	4.57409	0.00000	4.10006	0.00000	3.74165
73500	0.00000	6.18129	-0.00000	5.22645	0.00000	4.57409	0.00000	4.10006	0.00000	3.74165	0.00000
73600	6.18129	-0.00000	5.22645	0.00000	4.57409	0.00000	4.10006	0.00000	3.74165	0.00000	3.46290
73700	-0.00000	5.22645	0.00000	4.57409	0.00000	4.10006	0.00000	3.74165	0.00000	3.46290	-0.00000
73800	5.22645	0.00000	4.57409	0.00000	4.10006	0.00000	3.74165	0.00000	3.46290	-0.00000	3.24141
73900	0.00000	4.57409	0.00000	4.10006	0.00000	3.74165	0.00000	3.46290	-0.00000	3.24141	-0.00000
74000	4.57409	0.00000	4.10006	0.00000	3.74165	0.00000	3.46290	-0.00000	3.24141	-0.00000	3.06237
74100											
74200											
74300											
74400											
74500											
74600	J = 1	1.05407	0.00946	-0.06386	-0.17969	-2.17726	1.30080	1.75289	-2.35902	0.99482	1.35102
74700		-1.27218									
74800	J = 2	0.09199	-0.01678	0.12962	0.23842	-1.63679	-0.65025	1.62619	1.30540	-1.17223	-0.99595
74900		0.67261									
75000	J = 3	-0.14817	-0.01482	0.27346	0.19518	0.85676	-0.34804	-2.45327	0.50034	2.46250	-0.32998
75100		-0.98946									
75200	J = 4	-0.00560	0.00706	0.00830	0.22146	0.35303	-0.73910	-1.28967	1.22848	1.99487	-0.70358
75300		-1.06058									
75400	J = 5	0.00609	-0.01300	-0.15698	0.25859	1.06743	-1.09828	-2.79267	1.64747	3.20478	-0.79404
75500		-1.32815									
75600	J = 6	0.00030	-0.01009	-0.05990	0.14915	0.62418	-0.50594	-1.88688	0.62018	2.24264	-0.25234
75700		-0.92001									
75800	J = 7	0.00096	0.00142	-0.01278	-0.01756	0.05647	0.09382	-0.03964	-0.17823	-0.08303	0.10064
75900		0.07818									

THE POLYNOMIAL COEFFICIENTS

THE MATRIX BS

TABLE 4 - (continued)

	X	Y	S(1)=*****
76000			
76100			
76200			
76300			
76400	-0.00200	0.00000	
76500	-0.00197	-0.01188	
76600	-0.00187	-0.02373	
76700	-0.00171	-0.03552	
76800	-0.00150	-0.04721	
76900	-0.00126	-0.05879	
77000	-0.00101	-0.07024	
77100	-0.00076	-0.08153	
77200	-0.00054	-0.09267	
77300	-0.00035	-0.10365	
77400	-0.00021	-0.11446	
77500	-0.00013	-0.12512	
77600	-0.00011	-0.13563	
77700	-0.00014	-0.14598	
77800	-0.00022	-0.15620	
77900	-0.00033	-0.16627	
78000	-0.00046	-0.17620	
78100	-0.00059	-0.18597	
78200	-0.00069	-0.19556	
78300	-0.00076	-0.20497	
78400	-0.00077	-0.21415	
78500	-0.00072	-0.22310	
78600	-0.00061	-0.23176	
78700	-0.00043	-0.24011	
78800	-0.00020	-0.24813	
78900	0.00006	-0.25579	
79000	0.00034	-0.26307	
79100	0.00062	-0.26995	
79200	0.00087	-0.27644	
79300	0.00108	-0.28253	
79400	0.00122	-0.28824	
79500	0.00130	-0.29356	
79600	0.00130	-0.29853	
79700	0.00122	-0.30315	
79800	0.00109	-0.30744	
79900	0.00091	-0.31141	
80000	0.00070	-0.31506	
80100	0.00048	-0.31839	
80200	0.00028	-0.32140	
80300	0.00010	-0.32406	
80400	-0.00003	-0.32637	
80500	-0.00011	-0.32829	
80600	-0.00013	-0.32982	
80700	-0.00012	-0.33092	
80800	-0.00007	-0.33159	
80900	0.00000	-0.33181	

TABLE 4 - (continued)

	X	Y	S(2)=-.9500
81100			
81200			
81300			
81400	0.09246	0.00000	
81500	0.09204	-0.01806	
81600	0.09081	-0.03628	
81700	0.08885	-0.05481	
81800	0.08627	-0.07376	
81900	0.08322	-0.09321	
82000	0.07989	-0.11319	
82100	0.07644	-0.13368	
82200	0.07304	-0.15461	
82300	0.06983	-0.17587	
82400	0.06693	-0.19732	
82500	0.06441	-0.21880	
82600	0.06229	-0.24013	
82700	0.06055	-0.26115	
82800	0.05914	-0.28172	
82900	0.05797	-0.30170	
83000	0.05692	-0.32102	
83100	0.05589	-0.33965	
83200	0.05477	-0.35756	
83300	0.05346	-0.37480	
83400	0.05189	-0.39141	
83500	0.05003	-0.40747	
83600	0.04787	-0.42305	
83700	0.04544	-0.43823	
83800	0.04280	-0.45306	
83900	0.04003	-0.46758	
84000	0.03721	-0.48180	
84100	0.03444	-0.49568	
84200	0.03182	-0.50918	
84300	0.02940	-0.52222	
84400	0.02723	-0.53470	
84500	0.02534	-0.54653	
84600	0.02372	-0.55759	
84700	0.02232	-0.56780	
84800	0.02109	-0.57707	
84900	0.01996	-0.58535	
85000	0.01883	-0.59261	
85100	0.01761	-0.59886	
85200	0.01624	-0.60412	
85300	0.01464	-0.60845	
85400	0.01279	-0.61192	
85500	0.01065	-0.61461	

TABLE 4 - (continued)

	AN(I, J)						
201200							
201300							
201400							
201500	0.16591	-0.16644	-0.00086	0.00016	-0.00022	-0.00062	0.00007
201600	0.34432	-0.27111	0.00894	0.00412	0.00232	0.00372	0.00016
201700	0.44933	-0.29486	0.01181	0.00264	0.00296	0.00338	0.00049
201800	0.52158	-0.27320	0.01079	0.00015	0.00375	0.00229	0.00030
201900	0.58279	-0.22889	0.00689	-0.00171	0.00479	0.00177	0.00097
202000	0.70380	-0.12268	-0.00905	-0.00303	0.00550	0.00217	0.00089
202100	0.82412	-0.03057	-0.03465	-0.00336	0.00265	0.00206	0.00053
202200	0.92493	0.03424	-0.06531	-0.00413	-0.00167	0.00058	0.00017
202300	0.99372	0.07303	-0.09475	-0.00493	-0.00377	-0.00094	0.00001
202400	1.03136	0.09142	-0.11833	-0.00513	-0.00198	-0.00117	0.00009
202500	1.04730	0.09630	-0.13450	-0.00492	0.00221	-0.00023	0.00036
202600	1.05244	0.09457	-0.14406	-0.00499	0.00568	0.00063	0.00071
202700	1.05407	0.09199	-0.14817	-0.00560	0.00609	0.00030	0.00096
202800	1.05399	0.09168	-0.14664	-0.00597	0.00358	-0.00110	0.00096
202900	1.04899	0.09302	-0.13752	-0.00464	0.00049	-0.00219	0.00070
203000	1.03268	0.09160	-0.11817	-0.00049	-0.00046	-0.00137	0.00037
203100	0.99791	0.08055	-0.08729	0.00629	0.00142	0.00162	0.00029
203200	0.93919	0.05292	-0.04656	0.01442	0.00397	0.00485	0.00067
203300	0.85530	0.00412	-0.00089	0.02349	0.00419	0.00534	0.00128
203400	0.75152	-0.06681	0.04288	0.03612	0.00279	0.00207	0.00113
203500	0.63963	-0.15837	0.07622	0.05578	0.00687	-0.00084	-0.00100
203600	0.58464	-0.21171	0.08440	0.06598	0.01310	0.00008	-0.00276
203700	0.53079	-0.27067	0.08160	0.06994	0.02025	0.00289	-0.00427
203800	0.47541	-0.33642	0.05972	0.05556	0.02147	0.00543	-0.00414
203900	0.41105	-0.41076	0.00449	0.00054	0.00126	0.00128	0.00025

sets and different values of KPK are tried in order to get the smoothest representation in the lengthwise direction. Three or four runs should be sufficient to develop a good choice of KPK.

3.2 TBL18

The primary output from this program is the boundary-layer parameters obtained using the small cross-flow assumption. Input data is entered either through the keyboard in answer to prompting from the computer, or it is read from one of two stored files: File 8 TITLE = "BNSAL" is the BNS file created by MAP22; FILE 10 TITLE = "TBLINP" is a file created by the present program during a previous run.

The example form of TBLINP is shown in Table 5.

Line 100: 40 is the free stream velocity U

Line 200: is the beam-to-length ratio BL

Line 300: NSTAT the number of stations at which the boundary layer is calculated.

Line 400: S1, S2 the first and last stations in the calculations.

Line 500: NTH, the number of streamlines being used.

Lines 600-2100: THATA values of the streamlines at station S1.

Lines 2200-3700: initial values of H, θ_{11} , $\tan \beta$ for each value of THATA.

Line 3800: Reynolds number RE

File BNS is read first and is printed at the terminal, so a quick glance can determine if the correct file was loaded. Table 6 shows the terminal display as it should appear after going through all the computer prompts. The computer first prompts you for a choice of input method: Enter 1 if the input data is on cards, 2 if it is to come from the terminal, and 3 if it is to be read from TBLINP. If you choose 1 or 3, you merely wait for the program to finish; if you choose 2, the computer will prompt you to enter U0. It prompts you by displaying the desired variable with an equal sign, some space, and the format; in the case of U0 one gets U0=1F10.5. Simply enter a space and 40.0, and it responds with 40.00000. The computer will then prompt you for the remaining variables BL, NSTAT, S1, etc. These are the variables that are previously described under FILE TBLINP. In running the example for the first time, have Table 5 next to you and simply type the numbers from it.

In general the choice of θ values depends on how the streamlines behave at the other stations. If they converge too closely in a region of interest addi-

TABLE 5 - FILE TBLINP

100	40.00000		
200	0.14150		
300	21		
400	-0.60000	0.60000	
500	16		
600	0.00000		
700	-0.10472		
800	-0.20944		
900	-0.31416		
1000	-0.41888		
1100	-0.52360		
1200	-0.62832		
1300	-0.73304		
1400	-0.83776		
1500	-0.94248		
1600	-0.04720		
1700	-1.15192		
1800	-1.25664		
1900	-1.36136		
2000	-1.46608		
2100	-1.57080		
2200	1.80000	0.00040	0.00000
2300	1.80000	0.00040	0.00000
2400	1.80000	0.00040	0.00000
2500	1.80000	0.00040	0.00000
2600	1.80000	0.00040	0.00000
2700	1.80000	0.00040	0.00000
2800	1.80000	0.00040	0.00000
2900	1.80000	0.00040	0.00000
3000	1.80000	0.00040	0.00000
3100	1.80000	0.00040	0.00000
3200	1.80000	0.00040	0.00000
3300	1.80000	0.00040	0.00000
3400	1.80000	0.00040	0.00000
3500	1.80000	0.00040	0.00000
3600	1.80000	0.00040	0.00000
3700	1.80000	0.00040	0.00000
3800		1600000.0	

TABLE 6 - TERMINAL DISPLAY FOR TBL18

SSPA MODEL 720

7	11						
1.05407	0.00946	-0.06386	-0.17969	-2.17726	1.30080	1.75289	
-2.35902	0.99482	1.35102	-1.27218				
0.09199	-0.01678	0.12962	0.23842	-1.63679	-0.65025	1.62619	
1.30540	-1.17223	-0.99895	0.67261				
-0.00017	-0.01482	0.27346	0.19518	0.85676	-0.34804	-2.45327	
0.00034	2.46250	-0.32998	-0.98946				
-0.00560	-0.00706	0.00830	0.22146	0.35303	-0.73910	-1.28967	
1.22848	1.99487	-0.70358	-1.06058				
0.00609	-0.01300	-0.15698	0.25859	1.06743	-1.09828	-2.79267	
1.64747	3.20478	-0.79404	-1.32815				
0.00030	-0.01009	-0.05990	0.14915	0.62418	-0.50594	-1.88688	
0.62018	2.24264	-0.25234	-0.92001				
0.00096	0.00142	-0.01278	-0.01756	0.05647	0.09382	-0.03964	
-0.17823	-0.08303	0.10064	0.07818				

NINPT DENOTES THE TYPE OF INPUT

- NINPT=1 CARD DATA
- 2 REMOTE
- 3 STORED DATA

```

NINPT  111
N?
2
UU=    1F10.5
  40.0
  40.00000
BL=    1F10.5
  0.14150
  0.14150
NSTAT=  115
  21
  21
    S1      82      7F10.5
  -0.6      0.6
  -0.60000  0.60000
NTH=    115
  16
  16
TTH=    1F10.5
  0.0000
  -0.10472
  -0.20944
  -0.341416
  -0.41888
    
```

TABLE 6 - (continued)

-0.52360
 -0.62832
 -0.73304
 -0.83776
 -0.9248
 -0.04720
 -1.15192
 -1.25664
 -1.36136
 -1.46608
 -1.57080

0.00000
 -0.10472
 -0.20944
 -0.34142
 -0.41888
 -0.52360
 -0.62832
 -0.73304
 -0.83776
 -0.92480
 -0.04720
 -1.15192
 -1.25664
 -1.36136
 -1.46608
 -1.57080

HB	THMB	BETB 3F10.5
1.80000	0.0004	0.0
1.80000	0.00040	0.00000
1.8	0.0004	0.0
1.80000	0.00040	0.00000
1.8	0.0004	0.0
1.80000	0.00040	0.00000
1.8	0.0	0.0
1.80000	0.00000	0.00000
1.8	0.0004	0.0
1.80000	0.00040	0.00000
1.8	0.0004	0.0
1.80000	0.00040	0.00000
1.8	0.0004	0.0
1.80000	0.00040	0.00000
1.8	0.0004	0.0
1.80000	0.00040	0.00000
1.8	0.0004	0.0
1.80000	0.00040	0.00000
1.8	0.0004	0.0
1.80000	0.00040	0.00000

TABLE 6 - (continued)

1.8	0.0004	0.0
1.80000	0.00040	0.00000
1.8	0.0004	0.0
1.80000	0.00040	0.00000
1.8	2.0004	0.0
1.80000	0.00040	0.00000
1.8	0.0004	0.0
1.80000	0.00040	0.00000
1.8	0.0004	0.0
1.80000	0.00040	0.00000
1.8	0.0004	0.0
1.80000	0.00040	0.00000
RE=	1F20.1	
1600000.0		
	1600000.0	

tional streamlines should be presented. The initial boundary layer parameters have to be determined experimentally at this time. Interval size should be halved and the solution repeated until it converges, since the integration method is the simple Euler method.

In the process of entering the data from the terminal the computer creates FILE TBLSAVE. This file is exactly FILE TBLINP with a different name; thus, once a set of data has been entered it is saved. If you are nearly finished entering data and an error is made simply continue with the rest of the entries and make editorial corrections to TBLSAVE.

All the computational results are output to the printer. Table 7 shows the output for the example case.

Line 300: INPUT signifies that the lines between 300 and 2300 are the input data.

Line 500: N is the number of parameters used in the mapping, and KPK is the degree plus one of the longitudinal polynomials, which are used to interpret the mapping parameters. Here, as before, BL is the beam-to-length ratio.

Line 600: NTH and NSTAT

Line 700: S1 and S2

Lines 800-2100: present the components of the matrix BN(I,J).

Line 2300: Alpha-numeric Title identifying the data

The remaining data are organized by station and are tabulated in twenty columns each headed by the variable name. Lines 49600-54000 show the example data for station $S = -0.0286$.

TH	θ coordinate of streamline measured from x-axis
X	x-coordinate of the streamline
Y	y-coordinate of the streamline
CP	pressure coefficient
PSI	$\partial p / \partial S$ rate of change in pressure along streamline
DTDS	$d\theta / ds$
R	metric along the streamline
K1	geodesic curvature of the streamlines
K2	geodesic curvature of the orthogonal trajectories to the streamlines
PHI	potential at (S, θ)

S(11) = -0.0286

TABLE 7 - (continued)

49500	TH	A	Y	CF	FSI	BTUS	R	K1	K2	PHI	
49500	0.0000	0.14167	0.00000	-0.06067	-0.07010	0.00000	.18083E+01	0.15363	0.00000	-0.02928	
49500	-0.08270	0.14160	-0.01603	-0.06029	-0.06811	0.02576	.28599E+01	0.14098	0.02108	-0.02930	
49500	-0.17462	0.14143	-0.03362	-0.05934	-0.06232	0.05063	.73034E+01	0.10569	0.02549	-0.02936	
50200	-0.28409	0.14129	-0.05373	-0.05881	-0.05392	0.07350	.35798E+01	0.05935	-0.00799	-0.02947	
50800	-0.44192	0.14109	-0.07917	-0.06166	-0.04383	0.09961	.27284E+00	0.03128	-0.10535	-0.02965	
50900	-0.97222	0.11231	-0.11589	-0.07080	-0.00513	-0.00392	.10558E+00	-0.26025	0.03616	-0.03004	
51000	-1.16073	0.08351	-0.11658	-0.06923	0.00428	-0.02026	.26076E+01	-0.00244	0.01301	-0.03006	
51000	-1.24728	0.06741	-0.11651	-0.06898	0.00788	-0.01138	.25394E+01	0.02966	0.00230	-0.03006	
51000	-1.30579	0.05581	-0.11651	-0.06898	0.01010	-0.00481	.22215E+01	0.02937	-0.00247	-0.03006	
51000	-1.35367	0.04601	-0.11655	-0.06906	0.01179	-0.00040	.30584E+01	0.01927	-0.00468	-0.03007	
51400	-0.03668	0.14165	-0.00712	-0.06060	-0.06970	0.01164	.19587E+01	0.15104	0.01039	-0.02928	
51500	-1.43678	0.02859	-0.11669	-0.06926	0.01429	0.00351	.18349E+02	-0.00955	-0.00511	-0.03008	
51600	-1.47371	0.02074	-0.11676	-0.06933	0.01513	0.00352	.44627E+01	-0.02247	-0.00419	-0.03009	
51700	-1.50802	0.01343	-0.11681	-0.06939	0.01570	0.00272	.28622E+01	-0.03212	-0.00291	-0.03009	
51800	-1.54011	0.00657	-0.11683	-0.06942	0.01603	0.00145	.23597E+01	-0.03788	-0.00148	-0.03010	
51900	-1.57080	-0.00000	-0.11684	-0.06943	0.01613	-0.00000	.22320E+01	-0.03977	0.00000	-0.03010	
52000	BOUNDARY LAYER AT S(11) = -0.0286										
52100	H	TH //	DE	BETA	CF	KS	RIH	CFU	CFB	CFS	
52100											
52400											
52500	1.48872	0.00163	0.00242	0.00000	3.61366	0.	.1341E+04	0.00014	3.61366	3.61365	
52500	1.48933	0.00162	0.00242	0.01136	3.61365	0.	.1337E+04	-0.04090	3.61365	3.61364	
52700	1.49091	0.00161	0.00240	0.02372	3.61478	0.	.1327E+04	-0.08559	3.61478	3.61477	
52900	1.49305	0.00157	0.00234	0.03282	3.63944	0.	.1291E+04	-0.11930	3.63944	3.63942	
52900	1.49815	0.00133	0.00200	0.03242	3.83824	0.	.1099E+04	-0.12429	3.83824	3.83823	
53000	1.49227	0.00113	0.00169	0.03145	4.12243	0.	.9367E+03	-0.12948	4.12243	4.12241	
53100	1.47315	0.00150	0.00221	0.04327	3.80806	0.	.1242E+04	-0.16464	3.80806	3.80805	
53300	1.46550	0.00165	0.00241	0.05156	3.72506	0.	.1362E+04	-0.19192	3.72506	3.72505	
53300	1.46108	0.00173	0.00252	0.05715	3.68684	0.	.1428E+04	-0.21054	3.68684	3.68683	
53400	1.45820	0.00177	0.00258	0.05824	3.67360	0.	.1461E+04	-0.21380	3.67360	3.67359	
53500	1.48885	0.00163	0.00242	0.00499	3.61367	0.	.1340E+04	-0.01789	3.61367	3.61366	
53600	1.45421	0.00180	0.00262	0.05075	3.67255	0.	.1488E+04	-0.18622	3.67255	3.67254	
53700	1.45277	0.00181	0.00262	0.04277	3.67585	0.	.1494E+04	-0.15705	3.67585	3.67584	
54000	1.45178	0.00181	0.00262	0.03145	3.68022	0.	.1495E+04	-0.11560	3.68022	3.68020	
54000	1.45129	0.00181	0.00262	0.01684	3.68400	0.	.1494E+04	-0.06191	3.68400	3.68398	
54000	1.45105	0.00181	0.00262	-0.00000	3.68551	0.	.1494E+04	0.00015	3.68551	3.68549	

H shape parameter
 TH11 θ_{11}
 DE H θ_{11}
 BETA β
 CF coefficient of skin friction
 RS not used
 RTH local Reynolds number
 CFW skin-friction coefficient in normal direction
 CFB skin-friction coefficient in crossflow direction
 CFS skin-friction coefficient in streamwise direction

3.3 TBL3D

The primary output from TBL3D are the surface coordinates, distance metrics, and geodesic curvatures. The actual operation of the program is very similar to the operation of TBL18. Input data is entered either through the keyboard in response to the computer's prompting or it is entered from one of two stored files: FILE 8 TITLE = "BNSAL" is the BNS file created by MAP 22, FILE 10 TITLE = "TBL3DINP" is a file created by the present program during a previous run.

The example form of TBL3DINP is shown in Table 8.

TABLE 8 - FILE TBL3DINP

100	0.14150
200	-0.60000
300	0.60000
400	21
500	16

Table 8: File TBL3DINP

Line 100: BL
 Line 200: START value of S at initial station
 Line 300: SFINIS value of S at final station
 Line 400: NSTAT number of stations
 Line 500: NTH number of θ -values used

There is no reason to enter the θ -values as they are equally spaced from 0 to $-\pi/2$.

Since one does not have to enter θ or the initial boundary layer parameters,

the initial run with this program is simpler than it is with TBL18. Table 9 shows the terminal display as it would appear for TBL3D after going through an initial run with it. Again the same menu for NINPT. One then enters a 1, if a BL other than 1 is to be used followed by BY, START, SFINIS, NSTAT, and NTH. After the computer responds with the value of NTH, 16 in this case, it prints the NTH values of θ .

In general, several values of NTH should be tried such as 20, 30, 40, and 50 to check the accuracy of the final calculation. The number of stations should be chosen experimentally after examining the results from TBL3D. Making several runs with NTH = 20 and NSTAT chosen so that $\Delta S = 0.050, 0.025, 0.020,$ and 0.010 should provide a good feel for how the solution behaves with respect to ΔS . If some area needs further refinement of the grid in the s-direction for convergence, the program can be run just for this area with appropriate choices of START and SFINIS.

The output file TBLSAVE is created in the process of entering the data from the terminal; it is exactly TBL3DINP with a different name.

All the computational results are saved in TBL3DOPT and also an identical copy on the PRINTER file. Table 10 shows typical output.

Line 100: identifies the data

Line 200: N, KPK, BL

Line 300: START, SFINIS, NSTAT, NTH

Line 400: Station number, value of S at the station

Except for lines 400, 5300, and 6200, lines beginning with 1, 2, 3 have the following interpretation:

1	x	y	θ	u	v
2	w	u_ϕ	u_θ	u_n	S
3	K_ϕ	K_θ	H	F	G

The values (x, y, θ , S) are the respective coordinates of the grid point. u, v, w, are the inviscid velocities of the fluid in the (x, y, s) coordinate system; u_ϕ and u_θ are the velocities in the ϕ and θ -directions respectively, while u_n is a normal velocity to the surface. Since the inviscid velocities are obtained here with slender body theory, u_n is not identically zero. K_ϕ and K_θ are the respective geodesic curvatures. H, F, and G are functions used in computing the metrics of the surface coordinates and are defined in Equations (22) and (23).

TABLE 9 - TERMINAL DISPLAY FOR TBL3D

SSPA MODEL 720

7	11	1.00000						
1.05407	0.00946	-0.06386	-0.17969	-2.17726	1.30080	1.75289		
-2.35902	0.99482	1.35102	-1.27218					
0.09199	-0.01678	0.12962	0.23842	-1.63679	-0.65025	1.62619		
1.30540	-1.17223	-0.99895	0.67261					
-0.14817	-0.01482	0.27346	0.19518	0.85676	-0.34804	-2.45327		
0.50034	2.46250	-0.32998	-0.98946					
-0.00560	-0.00706	0.00830	0.22146	0.35303	-0.73910	-1.28967		
1.22848	1.99487	-0.70358	-1.06058					
0.00609	-0.01300	-0.15698	0.25859	1.06743	-1.09828	-2.79267		
1.64747	3.20478	-0.79404	-1.32815					
0.00030	-0.01009	-0.05990	0.14915	0.62418	-0.50594	-1.88688		
0.62018	2.24264	-0.25234	-0.92001					
0.00096	0.00142	-0.01278	-0.01756	0.05647	0.09382	-0.03964		
-0.17823	-0.08303	0.10064	0.07818					

NINPT DENOTES THE TYPE OF INPUT
 NINPT=1 CARD DATA
 2 REMOTE
 3 STORED DATA

NINPT 111
 #?
 2
 DO YOU WISH TO ENTER NEW BL?
 IF YES TYPE 1 NO TYPE 0
 1
 BL= 1F10.5
 0.1415
 0.14150
 START= 1F10.5
 -0.6
 -0.60000
 SFINIS= 1F10.5
 0.6
 0.60000
 NSTAT= 113
 21
 21
 NTH= 113
 16
 16
 0.00000 -0.10472 -0.20944 -0.31416 -0.41888 -0.52360 -0.62832
 -0.73304 -0.83776 -0.94248 -1.04720 -1.15192 -1.25664 -1.36136
 -1.46608 -1.57080

TABLE 10 - FILE TBL3DOPT

SSPA MODEL 720						
100	7	11	0.14150			
200						
300	-0.60000	0.60000	21	16		
400	1-0.60000					
500	1	.1076487E+00	0.	0.	.2228504E+00	0.
600	2	.1015192E+01	.1039358E+01	0.	-.3304451E-02	-.6000000E+00
700	3	-.1746801E+01	0.	.1024530E+01	0.	.1319586E+00
800	1	.1070765E+00	-.1383975E-01	-.1047198E+00	.2260201E+00	-.2191698E-01
900	2	.1013320E+01	.1038446E+01	.1965947E-02	-.2949163E-02	-.6000375E+00
1000	3	-.1503672E+01	-.1295564E+00	.1025467E+01	.7161124E-03	.1329239E+00
1100	1	.1055012E+00	-.2772837E-01	-.2094395E+00	.2319952E+00	-.5121243E-01
1200	2	.1008394E+01	.1035967E+01	-.8533613E-02	-.1934473E-02	-.6001429E+00
1300	3	-.9778611E+00	-.2079391E+00	.1027713E+01	.1298230E-02	.1337157E+00
1400	1	.1032734E+00	-.4142935E-01	-.3141593E+00	.2340332E+00	-.9026146E-01
1500	2	.1002341E+01	.1032607E+01	-.3647421E-01	-.5599877E-03	-.6002810E+00
1600	3	-.4966123E+00	-.2322699E+00	.1029947E+01	.1340198E-02	.1305186E+00
1700	1	.1005888E+00	-.5440810E-01	-.4188790E+00	.2253837E+00	-.1386364E+00
1800	2	.9977760E+00	.1029084E+01	-.8100021E-01	.5415959E-03	-.6003663E+00
1900	3	-.2271720E+00	-.2369821E+00	.1031042E+01	.2899263E-03	.1218179E+00
2000	1	.9729968E-01	-.6610241E-01	-.5235988E+00	.1966588E+00	-.1923716E+00
2100	2	.9979270E+00	.1025890E+01	-.1381673E+00	.5000537E-03	-.6002711E+00
2200	3	-.2092135E+00	-.2620306E+00	.1030629E+01	-.2110958E-02	.1102132E+00
2300	1	.9300203E-01	-.7623009E-01	-.6283185E+00	.1398962E+00	-.2356202E+00
2400	2	.1005483E+01	.1023250E+01	-.1975938E+00	-.1257294E-02	-.5998721E+00
2500	3	-.3959048E+00	-.3209001E+00	.1028985E+01	-.5511206E-02	.1011199E+00
2600	1	.8731204E-01	-.8487507E-01	-.7330383E+00	.6810752E-01	-.2464628E+00
2700	2	.1017983E+01	.1021275E+01	-.2421879E+00	-.3708640E-02	-.5991074E+00
2800	3	-.6347227E+00	-.3778387E+00	.1026544E+01	-.9095525E-02	.9855563E-01
2900	1	.8008766E-01	-.9231333E-01	-.8377580E+00	.7477900E-02	-.2276362E+00
3000	2	.1029067E+01	.1020127E+01	-.2648933E+00	-.5146658E-02	-.5979988E+00
3100	3	-.8234540E+00	-.3865944E+00	.1023555E+01	-.1207957E-01	.1015647E+00
3200	1	.7144016E-01	-.9876676E-01	-.9424778E+00	-.3567305E-01	-.1962318E+00
3300	2	.1036653E+01	.1020007E+01	-.2720005E+00	-.5376520E-02	-.5966297E+00
3400	3	-.1024584E+01	-.3442213E+00	.1020069E+01	-.1407044E-01	.1065337E+00
3500	1	.6158783E-01	-.1042907E+00	-.1047198E+01	-.6534221E-01	-.1586447E+00
3600	2	.1041305E+01	.1020983E+01	-.2670684E+00	-.4768951E-02	-.5951080E+00
3700	3	-.1324703E+01	-.2781963E+00	.1016111E+01	-.1499069E-01	.1113336E+00
3800	1	.5071863E-01	-.1088297E+00	-.1151917E+01	-.8108497E-01	-.1162588E+00
3900	2	.1042699E+01	.1022812E+01	-.2472945E+00	-.3561000E-02	-.5935497E+00
4000	3	-.1750850E+01	-.2162065E+00	.1011814E+01	-.1477011E-01	.1158369E+00
4100	1	.3896382E-01	-.1123276E+00	-.1256637E+01	-.8049241E-01	-.7350396E-01
4200	2	.1040407E+01	.1024973E+01	-.2091745E+00	-.2133289E-02	-.5920854E+00
4300	3	-.2265229E+01	-.1680927E+00	.1007493E+01	-.1319614E-01	.1203060E+00
4400	1	.2645479E-01	-.1147821E+00	-.1361357E+01	-.6403464E-01	-.3700986E-01
4500	2	.1035503E+01	.1026886E+01	-.1524454E+00	-.9640050E-03	-.5908674E+00

TABLE 10 - (continued)

4600	3	-.2774616E+01	-.1236639E+00	.1003665E+01	-.1006735E-01	.1243714E+00
4700	1	.1338044E-01	-.1162293E+00	-.1466077E+01	-.3535184E-01	-.1254040E-01
4800	2	.1030611E+01	.1028144E+01	-.8053472E-01	-.3055445E-03	-.5900534E+00
4900	3	-.3155131E+01	-.6822705E-01	.1000980E+01	-.5482609E-02	.1272438E+00
5000	1	-.3423824E-11	-.1167067E+00	-.1570796E+01	.9717779E-11	-.3919891E-02
5100	2	.1028565E+01	.1028572E+01	.2101379E-10	-.1119698E-03	-.5897664E+00
5200	3	-.3296611E+01	.1903118E-10	.1000008E+01	.1400675E-11	.1282839E+00
5300	2	0.540000				
5400	1	.1195187E+00	0.	0.	.1726273E+00	0.
5500	2	.1033713E+01	.1048012E+01	0.	-.5734943E-02	-.5400000E+00
5600	3	-.1740499E+01	0.	.1014791E+01	0.	.1470473E+00
5700	1	.1191601E+00	-.1535795E-01	-.1047198E+00	.1772038E+00	-.1206576E-01
5800	2	.1032026E+01	.1047081E+01	.1461010E-01	-.5599279E-02	-.5401326E+00
5900	3	-.1518471E+01	-.4281959E-01	.1015801E+01	.2532992E-02	.1459863E+00
6000	1	.1180539E+00	-.3042280E-01	-.2094395E+00	.1877014E+00	-.3271569E-01
6100	2	.1027007E+01	.1044418E+01	.1453239E-01	-.5038358E-02	-.5404757E+00
6200	3	-.9943291E+00	-.8043367E-01	.1018293E+01	.4022324E-02	.1420645E+00
6300	1	.1162987E+00	-.4481263E-01	-.3141593E+00	.1959246E+00	-.6581593E-01
6400	2	.1019780E+01	.1040483E+01	-.6936500E-02	-.3939546E-02	-.5408890E+00
6500	3	-.4560700E+00	-.1161479E+00	.1020887E+01	.3873435E-02	.1343139E+00
6600	1	.1138514E+00	-.5808049E-01	-.4188790E+00	.1928358E+00	-.1103753E+00
6700	2	.1012894E+01	.1035865E+01	-.4796494E-01	-.2676672E-02	-.5412007E+00
6800	3	-.1365952E+00	-.1631869E+00	.1022421E+01	.2080112E-02	.1231138E+00
6900	1	.1104692E+00	-.6985761E-01	-.5235988E+00	.1689584E+00	-.1599221E+00
7000	2	.1009655E+01	.1031139E+01	-.1013624E+00	-.2015065E-02	-.5412615E+00
7100	3	-.1269329E+00	-.2349296E+00	.1022553E+01	-.9208378E-03	.1113327E+00
7200	1	.1057922E+00	-.8000459E-01	-.6283185E+00	.1197554E+00	-.1984719E+00
7300	2	.1012270E+01	.1026784E+01	-.1553451E+00	-.2455138E-02	-.5409783E+00
7400	3	-.3184498E+00	-.3224631E+00	.1021645E+01	-.4491291E-02	.1033845E+00
7500	1	.9953330E-01	-.8862767E-01	-.7330383E+00	.5818337E-01	-.2093423E+00
7600	2	.1019080E+01	.1023169E+01	-.1970950E+00	-.3464412E-02	-.5403230E+00
7700	3	-.4661877E+00	-.3861478E+00	.1020180E+01	-.8029229E-02	.1019603E+00
7800	1	.9161058E-01	-.9594043E-01	-.8377580E+00	.3955355E-02	-.1950477E+00
7900	2	.1026277E+01	.1020569E+01	-.2229923E+00	-.4124788E-02	-.5393205E+00
8000	3	-.5293835E+00	-.3977186E+00	.1018335E+01	-.1112082E-01	.1058141E+00
8100	1	.8212461E-01	-.1020975E+00	-.9424778E+00	-.3674202E-01	-.1666526E+00
8200	2	.1032070E+01	.1019121E+01	-.2359362E+00	-.4142793E-02	-.5380306E+00
8300	3	-.6824672E+00	-.3690810E+00	.1015992E+01	-.1351804E-01	.1120865E+00
8400	1	.7123166E-01	-.1071360E+00	-.1047198E+01	-.6320235E-01	-.1296369E+00
8500	2	.1035474E+01	.1018731E+01	-.2349075E+00	-.3510063E-02	-.5365384E+00
8600	3	-.1017475E+01	-.3273791E+00	.1013031E+01	-.1498298E-01	.1193080E+00
8700	1	.5904541E-01	-.1110407E+00	-.1151917E+01	-.7316435E-01	-.8907492E-01
8800	2	.1035281E+01	.1019042E+01	-.2159651E+00	-.2411822E-02	-.5349592E+00
8900	3	-.1465833E+01	-.2884922E+00	.1009573E+01	-.1517921E-01	.1273743E+00
9000	1	.4564362E-01	-.1138421E+00	-.1256637E+01	-.6700109E-01	-.5243209E-01

3.4 CBLGEO

CBLGEO sets up the panels for the potential calculation. It computes the (S, x, y) coordinates of the panel corners and stores these values in FILE 7 TITLE = "CBL3DOPT".

Input data is again entered through the keyboard in response to the computer's prompting or it is entered from one of two stored files: FILE 8 TITLE = "BNSAL", the BNS file created by MAP 22, or FILE 10 TITLE = "CBL3DINP", a file created by the present program during a previous run. This latter file has the same entries as TBL3DINP.

Table 11 shows the example problem being run from the terminal with entries made remotely NINPT = 2. Note that there is no need to enter START or SFINIS as these values are set by the program. After NTH = 10 is entered and echoed the ten values of θ are printed at the terminal.

There are two output files: FILE 11 TITLE = "CBLTAPE" and CBL3DOPT. CBLTAPE is created in the process of entering the data from the terminal; it is exactly CBL3DINP with a different name. CBL3DOPT contains all the panel data to be passed onto the potential program. Table 12 shows the output for the example case. Only the first four panels are shown.

Line 100: identifies the data

Line 200: N, KPK, BL

Line 300: START, SFINIS, NSTAT, NTH

Line 400: 1 first panel

Note that lines 400, 1000, 1600, 2200 give the panel number for the next five lines of data. Lines beginning with 1, 2, 3, 4, and 5 have the following nomenclature:

1	S(I)	x(I,J)	y(I,J)	TTH(J)
2	S(I+1)	x(I+1,J)	y(IH,J)	TTH(J)
3	S(I+1)	x(I+1, J+1)	y(I+1, J+1)	TTH(J+1)
4	S(I)	x(I, J+1)	y(I, J+1)	TTH(J+1)
5	SE(MD)	XE(MP)	YE(MP)	TTE(MP)

TABLE 11 - TERMINAL DISPLAY FOR CBL

SSPA MODEL 720

?	11	1.00000					
1.05407	0.00946	-0.06386	-0.17969	-2.17726	1.30080	1.75289	
-2.35902	0.99482	1.35102	-1.27218				
0.09199	-0.01678	0.12962	0.23842	-1.63679	-0.65025	1.62619	
1.30540	-1.17223	-0.99895	0.67261				
-0.14817	-0.01482	0.27346	0.19518	0.85676	-0.34804	-2.45327	
0.50034	2.46250	-0.32998	-0.98946				
-0.00560	-0.00706	0.00830	0.22146	0.35303	-0.73910	-1.28967	
1.22848	1.99487	-0.70358	-1.06058				
0.00609	-0.01300	-0.15698	0.25859	1.06743	-1.09828	-2.79267	
1.64747	3.20478	-0.79404	-1.32815				
0.00030	-0.01009	-0.05990	0.14915	0.62418	-0.50594	-1.88688	
0.62018	2.24264	-0.25234	-0.92001				
0.00096	0.00142	-0.01278	-0.01756	0.05647	0.09382	-0.03964	
-0.17823	-0.08303	0.10064	0.07818				

NINPT DENOTES THE TYPE OF INPUT

- NINPT=1 CARD DATA
- 2 REMOTE
- 3 STORED DATA

NINPT 111

N?

2

DO YOU WISH TO ENTER NEW BL?

IF YES TYPE 1 NO TYPE 0

1

BL= 1F10.5

0.1415

0.14150

NSTAT= 113

25

25

NTH= 113

10

10

0.00000 -0.17453 -0.34907 -0.52360 -0.69813 -0.87266 -1.04720

-1.22173 -1.39626 -1.57080

TABLE 12 - FILE CBL3DOPT

		SEPA	ADJST	7	11	14	18
		-1.00000	1.00000				
100							
200							
300							
400	1						
500	1	-1.00000	0.00000	0.00000	0.00000	0.00000	0.00000
600	2	-0.91667	0.02089	0.00000	0.00000	0.00000	0.00000
700	3	-0.91667	0.01938	-0.01841	-0.01841	-0.01841	-0.17453
800	4	-1.00000	0.00000	-0.00832	-0.00832	-0.00832	-0.17453
900	5	-0.95833	0.01075	-0.00611	-0.00611	-0.00611	-0.08727
1000	2						
1100	1	-1.00000	0.00000	-0.00832	-0.00832	-0.00832	-0.17453
1200	2	-0.91667	0.01938	-0.01841	-0.01841	-0.01841	-0.17453
1300	3	-0.91667	0.01652	-0.03242	-0.03242	-0.03242	-0.34907
1400	4	-1.00000	0.00000	-0.01600	-0.01600	-0.01600	-0.34907
1500	5	-0.95833	0.00379	-0.01938	-0.01938	-0.01938	-0.26180
1600	3						
1700	1	-1.00000	0.00000	-0.01600	-0.01600	-0.01600	-0.34907
1800	2	-0.91667	0.01652	-0.03242	-0.03242	-0.03242	-0.34907
1900	3	-0.91667	0.01474	-0.04934	-0.04934	-0.04934	-0.52360
2000	4	-1.00000	0.00000	-0.02352	-0.02352	-0.02352	-0.52360
2100	5	-0.95833	0.00710	-0.03357	-0.03357	-0.03357	-0.47633
2200	4						
2300	1	-1.00000	0.00000	-0.02352	-0.02352	-0.02352	-0.52360
2400	2	-0.91667	0.01474	-0.04934	-0.04934	-0.04934	-0.52360
2500	3	-0.91667	0.01301	-0.06395	-0.06395	-0.06395	-0.69813
2600	4	-1.00000	0.00000	-0.03030	-0.03030	-0.03030	-0.69813
2700	5	-0.95833	0.00641	-0.04668	-0.04668	-0.04668	-0.61087

The lines 1 thru 3 give the corner coordinates of the panel, the fourth and fifth line gives the panel midpoint coordinates, and the remaining lines give the values of the potential will be given.

3.5 DOUBDD

DOUBDD computes the potential for flow of the slender body based on the panel geometry from CBL3DPT. Since it requires a large amount of data, it is an actively simple set procedural time at 25 seconds per panel. It is used in conjunction with the appropriate time requirements. The program is used in the following manner: FILE 7 TITLE = "CBL3DPT" which is the input file, FILE 8 TITLE = "DOUBDDPT" which is the output file, FILE 8 TITLE = "DOUBDDPT". Table 14 shows the output of the program. For the example case, only the first 300 lines of the output file are shown as follows:

- Line 100: NP the total number of panels used.
- Line 200: NSMC number of panels in each direction, the first number is the number of panels in first direction.
- Line 300: S-coordinate at which the potential is calculated, the first number is the S-coordinate and PFI value of potential at S-coordinate.

typical of the remaining lines in the file.

3.6 TBLIPC

This program is mainly a utility program used to format the data from CBL3DPT and the geometry data for input into the boundary-layer program. The program is reasonably fast taking about a minute for slender body data and about four minutes for the data from DOUBDD. There are two input files, FILE 7 TITLE = "TBL3DPT" and FILE 12 TITLE = "DOUBDDPT". Both files are required, but only the data from DOUBDDPT is used, but only TBLIPC is used in the case of the slender body potential flow.

In the interactive operation there is only one data value to be given by the operator. If the data is from DOUBDD, the data is simply identified by the number 0. The number that appears on the screen identifies the data being entered.

Table 14 shows the screen for the test case. The data is identified as data for the SSPA Model 720. In the next line N=7, EPE=11, and the bearing length

100	216			
200	24			
300		.11081E-01	-.61080E-02	-.98096E+00
400		.10129E-01	-.18814E-01	-.98506E+00
500		.88904E-02	-.31775E-01	-.98556E+00
600		.78013E-02	-.43626E-01	-.98103E+00
700		.65537E-02	-.54083E-01	-.98019E+00
800		.50183E-02	-.63442E-01	-.97167E+00
900		.35301E-02	-.70900E-01	-.96938E+00
1000		.22263E-02	-.75735E-01	-.95868E+00
1100		.87134E-03	-.81059E-01	-.99335E+00
1200		.31066E-01	-.84092E-02	-.92329E+00
1300		.28746E-01	-.25893E-01	-.91733E+00
1400		.25951E-01	-.43865E-01	-.91827E+00
1500		.23302E-01	-.60468E-01	-.92244E+00
1600		.19858E-01	-.74786E-01	-.92817E+00
1700		.15450E-01	-.87115E-01	-.92725E+00
1800		.10947E-01	-.97344E-01	-.91640E+00
1900		.66362E-02	-.10474E+00	-.93054E+00
2000		.22285E-02	-.10854E+00	-.92566E+00

TABLE 14 - TERMINAL DISPLAY FOR TBLIPC

```

SSPA MODEL 720
7      11  0.14150
-0.60000  0.60000  21  16
16  -.1047198E+00  -.9549294E+01
IF THREE DIMENSIONAL POTENTIAL IS FROM MING'S PROGRAM
ENTER 1 SLENDER BODY ENTER 0
8?
1
216
24      9
216
    
```

ratio is 0.14150. The initial station for the data is $S = -0.6$, and the last station is $S=0.0$. There are 21 stations to be used in the lengthwise direction, and 16 values of θ . The 16 values of θ are equally spaced with $\Delta\theta = -0.1047198$ and $1/\Delta\theta = -9.549294$. In the case shown the doublet distribution potential is used so a 1 appears as the answer to the question. There were 216 panels used in the doublet method: 24 in the lengthwise direction and 9 in the girth direction. The final 216 indicates that the program is reading the data for the 216 panels.

There is only one output file, FILE 7 TITLE = "TBLIPCOPT". A representative portion of it is shown in Table 15.

Line 100: Data identification

Line 200: N, KPK, and BL

Line 300: START, SFINS, NSTAT, and NTH

Line 400: The station number and the value of S at this station.

Lines beginning with 1, 2, and 3 have the data organized as follows:

1	x	y	θ	u_{ϕ}	u_{θ}
2	S	$du_{\phi}/d\zeta_{\phi}$	$du_{\phi}/d\zeta_{\theta}$	$du_{\theta}/d\zeta_{\phi}$	$du_{\theta}/d\zeta_{\theta}$
3	K_{ϕ}	K_{θ}	H	F	G

3.7 TBL SOL

This final program of the suite, TBL SOL, computes the boundary-layer momentum thickness, $TH11$, the tangent of the crossflow angle T, and the shape parameter H. Input data is entered either through the keyboard in answer to prompting from the computer, or it is read from the stored file: FILE 8 TITLE = "TBLIPCOPT". This file is created by TBLIPC and is described in Section 3.6.

TABLE 15 - FILE TBLIPCOPT

	SSPA MODEL 720	11	0.14157	16	21
100	-0.50000	0.	.1421511E+00	0.	.5599764E+01
200	1-0.50000	0.	.1421511E+00	0.	.1024530E+01
300	1	.1076487E+00	.1383975E-01	.1047198E+00	.1036446E+01
400	2	.6090000E+00	.1419029E+00	.1218052E+00	.2077872E+00
500	3	.1740601E+01	.1295564E+00	.1025467E+01	.7161124E-03
600	1	.1070760E+00	.2772837E-01	.2094395E+00	.1035957E+01
700	2	.6090000E+00	.1390912E+00	.2094959E+00	.3796330E+00
800	3	.1740601E+01	.2079321E+00	.102713E+01	.1298230E-02
900	1	.1032734E+00	.4142935E-01	.1141593E+00	.1032607E+01
1000	2	.6090000E+00	.1390973E+00	.2517947E+00	.4871595E+00
1100	3	.1740601E+01	.2322699E+00	.1029947E+01	.134019E-02
1200	1	.1035482E+00	.5600510E-01	.4188790E+00	.1029034E+01
1300	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
1400	3	.1740601E+01	.2354821E+00	.1031042E+01	.2899263E-03
1500	1	.1032734E+00	.4142935E-01	.1141593E+00	.1025890E+01
1600	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
1700	3	.1740601E+01	.2322699E+00	.1029947E+01	.134019E-02
1800	1	.1035482E+00	.5600510E-01	.4188790E+00	.1029034E+01
1900	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
2000	3	.1740601E+01	.2354821E+00	.1031042E+01	.2899263E-03
2100	1	.1032734E+00	.4142935E-01	.1141593E+00	.1025890E+01
2200	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
2300	3	.1740601E+01	.2322699E+00	.1029947E+01	.134019E-02
2400	1	.1035482E+00	.5600510E-01	.4188790E+00	.1029034E+01
2500	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
2600	3	.1740601E+01	.2354821E+00	.1031042E+01	.2899263E-03
2700	1	.1032734E+00	.4142935E-01	.1141593E+00	.1025890E+01
2800	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
2900	3	.1740601E+01	.2322699E+00	.1029947E+01	.134019E-02
3000	1	.1035482E+00	.5600510E-01	.4188790E+00	.1029034E+01
3100	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
3200	3	.1740601E+01	.2354821E+00	.1031042E+01	.2899263E-03
3300	1	.1032734E+00	.4142935E-01	.1141593E+00	.1025890E+01
3400	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
3500	3	.1740601E+01	.2322699E+00	.1029947E+01	.134019E-02
3600	1	.1035482E+00	.5600510E-01	.4188790E+00	.1029034E+01
3700	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
3800	3	.1740601E+01	.2354821E+00	.1031042E+01	.2899263E-03
3900	1	.1032734E+00	.4142935E-01	.1141593E+00	.1025890E+01
4000	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
4100	3	.1740601E+01	.2322699E+00	.1029947E+01	.134019E-02
4200	1	.1035482E+00	.5600510E-01	.4188790E+00	.1029034E+01
4300	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
4400	3	.1740601E+01	.2354821E+00	.1031042E+01	.2899263E-03
4500	1	.1032734E+00	.4142935E-01	.1141593E+00	.1025890E+01
4600	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
4700	3	.1740601E+01	.2322699E+00	.1029947E+01	.134019E-02
4800	1	.1035482E+00	.5600510E-01	.4188790E+00	.1029034E+01
4900	2	.6090000E+00	.1390973E+00	.2532725E+00	.5461084E+00
5000	3	.1740601E+01	.2354821E+00	.1031042E+01	.2899263E-03

Table 16 shows the example problem being run from the terminal. The first two rows of numbers are the polynomial coefficients in the skin friction formula given by Head and Patel⁹. These are fixed values and appear as data items in the program itself. In the example case the number of stations at which the boundary layer is to be calculated is NS=21 with only one iteration at each step, NT=1. TOL is the present value of tolerance used in the program; any number less than TOL is considered zero. There should be no need to change the value from 10^{-9} . If a larger TOL is needed, it is likely that something is going wrong in the computation or the initial data is not physically correct.

In the example, constant values were used for the initial data; these were TH11 = 0.0004, T=0, and H=1.4 where TH11 has been nondimensionalized with respect to the semi-hull length. These values are entered at the appropriate prompting.

The initial data will generally be obtained from experimental data. In the four cases that this program was used with experimental data, different interpolation methods were used. This amounted to writing a subroutine to interpolate the experimental data in each case as a function of θ and simply calling the subroutine to set the initial values. Enter a call for this subroutine at line 2500 in Subroutine INITIAL and change 2500 from STOP to RETURN. If initial data is entered as a function of θ merely respond with a 1 and the program will ask for the initial data at $j\Delta\theta$, $j = 0, 1, 2, \dots, NTH$.

The output for the test case is presented in Table 17.

Line 100: NS and NI

Line 200: Title

Line 300: N, KPK, and BL

Line 400: START, SFINIS, NSTAT, and NTH

Line 500: TH11, T, and H

Line 1200: Station 1, S at initial station

Lines 1300 Through 6100 have the following format:

1	x	y	θ	u_ϕ	u_θ
2	s	$du_\phi/d\ell_\phi$	$du_\theta/d\ell_\theta$	$du_\phi/d\ell_\phi$	$du_\theta/d\ell_\theta$
3	K	K	H	F	G

Lines 6200 through 7900 are typical of the output for each station. Line 6200 has

TABLE 16 - TERMINAL DISPLAY FOR TBL SOL

```
RUN TBL SOL
MRUNNING 4359
-0.000701  0.028345 -0.386748  0.019521
-0.001953  0.062588 -0.834800  0.191511
NS IS NUMBER OF STATIONS AT WHICH BOUNDARY LAYER IS
TO BE CALCULATED
NS=  113
M?
  21
  21
NI IS NUMBER OF ITERATIONS AT EACH STEP
NI=  113
  1
  1
TOL=  .10000E-03
IF THE INITIAL VALUES ARE CONSTANT ENTER 0
OTHERWISE ENTER A 1
0
TH11=  1F10.5
  .0004
  0.00040
T= TAN(BETA)=  1F10.5
  0.0
  0.00000
H=  1F10.5
```

TABLE 17 - PRINTER FILE TBL SOL

21	10	SSPA MODEL 720		T=	H=		
7	11	0.14150	21	0.0000	1.4000		
100	-0.60000	0.60000	0.60000				
200	0.00040	0.00000	1.40000				
300	INITIAL VALUES						
400	TH11=	0.0004	T=	0.0000	H=	1.4000	
500	1-0.600000	0.	0.	0.	.1039358E+01	0.	
600	1	.1076487E+00	.1421311E+00	.6599764E-01	0.	-.1422674E+00	
700	2	-.6000000E+00	.1024530E+01	0.		.1319586E+00	
800	3	-.1746801E+01	0.	0.		.1965947E-02	
900	1	.1070765E+00	-.1383975E-01	-.1047198E+00	.1038446E+01	.3065287E+00	
1000	2	-.6000375E+00	.1419029E+00	.1218052E+00	.2077872E+00	.1329239E+00	
1100	3	-.1503672E+01	-.1295564E+00	.1025467E+01	.7161124E-03	-.8533613E-02	
1200	1	.1055012E+00	-.2772837E-01	-.2094395E+00	.1035967E+01	.1372600E+01	
1300	2	-.6001429E+00	.1390912E+00	.2084958E+00	.3796330E+00	.1337157E+00	
1400	3	-.9778611E+00	-.2079391E+00	.1027713E+01	.1298230E-02	-.3647421E-01	
1500	1	.1032734E+00	-.4142935E-01	-.3141593E+00	.1032607E+01	.2650982E+01	
1600	2	-.6002810E+00	.1298973E+00	.2517947E+00	.4871595E+00	.1305186E+00	
1700	3	-.4966123E+00	-.2322699E+00	.1029947E+01	.1340198E-02		
1800	2-0.540000						
1900	THATA	X	Y	S	TH11	T	H
2000	0.0000	0.1195	0.0000	-0.5400	.56653797E-03	0.	.15446615E+01
2100	-0.1047	0.1192	-0.0154	-0.5401	.56673740E-03	.11607606E-01	.15327303E+01
2200	-0.2094	0.1181	-0.0304	-0.5405	.56797616E-03	.10985056E-01	.15079857E+01
2300	-0.3142	0.1163	-0.0448	-0.5409	.56935764E-03	-.42664793E-02	.14840518E+01
2400	-0.4189	0.1139	-0.0581	-0.5412	.56752441E-03	-.31504725E-01	.14629403E+01
2500	-0.5236	0.1105	-0.0699	-0.5413	.56117968E-03	-.67427577E-01	.14446639E+01
2600	-0.6283	0.1058	-0.0800	-0.5410	.55403075E-03	-.10651129E+00	.14341994E+01
2700	-0.7330	0.0995	-0.0886	-0.5403	.55261012E-03	-.14042963E+00	.14376116E+01
2800	-0.8378	0.0916	-0.0959	-0.5393	.55793645E-03	-.16423704E+00	.14528439E+01
2900	-0.9425	0.0821	-0.1021	-0.5380	.56499103E-03	-.17867669E+00	.14724043E+01
3000	-1.0472	0.0712	-0.1071	-0.5365	.57059685E-03	-.18476568E+00	.14936899E+01
3100	-1.1519	0.0590	-0.1110	-0.5350	.57523754E-03	-.18079522E+00	.15193881E+01
3200	-1.2566	0.0456	-0.1138	-0.5334	.58074369E-03	-.16254177E+00	.15528803E+01
3300	-1.3614	0.0311	-0.1157	-0.5322	.58815735E-03	-.12539799E+00	.15928778E+01
3400	-1.4661	0.0158	-0.1167	-0.5313	.59551192E-03	-.68898905E-01	.16282336E+01
3500	-1.5708	-0.0000	-0.1170	-0.5310	.59867344E-03	0.	.16426010E+01

the station number and station s coordinate. The remaining lines give the boundary-layer parameters around the girth at station s as a function of θ .

Running time for this program has always been less than five minutes.

4. CONCLUDING REMARKS

This report presents a suite of computer programs for computing three dimensional boundary layers for ships. Most of the technical details for carrying out the computations are explained and demonstrated in the form of a worked out example. The computer programs are in a structured form that permits modification to empirical formulas as better empirical approximations are developed, e.g. an improved crossflow model when new experimental data becomes available. Alternatively, individual programs described in this report, such as MAP22, and DOUBDD can be used independently or with other methods for calculating the boundary layer.

There are some choices that must be made in the input data to these programs. The number of hull offsets to be used in obtaining the rational hull approximation together with the degree of the lengthwise approximating polynomial and the degree of the conformal mapping are the most difficult choices to be made. The values used in the example are a good starting guess; however, these numbers should be determined by comparing the agreement of the approximating offsets with the physical hull offsets and by examining the smoothness of the approximating hull shape. Changing the degree of the lengthwise polynomial and the number of parameters in the conformal mapping will change the smoothness of the approximating hull form. 250 panels should be used in the potential calculation; the panel dimensions should be chosen so that the longest panel dimension is in the direction of the freestream flow. In the boundary layer calculation 40 girthwise values should be used.

von Kerczek and Langan concluded that the present boundary layer calculations method can predict boundary layers on relatively fine double ship models with fair accuracy to within a distance of the stern of about 10 percent of the ship's length for the SSPA Model 720. In the stern area, the boundary layer thickens very rapidly and approaches separation. Calculation of this near-separated boundary layer region must await further developments of boundary layer theory.

REFERENCES

1. Tuck, E.O. and C. von Kerczek, "Streamlines and Pressure Distribution on Arbitrary Ship Hulls at Zero Froude Number," *Journal of Ship Research*, Vol. 12, pp. 231-236 (1968).
2. von Kerczek, C., "Calculation of the Turbulent Boundary Layer on a Ship Hull," *Journal of Ship Research*, Vol. 17, pp. 106-120 (1973).
3. von Kerczek, C. and E.O. Tuck, "The Representation of Ship Hulls by Conformal Mapping Functions," *Journal of Ship Research*, Vol. 13, pp. 284-298 (1969).
4. Chang, M.S. and P.C. Pien, "Hydrodynamic Forces on a Body Moving Beneath a Free Surface," *Proceedings of the First International Conference on Numerical Ship Hydrodynamics*, J. Schot and N. Salvesen, editors, Gaithersburg (1975).
5. von Kerczek, C. and T.J. Langan, "An Integral Prediction Method for Three-Dimensional Turbulent Boundary Layers on Ships," *David W. Taylor Naval Ship R&D Center Report 79/006* (1979).
6. Larsson, L., "Boundary Layers on Ships, Part IV: Calculations of the Turbulent Boundary Layer on a Ship Model," *The Swedish State Shipbuilding Experimental Tank, Goteborg, Sweden, Report 47* (1974).
7. Cumpsty, N.A. and M.R. Head, "The Calculation of Three-Dimensional Turbulent Boundary Layers, Part I: Flow Over the Rear of an Infinite Swept Wing," *The Aeronautical Quarterly*, Vol. 18, pp. 55-84 (1965).
8. Smith, P.D., "An Integral Prediction Method for Three-Dimensional Compressible Turbulent Boundary Layers," *Royal Aeronautical Establishment, Reports and Memoranda 3739* (1972).
9. Head, M.R. and V.C. Patel, "Improved Entrainment Method for Calculating Turbulent Boundary Layer Development," *Aeronautical Research Council (Great Britain), R and M 3643* (1968).

APPENDIX
PROGRAM LISTINGS

```

100 FILE 5(TITLE="PFN",KIND=DISK,FILETYPE=7)
110 FILE 8(TITLE="ANS",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
115 FILE 9(TITLE="BNS",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
120 FILE 6(KIND=PRINTER,MAXRECSIZE=22)
130 $RESET FREE
135 C      PROGRAM MAP22
140      COMMON /A1/ XMS(50,50),YMS(50,50)
150      COMMON /A2/ AN(50,10)
160      COMMON /A3/ BN(25,10)
170      COMMON /A4/ BS(25,25)
180 C
190 C      MAP FITS AN ANALYTIC SURFACE TO A SHIP HULL
200 C      POINTS AT THE WATERLINE AND AT THE KEEL CONSTRAINED - ALL STATIONS
210
220 C      XMS      X OFFSET OF A POINT ON THE HULL
230 C      YMS      Y OFFSET OF A POINT ON THE HULL
240 C      AN(I,J)   MAPPING COEFFICIENTS AT STATION I
250 C      BN(I,J)   CALCULATED A SUB MN
260 C      BS(I,J)   COEFFICIENT MATRIX
270 C      KT(I)    NUMBER OF OFFSETS GIVEN AT STATION I
280      COMMON /A5/ KT(50)
290      COMMON /A6/ IFS(50)
300      COMMON /A7/ JFS(50)
310      COMMON /A8/ TITLE(12)
320      COMMON /A9/ THETA(50)
330      COMMON /B1/ S(50)
340      COMMON /B2/ AP(10)
350      COMMON /B3/ B(50),H(50),SA(50)
360      CALL INPUT(K,N,IST1,IST2,KP1,KP2,TOL,BL)
370      CALL CONTRA(K,N,TOL)
380      IF(K.LE.2) GO TO 5
390      CALL POLY(1,IST1,N,KP1)
400      5 CALL OUTPUT(1,IST1,N,KP1,BL)
410      IF(IST1-K) 1,2,2
420      1 CALL POLY(IST2,K,N,KP2)
430      CALL OUTPUT(IST2,K,N,KP2,BL)
440      IF(IST2-IST1) 2,2,3
450      3 DO 4 J=1,N
460          BN(1,J)=BL*AN(IST1,J)
470      4 BN(2,J)=0.0
480      CALL OUTPUT(IST1,IST2,N,2,BL)
490      2 STOP
500      END

```

```

510      SUBROUTINE INPUT(K,N,IST1,IST2,KP1,KP2,TOL,BL)
520
530 C    SUBROUTINE INPUT CONTAINS ALL READ STATEMENTS FOR MAP
540
550 C      XMS      X OFFSET OF A POINT ON THE HULL
560 C      YMS      Y OFFSET OF A POINT ON THE HULL
570 C      AN(I,J)   MAPPING COEFFICIENTS AT STATION I
580 C      BN(I,J)   CALCULATED A SUB MN
590 C      BS(I,J)   COEFFICIENT MATRIX
600 C      KT(I)    NUMBER OF OFFSETS GIVEN AT STATION I
610
620 C      IFS(I)    1 NO CONSTRAINTS AT STATION I
630 C                2 SURFACE AT STATION I TO PASS THROUGH OFFSETS
640 C                3 AT STATION I SLOPE WITH RESPECT TO S IS 0
650 C                4 2 AND 3 COMBINED
660 C                5 AT STATION I SLOPE AND CURVATURE ARE 0
670 C                6 2,3,AND 5 COMBINED
680
690 C      JFS(I)    0 AN READ IN AT STATION I
700 C                1 COMPUTE AN
710
720 C      TITLE    IDENTIFIES COMPUTATIONS- 144 CHARACTERS MAX
730 C      THETA(J) PHI SUB J AT A GIVEN STATION
740 C      S(I)     S AT STATION I
750 C      AP      MAPPING COEFFICIENTS AT A GIVEN STATION
760 C      B(I)    B EAM AT STATION I
770 C      H(I)    DRAFT AT STATION I
780 C      SA(I)   CROSS SECTIONAL AREA AT STATION I
790
800      COMMON /A1/ XMS(50,50),YMS(50,50)
810      COMMON /A2/ AN(50,10)
820      COMMON /A5/ KT(50)
830      COMMON /A6/ IFS(50)
840      COMMON /A7/ JFS(50)
850      COMMON /AB/ TITLE(12)
860      COMMON /B1/ S(50)
870      DIMENSION IX(50),IY(50)
880      READ(5,24) (TITLE(J),J=1,12)
890      READ(5,3) IST1,IST2,KP1,KP2
900      READ(5,1) K,N,TOL,BL
910      DO 58 I=1,K
920      DO 58 J=1,N
930      58 AN(I,J)=0.0
940      READ(5,3) (IFS(I),I=1,K)
950      READ(5,3) (JFS(I),I=1,K)

```

```

960     READ(5,25)(KT(I),I=1,K)
970     DO 33 I=1,K
980     M=KT(I)
990     READ(5,4) (IX(J),IY(J),J=1,M)
1000    DO 34 J=1,M
1010    XMS(I,J)=FLOAT(IX(J))
1020    34 YMS(I,J)=FLOAT(IY(J))
1030    33 CONTINUE
1040    XMAX=XMS(10,1)
1050    DO 35 I=1,14
1060    M=KT(I)
1070    DO 35 J=1,M
1080    XMS(I,J)=XMS(I,J)/XMAX
1090    35 YMS(I,J)=YMS(I,J)/XMAX
1100    DO 36 I=15,K
1110    M=KT(I)
1120    DO 36 J=1,M
1130    XMS(I,J)=-XMS(I,J)/XMAX
1140    36 YMS(I,J)=YMS(I,J)/XMAX
1150    READ(5,2) (S(I),I=1,K)
1160    DO 31 I=1,K
1170    IF(JFS(I)) 31,32,31
1180    32 READ(5,2) (AN(I,J),J=1,N)
1190    31 CONTINUE
1200    WRITE(6,40)
1210    WRITE(6,24) (TITLE(J),J=1,12)
1215    WRITE(9,24) (TITLE(J),J=1,12)
1220    WRITE(6,19)
1230    DO 21 I=1,K
1240    M=KT(I)
1250    WRITE(6,22) I,S(I)
1260    21 WRITE(6,23)(XMS(I,J),YMS(I,J),J=1,M)
1270    100 RETURN
1280    1 FORMAT(2I10,2F10.5)
1290    4 FORMAT(6(1X,2I6))
1300    2 FORMAT (7F10.5)
1310 3   FORMAT(14I5)
1320    25 FORMAT(7I10)
1330    24 FORMAT(12A6)
1340    40 FORMAT(1H1)
1350    19 FORMAT(19H THE INPUT OFFSETS   ///)
1360    22 FORMAT(//56H                      XMS                      YMS
1370    IS(,I2,2H)=F5.2//)
1380    23 FORMAT(F30.5,F16.5)
1390    END

```



```

1400      SUBROUTINE CONTRA(K,N,TOL)
1410
1420 C      SUBROUTINE CONTRA COMPUTES THE PARAMETERS IN THE CONFORMAL MAP OF TH
1430 C      UNIT CIRCLE ONTO THE CROSS SECTION
1440
1450 C          XMS      X OFFSET OF A POINT ON THE HULL
1460 C          YMS      Y OFFSET OF A POINT ON THE HULL
1470 C          AN(I,J)   MAPPING COEFFICIENTS AT STATION I
1480 C          BS(I,J)   COEFFICIENT MATRIX
1490 C          KT(I)     NUMBER OF OFFSETS GIVEN AT STATION I
1500
1510 C          JFS(I)    0 AN READ IN AT STATION I
1520 C                  1 COMPUTE AN
1530
1540 C          THETA(J)  PHI SUB J AT A GIVEN STATION
1550 C          AP        MAPPING COEFFICIENTS AT A GIVEN STATION
1560 C          B(I)      B EAM AT STATION I
1570 C          H(I)      DRAFT AT STATION I
1580 C          SA(I)     CROSS SECTIONAL AREA AT STATION I
1590
1600      COMMON /A1/ XMS(50,50),YMS(50,50)
1610      COMMON /A2/ AN(50,10)
1620      COMMON /A4/ BS(25,25)
1630      COMMON /A5/ KT(50)
1640      COMMON /A7/ JFS(50)
1650      COMMON/A9/ THETA(50)
1660      COMMON /B2/ AP(10)
1670      COMMON /B3/ B(50),H(50),SA(50)
1680      DIMENSION ANU(25,1),IP(25)
1690      5 FORMAT(23H THE MATRIX IS SINGULAR I5,F10.5)
1700      70 FORMAT(36H BEST FIT AT THIS NN NOT YET REACHED//)
1710      100 FORMAT(///13H INITIAL AN(I2,4H,J)=,3F9.5,21H A-EA COEFFICIENT=
1720      1F9.5//)
1730      104 FORMAT(10F12.8)
1740      107 FORMAT(18H NO CONVERGENCE AT I5)
1750      108 FORMAT(17H NO LEWIS FORM AT I5)
1760      110 FORMAT(6H DUT=F8.5,5H AN=11F9.5)
1770      THETA(1)=0.0
1780      PI=-3.14159265
1790      DO 103 L=1,K
1800      M=KT(L)
1810      B(L)=XMS(L,1)
1820      H(L)=-YMS(L,M)
1830      SUM=0.0
1840      DO 112 J=2,M

```

```

1850 112 SUM=SUM+0.5*(XMS(L,J)-XMS(L,J-1))*(YMS(L,J)-YMS(L,J))
1860 SA(L)=2.*SUM
1870 103 CONTINUE
1880 DO 54 I=1,K
1890 IF(JFS(I)) 11,54,11
1900 11 AN(I,2)=0.5*(B(I)-H(I))
1910 BH=B(I)+H(I)
1920 HBB=BH**2+8.*(2.*SA(I)/PI+B(I)*H(I))
1930 IF(HBB) 63,64,64
1940 63 WRITE(6,108) I
1950 AN(I,3)=0.0
1960 GO TO 54
1970 64 AN(I,3)=-0.25*(BH-SQRT(HBB))
1980 54 CONTINUE
1990 DO 10 I=1,K
2000 IF(JFS(I)) 12,13,12
2010 13 WRITE(6,6) I, (AN(I,J),J=1,N)
2020 6 FORMAT(//15H AN GIVEN AT I=I2,4H AN=10F9.5)
2030 GO TO 10
2040 12 M=KT(I)
2050 THETA(M)=-1.57079632
2060 MM=M-1
2070 INT=-1
2080 DUTP=0.0
2090 DO 91 J=1,10
2100 91 AP(J)=0.0
2110 FAC=B(I)*H(I)*2.
2120 IF(FAC) 86,92,86
2130 92 SIGMA=-1.0
2140 GO TO 65
2150 86 SIGMA=SA(I)/FAC
2160 65 WRITE(6,100) I,AN(I,1),AN(I,2),AN(I,3),SIGMA
2170 NN=2
2180 67 NN=NN+1
2190 NN2=NN-2
2200 IF(NN-N) 68,68,21
2210 68 KLK=0
2220 35 KLK=KLK+1
2230 IF(KLK-20) 55,55,56
2240 56 WRITE(6,107) I
2250 IF(NN-N) 67,10,10
2260 55 IF(INT) 1,1,2
2270 2 DO 28 J=1,NN2
2280 U=J+2
2290 UJ=(-1.)** ( J+2)

```

```

2300      DO 29 JJ=1,NN2
2310      UN=JJ+2
2320      ON=(1.-1.)**(JJ+2)
2330      SUM=0.0
2340      DO 30 JJJ=2,MM
2350      TH1=THETA(JJJ)
2360      TH2=(3.-2.*OJ)*TH1
2370      TH3=(3.-2.*ON)*TH1
2380      C1=COS(TH1)
2390      C2=COS(TH2)
2400      C3=COS(TH3)
2410      S1=SIN(TH1)
2420      S2=SIN(TH2)
2430      S3=SIN(TH3)
2440      30 SUM=SUM + (C1+C2 - C1*C2 - C1*C3 + C2*C3 + ON*OJ*S1*S1 + ON*OJ*S2
2450      1 + OJ*S1*S3 + S2*S3)
2460      29 BS(J,JJ)=SUM
2470      28 CONTINUE
2480      DO 26 J=1,NN2
2490      U=J+2
2500      OJ=(-1.)**(J+3)
2510      SUM=0.0
2520      DO 27 JJ=2,MM
2530      TH1=THETA(JJ)
2540      TH2=(3.-2.*U)*TH1
2550      C1=COS(TH1)
2560      C2=COS(TH2)
2570      S1=SIN(TH1)
2580      S2=SIN(TH2)
2590      27 SUM=SUM + (XMS(I,JJ) - B(I)*C1) * (C1-C2) +
2600      1 (YMS(I,JJ) - H(I)*S1) * (OJ*S1 - S2)
2610      26 ANU(J,1)=SUM
2620      CALL DECOMP(NN2,25,BS,IP)
2630      CALL SOLVE(NN2,25,BS,ANU,IP)
2640      52 DO 59 J=1,NN2
2650      59 AN(I,J+2)=ANU(J,1)
2660      1 SUM1=0.0
2670      SUM2=0.0
2680      DO 200 J=3,NN
2690      V1=1.-(-1.)**J
2700      V2=1.+(-1.)**J
2710      SUM1=SUM1+V1*AN(I,J)
2720      200 SUM2=SUM2+V2*AN(I,J)
2730      AN(I,1)=0.5*(B(I)+H(I)-SUM1)
2740      AN(I,2)=0.5*(B(I)-H(I)-SUM2)

```

```

2750      CALL ATHETA(I,NN,MM,IBTH,N)
2760      IF (IBTH) 21,21,4
2770      4 DUT=0.0
2780      DO 32 J=1,M
2790      SUM1=0.0
2800      SUM2=0.0
2810      DO 33 JJ=1,NN
2820      PJ=JJ
2830      ARG=(3.-2.*PJ)*THETA(J)
2840      SUM1=SUM1+AN(I,JJ)*COS(ARG)
2850      33 SUM2=SUM2+AN(I,JJ)*SIN(ARG)
2860      32 DUT=DUT+(XMS(I,J)-SUM1)**2+(YMS(I,J)-SUM2)**2
2870      TUD=DUT-DUTP
2880      IF (INT) 78,78,79
2890      78 INT=1
2900      WRITE(6,110) DUT,(AN(I,J),J=1,NN)
2910      DO 3 J=1,NN
2920      3 AP(J)=AN(I,J)
2930      DUTP=DUT
2940      GO TO 67
2950      79 IF (TUD) 88,88,87
2960      87 DO 89 J=1,NN
2970      89 AN(I,J)=AP(J)
2980      WRITE(6,90) DUT
2990      90 FORMAT(25H FIT IS GOING BAD DUT=F7.5)
3000      GO TO 67
3010      88 IF (ABS(TUD)-TOL) 69,69,66
3020      69 WRITE(6,110) DUT,(AN(I,J),J=1,NN)
3030      DO 77 J=1,NN
3040      77 AP(J)=AN(I,J)
3050      DUTP=DUT
3060      GO TO 67
3070      66 DUTP=DUT
3080      WRITE(6,110) DUT,(AN(I,J),J=1,NN)
3090      DO 76 J=1,NN
3100      76 AP(J)=AN(I,J)
3110      GO TO 35
3120      21 WRITE(8,111) (AN(I,J),J=1,N)
3130      111 FORMAT(7F10.5)
3140      10 CONTINUE
3150      50 RETURN
3160      END
#

```

```

3170     SUBROUTINE ATHETA(I,NN,MM,IBTH,N)
3180     COMMON /A1/ XMS(50,50),YMS(50,50)
3190     COMMON /A2/ AN(50,10)
3200     COMMON /A9/ THETA(50)
3210     IBTH=1
3220     II=MM+1
3230     KMM=1000
3240     OME1=0.0174533
3250     DO 11 J=2,MM
3260         THETA(J)=THETA(J-1)
3270     OMEG=THETA(J-1)
3280     DTHE=(OMEG-OME1)/10.
3290     JJJ=-1
3300     KKK=1
3310     LLL=1
3320     23 SUM1=0.0
3330     SUM2=0.0
3340     KKK=KKK+1
3350     DO 15 L=1,NN
3360     PN=L
3370     ARG=(3.-2.*PN)*THETA(J)
3380     SUM1=SUM1+AN(I,L)*COS(ARG)
3390     15 SUM2=SUM2+AN(I,L)*SIN(ARG)
3400     DU=(XMS(I,J)-SUM1)**2+(YMS(I,J)-SUM2)**2
3410     IF(JJJ) 12,13,14
3420     12 DU1=DU
3430     JJJ=0
3440     GO TO 16
3450     13 IF(DU-DU1) 17,18,19
3460     18 DU1=DU
3470     GO TO 16
3480     17 JJJ=1
3490     DU1=DU
3500     GO TO 16
3510     14 IF(DU-DU1) 20,19,19
3520     20 DU1=DU
3530     16 IF(KKK-KMM) 21,21,37
3540     21 THETA(J)=THETA(J)+DTHE
3550     GO TO 23
3560     37 IBTH=-1
3570     WRITE(6,4) I,J
3580     4 FORMAT(16H BAD THETA AT I=12,3H J=12//)
3590     GO TO 10
3600     19 IF(LLL-3) 24,25,25
3610     24 THETA(J)=THETA(J)-2.*DTHE
3620     DTHE=DTHE/10.
3630     JJJ=-1
3640     KKK=1
3650     LLL=LLL+1
3660     GO TO 23
3670     25 THETA(J)=THETA(J)-DTHE
3680     OME1=OMEG
3690     11 CONTINUE
3700     10 RETURN
3710     END

```

```

3720     SUBROUTINE POLY(KB,KE,N,KPK)
3730     COMMON /A2/ AN(50,10)
3740     COMMON /A3/ BN(25,10)
3750     COMMON /A4/ BS(25,25)
3760     COMMON /A6/ IFS(50)
3770     COMMON /B1/ S(50)
3780     COMMON /B3/ B(50),H(50),SA(50)
3790     DIMENSION IP(25),VE(25)
3800     INT=0
3810     DO 12 I=KB,KE
3820     IS=IFS(I)
3830     GO TO (12,13,13,28,28,45),IS
3840 28 INT=INT+2
3850     GO TO 12
3860 13 INT=INT+1
3870     GO TO 12
3880 45 INT=INT+3
3890 12 CONTINUE
3900     MP=KPK+INT
3910     KPK1=KPK+1
3920     WRITE(6,80) MP,KPK,INT
3930 80 FORMAT(/,4H MP=15,5H KPK=15,5H INT=15/)
3940     DO 27 I=1,MP
3950     DO 27 J=1,MP
3960 27 BS(I,J)=0.0
3970     DO 1 II=1,N
3980     DO 1 I=2,KPK
3990     JI=I-1
4000     SUM=0.0
4010     DO 2 J=KB,KE
4020     IS=IFS(J)
4030     GO TO (3,2,3,2,3,2),IS
4040 3 SUM=SUM+(S(J)**JI)*AN(J,II)
4050 2 CONTINUE
4060 1 BN(I,II)=SUM
4070     DO 7 II=1,N
4080     BN(1,II)=0.0
4090     DO 7 J=KB,KE
4100     IS=IFS(J)
4110     GO TO (5,7,5,7,5,7),IS
4120 5 BN(1,II)=BN(1,II)+AN(J,II)
4130 7 CONTINUE
4140     DO 4 II=1,N
4150     I=KPK1
4160     DO 4 J=KB,KE

```

```

4170      IS=IFS(J)
4180      GO TO (4,6,46,29,47,48),IS
4190      29 BN(I,II)=AN(J,II)
4200      BN(I+1,II)=0.0
4210      I=I+2
4220      GO TO 4
4230      46 BN(I,II)=0.0
4240      I=I+1
4250      GO TO 4
4260      47 BN(I,II)=0.0
4270      BN(I+1,II)=0.0
4280      I=I+2
4290      GO TO 4
4300      48 BN(I,II)=AN(J,II)
4310      I=I+1
4320      GO TO 47
4330      6 BN(I,II)=AN(J,II)
4340      I=I+1
4350      4 CONTINUE
4360      WRITE(6,71)
4370      DO 70 II=1,N
4380      70 WRITE(6,72) (BN(I,II),I=1,MF)
4390      71 FORMAT (//19H      THE VECTORS BN //)
4400      72 FORMAT (12F10.5)
4410      DO 8 I=1,KPK
4420      DO 8 J=1,KPK
4430      SUM=0.0
4440      IJ=I+J-2
4450      DO 9 M=KB,KE
4460      IS=IFS(M)
4470      GO TO (10,9,10,9,10,9),IS
4480      10 IF(IJ) 11,14,11
4490      14 IF(S(M)) 11,15,11
4500      15 SUM=SUM+1.0
4510      GO TO 9
4520      11 SUM=SUM+S(M)**IJ
4530      9 CONTINUE
4540      8 BS(I,J)=SUM
4550      L=KPK1
4560      DO 30 J=KB,KE
4570      IS=IFS(J)
4580      GO TO (30,31,49,32,50,51),IS
4590      31 BS(1,L)=1.0
4600      L=L+1
4610      GO TO 30

```

```

4620      49 BS(1,L)=0.0
4630          L=L+1
4640          GO TO 30
4650      50 BS(1,L)=0.0
4660          BS(1,L+1)=0.0
4670          L=L+2
4680          GO TO 30
4690      51 BS(1,L)=1.0
4700          L=L+1
4710          GO TO 50
4720      52 BS(1,L)=1.0
4730          BS(1,L+1)=0.0
4740          L=L+2
4750      30 CONTINUE
4760          DO 16 I=2,KPK
4770              L=KPK1
4780              I1=I-1
4790              DO 17 J=KB,KE
4800                  IS=IFS(J)
4810                  GO TO (17,18,52,33,55,59),IS
4820      18 BS(I,L)=S(J)**I1
4830          L=L+1
4840          GO TO 17
4850      52 I2=I1-1
4860          IF(I2) 53,54,53
4870      54 BS(I,L)=1.0
4880          L=L+1
4890          GO TO 17
4900      53 AI=I1
4910          BS(I,L)=AI*S(J)**I2
4920          L=L+1
4930          GO TO 17
4940      55 I2=I1-1
4950          I3=I1-2
4960          IF(I3) 56,57,58
4970      56 BS(I,L)=1.0
4980          BS(I,L+1)=0.0
4990          L=L+2
5000          GO TO 17
5010      57 AI1=I1
5020          BS(I,L)=AI1*S(J)
5030          BS(I,L+1)=2.0
5040          L=L+2
5050          GO TO 17
5060      58 AI1=I1

```



```

5070      AI2=I2
5080      BS(I,L)=AI1*S(J)**I2
5090      BS(I,L+1)=AI1*AI2*S(J)**I3
5100      L=L+2
5110      GO TO 17
5120 59 BS(I,L)=S(J)**I1
5130      L=L+1
5140      GO TO 55
5150 33 BS(I,L)=S(J)**I1
5160      I2=I1-1
5170      IF(I2) 34,35,34
5180 35 BS(I,L+1)=1.0
5190      GO TO 36
5200 34 AI=I1
5210      BS(I,L+1)=AI*S(J)**I2
5220 36 L=L+2
5230 17 CONTINUE
5240 16 CONTINUE
5250      L=KPK1
5260      DO 19 J=KB,KE
5270      IS=IFS(J)
5280      GO TO (19,20,60,37,62,65),IS
5290 20 DO 21 I=2,KPK
5300      I1=I-1
5310 21 BS(L,I)=S(J)**I1
5320      L=L+1
5330      GO TO 19
5340 60 BS(L,2)=1.0
5350      DO 61 I=3,KPK
5360      AI=I-1
5370      I2=I-2
5380 61 BS(L,I)=AI*S(J)**I2
5390      L=L+1
5400      GO TO 19
5410 62 BS(L,2)=1.0
5420      BS(L+1,2)=0.0
5430      DO 63 I=3,KPK
5440      AI=I-1
5450      I2=I-2
5460 63 BS(L,I)=AI*S(J)**I2
5470      L1=L+1
5480      BS(L1,3)=2.0
5490      DO 64 I=4,KPK
5500      AI1=I-1
5510      AI2=I-2

```

```

5520      I3=I-3
5530      64 BS(L1,I)=AI1*AI2*S(J)**I3
5540      L=L+2
5550      GO TO 19
5560      65 DO 66 I=2,KPK
5570      I1=I-1
5580      66 BS(L,I)=S(J)**I1
5590      L=L+1
5600      GO TO 62
5610      37 DO 44 I=2,KPK
5620      I1=I-1
5630      BS(L,I)=S(J)**I1
5640      I2=I1-1
5650      IF(I2) 38,39,38
5660      39 BS(L+1,I)=1.0
5670      GO TO 44
5680      38 AI=I1
5690      BS(L+1,I)=AI*S(J)**I2
5700      44 CONTINUE
5710      40 L=L+2
5720      19 CONTINUE
5730      L=KPK1
5740      DO 41 J=KB,KE
5750      IS=IFS(J)
5760      GO TO (41,42,67,43,68,69),IS
5770      43 BS(L,1)=1.0
5780      BS(L+1,1)=0.0
5790      L=L+2
5800      GO TO 41
5810      67 BS(L,1)=0.0
5820      L=L+1
5830      GO TO 41
5840      68 BS(L,1)=0.0
5850      BS(L+1,1)=0.0
5860      L=L+2
5870      GO TO 41
5880      69 BS(L,1)=1.0
5890      L=L+1
5900      GO TO 68
5910      42 BS(L,1)=1.0
5920      L=L+1
5930      41 CONTINUE
5940      WRITE(6,73)
5950      73 FORMAT (//18H      THE MATRIX BS //)
5960      DO 74 J=1,MP

```

```
5970 74 WRITE (6,72) (BS(J,I),I=1,MP)
5980 CALL DECOMP(MP,25,BS,IP)
5990 DO 23 I=1,N
6000 DO 24 J=1,MP
6010 VE(J)=BN(J,I)
6020 24 CONTINUE
6030 CALL SOLVE(MP,25,BS,VE,IP)
6040 DO 25 J=1,MP
6050 BN(J,I)=VE(J)
6060 25 CONTINUE
6070 23 CONTINUE
6080 26 RETURN
6090 END
```

```

6100      SUBROUTINE OUTPUT(KB,KE,N,KPK,BL)
6110      COMMON /A1/ XMS(50,50),YMS(50,50)
6120      COMMON /A2/ AN(50,10)
6130      COMMON /A3/ BN(25,10)
6140      COMMON /A5/ KT(50)
6150      COMMON /A8/ TITLE(12)
6160      COMMON /B1/ S(50)
6170      COMMON /B3/ B(50),H(50),SA(50)
6180      IF(KPK.LE.1) GO TO 13
6190      WRITE(6,70)
6200      70 FORMAT(/30H  THE POLYNOMIAL COEFFICIENTS//)
6210      DO 71 J=1,N
6220      IF(KPK-10) 72,72,73
6230      72 WRITE(6,74) J, (BN(I,J),I=1,KPK)
6240      GO TO 71
6250      73 WRITE(6,74) J, (BN(I,J),I=1,10)
6260      WRITE(6,75) (BN(I,J),I=11,KPK)
6270      71 CONTINUE
6280      74 FORMAT (4H J =, I2, 10F10.5)
6290      75 FORMAT(10H           ,10F10.5)
6300      111 FORMAT(7F10.5)
6310      DO 80 I=1,KE
6320      80 WRITE(8,111) (AN(I,J),J=1,N)
6330      13 DO 1 I=KB,KE
6340      IF(KPK.LE.1) GO TO 11
6350      ST=S(I)
6360      WRITE(6,7) I,ST
6370      7 FORMAT(/26H           X           Y,12H           S(I2,2H)=
6380      1F6.4//)
6390      DO 2 J=1,N
6400      SUM=0.0
6410      DO 3 II=2,KPK
6420      II=II-1
6430      3 SUM=SUM+BN(II,J)*ST**II
6440      2 AN(I,J)=SUM+BN(1,J)
6450      11 TH=0.03491
6460      DO 4 L=1,46
6470      X=0.0
6480      Y=0.0
6490      TH=TH-0.03491
6500      DO 5 J=1,N
6510      AJ=3-2*J
6520      X=X+AN(I,J)*COS(AJ*TH)
6530      5 Y=Y+AN(I,J)*SIN(AJ*TH)
6540      4 WRITE(6,6) X,Y

```

```

6550      6 FORMAT(F20.5,F10.5)
6560      1 CONTINUE
6570      IF(KPK.GT.1) GO TO 12
6580      RETURN
6590     12 WRITE(6,8)
6600      3 FORMAT(///17H          AN(I,J)/)
6610      DO 9 I=KB,KE
6620      9 WRITE(6,10) (AN(I,J),J=1,N)
6630     10 FORMAT(F20.5,9F10.5)
6635      WRITE(9,400) N,KPK,BL
6640      DO 76 J=1,N
6650     76 WRITE(9,111) (BN(I,J),I=1,KPK)
6660      RETURN
6665     400 FORMAT(2I5,1F10.5)
6670      END

```

```

6680      SUBROUTINE DECOMP(N,NDIM,A,IP)
6690
6700 C      SUBROUTINE DECOMP GAUSSIAN REDUCTION OF A TO TRIANGULAR FORM
6710
6720 C          N          NUMBER OF NON ZERO ROWS IN A
6730 C          NDIM       DIMENSION OF A
6740 C          A          MATRIX TO BE REDUCED
6750 C          IP         NUMBER OF ROW BEING PIVOTED WITH K ROW
6760
6770      DIMENSION A(NDIM,NDIM),IP(NDIM)
6780      IP(N)=1
6790      DO 6 K=1,N
6800      IF(K.EQ.N) GO TO 5
6810      KP1=K+1
6820      M=K
6830      DO 1 I=KP1,N
6840      IF(ABS(A(I,K)).GT.ABS(A(M,K))) M=I
6850 1 CONTINUE
6860      IP(K)=M
6870      IF(M.NE.K) IP(N)=-IP(N)
6880      T=A(M,K)
6890      A(M,K)=A(K,K)
6900      A(K,K)=T
6910      IF(T.EQ.0.) GO TO 5
6920      DO 2 I=KP1,N
6930 2 A(I,K)=-A(I,K)/T
6940      DO 4 J=KP1,N
6950      T=A(M,J)
6960      A(M,J)=A(K,J)
6970      A(K,J)=T
6980      IF(T.EQ.0.) GO TO 4
6990      DO 3 I=KP1,N
7000 3 A(I,J)=A(I,J)+A(I,K)*T
7010 4 CONTINUE
7020 5 IF(A(K,K).EQ.0.) IP(N)=0
7030 6 CONTINUE
7040      RETURN
7050      END

```

```

7060     SUBROUTINE SOLVE(N,NDIM,A,B,IP)
7070     DIMENSION A(NDIM,NDIM),B(NDIM),IP(NDIM)
7080     IF(N.EQ.1) GO TO 9
7090     NM1=N-1
7100     DO 7 K=1,NM1
7110     KP1=K+1
7120     M=IP(K)
7130     T=B(M)
7140     B(M)=B(K)
7150     B(K)=T
7160     DO 7 I=KP1,N
7170     7 B(I)=B(I)+A(I,K)*T
7180     DO 8 KB=1,NM1
7190     KM1=N-KB
7200     K=KM1+1
7210     B(K)=B(K)/A(K,K)
7220     T=-B(K)
7230     DO 8 I=1,KM1
7240     8 B(I)=B(I) +A(I,K)*T
7250     9 B(1)=B(1)/A(1,1)
7260     RETURN
7270     END

```

```

#FILE (CHXL)TBL18 ON DTNSRDC
100 FILE 5(KIND=REMOTE,MAXRECSIZE=22)
110 FILE 6(KIND=PRINTER,MAXRECSIZE=132)
120 FILE 8(TITLE="BNSAL",KIND=DISK,FILETYPE=7)
130 FILE 9(KIND=REMOTE,MAXRECSIZE=22)
140 FILE 10(TITLE="TBLINP",KIND=DISK,FILETYPE=7)
150 FILE 11(TITLE="TBLSAVE",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
160 $RESET FREE
170 COMMON / /TITLE(12)
180 COMMON /A/TTH(50),HB(50),THMB(50),BEIB(50),DTA(50),
190 1 RS(50),RTH(50),RD(50),CFF(50),XX(50),YY(50)
200 COMMON /B2/ THNB(50),DTHS(50),DGS(50),DTNS(50),USR(50)
210 COMMON /U/AN(10),ANP(10),ANPP(10),CN(10),CNP(10),GN(10),CNPP(10).
220 1PAC
230 COMMON /G/BN(25,10),WK(50),DK(50),TM(50),TMD
240 COMMON / /TITL(3)
250 COMMON /SAVE/GRU(50)
260 DO 120 I=1,25
270 DO 120 J=1,10
280 120 BN(I,J)=0.0
290 READ(8,1) (TITLE(J),J=1,12)
300 WRITE(9,1) (TITLE(J),J=1,12)
310 READ(8,405)N,KPK
320 WRITE(9,405)N,KPK
330 DO 13 J=1,N
340 READ(8,3) (BN(I,J),I=1,KPK)
350 WRITE(9,3) (BN(I,J),I=1,KPK)
360 13 CONTINUE
370 WRITE(9,414)
380 WRITE(9,415)
390 WRITE(9,416)
400 WRITE(9,417)
410 WRITE(9,418)
420 READ(5,412) NINPT
430 GOTO (301,302,303) NINPT
440 301 WRITE(9,419)
450 SIOF
460 302 WRITE(9,400)
470 READ(5,401) UO
480 WRITE(9,401) UO
490 WRITE(9,402)
500 READ(5,401) BL
510 WRITE(9,401) BL
520 WRITE(9,409)
530 READ(5,405) NSTAT
540 WRITE(9,405) NSTAT

```



```

550     WRITE(9,410)
560     READ(5,3) S1,S2
570     WRITE(9,3) S1,S2
580     WRITE(9,404)
590     READ(5,405) NTH
600     WRITE(9,405) NTH
610     WRITE(9,411)
620     READ(5,401) (TTH(I),I=1,NTH)
630     WRITE(9,401) (TTH(I),I=1,NTH)
640     WRITE(9,403)
650     DO 200 IN=1,NTH
660         READ(5,406) HB(IN),THMB(IN),BETB(IN)
670         WRITE(9,406) HB(IN),THMB(IN),BETB(IN)
680     200 CONTINUE
690     WRITE(9,407)
700     READ(5,6) RE
710     WRITE(9,6) RE
720     WRITE(11,401) UO
730     WRITE(11,401) BL
740     WRITE(11,405) NSTAT
750     WRITE(11,3) S1,S2
760     WRITE(11,405) NTH
770     WRITE(11,401) (TTH(I),I=1,NTH)
780     DO 201 IN=1,NTH
790         WRITE(11,406) HB(IN),THMB(IN),BETB(IN)
800     201 CONTINUE
810     WRITE(11,6) RE
820     GO TO 304
830     303 READ(10,401) UO
840         READ(10,401) BL
850         READ(10,405) NSTAT
860         READ(10,3) S1,S2
870         READ(10,405) NTH
880         READ(10,401) (TTH(I),I=1,NTH)
890         DO 202 IN=1,NTH
900             READ(10,406) HB(IN),THMB(IN),BETB(IN)
910         202 CONTINUE
920         READ(10,6) RE
930     304 WRITE(6,130)
940         WRITE(6,110)
950         WRITE(6,111)N,KPF,BL
960         WRITE(6,113) NTH,NSTAT
970         WRITE(6,12) S1,S2
980         DO 121 J=1,N
990     121 WRITE(6,112)(BN(I,J),I=1,KPF)

```

```

1000      DO 21 I=1,KPK
1010      DO 21 J=1,N
1020      21 BN(I,J)=BL*BN(I,J)
1030      DO 50 I=1,NTH
1040      DTA(I)=0.0
1050      50 CONTINUE
1060      DO 14 I=1,NTH
1070      G=1.535*((HB(I)-0.7)**(-2.715))+3.3
1080      E=- (G+HB(I))*(2./(HB(I)*(HB(I)+1.)*(HB(I)+2.)))
1090      14 THNB(I)=THMB(I)*E*BETB(I)
1100      20 WRITE(6,8)
1110      WRITE(6,1) (TITLE(J),J=1,12)
1120      RN=RE/2.
1130      N1=NSTAT+1
1140      DS=(S2-S1)/NSTAT
1150      ST=S1-DS
1160      CALL GLUG(N,KPK)
1170      DO 7 JJ=1,N1
1180      ST=ST+DS
1190      CALL UGH(ST,N,KPK,FAC,FACP,IENDS)
1200      WRITE(6,27) JJ,ST
1210      CALL AGH(FAC,FACP,N,NTH,JJ,DS,RN,S1)
1220      7 CONTINUE
1230      100 CONTINUE
1240      STOP
1250      500 WRITE(9,413)
1260      STOP
1270      1 FORMAT(12A6)
1280      2 FORMAT(2I10,2F10.5)
1290      3 FORMAT(7F10.5)
1300      5 FORMAT(2I10,2F10.5)
1310      6 FORMAT(F20.1)
1320      8 FORMAT(1H1)
1330      9 FORMAT(7F10.8)
1340      12 FORMAT(5H S1=F10.5,4H S2=F10.5)
1350      27 FORMAT(/712H S(13,2H)=F8.4)
1360      110 FORMAT(/7H INPUT/)
1370      111 FORMAT(4H N=I3,5H KPK=I3,4H BL=F10.5)
1380      112 FORMAT(7(E12.5,5X))
1390      113 FORMAT(6H NTH=I5,7H NSTAT=I5)
1400      130 FORMAT(1H1)
1410      400 FORMAT(20H UG=      1F10.5      )
1420      401 FORMAT(1F10.5)
1430      402 FORMAT(20H BL=      1F10.5      )
1440      403 FORMAT(38H  HB      THMB      BETB  3F10.5

```

```

1450 404 FORMAT(15H NTH= 115 )
1460 405 FORMAT(2I5)
1470 406 FORMAT(3F10.5)
1480 407 FORMAT(24H RE= 1F20.1 )
1490 408 FORMAT(4H RE=)
1500 409 FORMAT(24H NSTAT= 115 )
1510 410 FORMAT(30H S1 52 7F10.5 )
1520 411 FORMAT(25H TTH= 1F10.5 )
1530 412 FORMAT(1I1)
1540 413 FORMAT(25H END OF FILE REACHED )
1550 414 FORMAT(38H NINPT DENOTES THE TYPE OF INPUT )
1560 415 FORMAT(23H NINPT=1 CARD DATA )
1570 416 FORMAT(23H 2 REMOTE )
1580 417 FORMAT(23H 3 STORED DATA )
1590 418 FORMAT(2715H NINPT 111 )
1600 419 FORMAT(43H PROVIDE READ STATEMENTS FOR CARDS AT 301 )
1610 END

```

```

1610 SUBROUTINE AGH(FAC,FACP,N,NTH,JJ,DS,RN,SI)
1630 COMMON /A/TTH(50),HH(50),THMB(50),RETR(50),DIA(50),
1640 RS(50),RTH(50),RD(50),CFE(50),X(50),Y(50)
1650 COMMON /B2/ THNB(50),DTHS(50),DBS(50),DTHS(50),DSR(50)
1660 COMMON /U/AN(10),ANP(10),ANPP(10),EN(10),LNR(10),GN(10),GR(10)
1670 IFAC
1680 COMMON /SAVE/GRU(50)
1690 DIMENSION TP(50),HH2(50),CRK1(50),CRK2(50)
1700 DIMENSION PFS(50),PHI(50),THH(50)
1710 DIMENSION RAD(50)
1720 DIMENSION CFL(50),CFUL(50),CFBL(50),CSL(50),PSIN(50)
1730 DO 1 I=1,NTH
1740 TH=TTH(I)
1750 X=0.0
1760 Y=0.0
1770 SUM1=0.0
1780 SUM2=0.0
1790 SUM3=0.0
1800 SUM4=0.0
1810 SUM5=0.0
1820 SUM6=0.0
1830 SUM7=0.0
1840 SUM8=0.0
1850 SUM9=0.0
1860 SUM10=0.0
1870 SUM11=0.0
1880 SUM12=0.0
1890 SUM13=0.0
1900 SUM14=0.0
1910 SUM15=0.0
1920 SUM17=0.0
1930 SUM18=0.0
1940 SUM19=0.0
1950 SUM20=0.0
1960 SUM21=0.0
1970 DO 23 J=1,N
1980 C1=2+J
1990 C4=3.-C1
2000 CS4=COS(C4+TH)
2010 STN(SIN(C4+TH))
2020 SUM1=SUM1+C4*AN(J)*SS4
2030 SUM2=SUM2+C4*AN(J)*CS4
2040 SUM3=SUM3+ANP(J)*CS4
2050 SUM4=SUM4+ANP(J)*SS4
2060 SUM5=SUM5+(C4+2)*AN(J)*CS4

```

```

2070 SUM6=SUM6-(C4**2)*AN(J)*SS4
2080 SUM7=SUM7-C4*ANF(J)*SS4
2090 SUM8=SUM8+C4*ANF(J)*CS4
2100 SUM19=SUM19+ANFF(J)*CS4
2110 SUM20=SUM20+ANFF(J)*SS4
2120 X=X+AN(J)+CS4
2130 23 Y=Y+AN(J)*SS4
2140 DO 25 J=2,N
2150 C2=2+J-2
2160 SS2=SIN(C2*TH)
2170 CS2=COS(C2*TH)
2180 SUM9=SUM9+CN(J)*SS2
2190 SUM10=SUM10+CN(J)*CS2
2200 SUM11=SUM11+C2*CN(J)*CS2
2210 SUM12=SUM12-C2*CN(J)*SS2
2220 SUM13=SUM13-CNF(J)*CS2/C2
2230 SUM14=SUM14-CN(J)*CS2/C2
2240 52 SUM15=SUM15-CNFF(J)*CS2/C2
2250 SUM21=SUM21+GN(J)*SS2
2260 SUM17=SUM17+CNF(J)*SS2
2270 25 SUM18=SUM18+CNF(J)*CS2
2280 SUM10=CN(1)+SUM10
2290 SUM13 =SUM13+FAC+1.0
2300 SUM31=SUM13*SUM13
2310 SUM14=SUM14+FAC
2320 SUM18=SUM18+CNF(1)
2330 DE=SUM1**2+SUM2**2
2340 131 DE2=DE**2
2350 HE=(SUM5+SUM2-SUM6+SUM1)**2
2360 U=(SUM9*SUM1+SUM10+SUM2)/DE
2370 V=(SUM9*SUM2-SUM10+SUM1)/DE
2380 W=SUM13-U*SUM3-V*SUM4-1.0
2390 DN=SUM1*SUM7+SUM2*SUM8
2400 US=(+SUM18+SUM2+SUM10*SUM6+SUM17*SUM1+SUM9*SUM7)/DE
2410 US=US-2.*DN*U/DE
2420 VS=(-SUM18+SUM1-SUM10*SUM7+SUM17*SUM2+SUM9*SUM8)/DE
2430 VS=VS-2.*DN*V/DE
2440 WS=FACP+SUM15-(US*SUM3+U*SUM19+VS*SUM4+V*SUM20)
2450 DTD5=2.0*SUM21/DE
2460 DM=SUM1*SUM5+SUM2*SUM6
2470 UTH=(SUM10+SUM6+SUM12+SUM2+SUM11+SUM1+SUM9+SUM5)/DE
2480 UTH=UTH-2.*DM*U/DE
2490 VTH=(-SUM12+SUM1-SUM10*SUM5+SUM11+SUM2+SUM9+SUM6)/DE
2500 VTH=VTH-2.*DM*V/DE
2510 WTH=SUM17-U*SUM3-V*SUM7-V*SUM4-V*SUM8

```

```

2520      G1=DE**(-0.5)
2530      HX=SUM3+DTDS*SUM1
2540      HY=SUM4+DTDS*SUM2
2550      PHI(I)=SUM14
2560      RAD(I)=(SUM5*SUM2**2-SUM6*SUM1*SUM2)**2+(SUM6*SUM1**2-SUM5
2570      1*SUM1*SUM2)**2
2580      RAD(I)=SQRT(RAD(I))
2590      RAD(I)=DE2/RAD(I)
2600      USS=U**2+V**2+2.0*W+1.0
2610      F=-USS+1.0
2620      PN=G1*(U*UTH+V*VTH+WTH)
2630      CK2=PN/USS
2640      5.3 PS=U*US+V*VS+W5+DTDS*PN/G1
2650      PPS(I)=PS/USS
2660      135 USS=SQRT(USS)
2670      200 THH(I)=TH
2680      F=-SUM9/SUM13
2690      FS=-SUM17/SUM13+SUM19*(FACP+SUM15)/SUM31
2700      FT=-SUM11/SUM13+SUM9*SUM17/SUM31
2710      GX=F+SUM3+SUM1
2720      GY=F+SUM4+SUM2
2730      GG=GX*GX+GY*GY+F*F
2740      HH=GX*HX+GY*HY+1.0
2750      HH=SQRT(HH)
2760      GXP=F*(SUM19*F+SUM3+FS+SUM7)+F*SUM7+SUM3*FT+SUM5
2770      GYP=F*(SUM20*F+SUM4+FS+SUM8)+F*SUM8+SUM4*FT+SUM5
2780      GZP=F+FS+FT
2790      RG=(GX*GXP+GY*GYP+F*GZP)/GG
2800      CK1=(HX*(GXP-GX+RG)+HY*(GYP-GY+RG)+(GZP-F*RG))/(GG+HH)
2810      XX(I)=X
2820      YY(I)=Y
2830      PP(I)=F
2840      HH2(I)=DTDS
2850      USR(I)=USS
2860      CKK1(I)=CK1
2870      CKK2(I)=CK2
2880      1 CONTINUE
2890      14 WRITE(6,32)
2900      WRITE(6,31) (THH(I),XX(I),Y(I),PP(I),PPS(I),HH2(I),RAD(I),
2910      1CKK1(I),CKK2(I),PHI(I),I=1,NTH)
2920      IF(JJ.EQ.1) GO TO 3
2930      DO 4 I=1,NTH
2940      GRU(I)=(PHI(I)-GRU(I))/USR(I)
2950      THMB(I)=THMB(I)+1.*GRU(I)*DTH5(I)
2960      G=1.535*((HB(I)-0.7)**(-2.715))+3.3

```

```

2970      G=G+GRU(I)*DGS(I)
2980      THNB(I)=THNB(I)+GRU(I)*DTNS(I)
2990      IF(G.LE.3.3) GO TO 501
3000      HB(I)=(1.535/(G-3.3))*0.3646+0.7
3010      H=HB(I)
3020      E=-(G+H)*(2./(H*(H+1.)*(H+2.)))
3030      BETB(I)=1.*THNB(I)/(THMB(I)*E)
3040      4 CONTINUE
3050      3 DO 2 I=1,NTH
3060          H=HB(I)
3070          THM=THMB(I)
3080          BET=BETB(I)
3090          USO=USR(I)
3100          P50=PPS(I)
3110          CK1=CKK1(I)
3120          CK2=CKK2(I)
3130          CALL BLAY(RN,USO,P50,CK1,CK2,H,THM,BET,CF,DTH,DG,DTN,THN)
3140          RTH(I)=RN*USO*THM
3150          RD(I)=RTH(I)*H
3160          DTA(I)=H*THMB(I)
3170          CFL(I)=2000.*CF
3180          CF=CFL(I)
3190          CALL FCF(SUM1,SUM2,SUM3,SUM4,DTOS,BET,CF,CFWW,CFBB,CS3)
3200          CFWL(I)=CFWW
3210          CFBL(I)=CFBB
3220          CSL(I)=CS3
3230          GRU(I)=FHI(I)
3240          TTH(I)=TTH(I)+DG*HH2(I)
3250          DTHS(I)=DTH
3260          DGS(I)=DG
3270          DTN(I)=DTN
3280      2 CONTINUE
3290      WRITE(6,40) JJ,SI
3300      WRITE(6,34)
3310      WRITE(6,35) (HB(I),THMB(I),DTA(I),BETB(I),CFL(I),RD(I),RTH(I),
3320      1CFWL(I),CFBL(I),CSL(I),I=1,NTH)
3330      RETURN
3340      501 WRITE(9,406)
3350      WRITE(9,407) G
3355      WRITE(9,405) JJ,SI
3360      STOP
3370      31 FORMAT(6I12.5,E12.5,3F12.5)
3380      32 FORMAT(//120H          TH          X          Y          CP
3390      1 PSI          DTDS          R          K1          K2          FHI
3400      1 //)

```

```
3410 34 FORMAT(/119H      H      TH      DE      BETA
3420      1 CF      RS      RTH      CFW      CFB      C:
3430      2/)
3440 35 FORMAT(5F12.5,1H ,E11.4,1H ,E11.4,1H .3F12.5)
3450 40 FORMAT(//25H      BOUNDARY LAYER AT S(I3,2H)=F8.4)
3455 405 FORMAT(19H FAILED AT STA NO =,1I4,3H S=,1F8.5)
3460 406 FORMAT(46H G IS LESS THAN 3.3 CANNOT CONTINUE IN AGH
3470 407 FORMAT(3H G=,1E15.5)
3480      END
#
```



```

3490      SUBROUTINE BLAY(RN,USO,PSO,CKO1,CKO2,H,THM,BET,CF,DTH,DG,DTN,THN)
3500      G=1.535*((H-0.7)**(-2.715))+3.3
3510      F=0.0306*((G-3.0)**(-0.653))
3520      THN2=THN**2
3530      RT=RN*USO*THM
3540      C=ALOG(RT)
3550      C2=C*C
3560      C3=C2*C
3570      A=0.019521-0.386768*C+0.028345*C2-0.000701*C3
3580      B=0.191511-0.834891*C+0.062588*C2-0.001953*C3
3590      CF=0.5*EXP(A*H+B)
3600      20 DTH=CF+THM*(CKO1-(2.+H)*PSO)
3610      DG=F/THM+G*(CKO1-PSO-DTH/THM)
3620      DTN=BET*CF+2.*THN*(CKO1-PSO)-THM*(1.+H)*CKO2
3630      10 RETURN
3640      END

```

```

3650      SUBROUTINE FCF(SUM1,SUM2,SUM3,SUM4,DTDS,BET,CF,CFW,CFB,C3)
3660      X2=SUM1*SUM1+SUM2*SUM2
3670      Y=SUM1*SUM4-SUM2*SUM3
3680      Y2=Y*Y
3690      D=SQRT(X2+Y2)
3700      XN1=SUM2/D
3710      XN2=-SUM1/D
3720      XN3=Y/D
3730      S1=SUM3+SUM1*DTDS
3740
3750
3760      S2=SUM4+SUM2*DTDS
3770      SD=SQRT(S1*S1+S2*S2+1.)
3780      S1=S1/SD
3790      S2=S2/SD
3800      S3=1./SD
3810      B1=S2*XN3-S3*XN2
3820      B2=S3*XN1-S1*XN3
3830      B3=S1*XN2-S2*XN1
3840      C31=CF*(S1+BET*B1)
3850      C32=CF*(S2+BET*B2)
3860      C33=CF*(S3+BET*B3)
3870      IF(SUM2) 1,2,1
3880      2 W1=0.
3890      W3=1.0
3900      GO TO 3
3910      1 W1=SUM3- SUM1*SUM4/SUM2
3920      WD=SQRT(W1*W1+1.)
3930      W1=W1/WD
3940      W3=1./WD
3950      3 IF(SUM1) 4,5,4
3960      5 D2=0.
3970      D3=1.0
3980      GO TO 6
3990      4 D2=SUM4-SUM2*SUM3-SUM1
4000      WD=SQRT(D2*D2+1.)
4010      D2=D2/WD
4020      D3=1./WD
4030      6 CFW=W1*C31+W3*C33
4040      CFB=D2*C32+D3*C33
4050      RETURN
4060      END

```

```

4070     SUBROUTINE GLUG(N,M)
4080     COMMON /G/BN(25,10),WK(50),DK(50),TM(50),TMO
4090     DIMENSION AMK(10,50)
4100     M2=2*M-1
4110     M3=M2-3
4120     DO 1 I=1,N
4130     DO 2 J=2,M2
4140     SUM=0.0
4150     IF(J-M) 3,3,4
4160     3 DO 5 L=1,J
4170     C=L-1
4180     L1=J+1-L
4190     5 SUM=SUM+C*(BN(L1,I)*BN(L,I))
4200     GO TO 7
4210     4 J1=J+1-M
4220     DO 6 L=J1,M
4230     C=L-1
4240     L1=J+1-L
4250     6 SUM=SUM+C*(BN(L1,I)*BN(L,I))
4260     7 AMK(I,J)=SUM
4270     WK(J)=0.0
4280     2 CONTINUE
4290     1 CONTINUE
4300     DO 29 J=3,M2
4310     J2=J-2
4320     SUM=0.0
4330     DO 30 L=1,J2
4340     C=L
4350     C=1./C
4360     30 SUM=SUM+C
4370     29 WK(J)=SUM
4380     DO 8 J=2,M2
4390     SUM=0.0
4400     DO 9 I=1,N
4410     C=3-2*I
4420     9 SUM=SUM+C*AMK(I,J)
4430     8 DK(J)=SUM
4440     DO 27 LM=1,M3
4450     SUM=0.0
4460     LMM=LM+3
4470     UN=1.
4480     DO 28 LK=LMM,M2
4490     KM=LK-LM-2
4500     XMK=KM
4510     UN=-1.*UN

```

```
4520      28 SUM=SUM+DK(LK)*(1.+UN)/XMK
4530      27 TM(LM)=SUM
4540          TMO=0.0
4550          DO 10 J=3,M2
4560          AJ=J-2
4570          U=(-1.)**(J-2)
4580      10 TMO=TMO+DK(J)*(1.+U)/AJ
4590          RETLN=
4600          END
#
```

```

4610     SUBROUTINE UGH(ST,N,M,FAC,FACP,IENDS)
4620     COMMON /G/BN(25,10),WK(50),DK(50),Im(50),TMO
4630     COMMON /U/AN(10),ANP(10),ANPP(10),CN(10),CNP(10),GN(10),CNPP(10).
4640     IFAC
4650     DIMENSION ANPPP(10),SP(50)
4660     M2=2*M-1
4670     M3=M2-3
4680     SP(1)=1.0
4690     DO 11 L=2,M2
4700 11 SP(L)=SP(L-1)*ST
4710     DO 12 J=1,N
4720     AN(J)=0.0
4730     ANP(J)=0.0
4740     ANPP(J)=0.0
4750     ANPPP(J)=0.0
4760     DO 13 L=1,M
4770 13 AN(J)=AN(J)+BN(L,J)*SP(L)
4780     DO 14 L=2,M
4790     C=L-1
4800 14 ANP(J)=ANP(J)+BN(L,J)*SP(L-1)*C
4810     DO 15 L=3,M
4820     C=(L-2)*(L-1)
4830 15 ANPP(J)=ANPP(J)+BN(L,J)*SP(L-2)*C
4840     DO 22 L=4, M
4850     C=(L-3)*(L-2)*(L-1)
4860 22 ANPPP(J)=ANPPP(J)+C*BN(L,J)*SP(L-3)
4870 12 CONTINUE
4880     GN(1)=0.0
4890     CN(1)=0.0
4900     CNP(1)=0.0
4910     CNPP(1)=0.0
4920     DO 16 L=1,N
4930     C=3-2*L
4940     CN(L)=CN(L)+ANP(L)*AN(L)*C
4950     CNPP(L)=CNPP(L)+3.0*C*ANP(L)*ANPP(L)+C*AN(L)*ANPPP(L)
4960 16 CNP(L)=CNP(L)+(ANPP(L)*AN(L)+ANP(L)**2)*C
4970 27 DO 17 J=2,N
4980     CN(J)=0.0
4990     CNP(J)=0.0
5000     CNPP(J)=0.0
5010     LU=N-J+1
5020     DO 18 L=1,LU
5030     C=3-2*L
5040     LJ=J+L-1
5050     CN(J)=CN(J)+ANP(LJ)*AN(L)*C

```

```

5060      CNPP(J)=CNPP(J)+2.0*C*ANP(L)*ANPP(LJ)+C*ANPP(L)*ANP(LJ)
5070      1+C*AN(L)*ANPPP(LJ)
5080      18 CNP(J)=CNP(J)+(ANPP(LJ)*AN(L)+ANP(LJ)*ANP(L))+C
5090      GN(J)=CN(J)
5100      DO 19 L=J,N
5110      C=3-2*L
5120      LJ=L+1-J
5130      CN(J)=CN(J)+ANP(LJ)*AN(L)*C
5140      CNPP(J)=CNPP(J)+2.0*C*ANP(L)*ANPP(LJ)
5150      1+C*ANPP(L)*ANP(LJ)+C*AN(L)*ANPPP(LJ)
5160      19 CNP(J)=CNP(J)+(ANPP(LJ)*AN(L)+ANP(LJ)*ANP(L))*C
5170      17 CONTINUE
5180      SUM1=0.0
5190      SUM2=0.0
5200      DO 20 L=3,M2
5210      CL=L-2
5220      20 SUM1=SUM1+CL*DK(L)*WK(L)*SP(L-2)
5230      DO 21 L=1,M3
5240      CL=L
5250      21 SUM2=SUM2+CL*TM(L)*SP(L)
5260      SUM3=0.0
5270      DO 23 L=4,M2
5280      CL=(L-2)*(L-3)
5290      23 SUM3=SUM3+CL*DK(L)*WK(L)*SP(L-3)
5300      SUM4=0.0
5310      DO 24 L=2,M3
5320      CL=L*(L-1)
5330      24 SUM4=SUM4+CL*TM(L)*SP(L-1)
5340      PS=4.-4.*SP(3)
5350      BOP=-0.5+CNP(1)*ALOG(PS)+4.*SP(2)*CN(1)/PS+SUM1-0.5*SUM2
5360      FAC=BOP+CNP(1)*ALOG(AN(1))+CN(1)*ANP(1)/AN(1)
5370      BOPP=-0.5*CNPP(1)*ALOG(PS)+4.*(CN(1)+2.*SP(2)*CN(1))/PS
5380      1+30.*SP(3)*CN(1)/(PS+PS)+SUM3-0.5*SUM4
5390      41 FACB=BOPP+CNPP(1)*ALOG(AN(1))+(2.0+CNP(1)*ANP(1)+CN(1)*ANP(1)
5400      1/AN(1)-CN(1)*(ANP(1)/AN(1)))*2
5410      BO=0.0
5420      DO 25 J=3,M2
5430      25 BO=BO+WK(J)*DK(J)*SP(J-1)
5440      DO 26 J=1,M3
5450      26 BO=BO-0.5*TM(J)*SP(J+1)
5460      PO=BO-0.5*TM0
5470      FAC=LN(1)*ALOG(AN(1))+BO+ST-0.5*LN(1)*ALOG(PS)
5480      RETURN
5490      END

```

```

100 FILE 5(KIND=REMOTE,MAXRECSIZE=22)
110 FILE 6(KIND=PRINTER,MAXRECSIZE=132)
120 FILE 7(TITLE="TBL3DOPT",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
130 FILE 8(TITLE="BNSAL",KIND=DISK,FILETYPE=7)
140 FILE 9(KIND=REMOTE,MAXRECSIZE=22)
150 FILE 10(TITLE="TBL3DIMP",KIND=DISK,FILETYPE=7)
160 FILE 11(TITLE="TBLSAVE",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
170 $RESET FREE
180 C PROGRAM TURBULENT BOUNDARY LAYER COORDINATE SYSTEM. BODY
190 C GEOMETRY, AND POTENTIAL
200 COMMON /A2/ BN(25,10)
210 COMMON /A3/ WK(50),DK(50),TM(50),LN(10),CRP(10),GN(10)
220 COMMON /A4/ AN(10),ANP(10),ANPP(10)
230 COMMON /A6/ TTH(50)
240 COMMON /B5/ XA(100,50),YA(100,50),SV(100,50),CF(100,50)
250 COMMON / / TITLE(12),TITL(3)
260 DO 21 I=1,25
270 DO 21 J=1,10
280 21 BN(I,J)=0.0
290 DO 22 I=1,50
300 WK(I)=0.0
310 DK(I)=0.0
320 22 TM(I)=0.0
330 READ(8,1) (TITLE(J),J=1,12)
340 WRITE(9,1) (TITLE(J),J=1,12)
350 WRITE(7,1) (TITLE(J),J=1,12)
360 READ(8,405)N,KPK,BL
370 WRITE(9,405)N,KPK,BL
380 DO 13 J=1,N
390 READ(8,3) (BN(I,J),I=1,KPK)
400 WRITE(9,3) (BN(I,J),I=1,KPK)
410 13 CONTINUE
420 WRITE(9,414)
430 WRITE(9,415)
440 WRITE(9,416)
450 WRITE(9,417)
460 WRITE(9,418)
470 READ(5,412) NINPT
480 GOTO (301,302,303) NINPT
490 301 WRITE(9,419)
500 STOP
510 302 WRITE(9,400)
520 WRITE(9,420)
530 READ(5,412) LBL
540 IF(LBL.NE.1) GOTO 210

```

```

550      WRITE(9,421)
560      READ(5,401)BL
570      WRITE(9,401)BL
580 210  WRITE(9,422)
590      READ (5,401)START
600      WRITE(9,401)START
610      WRITE(9,423)
620      READ (5,401)SFINIS
630      WRITE(9,401)SFINIS
640      WRITE(9,424)
650      READ (5,405)NSTAT
660      WRITE(9,405)NSTAT
670      WRITE(9,425)
680      READ (5,405)NTH
690      WRITE(9,405)NTH
700      WRITE(11,401) BL
710      WRITE(11,401) START
720      WRITE(11,401) SFINIS
730      WRITE(11,405) NSTAT
740      WRITE(11,405) NTH
750      GO TO 304
760 303  READ (10,401)BL
770      READ (10,401) START
780      READ (10,401) SFINIS
790      READ (10,405) NSTAT
800      READ (10,405) NTH
810 304  CONTINUE
820      WRITE(7,2) N,KPK,BL
830      WRITE(7,5) START,SFINIS,NSTAT,NTH
840      TTH(1)=0.0
850      PI02=-1.57079632679
860      EN=NTH-1
870      DT=PI02/EN
880      DO 200 I=2,NTH
890 200  TTH(I)=TTH(I-1)+DT
900      WRITE(9,3)(TTH(I),I=1,NTH)
910 20  WRITE(6,8)
920      WRITE(6,1) (TITLE(J),J=1,12)
930      DO 4 I=1,KPK
940      DO 4 J=1,N
950 4  BN(I,J)=BL+BN(I,J)
960      EN=NSTAT-1
970      DS=(SFINIS-START)/EN
980      DSD=DS
990      LD=-START/DS+1.

```



```

1000      DO 23 I=1,10
1010      AN(I)=0.0
1020      ANP(I)=0.0
1030      ANPP(I)=0.0
1040      CN(I)=0.0
1050      CNP(I)=0.0
1060      23 GN(I)=0.0
1070      CALL GLUG(N,KPK)
1080      ST=START-DS
1090      DO 7 JJ=1,NSTAT
1100      ID=0
1110      INDEX=0
1120      6 ST=ST+DS
1130      CALL AGH(FAL,N,NTH,JJ,DS,ST,ID,INDEX,KPK)
1140      DS=DSO
1150      500 CONTINUE
1160      7 CONTINUE
1170      1 FORMAT(12A6)
1180      2 FORMAT(2I10,F10.5)
1190      3 FORMAT(7F10.5)
1200      5 FORMAT(2F10.5,2I10)
1210      8 FORMAT('H1)
1220      70 FORMAT(3A10)
1230      400 FORMAT(30H DO YOU WISH TO ENTER NEW BLE      )
1240      401 FORMAT(1F10.5)
1250      402 FORMAT(20H BL=      1F10.5      )
1260      403 FORMAT(38H      HB      THMB      BETB  3F10.5      )
1270      404 FORMAT(15H NTH=      115      )
1280      405 FORMAT(215,1F10.5)
1290      406 FORMAT(3F10.5)
1300      407 FORMAT(24H RE=      1F20.1      )
1310      408 FORMAT(4H RE=)
1320      409 FORMAT(24H NSTAT=      115      )
1330      410 FORMAT(30H      S1      S2      7F10.5      )
1340      411 FORMAT(25H TTH=      1F10.5      )
1350      412 FORMAT(111)
1360      413 FORMAT(25H END OF FILE REACHED      )
1370      414 FORMAT(38H NINPT DENOTES THE TYPE OF INPUT      )
1380      415 FORMAT(23H NINPT=1  CARD DATA      )
1390      416 FORMAT(23H      2  REMOTE      )
1400      417 FORMAT(23H      3  STORED DATA      )
1410      418 FORMAT(215H NINPT      111      )
1420      419 FORMAT(43H PROVIDE HEAD STATEMENTS FOR CARDS AT 301      )
1430      420 FORMAT(27H IF YES TYPE 1 NO TYPE 0      )
1440      421 FORMAT(15H BL=      1F10.5      )

```

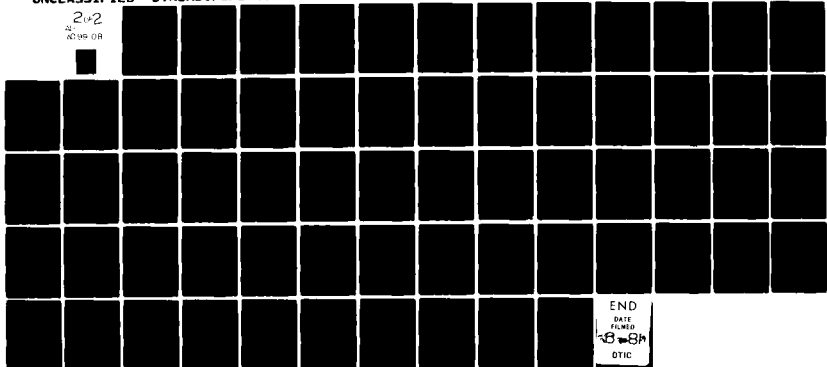
AD-A099 108

DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CE--ETC F/6 20/4
A USER'S MANUAL FOR THE DTNSRDC MOMENTUM INTEGRAL BOUNDARY-LAYER--ETC(U)
DEC 80 T J LANGAN, C VON KERCEK
DTNSRDC/SPD-0972-01

NL

UNCLASSIFIED

242
NOV 88



END
DATE
FILMED
8-88
DTIC

```
1450 422 FORMAT(20H START= 1F10.5 )
1460 423 FORMAT(20H SFINIS= 1F10.5 )
1470 424 FORMAT(20H NSTAT= 1I3 )
1480 425 FORMAT(20H NTH= 1I3 )
1490 END
#
```

```

1500     SUBROUTINE GLUG(N,M)
1510     COMMON /A2/ BN(25,10)
1520     COMMON /A3/ WK(50),DK(50),TM(50),CN(10),CNP(10),GN(10)
1530     DIMENSION AMK(10,50)
1540     M2=2*M-1
1550     M3=2*M-4
1560     DO 31 I=1,10
1570     DO 31 J=1,50
1580 31 AMK(I,J)=0.0
1590     DO 1 I=1,N
1600     DO 2 J=2,M2
1610     SUM=0.0
1620     IF(J-M) 3,3,4
1630 3 DO 5 L=1,J
1640     C=L-1
1650     L1=J+1-L
1660 5 SUM=SUM+C*(BN(L1,I)*BN(L,I))
1670     GO TO 7
1680 4 J1=J+1-M
1690     DO 6 L=J1,M
1700     C=L-1
1710     L1=J+1-L
1720 6 SUM=SUM+C*(BN(L1,I)*BN(L,I))
1730 7 AMK(I,J)=SUM
1740     WK(J)=0.0
1750 2 CONTINUE
1760 1 CONTINUE
1770     DO 29 J=3,M2
1780     J2=J-2
1790     SUM=0.0
1800     DO 30 L=1,J2
1810     C=L
1820     C=1./C
1830 30 SUM=SUM+C
1840 29 WK(J)=SUM
1850     DO 8 J=2,M2
1860     SUM=0.0
1870     DO 9 I=1,N
1880     C=3-2*I
1890 9 SUM=SUM+C*AMK(I,J)
1900 8 DK(J)=SUM
1910     DO 27 LM=1,M3
1920     SUM=0.0
1930     LMM=LM+3
1940     UN=1.

```

```
1950      DO 28 LK=LMM,M2
1960      KM=LK-LM-2
1970      XMK=KM
1980      UN=-1.*UN
1990      28 SUM=SUM+DK(LK)*(1.+UN)/XMK
2000      27 TH(LM)=SUM
2010      RETURN
2020      END
#
```

```

2030     SUBROUTINE POT(ST,N,M,FAC)
2040     COMMON /A4/ AN(10),ANP(10),ANPP(10)
2050     COMMON /A3/ WK(50),DK(50),TM(50),CN(10),CNP(10),GN(10)
2060     COMMON /C1/ SP(50)
2070     M2=2*M-1
2080     M3=M2-3
2090     GN(1)=0.0
2100     CN(1)=0.0
2110     CNP(1)=0.0
2120     DO 16 L=1,N
2130     C=3-2*L
2140     CN(L)=CN(L)+ANP(L)*AN(L)*C
2150 16  CNP(L)=CNP(L)+(ANPP(L)*AN(L)+ANP(L)**2)*C
2160     DO 17 J=2,N
2170     CN(J)=0.0
2180     CNP(J)=0.0
2190     LU=N-J+1
2200     DO 18 L=1,LU
2210     C=3-2*L
2220     LJ=J+L-1
2230     CN(J)=CN(J)+ANP(LJ)*AN(L)*C
2240 18  CNP(J)=CNP(J)+(ANPP(LJ)*AN(L)+ANP(LJ)*ANP(L))*C
2250     GN(J)=CN(J)
2260     DO 19 L=J,N
2270     C=3-2*L
2280     LJ=L+1-J
2290     CN(J)=CN(J)+ANP(LJ)*AN(L)*C
2300 19  CNP(J)=CNP(J)+(ANPP(LJ)*AN(L)+ANP(LJ)*ANP(L))*C
2310 17  CONTINUE
2320     SUM1=0.0
2330     SUM2=0.0
2340     DO 20 L=3,M2
2350     CL=L-2
2360 20  SUM1=SUM1+CL*DK(L)*WK(L)*SP(L-2)
2370     DO 21 L=1,M3
2380     CL=L
2390 21  SUM2=SUM2+CL*TM(L)*SP(L)
2400     PS=4.-4.*SP(3)
2410     BOP=4.*SP(2)+CN(1)/PS
2420     BOP=BOP-0.5*CNP(1)*ALOG(PS)+SUM1-0.5*SUM2
2430     FAC=BOP+CNP(1)*ALOG(AN(1))+CN(1)*ANP(1)/AN(1)
2440     RETURN
2450     END
#

```

```

2460      SUBROUTINE BODY(ST,N,M)
2470      COMMON /A2/ BN(25,10)
2480      COMMON /A4/ AN(10),ANP(10),ANPP(10)
2490      COMMON /C1/ SP(50)
2500      M2=2*M-1
2510      M3=2*M-4
2520      M4=M2+1
2530      SP(1)=1.0
2540      DO 11 L=2,M2
2550      11 SP(L)=SP(L-1)*ST
2560      DO 22 L=M4,50
2570      22 SP(L)=0.0
2580      DO 12 J=1,N
2590      AN(J)=0.0
2600      ANP(J)=0.0
2610      ANPP(J)=0.0
2620      DO 13 L=1,M
2630      13 AN(J)=AN(J)+BN(L,J)*SP(L)
2640      DO 14 L=2,M
2650      C=L-1
2660      14 ANP(J)=ANP(J)+BN(L,J)*SP(L-1)*C
2670      DO 15 L=3,M
2680      C=(L-2)*(L-1)
2690      15 ANPP(J)=ANPP(J)+BN(L,J)*SP(L-2)*C
2700      12 CONTINUE
2710      RETURN
2720      END
#

```

```

2730     SUBROUTINE BDER(DSDT,S,N,KPK,TH)
2740     COMMON /A4/ AN(10),ANP(10),ANPP(10)
2750     CALL BODY(S,N,KPK)
2760     XS=0.0
2770     YS=0.0
2780     XT=0.0
2790     YT=0.0
2800     DO 1 J=1,N
2810     C=3-2*J
2820     CC=COS(C*TH)
2830     SS=SIN(C*TH)
2840     XS=XS+CC*ANP(J)
2850     YS=YS+SS*ANP(J)
2860     XT=XT-SS*C*AN(J)
2870     1 YT=YT+CC*C*AN(J)
2880     DSDT=- (XS*XT+YS*YT)/(XS*XS+YS*YS+1.)
2890     RETURN
2900     END

```



```

2910      SUBROUTINE AGH(FAC,N,NTH,JJ,DS,ST,ID,INDEX,KPK)
2920      COMMON /A3/ WK(50),DK(50),TH(50),CN(10),CNP(10),GN(10)
2930      COMMON /A4/ AN(10),ANP(10),ANPP(10)
2940      COMMON /A6/ TTH(50)
2950      COMMON /B5/ XA(100,50),YA(100,50),SV(100,50),CP(100,50)
2960      DIMENSION SS(50)
2970      WRITE(6,27) JJ,ST
2980      WRITE(7,427) JJ,ST
2990      SS(1)=ST
3000      DO 1 I=1,NTH
3010      IF(I.GT.1) GO TO 2
3020      TH=TTH(I)
3030      S=SS(I)
3040      CALL BDER(DSDT,S,N,KPK,TH)
3050      GO TO 3
3060      2 TH=TTH(I-1)
3070      S=SS(I-1)
3080      CALL BDER(DSDT,S,N,KPK,TH)
3090      S1=SS(I-1)+(TTH(I)-TTH(I-1))*DSDT
3100      TH=TTH(I)
3110      CALL BDER(DSDTC,S1,N,KPK,TH)
3120      SS(I)=SS(I-1)+0.5*(TTH(I)-TTH(I-1))*(DSDTC+DSDT)
3130      S=SS(I)
3140      3 CALL POT(S,N,KPK,FAC)
3150      X=0.0
3160      Y=0.0
3170      SUM1=0.0
3180      SUM2=0.0
3190      SUM3=0.0
3200      SUM4=0.0
3210      SUM5=0.0
3220      SUM6=0.0
3230      SUM7=0.0
3240      SUM8=0.0
3250      SUM9=0.0
3260      SUM10=0.0
3270      SUM11=0.0
3280      SUM12=0.0
3290      SUM13=0.0
3300      SUM14=0.0
3310      SUM15=0.0
3320      SUM16=0.0
3330      SUM17=0.0
3340      SUM18=0.0
3350      SUM19=0.0

```

```

3360      SUM20=0.0
3370      SUM21=0.0
3380      DO 23 J=1,N
3390      C1=2*J
3400      C2=1.-C1
3410      C3=2.-C1
3420      C4=3.-C1
3430      CS1=COS(C1*TH)
3440      CS2=COS(C2*TH)
3450      CS3=COS(C3*TH)
3460      CS4=COS(C4*TH)
3470      SS2=SIN(C2*TH)
3480      SS3=SIN(C3*TH)
3490      SS4=SIN(C4*TH)
3500      SUM1=SUM1+CNP(J+1)*CS1/C1
3510      SUM2=SUM2+CN(J)*CS2
3520      SUM3=SUM3+CN(J)*SS2
3530      SUM4=SUM4+ANP(J)*CS4
3540      SUM5=SUM5+ANP(J)*SS4
3550      SUM6=SUM6+AN(J)*CS3*C4
3560      SUM7=SUM7+AN(J)*SS3*C4
3570      SUM8=SUM8+ANPP(J)*CS4
3580      SUM9=SUM9+ANPP(J)*SS4
3590      SUM10=SUM10-AN(J)*C4*SS4
3600      SUM11=SUM11+AN(J)*C4*CS4
3610      SUM12=SUM12-ANP(J)*C4*SS4
3620      SUM13=SUM13+ANP(J)*C4*CS4
3630      SUM14=SUM14+CNP(J)*CS2
3640      SUM15=SUM14+CNP(J)*SS2
3650      SUM16=SUM16+C2*CN(J)*CS2
3660      SUM17=SUM17-C2*CN(J)*SS2
3670      SUM18=SUM18+C4*ANP(J)*CS3
3680      SUM19=SUM19+C4*ANP(J)*SS3
3690      SUM20=SUM20-C4*C3*AN(J)*SS3
3700      SUM21=SUM21+C4*C3*AN(J)*CS3
3710      X=X+AN(J)*CS4
3720      23 Y=Y+AN(J)*SS4
3730      DE=SUM6**2+SUM7**2
3740      DEI=1.0/DE
3750      U=(SUM2*SUM6+SUM3*SUM7)/DE
3760      V=(SUM2*SUM7-SUM3*SUM6)/DE
3770      W=-(U*SUM4+V*SUM5)-SUM1+FAC
3780      P=-2.*W-U**2-V**2
3790      XYSS=SUM6*SUM18+SUM7*SUM19
3800      XYST=SUM6*SUM20+SUM7*SUM21

```

```

3810      US=(SUM14*SUM6+SUM2*SUM18+SUM15*SUM7+SUM3*SUM19)
3820      US=(US-2.0*U*XYSS)*DEI
3830      UT=(SUM17*SUM6+SUM2*SUM20+SUM16*SUM7+SUM3*SUM21)
3840      UT=(UT-2.0*U*XYST)*DEI
3850      VS=SUM14*SUM7+SUM2*SUM19-SUM15*SUM6-SUM3*SUM18
3860      VS=(VS-2.0*V*XYSS)*DEI
3870      VT=SUM17*SUM7+SUM2*SUM21-SUM16*SUM6-SUM3*SUM20
3880      VT=(VT-2.0*V*XYST)*DEI
3890      H=SUM4*SUM4+SUM5*SUM5+1.0
3900      H2=SQR(H)
3910      H2I=1.0/H2
3920      H3=H*H2
3930      F=-(SUM10*SUM4+SUM11*SUM5)/H
3940      G=(SUM10+SUM4*F)**2+(SUM11+SUM5*F)**2+F*F
3950      G=SQR(G)
3960      GI=1.0/G
3970      DUDLP=H2I*US
3980      DVDLP=H2I*VS
3990      DUDLP=GI*(F*US+UT)
4000      DVBLT=GI*(F*VS+VT)
4010      D=SUM10*SUM10+SUM11*SUM11+(SUM10*SUM5-SUM11*SUM4)**2
4020      D=SQR(D)
4030      EN1=SUM11/D
4040      EN2=-SUM10/D
4050      EN3=(SUM10*SUM5-SUM11*SUM4)/D
4060      ET1=(SUM5*EN3-EN2)/H2
4070      ET2=(EN1-SUM4*EN3)/H2
4080      ET3=(SUM4*EN2-SUM5*EN1)/H2
4090      EP1=SUM4*H2I
4100      EP2=SUM5*H2I
4110      EP3=H2I
4120 C    COMPUTATION OF XKT THE GEODESIC CURVATURE OF THE PHI COORDINATE
4130      HD=(SUM4*SUM8+SUM5*SUM9)/H3
4140      EP1S= SUM8*H2I-SUM4*HD
4150      EP2S=SUM9*H2I-SUM5*HD
4160      EP3S = -HD
4170      XKT = (ET1*EP1S+ET2*EP2S+ET3*EP3S)*H2I
4180 C    COMPUTATION OF XKP GEODESIC CURVATURE OF THATA COORDINATE LINES
4190      HD=(SUM4*SUM12+SUM5*SUM13)/H3
4200      EP1T = SUM12*H2I-SUM4*HD
4210      EP2T =SUM13*H2I-SUM5*HD
4220      EP3T = -HD
4230      GI=1.0/G
4240      P1LT= GI*(F*EP1S+EP1T)
4250      P2LT= GI*(F*EP2S+EP2T      )

```

```

4260      P3LT =GI*(F*EP3S+EP3T)
4270      E1=P2LT*EN3-P3LT*EN2
4280      E2=P3LT*EN1-P1LT*EN3
4290      E3=P1LT*EN2-P2LT*EN1
4300      XKP=(SUM4*E1+SUM5*E2+E3)*H2I
4310 C    COMPUTATION OF UP,UT, AND UN VELOCITIES IN ORTHOGONAL COORDINATES.
4320      W=1.0  +W
4330      UN=EN1*U+EN2*V+EN3*W
4340      UT=ET1*U+ET2*V+ET3*W
4350      UP=EP1*U+EP2*V+EP3*W
4360      H=H2
4370      WRITE(6,408) 1,X,Y,TH,U,V
4380      WRITE(7,408) 1,X,Y,TH,U,V
4390      WRITE(6,408) 2,W,UP,UT,UN,SS(I)
4400      WRITE(7,408) 2,W,UP,UT,UN,SS(I)
4410      WRITE(6,408)3,XKP,XKT,H,F,G
4420      WRITE(7,408)3,XKP,XKT,H,F,G
4430      1 CONTINUE
4440      500 RETURN
4450      27 FORMAT(//7H      S(I2,2H)=F9.6)
4460      400 FORMAT(8E15.7)
4470      408 FORMAT(1I3,5E15.7)
4480      427 FORMAT(1I3,1F9.6)
4490      END

```

```

#FILE (CHXL)CBLGEO ON DTNSRDC
100 FILE 5(KIND=REMOTE,MAXRECSIZE=22)
120 FILE 7(TITLE="CBL3DOPT",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
130 FILE 8(TITLE="BNSAL",KIND=DISK,FILETYPE=7)
140 FILE 9(KIND=REMOTE,MAXRECSIZE=22)
150 FILE 10(TITLE="CBL3DINP",KIND=DISK,FILETYPE=7)
160 FILE 11(TITLE="CBLTAPE",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
170 $RESET FREE
180 C PROGRAM CBL
190 COMMON /A2/ BN(25,10)
200 COMMON /A3/ WK(50),DK(50),TM(50),CN(10),CNP(10),GN(10)
210 COMMON /A4/ AN(10),ANP(10),ANPP(10)
220 COMMON /A6/ TTH(50)
230 COMMON /B5/ XA(100,50),YA(100,50),SV(100,50),CP(100,50)
240 COMMON / / TITLE(12),TITL(3)
250 DIMENSION S(50),X(50,50),Y(50,50)
260 DIMENSION SE(500),XE(500),YE(500),THE(500)
270 DO 21 I=1,25
280 DO 21 J=1,10
290 21 BN(I,J)=0.0
300 DO 22 I=1,50
310 WK(I)=0.0
320 DK(I)=0.0
330 22 TM(I)=0.0
340 READ(8,1) (TITLE(J),J=1,12)
350 WRITE(9,1) (TITLE(J),J=1,12)
360 WRITE(7,1) (TITLE(J),J=1,12)
370 READ(8,405)N,KPK,BL
380 WRITE(9,405)N,KPK,BL
390 DO 13 J=1,N
400 READ(8,3) (BN(I,J),I=1,KPK)
410 WRITE(9,3) (BN(I,J),I=1,KPK)
420 13 CONTINUE
430 WRITE(9,414)
440 WRITE(9,415)
450 WRITE(9,416)
460 WRITE(9,417)
470 WRITE(9,418)
480 READ(5,412) NINPT
490 GOTO (301,302,303) NINPT
500 301 WRITE(9,419)
510 STOP
520 302 WRITE(9,400)
530 WRITE(9,420)
540 READ(5,412) LBL
550 IF(LBL.NE.1) GOTO 210
560 WRITE(9,421)
570 READ(5,401)BL

```

```

580     WRITE(9,401)BL
590 210 START=-1.000
600     SFINIS=1.000
650     WRITE(9,424)
660     READ (5,429)NSTAT
670     WRITE(9,405)NSTAT
680     WRITE(9,425)
690     READ (5,429)NTH
700     WRITE(9,405)NTH
710     WRITE(11,401) BL
720     WRITE(11,401) START
730     WRITE(11,401) SFINIS
740     WRITE(11,405) NSTAT
750     WRITE(11,405) NTH
760     GO TO 304
770 303 READ (10,401)BL
780     READ (10,401) START
790     READ (10,401) SFINIS
800     READ (10,405) NSTAT
810     READ (10,405) NTH
820 304 CONTINUE
830     WRITE(7,2) N,KPK,BL
840     WRITE(7,5) START,SFINIS,NSTAT,NTH
850     TTH(1)=0.0
860     PI02= -1.57079632679
870     EN=NTH-1
880     DT=PI02/EN
890     DO 200 I=2,NTH
900 200 TTH(I)=TTH(I-1)+DT
910     WRITE(9,3)(TTH(I),I=1,NTH)
940     DO 4 I=1,KPK
950     DO 4 J=1,N
960 4 BN(I,J)=BL*BN(I,J)
970     EN=NSTAT-1
980     DS=(SFINIS-START)/EN
990     DSO=DS
1000     KD=-START/DS+1.
1010     DO 23 I=1,10
1020     AN(I)=0.0
1030     ANP(I)=0.0
1040     ANPP(I)=0.0
1050     CN(I)=0.0
1060     CNP(I)=0.0
1070 23 GN(I)=0.0
1080     ST=START-DS
1090     MTHMO=NTH-1
1100     S(1)=ST
1110     DO 201 I=1,NSTAT
1120     S(I)=S(I)+DS
1130     DO 211 J=1,NTH
1140     CALL CORD(N,S(I),TTH(J),X(I,J),Y(I,J),KPK)

```

```

1150 211 CONTINUE
1160 S(I+1)=S(I)
1170 201 CONTINUE
1171 DO 204 J=1,NTH
1172 X(1,J)=0.0
1173 204 X(NSTAT,J)=0.0
1180 NSMO=NSTAT-1
1190 NTHMO=NTH-1
1200 DO 202 I=1,NSMO
1210 SSE=S(I)+0.5*DS
1220 DO 202 J=1,NTHMO
1230 MP=J+NTHMO*(I-1)
1240 SE(MP)=SSE
1250 THE(MP)=TTH(J)+0.5*DT
1260 CALL CORD(N,SE(MP),THE(MP),XE(MP),YE(MP),KPK)
1270 202 CONTINUE
1280 DO 203 I=1,NSMO
1290 DO 203 J=1,NTHMO
1300 MP=J+NTHMO*(I-1)
1320 WRITE(7,426) MP
1340 WRITE(7,427) 1,S(I),X(I,J),Y(I,J),TTH(J)
1360 WRITE(7,427)2,S(I+1),X(I+1,J),Y(I+1,J),TTH(J)
1380 WRITE(7,427)3,S(I+1),X(I+1,J+1),Y(I+1,J+1),TTH(J+1)
1400 WRITE(7,427)4,S(I),X(I,J+1),Y(I,J+1),TTH(J+1)
1420 WRITE(7,427)5,SE(MP),XE(MP),YE(MP),THE(MP)
1430 203 CONTINUE
1720 1 FORMAT(12A6)
1730 2 FORMAT(2I10,F10.5)
1740 3 FORMAT(7F10.5)
1750 5 FORMAT(2F10.5,2I10)
1760 8 FORMAT(1H1)
1770 70 FORMAT(3A10)
1775 207 CONTINUE
1780 400 FORMAT(30H DO YOU WISH TO ENTER NEW BL? )
1790 401 FORMAT(1F10.5)
1800 402 FORMAT(20H BL= 1F10.5 )
1810 403 FORMAT(38H HB THMB BETB 3F10.5 )
1820 404 FORMAT(15H NTH= 1I5 )
1830 405 FORMAT(2I5,1F10.5)
1840 406 FORMAT(3F10.5)
1850 407 FORMAT(24H RE= 1F20.1 )
1860 408 FORMAT(4H RE=)
1870 409 FORMAT(24H NSTAT= 1I5 )
1880 410 FORMAT(30H S1 S2 7F10.5 )
1890 411 FORMAT(25H TTH= 1F10.5 )
1900 412 FORMAT(1I1)
1910 413 FORMAT(25H END OF FILE REACHED )
1920 414 FORMAT(38H NINPT DENOTES THE TYPE OF INPUT )
1930 415 FORMAT(23H NINPT=1 CARD DATA )
1940 416 FORMAT(23H 2 REMOTE )
1950 417 FORMAT(23H 3 STORED DATA )

```

```

1960 418 FORMAT(//15H NINPT 111 )
1970 419 FORMAT(43H PROVIDE READ STATEMENTS FOR CARDS AT 301 )
1980 420 FORMAT(27H IF YES TYPE 1 NO TYPE 0 )
1990 421 FORMAT(15H BL= 1F10.5 )
2000 422 FORMAT(20H START= 1F10.5 )
2010 423 FORMAT(20H SFINIS= 1F10.5 )
2020 424 FORMAT(20H NSTAT= 113 )
2030 425 FORMAT(20H NTH= 113 )
2031 426 FORMAT(115)
2033 427 FORMAT(115,4F10.5)
2035 428 FORMAT(12F5.3)
2036 429 FORMAT(113)
2040 END
2050 SUBROUTINE CORD(N,ST,TH,X,Y,KPK)
2060 COMMON /A4/ AN(10),ANP(10),ANPP(10)
2070 CALL BODY(ST,N,KPK)
2080 X=0.0
2090 Y=0.0
2100 DO 23 J=1,N
2110 C1=2*J
2120 C4=3.-C1
2130 CS4=COS(C4*TH)
2140 SS4=SIN(C4*TH)
2150 X=X+AN(J)*CS4
2160 23 Y=Y+AN(J)*SS4
2170 END
2180 SUBROUTINE BODY(ST,N,M)
2190 COMMON /A2/ BN(25,10)
2200 COMMON /A4/ AN(10),ANP(10),ANPP(10)
2210 COMMON /C1/ SP(50)
2220 M2=2*M-1
2230 M3=2*M-4
2240 M4=M2+1
2250 SP(1)=1.0
2260 DO 11 L=2,M2
2270 11 SP(L)=SP(L-1)*ST
2280 DO 22 L=M4,50
2290 22 SP(L)=0.0
2300 DO 12 J=1,N
2310 AN(J)=0.0
2320 ANP(J)=0.0
2330 ANPP(J)=0.0
2340 DO 13 L=1,M
2350 13 AN(J)=AN(J)+BN(L,J)*SP(L)
2360 DO 14 L=2,M
2370 C=L-1
2380 14 ANP(J)=ANP(J)+BN(L,J)*SP(L-1)*C
2390 DO 15 L=3,M
2400 C=(L-2)*(L-1)
2410 15 ANPP(J)=ANPP(J)+BN(L,J)*SP(L-2)*C
2420 12 CONTINUE
2430 RETURN
2440 END

```



```

#FILE (CHXL)DOUBDD ON DTNSRDC
110 FILE 6(KIND=PRINTER,MAXRECSIZE=132)
115 FILE 7(TITLE="CBL3DOPT",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
120 FILE 8(TITLE="DOUBDOPT",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
140 FILE 9(KIND=REMOTE,MAXRECSIZE=22)
160 FILE 11(TITLE="DBLTAPE",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
170 $RESET FREE
3500 C PROGRAM DOUBDD
3600 IMPLICIT REAL*8 (A-H,$,O-Z)
3700 DIMENSION VPP(250,250 ),T(3500),TP(1500),XINTER(250
3800 1,10,3),SIG(500 ),INDEX(500 ,3),SY(2)
3900 DIMENSION NINTER(500 )
4000 DIMENSION IP(500 )
4100 DIMENSION VTEMP(250 ,12,10)
4110 DIMENSION TITLE(12)
4120 DIMENSION TTH(50)
4125 DIMENSION MP(50,50)
4130 COMMON /C1/ SE(250),XE(250),YE(250),THE(250)
4140 COMMON /C2/ S(50),X(50,50),Y(50,50)
4145 COMMON /C3/ NSTAT,NTH,NSMO,NTHMO
4150 READ (7,401) (TITLE(J),J=1,12)
4160 READ (7,402) N,KPK,BL
4170 READ(7,405) START,SFINIS,NSTAT,NTH
4171 NSMO=NSTAT -1
4172 NTHMO=NTH-1
4175 DO 200 I=1,NSMO
4176 DO 200 J=1,NTHMO
4180 READ (7,426) MP(I,J)
4185 LP=MP(I,J)
4190 READ (7,427) M1,S(I),X(I,J),Y(I,J),TTH(J)
4200 READ (7,427)M2,S(I+1),X(I+1,J),Y(I+1,J),TTH(J)
4210 READ (7,427)M3,S(I+1),X(I+1,J+1),Y(I+1,J+1),TTH(J+1)
4220 READ (7,427)M4,S(I),X(I,J+1),Y(I,J+1),TTH(J+1)
4230 READ (7,427)M5,SE(LP),XE(LP),YE(LP),THE(LP)
4231 WRITE(11,426) MP(I,J)
4232 WRITE(11,427) M1,S(I),X(I,J),Y(I,J),TTH(J)
4233 WRITE(11,427)M2,S(I+1),X(I+1,J),Y(I+1,J),TTH(J)
4234 WRITE(11,427)M3,S(I+1),X(I+1,J+1),Y(I+1,J+1),TTH(J+1)
4235 WRITE(11,427)M4,S(I),X(I,J+1),Y(I,J+1),TTH(J+1)
4236 WRITE(11,427)M5,SE(LP),XE(LP),YE(LP),THE(LP)
4240 200 CONTINUE
4400 NP=(NSTAT-1)*(NTH-1)
4500 IF(NP.GT.250) STOP

```

```

4501      WRITE(8,402) NP
4502      WRITE(8,406) NSMO,NTHMO
4510      DO 201 I=1,250
4520      DO 201 J=1,10
4530      DO 201 K=1,3
4540      201 XINTER(I,J,K)=0.0D00
4700      CALL XPAPS(NP,VPP,T,TP,XINTER,SIG,SY,VTEMP,NINTER,INDEX)
4800      SY(1)=1.0
4900      SY(2)=-1.0
5000      FPI=12.5664D00
5100      DO 10 I=1,NP
5200      L=(I-1)*3+1
5300      K=(I-1)*7+1
5400      TP(L)=T(K)
5500      TP(L+1)=T(K+1)
5600      TP(L+2)=T(K+2)
5700      SIG(I)=-TP(L)*12.5664D00
5800      DO 10 J=1,NP
5900      VPP(I,J)=0.0
6000      10 CONTINUE
6100      NUMB=0
6200      DO 20 I=1,NP
6300      NUMB=MAX0(NINTER(I),NUMB)
6400      20 CONTINUE
6500      DO 21 ITER=1,10
6600      DO 21 I=1,NP
6700      INDEX(I,1)=NINTER(I)-ITER
6800      VTEMP(I,12,ITER)=DFLOAT(MIN0(INDEX(I,1),0))+1.0D00
6900      VTEMP(I,12,ITER)=DMAX1(VTEMP(I,12,ITER),0.0D00)
7000      21 CONTINUE
7100      WRITE(6,1001)(VTEMP(I,12,1),I=1,NP)
7200      WRITE(6,1001)(VTEMP(I,12,3),I=1,NP)
7300      DO 1 I=1,NP
7400      DO 23 J=1,NP
7500      VTEMP(J,11,1)=0.0D00
7600      23 CONTINUE
7700      VTEMP(I,11,1)=1.0D10
7800      L=(I-1)*3+1
7900      DO 4 KY=1,2
8000      24 DO 2 ITER=1,NUMB
8100      22 DO 3 J=1,NP
8200      K=(J-1)*7+1
8300      VTEMP(J,KY,ITER)=TP(L)-XINTER(J,ITER,1)
8400      VTEMP(J,KY,ITER)=VTEMP(J,KY,ITER)+VTEMP(J,11,1)
8500      VTEMP(J,KY+2,ITER)=TP(L+1)-XINTER(J,ITER,2)*SY(KY)
8600      VTEMP(J,KY+4,ITER)=TP(L+2)-XINTER(J,ITER,3)
8700      VTEMP(J,KY+6,ITER)=VTEMP(J,KY,ITER)**2+VTEMP(J,KY+2,ITER)**2
8800      1+VTEMP(J,KY+4,ITER)**2
8900      VTEMP(J,KY+8,ITER)=DSQRT(VTEMP(J,KY+6,ITER))
9000      VTEMP(J,KY+8,ITER)=VTEMP(J,KY+8,ITER)*VTEMP(J,KY+6,ITER)
9100      VPP(I,J)=VPP(I,J)+(VTEMP(J,KY,ITER)*T(K+3)+VTEMP(J,KY+2,ITER)*SY(K)

```

```

9200      1Y)*T(K+4)+VTEMP(J,KY+4,ITER)*T(K+5))*T(K+6)/VTEMP(J,KY+8,ITER)
9300      2*VTEMP(J,12,ITER)
9400      3 CONTINUE
9500      2 CONTINUE
9600      4 CONTINUE
9700      DO 17 ITER=1,NUMB
9800      DO 18 KY=1,2
9900      DO 7 J=1,NP
10000     K=(J-1)*7+1
10100     VTEMP(J,KY+4,ITER)=TP(L+2)+XINTER(J,ITER,3)
10200     VTEMP(J,KY+6,ITER)=VTEMP(J,KY,ITER)**2+VTEMP(J,KY+2,ITER)**2
10300     1+VTEMP(J,KY+4,ITER)**2
10400     VTEMP(J,KY+8,ITER)=DSORT(VTEMP(J,KY+6,ITER))
10500     VTEMP(J,KY+8,ITER)=VTEMP(J,KY+8,ITER)*VTEMP(J,KY+6,ITER)
10600     VPP(I,J)=VPP(I,J)+(VTEMP(J,KY,ITER)*T(K+3)+VTEMP(J,KY+2,ITER)*T(K+
10700     14)*SY(KY)+VTEMP(J,KY+4,ITER)*(-T(K+5)))/VTEMP(J,KY+8,ITER)*T(K+6)
10800     2*VTEMP(J,12,ITER)
10900     7 CONTINUE
11000     18 CONTINUE
11100     17 CONTINUE
11200     1 CONTINUE
11300     DO 12 I=1,NP
11400     VPP(I,I)=0.0
11500     VTEMP(I,11,1)=0.0000
11600     12 CONTINUE
11700     DO 8 I=1,NP
11800     DO 6 J=1,NP
11900     VPP(I,J)=VPP(I,J)/DFLOAT(NINTER(J))
12000     6 CONTINUE
12100     8 CONTINUE
12200     DO 15 I=1,NP
12300     SUM=0.0000
12400     DO 14 J=1,NP
12500     SUM=SUM+VPP(I,J)
12600     14 CONTINUE
12700     VTEMP(I,11,1)=SUM
12800     15 CONTINUE
12900     DO 9 I=1,NP
13000     VPP(I,I)=-FP[-VTEMP(I,11,1)]
13100     9 CONTINUE
13200     WRITE(6,1001)(VPP(I,I),I=1,NP)
13300     WRITE(6,1001)(VPP(I,1),I=1,NP,5)
13400     WRITE(11,1002)(VPP(I,I),I=1,NP)
13500     WRITE(6,1003)(NINTER(I),I=1,NP)
13600     WRITE(6,1003) NUMB
13700     WRITE(6,1001)((XINTER(I,J,K),K=1,3),J=1,3),VTEMP(I,11,1),I=1,NP)
13800     CALL DECOMP(NP,NP,VPP,IP)
13900     CALL SOLVE(NP,NP,VPP,SIG,IP)
14000     WRITE(6,1000)
14100     DO 5 I=1,NP
14200     L=(I-1)*3

```

```

14250          WRITE(8,400) (-TP(L+1),TP(L+2),TP(L+3),-SIG(I))
14300      5 WRITE(6,1001) (TP(L+1),TP(L+2),TP(L+3),SIG(I))
14400          WRITE (11,1002) ((T(7*I-7+J),J=1,7),SIG(I),I=1,NP)
14410      STOP
14500      1002 FORMAT(8D10.4)
14600      1000 FORMAT('1',20X,'POTENTIAL')
14700      1001 FORMAT(1H ,10D12.5)
14710      400 FORMAT(10E12.5)
14810      401 FORMAT(12A6)
14820      402 FORMAT(2I10,1F10.5)
14840      405 FORMAT(2F10.5,2I10)
14841      406 FORMAT(2I10)
14850      426 FORMAT(1I5)
14860      427 FORMAT(1I5,4F10.5)
14890      1003 FORMAT(8I10)
14900      END
15000      SUBROUTINE MATINS(A,NR,N1,B,NC,M1,DETERM,ID,INDEX)
15100 C      PIVOT METHOD
15200 C      MATRIX INVERSION WITH ACCOMPANYING SOLUTION OF SIMUL. EQ.
15300 C      PIVOT METHOD
15400 C      FORTRAN IV SINGLE PRECISION WITH ADJUSTABLE DIMENSION
15500 C      FEBRUARY 1966 S GOOD DAVID TAYLOR MODEL BASIN AM MAT4
15600 C      WHERE CALLING PROGRAM MUST INCLUDE
15700 C      DIMENSION A( ), B( ), INDEX( )
15800 C      N IS THE ORDER OF A
15900 C      M IS THE NUMBER OF COLUMN VECTORS IN B(MAY BE 0)
16000 C      DETERM WILL CONTAIN DETERMINANT ON EXIT
16100 C      ID WILL BE SET BY ROUTINE TO 2 IF MATRIX A IS SINGULAR
16200 C      1 IF INVERSION WAS SUCCESSFUL
16300 C      A THE INPUT MATRIX WILL BE REPLACED BY A INVERSE
16400 C      B THE COLUMN VECTORS WILL BE REPLACED BY CORRESPONDING
16500 C      SOLUTION VECTORS
16600 C      INDEX WORKING STORAGE ARRAY
16700 C      IF IT IS DESIRED TO SCALE THE DETERMINANT CARD MAY BE
16800 C      DELETED AND DETERM PRESET BEFORE ENTERING THE ROUTINE
16900 C
17000      IMPLICIT REAL*8 (A-H,$,O-Z)
17100      EQUIVALENCE (IROW,JROW), (ICOLU,JCOLU), (AMAX, T)
17200      DIMENSION A(NR ,NR ),B(NR ,NC ),INDEX(NR ,3)
17300      DIMENSION VT(500),ILI(500),SWAP(500)
17400 C
17500 C      INITIALIZATION
17600 C
17700      N=N1
17800      M=M1
17900      DETERM = 1.0
18000      DO 20 J=1,N
18100      20 INDEX(J,3) = 0
18200      DO 550 I=1,N
18300 C
18400 C      SEARCH FOR PIVOT ELEMENT

```

```

18500 C
18600     AMAX = 0.0
18700     DO 105 J=1,N
18800     IF(INDEX(J,3)-1) 60, 105, 60
18900     60 DO 100 K=1,N
19000     IF(INDEX(K,3)-1) 80, 100, 715
19100     80 IF (     AMAX -DABS (A(J,K))) 85, 100, 100
19200     85 IROW=J
19300     ICOLUM =K
19400     AMAX = DABS (A(J,K))
19500     100 CONTINUE
19600     105 CONTINUE
19700     INDEX(ICOLUM,3) = INDEX(ICOLUM,3) +1
19800     INDEX(I,1)=IROW
19900 C
20000     INDEX(I,2)=ICOLUM
20100 C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL
20200 C
20300     IF (IROW-ICOLUM) 140, 310, 140
20400     140 DETERM=-DETERM
20500     DO 201 L=1,N
20600     SWAP(L)=A(IROW,L)
20700     201 CONTINUE
20800     DO 202 L=1,N
20900     A(IROW,L)=A(ICOLUM,L)
21000     202 CONTINUE
21100     DO 200 L=1,N
21200     200 A(ICOLUM,L)=SWAP(L)
21300     IF(M) 310, 310, 210
21400     210 DO 251 L=1, M
21500     SWAP(L)=B(IROW,L)
21600     251 CONTINUE
21700     DO 252 L=1,M
21800     B(IROW,L)=B(ICOLUM,L)
21900     252 CONTINUE
22000     DO 250 L=1,M
22100     250 B(ICOLUM,L)=SWAP(L)
22200 C
22300 C     DIVIDE PIVOT ROW BY PIVOT ELEMENT
22400 C
22500     310 PIVOT  =A(ICOLUM,ICOLUM)
22600 C     DETERM=DETERM*PIVOT
22700     330 A(ICOLUM,ICOLUM)=1.0
22800     DO 350 L=1,N
22900     350 A(ICOLUM,L)=A(ICOLUM,L)/PIVOT
23000     IF(M) 380, 380, 360
23100     360 DO 370 L=1,M
23200     370 B(ICOLUM,L)=B(ICOLUM,L)/PIVOT
23300 C
23400 C     REDUCE NON-PIVOT ROWS
23500 C

```

```

23600 380 DO 550 L1=1,N
23700     IL1(L1)=L1
23800     IF(IL1(L1)-ICOLUM) 400, 550, 400
23900 400 VT(L1)=A(L1,ICOLUM)
24000     A(L1,ICOLUM)=0.0
24100     DO 450 L=1,N
24200 450 A(L1,L)=A(L1,L)-A(ICOLUM,L)*VT(L1)
24300     IF(M) 550, 550, 460
24400 460 DO 500 L=1,N
24500 500 B(L1,L)=B(L1,L)-B(ICOLUM,L)*VT(L1)
24600 550 CONTINUE
24700 C
24800 C     INTERCHANGE COLUMNS
24900 C
25000     DO 710 I=1,N
25100     L=N+1-I
25200     IF (INDEX(L,1)-INDEX(L,2)) 630, 710, 630
25300 630 JROW=INDEX(L,1)
25400     JCOLUM=INDEX(L,2)
25500     DO 706 K=1,N
25600     SWAP(K)=A(K,JROW)
25700 706 CONTINUE
25800     DO 707 K=1,N
25900     A(K,JROW)=A(K,JCOLUM)
26000 707 CONTINUE
26100     DO 705 K=1,N
26200     A(K,JCOLUM)=SWAP(K)
26300 705 CONTINUE
26400 710 CONTINUE
26500     DO 730 K = 1,N
26600     IF(INDEX(K,3) -1) 715,720,715
26700 720 CONTINUE
26800 730 CONTINUE
26900     ID = 1
27000 810 RETURN
27100 715 ID = 2
27200     GO TO 810
27300     END
27400     SUBROUTINE SOLVE(N,NDIM,A,B,IP)
27500     IMPLICIT REAL*8 (A-H,$,O-Z)
27600     DIMENSION A(NDIM ,NDIM ),B(NDIM ),IP(NDIM )
27700     IF(N.EQ.1) GO TO 9
27800     NM1=N-1
27900     DO 7 K=1,NM1
28000     KP1=K+1
28100     M=IP(K)
28200     T=B(M)
28300     B(M)=B(K)
28400     B(K)=T
28500     DO 7 I=KP1,N
28600     7 B(I)=B(I)+A(I,K)*T

```

```

28700      DO 8 KB=1,NM1
28800      KM1=N-KB
28900      K=KM1+1
29000      B(K)=B(K)/A(K,K)
29100      T=-B(K)
29200      DO 8 I=1,KM1
29300      8 B(I)=B(I) +A(I,K)*T
29400      9 B(1)=B(1)/A(1,1)
29500      RETURN
29600      END
29700      SUBROUTINE XPAPS(NP,VPP,T,TP,XINTER,SIG,SY,VTEMP,NINTER,INDEX)
29705      IMPLICIT REAL*8 (A-H,$,O-Z)
29710      COMMON /C1/ SE(250),XE(250),YE(250),THE(250)
29720      COMMON /C2/ S(50),X(50,50),Y(50,50)
29730      COMMON /C3/ NSTAT,NTH,NSMD,NTHMD
29900      DIMENSION VPP(NP,NP),T(3500),TP(1500),XINTER(NP,
30000      110,3),SIG(NP),INDEX(NP,3),SY(2)
30100      DIMENSION NINTER(NP)
30200      DIMENSION VTEMP(NP,12,10)
30300      WRITE(6,1000) NP
30305      L=0
30410      DO 200 I=1,NSMD
30420      DO 200 J=1,NTHMD
30430      L=L+1
30440      VPP(L,1)=-S(I)
30450      VPP(L,2)=X(I,J)
30460      VPP(L,3)=Y(I,J)
30470      VPP(L,4)=-S(I+1)
30480      VPP(L,5)=X(I+1,J)
30490      VPP(L,6)=Y(I+1,J)
30500      VPP(L,7)=-S(I+1)
30510      VPP(L,8)=X(I+1,J+1)
30520      VPP(L,9)=Y(I+1,J+1)
30530      VPP(L,10)=-S(I)
30540      VPP(L,11)=X(I,J+1)
30550      VPP(L,12)=Y(I,J+1)
30560      VPP(L,13)=-SE(L)
30570      VPP(L,14)=XE(L)
30580      VPP(L,15)=YE(L)
30584      WRITE(11,402)I,J,L
30585      WRITE(11,403)(VPP(L,IP),IP=1,15)
30586      402 FORMAT(3I5)
30587      403 FORMAT(5F10.5)
30590      200 CONTINUE
30600      DO 1 I=1,NP
30700      L=(I-1)*3
30800      DO 1 J=1,3
30900      TP(L+J)=VPP(I,12+J)
31000      1 CONTINUE
31100      DO 11 I=1,NP
31200      VTEMP(I,1,1)=VPP(I,10)-VPP(I,4)

```

```

31300      VTEMP(I,2,1)=VPP(I,1)-VPP(I,7)
31400      VTEMP(I,3,1)=VPP(I,11)-VPP(I,5)
31500      VTEMP(I,4,1)=VPP(I,2)-VPP(I,8)
31600      VTEMP(I,5,1)=VPP(I,12)-VPP(I,6)
31700      VTEMP(I,6,1)=VPP(I,3)-VPP(I,9)
31800      VTEMP(I,4,2)= VTEMP(I,4,1)*VTEMP(I,5,1)-VTEMP(I,3,1)*VTEMP(I,6,1)
31900      VTEMP(I,5,2)= VTEMP(I,1,1)*VTEMP(I,6,1)-VTEMP(I,2,1)*VTEMP(I,5,1)
32000      VTEMP(I,6,2)= VTEMP(I,2,1)*VTEMP(I,3,1)-VTEMP(I,1,1)*VTEMP(I,4,1)
32100      VTEMP(I,1,10)=DSQRT(VTEMP(I,4,2)**2+VTEMP(I,5,2)**2+VTEMP(I,6,2)**
32200      12)
32210      WRITE(11,400)I,(VTEMP(I,J,1),J=1,6),(VTEMP(I,J,2),J=4,6)
32220      WRITE(11,401) VTEMP(I,1,10)
32230      400 FORMAT(1I5,/6F10.5,/3F10.5)
32240      401 FORMAT(1E15.8)
32245      V=VTEMP(I,1,10)
32250      TEST=DABS(V)
32260      IF(TEST .LT.1.0E- 8)GOTO 11
32300      VTEMP(I,4,2)=VTEMP(I,4,2)/VTEMP(I,1,10)
32400      VTEMP(I,5,2)=VTEMP(I,5,2)/VTEMP(I,1,10)
32500      VTEMP(I,6,2)=VTEMP(I,6,2)/VTEMP(I,1,10)
32600      VTEMP(I,7,2)=.5*VTEMP(I,1,10)
32700 C      G3 COMPUTE COVTEMP(I,1,10)NEVTEMP(I,1,10) PTS. (VTEMP(I,1,1),VTEMP
32800      VTEMP(I,1,1)=(VPP(I,1)+VPP(I,4)+VPP(I,7)+VPP(I,10))*0.25
32900      VTEMP(I,2,1)=(VPP(I,2)+VPP(I,5)+VPP(I,8)+VPP(I,11))*0.25
33000      VTEMP(I,3,1)=(VPP(I,3)+VPP(I,6)+VPP(I,9)+VPP(I,12))*0.25
33100      VTEMP(I,4,1)=VTEMP(I,4,2)*(VTEMP(I,1,1)-VPP(I,4))+VTEMP(I,5,2)*
33200      1 VTEMP(I,2,1)-VPP(I,5)+VTEMP(I,6,2)*(VTEMP(I,3,1)-VPP(I,6))
33300      VTEMP(I,1,3)=VTEMP(I,4,2)*VTEMP(I,4,1)
33400      VTEMP(I,2,3)=VTEMP(I,5,2)*VTEMP(I,4,1)
33500      VTEMP(I,3,3)=VTEMP(I,6,2)*VTEMP(I,4,1)
33600      VTEMP(I,1,1)=VPP(I,4)+VTEMP(I,1,3)
33700      VTEMP(I,2,1)=VPP(I,5)+VTEMP(I,2,3)
33800      VTEMP(I,3,1)=VPP(I,6)+VTEMP(I,3,3)
33900      VTEMP(I,4,1)=VPP(I,7)-VTEMP(I,1,3)
34000      VTEMP(I,5,1)=VPP(I,8)-VTEMP(I,2,3)
34100      VTEMP(I,6,1)=VPP(I,9)-VTEMP(I,3,3)
34200      VTEMP(I,7,1)=VPP(I,10)+VTEMP(I,1,3)
34300      VTEMP(I,8,1)=VPP(I,11)+VTEMP(I,2,3)
34400      VTEMP(I,9,1)=VPP(I,12)+VTEMP(I,3,3)
34500      VTEMP(I,10,1)=VPP(I,1)-VTEMP(I,1,3)
34600      VTEMP(I,11,1)=VPP(I,2)-VTEMP(I,2,3)
34700      VTEMP(I,12,1)=VPP(I,3)-VTEMP(I,3,3)
34800 C      G4 COMPUTE CENTVTEMP(I,1,10)OID (VTEMP(I,1,2),VTEMP(I,2,2),VTEM
34900      345 VTEMP(I,1,3)=VTEMP(I,4,1)-VTEMP(I,1,1)
35000      VTEMP(I,2,3)=VTEMP(I,10,1)-VTEMP(I,1,1)
35100      VTEMP(I,3,3)=VTEMP(I,5,1)-VTEMP(I,2,1)
35200      VTEMP(I,4,3)=VTEMP(I,11,1)-VTEMP(I,2,1)
35300      VTEMP(I,5,3)=VTEMP(I,6,1)-VTEMP(I,3,1)
35400      VTEMP(I,6,3)=VTEMP(I,12,1)-VTEMP(I,3,1)
35500      VTEMP(I,1,4)=VTEMP(I,3,3)*VTEMP(I,6,3)-VTEMP(I,4,3)+VTEMP(I,5,3)
35600      VTEMP(I,2,4)=VTEMP(I,5,3)*VTEMP(I,2,3)-VTEMP(I,1,3)*VTEMP(I,6,3)

```



```

35700      VTEMP(I,3,4)=VTEMP(I,1,3)*VTEMP(I,4,3)-VTEMP(I,2,3)*VTEMP(I,3,3)
35800      VTEMP(I,7,3)=DSQRT(VTEMP(I,1,4)**2+VTEMP(I,2,4)**2+VTEMP(I,3,4)**2
35900      1)
36000  348 VTEMP(I,3,2)=(VTEMP(I,6,1)+VTEMP(I,12,1)+(VTEMP(I,7,3)*VTEMP(I,3,1
36100      1)+(VTEMP(I,1,10)-VTEMP(I,7,3))*VTEMP(I,9,1))/VTEMP(I,1,10))/3.
36200      VTEMP(I,2,2)=(VTEMP(I,5,1)+VTEMP(I,11,1)+(VTEMP(I,7,3)*VTEMP(I,2,1
36300      1)+(VTEMP(I,1,10)-VTEMP(I,7,3))*VTEMP(I,8,1))/VTEMP(I,1,10))/3.
36400      VTEMP(I,1,2)=(VTEMP(I,4,1)+VTEMP(I,10,1)+(VTEMP(I,7,3)*VTEMP(I,1,1
36500      1)+(VTEMP(I,1,10)-VTEMP(I,7,3))*VTEMP(I,7,1))/VTEMP(I,1,10))/3.
36600      11 CONTINUE
36700  C      H3 STOVTEMP(I,1,10)E CENTVTEMP(I,1,10)OID AND AVTEMP(I,1,10)EA
36800      DO 10 I=1,NP
36900      K=(I-1)*7
37000  950 T(K+1)=VTEMP(I,1,2)
37100      T(K+2)=VTEMP(I,2,2)
37200      T(K+3)=VTEMP(I,3,2)
37300  C      H4 STOVTEMP(I,1,10)E NOVTEMP(I,1,10)MAL VECTOVTEMP(I,1,10)
37400      T(K+4)=VTEMP(I,4,2)
37500      T(K+5)=VTEMP(I,5,2)
37600      T(K+6)=VTEMP(I,6,2)
37700      T(K+7)=VTEMP(I,7,2)
37800      10 CONTINUE
37900      DO 70 I=1,NP
38000      VTEMP(I,1,5)=(VPP(I,4)+VPP(I,7))*0.5
38100      VTEMP(I,2,5)=(VPP(I,5)+VPP(I,8))*0.5
38200      VTEMP(I,3,5)=(VPP(I,6)+VPP(I,9))*0.5
38300      VTEMP(I,1,6)=(VPP(I,1)+VPP(I,10))*0.5
38400      VTEMP(I,2,6)=(VPP(I,2)+VPP(I,11))*0.5
38500      VTEMP(I,3,6)=(VPP(I,3)+VPP(I,12))*0.5
38600      NINTER(I)=(VTEMP(I,1,6)-VTEMP(I,1,5))*2/(VTEMP(I,7,2)+1.5D00)
38700      NINTER(I)=MAX0(NINTER(I),1)
38800      NINTER(I)=MIN0(NINTER(I),8)
38900      NUMB=NINTER(I)
39000      VTEMP(I,1,6)=(VTEMP(I,1,6)-VTEMP(I,1,5))/NINTER(I)
39100      VTEMP(I,2,6)=(VTEMP(I,2,6)-VTEMP(I,2,5))/NINTER(I)
39200      VTEMP(I,3,6)=(VTEMP(I,3,6)-VTEMP(I,3,5))/NINTER(I)
39300      DO 999 NIX=1,NUMB
39400      DNIX=DFLOAT(NIX)-0.5D00
39500      XINTER(I,NIX,1)=VTEMP(I,1,5)+VTEMP(I,1,6)*DNIX
39600      XINTER(I,NIX,2)=VTEMP(I,2,5)+VTEMP(I,2,6)*DNIX
39700      XINTER(I,NIX,3)=VTEMP(I,3,5)+VTEMP(I,3,6)*DNIX
39800      999 CONTINUE
39900      IF(NINTER(I).NE.1) GO TO 998
40000      DO 997 NIX=1,3
40100      XINTER(I,1,NIX)=VTEMP(I,NIX,2)
40200      997 CONTINUE
40300      998 CONTINUE
40400  C      WX(I)=VPP(I,1)-VPP(I,4)
40500      70 CONTINUE
40600  C      IF(LIN.LT.60) GO TO 80
40700      WRITE (6,7)

```

```

40800     WRITE (6,8)
40900     LIN=0
41000 C 80 LIN=LIN+1
41100     DO 55 I=1,NP
41200     M=(I-1)*7+1
41300     55 WRITE (6,6) T(M),T(M+1),T(M+2),T(M+3),T(M+4),T(M+5),T(M+6)
41400     4 CONTINUE
41500     NQE=NP
41600     NQ7=NQE*7
41700     WRITE(6,28) NP
41800     6 FORMAT (10D12.5)
41900     7 FORMAT(38H1 INPUT DATA TO POTENTIAL FLOW PROGRAM)
42000     8 FORMAT(4H0 ,2HXP,10X,2HYP,10X,2HZP,10X,2HYN,10X,2HZN
42100     1,10X,2HAQ)
42200     24 FORMAT (1H ,I1,19H PLANES OF SYMMETRY)
42300     28 FORMAT (33H0TOTAL NUMBER OF QUADRILATE = ,I5)
42400     1000 FORMAT(10I5)
42500     1001 FORMAT(16F5.1)
42600     RETURN
42700     END
42710     SUBROUTINE DECOMP(N,NDIM,A,IP)
42720     IMPLICIT REAL*8 (A-H,$,O-Z)
42730     DIMENSION A(NDIM,NDIM),IP(NDIM)
42740     IF(N)=1
42750     DO 6 K=1,N
42760     IF(K.EQ.N) GO TO 5
42770     KP1=K+1
42780     M=K
42790     DO 1 I=KP1,N
42800     IF(DABS(A(I,K)).GT.DABS(A(M,K))) M=I
42810     1 CONTINUE
42820     IF(K)=M
42830     IF(M.NE.K) IP(N)=-IP(N)
42840     T=A(M,K)
42850     A(M,K)=A(K,K)
42860     A(K,K)=T
42870     IF(T.EQ.0.) GO TO 5
42880     DO 2 I=KP1,N
42890     2 A(I,K)=-A(I,K)/T
42900     DO 4 J=KP1,N
42910     T=A(M,J)
42920     A(M,J)=A(K,J)
42930     A(K,J)=T
42940     IF(T.EQ.0.) GO TO 4
42950     DO 3 I=KP1,N
42960     3 A(I,J)=A(I,J)+A(I,K)*T
42970     4 CONTINUE
42980     5 IF(A(K,K).EQ.0.) IP(N)=0
42990     6 CONTINUE
43000     RETURN
43010     END

```

```

100 FILE 5(KIND=REMOTE,MAXRECSIZE=22)
110 FILE 6(KIND=PRINTER,MAXRECSIZE=132)
120 FILE 7(TITLE="TBLIPCOPT",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
130 FILE 8(TITLE="TBL3DOPT",KIND=DISK,FILETYPE=7)
140 FILE 9(KIND=REMOTE,MAXRECSIZE=22)
150 FILE 10(TITLE="TBL3DINP",KIND=DISK,FILETYPE=7)
160 FILE 11(TITLE="TBLSAVE",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
165 FILE 12(TITLE="THREEDPHI",KIND=DISK,FILETYPE=7)
170 $RESET FREE
1010 C PROGRAM TBLIPC
1030 C PROGRAM COMPUTES DERIVATIVES OF VELOCITIES ALONG THE ORTHOGONAL
1040 C COORDINATES.
1050 DIMENSION TITLE(12),TITL(3)
1060 DIMENSION X(3,120),Y(3,120),TH(3,120),UP(3,120),UT(3,120),S(3,120)
1070 DIMENSION XKP(3,120),XKT(3,120),H(3,120),F(3,120),G(3,120)
1080 DIMENSION JJ(3),SS(3)
1090 COMMON/SPLIN/SF(30,60),SPHI(30,60),SPHP(30,60),SPHPP(30,60),
1100 $ DTP(30,60)
1110 COMMON/VELOC/UP,UT,S,TH
1120 READ (8,401) (TITLE(J),J=1,12)
1130 WRITE(6,401) (TITLE(J),J=1,12)
1140 WRITE(7,401) (TITLE(J),J=1,12)
1145 WRITE(9,401) (TITLE(J),J=1,12)
1180 READ (8,402) N,KPK,BL
1190 WRITE(6,402) N,KPK,BL
1200 WRITE(7,402) N,KPK,BL
1205 WRITE(9,402) N,KPK,BL
1210 READ (8,405) START,SFINS,NSTAT,NTH
1220 WRITE(6,405) START,SFINS,NSTAT,NTH
1230 WRITE(7,405) START,SFINS,NSTAT,NTH
1235 WRITE(9,405) START,SFINS,NSTAT,NTH
1240 PIOT=1.57079682679
1250 EN=NTH-1
1260 MTH=NTH-1
1270 DT=PIOT/EN*(-1.0)
1280 DTI=1.0/DT
1290 WRITE(9,408)NTH,DT,DTI
1300 DTF=0.5*DTI
1310 J=1
1320 K=2
1330 L=3
1331 WRITE(9,406)
1332 WRITE(9,407)
1335 READ (5,404) NS
1338 IF(NS.EQ.0) GO TO 300

```

```

1340     READ (12,403)NS,NT
1350     WRITE(6,403)NS,NT
1370     CALL SPOTEN(NS,NT,DTH,NTH)
1380     TEST=SP(1,1)
1390 300 CONTINUE
1400     DO 205 I=1,2
1410 301 CONTINUE
1420     READ (8,427) JJ(I),SS(I)
1430     WRITE(6,427) JJ(I),SS(I)
1440     DO200 IT=1,NTH
1450     READ (8,408) NDUM,X(I,IT),Y(I,IT),TH(I,IT),U,V
1460     WRITE(6,408) NDUM,X(I,IT),Y(I,IT),TH(I,IT),U,V
1470     READ (8,408) NDUM,W,UP(I,IT),UT(I,IT),UN,S(I,IT)
1480     WRITE(6,408) NDUM,W,UP(I,IT),UT(I,IT),UN,S(I,IT)
1490     READ (8,408) NDUM,XKP(I,IT),XKT(I,IT),H(I,IT),F(I,IT),G(I,IT)
1500     WRITE(6,408) NDUM,XKP(I,IT),XKT(I,IT),H(I,IT),F(I,IT),G(I,IT)
1510 200 CONTINUE
1520     NSTAT=NSTAT-1
1530     IF(TEST.GE.SS(1)) GOTO 301
1540     NSTAT=NSTAT+1
1550 205 CONTINUE
1560     IF(NS.EQ.0) GO TO 302
1570     CALL EVALVO(NS,DTH,NTH,1)
1580     CALL EVALVO(NS,DTH,NTH,2)
1590 302 CONTINUE
1600     HI=1.0/H(2,1)
1610     DTIG=DTI/6(1,1)
1620     DUPDLP=(UP(2,1)-UP(1,1))/(S(2,1)-S(1,1))
1630     DUPDLP=HI*DUPDLP
1640     DUPDLT=DTIG*(UP(1,2)-UP(1,1))
1650     DUTDLP=(UT(2,1)-UT(1,1))/(S(2,1)-S(1,1))
1660     DUTDLP=HI*DUTDLP
1670     DUTDLT=DTIG*(UT(1,2)-UT(1,1))
1680     WRITE(7,427) JJ(1),SS(1)
1690     WRITE(6,400) S(1,1)
1700     WRITE(6,408) 1,X(1,1),Y(1,1),TH(1,1),UP(1,1),UT(1,1)
1710     WRITE(7,408) 1,X(1,1),Y(1,1),TH(1,1),UP(1,1),UT(1,1)
1720     WRITE(6,408) 2,S(1,1),DUPDLP,DUPDLT,DUTDLP,DUTDLT
1730     WRITE(7,408) 2,S(1,1),DUPDLP,DUPDLT,DUTDLP,DUTDLT
1740     WRITE(6,408) 3,XKP(1,1),XKT(1,1),H(1,1),F(1,1),G(1,1)
1750     WRITE(7,408) 3,XKP(1,1),XKT(1,1),H(1,1),F(1,1),G(1,1)
1760     DO 201 IT=2,NTH
1770     DS=1.0/(S(2,IT)-S(1,IT))
1780     DS=DS/H(2,IT)
1790     DUPDLP=DS*(UP(2,IT)-UP(1,IT))

```

```

1800      DUTDLP=DS*(UT(2,IT)-UT(1,IT))
1810      DTF6=DTF/G(1,IT)
1820      DUPDLT=DTFG*(UP(1,IT+1)-UP(1,IT-1))
1830      DUTDLT=DTFG*(UT(1,IT+1)-UT(1,IT-1))
1840      WRITE(6,408) 1,X(1,IT),Y(1,IT),TH(1,IT),UP(1,IT),UT(1,IT)
1850      WRITE(7,408) 1,X(1,IT),Y(1,IT),TH(1,IT),UP(1,IT),UT(1,IT)
1860      WRITE(6,408) 2,S(1,IT),DUPDLP,DUPDLT,DUTDLP,DUTDLT
1870      WRITE(7,408) 2,S(1,IT),DUPDLP,DUPDLT,DUTDLP,DUTDLT
1880      WRITE(6,408) 3,XKP(1,IT),XKT(1,IT),H(1,IT),F(1,IT),G(1,IT)
1890      WRITE(7,408) 3,XKP(1,IT),XKT(1,IT),H(1,IT),F(1,IT),G(1,IT)
1900  201 CONTINUE
1910      DS=1.0/(S(2,NTH)-S(1,NTH))
1920      DS=DS/H(2,NTH)
1930      DUPDLP=DS*(UP(2,NTH)-UP(1,NTH))
1940      DUTDLP=DS*(UT(2,NTH)-UT(1,NTH))
1950      DTIG=DTI/G(1,NTH)
1960      DUPDLT=DTIG*(UP(1,NTH)-UP(1,NTH))
1970      DUTDLT=DTIG*(UT(1,NTH)-UT(1,NTH))
1980      WRITE(6,408) 1,X(1,NTH),Y(1,NTH),TH(1,NTH),UP(1,NTH),UT(1,NTH)
1990      WRITE(7,408) 1,X(1,NTH),Y(1,NTH),TH(1,NTH),UP(1,NTH),UT(1,NTH)
2000      WRITE(6,408) 2,S(1,NTH),DUPDLP,DUPDLT,DUTDLP,DUTDLT
2010      WRITE(7,408) 2,S(1,NTH),DUPDLP,DUPDLT,DUTDLP,DUTDLT
2020      WRITE(6,408) 3,XKP(1,NTH),XKT(1,NTH),H(1,NTH),F(1,NTH),G(1,NTH)
2030      WRITE(7,408) 3,XKP(1,NTH),XKT(1,NTH),H(1,NTH),F(1,NTH),G(1,NTH)
2040
2050  C      OUTPUT FOR FIRST PHI-STATION IS COMPLETED.
2060
2070      DO 202 IS=3,NSTAT
2080      WRITE(7,427) JJ(K),SS(K)
2090      READ (8,427) JJ(L),SS(L)
2100      WRITE(6,427) JJ(L),SS(L)
2110      DO 203 IT=1,NTH
2120      READ (8,408) NDUM,X(L,IT),Y(L,IT),TH(L,IT),U,V
2130      READ (8,408) NDUM,W,UP(L,IT),UT(L,IT),UN,S(L,IT)
2140      READ (8,408) NDUM,XKP(L,IT),XKT(L,IT),H(L,IT),F(L,IT),G(L,IT)
2150  203 CONTINUE
2160      IF(NS.EQ.0) GO TO 303
2170      CALL EVALVO(NS,DTH,NTH,L)
2180  303 CONTINUE
2190      DTIG=DTI/G(K,1)
2200      DS=1.0/(S(L,1)-S(J,1))
2210      DS=DS/H(K,1)
2220      DUPDLP=DS*(UP(L,1)-UP(J,1))
2230      DUTDLP=DS*(UT(L,1)-UT(J,1))
2240      DUPDLT=DTIG*(UP(K,2)-UP(K,1))

```

```

2250      DUTDLT=DTIG*(UT(K,2)-UT(K,1))
2260      WRITE(6,400) S(K,1)
2270      WRITE(7,408) 1,X(K,1),Y(K,1),TH(K,1),UP(K,1),UT(K,1)
2280      WRITE(6,408) 2,S(K,1)          ,DUPDLP,DUPDLT,DUTDLP,DUTDLT
2290      WRITE(7,408) 2,S(K,1)          ,DUPDLP,DUPDLT,DUTDLP,DUTDLT
2300      WRITE(7,408) 3,XKP(K,1),XKT(K,1),H(K,1),F(K,1),G(K,1)
2310      DO 204 IT=2,NTH
2320      DS=1.0/(S(L,IT)-S(J,IT))
2330      DS=DS/H(K,IT)
2340      DTFG=DTF/G(K,IT)
2350      DUPDLP=DS*(UP(L,IT)-UP(J,IT))
2360      DUTDLP=DS*(UT(L,IT)-UT(J,IT))
2370      DUPDLT=DTFG*(UP(K,IT+1)-UP(K,IT-1))
2380      DUTDLT=DTFG*(UT(K,IT+1)-UT(K,IT-1))
2390      WRITE(7,408) 1,X(K,IT),Y(K,IT),TH(K,IT),UP(K,IT),UT(K,IT)
2400      WRITE(7,408) 2,S(K,IT),DUPDLP,DUPDLT,DUTDLP,DUTDLT
2410      WRITE(7,408) 3,XKP(K,IT),XKT(K,IT),H(K,IT),F(K,IT),G(K,IT)
2420 204 CONTINUE
2430      DS=1.0/(S(L,NTH)-S(J,NTH))
2440      DS=DS/H(K,NTH)
2450      DTIG=DTI/G(K,NTH)
2460      DUPDLP=DS*(UP(L,NTH)-UP(J,NTH)    )
2470      DUTDLP=DS*(UT(L,NTH)-UT(J,NTH)    )
2480      DUPDLT=DTIG*(UP(K,NTH)-UP(K,NTH))
2490      DUTDLT=DTIG*(UT(K,NTH)-UT(K,NTH))
2500      WRITE(7,408) 1,X(K,NTH),Y(K,NTH),TH(K,NTH),UP(K,NTH),UT(K,NTH)
2510      WRITE(7,408) 2,S(K,NTH),DUPDLP,DUPDLT,DUTDLP,DUTDLT
2520      WRITE(7,408) 3,XKP(K,NTH),XKT(K,NTH),H(K,NTH),F(K,NTH),G(K,NTH)
2530
2540 C      COMPLETED OUTPUT FOR K-TH PHI-STATION.
2550
2560      M=J
2570      J=K
2580      K=L
2590 202 L=M
2600
2610 CC      DATA COMPLETED FOR ALL BUT LAST PHI-STATION.
2620
2630 500 CONTINUE
2640 400 FORMAT(5X,28H W.L. STATION COORDINATE IS      ,1F7.4)
2642 401 FORMAT(12A6)
2644 402 FORMAT(2I10,1F10.5)
2646 403 FORMAT(2I5)
2647 404 FORMAT(1I1)
2648 405 FORMAT(2F10.5,2I10)
2650 406 FORMAT(55H IF THREE DIMENSIONAL POTENTIAL IS FROM MING'S PROGRAM )
2652 407 FORMAT(33H ENTER 1 SLENDER BODY ENTER 0      )
2690 408 FORMAT(1I3,5E15.7)
2700 427 FORMAT(1I3,1F9.6)
2710 470 FORMAT(3A10)
2750      END

```

```

2770      SUBROUTINE EVALVO(NS,DTH,NTH,IS)
2780
2790 C      COMPUTES UP AND UT FROM SPLINE DATA
2800 C          SP          S COORDINATES AT WHICH SPHI IS GIVEN
2810 C          SPHI       SPLINE POTENTIAL
2820 C          SPHP       DERIVATIVE OF SPHI IN S DIRECTION
2830 C          SPHPP      SECOND DERIVATIVE OR MOMENT OF SPHI
2840 C          DTP        SPACING INCREMENTS IN S
2850
2860      COMMON/SPLIN/SP(30,60),SPHI(30,60),SPHP(30,60),SPHPP(30,60),
2870      $ DTP(30,60)
2880
2890 C          UP          VELOCITY IN PHI DIRECTION
2900 C          UT          VELOCITY IN THATA DIRECTION
2910 C          S          S COORDINATE AT WHICH VELOCITIES ARE EVALUATED
2920 C          TH         THATA COORDINATE
2930
2940      COMMON/VELOC/UP,UT,S,TH
2950      DIMENSION UP(3,120),UT(3,120),S(3,120),TH(3,120)
2960
2970
2980 C          SPS         INTERPOLATION POINTS FOR LONGITUDINAL SPLINE
2990 C          PH          POTENTIAL AT SPS
3000 C          FM         MOMENTS
3010 C          DT         S INCREMENTS
3020 C          T          INTERPOLATION POINTS THATA DIRECTION
3030 C          SPHIS      POTENTIAL AT T
3040 C          PHP        DERIVATIVE OF THE POTENTIAL
3050 C          DELTA      INCREMENTS IN T
3060 C          MY         MOMENTS
3070
3080      DIMENSION SPS(120),PH(120),FM(120),DT(120),T(120),SPHIS(120)
3090      $,PHP(120),DELTA(120),MY(120)
3100      REAL MY
3110      PI=3.1415962
3120      TOL=1.0E-06
3130      MTH=2*NTH
3140
3150 C      COMPUTE UP
3160
3170      DO 200 J=1,NTH
3180      SL=S(IS,J)
3190      T(J)=TH(IS,J)
3200      L=MTH+1-J
3210      I(L)=PI-T(J)

```

```

3220      DO201 I=1,NS
3230      SPS(I)=SP(I,J)
3240      PH(I)=SPHI(I,J)
3250      FM(I)=SPHPP(I,J)
3260      201 DT(I)=DTP(I,J)
3270      CALL SPLEVN(SL,SPHIS(J),UP(IS,J),NS,TOL,SPS,PH,FM,DT)
3280      SPHIS(L)=SPHIS(J)
3290      200 CONTINUE
3300
3310 C    COMPUTE UT
3320
3330      CALL PLINE(MTH,TOL,T,SPHIS,FM,DELTA)
3340      DO 202 J=1,MTH
3350      CALL SPLEVN(T(J),DUM,PHP(J),MTH,TOL,T,SPHIS,FM,DELTA)
3360      202 CONTINUE
3370      CALL PLINE (MTH,TOL,T,PHP,FM,DELTA)
3380      DO 203 J=1,MTH
3390      CALL SPLEVN(T(J),DUM,MY(J),MTH,TOL,T,PHP,FM,DELTA)
3400      203 CONTINUE
3410      DO 204 J=1,MTH
3420      CALL SPLEVN(T(J),DUM,UT(IS,J),MTH,TOL,T,PH,MY,DELTA)
3430      204 CONTINUE
3440      RETURN
3450      END

```



```

3470      SUBROUTINE SPOTEN(NS,NT,DTH,NTH)
3480
3490 C      SUBROUTINE USES SPLINE ON SPLINE INTERPOLATION TO FIT POTENTIAL
3500 C          SP          S COORDINATES AT WHICH SPHI IS GIVEN
3510 C          SPHI       SPLINE POTENTIAL
3520 C          SPHP       DERIVATIVE OF SPHI IN S DIRECTION
3530 C          SPHPP      SECOND DERIVATIVE OR MOMENT OF SPHI
3540 C          DTP        SPACING INCREMENTS IN S
3550
3560      COMMON/SPLIN/SP(30,60),SPHI(30,60),SPHP(30,60),SPHPP(30,60),
3570      $ DTP(30,60)
3580
3590 C          T          ARCTAN(X/Y)
3600 C          X          X COORDINATE
3610 C          Y          Y COORDINATE
3620 C          PHI       POTENTIAL FROM MING'S PROGRAM
3630
3640      DIMENSION T(30,30),X(30,30),Y(30,30),PHI(30,30)
3650
3660 C          TL          SPLINE INTERPOLATION COORDINATES
3670 C          PH          SPECIFIED FUNCTION VALUES FOR SPLINE FIT
3680 C          FM          MOMENTS
3690 C          DT          SPACING
3700 C          S          LONGITUDINAL POINTS AT WHICH POTENTIAL IS SPECIFIED
3710 C          TH          THETA COORDINATES AT WHICH SPHI IS SPECIFIED
3720 C          MY          MOMENTS
3730
3740      DIMENSION TL(60),PH(60),FM(60),DT(60),S(60),MY(60)
3750      DIMENSION PHP(60),TH(60)
3760      REAL MY
3770
3780 C      READ INPUT AND CONVERT TO PROGRAMS COORDINATE SYSTEM
3790
3800      READ (12,400) ((SP(I,J),X(I,J),Y(I,J),D,DU,DUM,PHI(I,J),J=1,NT)
3810      $,I=1,NS)
3820      WRITE(6,400) ((SP(I,J),X(I,J),Y(I,J),D,DU,DUM,PHI(I,J),J=1,NT)
3830      $,I=1,NS)
3840      PI=3.1415926
3850      TOL=1.0E-06
3860      LBC=3
3870      NT=2*NT
3880      TH(1)=0.0
3890      DO 200 I=1,NS
3900      DO 200 J=1,NT
3910      SP(I,J)=-SP(I,J)

```

```

3920     PHI(I,J)=-PHI(I,J)
3930 200 T(I,J)=ATAN2(X(I,J),Y(I,J))
3940
3950 C   BEGIN SPLINE ON SPLINE FIT OF POTENTIAL IN TRANSVERSE PLANES.
3960
3970     DO 201 IS=1,NS
3980     DO 202 J=1,NT
3990     L=NT+1-J
4000     TL(J)=T(IS,J)
4010     PH(J)=PHI(IS,J)
4020     TL(L)=PI-TL(J)
4030 202 PH(L)=PH(J)
4040     CALL PLINE(MT,TOL,TL,PH,FM,DT)
4050     DO 203 J=1,MT
4060     CALL SPLEVN(TL(J),DUMMY,PHP(J),MT,TOL,TL,PH,FM,DT)
4070 203 CONTINUE
4080     CALL PLINE(MT,TOL,TL,PHP,FM,DT)
4090     DO 204 J=1,MT
4100     CALL SPLEVN(TL(J),DUMMY,MY(J),MT,TOL,TL,PHP,FM,DT)
4110 204 CONTINUE
4120     DO 201 J=1,NTH
4130     CALL SPLEVN(TL(J),SPHI(IS,J),DUMMY,MT,TOL,TL,PH,MY,DT)
4140 201 CONTINUE
4150
4160 C   BEGIN LONGITUDINAL SPLINE FIT
4170
4180     DO 206 J=1,NTH
4190     DO 207 I=1,NS
4200     S(I)=SP(I,J)
4210     PH(I)=SPHI(I,J)
4220 207 CONTINUE
4230     CALL SPLINE(NS,TOL,S,PH,FM,DT)
4240     DO 208 I=1,NS
4250     CALL SPLEVN(S(I),DUMMY,PHP(I),NS,TOL,S,PH,FM,DT)
4260 208 CONTINUE
4270     CALL SPLINE(NS,TOL,S,PHP,FM,DT)
4280     DO 209 I=1,NS
4290     CALL SPLEVN(S(I),DUM,MY(I),NS,TOL,S,PHP,FM,DT)
4300 209 CONTINUE
4310     DO 206 I=1,NS
4320     CALL SPLEVN(S(I),SPHI(I,J),DUM,NS,TOL,S,PH,MY,DT)
4330     SPHP(I,J)=DUM
4340     SPHPP(I,J)=MY(I)
4350     DTP(I,J)=DT(I)
4360 206 CONTINUE
4370     RETURN
4380 400 FORMAT(8D10.4)
4390     END

```

```

4410      SUBROUTINE SPLINE(N,TOL,X,Y,M,H)
4420 C      NONPERIODIC SPLINE FIT USING THE MOMENTS M(J) TO SPECIFY THE SPLIN
4430 C      N NUMBER OF INTERPOLATION POINTS
4440 C      TOL TOLERANCE ON SOLUTION
4450 C      X MESH POINTS A=X(1).LT.X(2).LT. ... .LT.X(N)=B
4460 C      Y ORDINATES Y(1)=F(X(1)),ETC.
4470 C      H INTERVAL LENGTHS H(J)=X(J)-X(J-1)
4480 C      LBC CONTROL NUMBER TO PICK END CONDITIONS LBC=
4490 C      1 FIRST DERIVATIVE OF F AT X(1) AND AT X(N+1) IS GIVEN
4500 C      2 M(1)=YOP,M(N)=YNP
4510 C      3 M(1)=M(2),M(N-1)=M(N)
4520 C      YOP DERIVATIVE AT X(1) OR VALUE OF M(1)
4530 C      YNP DERIVATIVE AT X(N) OR VALUE OF M(N)
4540      COMMON/ENDCOD/LBC,YOP,YNP
4550      DIMENSION X(N),Y(N),M(N),H(N)
4560      DIMENSION L(99),MU(99),D(99)
4570      DIMENSION P(99),Q(99),U(99)
4580      DIMENSION DUMMYB(99),DC(99),TC(99)
4590      DIMENSION R(99),DT(99),UP(99),UN(99)
4600      REAL M,L,MU
4610      NMO=N-1
4620      DO 210 I=2,N
4630      H(I)=X(I)-X(I-1)
4640 210 CONTINUE
4650      DO 200 I=2,NMO
4660      L(I)=H(I+1)/(H(I)+H(I+1))
4670      MU(I)=1.0-L(I)
4680      D(I)=6.0*L(I)/H(I+1)*((Y(I+1)-Y(I))/H(I+1)-(Y(I)-Y(I-1))/H(I))
4690 200 CONTINUE
4700      H(N)=X(N)-X(N-1)
4710      IF(LBC=2) 300,301,302
4720 300 L(1)=1.0
4730      D(1)=6.0/H(2)*((Y(2)-Y(1))/H(2)-YOP)
4740      MU(N)=1.0
4750      D(N)=6.0/H(N)*((YNP-(Y(N)-Y(N-1))/H(N))
4760      GOTO 303
4770 301 L(1)=0.0
4780      D(1)=2.0*YOP
4790      MU(N)=0.0
4800      D(N)=2.0*YNP
4810      GOTO 303
4820 302 L(1)=-2.0
4830      D(1)=0.0
4840      MU(N)=-2.0
4850      D(N)=0.0

```

```

4860 303 B=2.0
4870 P(1)=B
4880 Q(1)=-L(1)/P(1)
4890 U(1)=D(1)/P(1)
4900 L(N)=0.0
4910 R(1)=0.0
4920 DT(1)=D(1)/P(1)
4930 DO 201 I=2,N
4940 P(I)=MU(I)*Q(I-1)+B
4950 Q(I)=-L(I)/P(I)
4960 R(I)=MU(I)/P(I)
4970 201 DT(I)=D(I)/P(I)
4980 UP(1)=U(1)
4990 UN(1)=0.0
5000 DO 205 I=1,N
5010 UP(I)=DT(I)+R(I)*UN(I-1)
5020 UN(I)=R(I)*UP(I-1)
5030 205 U(I)=UP(I)-UN(I)
5040 M(N)=U(N)
5050 DO 202 I=2,N
5060 K=N+1-I
5070 202 M(K)=Q(K)+M(K+1)+U(K)
5080 DUMMYB(1)=B
5090 DC(1)= 2.0*M(1)+L(1)*M(2)
5100 TC(1)=D(1)-DC(1)
5110 DUMMYB(N)=B
5120 DC(N)=2.0*M(N)+MU(N)*M(N-1)
5130 TC(N)=U(N)-DC(N)
5140 DO 207 I=2,NM0
5150 DUMMYB(I)=B
5160 DC(I)=2.0*M(I)+MU(I)*M(I-1)+L(I)*M(I+1)
5170 207 TC(I)=D(I)-DC(I)
5180 K=1
5190 TL=D(1)-2.0*M(1)-L(1)*M(2)
5200 TL=ABS(TL)
5210 IF(TL.GE.TOL) GOTO 501
5220 K=N
5230 TL=D(N)-2.0*M(N)-MU(N)*M(N-1)
5240 TL=ABS(TL)
5250 IF(TL.GE.TOL) GOTO 501
5260 DO 203 I=2,NM0
5270 K=1
5280 T=D(I)-2.0*M(I)-MU(I)*M(I-1)-L(I)*M(I+1)
5290 T=ABS(T)
5300 TL=T

```

```
5310     IF(T.GE.TOL) GOTO 501
5320     203 CONTINUE
5330     500 RETURN
5340     501 WRITE(6,400)K,TL,TOL
5345     WRITE(9,400)K,TL,TOL
5350     MU(1)=-999999.9
5360     L (N)=-999999.9
5370     WRITE(6,401) (      MU(I),DUMMYB(I),L(I),M(I),B(I),DC(I),TC(I),
5380     1 I=1,N)
5390     RETURN
5400     400 FORMAT(15,2E15.4)
5410     401 FORMAT(7E15.8)
5420     END
```

```

5440     SUBROUTINE PLINE(N,TOL,X,Y,M,H)
5450 C    PERIODIC SPLINE FIT USING THE MOMENTS M(J) TO SPECIFY THE SPLINE
5460 C    N NUMBER OF INTERPOLATION POINTS
5470 C    TOL TOLERANCE ON SOLUTION
5480 C    X MESH POINTS A=X(1).LT.X(2).LT....LT.X(N)=B
5490 C    Y ORDINATES Y(J)=F(X(J))
5500 C    M MOMENTS M(J)=F''(X(J))
5510 C    H INTERVAL LENGTHS H(J)=X(J)-X(J-1)
5520     DIMENSION X(N),Y(N),M(N),H(N)
5530     DIMENSION L(99),MU(99),D(99)
5540     DIMENSION P(99),Q(99),U(99),S(99),T(99),V(99)
5550     DIMENSION DC(99),TC(99)
5560     REAL M,L,MU
5570     TY=ABS (Y(1)-Y(N))
5580     IF(TY.GE.TOL) WRITE(6,403) Y(1),Y(N)
5590     NMO=N-1
5600     DO 210 I=2,N
5610     H(I)=X(I)-X(I-1)
5620 210 CONTINUE
5630     H(N+1)=H(2)
5640     Y(N+1)=Y(2)
5650     DO 200 I=2,N
5660     L(I)=H(I+1)/(H(I)+H(I+1))
5670     MU(I)=1.0-L(I)
5680     D(I)=6.0*L(I)/H(I+1)*((Y(I+1)-Y(I))/H(I+1)-(Y(I)-Y(I-1))/H(I))
5690 200 CONTINUE
5700     B=2.0
5710     P(2)=B
5720     Q(2)=-L(2)/P(2)
5730     U(2)=D(2)/P(2)
5740     S(2)=-MU(2)/P(2)
5750     DO 201 I=3,N
5760     P(I)=MU(I)*Q(I-1)+B
5770     Q(I)=-L(I)/P(I)
5780     S(I)=-MU(I)*S(I-1)/P(I)
5790 201 U(I)=(D(I)-MU(I)*U(I-1))/P(I)
5800     T(N)=1.0
5810     V(N)=0.0
5820     DO 205 I=2,NMO
5830     K=N+1-I
5840     T(K)=Q(K)*T(K+1)+S(K)
5850     V(K)=Q(K)*V(K+1)+U(K)
5860 205 CONTINUE
5870     C=L(N)*T(2)+MU(N)*T(N-1)+B
5880     DV=L(N)*V(2)+MU(N)*V(N-1)

```

```

5890      M(N)=(D(N)-DV)/C
5900      DO 202 I=2,NMO
5910      K=N+1-I
5920 202  M(K)=T(K)*M(N)+V(K)
5930      M(1)=M(N)
5940      M(N+1)=M(2)
5950      DO 206 I=2,N
5960      DC(I)=2.0*M(I)+MU(I)+M(I-1)+L(I)*M(I+1)
5970 206  TC(I)=D(I)-DC(I)
5980      DO 203 I=2,N
5990      K=I
6000      TL=TC(I)
6010      TL=ABS (TL)
6020      IF(TL.GE.TOL) GOTO 501
6030 203  CONTINUE
6040      RETURN
6050 501  WRITE(6,400)K,TL,TOL
6060      WRITE(6,401) (M(I),D(I),DC(I),TC(I),I=2,N)
6070      RETURN
6080 400  FORMAT(15,2E15.4)
6090 401  FORMAT(4E15.8)
6100 403  FORMAT(40H2Y(1) IS NOT EQUAL Y(N), NOT PERIODIC ,2E15.8)
6110      END

```

```

6120      SUBROUTINE SPLEVM(XE,YE,YP,N,TOL,X,Y,M,H)
6130 C    EVALUATES A SPLINE INTERPOLATION AT XE WHEN THE MOMENTS M(J) ARE
6140 C    GIVEN AT X(J).
6150 C    YE      IS THE VALUE OF THE SPLINE AT XE
6160 C    H(J)    INTERVAL LENGTHS
6170 C    N      NUMBER OF MESH POINTS
6180 C    Y(J)    INTERPOLATION VALUES AT INTERPOLATION POINTS X(J) RESP.
6190      DIMENSION X(N),Y(N),M(N),H(N)
6200      REAL M
6210      J=1
6220 300 J=J+1
6230      IF(J .GT.N) GOTO 501
6240      IF(XE.GT.X(J))GO TO 300
6250      YE=(M(J-1)*((X(J)-XE)**3)+M(J)*((XE-X(J-1))**3))/6.0
6260      YE=YE+(Y(J-1)-M(J-1)*H(J)*H(J)/6.0)*(X(J)-XE)+(Y(J)-M(J)*H(J)*
6270 1 H(J)/6.0)*(XE-X(J-1))
6280      YE=YE/H(J)
6290      YP=(-M(J-1)*(X(J)-XE)*(X(J)-XE)+M(J)*(XE-X(J-1))*(XE-X(J-1)))*0.5
6300      YP=(YP+Y(J)-Y(J-1))/H(J)
6310      YP=YP-(M(J)-M (J-1))*H(J)/6.0
6320      RETURN
6330 501 WRITE(6,400) XE,X(N)
6340      RETURN
6350 400 FORMAT(4H XE,1E10.4,13H ,EXCEEDS DX      ,1E10.4)
6360      END

```



```

#FILE (CHXL)TBL SOL ON DTNSRDC
100 FILE 5(KIND=REMOTE,MAXRECSIZE=22)
110 FILE 6(KIND=PRINTER,MAXRECSIZE=132)
120 FILE 7(TITLE="TBL SOLOFT",KIND=DISK,PROTECTION=SAVE,MAXRECSIZE=22)
130 FILE 8(TITLE="TBL IPOFT",KIND=DISK,FILETYPE=7)
140 FILE 9(KIND=REMOTE,MAXRECSIZE=22)
150 FILE 10(TITLE="MING3DOFT",KIND=DISK,FILETYPE=7)
160 FILE 11(TITLE="THHBETA",KIND=DISK,FILETYPE=7)
170 $RESET FREE
180 C PROGRAM TBL SOL
190 C PROGRAM TO SOLVE PARTIAL DIFFERENTIAL EQUATIONS . USES THE O BRIEN
200 C $ HYMAN,AND KAPLAN IMPLICIT FORMULATION
210 C THE RESULTING SYSTEM OF LINEAR EQUATIONS ARE SOLVED USING A
220 C $ DIRECT GAUSSIAN REDUCTION APPLIED TO THE SPARSE COEFFICIENT MATRI
230 COMMON/INT/NTH,NTH.NS,DELTAT
240 COMMON/VAR/XV
250 COMMON/COOD/ODL,B,ODR,CR
260 COMMON/COEFC/PC
270 COMMON/DATAIN/ TH,UP,UT,S,UPLP,UTLP,UPLT,UTLT
280 COMMON/GEDIN/RP,KT,H,F,G,G
290 COMMON/JON/JD, JN
300 COMMON/TOLERN/TOL
310 REAL RP,KT
320 DIMENSION TITLE(12),TITL(3)
330 DIMENSION X(2,150),Y(2,150), UP(2,150),UT(2,150)
340 DIMENSION TH(2,150)
350 DIMENSION PC(2,4)
360 DIMENSION AV(150,3)
370 DIMENSION G(2,150),UPLP(2,150),UTLP(2,150),UPLT(2,150),UTLT(2,150)
380 DIMENSION PP(2,150),TT(2,150),RR(2,150),F(2,150),GG(2,150)
390 DIMENSION D(150,3,3),ODL(150,3,3),ODR(150,3,3),CR(150,3)
400 DATA(PC(I,J),J=1,4),I=1,2)
410 $-0.000701,0.028345,-0.386768,0.019521,
420 $ -0.001953,0.062588,-0.83480,0.191511)
430 WRITE(9,411)((PC(I,J),J=1,4),I=1,2)
440 WRITE(9,412)
450 WRITE(9,413)
460 WRITE(9,414)
470 READ(5,415) NS
480 WRITE(9,415) NS
490 WRITE(9,416)
500 WRITE(9,417)
510 READ (5,415) NI
520 WRITE(9,415) NI
530 WRITE(6,410) NS,NI
540 DATA TOL/0.100E-09/

```

```

550      WRITE(9,409) TOL
560 C    INPUT PRELIMINARY DATA FROM TAPE
570      READ (8,401) (TITLE(J),J=1,12)
580      WRITE(6,401) (TITLE(J),J=1,12)
590      WRITE(7,401) (TITLE(J),J=1,12)
600      READ (8,402) N,KPK,BL
610      WRITE(6,402) N,KPK,BL
620      WRITE(7,402) N,KPK,BL
630      READ (8,405) START,SFINS,NSTAT,NTH
640      WRITE(6,405) START,SFINS,NSTAT,NTH
650      WRITE(7,405) START,SFINS,NSTAT,NTH
660      NTH=NTH-1
670 C    SUBROUTINE INITIAL DEFINES THE INITIAL DATA
680      CALL INITAL
690      JO=1
700      JN=2
710      READ (8,427) JJ,SS
720      WRITE(6,427) JJ,SS
730      READ (8,427) JJ,SS
740      WRITE(6,427) JJ,SS
750      WRITE(7,427) JJ,SS
760 C    ENTER GEOMETRY AND POTENTIAL DATA FOR INITIAL LINE.
770      DO 200 ITH=1,NTH
780      READ (8,408) NDUM,X(1,ITH),Y(1,ITH),TH(1,ITH),UP(1,ITH),UT(1,ITH)
790      WRITE(6,408) NDUM,X(1,ITH),Y(1,ITH),TH(1,ITH),UP(1,ITH),UT(1,ITH)
800      WRITE(7,408) NDUM,X(1,ITH),Y(1,ITH),TH(1,ITH),UP(1,ITH),UT(1,ITH)
810      READ (8,408) NDUM,S(1,ITH),UPLP(1,ITH),UPLT(1,ITH),UTLP(1,ITH),
820      $ UTLT(1,ITH)
830      WRITE(6,408) NDUM,S(1,ITH),UPLP(1,ITH),UPLT(1,ITH),UTLP(1,ITH),
840      $ UTLT(1,ITH)
850      WRITE(7,408) NDUM,S(1,ITH),UPLP(1,ITH),UPLT(1,ITH),UTLP(1,ITH),
860      $ UTLT(1,ITH)
870      READ (8,408) NDUM,KP(1,ITH),KT(1,ITH),H(1,ITH),F(1,ITH),GG(1,ITH)
880      WRITE(6,408) NDUM,KP(1,ITH),KT(1,ITH),H(1,ITH),F(1,ITH),GG(1,ITH)
890      WRITE(7,408) NDUM,KP(1,ITH),KT(1,ITH),H(1,ITH),F(1,ITH),GG(1,ITH)
900 200 CONTINUE
910      DELTAT=TH(1,2)
920      DO 201 IS=1,NSTAT
930      MS=IS
940      IF(IS.GT.N5) GO TO 500
950      READ(8,427) JJ,SS
960      WRITE(6,427) JJ,SS
970      DO 202 IT=1,NTH
980 C    ENTER GEOMETRY AND POTENTIAL DATA FOR PHI-STATION JN.
990      READ (8,408) NDUM,X(JN,IT),Y(JN,IT),TH(JN,IT),UP(JN,IT),UT(JN,IT)

```

```

1000     READ (8,408) NDUM,S(JN,IT),UPLP(JN,IT),UPLT(JN,IT),UTLP(JN,IT)
1010     $ ,UTLT(JN,IT)
1020     READ (8,408) NDUM,KP(JN,IT),KT(JN,IT),H(JN,IT),F(JN,IT),GG(JN,IT)
1030     202 CONTINUE
1040     DO 205 II=1,NI
1050 C     COMPUTE COEFFICIENTS FOR LINEAR EQUATIONS
1060 C     SUBROUTINE ZSYPLN COMPUTES THE COEFFICIENT ALONG THE SYMMETRY
1070 C     $ LINE THATA EQUAL ZERO, THE LOAD WATER LINE.
1080     CALL ZSYPLN
1090 C
1100 C
1110 C     SUBROUTINE LECOEF COMPUTES THE COEFFICIENTS OF THE LINEAR
1120 C     EQUATIONS EXCEPT ON THE SYMMETRY LINES.
1130 C
1140 C
1150     CALL LECOEF
1160     203 CONTINUE
1170 C
1180 C     SUBROUTINE FSYPLN COMPUTES THE COEFFICIENTS ON THE SYMMETRY PLANE.
1190 C     THATA = PI/2.
1200 C
1210     CALL FSYPLN
1220 C     SOLUTION OF LINEAR EQUATIONS IS XV
1230 C     SUBROUTINE SOLVE COMPUTES THE SOLUTION.
1240     CALL SOLVE
1250     602 CONTINUE
1260     WRITE(8,404)
1270     K=JN
1280     DO 204 ITH=1,NTH
1290     WRITE(6,403) TH(K,ITH),X(K,ITH),Y(K,ITH),S(K,ITH),(XV(ITH,I),I=1,3)
1300     WRITE(7,403) TH(K,ITH),X(K,ITH),Y(K,ITH),S(K,ITH),(XV(ITH,I),I=1,3)
1310     204 CONTINUE
1320     205 CONTINUE
1330     JO=JN
1340     IF(JO.EQ.1) JN=2
1350     IF(JO.EQ.2) JN=1
1360     201 CONTINUE
1370     500 CONTINUE
1380     401 FORMAT(12A6)
1390     402 FORMAT(2I10,1F10,5)
1400     403 FORMAT(4F10,4,3E20,8)
1410     404 FORMAT(1H2,4X,6H THATA,6X,2H X,9X,2H Y,9X,2H S,11X,5H THATA,10X,
1420     $ 2H T,19X,2H H)
1430     405 FORMAT(2F10,5,2(10)
1440     406 FORMAT(10X,3E15,7)

```

```
1450 407 FORMAT(5X,3E15.7)
1460 408 FORMAT(1I3,5E15.7)
1470 409 FORMAT(1E14.5)
1480 410 FORMAT(5I4      )
1490 411 FORMAT(4F10.6)
1500 412 FORMAT(54H NS IS NUMBER OF STATIONS AT WHICH BOUNDARY LAYER IS  )
1510 413 FORMAT(19H TO BE CALCULATED  )
1520 414 FORMAT(19H NS=  1I3      )
1530 415 FORMAT(1I3)
1540 416 FORMAT(43H NI IS NUMBER OF ITERATIONS AT EACH STEP      )
1550 417 FORMAT(16H NI=  1I3      )
1560 421 FORMAT(26H INPUT DATA OUT OF ORDER.  /218)
1570 427 FORMAT(1I3,1F9.6)
1580 470 FORMAT(3A10)
1590      END
```

```

1600      SUBROUTINE INVERS(A,AI,T)
1610 C    COMPUTES INVERSE OF 3*3 MATRIX A
1620 C    AI IS INVERSE TIS TOLERANCE
1630      DIMENSION A(3,3),AI(3,3) ,TM(3,3),EM(3,3)
1640      DET=A(1,1)*A(2,2)*A(3,3)+A(1,2)*A(2,3)*A(3,1)+A(2,1)*A(3,2)*A(1,3)
1650      D=A(1,1)*A(2,3)*A(3,2)+A(1,2)*A(2,1)*A(3,3)+A(1,3)*A(2,2)*A(3,1)
1660      DET=DET-D
1670      DAB=ABS (DET)
1680      IF(DAB.LT.T) GO TO 501
1690      BI=1.0/DET
1700      AI(1,1)= A(2,2)*A(3,3)-A(3,2)*A(2,3)
1710      AI(2,1)= A(2,3)*A(3,1)-A(2,1)*A(3,3)
1720      AI(3,1)=-A(2,2)*A(3,1)+A(2,1)*A(3,2)
1730      AI(1,2)=+A(1,3)+A(3,2)-A(1,2)*A(3,3)
1740      AI(3,2)=-A(1,1)*A(3,2)+A(1,2)*A(3,1)
1750      AI(2,2)= A(1,1)*A(3,3)-A(1,3)*A(3,1)
1760      AI(1,3)= A(1,2)*A(2,3)-A(1,3)*A(2,2)
1770      AI(2,3)=-A(1,1)*A(2,3)+A(1,3)*A(2,1)
1780      AI(3,3)= A(1,1)*A(2,2)-A(1,2)*A(2,1)
1790      DO 200 I=1,3
1800      DO 200 J=1,3
1810      AI(I,J)=BI*AI(I,J)
1820      200 CONTINUE
1830      DO 201 I=1,3
1840      DO 201 J=1,3
1850      TM(I,J)=0.0
1860      EM(I,J)=0.0
1870      IF(I.EQ.J) EM(I,J)=1.0
1880      DO 201 K=1,3
1890      201 TM(I,J)=TM(I,J)+A(I,K)*AI(K,J)
1900      DO 202 I=1,3
1910      DO 202 J=1,3
1920      IS=I
1930      JS=J
1940      DTM=TM(I,J)-EM(I,J)
1950      ATM=ABS(DTM)
1960      IF(ATM.GT.1) GO TO 502
1970      202 CONTINUE
1980      500 RETURN
1990      501 WRITE(6,400) DET,T
2000      WRITE(6,401) A
2010      STOP
2020      502 WRITE(6,400) DET,T
2030      WRITE(6,401) A
2040      WRITE(6,401) AI

```

```
2050      WRITE(6,403) IS,JS
2060      WRITE(6,401) TM
2070      WRITE(6,401) EM
2080      WRITE(6,402)
2090      STOP
2100  400  FORMAT(5H2DET=,1E15.7,5X,3H T=,1E15.7 //2H A//)
2110  401  FORMAT(3E15.7)
2120  402  FORMAT(25H INVERS IS NOT CORRECT      )
2130  403  FORNAT(2I5)
2140      END
```

```

2150      SUBROUTINE INITAL
2160 C    DEFINES INITIAL CONDITIONS
2170      COMMON/VAR/XV
2180      COMMON/INT/NTH,NTH.NS,DELTAT
2190      DIMENSION XV(150,3)
2200      WRITE(9,402)
2210      WRITE(9,403) ← WRITE (9,411)
2220      READ(5,404) NC
2230      IF(NC-1)100,101,102
2240 100  WRITE(9,405)
2250      READ (5,401) TH11
2260      WRITE(9,401) TH11
2270      WRITE(9,406)
2280      READ (5,401) T
2290      WRITE(9,401) T
2300      WRITE(9,407)
2310      READ (5,401) H
2320      WRITE(9,401) H
2330      WRITE(6,401)TH11,T,H
2340      DO 200 ITH=1,NTH
2350      XV(ITH,1)=TH11
2360      XV(ITH,2)=T
2370 200  XV(ITH,3)=H
2380      WRITE(6,400) TH11,T,H
2390      WRITE(7,401) TH11,T,H
2400      RETURN
2410 101  READ (11,408) NTHCK
2420      IF(NTH.NE.NTHCK) GOTO 300
2430      READ (11,401) ((XV(I,J),J=1,3),I=1,NTH)
2440      WRITE( 6,401) ((XV(I,J),J=1,3),I=1,NTH)
2450      WRITE( 7,401) ((XV(I,J),J=1,3),I=1,NTH)
2460      RETURN
2470 300  WRITE(9,408)NTH,NTHCK
2480      WRITE(9,409)
2490      STOP
2500 102  WRITE(9,410)
2510      STOP
2520 400  FORMAT(16H1INITIAL VALUES //6H TH11=1F10.4,5X,3H T= 1F10.4,
2530      $ 5X,3H H=1F10.4//)
2540 401  FORMAT(3F10.5)
2550 402  FORMAT(45H IF THE INITIAL VALUES ARE CONSTANT ENTER 0 )
2560 403  FORMAT(22H OTHERWISE ENTER A X ) 2
2570 404  FORMAT(11H )
2580 405  FORMAT(19H TH11= 1F10.5 )
2590 406  FORMAT(36H T= TAN(BE1A)= 1F10.5 )

```

411 Format (if the Initial Values
are specified as a function of them Enter 1)

```
2600 407 FORMAT(19H H=      1F10.5  )
2610 408 FORMAT(5H NTH=,1I3,/7H NTHCK=,1I3 ///)
2620 409 FORMAT(48H NUMBER OF INITIAL DATA NOT COMPATIBLE WITH NTH )
2630 410 FORMAT(47H WRITE OWN ROUTINE FOR ENTERING INITIAL DATA  )
2640     END
```



```

2650      SUBROUTINE COEFF(TH,H,SA,CA,T)
2660 C      ALL FUNCTIONS COMPUTED IN THIS SUBROUTINE ARE FUNCTIONS OF THATA.
2670 C      H, ALPHA, AND T
2680 C      SA=SIN(ALPHA), CA=COS (ALPHA)
2690 C      A IS THE COEFFICIENT MATRIX FROM THE LEFT SIDE OF THE PDE
2700 C      B IS THE COEFFICIENT MATRIX FROM THE RIGHT SIDE
2710 C      C IS THE FORCING FUNCTION
2720 C      VARIABLES BEGINNING WITH G OR H ARE DEFINED IN SUBROUTINE GS
2730 C      PC ARE THE COEFFICIENTS OF THE PLYNOMIALS A AND B (NOT RELATED
2740 C      TO COEFFICIENT MATRICES) WHICH BRING IN THE REYNOLDS NUMBER
2750 C      DEPENDENCE.
2760      COMMON/CH/FG,DGDH
2770      COMMON/CABC/A,B,C
2780      COMMON/CG/G,H1,H2,TFG,SFG
2790      COMMON/CGD/GH,GA,GT,H1T,H1H,H2T,H2H
2800      COMMON/CGDD/GHH,GHA,GHT,GAA,GAT,GTT
2810      COMMON/CINPUT/U/LP,VLP,ULTH,VLTH,UM,KP,KT,UMLTH,UNLP
2820      COMMON/COEFPC/PC
2830      COMMON HT
2840      REAL KP,KT
2850      DIMENSION A(3,3),B(3,3),C(3)
2860      DIMENSION G(2,2),GH(2,2),GA(2,2),GT(2,2)
2870      DIMENSION GHH(2,2),GHT(2,2),GHA(2,2),GAA(2,2),GAT(2,2),GTT(2,2)
2880      DIMENSION PC(2,4)
2890      UMLT=UMLTH
2900      CALL GS(H,SA,CA,T,1)
2910 C      COMPUTE A
2920      600 A(1,1)=G(1,1)
2930      A(1,2)=TH*GT(1,1)
2940      A(1,3)=TH*GH(1,1)
2950      A(2,1)=G(2,1)
2960      A(2,2)=TH*GT(2,1)
2970      A(2,3)=TH*GH(2,1)
2980      A(3,1)=TFG
2990      A(3,2)=TH*H1T
3000      A(3,3)=TH*H1H
3010 C      COMPUTE B
3020      B(1,1)=G(1,2)
3030      B(1,2)=TH*GT(1,2)
3040      B(1,3)=TH*GH(1,2)
3050      B(2,1)=G(2,2)
3060      B(2,2)=TH*GT(2,2)
3070      B(2,3)=TH*GH(2,2)
3080      B(3,1)=SFG
3090      B(3,2)=TH*H2T

```

```

3100      B(3,3)=TH*H2H
3110      IF(MT.NE.3) GOTO 100
3120      WRITE(6,400) FG,DGDH,TH,H,T,SA,CA
3130      WRITE(6,403) G
3140      WRITE(6,403) GH
3150      WRITE(6,403)GT
3160      WRITE(6,402) A
3170      WRITE(6,402) B
3180      100 CONTINUE
3190 C      REYNOLDS NUMBER INFLUENCE
3200 C      REYNOLD'S NUMBER BASED ON LENGTH IS SET AT 10**6
3210      RL=1.0E 06
3220      RT=RL*TH*UM
3230      RT=ABS(RT)
3240      RC=ALOG(RT)
3250      PA=PC(1,1)
3260      PB=PC(2,1)
3270      DO 200 J=2,4
3280      PA=RC*PA+PC(1,J)
3290      200 PB=RC*PB+PC(2,J)
3300      CF1=EXP (PA*H+PB)
3310      CFP=CF1*(CA+T*SA)
3320      CFT=CF1*(-SA+T*CA)
3330      DELTA1=TH*H1
3340      DELTA2=TH*H2
3350      TF=TH*TFG
3360      SF=TH*SFG
3370 C      COMPUTE C
3380      UMI=1.0/(1.0+UM)
3390      C(1)= 0.5*CFP-2.0*TH*UMI*(G(1,1)*UMLP+G(1,2)*UMLT)
3400      IF(MT.EQ.3) WRITE(6,401) C(1)
3410      C(1)= C(1)-UMI*(DELTA1*ULP+DELTA2*ULTH)
3420      IF(MT.EQ.3) WRITE(6,401) C(1)
3430      XKT=G(1,2)+G(2,1)+SA*H1
3440      XKP=G(1,1)-G(2,2)-SA*H2
3450      IF(MT.NE.3)GO TO 300
3460      WRITE(6,400) PA,PB,CF1,CFP,CFT
3470      WRITE(6,400) DELTA1,DELTA2,TF,SF,UMI
3480      WRITE(6,400) XKT,XKP
3490      300 CONTINUE
3500      C(1)= C(1)+TH*(KP*XKP+KT*XK1)
3510      IF(MT.EQ.3) WRITE(6,401) C(1)
3520      C(2)=0.5*CFT-2.0*UMI*(G(2,1)*UMLP+G(2,2)*UMLT)+TH
3530      C(2)= C(2)-UMI*(DELTA1*VLP+DELTA2*VLTH)
3540      XKF=G(1,2)+G(2,1)+CA*H2

```

```

3550      XKT=G(2,2)-G(1,1)-CA*H1
3560      C(2)= C(2)+TH*(KP*XKP+KT*XKT)
3570      C(3)=0.025*H-0.022
3580      IF(MT.EQ.3) WRITE(6,401) C(3)
3590      UMI=1.0/UM
3600      C(3)= C(3)-UMI*(TF*UMLP+SF*UMLT)
3610      IF(MT.EQ.3) WRITE(6,401) C(3)
3620      C(3)= C(3)+TF*KP+SF*KT
3630      IF(MT.EQ.3) WRITE(6,401) C(3)
3640      500 RETURN
3650      400 FORMAT(3X,5E15.6)
3660      401 FORMAT(3H C= ,1E15.6)
3670      402 FORMAT(3E15.7)
3680      403 FORMAT(2E15.7)
3690 CATALOG(BLTDMY,BTBLCOEF,ID=CHXL,RP=120)
3700      END

```

```

3710      SUBROUTINE GS(H,SA,CA,T,MG)
3720 C      ALL FUNCTIONS COMPUTED IN THIS SUBROUTINE ARE FUNCTIONS OF H.
3730 C      $ ALPHA AND T
3740 C      SA IS SIN (ALPHA) AND CA IS COS(ALPHA)
3750 C      G(I,J) ARE THE FUNCTIONS G SUBSCRIP IJ
3760 C      GH IS THE DERIVATIVE OF G WITH RESPECT TO H
3770 C      GA "" "" "" "" "" "" "" ALPHA
3780 C      GT "" "" "" "" "" "" "" T
3790 C      GHH,GHT,GHA,GGT,GTA,GAA ARE THE RESPECTIVE SECOND PARTIAL
3800 C      $ DERIVATIVES.
3810      COMMON/CF/FB
3820      COMMON/CFD/FBP,FBPP
3830      COMMON/CG/G,H1,H2,TFG,SFG
3840      COMMON/CGD/GH,GA,GT,H1H,H1H,H2H,H2H
3850      COMMON/CFG/FG,DGDH
3860      DIMENSION FB(4),FBP(4),FBPP(4)
3870      DIMENSION G(2,2),GH(2,2),GA(2,2),GT(2,2)
3880      CALL FS(H)
3890 C      COMPUTE H1,H2,H11. AND H22
3900 C      H11 WILL BE COMPUTED AS TFG
3910 C      H22 WILL BE COMPUTED AS SFG
3920 C      H1H,H1H,H2H,H2H ARE PARTIAL DERIVATIVES WITHWITH RESPECT TO T AND
3930 C      $ H OF THE FUNCTIONS H11 AND H22
3940      H1T=-FB(4)*SA
3950      H2T=FB(4)*CA
3960      H1=H*CA+T*H1T
3970      H2=H*SA+T*H2T
3980 C      COMPUTE THE FUNCTION T=G*COS(ALPHA)+T*FB4*SIN(ALPHA)
3990 C      COMPUTE THE FUNCTION          S=G*SIN(ALPHA)-T*FB4*COS(ALPHA)
4000      FG=2.0*H/(H-1.0)
4010      TFG=FG*CA+T*FB(4)*SA
4020      SFG=FG*SA-T*FB(4)*CA
4030      H1T=-H1T
4040      H2T=-H2T
4050      DGDH=-2.0/((H-1.0)*(H-1.0))
4060      H1H=CA*DGDH+T*FBP(4)*SA
4070      H2H=SA*DGDH-T*FBP(4)*CA
4080 C      COMPUTATION OF G"S
4090      CAS=CA*CA
4100      SAS=SA*SA
4110      SACA=SA*CA
4120      TS=T*T
4130      FOP1=FB(1)+FB(2)
4140      G(1,1)=CAS-T*(FOP1*SACA+TS+FB(3)*SAS
4150      F=1.0-TS*FB(3)

```

```

4160      G(1,2)=P*SACA+T*(FB(2)*CAS-FB(1)*SAS)
4170      G(2,1)=P*SACA+T*(FB(1)*CAS-FB(2)*SAS)
4180      G(2,2)=SAB+T*FOPT*SALA      +TS*FB(3)*CAS
4190 C    COMPUTE FIRST DERIVATIVES
4200      DSMSS=CAS-SAS
4210      GT(1,1)=FOPT*SACA+2.0*T*FB(3)*SAB
4220      GH(1,1)=-T*(FB(1)+FB(2))*SACA+TS*(FB(1)+FB(2))*SAS
4230      GT(1,2)=-2.0*T*FB(3)*SACA+FB(2)*CAS-FB(1)*SAS
4240      GH(1,2)=-TS*FB(3)*SALA+T*(FB(2)*CAS-FB(1)*SAS)
4250      GA(2,1)=-P*DSMSS+2.0*T*FOPT*SABA
4260      GB(2,1)=-2.0*T*FB(3)*SACA+FB(2)*CAS-FB(1)*SAS
4270      GH(2,1)=-TS*FB(3)*SALA+T*(FB(2)*CAS-FB(1)*SAS)
4280      GT(2,2)=FOPT*SACA+2.0*T*FB(3)*SAB
4290      GH(2,2)=-T*(FB(1)+FB(2))*SACA+TS*(FB(1)+FB(2))*SAS
4300      RETURN
4310      END

```

```

4320      SUBROUTINE VECDEF
4330 C      COMPUTES THE COEFFICIENTS OF THE LINEAR EQUATIONS RESULTING FROM
4340 C      THE IMPLICIT FORMULATION OF THE FDE.
4350      COMMON INT NTRM,N,NODE,TAT
4360      COMMON COE D(10,10),ODR(1,4)
437      COMMON VAR X,Y
438      COMMON COE KX,KY
4390      COMMON DATA I,NODE,DT,STORE,DIFF,DIFF2,DIFF3
440      COMMON BEIN P(1,1),C(1,1)
441      COMMON BA(1,1),BB(1,1)
442      COMMON DINC D(10,10),D(10,10),D(10,10),D(10,10),D(10,10)
4430      COMMON ION IONIN
4440      COMMON ION IONOUT
4450      COMMON RI
4460      REAL DT
4470      DIMENSION X(10),Y(10)
4480      DIMENSION D(10,10),D(10,10),D(10,10),D(10,10),D(10,10)
4490      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4500      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4510      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4520      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4530      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4540      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4550      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4560      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4570      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4580      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4590      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4600      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4610      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4620      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4630      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4640      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4650      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4660      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4670      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4680      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4690      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4700      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4710      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4720      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4730      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4740      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4750      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4760      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4770      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4780      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4790      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4800      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4810      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4820      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4830      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4840      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4850      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4860      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4870      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4880      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4890      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4900      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4910      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4920      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4930      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4940      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4950      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4960      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4970      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4980      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
4990      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)
5000      DIMENSION ION(10),ION(10),ION(10),ION(10),ION(10),ION(10)

```

```

4770      WRITE(6,400) XKP,XKT
4780      WRITE(6,406) A
4790      WRITE(6,407) B
4800      WRITE(6,406) C
4810  301 CONTINUE
4820      ADEL=ABS(      DELTAT)
4830      IF(ADEL.LE.TOL) GO TO 500
4840      IF(H(JN,IT).LT.TOL) GO TO 300
4850      IF(GG(JN,IT).LT.TOL) GO TO 300
4860      SS=S(JN,IT)-S(JO,IT)
4870      SST=0.5*SS/DELTAT
4880      HI=1.0/H(JN,IT)
4890      GG1=1.0/GG(JN,IT)
4900      DO 201 IS=1,3
4910      C(IS)=SS*C(IS)
4920      DO 201 JS=1,3
4930      B(IS,JS)=SST+B(IS,JS)+GG1
4940  401 AS(IS,JS)=HI*A(IS,JS)
4950      CALL INVERS(AS,ASI,TOL)
4960      IF(IT.NE.3) GO TO 302
4970      WRITE(6,407) AS
4980      WRITE(6,406) ASI
4990      WRITE(6,407) B
5000      WRITE(6,406) C
5010  302 CONTINUE
5020      DO 202 IB=1,3
5030      DO 202 JB=1,3
5040      ODR(IT,IB,JB)=0.0
5050      DO 202 K=1,3
5060  202 ODR(IT,IB,JB)=ODR(IT,IB,JB)+ASI(IB,K)*BCK(JB)
5070      DO 203 IB=1,3
5080      CR(IT,IB)=XV(IT,IB)
5090      DO 203 K=1,3
5100  203 CR(IT,IB)=CR(IT,IB)+SI(IB,K)*C(K)
5110  200 CONTINUE
5120  500 RETURN
5130  400 WRITE(6,401)
5140      WRITE(6,402) JN,MT,DELTAT,H(JN,MT),GG(JN,MT)
5150  400 FORMAT(10X,5E15.7)
5160  401 FORMAT(5X,30H DELTAT,H, OR GG ARE ZERO.
5170  402 FORMAT(5X,215.3E15.7)
5180  403 FORMAT(16H3FROM LECOEF MT= ,14//10X,5H TH11,15X,2H (1.15*,2H H)
5190  406 FORMAT(15X,3E15.7 )
5200  407 FORMAT (5X,3E15.6)
5210      END

```

```

5210 SUBROUTINE ZSYPLN
5220 C COMPUTES THE COEFFICIENTS OF THE LINEAR EQUATIONS RESULTING FROM
5240 C THE IMPLICIT FORMULATION OF THE PDE ON THE LINE OF SYMMETRY
5250 C THATA= 0.0
5260 COMMON/INT/NTH,MTH,NS,DELTAI
5270 COMMON/CDOD/DDL,D,ODR,CR
5280 COMMON/VAR/XV
5290 COMMON/COEFC/PC,RL
5300 COMMON/DATAIN/ TH,UP,UT,S,UPLP,UTLP,UPLT,UTLT
5310 COMMON/GEOIN/KP,KT,H,F,GG
5320 COMMON/CABC/ A,B,C
5330 COMMON/CINPUT/VLT,VLP,ULT,VLP,UM,XKP,XRT,UMLT,UMLP
5340 COMMON/CON/DB,IN
5350 REAL KP,KT
5360 DIMENSION XV(150,3)
5370 DIMENSION PC(2,4)
5380 DIMENSION UP(2,150),UT(2,150),TH(2,150)
5390 DIMENSION S(2,150),UPLP(2,150),UTLP(2,150),UPLT(2,150),UTLT(2,150)
5400 DIMENSION KP (2,150),KT (2,150),H(2,150),F(2,150),GG(2,150)
5410 DIMENSION D(150,3,3),DDL(150,3,3),ODR(150,3,3),CR(150,3)
5420 DIMENSION A(3,3),B(3,3),C(3)
5430 DIMENSION AS(3,3), ASI(3,3)
5440 IT=1
5450 NT=IT
5460 ALPHA=ATAN2(UT(JN,IT),UP(JN,IT))
5470 UM=UT(JN,IT)*UT(JN,IT)+UP(JN,IT)*UP(JN,IT)
5480 UM=SQRT(UM)
5490 SA=UT(JN,IT)/UM
5500 CA=UP(JN,IT)/UM
5510 ULP=UPLP(JN,IT)
5520 VLP=UTLP(JN,IT)
5530 ULT=UPLT(JN,IT)
5540 VLT=UTLT(JN,IT)
5550 XKP=KP(JN,IT)
5560 XRT=RT(JN,IT)
5570 UMLT=CA*ULT+SA*VLT
5580 UMLP=CA*ULP+SA*VLP
5590 CALL COEFF(XV(IT,1),XV(IT,3),SA,CA,XV(IT,2))
5600 IF(H(JN,IT).LT.0.000001) GO TO 590
5610 IF(GG(JN,IT).LT.0.000001) GO TO 590
5620 SS=S(JN,IT)+0.11
5630 SST=0.5*(S+BBETA)
5640 SST=2.0*SST
5650 HI=1.0/H(JN,IT)
5660 GG1=1.0/G(JN,IT)

```



```

5670      DO 201 IS=1,3
5680      CR(IT,IS)=0.0
5690      C(IS)=SS*C(IS)
5700      DO 201 JS=1,3
5710      ODR(IT,IS,JS)=0.0
5720      B(IS,JS)=SST*B(IS,JS)*GGI
5730 201 AS(IS,JS)=HI*A(IS,JS)
5740      DET=AS(1,1)*AS(3,3)-AS(1,3)*AS(3,1)
5750      DETI=1.0/DET
5760      ODR(IT,1,2)= DETI*(AS(3,3)*B(1,2)-AS(1,3)*B(3,2))
5770      ODR(IT,3,2)=-DETI*(AS(3,1)*B(1,2)-AS(1,1)*B(3,2))
5780      CR(IT,1)=XV(IT,1)+DETI*(AS(3,3)*C(1)-AS(1,3)*C(3))
5790      CR(IT,2)=XV(IT,2)
5800      CR(IT,3)=XV(IT,3)-DETI*(AS(3,1)*C(1)-AS(1,1)*C(3))
5810      500 RETURN
5820      300 WRITE(6,401)
5830      WRITE(6,402) JN,MT,DELTAT,H( JN,MT),GG( JN,MT)
5840      STOP
5850      400 FORMAT(10X,5E15.7)
5860      401 FORMAT(5X, 30H DELTAT,H. OR GG ARE ZERO.
5870      402 FORMAT(5X,2I5,3E15.7)
5880      403 FORMAT (5X,3E15.6)
5890      END

```

```

5900      SUBROUTINE FSYPLN
5910 C      COMPUTES THE COEFFICIENTS OF THE LINEAR EQUATIONS RESULTING FROM
5920 C      THE IMPLICIT FORMULATION OF THE PDE ON THE LINE OF SYMMETRY
5930 C      THATA= 0.0
5940      COMMON/INT/NTH,MTH,NS,DELTAI
5950      COMMON/COOD/DDL,D,DDR,CR
5960      COMMON/VAR/XV
5970      COMMON/COEFC/PC,KL
5980      COMMON/DATAIN/ TH,UP,UT,S,UPLP,UILP,UPLT,UTLT
5990      COMMON/GEOIN/KP,KT,H,F,GG
6000      COMMON/CABC/      A,B,C
6010      COMMON/CINPUT/ULP,VLP,ULT ,VLT,UM,XKP,XKT,UMLI,UMLP
6020      COMMON/JON/JO,JN
6030      REAL KP,KT
6040      DIMENSION XV(150,3)
6050      DIMENSION PC(2,4)
6060      DIMENSION UP(2,150),UT(2,150),TH(2,150)
6070      DIMENSION S(2,150),UPLP(2,150),UILP(2,150),UPLT(2,150),UTLT(2,150)
6080      DIMENSION KP (2,150),KT (2,150),H(2,150),F(2,150),GG(2,150)
6090      DIMENSION D(150,3,3),DDL(150,3,3),DDR(150,3,3),CR(150,3)
6100      DIMENSION A(3,3),B(3,3),C(3)
6110      DIMENSION AS(5,3), ASL(5,3)
6120      I1=NTH
6130      M=I1
6140      ALPHA=ARAB/DELTAI*(JN,I1),UP(JN,I1)
6150      UM=UT(JN,I1)+UT(JN,I1)+UP(JN,I1)+UP(JN,I1)
6160      UM=SQRT(UM)
6170      SA=UT(JN,I1)/UM
6180      CA=UP(JN,I1)/UM
6190      ULP=UPLP(JN,I1)
6200      VLP=UILP(JN,I1)
6210      ULT=UPLT(JN,I1)
6220      VLT=UTLT(JN,I1)
6230      XKP=KP(JN,I1)
6240      XKT=KT(JN,I1)
6250      DMLI=CA*H(1,3)+S*VLT
6260      UMLP=CA*DLI
6270      CALL COEFF(XV(1,3),SA,CA,XV(1,2))
6280      IF(H(JN,I1).LT.0.000001) GO TO 300
6290      IF(GG(JN,I1).LT.0.000001) GO TO 300
6300      SS=S(JN,I1)-S(JD,I1)
6310      SST=0.5*SS/DELTAI
6320      SST=2.0*SST
6330      HI=1.0/H(JN,I1)
6340      GG1=1.0/GG(JN,I1)

```

```

6350      DO 201 IS=1,3
6360      CR(IT,IS)=0.0
6370      C(IS)=SS*C(IS)
6380      DO 201 JS=1,3
6390      ODR(IT,IS,JS)=0.0
6400      B(IS,JS)=SST*B(IS,JS)*GGI
6410 201 AS(IS,JS)=HI*A(IS,JS)
6420      DET=AS(1,1)*AS(3,3)-AS(1,3)*AS(3,1)
6430      DETI=1.0/DET
6440      ODR(IT,1,2)= DETI*(AS(3,3)*B(1,2)-AS(1,3)*B(3,2))
6450      ODR(IT,3,2)=-DETI*(AS(3,1)*B(1,2)-AS(1,1)*B(3,2))
6460      CR(IT,1)=XV(IT,1)+DETI*(AS(3,3)*C(1)-AS(1,3)*C(3))
6470      CR(IT,2)=XV(1,2)
6480      CR(IT,3)=XV(IT,3)-DETI*(AS(3,1)*C(1)-AS(1,1)*C(3))
6490      500 RETURN
6500      300 WRITE(6,401)
6510      WRITE(6,402) UN,MT,DELTA,H, UN,MT,GG,UN,P,
6520      STOP
6530      400 FORMAT(10X,5E15.7)
6540      401 FORMAT(5X, 50H DELTA,H, OR GG ARE ZERO.
6550      402 FORMAT(5X,2I5,3E15.7)
6560      403 FORMAT(5X,3E15.6)
6570      END

```

```

6580      SUBROUTINE FS(H)
6590
6600 C      ALL FUNCTIONS COMPUTED HERE ARE A FUNCTION OF H
6610 C      FB(J) ARE THE FUNCTIONS F BAR SUBSCRIP J
6620 C      FBP AND FBPP ARE THE FIRST AND SECOND DERIVATIVES OF FB
6630 C      GPH IS G+H WHERE G=2*H/(H-1)
6640
6650      COMMON/CF/FB
6660      COMMON/CFD/FBP,FBPP
6670      DIMENSION FB(4),FBP(4),FBPP(4),F(4),FP(4),FPP(4)
6680      HM1= H-1.0
6690
6700 C      THE FOLLOWING IF STATEMENT DETERMINES IF THERE IS A SINGULARITY IN
6710 C      ONE OF THE FB FUNCTIONS.
6720
6730      IF(HM1) 501,501,101
6740      101 HM11=1.0/HM1
6750      HM115=HM11*HM11
6760      100 HI=1.0/H
6770      HIS=HI*HI
6780      HS=H*H
6790      HP1=H+1.0
6800      102 HP11=1.0/HP1
6810      HP115=HP11*HP11
6820      HP2=H+2.0
6830      103 HP21=1.0/HP2
6840      HP215=HP21*HP21
6850      HP3=H+3.0
6860      104 HP31= 1.0/HP3
6870      HP315=HP31*HP31
6880      HP4=H+4.0
6890      105 HP41=1.0/HP4
6900      HP415=HP41*HP41
6910      HP5=H+5.0
6920      106 HP51=1.0/HP5
6930      HP515=HP51*HP51
6940
6950 C      COMPUTE GPH AND ITS DERIVATIVES
6960
6970      GPH=H*HP11*HM11
6980      P=HS-2.0*H-1.0
6990      GHP =P*HM115
7000
7010 C      COMPUTE F AND ITS FIRST DERIVATIVE FP
7020

```

```

7030      F(1)=HI-2.0*HP1I+HP2I
7040      FP(1)=- (HIS-2.0*HP1IS+HP2IS)
7050      F(2)=HI-4.0*HP1I+HP2I+4.0*HP3I-2.0*HP5I
7060      FP(2)=- (HIS-4.0*HP1IS+HP2IS+4.0*HP3IS-2.0*HP5IS)
7070      F(3)= (HI-4.0*HP1I+6.0*HP2I-4.0*HP3I+HP4I)
7080      FP(3)=- (HIS-4.0*HP1IS+6.0*HP2IS-4.0*HP3IS+HP4IS)
7090      F(4)=2.0*(HP1I-2.0*HP3I+HP5I)
7100      FP(4)=-2.0*(HP1IS-2.0*HP3IS+HP5IS)
7110      F(4)=-F(4)
7120      FP(4)=-FP(4)
7130
7140 C    COMPUTE FB,FBP
7150
7160      DO 200 I=1,4
7170      FB(I)=-F(I)*GPH
7180      200 FBP(I)=- (FP(I)*GPH+F(I)*GPHF)
7190      500 RETURN
7200
7210      501 WRITE(6,400) H
7220      STOP
7230      400 FORMAT(44H1SINGULARITY IN ONE OF THE F FUNCTIONS
7240      13H2H= 1F7.4      )
7250      END

```

```

7260      SUBROUTINE SOLVE
7270 C      THIS SUBROUTINE SOLVES THE LINEAR EQUATIONS RESULTING FROM THE
7280 C      O BRIEN, HYMAN, AND KAPLAN IMPLICIT FORMULATION. A DIRECT GAUSSIAN
7290 C      REDUCTION IS APPLIED TO THE SPARSE COEFFICIENT MATRIX.
7300 C      THE COEFFICIENT MATRIX HAS THE TRI-DIAGONAL FORM
7310 C      (ODL(I), D(I), ODR(I)) WHERE I STANDS FOR THE I-TH ROW AND ODL(I),
7320 C      D(I), AND ODR(I) ARE 3*3 MATRICES.
7330 C      THE SOLUTION IS XV.
7340      COMMON/INT/NTH, MTH, MS, DELTAT
7350      COMMON/VAR/XV
7360      COMMON/CDOD/ODL, D, ODR, CR
7370      COMMON/TOLERN/TOL
7380      DIMENSION D(150, 3, 3), ODL(150, 3, 3), ODR(150, 3, 3), CR(150, 3)
7390      DIMENSION XV(150, 3)
7400      DIMENSION A(3, 3), AI(3, 3), B(3, 3), UM(3, 3), Y(3)
7410      WRITE(6, 405) ((ODR(3, I, J), J=1, 3), CR(3, I), XV(3, I), I=1, 3)
7420 C
7430 C      GAUSSIAN REDUCTION BEGINS
7440      DO 205 II=1, NTH
7450          DO 205 I=1, 3
7460      205  XV(II, I)=CR(II, I)
7470 C      AT THIS POINT XV IS RIGHT SIDE OF EQUATION.
7480      DO 206 I=1, 3
7490      DO 206 J=1, 3
7500          UM(I, J)=0.0
7510          IF(I.EQ.J) UM(I, J)=1.0
7520          D(I, I, J)=UM(I, J)
7530          ODL(I, I, J)=ODL(I, I, J)
7540      206  CONTINUE
7550          DO 209 II=2, NTH
7560              MII=II
7570              DO 201 I=1, 3
7580              DO 207 J=1, 3
7590                  D(II, I, J)=UM(I, J)
7600                  ODL(II, I, J)=ODR(II, I, J)
7610                  DO 204 K=1, 3
7620      204  D(II, I, J)=D(II, I, J)+ODR(II, I, K)*ODR(II-1, K, J)
7630      207  A(I, J)=D(II, I, J)
7640                  DO 201 K=1, 3
7650                      XV(II, I)=XV(II, I)+ODR(II, I, K)*XV(II-1, K)
7660      201  CONTINUE
7670 C      INVERS COMPUTES THE INVERSE OF THE 3*3 MATRIX A.
7680      CALL INVERS(A, AI, TOL)
7690      IF(II.NE.3) GO TO 300
7700      WRITE(6, 404) A

```

```

7710      WRITE(6,404) AI
7720      WRITE(6,404) (XV(3,I),I=1,3)
7730      300 CONTINUE
7740      DO 202 I=1,3
7750      DO 203 J=1,3
7760      A(I,J)=0.0
7770      B(I,J)=0.0
7780      Y(I)=0.0
7790      DO 203 K=1,3
7800      R(I,J)=A(I,J)+AI(I,K)*D(II,K,J)
7810      203 B(I,J)=B(I,J)+AI(I,K)*DDR(II,K,J)
7820      DO 202 K=1,3
7830      202 Y(I)=Y(I)+AI(I,K)*XV(II,K)
7840      DO 208 I=1,3
7850      YAV=ABS(Y(I))
7860      IF(YAV.LT.1.0E-12) Y(I)=0.0
7870      XV(II,I)=Y(I)
7880      DO 208 J=1,3
7890      BAV=ABS(A(I,J))
7900      IF(BAV.LT.1.0E-12) A(I,J)=0.0
7910      BAV=ABS(B(I,J))
7920      IF(BAV.LT.1.0E-12) B(I,J)=0.0
7930      D(II,I,J)=A(I,J)
7940      DDR(II,I,J)=B(I,J)
7950      ABV=ABS(A(I,J)-B(I,J))
7960      IF(ABV.GE.1.0E-12) GOTO 502
7970      208 CONTINUE
7980      200 CONTINUE
7990      C
8000      C      THE GAUSSIAN REDUCTION IS COMPLETE. ANY ELEMENT LESS THAN .0001
8010      C      HAS BEEN REPLACED BY 0.0.
8020      C
8030      C      BEGIN BACK SUBSTITUTION.
8040      C
8050      DO 220 IB=1,MTN
8060      L=NTN-IB
8070      DO 221 I=1,3
8080      DO 221 K=1,3
8090      221 XV(L,I)=XV(L,I)-DDR(L,I,K)*XV(L+1,K)
8100      220 CONTINUE
8110      C
8120      C      SOLUTION COMPLETED.
8130      C      XV IS NOW THE SOLUTION.
8140      C
8150      C      CHECK SOLUTION

```

```

8160 C
8170      DO 211 II=1,NTH
8180 211 CONTINUE
8190      MII=1
8200      DO 209 I=1,3
8210          Y(I)=XV(1,I)
8220          DO 210 K=1,3
8230 210 Y(I)=Y(I)-OBL(1,I,K)*XV(2,K)
8240          ABSV=ABS(Y(I)-CR(1,I))
8250          IF(ABSV.GE.1.0E-05) GOTO 503
8260 209 CONTINUE
8270      DO 212 II=2,MTH
8280          MII=II
8290          DO 213 I=1,3
8300              Y(I)=XV(II,I)
8310              DO 213 K=1,3
8320 213 Y(I)=Y(I)+OBL(II,I,K)*XV(II-1,K)-OBL(II,I,K)*XV(II+1,K)
8330              ABSV=ABS(Y(I)-CR(II,I))
8340              IF(ABSV.GE.1.0E-05) GOTO 503
8350 212 CONTINUE
8360          MII=NTH
8370          DO 214 I=1,3
8380              Y(I)=XV(NTH,I)
8390              DO 215 K=1,3
8400 215 Y(I)=Y(I)+OBL(NTH,I,K)*XV(MTH,K)
8410              ABSV=ABS(Y(I)-CR(NTH,I))
8420              IF(ABSV.GE.1.0E-05) GOTO 503
8430 214 CONTINUE
8440 500 RETURN
8450 501 THATA=(MII-1)*DELTA
8460          NS=MII
8470          WRITE(6,400) NS,THATA
8480          WRITE(6,403)
8490          WRITE(6,407) ((OBL(MII,I,J),J=1,3),CR(MII,I),I=1,3)
8500          WRITE(6,406) ((OBL(MII,I,J),J=1,3),(OBL(MII,I,J),J=1,3),I=1,3)
8510          STOP
8520 502 THATA=(MII-1)*DELTA
8530          NS=MII
8540          WRITE(6,401) NS,THATA
8550          STOP
8560 503 THATA=(MII-1)*DELTA
8570          NS=MII
8580          WRITE(6,402) NS,THATA
8590          WRITE(6,404) ((CR(MII,I),I=1,3)
8600          STOP

```



```

8610 C      NOSOL MEANS NO SOLUTION WAS OBTAINED FOR U WAS EITHER NOT THE
8620 C      IDENTITY MATRIX OR A SUBSTITUTION OF THE SOLUTION DID NOT
8630 C      YIELD THE CORRECT RIGHT SIDE.
8640 400 FORMAT (33H2 MATRIX IS SINGULAR AT STATION =      ,1I5,5X,8H THATA =
8650      $,1F10.4)
8660 401 FORMAT( 49H2NO SOLUTION FOR U IS NOT IDENTITY AT STATION=      ,
8670      $1I5,5X,7H THATA=      ,1F10.4)
8680 402 FORMAT ( 56H2NO SOLUTION FOR Y IS NOT EQUAL RIGHT SIDE AT STATION
8690      $=      ,1I5,5X,7H THATA=      ,1F10.4)
8700 404 FORMAT(///(3E15.7))
8710 405 FORMAT(5E15.7)
8720 406 FORMAT(6E15.7)
8730 407 FORMAT(5X,4E15.7)
8740 408 FORMAT(5X,45H FROM SOLVE U, OR, UDL, AND UDR DUCK LOGS
8750      END

```

DTNSRDC ISSUES THREE TYPES OF REPORTS

1. DTNSRDC REPORTS, A FORMAL SERIES, CONTAIN INFORMATION OF PERMANENT TECHNICAL VALUE. THEY CARRY A CONSECUTIVE NUMERICAL IDENTIFICATION REGARDLESS OF THEIR CLASSIFICATION OR THE ORIGINATING DEPARTMENT.

2. DEPARTMENTAL REPORTS, A SEMIFORMAL SERIES, CONTAIN INFORMATION OF A PRELIMINARY, TEMPORARY, OR PROPRIETARY NATURE OR OF LIMITED INTEREST OR SIGNIFICANCE. THEY CARRY A DEPARTMENTAL ALPHANUMERICAL IDENTIFICATION.

3. TECHNICAL MEMORANDA, AN INFORMAL SERIES, CONTAIN TECHNICAL DOCUMENTATION OF LIMITED USE AND INTEREST. THEY ARE PRIMARILY WORKING PAPERS INTENDED FOR INTERNAL USE. THEY CARRY AN IDENTIFYING NUMBER WHICH INDICATES THEIR TYPE AND THE NUMERICAL CODE OF THE ORIGINATING DEPARTMENT. ANY DISTRIBUTION OUTSIDE DTNSRDC MUST BE APPROVED BY THE HEAD OF THE ORIGINATING DEPARTMENT ON A CASE-BY-CASE BASIS.

